Santa Barbara Tiny BASIC for 6809's

by James A. Hinds

DDJ is pleased to present this version of Tiny BASIC for the 6809. It is a direct descendant of Li-Chen Wang's Palo Alto Tiny BASIC (refer to DDJ #5) which its author, James A. Hinds, has named Santa Barbara Tiny BASIC.

SBTB is an example of a transliteration of a large 8080 program, and contains a number of 6809 techniques which should be useful to those who want to get familiar with the 6809. The program was developed on an Apple II with The Mill (which puts the 6809 under the hood) and Technical Systems Consultants' operating system, FLEX.

The following introductory paragraphs are the same as those printed with the original Palo Alto Tiny BASIC. The few differences in documentation can be noted as follows: in this version, there is no Control O function; NEXT statements do not require a variable name; there is no overprint function; and no spaces are allowed in the input language except in GO TO commands.

In addition, this program is completely relocatable on a page basis to any position in memory. It requires about the same amount of space as did Li-Chen Wang's PATB. – Ed.

Original Introduction to Palo Alto Tiny BASIC by Li-Chen Wang

Numbers. All numbers are integers and must be less than 32767.

Variables. There are 26 variables denoted by letters A through Z. There is also a single array @(I). The dimension of this array is set automatically to make use of all the memory space that is left unused by the program. (I.e., 0 through SIZE/2, see SIZE function below.)

Functions. There are 3 functions:
ABS(X) gives the absolute value of X.
RND(X) gives a random number between 1 and X (inclusive).
SIZE gives the number of bytes left unused by the program.

Arithmetic and Compare Operators:

/	divide.
*	multiply.
-	subtract.
+	add.
>	greater than (compare).
<	less than (compare).
=	equal to (compare).
#	not equal to (compare).
>=	greater than or equal to (compare).
<=	less than or equal to (compare).

+, -, *, and / operations result in a value between -32767 and 32767. (-32768 is also allowed in some cases.) All compare operators result in a 1 if true and a 0 if not true.

Expressions. Expressions are formed with numbers, variables, and functions with arithmetic and compare operators between them. + and - signs can also be used at the beginning of an expression. The value of an expression is evaluated from left to right, except that * and / are always done first, and then + and -, and then compare operators. Parentheses can also be used to alter the order of evaluation. Note that compare operators can be used in any expression. For example:

```
10 LET A= (X>Y)*123+(X=Y)*456+(X<Y)*789
20 IF (U=1)*(V<2)+(U>V)*(U<99)*(V>3) PRINT
   "YES"
30 LET R=RND(100), A=(R>3)+(R>15)+(R>56)+
   (R>98)
```

In statement 10, A will be set to 123 if X>Y, to 456 if X=Y, and to 789 if X<Y. In statement 20, the "*" operator acts like a logical AND, and the "+" operator acts like a logical OR. In statement 30, Y will be a random number between 0 and 4 with a prescribed probability distribution of: 3% of being 0, 15-3=12% of being 1, 56-15=41% of being 2, 98-56=42% of being 3, and 100-98=2% of being 4.

Direct Commands. All the commands described later can be used as direct commands except the following three; they can only be used as direct commands and not as part of a statement:

```
RUN
```

will start to execute the program starting at the lowest statement number.

```
LIST
```

will print out all the statements in numerical order.

```
LIST 120
```

will print out all the statements in numerical order starting at statement 120.

```
NEW
```

will delete all statements.

Abbreviation and blanks. You may use blanks freely, except that numbers, command keywords, and function names can not have embedded blanks.

You may truncate all command keywords and function names and follow them by a period. "P.", "PR.", "PRI.", and "PRIN." all stand for "PRINT". Also the word LET in LET command can be omitted. The "shortest" abbreviation for all keywords are as follows:

A.=ABS	F.=FOR	GOS.=GOSUB	G.=GOTO
IF=IF	IN.=INPUT	L.=LIST	N.=NEW
N.=NEXT	P.=PRINT	REM=REMARK	R.=RETURN
R.=RND	R.=RUN	S.=SIZE	S.=STEP
S.=STOP	TO=TO		
Implied=LET			

Statements. A statement consists of a statement number of betweeen 1 and 32767 followed by one or more commands. Commands in the same statement are separated by a semi-colon ";". "GOTO", "STOP", and "RETURN" commands must be the last command in any given statement.

Commands. Tiny BASIC commands are listed below with examples. Remember that commands can be concatenated with semi-colons. In order to store the statement, you must also have a statement number in front of the commands. The statement number and the concatenation are not shown in the examples.

REM or REMARK Command

```
REM anything goes
```

This line will be ignored by TBI.

LET Command

```
LET A=234-5*6, A=A/2, X=A-100, @(X+9)=A-1
```

will set the variable A to the value of the expression 234-5*6 (i.e., 204), set the variable A (again) to the value of the expression A/2 (i.e., 102), set the variable X to the value of the expression A-100 (i.e., 2), and then set the variable @(11) to 101 (where 11 is the value of the expression X+9 and 101 is the value of the expression A-1).

```
LET U=A#B, V=(A>B)*X+(A<B)*Y
```

will set the variable U to either 1 or 0 depending on whether A is not equal to or is equal to B; and set the variable V to either X, Y or 0 depending on whether A is greater than, less than, or equal to B.

PRINT Command

PRINT

will cause a carriage-return (CR) and a line-feed (LF) on the output device.

PRINT A*3+1, "ABC 123 !@#", ' CBA '

will print the value of the expression A*3+1 (i.e., 307), the string of characters "ABC 123 !@#", and the string " CBA ", and then a CR-LF. Note that either single or double quotes can be used to quote strings, but pairs must be matched.

PRINT A*3+1, "ABC 123 !@#", ' CBA ',

will produce the same output as before, except that there is no CR-LF after the last item is printed. This enables the program to continue printing on the same line with another "PRINT".

PRINT A, B, #3, C, D, E, #10, F, G

will print the values of A and B in six spaces, the values of C, D, an E in 3 spaces, and the values of F and G in 10 spaces. If there are not enough spaces specified for a given value to be printed, the value will be printed with enough spaces anyway.

PRINT 'ABC',-,'XXX'

will print the string "ABC", a CR without a LF, and then the string "XXX" (over ABC) followed by a CR-LF.

INPUT Command

INPUT A, B

When this command is executed, Tiny BASIC will print "A:" and wait to read in an expression from the input device. The variable A will be set to the value of this expression. Then "B:" is printed and variable B is set to the value of the next expression read from the input device. Note that not only numbers, but also expressions can be read as input.

INPUT 'WHAT IS THE WEIGHT'A, "AND SIZE"B

This is the same as the command above, except the prompt "A:" is replaced by "WHAT IS THE WEIGHT:" and the prompt "B:" is replaced by "AND SIZE:". Again, both single and double quotes can be used as long as they are matched.

INPUT A, 'STRING', -, "ANOTHER STRING", B

The strings and the "-" have the same effect as in "PRINT".

IF Command

IF A<B LET X=3; PRINT 'THIS STRING'

will test the value of the expression A<B. If it is not zero (i.e., if it is true), the commands in the rest of this statement will be executed. If the value of the expression is zero (i.e., if it is not true), the rest of this statement will be skipped over and execution continues at next statement. Note that the word "THEN" is not used.

GOTO Command

GOTO 120

will cause the execution to jump to statement 120. Note that GOTO command cannot be followed by a semi-colon and other commands. It must be ended with a CR.

GOTO A*10+B

will cause the execution to jump to a different statement number as computed from the value of the expression.

GOSUB and RETURN Commands

GOSUB command is similar to GOTO command except that: a) the current statement number and position within the statement is remembered; and b) a semi-colon and other commands can follow it in the same statement.

GOSUB 120

will cause the execution to jump to statement 120.

GOSUB A*10+B

will cause the execution to jump to different statements as computed from the value of the expression A*10+B.

RETURN

A RETURN command must be the last command in a statement and followed by a CR. When a RETURN command is encountered, it will cause the execution to jump back to the command following the most recent GOSUB command.

GOSUB can be nested. The depth of nesting is limited only by the stack space.

FOR and NEXT Commands

FOR X=A+1 TO 3*B STEP C-1

The variable X is set to the value of the expression A+1. The values of the expressions (not the expressions themselves) 3*B and C-1 are remembered. The name of the variable X, the statement number and the position of this command within the statement are also remembered. Execution then continues the normal way until a NEXT command is encountered.

The STEP can be positive, negative or even zero. The word STEP and the expression following it can be omitted if the desired STEP is +1.

NEXT X

The name of the variable (X) is checked with that of the most recent FOR command. If they do not agree, that FOR is terminated and the next recent FOR is checked, etc. When a match is found, this variable will be set to its current value plus the value of the STEP expression saved by the FOR command. The updated value is then compared with the value of the TO expression also saved by the FOR command. If this is within the limit, execution will jump back to the command following the FOR command. If this is outside the limit, execution continues following the NEXT command itself.

FOR can be nested. The depth of nesting is limited only by the stack space. If a new FOR command with the same control variable as that of an old FOR command is encountered, the old FOR will be terminated automatically.

STOP Command

This command stops the execution of the program and returns control to direct commands from the input device. It can appear many times in a program but must be the last command in any given statement; i.e., it cannot be followed by a semicolon and other commands.

Stopping the Execution. By the Control-C key on the input device.

Control of Output Device. The Control-O key on the input device can be used to turn the output device ON and OFF. This is useful when you want to read in a program punched on paper tape.

To produce such a paper tape, type "LIST" without CR. Turn on the paper tape punch and type a few Control-Shift-P's and then a CR. When listing is finished, type more Control-Shift-P's and turn off the punch.

To read back such a paper tape, type "NEW", CR, and Control-O, then turn on the paper tape reader. When the paper tape is read, turn the reader off and type a Control-O again.

Error Report. There are only three error conditions in Tiny

(Continued on page 45)

LISTING ON PAGE 24

S. B. Tiny Basic for 6809's

```
                    * PALO ALTO TINY BASIC BY LI-CHEN WANG
                    *
                    *CONVERTED FOR  'THE MILL' OF STELLATION TWO
                    *  BY JAMES A. HINDS
                    *
                    ********************************
                    RATS     MACRO
                    RETURN   SET     *
                             RTS
                             ENDM
                    *
                    HERE     MACRO
                    TMP      SET     *
                             ORG     TABLE
                             IFC     &1,,1
                             FCC     &1
                             FDB     TMP+$8000
                             IFC     'GOTO',&1
                             FCC     "GO TO"
                             FDB     TMP+$8000
                             ENDIF
                    TABLE    SET     *
                             ORG     TMP
                             ENDM
                    *
                    DISPAT   MACRO
                             JSR     <EXEC
                             FDB     TABLE
                             ENDM
                    *
                    TSTC     MACRO
                             JSR     <IGNBLK
                             CMPA    #&1
                             BNE     &2
                             LEAY    1,Y
                             ENDM
                    * * * * * * * * * * * * * * * *
                    * VARIABLE AREA AT U
0000                         ORG     0           OFFSET FROM U
                FFCA VARBGN  EQU     -27*2       VARIABLE SPACE
0000                OCSW     RMB     1           OUTPUT ON/OFF FLAG
0001                FLDCNT   RMB     1           PLACES FOR NUMERIC OUTPUT
0002                CURRNT   RMB     2           POINTS TO CURR LINE
0004                TXTUNF   RMB     2           -> TO LAST OF PRG
0006                STKGOS   RMB     0           SAVE OF S IN GOSUB
                0006 VARNXT  EQU     *
0006                STKINP   RMB     2           SAVE S DURING INPUT
0008                LOPVAR   RMB     2           FOR LOOP INDEX
000A                LOPINC   RMB     2           FOR LOOP STEP
000C                LOPLMT   RMB     2           LIMIT
000E                LOPLN    RMB     2           LOOP BEGINNING LINE
0010                LOPPT    RMB     2           LOOP PROG RESTART
0012                SEED     RMB     2
0014                HIWAT    RMB     2           END OF @ ARRAY
0016                         RMB     2           EXTRA FOR BUFFER
0018                BUFFER   RMB     72
                0060 BUFEND  EQU     *
                    ****************
                    * PAGE ZERO SUBROUTINES OF 8080
                    * ARE REPLACED ON 6809 BY DIRECT PAGE
                    * SUBROUTINES AND CAN BE ACCESSED
                    * BY 2 BYTE JSR
                    * DIRECT PAGE IS PAGE 1 ($100-$1FF)
```

```
                        ********************************
                        *  ENDCHK IS USED TO VERIFY
                        *  A COMMAND IS TERMINATED BY
                        *    A CR BEFORE PERFORMING
                        *    THE COMMAND.
0153                    ENDCHK  TSTC    $D,QWHAT
015B                            RATS
                        ********************************
                        *  FINE  USED TO CONTINUE EXECUTION
                        *     WITH NEXT STATEMENT. IF CURRENT
                        *     STATEMENT IS NOT AT END
                        *     A 'WHAT' ERROR IS SIGNALLED
015C 8D   DD            FINE    BSR     FIN         IF ERROR THEN FALL THROUGH
                        ********************************
015E 17   0465          QWHAT   LBSR    ERROR
0161 57 48 41 54                FCC     'WHAT?',13
0165 3F 0D
0167 17   045C          QSORRY  LBSR    ERROR
016A 53 4F 52 52                FCC     'SORRY',13
016E 59 0D
                        ********************************
                        *   IGNBLK     SCANS PAST ANY
                        *      SPACES AND RETURNS FIRST
                        *      CHAR WITH PROG PTR UPDATED
                        *
0170 31   21            I1      LEAY    1,Y
0172 A6   A4            IGNBLK  LDA     0,Y
0174 81   20                    CMPA    #'          A BLANK
0176 27   F8                    BEQ     I1          GET NEXT
0178                            RATS
                        ********************************
                        *  OUTPUT ROUTINES
                        *      SPACE      PUTS ONE BLANK
                        *      CRLF       PUTS CR
                        *      OUTC       PUTS CHAR IN MAILBOX
                        *        FOR 6502 TO GRAB AND
                        *        CLEAR MAILBOX
0179 86   20            SPACE   LDA     #'          A SPACE
017B 20   02                    BRA     OUTC
017D 86   0D            CRLF    LDA     #13
017F 1E   88            OUTC    EXG     A,A         GIVE 6502 SOME EXTRA CYCLES
0181 7D   FBFF                  TST     $FBFF


0184 26   F9                    BNE     OUTC
0186 B7   FBFF                  STA     $FBFF
0189                            RATS
                        ********************************
                        *  INPUT ROUTINE
                        *     CHKIO  RETURN ZERO IF NO
                        *        CHAR, OTHERWISE GRAB IT
                        *        RESTART BASIC IF IT IS
                        *        CONTROL-C
018A B6   FBFE          CHKIO   LDA     $FBFE       CHAR LOC
018D 27   FA                    BEQ     RETURN      IF NO CHAR JUST SAY SO
018F 7F   FBFE                  CLR     $FBFE       OTHERWISE CLEAR IT
0192 81   03                    CMPA    #$03        IS IT CNTL C
0194 26   F3                    BNE     RETURN      NO RETURN CHAR
                        ********************************
                        *   START  RELAY TO FIRST OF BASIC
                        *
0196 16   04AB          START   LBRA    ST1
```

```
                        *  PAGE 0 CONTAINS TABLE OF COMMAND NAMES
                        * AND THEIR CODE LOCATION EQUIV LOCATIONS
                        * ENTRIES INTO THIS TABLE IS BY
                        * MACRO "HERE"
0000                            ORG     0
0000 16   0193                  LBRA    START
              0003      TABLE   SET     *
              0000      TMP     SET     0
0100                            ORG     $100        ;LEAVE LOTS OF SPACE FOR TABLE
              0001              SETDP   *>>8
                        ******************************
                        * TSTNUM  RETURNS VALUE AND
                        *    LENGTH OF ASCII CODED INTEGER
                        *    IN TEXT. A NON-NUMBER WILL HAVE
                        *    A LENGTH OF ZERO.
0100 5F                 TSTNUM  CLRB
0101 4F                         CLRA
0102 1F   01                    TFR     D,X         CLEAR X AND ACCUM
0104 34   16                    PSHS    D,X         TOS= NUM AND NOS=#DIGITS
0106 9D   72            TN1     JSR     <IGNBLK
0108 80   30                    SUBA    #'0
010A 25   28                    BLO     NUMOUT      CHAR TOO LOW
010C 81   09                    CMPA    #9
010E 22   24                    BHI     NUMOUT      HIT CHAR THAT ISNT DIGIT
0110 31   21                    LEAY    1,Y         BUMP TO NEXT CHAR
0112 34   02                    PSHS    A           SAVE CHAR
0114 6C   64                    INC     4,S         INCREMENT DIGIT COUNT
0116 EC   61                    LDD     1,S         GET ACCUMULATOR
0118 81   0F                    CMPA    #$0F        CHECK IF ROOM
011A 22   10                    BHI     QHOW        TOO BIG TO MULT BY 10
011C 58                         ASLB
011D 49                         ROLA                DOUBLE
011E 58                         ASLB
011F 49                         ROLA                TIMES 4
0120 E3   61                    ADDD    1,S         TIMES 5
0122 58                         ASLB
0123 49                         ROLA                TIMES TEN
0124 EB   E0                    ADDB    ,S+         ADD IN DIGIT
0126 89   00                    ADCA    #0
0128 ED   E4                    STD     0,S         STORE ACCUM BACK
012A 2A   DA                    BPL     TN1         NO ERR? THEN GET NEXT DIGIT
012C 17   0497          QHOW    LBSR    ERROR
012F 48 4F 57 3F                FCC     'HOW?',13
0133 0D
0134 6D   63            NUMOUT  TST     3,S         SET FLAGS ON # DIGITS
0136 35   96                    PULS    D,X,PC      AND GO AWAY
0138 29   F2            TVHOW   BVS     QHOW        DIRECT OVERFLOW TEST FOR ERROR
013A                            RATS

                        ********************************
                        *  FIN IS CALLED TO CONTINUE
                        *     EXECUTION OF NEXT STMT
                        *     ON SAME OR NEXT LINE.
                        *     HOWEVER, IF CURRENT STMT
                        *     HAS MORE TO PROCESS CONTROL
                        *     GOES BACK TO CALLER
013B                    FIN     TSTC    ':,FI1
0143 32   62                    LEAS    2,S         VOID RETURN
0145 16   01EA                  LBRA    RUNSML
0148 81   0D            FI1     CMPA    #$D
014A 26   EE                    BNE     RETURN
014C 31   21                    LEAY    1,Y
014E 32   62                    LEAS    2,S
0150 16   01D1                  LBRA    RUNNXL
```

(continued on the top of page 24)

```
                        **********************************
                        * TSTVE  RETURN ADDRESS AND
                        *      VALUE OF VARIABLE
0199 8D   D7            TSTVE   BSR     IGNBLK
019B 80   40                    SUBA    #'@
019D 25   EA                    BCS     RETURN
019F 27   0F                    BEQ     RLYAT       IS STRING SPACE
01A1 81   1B                    CMPA    #27         IS IN ALPHABET??
01A3 2C   0E                    BGE     TVN
01A5 31   21                    LEAY    1,Y
01A7 48                         ASLA
01A8 40                         NEGA
01A9 30   C6                    LEAX    A,U
01AB EC   84                    LDD     ,X
01AD 1C   FE                    CLC
01AF                            RATS    RETURN      WITH CARRY CLEAR
01B0 16   011C          RLYAT   LBRA    ATSIGN
01B3 1A   01            TVN     SEC                 SET CARRY
01B5                            RATS
                        **********************************
                        *   EXEC AND TABLE DISPATCH
                        *
                        *     SCAN TABLE FOR KEYWORD
                        *      JUMP TO ASSOCIATED CODE
01B6 9D   72            EXEC    JSR     <IGNBLK
01B8 AE   F4                    LDX     [0,S]       GET TABLE ADDRESS FROM CALLER
01BA 10AF E4                    STY     0,S
01BD A6   A0            LCT     LDA     0,Y+
01BF A1   80                    CMPA    0,X+        MATCH PRG AGAINST TABLE
01C1 27   FA                    BEQ     LCT
01C3 A6   1F                    LDA     -1,X        LOOK AT MISMATCH
01C5 2B   17                    BMI     EXECF       FOUND!
01C7 A6   3F                    LDA     -1,Y        IS IT AN ABBREV
01C9 81   2E                    CMPA    #'.
01CB 27   0B                    BEQ     EXECP       YES! FORCE MATCH
01CD 10AE E4                    LDY     0,S         RESTORE PROG PTR
01D0 A6   80            EXEC1   LDA     0,X+
01D2 2A   FC                    BPL     EXEC1       SKIP REST OF KEYWORD IN TABLE
01D4 A6   80                    LDA     0,X+
01D6 20   E5                    BRA     LCT         RETRY
01D8 A6   80            EXECP   LDA     ,X+
01DA 2A   FC                    BPL     EXECP
01DC 31   21                    LEAY    1,Y
01DE 31   3F            EXECF   LEAY    -1,Y        BACK UP PRG PTR
01E0 E6   84                    LDB     0,X         NEXT TRANSFER BYTE
01E2 84   7F                    ANDA    #$7F
01E4 30   8D FE18               LEAX    0,PCR
01E8 30   8B                    LEAX    D,X
01EA AF   E4                    STX     0,S
01EC 39                         RTS
                        **********************************
                        *
                        * EXPRESSION ANALYZER
01ED 8D   38            EXPR    BSR     EXPR2
01EF 34   10                    PSHS    X           SAVE LEFT HAND VALUE
01F1                            DISPAT
01F5                            HERE    '>='
01F5 8D   26                    BSR     XP18
01F7 2E   1F                    BGT     ZER         M>=R
01F9 20   1C                    BRA     ONE
                        *
01FB                            HERE    '='
01FB 8D   20                    BSR     XP18        M=R
01FD 27   18                    BEQ     ONE
```

(continued on the next page)

S. B. Tiny Basic for 6809's (Continued)

```
01FF 20   17              BRA     ZER
                  *
0201                      HERE    '>'
0201 8D   1A              BSR     XP18        M>R
0203 2D   12              BLT     ONE
0205 20   11              BRA     ZER
                  *
0207                      HERE    '<='
0207 8D   14              BSR     XP18        M<=R
0209 2C   0C              BGE     ONE
020B 20   0B              BRA     ZER
                  *
020D                      HERE    '='
020D 8D   0E              BSR     XP18
020F 27   06              BEQ     ONE
0211 20   05              BRA     ZER
                  *
0213                      HERE    '<'
0213 8D   08              BSR     XP18        M<R
0215 23   01              BLS     ZER
0217 5C           ONE     INCB
0218 ED   E4      ZER     STD     ,S          TO SET FLAGS
021A                      HERE    .
021A EC   E1              LDD     ,S++        POP AND SET FLAGS
021C                      RATS    NO          LOGICAL OP
                  *
021D 8D   08      XP18    BSR     EXPR2
021F AC   62              CMPX    2,S         COMPARE WITH LEFT
0221 34   01              PSHS    CC
0223 4F                   CLRA
0224 5F                   CLRB                SET D TO ZERO (FALSE)
0225 35   81              PULS    CC,PC
                  ********************************
                  *  EXPR2 LOOKS AFTER + AND -
                  *        BINARY OPERATORS
0227 8D   26      EXPR2   BSR     EXPR3       GO AFTER LEFT HAND OPERAND
0229 34   06      EXPR21  PSHS    D
022B 9D   72              JSR     <IGNBLK
022D 81   2B              CMPA    #'+
022F 26   0A              BNE     XP290
0231 31   21              LEAY    +1,Y
0233 8D   1A              BSR     EXPR3
0235 E3   E1              ADDD    ,S++        ADD IN LEFT
0237 28   F0      XP235   BVC     EXPR21      AND AGAIN
0239 0E   2C              JMP     QHOW        OVERFLOW
023B 81   2D      XP290   CMPA    #'-
023D 26   0E              BNE     XP291
023F 31   21              LEAY    1,Y
0241 8D   0C              BSR     EXPR3
0243 34   06              PSHS    D
0245 EC   62              LDD     2,S
0247 A3   E4              SUBD    0,S
0249 32   64              LEAS    4,S         POP GARBAGE
024B 20   EA              BRA     XP235       CHECK FOR OVERFLOW
                  *
024D 35   90      XP291   PULS    X,PC
                  ********************************
                  * EXPR3  HANDLES THINGS WITH
                  *          / AND * IN THEM
                  *
024F 8D   1E      EXPR3   BSR     EXPR4
0251 34   06              PSHS    D
```

```
02AD 0E   38              JMP    <TVHOW
                  ********************************
                  * CHECK FOR NUMBER, VARIABLE OR
                  *  SUBEXPRESSION
                  *
02AF                      HERE   ,
02AF 9D   99              JSR    TSTVE      NOT A FUNCTION MUST BE VAR
02B1 24   F3              BCC    RETURN
02B3 9D   00              JSR    <TSTNUM    NOT VAR TRY NUM
02B5 26   EF              BNE    RETURN     DIGIT CNT<>0 MEANS NUMBER
02B7                PARM  TSTC   '(,XP43
02BF 9D   ED              JSR    <EXPR
02C1 34   07              PSHS   CC,D       SAVE CONDITIONS OF EXPRESSION
02C3                      TSTC   '),XP43
02CB 35   87              PULS   CC,D,PC    FETCH CONDITIONS AND RETURN
02CD 0E   5E        XP43  JMP    QWHAT

                  ****************************
                  * GET INDEX INTO STRING SPACE
                  * IF WE GO PASSED THE CURRENT
                  * HIGH WATER MARK WE BUMP THAT
                  * TO KEEP STACK FROM GOING
                  * BANANAS
                  *
02CF 31   21        ATSIGN LEAY   1,Y
02D1 8D   E4              BSR    PARM       GET THE INDEX
02D3 102B FE55            LBMI   QHOW
02D7 AE   44              LDX    TXTUNF,U   GET END OF PROG
02D9 30   02              LEAX   2,X
02DB 30   8B              LEAX   D,X
02DD 30   8B              LEAX   D,X
02DF AC   C8 14           CMPX   HIWAT,U
02E2 23   03              BLS    HIWOK
02E4 AF   C8 14           STX    HIWAT,U
02E7 8D   88        HIWOK BSR    SIZE
02E9 102B FE7A            LBMI   QSORRY     NO ROOM
02ED 1C   FE              CLC
02EF EC   84              LDD    0,X        GET VALUE AT INDEX
02F1                      RATS
                  ****************************
                  * DIRECT STATEMENT EXECUTE
02F2 31   22        DIRECT LEAY   2,Y        STEP PAST LINE NUMBER
02F4                      DISPAT AND         GO TO IT!
                  ****************************
                  * LIST PROGRAM POSSIBLY FROM
                  *       A BIVEN LINE NUMBER, OTHERWISE
                  *       THE WHOLE SHEBANG
                  *
02F8                      HERE   'LIST'
02F8 9D   00              JSR    <TSTNUM    TEST IF A LN #
02FA 34   06              PSHS   D
02FC 9D   53              JSR    <ENDCHK    IF NO # THEN USE 0
02FE 35   06              PULS   D
0300 17   01D5            LBSR   FNDLN      FIND IT
0303 1022 FE8F      LS1   LBHI   START      HIT END?
0307 17   0278            LBSR   PRTLN      PRINT THE LINE
030A 9D   8A              JSR    <CHKIO     USER INTERRUPT?
030C 17   01D2            LBSR   FNDLNP     FIND NEXT LINE
030F 20   F2              BRA    LS1
                  *
0311                      HERE   'NEW'
0311 9D   53              JSR    <ENDCHK    ABORT IF NOT ONLY THING IN STMT
0313 30   8D 045F         LEAX   TXTBGN-1,PCR
```

```
0253                    EXP3EL  DISPAT
0257                            HERE    '$'
0257 8D   16                    BSR     EXPR4
0259 17   046F                  LBSR    MULT
025C 27   F5                    BEQ     EXP3EL    SIGNAL ERROR IF D REG HAS SIGNIFIC
ANT BITS
025E 0E   2C            EXP3ER  JMP     <QHOW
0260                            HERE    '/'
0260 8D   0D                    BSR     EXPR4
0262 ED   7E                    STD     -2,S      SET FLAGS
0264 27   F8                    BEQ     EXP3ER
0266 17   04C4                  LBSR    DIV
0269 ED   E4                    STD     0,S       REPLACE REM BY QOU
026B 20   E6                    BRA     EXP3EL
026D                            HERE    ,
026D 35   86                    PULS    D,PC      GET RESULT AND RETURN
                        ******************************
                        * EXPR4 HANDLES FUNCTIONS AND
                        *     UNARY MINUS AND REPRESENTS
                        *     THE HIGHEST PRIORITY
                        *     IF THERE ARE () THEN EXPR
                        *     IS CALLED (OH! NO!) RECURSIVLY
                        *
026F                    EXPR4   DISPAT ON         FUNCTIONS
0273                            HERE    'RND'
0273 EC   C8 12                 LDD     SEED,U
0276 34   06                    PSHS    D
0278 CC   0023                  LDD     #%100011
027B 17   044D                  LBSR    MULT
027E 35   06                    PULS    D
0280 C3   0123                  ADDD    #$0123
0283 84   7F                    ANDA    #$7F      MASK SIGN
0285 ED   C8 12                 STD     SEED,U
0288 34   06                    PSHS    D         SAVE FOR DIVISION
028A 8D   2B                    BSR     PARM
028C 102F FE9C                  LBLE    QHOW      IF ZERO OR NEG
0290 17   049A                  LBSR    DIV
0293 35   86                    PULS    D,PC      GET REMAINDER AND RETURN
                        ******************************
                        *  ABSOLUTE VALUE
                        *
0295                            HERE    'ABS'
0295 8D   20                    BSR     PARM      GET THE VALUE
0297 2A   83                    BPL     RETURN
0299 34   06                    PSHS    D
029B 4F                         CLRA
029C 5F                         CLRB
029D A3   E1                    SUBD    S++
029F 0E   38                    JMP     TVHOW     IMPLIED RETURN IF NO OVERFLOW
                        ******************************
                        *  SIZE  HOW MUCH STORAGE LEFT
                        *     BETWEEN STACK AND END OF
                        *     PROGRAM
02A1                            HERE    'SIZE'
02A1 1F   40            SIZE    TFR     S,D
02A3 A3   C8 14                 SUBD    HIWAT,U   SUBTRACT END OF STRING ADDR
02A6                            RATS
                        ******************************
                        * UNARY MINUS HANDLED AS FNCT
02A7                            HERE    '-'       UNARY MINUS
02A7 9D   ED                    JSR     <EXPR
02A9 43                         COMA
02AA 50                         NEGB
02AB 82   00                    SBCA    #0
```

(continued on the top of page 26)

```
0317 AF   44              STX    TXTUNF,U  RESET BEGINNING OF PROGRAM PTR
0319 AF   C8 14           STX    HIWAT,U   AND HI WATER STRING
031C 0E   96              JMP    <START
                  ********************************
                  *  RUN--- FIRE THE ROCKETS AND
                  *       BLAST OFF
                  *    EACH STATEMENT IS RESPONSIBLE
                  *    FOR COMING BACK TO RUNNXL,
                  *    RUNNNL,RUNTSL AND RUNSML
                  *    AS PROGRAM POINTER (Y) NEEDS
                  *    CARE AND FEEDING
031E              RUN       HERE   'RUN'
031E 9D   53              JSR    <ENDCHK
0320 31   8D 0453         LEAY   TXTBGN,PCR FIRST LINE
0324 4F           RUNNXL  CLRA
0325 5F                   CLRB              START AT ANY LINE >0
0326 17   01B8    RUNNNL  LBSR   FNDLNP
0329 1022 FE69            LBHI   START
032D 10AF 42      RUNTSL  STY    CURRNT,U
0330 31   22              LEAY   2,Y        BUMP PASSED LINE NUM
0332 9D   8A      RUNSML  JSR    <CHKIO     RUN SAME LINE
0334                      DISPAT START      DOING STATEMENT
                  ********************************
                  *   STOP WILL CAUSE GRACEFUL
                  *    TERMINATION
                  *
0338              STOP      HERE   'STOP'
033B 9D   53              JSR    <ENDCHK
033A 0E   96              JMP    <START
                  ********************************
                  *   PRINT  PRINT OUT EXPRESSIONS,
                  *       STRINGS. WITH POSSIBLE
                  *       FORMAT FOR DIGITS
                  *
033C              PRINT     HERE   'PRINT'
033C 86   06              LDA    #6
033E A7   41              STA    FLDCNT,U
0340 9D   72              JSR    <IGNBLK
0342 81   3B              CMPA   #';
0344 26   06              BNE    PR2
0346 31   21              LEAY   1,Y
0348 9D   7D              JSR    <CRLF
034A 20   E6              BRA    RUNSML
034C 81   0D      PR2     CMPA   #$0D
034E 26   06              BNE    PR0
0350 31   21              LEAY   1,Y
0352 9D   7D              JSR    <CRLF
0354 20   CE              BRA    RUNNXL
0356 81   23      PR0     CMPA   #'#        IS IT A FORMAT LENGTH?
0358 26   08              BNE    PR1
035A 31   21              LEAY   1,Y
035C 9D   ED              JSR    <EXPR
035E E7   41              STB    FLDCNT,U
0360 20   0A              BRA    PR3
0362 17   0236    PR1     LBSR   QTSTG      PRINT POSSIBLE STRING
0365 27   05              BEQ    PR3
0367 9D   ED              JSR    <EXPR
0369 17   01D5            LBSR   PRTNUM
036C 9D   72      PR3     JSR    <IGNBLK
036E 81   2C              CMPA   #',
0370 26   06              BNE    PR6
0372 31   21              LEAY   1,Y
0374 9D   3B              JSR    <FIN       IF NOT ENDLINE THEN COME BACK
0376 20   DE              BRA    PR0
```

(continued on the next page)

S. B. Tiny Basic for 6809's (Continued)

```
0378 9D   7D        PR6     JSR     <CRLF       END OF STATEMENT WITH
037A 0E   5C                JMP     <FINE       IMPLIED CR AND LF
                    ********************************
                    *   GOTO
                    *
037C                        HERE    'GOTO'
037C 9D   ED                JSR     <EXPR
037E 34   20                PSHS    Y           SAVE TXT POINTER FOR ERROR SUB
0380 17   0155              LBSR    FNDLN
0383 26   7C                BNE     GOERR       COULDN'T FIND DESTINATION
0385 32   62                LEAS    2,S         DISCARD OLD LINE NUM
0387 20   A4                BRA     RUNTSL
                    *
                    ********************************
                    *   IF
                    *
0389                        HERE    'IF'
0389 9D   ED                JSR     EXPR        EVALUATE THE EXPRESSION
038B 26   A5                BNE     RUNSML      IF <>0 THEN CONT
                    *     FALL THROUGH TO
                    *
                    ********************************
                    *  REM  IGNORE THIS WHOLE LINE
                    *
038D                        HERE    'REM'       LOOKS LIKE 'IF 0'
038D 10AE 42                LDY     CURRNT,U
0390 CC   0001              LDD     #1
0393 E3   A4                ADDD    0,Y         CURRENT LINE PLUS ONE
0395 20   8F                BRA     RUNNNL
                    ********************************
                    *  INPUT
                    *
0397 10EE 46        INPERR  LDS     STKINP,U    REGAIN
039A 35   36                PULS    D,X,Y       CURRNT,JUNK,TEXT PTR
039C ED   42                STD     CURRNT,U    AND RETRY THIS INPUT STMT
039E                        HERE    'INPUT'
039E 34   20        IP1     PSHS    Y           SAVE TEXT POINTER
03A0 17   01F8              LBSR    QTSTG       PRINT QUOTED PROMPT
03A3 26   06                BNE     IP2         NO TRY VARIABLE
03A5 9D   99                JSR     TSTVE
03A7 25   3D                BCS     IP4
03A9 20   17                BRA     IP3         YES INPUT VARIABLE
03AB 34   20        IP2     PSHS    Y           SAVE FOR PRTSTG
03AD 9D   99                JSR     TSTVE
03AF 1025 FDAB              LBCS    QWHAT
03B3 A6   A4                LDA     ,Y
03B5 6F   A4                CLR     ,Y          SET TO ZERO AS STOPPER
03B7 35   20                PULS    Y
03B9 34   02                PSHS    A
03BB 17   01CE              LBSR    PRTSTL      PRINT PROMPT
03BE 35   02                PULS    A
03C0 A7   A2                STA     ,-Y         REPLACE CHAR OF PROG
03C2 1F   10        IP3     TFR     X,D         REGAIN ADDRESS OF VAR
03C4 34   20                PSHS    Y
03C6 AE   42                LDX     CURRNT,U
03C8 34   10                PSHS    X
03CA 30   8C F5             LEAX    IP3,PCR     POINT CURRNT TO NG AS INPUT FLAG
03CD AF   42                STX     CURRNT,U
03CF 10EF 46                STS     STKINP,U    SAVE STACK POINTER
03D2 34   06                PSHS    D           THE DESTINATION ADDRESS
03D4 86   3A                LDA     #':         PROMPT
03D6 17   0125              LBSR    GETLN       GET THE INPUT FROM USER
```

```
044D EC   84       NEXT3   LDD    0,X        AND PUT VALUE INTO ACCUM
044F E3   4A               ADDD   LOPINC,U
0451 ED   84               STD    0,X
0453 6D   4A               TST    LOPINC,U   WHICH DIRECTION ARE WE GOING?
0455 2B   0F               BMI    NEXT4
0457 10A3 4C               CMPD   LOPLMT,U
045A 2E   0E               BGT    NEXT5      DONE
045C AE   4E       NEXT6   LDX    LOPLN,U
045E AF   42               STX    CURRNT,U
0460 10AE C8 10            LDY    LOPPT,U
0464 0E   5C               JMP    <FINE
0466 A3   4C       NEXT4   SUBD   LOPLMT,U
0468 2C   F2               BGE    NEXT6
046A 17   0183     NEXT5   LBSR   POPA
046D 0E   5C               JMP    <FINE
                   ********************************
                   * FOR
                   *
046F                       HERE   'FOR'
046F 20   05               BRA    FORE       JUMP AROUND DEFAULT STMT CODE
                   *          TO ALLOW TABLE ENTRY
                   *          FOR ASSIGNMENT TO BE
                   *          PRIOR TO KEYWORDS OF
                   *       FOR STATEMENT IN TABLE
                   *     THIS IS REQUIRED BY THE
                   *     MACRO HERE AND THE WAY IT
                   *     WORKS
0471                       HERE   ,          DEFAULT ASSIGNMENT
0471 9D   38               JSR    <FIN       CAN BE EMPTY
0473 16   FF7E             LBRA   LET        OTHERWISE TREAT AS LET
                   ********************************
                   * ACTUAL CODE OF FOR
                   *
0476 17   019E     FORE    LBSR   PUSHA      SAVE LOOP VARS
0479 8D   45               BSR    SETVAL
047B AF   48               STX    LOPVAR,U   SAVE ADDR OF LOOP VAR
047D                       DISPAT
0481                       HERE   'TO'       'FOR'I=1 'TO'
0481 9D   ED               JSR    <EXPR
0483 ED   4C               STD    LOPLMT,U   STOPPING VAL
0485 20   02               BRA    FOR2       JUMP AROUND DEFAULT
0487                       HERE   ,          DID NOT HAVE 'TO'
0487 0E   5E               JMP    <QWHAT     SO COMPLAIN
0489               FOR2    DISPAT
048D                       HERE   'STEP'
048D 9D   ED               JSR    <EXPR
048F 20   03               BRA    FOR3
0491                       HERE   ,          NO STEP VALUE
0491 CC   0001             LDD    #1         SO DEFAULT TO ONE
0494 ED   4A       FOR3    STD    LOPINC,U
0496 AE   42               LDX    CURRNT,U
0498 AF   4E               STX    LOPLN,U    LOOP LINE NUMBER
049A 10AF C8 10            STY    LOPPT,U    LOOP PROG POINTER
049E 31   76               LEAY   -10,S      START LOOKING AT OLD 'PUSHA' RECOR
DS
04A0 31   2A       FOR7    LEAY   10,Y
04A2 AE   A4               LDX    0,Y        GET LOOP VAR ADDRESS
04A4 27   14               BEQ    FOR8       EXHAUSTED ALL FORS
04A6 AC   48               CMPX   LOPVAR,U   IS OLD SAME AS THIS VAR?
04A8 26   F6               BNE    FOR7
04AA 30   2A               LEAX   +10,Y      FOUND MATCH MUST DELETE IT
04AC 10EF E3               STS    ,--S       SAVE STACK POINTER
04AF A6   A2       FOR7L   LDA    ,-Y
```

```
03D9 31   C8 18          LEAY   BUFFER,U
03DC 9D   ED             JSR    EXPR          AND EVALUATE IT AS AN EXPRESSION
03DE ED   F1             STD    [0,S++]       SAVE VALUE IN VAR
03E0 35   10             PULS   X
03E2 AF   42             STX    CURRNT,U
03E4 35   20             PULS   Y
03E6 35   06     IP4     PULS   D             PURGE JUNK
03E8                     TSTC   "',",IP5       IS NEXT A COMMA
03F0 20   AC             BRA    IP1           YES, DO ANOTHER VARIABLE
03F2 0E   5C     IP5     JMP    <FINE         FINISH
                 ********************************
                 * LET   THE ASSIGNMENT STATEMENT
                 *
03F4                     HERE   'LET'
03F4 17   00C9   LET     LBSR   SETVAL        DO THE ASSIGNMENT
03F7                     TSTC   "',",IP5       CHECK END LINE
03FF 20   F3             BRA    LET
                 ****************************
                 * GOSUB
                 *
0401 35   20     GOERR   PULS   Y             REGAIN CURRENT LINE
0403 0E   2C             JMP    <QHOW         AND PRINT DIAGNOSTIC
0405                     HERE   'GOSUB'
0405 17   020F           LBSR   PUSHA         SAVE FOR PARAMETERS
0408 9D   ED             JSR    <EXPR         GET VAL IN D
040A 34   20             PSHS   Y             SAVE TEXT POINTER
040C 17   00C9           LBSR   FNDLN
040F 26   F0             BNE    GOERR         NO LINE SAY 'HOW'
0411 EC   42             LDD    CURRNT,U
0413 34   06             PSHS   D
0415 EC   46             LDD    STKGOS,U
0417 34   06             PSHS   D
0419 4F                  CLRA
041A 5F                  CLRB
041B ED   48             STD    LOPVAR,U      SAVE NEW ONES
041D 10EF 46             STS    STKGOS,U
0420 16   FF0A           LBRA   RUNTSL        AND RUN DEST LINE
                 ********************************
                 *  RETURN   FROM GOSUB
                 *
0423                     HERE   'RETURN'
0423 9D   53             JSR    <ENDCHK       BAD IF NOT END OF LINE
0425 EC   46             LDD    STKGOS,U      RELOAD STACK POINTER
0427 1027 FD01           LBEQ   QHOW          TOO MANY RETURNS
042B 1F   04             TFR    D,S           RESTORE STACK POINTER
042D 35   36             PULS   D,X,Y
042F ED   46             STD    STKGOS,U
0431 AF   42             STX    CURRNT,U
0433 17   01BA           LBSR   POPA          RESTORE FOR--NEXT ENVIRONMENT
0436 0E   5C             JMP    <FINE
                 ********************************
                 * NEXT
                 *
0438                     HERE   'NEXT'
0438 AE   48             LDX    LOPVAR,U      IF NO FOR THEN
043A 27   0F             BEQ    NEXT2         COMPLAIN LOUDLY
043C 9D   99             JSR    <TSTVE
043E 24   02             BCC    NEXTO         IF NO VARIABLE SPECIFIED
0440 AE   48             LDX    LOPVAR,U      THEN UST USE LATEST FOR VARIABLE
0442 AC   48     NEXTO   CMPX   LOPVAR,U
0444 27   07             BEQ    NEXT3
0446 17   01A7           LBSR   POPA
0449 26   F7             BNE    NEXTO
044B 0E   2C     NEXT2   JMP    <QHOW
```

(Continued on top of page 28)

```
04B1 A7   82              STA    ,-X
04B3 10AC E4              CMPY   0,S
04B6 2E   F7              BGT    FOR7L
04B8 32   02              LEAS   2,X          CUT BACK STACK
04BA 10AE C8 10   FOR8    LDY    LOPPT,U      GET PROG POINTER
04BE 0E   5C              JMP    <FINE
                  ********************************
04C0 17   FCD6    SETVAL  LBSR   TSTVE        MUST BE VARIABLE
04C3 25   11              BCS    QW1          NO VARIABLE
04C5 34   10              PSHS   X
04C7                      TSTC   '=,QW1
04CF 9D   ED              JSR    <EXPR        EVALUATE THE EXPRESSION
04D1 35   10              PULS   X
04D3 ED   84              STD    0,X          LEAVE ADDRESS IN X (FOR USES IT)
04D5                      RATS
04D6 0E   5E      QW1     JMP    <QWHAT       RELAY TO QWHAT
                  ********************************
                  * UTILITIES AND EXTRAS
                  *
                  * FINDLINE LINE NUMBER IN D
04D8 4D           FNDLN   TSTA
04D9 102B FC4F            LBMI   QHOW         NEGATIVE LINE NUM
04DD 31   8D 0296         LEAY   TXTBGN,PCR   INITIAL LINE PTR
04E1 10AC 44      FNDLNP  CMPY   TXTUNF,U     PASSED END?
04E4 22   EF              BHI    RETURN
04E6 10A3 A4              CMPD   0,Y          LOOK AT LINE NUMBER
04E9 23   EA              BLS    RETURN       IF REG <= LINE NUM
            04EB  FNDNXT  EQU    *
04EB 10AC 44              CMPY   TXTUNF,U     DONT SCAN PAST END
04EE 22   E5              BHI    RETURN
04F0 31   22              LEAY   2,Y
04F2 34   02              PSHS   A            SAVE LINE NUMB HIGH BYTE
04F4 86   0D              LDA    #13
04F6 A1   A0      F1      CMPA   ,Y+
04F8 26   FC              BNE    F1           SCAN TO CR OF LINE
04FA 35   02              PULS   A            REGAIN  HIGH LINE NUM
04FC 20   E3              BRA    FNDLNP       TRY THIS LINE
                  ********************************
                  * GETLN     GET INPUT LINE AND
                  *           EDIT ACCORDING TO
                  *           BACKSPACE, CNTL X
                  *
04FE 9D   7F      GETLN   JSR    <OUTC        PROMPT
0500 31   C8 18           LEAY   BUFFER,U     START AT BEGINNING
0503 30   C8 60           LEAX   BUFEND,U
0506 34   30              PSHS   Y,X
0508 9D   8A      GL1     JSR    <CHKIO
050A 27   FC              BEQ    GL1          NOTHING YET
050C 9D   7F              JSR    <OUTC        ECHO
050E 81   0A              CMPA   #$0A         LINE FEED
0510 27   F6              BEQ    GL1          IGNORE IT
0512 81   08              CMPA   #$1F&'H       DELETE CODE
0514 27   0F              BEQ    GL3
0516 81   18              CMPA   #$1F&'X       CONTROL X
0518 27   1A              BEQ    GL4
051A A7   A0              STA    ,Y+
051C 81   0D              CMPA   #$D
051E 27   1C              BEQ    GL0          RETURN
0520 10AC E4              CMPY   S            WITH BUFFER END
0523 26   E3              BNE    GL1
0525 10AC 62      GL3     CMPY   2,S          ANYTHING IN BUFFER?
0528 27   0A              BEQ    GL4
052A 31   3F              LEAY   -1,Y
052C 9D   79              JSR    <SPACE
```

(Continued on next page)

S. B. Tiny Basic for 6809's (Continued)

```
052E 86   08          LDA    ##1F&'H   CONTROL H
0530 9D   7F          JSR    <OUTC     AND BACK UP OVER SPACE
0532 20   D4          BRA    GL1
0534 9D   7D    GL4   JSR    <CRLF
0536 32   64          LEAS   4,S       POP OFF TEMPORARIES
0538 86   3F          LDA    #'?       QUESTION MARK PROMPT
053A 20   C2          BRA    GETLN
053C 32   64    GL0   LEAS   4,S
053E A7   22          STA    2,Y       THROW ANOTHER RETURN
0540                  RATS
                      ********************************
                      * PRTNUM AND FRIENDS
                      *
0541 8E   000A  PRTNUM LDX    #10
0544 34   36          PSHS   D,Y,X     SAVE PRG POINTER AND STOPPER
0546 E6   41          LDB    FLDCNT,U
0548 5A               DECB
0549 49               ROLA             PUT SIGN OF QUANTITY INTO LOW B
054A 59               ROLB
054B 1D               SEX
054C 1F   02          TFR    D,Y       PUT INTO COUNTER
                      ********
054E EC   E1          LDD    0,S++     SET FLAGS
0550 2A   07          BPL    PN2       COMPLEMENT IF NEGATIVE
0552 43               COMA
0553 53               COMB
0554 C3   0001        ADDD   #1
0557 31   3E    PN1   LEAY   -2,Y      DECREMENT PLACE COUNT
0559 34   06    PN2   PSHS   D          SAVE QUOTIENT
055B CC   000A        LDD    #10       DIVIDE BY 10 TO GET
055E 17   01CC        LBSR   DIV       NEXT DIGIT
0561 26   F4          BNE    PN1       HAVE NOT GOT ALL
0563 1F   20          TFR    Y,D       PUT PLACE COUNT INTO B
0565 20   02          BRA    PN3
0567 9D   79    PN4   JSR    <SPACE    PRINT SPACE
0569 C0   02    PN3   SUBB   #2        DEC SPACE COUNT
056B 2A   FA          BPL    PN4
056D 56               RORB             PUT SIGN BIT INTO CARRY
056E 24   04          BCC    PN6       DONT PRINT MINUS
0570 86   2D          LDA    #'-       LEADING MINUS
0572 9D   7F    PN8   JSR    <OUTC
0574 35   06    PN6   PULS   D         GET REMAINDER
0576 C1   0A          CMPB   #10       DONE?
0578 27   06          BEQ    PN7
057A 1F   98          TFR    B,A
057C 8A   30          ORA    ##30
057E 20   F2          BRA    PN8
0580 35   A0    PN7   PULS   Y,PC      AND RETURN
                      ********************************
                      * PRTLN   PRINT LINE OF BASIC
                      *         PROGRAM
                      *
0582 C6   04    PRTLN LDB    #4        FOUR SPACES
0584 E7   41          STB    FLDCNT,U
0586 EC   A1          LDD    ,Y++      GET LINE NUMBER
0588 8D   B7          BSR    PRTNUM    AND PRINT IT
058A 9D   79          JSR    <SPACE
058C 5F               CLRB             FORCE NO MATCH
                      * AND FALL INTO
                      ********************************
                      * PRTSTG  PRINT STRING
                      *
058D E1   A0    PRTSTG CMPB    ,Y+       PRINT STRING AT Y
0613 32   62          LEAS   2,S
0615 1F   05          TFR    D,PC
                      *
0617 17   FC87  PUSHA LBSR   SIZE       ANU ROOM?
061A 83   000C        SUBD   #12
061D 1023 FB46        LBLS   QSORRY
0621 35   06          PULS   D          RETURN ADDR
0623 AE   48          LDX    LOPVAR,U   TEST THIS
0625 27   13          BEQ    PU1
0627 AE   C8 10       LDX    LOPPT,U
062A 34   10          PSHS   X
062C AE   4E          LDX    LOPLN,U
062E 34   10          PSHS   X
0630 AE   4C          LDX    LOPLMT,U
0632 34   10          PSHS   X
0634 AE   4A          LDX    LOPINC,U
0636 34   10          PSHS   X
0638 AE   48          LDX    LOPVAR,U
063A 34   10    PU1   PSHS   X
063C 1F   05          TFR    D,PC       AND RETURN
                      ********************************
                      * ST1 (ALIAS START)
                      *    INITIALIZE AND ENTER COMMAND
                      *    MODE
                      *
063E 4F 4B 0D   OK    FCC    'OK',13
0641 30   8D FABB ST1 LEAX   TSTNUM,PCR
0645 1F   10          TFR    X,D
0647 1F   8B          TFR    A,DP
0649 33   8D 0512     LEAU   DATAS,PCR
064D 32   C8 CA       LEAS   VARBGN,U
0650 9D   7D          JSR    <CRLF
0652 31   8C E9       LEAY   OK,PCR     PRINT OK AND PROMPT
0655 17   FF34       LBSR   PRTSTL
0658 4F               CLRA             B IS ALREADY ZERO
0659 ED   48          STD    LOPVAR,U   NO CURRENT FOR LOOPS
065B ED   46          STD    STKGOS,U   NOR GOSUB RETURNS
065D 86   3E    ST3   LDA    #'>        PROMPT TO ENTER A STATEMENT
065F 17   FE9C        LBSR   GETLN
0662 34   20          PSHS   Y          SAVE POINTER TO END OF TEXT
0664 31   C8 18       LEAY   BUFFER,U   POINT TO START
0667 9D   00          JSR    <TSTNUM    WILL RETURN ZERO IF NO NUM
0669 34   07          PSHS   CC,D
066B 9D   72          JSR    <IGNBLK
066D EC   61          LDD    1,S        REGAIN LINE NUM
066F ED   A3          STD    ,--Y       PUT LINE NUMBER IN FRONT
0671 10AF 42          STY    CURRNT,U   MAKE THE CURRENT LINE
0674 35   07          PULS   CC,D
0676 1027 FC78        LBEQ   DIRECT     DO IMMEDIATE MODE IF NO LEADING NU
MBER
                      * Y POINTS TO BEGINNING OF LINE
                      * AND TOP OF STACK POINTS TO END
067A 34   20          PSHS   Y
067C 17   FE59        LBSR   FNDLN
067F 34   20          PSHS   Y          TOS->TEXT AREA
0681 26   17          BNE    ST4        INSERT
0683 17   FE65        LBSR   FNDNXT
0686 AE   E4          LDX    0,S        BEGINNING OF LINE TO DELETE
0688 20   04          BRA    STL
068A A6   A0    STLE  LDA    ,Y+
068C A7   80          STA    ,X+
068E 10AC 44    STL   CMPY   TXTUNF,U   ARE WE DONE MOVING?
0691 23   F7          BLS    STLE
```

```
058F 27   AF              BEQ     RETURN      IF SAME AS STOPPER
0591 A6   3F              LDA     -1,Y        GET THAT CHAR
0593 9D   7F              JSR     <OUTC
0595 81   0D              CMPA    #13
0597 26   F4              BNE     PRTSTG      NOT A CR SO GO
0599 4D                   TSTA                IF RETURN  THEN NON ZERO
059A                      RTS
                    *
                    * PRINT QUOTED STRING
                    *  RETURN WITH NOT ZERO IF DIDNT RECOGNIZE AS OUR STUFF
059B                QTSTG   TSTC    '",QT3      DOUBLE QUOTE
05A3 C6   22              LDB     #'"
05A5 8D   E6        QT1     BSR     PRTSTG
05A7 1026 FD79            LBNE    RUNNXL      RUN NEXT LINE IF HIT RETURN
05AB                      RTS     RETURN      W ZERO SET
05AC                QT3     TSTC    $27,QT4     SINGLE QUOTE
05B4 C6   27              LDB     #$27        PRINT TILL MATCH
05B6 20   ED              BRA     QT1
05B8                QT4     TSTC    '!,RETURN
05C0 86   8D              LDA     #$8D        FUNNY RETURN
05C2 9D   7F              JSR     <OUTC       APPLE DOES NOT SUPPORT THIS
05C4 4F                   CLRA                ZET ZERO STATUS
05C5                      RTS
                    ********************************
                    *  ERROR REPORT
05C6 34   20        ERROR   PSHS    Y           SAVE LINE NUMBER
05C8 10AE 62              LDY     2,S         GET THE RETURN ADDR
05CB 8D   BF              BSR     PRTSTL      PRINT HOW OR WHAT
05CD 35   20              PULS    Y
05CF A6   A4              LDA     0,Y
05D1 34   02              PSHS    A           SAVE THE CHAR AT ERROR POINT
05D3 6F   A4              CLR     0,Y
05D5 10AE 42              LDY     CURRNT,U    IS THIS IN
05D8 EC   A4              LDD     0,Y         IMMEDIATE MODE?
05DA 27   12              BEQ     ERRO
05DC 102B FDB7            LBMI    INPERR      INPUT COMMAND??
05E0 8D   A0              BSR     PRTLN       PRINT UP TO ZERO
05E2 86   3F              LDA     #'?
05E4 9D   7F              JSR     <OUTC       PRINT QUESTION
05E6 35   02              PULS    A
05E8 A7   A2              STA     0,-Y        RESTORE LINE
05EA 32   62              LEAS    2,S         POP PHONY RETURN
05EC 8D   9E              BSR     PRTSTL
05EE 0E   96        ERRO    JMP     <START
                    ********************************
                    *   POPA AND PUSHA
                    *
                    *
05F0 EC   62        POPA    LDD     2,S
05F2 27   1B              BEQ     POPAO
05F4 ED   48              STD     LOPVAR,U
05F6 EC   64              LDD     4,S
05F8 ED   4A              STD     LOPINC,U
05FA EC   66              LDD     6,S
05FC ED   4C              STD     LOPLMT,U
05FE EC   68              LDD     8,S
0600 ED   4E              STD     LOPLN,U
0602 EC   6A              LDD     10,S
0604 ED   C8 10           STD     LOPPT,U
0607 35   06              PULS    D
0609 32   6A              LEAS    10,S
060B 1C   FB              ANDCC   #$FB        TURN OFF ZERO
060D 1F   05              TFR     D,PC
060F ED   48        POPAO   STD     LOPVAR,U
0611 35   06              PULS    D
```

(Continued on top of page 30)

```
0693 30   1F              LEAX   -1,X
0695 AF   44              STX    TXTUNF,U  UPDATE END POINTER
0697 AF   C8 14           STX    HIWAT,U   AND STRING SPACE
069A EC   64        ST4   LDD    4,S       POINT TO END OF LINE
069C A3   62              SUBD   2,S       FIND IF LENGTH  OF LINE IS EMPTY
069E 1083 0003            CMPD   #3
06A2 27   9D              BEQ    ST1       YES, DO NOT INSERT
                    *MAKE ROOM FOR NEW LINE
06A4 10AE 44              LDY    TXTUNF,U
06A7 30   AB              LEAX   D,Y       ADD IN LENGTH
06A9 AF   44              STX    TXTUNF,U  UPDATE END POINTER
06AB AF   C8 14           STX    HIWAT,U
06AE 86   0D              LDA    #$0D
06B0 A7   84              STA    0,X
06B2 A6   A2        ST2LE LDA    ,-Y
06B4 A7   82              STA    ,-X
06B6 10AC E4              CMPY   0,S       HAVE WE REACHED BEGINNING?
06B9 22   F7              BHI    ST2LE
                    * START MOVING LINE
06BB 35   20              PULS   Y
                    *X POINTS TO END OF LINE AREA
                    *Y POINTS TO BEGINNING OF LINE
                    *TOS POINTS TO  BEGINNING OF TEXT
                    *NOS POINTS TO END OF BUFFER
06BD 35   10              PULS   X
06BF A6   80        ST3LE LDA    ,X+
06C1 A7   A0              STA    ,Y+
06C3 AC   E4              CMPX   ,S
06C5 25   F8              BLO    ST3LE
06C7 35   10              PULS   X         DISCARD GARBAGE
06C9 20   92              BRA    ST3
                    *
                    ********************************
                    * MULTIPLY AND DIVIDE ROUTINES
                    * CALLING SEQUENCES:
                    * MULT  LDD #MULTIPLIER
                    *       PSHS D
                    *       LDD #MULTIPLICAND
                    *       LBSR MULT
                    * ON EXIT D HAS HIGH ORDER BITS
                    * TOS HAS LOW ORDER BITS
                    * EQAL ZERO IT TRUE IFF THE
                    * CONTEDTS OF D ARE THE SAME
                    * AS THE HIGH BIT OF TOS
                    *   NEG IS SET  ON THE SIGN
                    * OF  TOS
06CB 32   7D        MULT  LEAS   -3,S
06CD BD   3D              BSR    UNSIGN    RETURNS W/D=0
06CF 8E   0010            LDX    #16
06D2 66   65              ROR    5,S
06D4 66   66              ROR    6,S
06D6 24   02        M1    BCC    M2
06D8 E3   61              ADDD   1,S
06DA 46             M2    RORA
06DB 56                   RORB
06DC 66   65              ROR    5,S
06DE 66   66              ROR    6,S
                    * RESULT IS NOW IN D CAT 5,S
06E0 30   82              LEAX   ,-X
06E2 26   F2              BNE    M1
06E4 6D   E4        SIGN  TST    0,S       SIGN LOC
06E6 2A   10              BPL    SETFLG
06E8 43                   COMA
06E9 53                   COMB
```

(Continued on next page)

S. B. Tiny Basic for 6809's (Continued)

```
06EA 63   65              COM    5,S
06EC 60   66              NEG    6,S
06EE 26   08              BNE    SETFLG
06F0 6C   65              INC    5,S
06F2 26   04              BNE    SETFLG
06F4 5C                   INCB
06F5 26   01              BNE    SETFLG
06F7 4C                   INCA
06F8 6D   65    SETFLG    TST    5,S
06FA 2B   07              BMI    S2
06FC ED   7E              STD    -2,S       SET FLAGS
06FE 1C   F7              ANDCC  #$F7       SET POSITIVE
0700 32   63              LEAS   3,S        CUT GARBAGE
0702 39                   RTS
0703 1083 FFFF  S2        CMPD   #$FFFF     SET FLAG
0707 1A   08              ORCC   #8         SET NEG
0709 32   63              LEAS   3,S        CUT GARBAGE
070B 39                   RTS
070C ED   63    UNSIGN    STD    3,S
070E A7   62              STA    2,S        SIGN POSITION
0710 67   62              ASR    2,S        DOUBLE THE SIGN
0712 2A   06              BPL    US2
0714 4F                   CLRA
0715 5F                   CLRB
0716 A3   63              SUBD   3,S
0718 ED   63              STD    3,S
071A A6   67    US2       LDA    7,S
071C 2A   0C              BPL    USR
071E A6   62              LDA    2,S
0720 88   80              EORA   #$80
0722 A7   62              STA    2,S
0724 4F                   CLRA
0725 5F                   CLRB
0726 A3   67              SUBD   7,S
0728 ED   67              STD    7,S
072A 4F         USR       CLRA
072B 5F                   CLRB
072C 39                   RTS
                ********************************
                * DIVIDE ROUTINE
                *  ENTER WITH DIVISOR IN D
                * DIVIDEND ON STACK TOP
                * JSR DIV
                * REMAINDER ON TOP OF STACK
                * QUOTIENT IN D
                * FLAGS SET AS TO D
                *
072D 32   7D    DIV       LEAS   -3,S       MAKE ROOM
072F 8D   DB              BSR    UNSIGN
0731 8E   0011            LDX    #17
0734 20   02              BRA    DIV1
0736 59         DIV5      ROLB
0737 49                   ROLA
0738 A3   61    DIV1      SUBD   1,S
073A 2B   12              BMI    DIV3
073C 1A   01              SEC
073E 69   66    DIV2      ROL    6,S
0740 69   65              ROL    5,S
0742 30   1F              LEAX   -1,X
0744 26   F0              BNE    DIV5
0746 20   10              BRA    DIV4       DONE
0748 59         DIVR      ROLB
0749 49                   ROLA
074A E3   61              ADDD   1,S
```