Synthetically Speaking—Part II

by David L. Engelhardt

After building last month's speech board for your Model I/III, program it with applications for education and the visually handicapped.

Last month I showed you how to build your own speech board for a Model I or III ("Synthetically Speaking—Part I," p. 142). Although once an expensive undertaking, adding synthetic speech capability to your computer is now easy and inexpensive due to the Votrax speech chip.

My previous article covered the construction and testing of the speech board. This month I present four programs that let you use the speech board in educational applications and as an aid for the visually handicapped.

Program Listing 1 is a Basic program that converts text to speech on key entry via Program Listings 2 and 3—the Assembly-language routines that make up the Text-to-Speech program. Program Listing 4 converts, through the Text-to-Speech program, strings of text entered through the keyboard into speech.

The programs run on a 16K RAM or larger computer. However, if you own a 16K machine you'll have to get someone with a bigger system to assemble the controlling program due to its large size.

Text-to-Speech Program

Listing 2 consists of the Text-to-Speech program and Listing 3 is the Rules Table that controls the conversion of ASCII text to speech. I separate the two programs due to the size of the Rules Table. The commented source listing of the Text-to-Speech program is 17,153 bytes long and the object code 1,680 bytes long. It is complex due to the many steps required to convert ASCII text to simulated speech.

The program takes each character in an English-language string and scans the appropriate rules one at a time. When it finds a matching rule, it inserts the proper phoneme code into an output buffer, which it later sends to the SC-01 chip.

If it doesn't find a matching rule, the program enters an Error routine, and displays the deciphered portion of the input string along with its phoneme equivalents. The program then outputs an error message to the Votrax chip. This routine gives you a chance to see if you need any additional rules in the table.

The Text-to-Speech program consists of many routines controlled by special characters within the rules. Not only does the program examine a specific character in a word, it also examines what precedes and follows the character. A character's location and use dictate which routine the program calls. The Text-to-Speech program checks for inflection codes, nonalphabetic characters, one or more vowels,

Symbol	Definition
<	Causes jump to the Left-Scan routine.
>	Causes jump to the Right-Scan routine.
!	Causes call to a routine that attempts to match any non-alphabetic character in the input string pointed to by the IX register pair. (Blanks, numbers, etc.)
#	Causes call to a routine that attempts to match one or more vowels. (A,E,I,O,U,Y)
:	Causes call to a routine that attempts to match zero or more consonants. There is always a match. (B,C,D,F,G,H,J,K,L,M,N,P,Q,R,S,T,V,W,X,Z)
"	Causes call to a routine that attempts to match one consonant.
.	Causes call to a routine that attempts to match a voiced consonant. (B,D,G,J,L,M,N,R,V,W,Z)
+	Causes call to a routine that attempts to match front vowels. (E,I,Y)
%	Causes call to a routine that attempts to match a suffix at the end of a word. (E,ER,ES,ED,ING,ELY)
&	Causes call to a routine that attempts to match sibilants. (S,C,G,Z,X,J,CH,SH)
$	Causes call to a routine that attempts to match influencing consonants. (T,S,R,D,L,Z,N,J,TH,CH,SH)

Table 1. Text-to-Speech special symbols.

The Key Box

Models I and III
16K RAM Cassette Basic
32K RAM Disk Basic
Assembly Language
(If 16K—object code only)
Hardware project

```
10 '**************************************************
20 '            L I S T I N G  1
30 '**************************************************
40 '
50 '       KEY ENTRY PRONOUNCIATION PROGRAM
60 '                     BY
70 '               DAVID ENGELHARDT
80 '
100 '
110 ' DEPENDING UPON THE OPTION SELECTED, THE PROGRAM WILL
120 ' EITHER PRONOUNCE EACH KEY AS IT IS PRESSED OR NOT
130 ' PRONOUNCE THE SELECTED KEY.
140 ' THE PROGRAM WILL PRONOUNCE EACH WORD THAT IS CREATED
150 ' UPON THE ENTRY OF THE SPACEBAR.
160 ' WHEN THE -ENTER- KEY IS HIT, THE ENTIRE SENTENCE WILL BE
170 ' CREATED TO SPEECH.
180 '
190 '**************************************************
200 '
210 CLS:CLEAR 1000
220 DEF USR0=&HE950:CMD"L","TTSPRG/CMD":CMD"L","TABLE/CMD"
230 ON ERROR GOTO 720
240 A$="DO YOU WANT TO REPEAT E CH KEY" :
250 A=VARPTR(A$)
260 VV=USR0(A):
                'SPEAK ENTRY CHOICE MESSAGE
270 A$=INKEY$:IFA$=""GOTO270       'GET CHOICE
280 IFA$="Y" AN$="Y":GOTO 310
290 IFA$="N" AN$="N":GOTO 310
300 GOTO 270
310 C$="YOU HAVE REACHED THE MAXIMUN NUMBER OF ENTRYS.  THIS IS WH
AT YOU HAVE TYPED IN SO FAR" :
320 D$="ENTER PLEASE" :
330 CT=1:                          'BACKSPACE COUNTER
340 A=VARPTR(D$):VV=USR0(A):       'SPEAK ENTER PLEASE MESSAGE
350 A$=INKEY$:IFA$=""GOTO350
360 IFA$=CHR$(13) TS$=TS$+SB$:B$=TS$:GOTO 660  : 'ENTER KEY
370 IF A$<>CHR$(32) GOTO 460  :    'SPACEBAR
380 SB$=SB$+A$:                    'CREATE WORD STRING
390 TS$=TS$+SB$:                   'CREATE TOTAL STRING
400 IFLEN(TS$)>255 B$=TS$:GOTO 620
410 A=VARPTR(SB$)
420 VV=USR0(A):                    'SPEAK CREATED WORD
430 SB$=""
440 CT=0:                          'CLEAR BACKSPACE COUNTER
450 GOTO530
460 IFA$=CHR$(8) AND CT=1 GOTO350  :  'CHECK FOR BACKSPACE
470 IFA$<>CHR$(8) GOTO 510  :      'CHECK FOR MAX BACKSP COUNT
480 CT=CT-2
490 SB$=LEFT$(SB$,CT):             'RE-CREATE BACKSPACED WORD
500 GOTO530
510 SB$=SB$+A$:                    'CREATE WORD STRING
520 IFLEN(SB$)>255 B$=SB$:GOTO 620
530 PRINTA$;
540 IFAN$="N"GOTO590  :            'TEST FOR CHOICE QUESTION
550 ZZ$=TS$:                       'MUST USE TO RETAIN STRING
560 A=VARPTR(A$)
570 VV=USR0(A):                    'SPEAK THE ENTERED KEY
580 TS$=ZZ$:                       'RESTORE SAVED STRING
590 CT=CT+1
600 IFCT<255 GOTO 650  :           'TEST FOR MAX STRING COUNT
610 B$=SB$
620 A=VARPTR(C$)
630 VV=USR0(A):                  'SPEAK MAX STRING COUNT MESS
640 GOTO 660
650 GOTO350
660 PRINT:PRINTB$
670 A=VARPTR(B$):                'SPEAK TOTAL CREATED STRING
680 VV=USR0(A)
690 B$="":TS$="":SB$="":         'CLEAR VARIABLES FOR NEXT STRG
700 GOTO310
710 '
720 B$="ERROR CONDITION...PLEASE REDUE":
730 RESUME 320
740 END
```

Program Listing 1. Basic application program.

consonants, voiced consonants, front vowels, suffixes, sibilants, influencing consonants, and ASCII characters. Table 1 defines the Text-to-Speech program's special symbols.

Listing 3 includes the rules index, rules pertaining to each letter of the alphabet, and number and special character rules. I also incorporated all basic math operators so you can use simple math functions in education programs.

How It Works

After saving all of the register contents on entry to the Text-to-Speech program, lines 230–360 condition the string input buffer with zeros and the output buffer with Votrax stop codes. This clears out any of the buffers' old string text and sets the stop character (00), which indicates the end of the conversion process. The stop code (3F) clears the output buffer and indicates a finished output condition. Notice that the output buffer is 145 bytes longer than the input buffer. This is because there isn't always a one-to-one relationship of ASCII letters to phoneme codes and, therefore, the phoneme buffer always contains more codes.

Lines 410–430 check for a string to convert, exiting if there is no string. This section then inserts a blank at the first location in the INPBUF buffer for speech conversion alignment purposes. Lines 530–600 convert everything to uppercase, and transfer the string from where it resides in memory to INPBUF.

The next routine, CHTYPE, finds the location in the Rules Table that corresponds to each character in a string. The program then scans and compares for matches. The Rules Table dictates what action the Text-to-Speech program takes. If it can't find a rule location, the program drops down to the Inflection Check routine and tests for the special # symbol. If it doesn't find the symbol, the program branches to the Error routine.

On a rule match, the program jumps to the RULSCN routine. This routine double-checks the HL and IX pointers to ensure that they recognize the same character, but in different locations. The program increments the Rules Table pointer and the routine decides which side of the character to check, if any. If the direction symbols are present, control passes to either the Left or Right Scan routines according to the control symbols < (left) and > (right).

If neither one of the direction symbols is in the rules, this section assumes that an ASCII character exists and checks for a match between the input string and rule character. Match-checking continues until the program accesses any of the special control characters. An equals sign within a rule indicates a match.

At a match the program branches to the TXFR routine that transfers the appropriate phoneme codes to the output buffer (OUTBUF). After this the GNXTCH routine checks for input string code 00. If the program doesn't locate this code, it jumps back to CHTYPE to scan the next character.

The rest of the program routines make specific tests for conditions that surround the character in check, such as vowels, influential consonants, and so on. Some of the routines that scan the left side sometimes check for two characters instead of one. For example, the SIBCON routine makes a test for influencing consonants. To keep a correct alignment, for example, when making CH and SH checks, I put them into the table in reverse order (HC, HS). Since the scan goes to the left, the first character checked is the H and the program checks the second character if the first matches.

The Output routine sends the phoneme codes to the Votrax chip. After the program converts the input string buffer to phonemes, it branches to this routine. Output handles both the string and error message outputs. The only difference between the two is the location to which the HL register pair points. The routine continues to send out phoneme codes until it reaches the stop character 3F. At the stop code, the program restores all registers and returns to the calling program (Basic Listings 1 or 4 in this case).

The Output routine also checks for the special inflection codes, which the INFLT routine inserts into the output buffer in response to the special symbol, #. You must enter the value of 1, 3, 5, or 7 from the keyboard following the # symbol.

Program Listing 2. Text-to-Speech source code.

```
00010 ;*********************************
00020 ;        L I S T I N G  2
00030 ;*********************************
00040 ;
00050 ;     TEXT-TO-SPEECH PROGRAM
00060 ;              BY
00070 ;         DAVID ENGELHARDT
00100 ;
00110 ;*********************************
00120 ;
00130          ORG     0E950H          ;ORG'ED FOR 48K SYSTEM
00140 CHTBLE   EQU     0F000H          ;STRT LOOKUP TABLE
00150 ENDTBL   EQU     0F0C0H          ;LENGTH OF LOOKUP TBL
00160 ;
00170 VOTRAX   PUSH    HL              ;SAVE ALL REGISTORS
00180          PUSH    DE
00190          PUSH    BC
00200          PUSH    AF
00210          PUSH    IX
00220          PUSH    IY
00230          LD      HL,INPBUF       ;GET START OF INPUT BUFF
00240          LD      DE,OUTBUF       ;GET START OF OUTPUT BUFF
00250          LD      B,255           ;BYTE COUNTER
00260          LD      A,3FH           ;BYTE TO CONDITION BUFF
00270          LD      C,0             ;BYTE TO CONDITION BUFF
00280 CLRBF1   LD      (HL),C          ;CONDITION INPUT BUFF
00290          LD      (DE),A          ;CONDITION OUTPUT BUFF
00300          INC     HL
00310          INC     DE
00320          DJNZ    CLRBF1          ;REDUE UNTIL B=0
00330          LD      B,145           ;LOAD COUNTER
00340 CLRBF2   LD      (DE),A          ;CONTINUE TO CONDITION
00350          INC     DE
00360          DJNZ    CLRBF2          ;DO IT AGAIN
00370 ;
00380          CALL    0A7FH           ;GET PASSED VALUE
00390          LD      B,0             ;CLR B REG
00400          LD      C,(HL)          ;GET LENGTH OF STRING
00410          XOR     A               ;CLR A REG
00420          CP      C               ;BAIL OUT IF C=0
00430          JP      Z,RETURN        ;ERROR-BAIL OUT
00440          INC     HL
00450          LD      E,(HL)          ;GET LSB OF STR. LOC
00460          INC     HL
00470          LD      D,(HL)          ;GET MSB OF STR. LOC
00480          LD      HL,INPBUF       ;GET STRT OF INPUT BUFF
00490          LD      (HL),' '        ;FILL 1ST LOC WITH BLANK
00500          PUSH    HL              ;SAVE STRT OF BUFF
00510          INC     HL              ;STRT TO PUT DATA
00520          EX      DE,HL           ;SWAP PTRS
00530 ML1      LD      A,(HL)          ;GET CHAR FROM STRNG BUFF
00540          SUB     61H             ;TEST UPPER/LOWER CASE
00550          JP      M,ML2           ;SKIP IF UPPER CASE
00560          LD      A,-20H          ;LOAD CONVERT VALUE
00570          ADD     A,(HL)          ;MAKE LOWER CASE UPPER
00580          LD      (HL),A          ;STORE VALUE BACK
00590 ML2      LDI                     ;MOVE CHARACTER
00600          JP      PE,ML1          ;IS THE MOVE DONE?
00610          INC     HL
00620          LD      (HL),' '        ;LOAD SPACE TO END
00630          POP     IX              ;INPBUF LOC TO IX
00640          LD      IY,OUTBUF       ;SET IY TO OUTPUT BUFF
00650 ;
00660 ;*************************
00670 ; CHARACTER LOOKUP ROUTINE
00680 ;*************************
00690 ;
00700 CHTYPE   LD      HL,CHTBLE       ;GET START OF TABLE
00710          LD      A,(ENDTBL)      ;GET LENGTH OF TABLE
00720          ADD     A,-3            ;OBTAIN CORRECT LENGTH
00730          LD      B,0             ;CLR B
00740          LD      C,A             ;LOAD LOOP CTR
00750          LD      A,(IX)          ;GET STRG CHA
00760          INC     HL              ;INC TO NEXT ASCII CH
00770 LOOP     INC     HL              ;INC TWICE PAST WORD
00780          INC     HL              ;  LOCATION OF RULE
00790          CPI                     ;HL=HL+1  BC=BC-1
00800          JR      Z,MATCH         ;FOUND MATCH
00810          DEC     C               ;SET TO TRACK HL
00820          DEC     C               ;DEC LENGTH CTR
00830          JR      NZ,LOOP         ;SCAN AGAIN
00840          JP      INFLT           ;CHECK FOR INFLECT CHAR
00850 ;
00860 MATCH    LD      E,(HL)          ;GET LSB OF CH. RULES
00870          INC     HL
00880          LD      D,(HL)          ;GET MSB OF CH. RULES
00890          LD      (SAVEIX),IX     ;SAVE IX PTR
00900          EX      DE,HL           ;PUT STRT OF RULE IN HL
00910          JP      RULSCN          ;COMPARE RULES
00920 ;
00930 ;*********************************
00940 ; INFLECTION CHANGE ROUTINE = #
00950 ;*********************************
00960 ;
00970 INFLT    CP      '#'             ;INFLECTION CODE?
00980          JP      NZ,ERROR        ;NOT RECOGNIZABLE CHAR
00990          INC     IX              ;GET INFLECTION PITCH
01000          LD      A,(IX)          ;  TO A REG FOR TEST
01010          CP      '1'             ;1=NORMAL BASE
01020          JR      NZ,THREE        ;TEST FOR A 3
01030          LD      A,'O'           ;LOAD 1 CODE
01040          JR      CODE            ;GOTO CODE ROUTINE
01050 THREE    CP      '3'             ;IS CODE SET TO A 3?
01060          JR      NZ,FIVE         ;GO TEST FOR A 5
01070          LD      A,'T'           ;LOAD 3 CODE
01080          JR      CODE
01090 FIVE     CP      '5'             ;IS CODE SET TO A 5?
01100          JR      NZ,SEVEN        ;GO TEST FOR A 7
01110          LD      A,'F'           ;LOAD 5 CODE
01120          JR      CODE
01130 SEVEN    CP      '7'             ;IS CODE SET TO A 7
01140          JP      NZ,ERROR        ;ILLEGAL CODE ENTERED
01150          LD      A,'S'           ;LOAD 7 CODE
01160 ;
01170 CODE     LD      (IY),A          ;LOAD CODE TO OUTPUT BUFF
01180          INC     IY
01190          JP      G1              ;GET NEXT CHARACTER
01200 ;
01210 ;*******************
01220 ; PROGRAM VARIABLES
01230 ;*******************
01240 ;
01250 INFLEC   DEFB    1               ;INFLECT CODE SET TO 1
01260 LRFLAG   DEFB    0               ;USED TO CONTROL L/R ROUT
01270 SAVEIX   DEFW    00              ;SAVES CURR IX PTR
01280 IXPTR    DEFW    00              ;HOLDS CURR IX PTR
01290 ;
01300 ;*******************
01310 ; RULE SCAN ROUTINE
01320 ;*******************
01330 ;
01340 RULSCN   PUSH    IX
01350          POP     DE              ;MAKE DE PT TO STRT OF IX
01360 RULSN1   CP      (HL)            ;HL PTS TO RULE
01370          JR      Z,OK            ;DOUBLE CHECK
01380          JP      NEXTRL          ;RULE FAILURE..CHECK NEXT
01390 OK       INC     HL              ;PT TO NEXT RULE CH
01400          LD      A,(HL)          ;GET RULE CH
01410          CP      '<'             ;LEFT SYMBOL
01420          JP      Z,LEFT          ;GOTO LEFT ROUTINE
01430          CP      '>'             ;RIGHT SYMBOL
01440          JP      Z,RIGHT         ;GOTO RIGHT ROUTINE
01450          CP      '='             ;EQUAL MEANS MATCH
01460          JP      Z,TXFR          ;TRANS. DATA TO TALK BUFF
01470          INC     IX
01480          LD      A,(IX)          ;GET NEW STR. CH
01490          JR      RULSN1          ;SCAN AGAIN
01500 ;
01510 ;*******************
01520 ; LEFT SCAN ROUTINE
01530 ;*******************
01540 ;
01550 LEFT     INC     HL              ;SET FOR NEXT CH IN RULE
01560          PUSH    HL              ;SAVE HL
01570          LD      HL,LRFLAG       ;GET RIGHT/LEFT FLAG
01580          SET     0,(HL)          ;SET FOR LEFT SCAN
01590          POP     HL              ;RESTORE HL
01600          LD      A,(HL)          ;GET NEXT CH RULE
01610          CP      '!'
01620          JP      Z,NOALPH        ;GOTO BLANK ROUTINE
01630          CP      '#'
01640          JP      Z,VOWEL         ;GOTO CHECK VOWELS
01650          CP      ':'
01660          JP      Z,CONSTS        ;CHECK 0 OR MORE CONSNTS
01670          CP      '+'
01680          JP      Z,FRNTVL        ;CHECK FRONT VOWEL
01690          CP      '"'
01700          JP      Z,CONSNT        ;ROUTINE TO MATCH 1 CONS
01710          CP      '.'
01720          JP      Z,VCNSNT        ;MATCH VOICED CONSONANT
01730          CP      '&'
01740          JP      Z,SIBLNT        ;CHECK FOR SIBILANTS
01750          CP      '$'
01760          JP      Z,INFCON        ;CHECK INFLUENCING CONSNT
01770          CP      '>'
01780          JP      Z,RIGHT         ;PERFORM RIGHT ROUTINE
01790          CP      '='
01800          JP      Z,TXFR          ;TRANSFER DATA TO BUFFER
01810          JP      ASCCHK          ;CHECK FOR ASCII CHARACT
01820 ;
01830 ;*********************
01840 ; RIGHT SCAN ROUTINE
01850 ;*********************
01860 ;
01870 RIGHT    INC     HL              ;INC TO NEXT CH RULE
01880          PUSH    HL
01890          LD      HL,LRFLAG       ;GET LEFT/RIGHT FLAG
01900          BIT     4,(HL)          ;SET AT <>IX PTR TIME
01910          JR      NZ,RPASS        ;SAVE ONLY ONCE
01920          LD      (IXPTR),IX      ;SAVE PTR
01930          SET     4,(HL)
01940 RPASS    RES     0,(HL)          ;RESET FOR RIGHT SCAN
01950          POP     HL
01960          LD      A,(HL)          ;GET NEXT RULE CH
01970          CP      '!'
01980          JP      Z,NOALPH        ;GOTO BLANK ROUTINE
01990          CP      '#'
02000          JP      Z,VOWEL         ;CHECK FOR VOWELS
02010          CP      ':'
02020          JP      Z,CONSTS        ;CHK FOR 0 OR MORE CONS
02030          CP      '"'
02040          JP      Z,CONSNT        ;CHK FOR 1 CONS MATCH
02050          CP      '+'
02060          JP      Z,FRNTVL        ;MATCH FRONT VOWEL
02070          CP      '.'
02080          JP      Z,VCNSNT        ;CHK VOICED CONSONANTS
02090          CP      '%'
02100          JP      Z,SUFFIX        ;CHECK SUFFIXES
02110          CP      '='
02120          JP      Z,TXFR          ;PUT DATA TO TALK BUFFER
02130          JP      ASCCHK          ;CHK FOR ASCII CHARACTER
02140 ;
02150 ;*********************
02160 ; NON-ALPHA TEST = !
02170 ;*********************
02180 ;
02190 NOALPH   CALL    SIDE            ;RIGHT OR LEFT SIDE CALL?
02200          JR      NZ,LNO          ;LEFT RULE
02210          INC     IX              ;RIGHT RULE..NEXT STR
02220          LD      A,(IX)          ;GET STR. CH
02230          JR      RNO             ;GO COMPARE
02240 LNO      DEC     DE              ;SET TO BEFORE IX PTR
02250          LD      A,(DE)          ;GET CH
02260 RNO      SUB     64              ;TEST FOR NON-ALPHA
02270          JP      P,NEXTRL        ;GET NEXT RULE IF ANY
02280          CALL    SIDE            ;FIND OUT WHERE TO JUMP
```

Listing 2 continued

The INFLT routine assigns these values special characters inserted into the phoneme output buffer. Since these special inflection characters are greater in value than the largest phoneme code (3F hex), lines 5670–5690 test for values above or below 40 hex. The routine passes values below 40 hex as phoneme codes, and values greater than 40 hex as inflection codes.

At the proper inflection code, the Output routine loads the value into register A and sends it to port 18. This output latches the STB line and changes at the same time the inflection pitch. The pitch remains constant until the program sends a different inflection code to the Votrax chip.

Lines 5880 and 5890 make up the set-up time delay that I described earlier. The smallest value that you can send to the time delay subroutine at 60 hex is six. (Increase this value to 10 for a Model I computer. Anything less results in erratic conditions in the creation of speech.)

Since the Text-to-Speech program and the Rules Table are separate, you must link them somehow. The first two equates in Listing 2 take care of this. The CHTBLE value is the location where the Rules Table starts in memory and the ENDTBL value is the length of the Rules Table's index. The CHTYPE routine uses the CHTBLE value to find out where it should start scanning at the rules index. If you add more rules, change the ENDTBL value accordingly.

The Rules Index

Refer to the Rules Table (Listing 3). The equates at the beginning make entering the phoneme codes easier. I felt that entering standard phoneme codes was easier than entering the hex or decimal equivalents. You may notice that some of the codes have two-letter designations instead of just one. This is so the assembler can recognize the phoneme codes as equates instead of Assembly commands that would result in assembly errors.

The character index in Listing 3 is called CHTBLE. All of the characters that pertain to the rules are included in this table. If you want to add a new character that has no existing rules, insert it at the end just before the ENDTBL label. Examine the other rules as to how to add new characters and the appropriate rules.

The @ symbol between each rule indicates the end of the rule and the start of another one. If the next value is not an up arrow (or bracket), more rules

Listing 2 continued

```
02290           JP      Z,RIGHT         ;BACK TO RIGHT SCAN ROUT
02300           JP      LEFT            ;BACK TO LEFT SCAN ROUT
02310 ;
02320 ;*********************************************
02330 ; ROUTINE TO TEST FOR ONE OR MORE VOWELS = #
02340 ;*********************************************
02350 ;
02360 VOWEL     PUSH    HL              ;SAVE TABLE RULE POINTER
02370           LD      HL,VMESS        ;GET PARAMS
02380           LD      A,(VMLEN)       ;LENGTH
02390           LD      B,A             ;LOAD B WITH VMESS LENGTH
02400           CALL    SIDE            ;CHECK DIRECTION TO SCAN
02410           JR      NZ.LVOW         ;LEFT SIDE
02420           INC     IX              ;RIGHT RULE
02430           LD      A,(IX)          ;GET CH
02440           JR      RVOW            ;RIGHT DIRECTION
02450 LVOW      DEC     DE              ;SET TO BEFORE IX PTR
02460           LD      A,(DE)          ;GET CHARACTER
02470 RVOW      CP      (HL)            ;COMPARE WITH VMESS VALUE
02480           JP      Z,VCONT         ;MATCH?
02490           INC     HL              ;INC TO NEXT VMESS VALUE
02500           DJNZ    RVOW            ;GO BACK AND CHECK
02510           POP     HL              ;RESTORE RULE POINTER
02520           JP      NEXTRL          ;NO FOUND - NEXT RULE
02530 VCONT     POP     HL              ;FOUND MATCH
02540           CALL    SIDE            ;CHECK DIRECTION OF SCAN
02550           JP      Z,RIGHT         ;GO TO RIGHT SCAN
02560           JP      LEFT            ;GO TO LEFT SCAN
02570 ;
02580 VMESS     DEFM    'AEIOUY'        ;VOWELS
02590 VMLEN     DEFB    $-VMESS         ;LENGTH OF VMESS
02600 ;
02610 ;*****************************
02620 ; ZERO OR MORE CONSONANTS = :
02630 ;*****************************
02640 ;
02650 CONSTS    CALL    SIDE            ;CHECK SCAN DIRECTION
02660           JR      NZ.LCON         ;IS IT RIGHT OR LEFT?
02670           INC     IX              ;RIGHT..INC POINTER
02680           JR      RCON            ;GO CHECK
02690 LCON      DEC     DE              ;LEFT..DEC POINTER
02700 RCON      CALL    SIDE            ;CHECK SCAN DIRECTION
02710           JP      Z.RIGHT         ;GO RIGHT
02720           JP      LEFT            ;GO LEFT
02730 ;
02740 ;**************************
02750 ; MATCH ONE CONSONANT = "
02760 ;**************************
02770 ;
02780 CONSNT    LD      A,(CONLEN)      ;TBL LENGTH
02790           LD      B,A             ;B IS LOOP LENGTH
02800           CALL    SIDE            ;CHECK SCAN DIRECTION
02810           JR      NZ.LCONS        ;LEFT RULE
02820           INC     IX
02830           LD      A,(IX)          ;GET STR. CH
02840           JR      RCONS           ;GO RIGHT
02850 LCONS     DEC     DE              ;DEC TO LEFT
02860           LD      A,(DE)          ;VALUE BEFORE IX PTR
02870 RCONS     PUSH    HL              ;SAVE RULE PTR
02880           LD      HL,CONMES       ;STRT OF CONSONANTS
02890 CLOOP     CP      (HL)            ;COMPARE
02900           JR      Z.CMATCH        ;SKIP IF FOUND MATCH
02910           INC     HL              ;INC TO NEXT CONS
02920           DJNZ    CLOOP           ;CHECK NEXT VALUE
02930           POP     HL              ;RESTORE RULE PTR
02940           JP      NEXTRL          ;CHECK NEXT RULE
02950 CMATCH    POP     HL              ;RESTORE RULE PTR
02960           CALL    SIDE            ;CHECK DIRECTION
02970           JP      Z,RIGHT         ;GO RIGHT
02980           JP      LEFT            ;GO LEFT
02990 ;
03000 CONMES    DEFM    'BCDFGHJKLMNPQRSTVWXZ'  ;CONSONANTS
03010 CONLEN    DEFB    $-CONMES        ;CONMES LENGTH
03020 ;
03030 ;*******************************
03040 ; MATCH A VOICED CONSONANT = .
03050 ;*******************************
03060 ;
03070 VCNSNT    LD      A,(VCONLE)      ;TBLE LEN
03080           LD      B,A             ;SAVE LOOP COUNTER
03090           CALL    SIDE            ;CHECK DIRECTION
03100           JR      NZ,VCLFT        ;SKIP TO LEFT CHECK
03110           INC     IX              ;INC STRING POINTER
03120           LD      A,(IX)          ;GET VALUE
03130           JR      VCRT            ;GO TEST COMPARE
03140 VCLFT     DEC     DE              ;DEC LEFT POINTER
03150           LD      A,(DE)          ;GET CHARA
03160 VCRT      PUSH    HL              ;SAVE RULE POINTER
03170           LD      HL,VMESSL       ;GET STRT OF DATA
03180 VLOOP     CP      (HL)            ;MATCH?
03190           JR      Z.VMATCH        ;SKIP IF MATCH
03200           INC     HL              ;INC VMESSL POINTER
03210           DJNZ    VLOOP           ;GO BACK AND CHECK AGAIN
03220           POP     HL              ;RESTORE RULE PTR
03230           JP      NEXTRL          ;NO MATCH...NEXT RULE
03240 VMATCH    POP     HL              ;RESTORE RULE PTR
03250           CALL    SIDE            ;CHECK DIRECTION
03260           JP      Z,RIGHT         ;GO TO RIGHT SCAN
03270           JP      LEFT            ;BACK TO LEFT ROUTINE
03280 ;
03290 VMESSL    DEFM    'BDGJLMNRVWZ'   ;VOICED CONSONANTS
03300 VCONLE    DEFB    $-VMESS         ;LENGTH OF VMESSL
03310 ;
03320 ;************************
03330 ; MATCH FRONT VOWEL = +
03340 ;************************
03350 ;
03360 FRNTVL    CALL    SIDE            ;CHECK DIRECTION
03370           JR      NZ,FVL          ;LEFT RULE
03380           INC     IX              ;PT TO NEXT CH IN STR
03390           LD      A,(IX)          ;GET CH
03400           JR      FVR             ;CHECK RIGHT SIDE
03410 FVL       DEC     DE              ;DEC LEFT POINTER
03420           LD      A,(DE)          ;GET VALUE TO CHECK
03430 FVR       CP      'E'             ;CHECK FOR FRONT VOWLS
03440           JR      Z,FVMTCH
03450           CP      'I'
03460           JR      Z,FVMTCH
03470           CP      'Y'
03480           JR      Z.FVMTCH
03490           JP      NEXTRL
03500 FVMTCH    CALL    SIDE            ;CHECK DIRECTION
03510           JP      Z,RIGHT         ;GO RIGHT
03520           JP      LEFT            ;GO LEFT
03530 ;
03540 ;************************
03550 ; TEST FOR SUFFIX'S = %
03560 ;************************
03570 ;
03580 SUFFIX    PUSH    HL              ;RIGHT SIDE SCAN ONLY
03590           INC     IX              ;INC STRING POINTER
03600           PUSH    IX              ;SAVE IT
03610           LD      A,(SMESSL)      ;GET LENGTH
03620           LD      B,A             ;SET LOOP COUNTER
03630           LD      HL,SMESS        ;GET PARAM STRT
03640 SLOOP1    LD      A,(IX)          ;GET CHAR
03650           CP      (HL)            ;COMPARE VALUE
03660           JR      Z,SECCHK        ;GO CHK 2ND VALU IF THERE
03670 SLOOP2    DEC     B               ;DEC LOOP COUNTER
03680           JR      Z.ROUTER        ;NO MATCH
03690           INC     HL              ;INC SMESS POINTER
03700           LD      A,(HL)          ;GET NEXT VALUE
03710           CP      ','             ;NEXT SET OF VALUES?
03720           JR      NZ.SLOOP2       ;GO TEST AGAIN
03730           POP     IX              ;RESTORE POINTER LOCATION
03740           PUSH    IX              ;SAVE IT AGAIN
03750           INC     HL              ;INC SMESS POINTER
03760           JR      SLOOP1          ;GO BACK & CHECK AGAIN
03770 SECCHK    INC     HL              ;CHECK FOR FINAL MATCH
03780           LD      A,(HL)          ;GET VALUE
03790           CP      ','             ;MATCH?
03800           JR      Z,FNDMTH        ;SKIP IF FOUND MATCH
03810           INC     IX              ;INC STRINP POINTER
03820           DJNZ    SLOOP1          ;CHECK NEXT VALUE
03830 ;
03840 ROUTER    POP     IX              ;RESTORE INPUT BUFF PTR
03850           POP     HL              ;RESTORE RULE TABLE PTR
03860           JP      NEXTRL          ;NEXT RULE
03870 FNDMTH    POP     BC              ;BIT BUCKET..THROW AWAY
03880           POP     HL              ;RESTORE RULE TABLE PTR
03890           JP      RIGHT           ;NEXT CHAR
03900 ;
03910 SMESS     DEFM    'E,ER,ES,ED.ING,ELY.'   ;SUFFIX'S
03920 SMESSL    DEFB    $-SMESS         ;LENGTH OF SMESS
03930 ;
03940 ;**********************
03950 ; CHECK SIBILANTS = &
03960 ;**********************
03970 ;
03980 SIBLNT    PUSH    HL              ;SAVE RULE PTR
03990           LD      HL,SIBMES       ;GET STRT OF TABLE
04000           LD      A,(SIBLEN)      ;GET LENGTH
04010           LD      B,A             ;SAVE IT
04020           CALL    SIBCON          ;GO AND COMPARE
04030           POP     HL              ;RESTORE RULE PTR
04040           JP      NZ.NEXTRL       ;SKIP IF NO MATCH
04050           JP      LEFT            ;MATCH..GO LEFT
04060 ;
04070 ;**********************
04080 ; INFLUENCING CONS = $
04090 ;**********************
04100 ;
04110 INFCON    PUSH    HL              ;SAVE RULE PTR
04120           LD      HL,CONSM        ;GET TABLE LOCATION
04130           LD      A,(CONSLN)      ;GET LENGTH OF TABLE
04140           LD      B,A             ;SET UP LOOP COUNTER
04150           CALL    SIBCON          ;GO CHECK FOR MATCH
04160           POP     HL              ;RESTORE RULE POINTER
04170           JP      NZ.NEXTRL       ;NO MATCH...NEXT RULE
04180           JP      LEFT            ;MATCH...GO LEFT SCAN
04190 ;
04200 ;*************************************************
04210 ; COMMON ROUTINE FOR SIBILANT & INFLUENCING CONS CHECK
04220 ;*************************************************
04230 ;
04240 SIBCON    DEC     DE              ;DEC LEFT POINTER
04250           PUSH    DE              ; AND SAVE IT
04260 SIBLOP    LD      A,(DE)          ;GET VALUE
04270           CP      (HL)            ;MATCH WITH TABLE VALUE?
04280           JR      Z,CHKAGN        ;SKIP IF NO AND CHK AGAIN
04290 SIB2      DEC     B               ;DEC LOOP COUNTER
04300           JR      Z.SIBERR        ;ZERO = NO MATCH
04310           INC     HL              ;INC TABLE POINTER
04320           LD      A,(HL)          ;GET VALUE
04330           CP      ','             ;READY FOR NEXT VALUE?
04340           JR      NZ.SIB2         ;SKIP IF NOT
04350           POP     DE              ;RESTORE LEFT POINTER
04360           PUSH    DE              ;SAVE IT AGAIN
04370           INC     HL              ;INC TABLE POINTER
04380           JR      SIBLOP          ;CHECK AGAIN
04390 CHKAGN    INC     HL              ;INC TABLE POINTER
04400           LD      A,(HL)          ;GET VALUE
04410           CP      ','
04420           JR      Z,FNDIT         ;IF HERE = VALUE FOUND
04430           DEC     DE              ;CHK FOR 2ND CHARCTER
04440           DJNZ    SIBLOP          ;GO BACK AND CHECK
04450 ;
04460 SIBERR    POP     DE              ;RESTORE LEFT POINTER
04470           SET     7,A             ;SET ERROR BIT.
04480           BIT     7,A             ;TEST FOR JUMP ON RETURN
04490           RET                     ;RETURN TO CALLER
04500 FNDIT     POP     DE              ;BIT BUCKET
04510           RES     7,A             ;RESET ERROR BIT
04520           BIT     7,A             ;TEST FOR JUMP ON RETURN
04530           RET                     ;RETURN TO CALLER
04540 ;
04550 SIBMES    DEFM    'S,C.G,Z,X,J,HC,HS,'    ;&..(BACKWARDS)
04560 SIBLEN    DEFB    $-SIBMES        ;LENGTH OF SIBMES
04570 ;
04580 CONSM     DEFM    'T,S,R,D,L,Z,N,J,HT,HC.HS,'     ;INFL CON
```

Listing 2 continued

Listing 2 continued

```
04590 CONSLN  DEFB    $-CONSM          ;LENGTH OF CONSM
04600 ;
04610 ;****************************************
04620 ; CHECK FOR RIGHT OR LEFT ASCII CHARACTER
04630 ;****************************************
04640 ;
04650 ASCCHK  CALL    SIDE             ;CHECK SCAN DIRECTION
04660         JR      NZ,LASC          ;LEFT SIDE
04670         INC     IX               ;INC STRING POINTER
04680         LD      A,(IX)           ;GET CH RIGHT SIDE
04690         JR      CHECK            ;GO CHECK VALUES
04700 LASC    DEC     DE               ;DEC LEFT POINTER
04710         LD      A,(DE)           ;GET CH LEFT SIDE
04720 CHECK   CP      (HL)             ;INPUT AND RULE MATCH?
04730         JP      NZ.NEXTRL        ;NZ = NO MATCH
04740         CALL    SIDE             ;CHECK SCAN DIRECTION
04750         JP      Z,RIGHT          ;GO RIGHT SCAN ROUTINE
04760         JP      LEFT             ;GO LEFT SCAN ROUTINE
04770 ;
04780 ;**************************************************
04790 ; RIGHT OR LEFT TEST TO CHECK SCAN ROUTINE DIRECTION
04800 ;**************************************************
04810 ;
04820 SIDE    LD      A,(LRFLAG)       ;GET CONTROL WORD
04830         BIT     0,A              ;TEST BIT 0
04840         RET                      ;RETURN CALL
04850 ;
04860 ;*****************************
04870 ; SET PTRS FOR NEXT RULE SCAN
04880 ;*****************************
04890 ;
04900 NEXTRL  NOP                      ;JUMP ENTRY POINT
04910 NLOOP   LD      A,(HL)           ;GET CH
04920         CP      '@'              ;FIND RULE SEPERATOR
04930         JR      Z,FNDNXT         ;FOUND NEW RULE
04940         CP      '['              ;MEANS END OF THESE RULES
04950         JP      Z,ERROR          ;ERROR...FOUND NO MATCH
04960         INC     HL               ;INC RULE POINTER
04970         JR      NLOOP            ;CHECK NEXT
04980 ;
04990 FNDNXT  XOR     A                ;CLEAR A REGISTOR
05000         LD      (LRFLAG),A       ;CLR FLAG WORD
05010         LD      IX,(SAVEIX)      ;RESTORE IX FOR NEXT RULE
05020         LD      A,(IX)           ;GET CH
05030         INC     HL               ;SET PTR PAST @ MARKER
05040         JP      RULSCN           ;CHECK NEXT RULE
05050 ;
05060 ;****************************
05070 ; TXFR DATA TO TALK BUFFER
05080 ;****************************
05090 ;
05100 TXFR    INC     HL               ;SET PTR PAST = SIGN
05110 TLOOP   LD      A,(HL)           ;GET VALUE
05120         CP      '@'              ;DONE TXFR?
05130         JR      Z.GNXTCH         ;Z= TEST FOR DONE
05140         LD      (IY).A           ;PUT DATA TO TALK BUFF
05150         INC     IY               ;INC TALK BUFF POINTER
05160         INC     HL               ;PT TO NEXT RULE DATA
05170         JR      TLOOP            ;DO LOOP AGAIN
05180 ;
05190 GNXTCH  LD      A,(LRFLAG)       ;GET FLAG CONTROL WORD
05200         BIT     4,A              ;IS BIT 4 SET?
05210         JR      Z,G1             ;ZERO = SKIP IX RESTORE
05220         LD      IX,(IXPTR)       ;RESTORE PTR BE4 < OR >
05230 G1      INC     IX               ;INC TO NEXT VALUE
05240         XOR     A                ;CLR A REG
05250         CP      (IX)             ;COMPARE A & IX FOR 00
05260         JR      Z,OUTPUT         ;ZERO = DONE DECODING
05270         LD      (LRFLAG),A       ;CLR FLAG FOR NEXT RULE
05280         JP      CHTYPE           ;GET NEXT STR CH RULE
05290 ;
05300 ;***************
05310 ; ERROR ROUTINE
05320 ;***************
05330 ;
05340 ERROR   LD      A,0DH            ;LOAD CARRIGE CODE
05350         CALL    33H              ;MOVE CURSOR DWN 1 LINE
05360         INC     IX               ;INC INPUT BUFF PTR
05370         LD      (IX),A           ;PUT ETX AT END (0D)
05380         LD      HL,INPBUF        ;GET STRT OF BUFF
05390         CALL    21BH             ;MOVE INP DATA TO SCREEN
05400 ;
05410         LD      HL,OUTBUF        ;GET OUTBUF LOCATION
05420         LD      DE,(4020H)       ;CURSOR POS TO PRINT ERR
05430         LD      BC.255           ;# MAX BYTES THAT CAN MOV
05440 ERLOOP  LDI                      ;MOVE OUT DATA TO SCREEN
05450         LD      A,3FH            ;STOP CODE
05460         CP      (HL)             ;COMPARE FOR STOP CODE
05470         JR      NZ.ERLOOP        ;MOVE NEXT CHARACTER
05480         LD      A,0DH            ;LOAD CARRIAGE CODE
05490         CALL    33H              ;MOVE CURSOR DWN 1 LINE
05500 ERR     LD      HL,ERRMES        ;GET ERROR DATA
05510         JP      OUT1             ;OUTPUT ERROR MESSAGE
05520 ;
05530 ERRMES  DEFW    0000H            ;ERROR PHONEME CODES
05540         DEFW    2B00H            ;  USED TO SAY = ERROR.
05550         DEFW    3E3AH
05560         DEFW    3F3EH
05570 ;
05580 ;**********************
05590 ; OUTPUT DATA TO SC-01
05600 ;**********************
05610 ;
05620 OUTPUT  LD      HL,OUTBUF        ;GET STRT OF OUTPUT BUFF
05630 OUT1    LD      A,16
05640         OUT     (236),A          ;TURN ON EXTERNAL BUS
05650 OLOOP   LD      A,144            ;SET UP 8255 FOR MODE 0
05660         OUT     (19),A           ;  OPTION NUMBER 8
05670         LD      A,(HL)           ;GET PHONEME CODE
05680         CP      40H              ;TEST TOP OF PHONEME CODE
05690         JP      M,NOINFL         ; FOR INFLECTION VALUES
05700         CP      'O'              ;IS 1 INFLECTION?
05710         JR      NZ.I3            ;GO TEST FOR 3 CODE
05720         LD      A,1              ;SET TO 1
05730         JR      SETINF           ;GO STORE VALUE
05740 I3      CP      'T'              ;IS 3 INFLECTION?
05750         JR      NZ,I5            ;GO TEST FOR 5 CODE
05760         LD      A,3              ;SET TO 3
05770         JR      SETINF           ;GO STORE VALUE
05780 I5      CP      'F'              ;IS 5 INFLECTION?
05790         JR      NZ.I7            ;GO TEST FOR 7 CODE
05800         LD      A,5              ;SET TO 5
05810         JR      SETINF           ;GO STORE VALUE
05820 I7      CP      'S'              ;IS 7 INFLECTON?
05830         JR      NZ,OL1           ;SKIP AND IGNORE VALUE
05840         LD      A,7              ;SET TO 7
05850 SETINF  LD      (INFLEC).A       ;STORE INFLECTION CODE
05860         JR      OL1              ;GO GET NEXT BUFF VALUE
05870 NOINFL  OUT     (17).A           ;SEND PHONEME TO 8255
05880         LD      BC,0006H         ;SET UP TIME DELAY
05890         CALL    60H              ; TO ALLOW SET-UP TIME
05900         LD      A,(INFLEC)       ;GET INFLECTION CODE
05910         OUT     (18),A           ;TELL SC-01 TO DO IT
05920 RESCAN  IN      A,(16)           ;READ A/R LINE OF 8255
05930         BIT     0,A              ;IS BIT SET = DONE
05940         JR      Z,RESCAN         ;GO BACK AND SCAN AGAIN
05950         LD      A,(HL)           ;GET PHONEME VALUE AGAIN
05960         CP      3FH              ;IS IT THE STOP CHARACT?
05970         JR      Z,RETURN         ;ZERO = DONE & EXIT
05980 OL1     INC     HL               ;INC TO NEXT PHONEME VALU
05990         JR      OLOOP            ;GO GET NEXT PHONEME CODE
06000 RETURN  XOR     A                ;CLEAR A REG
06010         LD      (LRFLAG),A       ;CLEAR CONTROL WORD
06020         POP     IY               ;RESTORE ALL REGISTORS
06030         POP     IX
06040         POP     AF
06050         POP     BC
06060         POP     DE
06070         POP     HL
06080         RET
06090 ;
06100 ;***************
06110 ; TALK BUFFERS
06120 ;***************
06130 ;
06140 INPBUF  DEFS    255              ;INPUT STRING BUFFER
06150         DEFB    0                ;SAFETY STOP CHARACTER
06160 OUTBUF  DEFS    400              ;OUTPUT BUFFER
06170         DEFB    3FH              ;SAFETY STOP CHARACTER
06180 ;
06190         END     VOTRAX
```

follow. The up arrow signals the end of a specific group of rules to the TTSPRG program.

Many of these rules were developed by the Naval Research Laboratory (NRL). You can find this report in "Automatic Translation of English Text to Phonetics by Means of Letter to Sound Rules" by Honey Sue Elovitz et al., United States Naval Research Laboratory Report Number 7948, 1976.

I had to modify a few of these rules to fit this application. It appears that the new Votrax system has more phonemes for better pronunciation purposes than the old Votrax system did when the NRL originally designed the rules. I rewrote some rules to enhance pronunciation.

I also added many rules to increase the capability of creating speech. The Rules Table now comprises 416 rules. I will add more in the future. To help you, the beginning of each rule has either a commented number or semicolon to the right of it. I number the rules in increments of five.

As previously mentioned, the Rules Table guides the Text-to-Speech program in its decoding routines by using special symbols to control the program's operations. If all the symbols and codes match, and the TTSPRG program finds an equals sign, the rule is complete and the program inserts the following code into the output buffer until it reaches the @ symbol. The program returns to CHTYPE and decodes the next input string character if there is one.

Keep in mind that my Rules Table isn't 100 percent accurate, nor do any exist that are. You'll find words that the Votrax can't pronounce exactly, but they'll be close and recognizable. For instance, one example pertains to the long and short *i* sounds. Currently, these rules make the board pronounce the word "like" properly, but it pronounces "life" with a short *i* sound.

Program Listing 3. Rules Table source code. The phoneme codes begin in line 85. The length of the index is given in line 211. The rule table starts in line 213.

```
00001 ;***********************
00002 ;   L I S T I N G  3
00003 ;***********************
00004 ;
00005 ;  RULE TABLE PROGRAM
00006 ;          BY
00007 ;   DAVID ENGELHARDT
00010 ;
00011 ;***********************
00012 ;
00013           ORG     0F000H
00014 ;
00015 EH3       EQU     0
00016 EH2       EQU     1
00017 EH1       EQU     2
00018 PA0       EQU     3
00019 DT        EQU     4
00020 A2        EQU     5
00021 A1        EQU     6
00022 ZH        EQU     7
00023 AH2       EQU     8
00024 I3        EQU     9
00025 I2        EQU     10
00026 I1        EQU     11
00027 MM        EQU     12
00028 N         EQU     13
00029 BB        EQU     14
00030 V         EQU     15
00031 CH        EQU     16
00032 SH        EQU     17
00033 ZZ        EQU     18
00034 AW1       EQU     19
00035 NG        EQU     20
00036 AH1       EQU     21
00037 OO1       EQU     22
00038 OO        EQU     23
00039 LL        EQU     24
00040 K         EQU     25
00041 J         EQU     26
00042 HH        EQU     27
00043 G         EQU     28
00044 F         EQU     29
00045 DD        EQU     30
00046 S         EQU     31
00047 AA        EQU     32
00048 AY        EQU     33
00049 Y1        EQU     34
00050 UH3       EQU     35
00051 AH        EQU     36
00052 PP        EQU     37
00053 O         EQU     38
00054 II        EQU     39
00055 U         EQU     40
00056 Y         EQU     41
00057 T         EQU     42
00058 RR        EQU     43
00059 EE        EQU     44
00060 W         EQU     45
00061 AE        EQU     46
00062 AE1       EQU     47
00063 AW2       EQU     48
00064 UH2       EQU     49
00065 UH1       EQU     50
00066 UH        EQU     51
00067 O2        EQU     52
00068 O1        EQU     53
00069 IU        EQU     54
00070 U1        EQU     55
00071 THV       EQU     56
00072 TH        EQU     57
00073 ER        EQU     58
00074 EH        EQU     59
00075 E1        EQU     60
00076 AW        EQU     61
00077 PA1       EQU     62
00078 STOP      EQU     63
00079 ;
00080 ;************************
00081 ; CHARACTER LOOKUP TABLE
00082 ;************************
00083 ;
00084 CHTBLE    DEFS    3
00085           DEFB    'A'
00086           DEFW    AR
00087           DEFB    'B'
00088           DEFW    BR
00089           DEFB    'C'
00090           DEFW    CR
00091           DEFB    'D'
00092           DEFW    DR
00093           DEFB    'E'
00094           DEFW    ERULES
00095           DEFB    'F'
00096           DEFW    FR
00097           DEFB    'G'
00098           DEFW    GR
00099           DEFB    'H'
00100           DEFW    HR
00101           DEFB    'I'
00102           DEFW    IR
00103           DEFB    'J'
00104           DEFW    JR
00105           DEFB    'K'
00106           DEFW    KR
00107           DEFB    'L'
00108           DEFW    LR
00109           DEFB    'M'
00110           DEFW    MR
00111           DEFB    'N'
00112           DEFW    NR
00113           DEFB    'O'
00114           DEFW    OR
00115           DEFB    'P'
00116           DEFW    PR
00117           DEFB    'Q'
00118           DEFW    QR
00119           DEFB    'R'
00120           DEFW    RRULES
00121           DEFB    'S'
00122           DEFW    SR
00123           DEFB    'T'
00124           DEFW    TR
00125           DEFB    'U'
00126           DEFW    UR
00127           DEFB    'V'
00128           DEFW    VR
00129           DEFB    'W'
00130           DEFW    WR
00131           DEFB    'X'
00132           DEFW    XR
00133           DEFB    'Y'
00134           DEFW    YR
00135           DEFB    'Z'
00136           DEFW    ZR
00137           DEFB    ' '
00138           DEFW    BLANK
00139           DEFB    '.'
00140           DEFW    PERIOD
00141           DEFB    39
00142           DEFW    APOST
00143           DEFB    ','
00144           DEFW    COMMA
00145           DEFB    '?'
00146           DEFW    QUEST
00147           DEFB    '0'
00148           DEFW    ZERO
00149           DEFB    '1'
00150           DEFW    R1
00151           DEFB    '2'
00152           DEFW    R2
00153           DEFB    '3'
00154           DEFW    R3
00155           DEFB    '4'
00156           DEFW    R4
00157           DEFB    '5'
00158           DEFW    R5
00159           DEFB    '6'
00160           DEFW    R6
00161           DEFB    '7'
00162           DEFW    R7
00163           DEFB    '8'
00164           DEFW    R8
00165           DEFB    '9'
00166           DEFW    R9
00167           DEFB    '-'
00168           DEFW    MINUS
00169           DEFB    '+'
00170           DEFW    PLUS
00171           DEFB    '='
00172           DEFW    EQUALS
00173           DEFB    '*'
00174           DEFW    MULT
00175           DEFB    '/'
00176           DEFW    DIVIDE
00177           DEFB    8
00178           DEFW    LARROW
00179           DEFB    9
00180           DEFW    RARROW
00181           DEFB    91
00182           DEFW    UARROW
00183           DEFB    10
00184           DEFW    DARROW
00185           DEFB    31
00186           DEFW    CLEAR
00187           DEFB    '<'
00188           DEFW    LTHAN
00189           DEFB    '>'
00190           DEFW    GTHAN
00191           DEFB    ':'
00192           DEFW    COLON
00193           DEFB    ';'
00194           DEFW    SEMICO
00195           DEFB    '!'
00196           DEFW    EXCLA
00197           DEFB    '"'
00198           DEFW    QUOTE
00199           DEFB    '$'
00200           DEFW    DOLLAR
00201           DEFB    '%'
00202           DEFW    PERCNT
00203           DEFB    '&'
00204           DEFW    ANDD
00205           DEFB    '('
00206           DEFW    LBRACK
00207           DEFB    ')'
00208           DEFW    RBRACK
00209           DEFB    '@'
00210           DEFW    ATSYM
00211 ENDTBL    DEFB    $-CHTBLE
00212 ;
00213 ;*************
00214 ; RULE TABLES
00215 ;*************
00216 ;
00217 AR        DEFM    'A<!>!='        ;1
00218           DEFB    A1
00219           DEFB    AY
00220           DEFB    Y
00221           DEFB    '@'
```

Listing 3 continued

Sometimes the board pronounces *y* as in "why," as opposed to the correct *ies* ending.

You can add rules to correct this; however, adding rules is tricky and you should take great care. Remember that what holds true for one expression may not hold true for another. Feel free to change some of the spelling in words to obtain accurate results in speech creation. For example, you can get the same pronunciation by spelling TODAY as 2DAY.

Hello Again

Now use all of this theory to convert the word HELLO as you did in last month's article. A Basic program calls the Text-to-Speech program via a USR function call. The USR command passes a free parameter obtained from the VARPTR(B$) command that points to the string's memory location.

On entry, the Text-to-Speech program conditions both the input and output buffers as I described earlier. It then fills the first location in INPBUF with a blank and converts any lowercase letters to uppercase. The program then transfers the entire string to the input buffer. INPBUF now has the word HELLO in it.

Register pair pointers are dedicated for use in controlling the Text-to-Speech program. The HL pair points to the conversion rules. The IX register pair points to the input buffer and the IY register pair points to the output buffer. The program uses register pair DE for left side scan manipulations and keeps the BC pair as extras.

On entry to the CHTYPE routine, HL points to the Rules Table's index and IX points to the first input string character in INPBUF, a blank rule located on line 138 in CHTBLE (Listing 3). CHTYPE then loads the HL register pair with the next two values that point to the blank rule's memory location. The program then makes a jump to the RULSCN routine at line 1340 in Listing 2. Refer to the located blank rule, number 375 in Listing 3.

RULSCN first double-checks to ensure that the value IX points to is the same as the rule's HL value. Since both values are blanks, the HL pointer increments to the next value in the rule, the equals sign. RULSCN scans the new HL value for a left or right direction pointer and then an equals sign. An equals sign indicates a rule match and the routine jumps to TXFR.

TXFR increments the HL pointer to the value past the equals sign, 03 (PA0). It transfers the value of 03 to

Listing 3 continued

```
00222          DEFM    'A<!>H='        ;
00223          DEFB    UH2
00224          DEFB    '@'
00225          DEFM    'A>!='          ;
00226          DEFB    UH2
00227          DEFB    '@'
00228          DEFM    'ARE>!='        ;
00229          DEFB    AH
00230          DEFB    ER
00231          DEFB    RR
00232          DEFB    '@'
00233          DEFM    'AR<!>O='       ;5
00234          DEFB    UH2
00235          DEFB    RR
00236          DEFB    '@'
00237          DEFM    'AR>#='         ;
00238          DEFB    EH2
00239          DEFB    RR
00240          DEFB    '@'
00241          DEFM    'AS<"!>#='      ;
00242          DEFB    AE1
00243          DEFB    EH3
00244          DEFB    S
00245          DEFB    '@'
00246          DEFM    'A>WA='         ;
00247          DEFB    UH2
00248          DEFB    '@'
00249          DEFM    'AW='           ;
00250          DEFB    AW
00251          DEFB    '@'
00252          DEFM    'ANY<:!='       ;10
00253          DEFB    EH2
00254          DEFB    EH2
00255          DEFB    N
00256          DEFB    Y
00257          DEFB    '@'
00258          DEFM    'AGAIN='        ;
00259          DEFB    UH1
00260          DEFB    G
00261          DEFB    A2
00262          DEFB    EH1
00263          DEFB    N
00264          DEFB    '@'
00265          DEFM    'ALLY<:#='      ;
00266          DEFB    UH2
00267          DEFB    LL
00268          DEFB    EE
00269          DEFB    '@'
00270          DEFM    'AL<!>#='       ;
00271          DEFB    UH1
00272          DEFB    LL
00273          DEFB    '@'
00274          DEFM    'A>"+#='        ;
00275          DEFB    AA
00276          DEFB    AY
00277          DEFB    '@'
00278          DEFM    'AG<:#>E='      ;15
00279          DEFB    I1
00280          DEFB    DD
00281          DEFB    J
00282          DEFB    '@'
00283          DEFM    'A>"+:#='       ;
00284          DEFB    AE
00285          DEFB    '@'
00286          DEFM    'A<:!>"+!='     ;
00287          DEFB    AA
00288          DEFB    AY
00289          DEFB    '@'
00290          DEFM    'A>"%='         ;
00291          DEFB    AA
00292          DEFB    AY
00293          DEFB    '@'
00294          DEFM    'ARR<!='        ;
00295          DEFB    UH1
00296          DEFB    RR
00297          DEFB    '@'
00298          DEFM    'ARR='          ;20
00299          DEFB    AE1
00300          DEFB    EH3
00301          DEFB    '@'
00302          DEFM    'AR<:!>!='      ;
00303          DEFB    AH1
00304          DEFB    UH2
00305          DEFB    ER
00306          DEFB    '@'
00307          DEFM    'AR>!='         ;
00308          DEFB    ER
00309          DEFB    '@'
00310          DEFM    'AR='           ;
00311          DEFB    AH1
00312          DEFB    RR
00313          DEFB    '@'
00314          DEFM    'AIR='          ;
00315          DEFB    EH2
00316          DEFB    EH2
00317          DEFB    RR
00318          DEFB    '@'
00319          DEFM    'AI='           ;25
00320          DEFB    AA
00321          DEFB    AY
00322          DEFB    '@'
00323          DEFM    'AY='           ;
00324          DEFB    AA
00325          DEFB    AY
00326          DEFB    '@'
00327          DEFM    'AU='           ;
00328          DEFB    AW
00329          DEFB    '@'
00330          DEFM    'AL<:#>!='      ;
00331          DEFB    UH
00332          DEFB    LL
00333          DEFB    '@'
00334          DEFM    'ALS<:#>!='     ;
00335          DEFB    UH
00336          DEFB    LL
00337          DEFB    ZZ
00338          DEFB    '@'
00339          DEFM    'ALK='          ;30
00340          DEFB    AW
00341          DEFB    K
00342          DEFB    '@'
00343          DEFM    'AL>"='         ;
00344          DEFB    AW1
00345          DEFB    UH3
00346          DEFB    LL
00347          DEFB    '@'
00348          DEFM    'ABLE<!>!='     ;
00349          DEFB    A1
00350          DEFB    Y
00351          DEFB    BB
00352          DEFB    UH3
00353          DEFB    LL
00354          DEFB    '@'
00355          DEFM    'ABLE<:!='      ;
00356          DEFB    A1
00357          DEFB    Y
00358          DEFB    BB
00359          DEFB    UH3
00360          DEFB    LL
00361          DEFB    '@'
00362          DEFM    'ABLE='         ;
00363          DEFB    UH2
00364          DEFB    BB
00365          DEFB    UH2
00366          DEFB    LL
00367          DEFB    '@'
00368          DEFM    'ANG>+='        ;35
00369          DEFB    A1
00370          DEFB    AY
00371          DEFB    Y
00372          DEFB    N
00373          DEFB    DD
00374          DEFB    J
00375          DEFB    '@'
00376          DEFM    'A='            ;
00377          DEFB    AE1
00378          DEFB    '@'
00379          DEFB    '['
00380 ;
00381 BR       DEFM    'B<!>!='        ;
00382          DEFB    BB
00383          DEFB    E1
00384          DEFB    Y
00385          DEFB    '@'
00386          DEFM    'BE<!>"#='      ;
00387          DEFB    BB
00388          DEFB    Y
00389          DEFB    '@'
00390          DEFM    'BEING='        ;
00391          DEFB    BB
00392          DEFB    E1
00393          DEFB    I2
00394          DEFB    NG
00395          DEFB    '@'
00396          DEFM    'BOTH<!>!='     ;40
00397          DEFB    BB
00398          DEFB    O2
00399          DEFB    O2
00400          DEFB    TH
00401          DEFB    '@'
00402          DEFM    'BUS<!>#='      ;
00403          DEFB    BB
00404          DEFB    I3
00405          DEFB    I3
00406          DEFB    ZZ
00407          DEFB    '@'
00408          DEFM    'BUIL='         ;
00409          DEFB    BB
00410          DEFB    I2
00411          DEFB    I2
00412          DEFB    LL
00413          DEFB    '@'
00414          DEFM    'B='            ;
00415          DEFB    BB
00416          DEFB    '@'
00417          DEFB    '['
00418 ;
00419 CR       DEFM    'C<!>!='        ;
00420          DEFB    S
00421          DEFB    E1
00422          DEFB    Y
00423          DEFB    '@'
00424          DEFM    'CH<!>"='       ;45
00425          DEFB    K
00426          DEFB    '@'
00427          DEFM    'CH<E"='        ;
00428          DEFB    K
00429          DEFB    '@'
00430          DEFM    'CH='           ;
00431          DEFB    T
00432          DEFB    CH
00433          DEFB    '@'
00434          DEFM    'CI<S!>#='      ;
00435          DEFB    S
00436          DEFB    AH1
00437          DEFB    I3
00438          DEFB    '@'
00439          DEFM    'CI>A='         ;
00440          DEFB    SH
00441          DEFB    '@'
00442          DEFM    'CI>O='         ;50
00443          DEFB    SH
00444          DEFB    '@'
00445          DEFM    'CI>EN='        ;
00446          DEFB    SH
00447          DEFB    '@'
00448          DEFM    'C>+S!='        ;
00449          DEFB    S
00450          DEFB    II
```

Listing 3 continued

OUTBUF, to which the IY pair points, and increments HL again. TXFR tests for the @ symbol, which signifies the end. This symbol makes the program increment the IX pointer (INPBUF) and jump back to the CHTYPE routine to check the next character in the word (HELLO).

The next character rules the program locates are the H rules. The TTSPRG locates the beginning of the H rules at line 786 in the Rules Table. With the HL pointing at rule 132, it double-checks via TTSPRG and HL then increments to the letter A in HAV. The program then increments the IX pointer and the routine tests for a match. Since the A in HAV and the E in HELLO don't match, control passes to the NEXTRL routine.

The NEXTRL routine increments the HL pointer in a loop searching for the @ symbol. If it finds the up-arrow (or left bracket) symbol, it jumps to the Error routine since no rule exists for this character. If the routine finds the @ symbol, it restores the IX pair and points it to the H character. Next, it increments HL to the next rule's first character and goes back to RULSCN to test this rule. This operation continues until rule 136 matches.

Notice that this rule only has the right scan symbol (>), so control goes to the Right Scan routine. This routine increments the HL register and checks the symbol after the > symbol. The next value is a number symbol, so the routine jumps to the Vowel routine. You may notice that the special symbol for a vowel (#) is the same as that for the inflection symbol. This doesn't present a problem because the program tests vowels and inflections at different points.

The Vowel routine saves the IX pointer and increments it to test the next character after the H for a vowel. It matches because the E after H in HELLO is a vowel. The match passes control back to the Right Scan routine, which checks the next character after the # symbol. The Right Scan routine finds the equals sign, indicating another match. The equated phoneme code value HH transfers to the output buffer and the routine jumps back to CHTYPE to check the next character (the E in HELLO).

Now the program scans the E rules to find rule 198 (ELL = EH1,UH3,LL, UH3). RULSCN doesn't find any special characters so it checks the ASCII values. This routine bumps the IX string pointer and compares the L in the rule with the L in the string. Since a

Listing 3 continued

```
00451           DEFB    ZZ
00452           DEFB    '@'
00453           DEFM    'C>+='          ;
00454           DEFB    S
00455           DEFB    '@'
00456           DEFM    'CK='           ;
00457           DEFB    K
00458           DEFB    '@'
00459           DEFM    'COM>%='        ;55
00460           DEFB    K
00461           DEFB    UH
00462           DEFB    MM
00463           DEFB    '@'
00464           DEFM    'CON<!='        ;
00465           DEFB    K
00466           DEFB    UH1
00467           DEFB    N
00468           DEFB    '@'
00469           DEFM    'C='            ;
00470           DEFB    K
00471           DEFB    '@'
00472           DEFB    '['
00473 ;
00474 DR        DEFM    'D<!>!='        ;
00475           DEFB    DD
00476           DEFB    El
00477           DEFB    Y
00478           DEFB    '@'
00479           DEFM    'DED<:#>!='     ;
00480           DEFB    DD
00481           DEFB    I2
00482           DEFB    DD
00483           DEFB    '@'
00484           DEFM    'DG='           ;60
00485           DEFB    DD
00486           DEFB    J
00487           DEFB    '@'
00488           DEFM    'D<E.>!='       ;
00489           DEFB    DD
00490           DEFB    '@'
00491           DEFM    'D<E:"#>!='     ;
00492           DEFB    T
00493           DEFB    '@'
00494           DEFM    'DE<!>"#='      ;
00495           DEFB    DD
00496           DEFB    Y
00497           DEFB    '@'
00498           DEFM    'DIS<!='        ;
00499           DEFB    DD
00500           DEFB    Il
00501           DEFB    S
00502           DEFB    '@'
00503           DEFM    'DO<!>!='       ;65
00504           DEFB    DD
00505           DEFB    IU
00506           DEFB    Ul
00507           DEFB    Ul
00508           DEFB    '@'
00509           DEFM    'DOES<!='       ;
00510           DEFB    DD
00511           DEFB    UH2
00512           DEFB    UH1
00513           DEFB    ZZ
00514           DEFB    '@'
00515           DEFM    'DOING<!='      ;
00516           DEFB    DD
00517           DEFB    IU
00518           DEFB    Ul
00519           DEFB    I2
00520           DEFB    NG
00521           DEFB    '@'
00522           DEFM    'DOW<!='        ;
00523           DEFB    DD
00524           DEFB    AH1
00525           DEFB    UH3
00526           DEFB    Ul
00527           DEFB    '@'
00528           DEFM    'DU>A='         ;
00529           DEFB    DD
00530           DEFB    J
00531           DEFB    IU
00532           DEFB    U
00533           DEFB    '@'
00534           DEFM    'D='            ;70
00535           DEFB    DD
00536           DEFB    '@'
00537           DEFB    '['
00538 ;
00539 ERULES     DEFM    'E<!>!='        ;
00540           DEFB    El
00541           DEFB    Y
00542           DEFB    '@'
00543           DEFM    'E<!>Q='        ;
00544           DEFB    EE
00545           DEFB    '@'
00546           DEFM    'E<!>#='        ;
00547           DEFB    EE
00548           DEFB    '@'
00549           DEFM    'E<:#>!='       ;
00550           DEFB    '@'
00551           DEFM    'E<:"!>!='      ;75
00552           DEFB    '@'
00553           DEFM    'E<:!>!='       ;
00554           DEFB    El
00555           DEFB    '@'
00556           DEFM    'ED<#>!='       ;
00557           DEFB    I2
00558           DEFB    DD
00559           DEFB    '@'
00560           DEFM    'E<:#>D!='      ;
00561           DEFB    '@'
00562           DEFM    'ED<:>!='       ;
00563           DEFB    I2
00564           DEFB    DD
00565           DEFB    '@'
00566           DEFM    'EV>ER='        ;80
00567           DEFB    EH
00568           DEFB    V
00569           DEFB    '@'
00570           DEFM    'EX='           ;
00571           DEFB    EH1
00572           DEFB    EH2
00573           DEFB    K
00574           DEFB    PA0
00575           DEFB    S
00576           DEFB    '@'
00577           DEFM    'E<V='          ;
00578           DEFB    '@'
00579           DEFM    'E>"%='         ;
00580           DEFB    EE
00581           DEFB    '@'
00582           DEFM    'ERI>#='        ;
00583           DEFB    I1
00584           DEFB    RR
00585           DEFB    Y
00586           DEFB    '@'
00587           DEFM    'ERI='          ;85
00588           DEFB    EH1
00589           DEFB    RR
00590           DEFB    I3
00591           DEFB    '@'
00592           DEFM    'ERY>!='        ;
00593           DEFB    EH1
00594           DEFB    RR
00595           DEFB    Y
00596           DEFB    '@'
00597           DEFM    'ER<:#>#='      ;
00598           DEFB    ER
00599           DEFB    '@'
00600           DEFM    'ER>#='         ;
00601           DEFB    EH1
00602           DEFB    RR
00603           DEFB    '@'
00604           DEFM    'ER='           ;
00605           DEFB    ER
00606           DEFB    '@'
00607           DEFM    'EVEN<!='       ;90
00608           DEFB    EE
00609           DEFB    V
00610           DEFB    EH
00611           DEFB    N
00612           DEFB    '@'
00613           DEFM    'E<:#>W='       ;
00614           DEFB    '@'
00615           DEFM    'E>!='          ;
00616           DEFB    '@'
00617           DEFM    'EW<$='         ;
00618           DEFB    IU
00619           DEFB    U
00620           DEFB    '@'
00621           DEFM    'EW=Y'          ;
00622           DEFB    Y
00623           DEFB    IU
00624           DEFB    U
00625           DEFB    '@'
00626           DEFM    'E>O='          ;95
00627           DEFB    EE
00628           DEFB    '@'
00629           DEFM    'ES<&:#>!='     ;
00630           DEFB    I2
00631           DEFB    ZZ
00632           DEFB    '@'
00633           DEFM    'E<:#>S!='      ;
00634           DEFB    '@'
00635           DEFM    'ELL='          ;
00636           DEFB    EH1
00637           DEFB    UH3
00638           DEFB    LL
00639           DEFB    UH3
00640           DEFB    '@'
00641           DEFM    'ELY<:#>!='     ;
00642           DEFB    LL
00643           DEFB    EE
00644           DEFB    '@'
00645           DEFM    'EMENT<:#='     ;100
00646           DEFB    MM
00647           DEFB    EH3
00648           DEFB    N
00649           DEFB    T
00650           DEFB    '@'
00651           DEFM    'EN<!='         ;
00652           DEFB    EH1
00653           DEFB    N
00654           DEFB    '@'
00655           DEFM    'EFUL='         ;
00656           DEFB    F
00657           DEFB    OO1
00658           DEFB    LL
00659           DEFB    '@'
00660           DEFM    'EE='           ;
00661           DEFB    EE
00662           DEFB    '@'
00663           DEFM    'EARN='         ;
00664           DEFB    ER
00665           DEFB    RR
00666           DEFB    N
00667           DEFB    '@'
00668           DEFM    'EAR<!>"='      ;105
00669           DEFB    ER
00670           DEFB    RR
00671           DEFB    '@'
00672           DEFM    'EAD='          ;
00673           DEFB    EH1
00674           DEFB    EH3
00675           DEFB    DD
00676           DEFB    '@'
00677           DEFM    'EA<:#>!='      ;
00678           DEFB    EE
00679           DEFB    UH2
00680           DEFB    '@'
```

Listing 3 continued

match occurs, this routine bumps the IX and HL pointers to test the next two ASCII values.

ASCII testing continues until the routine comes across the equals sign to signify another rule match. If the program encounters a sign other than the equals sign, control branches to the appropriate routine. Control passes again to the TXFR section where it sends the equated values of EH1, UH3, LL, and UH3 to the OUTBUF buffer. Control jumps back to CHTYPE with the IX pointer at the O in HELLO.

CHTYPE now scans the O rules and eventually finds the rule O>!=O1,U1 (line 1290 in Listing 3). It transfers these values to OUTBUF, which now contains the values PA0, HH, EH1, UH3, LL, UH3, O1, U1, and the stop code 3F. Control goes to the Output routine which sends the phonemes out to the Votrax chip. After completing the transfer, the routine restores all registers and jumps back to the calling program.

This short conversion should give you a general idea of what goes on in the TTSPRG. I didn't cover all conversion routines due to time and space limitations. I suggest you use a piece of paper to help keep track of all the pointers as you go through the various routines.

Applications Software

Now that you know how the Text-to-Speech program operates, you need to control it through a Basic program. Here I show you how to use Basic to send the Text-to-Speech program the strings you want to convert to speech. Listing 1 is the first Basic text conversion program.

You can apply Listing 1 for handicapped and educational uses. It gives sound feedback by pronouncing the letter of any key (except #) you press via the INKEY command. Single-key entries may be slow, but it takes time to pronounce each pressed key.

As it pronounces each key, the program builds a word string and speaks the word at the press of the space bar. Hitting the enter key makes the program vocalize all words created in a single string and tells you when to start typing with a spoken "Enter Please" message.

I installed the option of pronouncing or not pronouncing each key as you type it. The program verbally asks you this question on execution. It still vocalizes words individually when you hit the space bar.

The secret to this program is in its

Listing 3 continued

```
00681          DEFM    'EA>SU='          ;
00682          DEFB    EH
00683          DEFB    '@'
00684          DEFM    'EA='             ;
00685          DEFB    EE
00686          DEFB    '@'
00687          DEFM    'EIGH='           ;110
00688          DEFB    A2
00689          DEFB    A2
00690          DEFB    Y
00691          DEFB    '@'
00692          DEFM    'EI='             ;
00693          DEFB    EE
00694          DEFB    '@'
00695          DEFM    'EYE<1='          ;
00696          DEFB    AH1
00697          DEFB    EH3
00698          DEFB    I3
00699          DEFB    Y
00700          DEFB    '@'
00701          DEFM    'EY='             ;
00702          DEFB    EE
00703          DEFB    '@'
00704          DEFM    'EU='             ;
00705          DEFB    Y
00706          DEFB    IU
00707          DEFB    U1
00708          DEFB    '@'
00709          DEFM    'E='              ;115
00710          DEFB    EH1
00711          DEFB    '@'
00712          DEFB    '['
00713 ;
00714 FR       DEFM    'F<I>1='          ;
00715          DEFB    EH1
00716          DEFB    EH2
00717          DEFB    F
00718          DEFB    '@'
00719          DEFM    'FUL='            ;
00720          DEFB    F
00721          DEFB    UH3
00722          DEFB    LL
00723          DEFB    '@'
00724          DEFM    'FY<I>1='         ;
00725          DEFB    F
00726          DEFB    AH1
00727          DEFB    EH3
00728          DEFB    Y
00729          DEFB    '@'
00730          DEFM    'F='              ;
00731          DEFB    F
00732          DEFB    '@'
00733          DEFB    '['
00734 ;
00735 GR       DEFM    'G<I>1='          ;120
00736          DEFB    DD
00737          DEFB    J
00738          DEFB    E1
00739          DEFB    Y
00740          DEFB    '@'
00741          DEFM    'GIV='            ;
00742          DEFB    G
00743          DEFB    I1
00744          DEFB    I3
00745          DEFB    V
00746          DEFB    '@'
00747          DEFM    'G<I>I"='         ;
00748          DEFB    G
00749          DEFB    '@'
00750          DEFM    'GE>T='           ;
00751          DEFB    G
00752          DEFB    EH1
00753          DEFB    EH3
00754          DEFB    '@'
00755          DEFM    'GGES<US='        ;
00756          DEFB    G
00757          DEFB    DD
00758          DEFB    J
00759          DEFB    EH
00760          DEFB    S
00761          DEFB    '@'
00762          DEFM    'GG='             ;125
00763          DEFB    G
00764          DEFB    '@'
00765          DEFM    'G<#B='           ;
00766          DEFB    G
00767          DEFB    '@'
00768          DEFM    'G>+='            ;
00769          DEFB    DD
00770          DEFB    J
00771          DEFB    '@'
00772          DEFM    'GREAT='          ;
00773          DEFB    G
00774          DEFB    RR
00775          DEFB    A1
00776          DEFB    Y
00777          DEFB    T
00778          DEFB    '@'
00779          DEFM    'GH<#='           ;
00780          DEFB    '@'
00781          DEFM    'G='              ;130
00782          DEFB    G
00783          DEFB    '@'
00784          DEFB    '['
00785 ;
00786 HR       DEFM    'H<I>1='          ;
00787          DEFB    A1
00788          DEFB    AY
00789          DEFB    Y
00790          DEFB    T
00791          DEFB    CH
00792          DEFB    '@'
00793          DEFM    'HAV<1='          ;
00794          DEFB    HH
00795          DEFB    AE1
00796          DEFB    EH3
00797          DEFB    V
00798          DEFB    '@'
00799          DEFM    'HERE<1='         ;
00800          DEFB    HH
00801          DEFB    AY
00802          DEFB    I3
00803          DEFB    RR
00804          DEFB    '@'
00805          DEFM    'HOUR<1='         ;
00806          DEFB    AH1
00807          DEFB    UH3
00808          DEFB    W
00809          DEFB    ER
00810          DEFB    '@'
00811          DEFM    'HOW='            ;135
00812          DEFB    HH
00813          DEFB    AH1
00814          DEFB    O2
00815          DEFB    U1
00816          DEFB    '@'
00817          DEFM    'H>#='            ;
00818          DEFB    HH
00819          DEFB    '@'
00820          DEFM    'H='              ;
00821          DEFB    '@'
00822          DEFB    '['
00823 ;
00824 IR       DEFM    'I<I>1='          ;
00825          DEFB    AH1
00826          DEFB    EH3
00827          DEFB    I3
00828          DEFB    Y
00829          DEFB    '@'
00830          DEFM    'IN<1='           ;
00831          DEFB    I1
00832          DEFB    N
00833          DEFB    '@'
00834          DEFM    'I<I>1='          ;140
00835          DEFB    AH1
00836          DEFB    EH3
00837          DEFB    I3
00838          DEFB    Y
00839          DEFB    '@'
00840          DEFM    'IN>D='           ;
00841          DEFB    I1
00842          DEFB    I3
00843          DEFB    N
00844          DEFB    '@'
00845          DEFM    'IDE='            ;
00846          DEFB    AH1
00847          DEFB    EH3
00848          DEFB    Y
00849          DEFB    DD
00850          DEFB    '@'
00851          DEFM    'IER='            ;
```

Listing 3 continued

```
10 '************************************************
20 '                  L I S T I N G  4
30 '************************************************
40 '
50 '       STRING ENTRY PRONOUNCIATION PROGRAM
60 '                     BY
70 '               DAVID ENGELHARDT
80 '
100 '
110 ' THIS PROGRAM MAY BE USED TO EXPERIMENT IN THE CREATION
120 ' OF WORDS OR LARGE STRINGS.
130 ' THIS PROGRAM MAY ALSO BE USED AS A SUBROUTINE IN A LARGER
140 ' PROGRAM.
150 '
160 '************************************************
170 '
180 CLS:CLEAR 1000
190 DEF USR0=&HE950:CMD"L","TTSPRG/CMD":CMD"L","TABLE/CMD"
200 LINE INPUT A$
210 A=VARPTR(A$)
220 VV=USR0(A)
230 GOTO 200
240 END
```

Program Listing 4. Basic Text-to-Speech routine.

string manipulations. You enter each key via the INKEY command. The program combines entries into a single word string that the program puts into a single string of up to 255 bytes. TS$ is the complete string while SB$ is the word string with A$ as the single-key entries.

Since the maximum string length is 255 bytes, any entry greater than this makes the program branch to an error routine. This error routine speaks a message indicating it has reached maximum count, and proceeds to speak the string created prior to the error. There is also a small error routine that keeps the program from stopping execution for any kind of error. This error is also spoken to alert you to a problem. After any error, the program prompts you with another "Enter Please" message to start string entries again.

If you want to create or change any of the messages, you must either change an existing message or define a new one in your program. For example, to hear a string message spoken, use A=VARPTR(A$) to find out where the string is located in memory. The next instruction is VV=USR(A) for tape or VV=USR0(A) for disk. This command passes the string's location in memory to the TTSPRG program for conversion to speech.

You could also adapt it to speak a screen full of ASCII text by using single-dimension arrays.

In addition to handicapped applications, Listing 1 also helps children learn the alphabet and letter pronunciations. It teaches them how to create and pronounce words. The program also vocalizes all numbers.

The board also vocalizes math operators, facilitating teaching young children math problems with voice tutoring. I am working on a routine to break a number down and pronounce the individual units. Take the number 120 for example. To date, the Text-to-Speech program pronounces this number as one, two, zero instead of one hundred twenty. I decided not to include this routine and associated rules due to the size of the program.

The only character not pronounced is the # symbol as I use this to change the inflection codes. If you want to change the inflection codes within or at the beginning of a sentence, enter the # symbol followed by a 1, 3, 5, or 7. A combination of these codes gives interesting results in a spoken sentence.

Listing 4 is a short subroutine that you can use in a bigger program to

enter strings via the Input or Line Input command. If you don't have a disk system, use the Input command in place of the Line Input command. The only difference between the two is that the Input command doesn't allow insertion of commas in a string. This program also uses the inflection codes.

This program can prompt the user for an answer to a question or any other information and store it as a unique string. You can then send this unique string to the TTSPRG program when desired, such as in educational responses, games, and so on. I use this program to create words and sentences for experimental purposes.

System Adaptations

In regard to the USR command, if you run these programs on a 16K or cassette-based system, you need to change all of the VV=USR0(A) commands to VV=USR(A) in the Basic programs. The USR pointers have to be set so change line 220 in Listing 1 or line 190 in Listing 4 to read POKE 16526,240: POKE16527,105. This links USR calls to a machine-language program at the Text-to-Speech program in our case. Set memory size to 27115 and load in the speech programs with the System command.

Remember, to assemble the Text-to-Speech program and Rules Table for a 16K system, you must have someone with a bigger system do it for you unless you obtain the object code from another source. If someone does it for you, change the ORG statement in Listing 2 to 69F0 hex and the ORG statement in Listing 3 to 70A0 hex.

If you are going to assemble these programs on a 32K system, you may have to do some linking in regard to the Rules Table. The Text-to-Speech program is no problem as its source, including comments, is only 17,153 bytes long with room to spare. The Text-to-Speech program should be ORGed at A0F0 hex and the Rules Table at B0A0 hex for a 32K system.

If you have to split the Rules Table, be sure to incorporate all of the phoneme equates on the second source. This allows easy and straightforward entries of phoneme codes to the rules as opposed to the hex or decimal equivalent. To link the Rules Table sources together, patch into or equate at the beginning of the first source where the start of each rule is located in the second source. This should present no problems if you are careful.

System	Revisions
48K	TTSPRG=E950, TABLE=F000, CHTBLE=F000, ENDTBL=F0C0, MEM SIZE=59720
32K	TTSPRG=A9F0, TABLE=B0A0, CHTBLE=B0A0, ENDTBL=B160, MEM SIZE=43500
16K	TTSPRG=69F0, TABLE=70A0, CHTBLE=70A0, ENDTBL=7160, MEM SIZE=27115

Table 2. System changes.

Listings 2 and 3 as shown are for a 48K system. The ORGed values, memory size, and the two equated values at the beginning of the Text-to-Speech program are all you need to change to run it on a smaller system. Table 2 lists all value changes needed for each system.

Remember to change the DEF USR=&HXXXX statements in the Basic programs, if you use the 32K system instead of a 48K system. ■

Write to David Engelhardt at 10221 W. 101st Place, Broomfield, CO 80020.

Listing 3 continued

```
00852           DEFB    El
00853           DEFB    I3
00854           DEFB    ER
00855           DEFB    '@'
00856           DEFM    'IED<R:#>!='    ;
00857           DEFB    EE
00858           DEFB    DD
00859           DEFB    '@'
00860           DEFM    'IED>!='        ;145
00861           DEFB    AH
00862           DEFB    El
00863           DEFB    DD
00864           DEFB    '@'
00865           DEFM    'IEN='          ;
00866           DEFB    EE
00867           DEFB    EH
00868           DEFB    N
00869           DEFB    '@'
00870           DEFM    'IE>T='         ;
00871           DEFB    AH1
00872           DEFB    EH3
00873           DEFB    AY
00874           DEFB    I2
00875           DEFB    '@'
00876           DEFM    'IQUE='         ;
00877           DEFB    EE
00878           DEFB    K
00879           DEFB    '@'
00880           DEFM    'IR>#='         ;
00881           DEFB    AH1
00882           DEFB    EH3
00883           DEFB    AY
00884           DEFB    RR
00885           DEFB    '@'
00886           DEFM    'IR='           ;150
00887           DEFB    ER
00888           DEFB    '@'
00889           DEFM    'IGH='          ;
00890           DEFB    AH1
00891           DEFB    EH3
00892           DEFB    Y
00893           DEFB    '@'
00894           DEFM    'ILD='          ;
00895           DEFB    I2
00896           DEFB    I2
00897           DEFB    LL
00898           DEFB    DD
00899           DEFB    '@'
00900           DEFM    'IGN>!='        ;
00901           DEFB    AH1
00902           DEFB    EH3
00903           DEFB    Y
00904           DEFB    N
00905           DEFB    '@'
00906           DEFM    'IGN>"='        ;
00907           DEFB    AH1
00908           DEFB    EH3
00909           DEFB    Y
00910           DEFB    N
00911           DEFB    '@'
00912           DEFM    'IGN>%='        ;155
00913           DEFB    AH1
00914           DEFB    EH3
00915           DEFB    Y
00916           DEFB    N
00917           DEFB    '@'
00918           DEFM    'ING>!='        ;
00919           DEFB    I2
00920           DEFB    NG
00921           DEFB    '@'
00922           DEFM    'IZ>%='         ;
00923           DEFB    AH1
00924           DEFB    Y
00925           DEFB    ZZ
00926           DEFB    '@'
00927           DEFM    'IS>%='         ;
00928           DEFB    AH1
00929           DEFB    Y
00930           DEFB    ZZ
00931           DEFB    '@'
00932           DEFM    'IE>!='         ;
00933           DEFB    AH1
00934           DEFB    EH3
00935           DEFB    Y
00936           DEFB    '@'
00937           DEFM    'I<L>E='        ;160
00938           DEFB    EE
00939           DEFB    '@'
00940           DEFM    'IE<.>"='       ;
00941           DEFB    AH1
00942           DEFB    EH3
00943           DEFB    Y
00944           DEFB    '@'
00945           DEFM    'I<L>F='        ;
00946           DEFB    II
00947           DEFB    '@'
00948           DEFM    'IE='           ;
00949           DEFB    EE
00950           DEFB    '@'
00951           DEFM    'I>D%='         ;
00952           DEFB    Il
00953           DEFB    I3
00954           DEFB    '@'
00955           DEFM    'I>T%='         ;165
00956           DEFB    AH
00957           DEFB    El
00958           DEFB    '@'
00959           DEFM    'I<"+>"+='      ;
00960           DEFB    II
00961           DEFB    '@'
00962           DEFM    'I<:"#>"+='     ;
00963           DEFB    II
00964           DEFB    '@'
00965           DEFM    'I<:!>%='       ;
00966           DEFB    AH
00967           DEFB    El
00968           DEFB    '@'
00969           DEFM    'I>"+!#='       ;
00970           DEFB    AH1
00971           DEFB    EH3
00972           DEFB    '@'
00973           DEFM    'I>"+:#='       ;170
00974           DEFB    II
00975           DEFB    '@'
00976           DEFM    'I>"+='         ;
00977           DEFB    AH2
00978           DEFB    I2
00979           DEFB    '@'
00980           DEFM    'I>%='          ;
00981           DEFB    EE
00982           DEFB    '@'
00983           DEFM    'I='            ;
00984           DEFB    Il
00985           DEFB    '@'
00986           DEFB    '['
00987 ;
00988 JR        DEFM    'J<!>!='        ;
00989           DEFB    DD
00990           DEFB    J
00991           DEFB    EH3
00992           DEFB    Al
00993           DEFB    AY
00994           DEFB    Y
00995           DEFB    '@'
00996           DEFM    'J='            ;175
00997           DEFB    DD
00998           DEFB    J
00999           DEFB    '@'
01000           DEFB    '['
01001 ;
01002 KR        DEFM    'K<!>!='        ;
01003           DEFB    K
01004           DEFB    EH3
01005           DEFB    Al
01006           DEFB    AY
01007           DEFB    Y
01008           DEFB    '@'
01009           DEFM    'K<I>N='        ;
01010           DEFB    '@'
01011           DEFM    'K='            ;
01012           DEFB    K
01013           DEFB    '@'
01014           DEFB    '['
```

Listing 3 continued on p. 148

Listing 3 continued from p. 144

```
01015 ;
01016 LR        DEFM    'L<I>I='        ;
01017           DEFB    EH1
01018           DEFB    EH3
01019           DEFB    UH3
01020           DEFB    LL
01021           DEFB    '@'
01022           DEFM    'LO>C#='        ;180
01023           DEFB    LL
01024           DEFB    UH3
01025           DEFB    O1
01026           DEFB    U1
01027           DEFB    '@'
01028           DEFM    'LO>"#='        ;
01029           DEFB    LL
01030           DEFB    O
01031           DEFB    O
01032           DEFB    '@'
01033           DEFM    'LE>I='         ;
01034           DEFB    UH3
01035           DEFB    LL
01036           DEFB    '@'
01037           DEFM    'L<L='          ;
01038           DEFB    '@'
01039           DEFM    'L<:"#>%='      ;
01040           DEFB    UH2
01041           DEFB    LL
01042           DEFB    '@'
01043           DEFM    'LEAD='         ;185
01044           DEFB    LL
01045           DEFB    E1
01046           DEFB    Y
01047           DEFB    DD
01048           DEFB    '@'
01049           DEFM    'LESS>I='       ;
01050           DEFB    LL
01051           DEFB    EH2
01052           DEFB    S
01053           DEFB    '@'
01054           DEFM    'LY>I='         ;
01055           DEFB    LL
01056           DEFB    Y
01057           DEFB    '@'
01058           DEFM    'L='            ;
01059           DEFB    LL
01060           DEFB    '@'
01061           DEFM    '['
01062 ;
01063 MR        DEFM    'M<I>I='        ;
01064           DEFB    EH1
01065           DEFB    EH2
01066           DEFB    MM
01067           DEFB    '@'
01068           DEFM    'MOV='          ;190
01069           DEFB    MM
01070           DEFB    U1
01071           DEFB    U1
01072           DEFB    V
01073           DEFB    '@'
01074           DEFM    'MENT>I='       ;
01075           DEFB    MM
01076           DEFB    EH3
01077           DEFB    N
01078           DEFB    T
01079           DEFB    '@'
01080           DEFM    'M='            ;
01081           DEFB    MM
01082           DEFB    '@'
01083           DEFB    '['
01084 ;
01085 NR        DEFM    'N<I>I='        ;
01086           DEFB    EH1
01087           DEFB    EH2
01088           DEFB    N
01089           DEFB    '@'
01090           DEFM    'NG<E>+='       ;
01091           DEFB    N
01092           DEFB    DD
01093           DEFB    J
01094           DEFB    '@'
01095           DEFM    'NG>R='         ;195
01096           DEFB    NG
01097           DEFB    G
01098           DEFB    '@'
01099           DEFM    'NG>#='         ;
01100           DEFB    NG
01101           DEFB    G
01102           DEFB    '@'
01103           DEFM    'NGL>%='        ;
01104           DEFB    NG
01105           DEFB    G
01106           DEFB    UH2
01107           DEFB    LL
01108           DEFB    '@'
01109           DEFM    'NG='           ;
01110           DEFB    NG
01111           DEFB    '@'
01112           DEFM    'NK='           ;
01113           DEFB    NG
01114           DEFB    K
01115           DEFB    '@'
01116           DEFM    'NOW<I>I='      ;200
01117           DEFB    N
01118           DEFB    AH1
01119           DEFB    UH3
01120           DEFB    U1
01121           DEFB    '@'
01122           DEFM    'NESS>I='       ;
01123           DEFB    N
01124           DEFB    EH3
01125           DEFB    S
01126           DEFB    '@'
01127           DEFM    'NON<I='        ;
01128           DEFB    N
01129           DEFB    AH1
01130           DEFB    UH3
01131           DEFB    N
01132           DEFB    '@'
01133           DEFM    'N='            ;
01134           DEFB    N
01135           DEFB    '@'
01136           DEFB    '['
01137 ;
01138 OR        DEFM    'O<I>I='        ;
01139           DEFB    O2
01140           DEFB    O1
01141           DEFB    U1
01142           DEFB    '@'
01143           DEFM    'OF>!='         ;205
01144           DEFB    UH2
01145           DEFB    V
01146           DEFB    '@'
01147           DEFM    'OROUGH='       ;
01148           DEFB    ER
01149           DEFB    IU
01150           DEFB    U1
01151           DEFB    '@'
01152           DEFM    'OR<:#>!='      ;
01153           DEFB    ER
01154           DEFB    '@'
01155           DEFM    'ORS<:#>!='     ;
01156           DEFB    ER
01157           DEFB    ZZ
01158           DEFB    '@'
01159           DEFM    'OR='           ;
01160           DEFB    O2
01161           DEFB    O2
01162           DEFB    RR
01163           DEFB    '@'
01164           DEFM    'ONE<I='        ;210
01165           DEFB    W
01166           DEFB    UH1
01167           DEFB    UH2
01168           DEFB    N
01169           DEFB    '@'
01170           DEFM    'OW<LL='        ;
01171           DEFB    AH1
01172           DEFB    UH3
01173           DEFB    U1
01174           DEFB    '@'
01175           DEFM    'OW='           ;
01176           DEFB    O1
01177           DEFB    U1
01178           DEFB    '@'
01179           DEFM    'OVER<I='       ;
01180           DEFB    O1
01181           DEFB    O2
01182           DEFB    V
01183           DEFB    ER
01184           DEFB    '@'
01185           DEFM    'OV='           ;
01186           DEFB    UH1
01187           DEFB    V
01188           DEFB    '@'
01189           DEFM    'O>"%='         ;215
01190           DEFB    O1
01191           DEFB    U1
01192           DEFB    '@'
01193           DEFM    'O>"EN='        ;
01194           DEFB    O1
01195           DEFB    U1
01196           DEFB    '@'
01197           DEFM    'O>"I#='        ;
01198           DEFB    O1
01199           DEFB    U1
01200           DEFB    '@'
01201           DEFM    'OL>D='         ;
01202           DEFB    O2
01203           DEFB    O2
01204           DEFB    LL
01205           DEFB    LL
01206           DEFB    '@'
01207           DEFM    'ON<I='         ;
01208           DEFB    UH3
01209           DEFB    N
01210           DEFB    '@'
01211           DEFM    'OUGHT='        ;220
01212           DEFB    AH1
01213           DEFB    UH3
01214           DEFB    T
01215           DEFB    '@'
01216           DEFM    'OUGH='         ;
01217           DEFB    UH
01218           DEFB    F
01219           DEFB    '@'
01220           DEFM    'OU<I='         ;
01221           DEFB    AH
01222           DEFB    O1
01223           DEFB    '@'
01224           DEFM    'OU<H>S#='      ;
01225           DEFB    AH
01226           DEFB    O1
01227           DEFB    '@'
01228           DEFM    'OUS='          ;
01229           DEFB    UH2
01230           DEFB    S
01231           DEFB    '@'
01232           DEFM    'OUR='          ;225
01233           DEFB    O
01234           DEFB    RR
01235           DEFB    '@'
01236           DEFM    'OULD='         ;
01237           DEFB    IU
01238           DEFB    IU
01239           DEFB    OO1
01240           DEFB    DD
01241           DEFB    '@'
01242           DEFM    'OU<">"L='      ;
01243           DEFB    UH
01244           DEFB    '@'
01245           DEFM    'OUP='          ;
01246           DEFB    IU
01247           DEFB    U
01248           DEFB    PP
01249           DEFB    '@'
01250           DEFM    'OU='           ;
01251           DEFB    AH
01252           DEFB    O1
01253           DEFB    '@'
01254           DEFM    'OY='           ;230
01255           DEFB    O1
01256           DEFB    E1
01257           DEFB    '@'
01258           DEFM    'OING='         ;
01259           DEFB    O1
01260           DEFB    U1
01261           DEFB    II
01262           DEFB    NG
01263           DEFB    '@'
01264           DEFM    'OI='           ;
01265           DEFB    O1
01266           DEFB    UH3
01267           DEFB    I3
01268           DEFB    AY
01269           DEFB    '@'
01270           DEFM    'OOR='          ;
01271           DEFB    O
01272           DEFB    RR
01273           DEFB    '@'
01274           DEFM    'OOK='          ;
01275           DEFB    OO
01276           DEFB    K
01277           DEFB    '@'
01278           DEFM    'OOD='          ;235
01279           DEFB    OO
01280           DEFB    DD
01281           DEFB    '@'
01282           DEFM    'OO='           ;
01283           DEFB    IU
01284           DEFB    U
01285           DEFB    '@'
01286           DEFM    'O>E='          ;
01287           DEFB    O1
01288           DEFB    U1
01289           DEFB    '@'
01290           DEFM    'O>!='          ;
01291           DEFB    O1
01292           DEFB    U1
01293           DEFB    '@'
01294           DEFM    'OA='           ;
01295           DEFB    O1
01296           DEFB    U1
01297           DEFB    '@'
01298           DEFM    'ONLY<I='       ;240
01299           DEFB    O1
01300           DEFB    U1
01301           DEFB    N
01302           DEFB    LL
01303           DEFB    EE
01304           DEFB    '@'
01305           DEFM    'ONCE<I='       ;
01306           DEFB    W
01307           DEFB    UH
01308           DEFB    N
01309           DEFB    S
01310           DEFB    '@'
01311           DEFM    'ON'            ;
01312           DEFB    39              ;'
01313           DEFM    'T='
01314           DEFB    O1
01315           DEFB    U1
01316           DEFB    N
01317           DEFB    T
01318           DEFB    '@'
01319           DEFM    'O<C>N='        ;
01320           DEFB    AH
01321           DEFB    '@'
01322           DEFM    'O>NG='         ;
01323           DEFB    AW
01324           DEFB    '@'
01325           DEFM    'O<:"I>N='      ;245
01326           DEFB    UH
01327           DEFB    '@'
01328           DEFM    'ON<I='         ;
01329           DEFB    UH2
01330           DEFB    N
01331           DEFB    '@'
01332           DEFM    'ON<:#>I='      ;
01333           DEFB    UH2
01334           DEFB    N
01335           DEFB    '@'
01336           DEFM    'ON<"#='        ;
01337           DEFB    UH2
01338           DEFB    N
01339           DEFB    '@'
01340           DEFM    'O>STI='        ;
01341           DEFB    O1
01342           DEFB    U1
01343           DEFB    '@'
01344           DEFM    'OF>"='         ;250
01345           DEFB    AW
01346           DEFB    F
01347           DEFB    '@'
01348           DEFM    'OTHER='        ;
01349           DEFB    UH
01350           DEFB    THV
01351           DEFB    ER
01352           DEFB    '@'
01353           DEFM    'OSS>I='        ;
01354           DEFB    AW
01355           DEFB    S
01356           DEFB    '@'
01357           DEFM    'OM<:"#='       ;
01358           DEFB    UH
```

Listing 3 continued

Listing 3 continued

```
01359           DEFB    MM
01360           DEFB    '@'
01361           DEFM    'O='            ;
01362           DEFB    AH
01363           DEFB    '@'
01364           DEFB    '['
01365 ;
01366 PR        DEFM    'P<!>!='        ;255
01367           DEFB    PP
01368           DEFB    El
01369           DEFB    Y
01370           DEFB    '@'
01371           DEFM    'PH='           ;
01372           DEFB    F
01373           DEFB    '@'
01374           DEFM    'PEOP='         ;
01375           DEFB    PP
01376           DEFB    EE
01377           DEFB    PP
01378           DEFB    '@'
01379           DEFM    'POW='          ;
01380           DEFB    PP
01381           DEFB    AH
01382           DEFB    Ol
01383           DEFB    '@'
01384           DEFM    'PUT>!='        ;
01385           DEFB    PP
01386           DEFB    OO
01387           DEFB    T
01388           DEFB    '@'
01389           DEFM    'P='            ;260
01390           DEFB    PP
01391           DEFB    '@'
01392           DEFB    '['
01393 ;
01394 QR        DEFM    'Q<!>!='        ;
01395           DEFB    K
01396           DEFB    Yl
01397           DEFB    IU
01398           DEFB    Ul
01399           DEFB    Ul
01400           DEFB    '@'
01401           DEFM    'QUAR='         ;
01402           DEFB    K
01403           DEFB    W
01404           DEFB    O
01405           DEFB    RR
01406           DEFB    '@'
01407           DEFM    'QU='           ;
01408           DEFB    K
01409           DEFB    W
01410           DEFB    '@'
01411           DEFM    'Q='            ;
01412           DEFB    K
01413           DEFB    '@'
01414           DEFM    '['
01415 ;
01416 RRULES    DEFM    'R<!>!='        ;265
01417           DEFB    AHl
01418           DEFB    UH2
01419           DEFB    ER
01420           DEFB    '@'
01421           DEFM    'RE<!>"#='      ;
01422           DEFB    RR
01423           DEFB    EE
01424           DEFB    '@'
01425           DEFM    'RING='         ;
01426           DEFB    RR
01427           DEFB    I2
01428           DEFB    NG
01429           DEFB    '@'
01430           DEFM    'RI='           ;
01431           DEFB    RR
01432           DEFB    AHl
01433           DEFB    EH3
01434           DEFB    I3
01435           DEFB    Y
01436           DEFB    '@'
01437           DEFM    'R='            ;
01438           DEFB    RR
01439           DEFB    '@'
01440           DEFM    '['
01441 ;
01442 SR        DEFM    'S<!>!='        ;270
01443           DEFB    EHl
01444           DEFB    EH2
01445           DEFB    S
01446           DEFB    '@'
01447           DEFM    'SH='           ;
01448           DEFB    SH
01449           DEFB    '@'
01450           DEFM    'SION<#='       ;
01451           DEFB    ZH
01452           DEFB    UH2
01453           DEFB    N
01454           DEFB    '@'
01455           DEFM    'SOME='         ;
01456           DEFB    S
01457           DEFB    UH
01458           DEFB    MM
01459           DEFB    '@'
01460           DEFM    'SUR<#>#='      ;
01461           DEFB    ZH
01462           DEFB    ER
01463           DEFB    '@'
01464           DEFM    'SUR>#='        ;275
01465           DEFB    SH
01466           DEFB    ER
01467           DEFB    '@'
01468           DEFM    'SU<#>#='       ;
01469           DEFB    ZH
01470           DEFB    IU
01471           DEFB    U
01472           DEFB    '@'
01473           DEFM    'SSU<#>#='      ;
01474           DEFB    SH
01475           DEFB    IU
01476           DEFB    U
01477           DEFB    '@'
01478           DEFM    'SED<#>!='      ;
01479           DEFB    ZZ
01480           DEFB    DD
01481           DEFB    '@'
01482           DEFM    'S<#>#='        ;
01483           DEFB    ZZ
01484           DEFB    '@'
01485           DEFM    'SAID='         ;280
01486           DEFB    S
01487           DEFB    EH
01488           DEFB    DD
01489           DEFB    '@'
01490           DEFM    'SION<"='       ;
01491           DEFB    SH
01492           DEFB    UH2
01493           DEFB    N
01494           DEFB    '@'
01495           DEFM    'S>S='          ;
01496           DEFB    '@'
01497           DEFM    'S<.>!='        ;
01498           DEFB    ZZ
01499           DEFB    '@'
01500           DEFM    'S<E.:#>!='     ;
01501           DEFB    ZZ
01502           DEFB    '@'
01503           DEFM    'S<##:"#>!='    ;285
01504           DEFB    ZZ
01505           DEFB    '@'
01506           DEFM    'S<#:"#>!='     ;
01507           DEFB    S
01508           DEFB    '@'
01509           DEFM    'S<U>!='        ;
01510           DEFB    S
01511           DEFB    '@'
01512           DEFM    'S<#:I>!='      ;
01513           DEFB    ZZ
01514           DEFB    '@'
01515           DEFM    'SCH<!='        ;
01516           DEFB    S
01517           DEFB    K
01518           DEFB    '@'
01519           DEFM    'S>C+='         ;290
01520           DEFB    '@'
01521           DEFM    'SM<#='         ;
01522           DEFB    ZZ
01523           DEFB    MM
01524           DEFB    '@'
01525           DEFM    'SN<#>!='       ;
01526           DEFB    ZZ
01527           DEFB    UH2
01528           DEFB    N
01529           DEFB    '@'
01530           DEFM    'S='            ;
01531           DEFB    S
01532           DEFB    '@'
01533           DEFB    '['
01534 ;
01535 TR        DEFM    'T<!>!='        ;
01536           DEFB    T
01537           DEFB    El
01538           DEFB    AY
01539           DEFB    Y
01540           DEFB    '@'
01541           DEFM    'THE<!>!='      ;295
01542           DEFB    THV
01543           DEFB    UH2
01544           DEFB    '@'
01545           DEFM    'TO>!='         ;
01546           DEFB    T
01547           DEFB    IU
01548           DEFB    U
01549           DEFB    '@'
01550           DEFM    'THAT>!='       ;
01551           DEFB    THV
01552           DEFB    AE
01553           DEFB    T
01554           DEFB    '@'
01555           DEFM    'THIS<!>!='     ;
01556           DEFB    THV
01557           DEFB    II
01558           DEFB    S
01559           DEFB    '@'
01560           DEFM    'THEY<!='       ;
01561           DEFB    THV
01562           DEFB    AA
01563           DEFB    AY
01564           DEFB    '@'
01565           DEFM    'THERE<!='      ;300
01566           DEFB    THV
01567           DEFB    EH
01568           DEFB    RR
01569           DEFB    '@'
01570           DEFM    'THER='         ;
01571           DEFB    THV
01572           DEFB    ER
01573           DEFB    '@'
01574           DEFM    'THEIR='        ;
01575           DEFB    THV
01576           DEFB    EH
01577           DEFB    RR
01578           DEFB    '@'
01579           DEFM    'THAN<!>!='     ;
01580           DEFB    THV
01581           DEFB    AE
01582           DEFB    N
01583           DEFB    '@'
01584           DEFM    'THEM<!>!='     ;
01585           DEFB    THV
01586           DEFB    EH
01587           DEFB    MM
01588           DEFB    '@'
01589           DEFM    'THESE>!='      ;305
01590           DEFB    THV
01591           DEFB    EE
01592           DEFB    ZZ
01593           DEFB    '@'
01594           DEFM    'THEN<!='       ;
01595           DEFB    THV
01596           DEFB    EH
01597           DEFB    N
01598           DEFB    '@'
01599           DEFM    'THROUGH='      ;
01600           DEFB    TH
01601           DEFB    RR
01602           DEFB    IU
01603           DEFB    U
01604           DEFB    '@'
01605           DEFM    'THOSE='        ;
01606           DEFB    THV
01607           DEFB    Ol
01608           DEFB    Ul
01609           DEFB    ZZ
01610           DEFB    '@'
01611           DEFM    'THOUGH>!='     ;
01612           DEFB    THV
01613           DEFB    Ol
01614           DEFB    Ul
01615           DEFB    '@'
01616           DEFM    'THUS<!='       ;310
01617           DEFB    THV
01618           DEFB    UH
01619           DEFB    S
01620           DEFB    '@'
01621           DEFM    'TH='           ;
01622           DEFB    TH
01623           DEFB    '@'
01624           DEFM    'TED<:#>!='     ;
01625           DEFB    T
01626           DEFB    II
01627           DEFB    DD
01628           DEFB    '@'
01629           DEFM    'TI<S>#N='      ;
01630           DEFB    T
01631           DEFB    CH
01632           DEFB    '@'
01633           DEFM    'TI>O='         ;
01634           DEFB    SH
01635           DEFB    '@'
01636           DEFM    'TI>A='         ;315
01637           DEFB    SH
01638           DEFB    '@'
01639           DEFM    'TIEN='         ;
01640           DEFB    SH
01641           DEFB    UH2
01642           DEFB    N
01643           DEFB    '@'
01644           DEFM    'TUR>#='        ;
01645           DEFB    T
01646           DEFB    CH
01647           DEFB    ER
01648           DEFB    '@'
01649           DEFM    'TU>A='         ;
01650           DEFB    T
01651           DEFB    CH
01652           DEFB    IU
01653           DEFB    U
01654           DEFB    '@'
01655           DEFM    'TWO<!='        ;
01656           DEFB    T
01657           DEFB    IU
01658           DEFB    U
01659           DEFB    '@'
01660           DEFM    'T='            ;320
01661           DEFB    T
01662           DEFB    '@'
01663           DEFB    '['
01664 ;
01665 UR        DEFM    'R<!>!='        ;
01666           DEFB    Yl
01667           DEFB    IU
01668           DEFB    Ul
01669           DEFB    Ul
01670           DEFB    '@'
01671           DEFM    'UN<!>I='       ;
01672           DEFB    Y
01673           DEFB    IU
01674           DEFB    U
01675           DEFB    N
01676           DEFB    '@'
01677           DEFM    'UN<!='         ;
01678           DEFB    UH
01679           DEFB    N
01680           DEFB    '@'
01681           DEFM    'UPON<!='       ;
01682           DEFB    UH2
01683           DEFB    PP
01684           DEFB    AW
01685           DEFB    N
01686           DEFB    '@'
01687           DEFM    'UR<$>#='       ;325
01688           DEFB    OO
01689           DEFB    RR
01690           DEFB    '@'
01691           DEFM    'UR>#='         ;
01692           DEFB    Y
01693           DEFB    OO
01694           DEFB    RR
01695           DEFB    '@'
01696           DEFM    'UR='           ;
01697           DEFB    ER
01698           DEFB    '@'
01699           DEFM    'US>!='         ;
01700           DEFB    UHl
```

Listing 3 continued

Listing 3 continued

```
01701           DEFB    UH2
01702           DEFB    S
01703           DEFB    '@'
01704           DEFM    'UE<">!='       ;
01705           DEFB    IU
01706           DEFB    Ul
01707           DEFB    Ul
01708           DEFB    '@'
01709           DEFM    'U>"!='         ;330
01710           DEFB    UH
01711           DEFB    '@'
01712           DEFM    'U>""='         ;
01713           DEFB    UH
01714           DEFB    '@'
01715           DEFM    'UY='           ;
01716           DEFB    AH
01717           DEFB    El
01718           DEFB    '@'
01719           DEFM    'U<G!>#='       ;
01720           DEFB    '@'
01721           DEFM    'U<G>%='        ;
01722           DEFB    '@'
01723           DEFM    'U<G>#='        ;335
01724           DEFB    W
01725           DEFB    '@'
01726           DEFM    'U<N#='         ;
01727           DEFB    Y
01728           DEFB    IU
01729           DEFB    U
01730           DEFB    '@'
01731           DEFM    'U<$='          ;
01732           DEFB    IU
01733           DEFB    U
01734           DEFB    '@'
01735           DEFM    'U='            ;
01736           DEFB    Y
01737           DEFB    IU
01738           DEFB    U
01739           DEFB    '@'
01740           DEFM    '['
01741 ;
01742 VR        DEFM    'V<!>!='        ;
01743           DEFB    V
01744           DEFB    El
01745           DEFB    AY
01746           DEFB    Y
01747           DEFB    '@'
01748           DEFM    'VIEW='         ;340
01749           DEFB    V
01750           DEFB    Y
01751           DEFB    IU
01752           DEFB    U
01753           DEFB    '@'
01754           DEFM    'VERI<!='       ;
01755           DEFB    V
01756           DEFB    EH1
01757           DEFB    RR
01758           DEFB    I3
01759           DEFB    '@'
01760           DEFM    'VE='           ;
01761           DEFB    V
01762           DEFB    '@'
01763           DEFM    'VAL<!='        ;
01764           DEFB    V
01765           DEFB    AEl
01766           DEFB    EH3
01767           DEFB    LL
01768           DEFB    '@'
01769           DEFM    'V='            ;
01770           DEFB    V
01771           DEFB    '@'
01772           DEFM    '['
01773 ;
01774 WR        DEFM    'W<!>!='        ;345
01775           DEFB    DD
01776           DEFB    UH1
01777           DEFB    BB
01778           DEFB    UH3
01779           DEFB    LL
01780           DEFB    Yl
01781           DEFB    IU
01782           DEFB    Ul
01783           DEFB    '@'
01784           DEFM    'WERE<!='       ;
01785           DEFB    W
01786           DEFB    ER
01787           DEFB    '@'
01788           DEFM    'WA>S='         ;
01789           DEFB    W
01790           DEFB    AH
01791           DEFB    '@'
01792           DEFM    'WA>T='         ;
01793           DEFB    W
01794           DEFB    AH
01795           DEFB    '@'
01796           DEFM    'WHERE='        ;
01797           DEFB    HH
01798           DEFB    W
01799           DEFB    EH
01800           DEFB    RR
01801           DEFB    '@'
01802           DEFM    'WHAT='         ;350
01803           DEFB    W
01804           DEFB    UH3
01805           DEFB    UH1
01806           DEFB    T
01807           DEFB    '@'
01808           DEFM    'WHOL='         ;
01809           DEFB    HH
01810           DEFB    Ol
01811           DEFB    Ul
01812           DEFB    LL
01813           DEFB    '@'
01814           DEFM    'WHO='          ;
01815           DEFB    HH
01816           DEFB    IU
01817           DEFB    U
01818           DEFB    '@'
01819           DEFM    'WH='           ;
01820           DEFB    HH
01821           DEFB    W
01822           DEFB    '@'
01823           DEFM    'WAR='          ;
01824           DEFB    W
01825           DEFB    O
01826           DEFB    RR
01827           DEFB    '@'
01828           DEFM    'WOR>"='        ;355
01829           DEFB    W
01830           DEFB    ER
01831           DEFB    '@'
01832           DEFM    'WR='           ;
01833           DEFB    RR
01834           DEFB    '@'
01835           DEFM    'W='            ;
01836           DEFB    W
01837           DEFB    '@'
01838           DEFB    '['
01839 ;
01840 XR        DEFM    'X<!>!='        ;
01841           DEFB    EH1
01842           DEFB    EH2
01843           DEFB    K
01844           DEFB    PA0
01845           DEFB    S
01846           DEFB    '@'
01847           DEFM    'X<!='          ;
01848           DEFB    EH1
01849           DEFB    EH2
01850           DEFB    K
01851           DEFB    PA0
01852           DEFB    S
01853           DEFB    '@'
01854           DEFM    'X='            ;360
01855           DEFB    K
01856           DEFB    S
01857           DEFB    '@'
01858           DEFM    '['
01859 ;
01860 YR        DEFM    'Y<!>!='        ;
01861           DEFB    W
01862           DEFB    AH1
01863           DEFB    EH3
01864           DEFB    I3
01865           DEFB    Y
01866           DEFB    '@'
01867           DEFM    'YOUNG='        ;
01868           DEFB    Y
01869           DEFB    UH
01870           DEFB    NG
01871           DEFB    '@'
01872           DEFM    'YOU<!='        ;
01873           DEFB    Y
01874           DEFB    IU
01875           DEFB    U
01876           DEFB    '@'
01877           DEFM    'YES<!='        ;
01878           DEFB    Y
01879           DEFB    EH
01880           DEFB    S
01881           DEFB    '@'
01882           DEFM    'Y<!='          ;365
01883           DEFB    Y
01884           DEFB    '@'
01885           DEFM    'Y<:"#>!='      ;
01886           DEFB    EE
01887           DEFB    '@'
01888           DEFM    'Y<:"#>I='      ;
01889           DEFB    EE
01890           DEFB    '@'
01891           DEFM    'Y<:I>!='       ;
01892           DEFB    AH
01893           DEFB    El
01894           DEFB    '@'
01895           DEFM    'Y<:I>#='       ;
01896           DEFB    AH
01897           DEFB    El
01898           DEFB    '@'
01899           DEFM    'Y<:I>"+:#='    ;370
01900           DEFB    II
01901           DEFB    '@'
01902           DEFM    'Y<:I>"#='      ;
01903           DEFB    AH
01904           DEFB    El
01905           DEFB    '@'
01906           DEFM    'Y='            ;
01907           DEFB    II
01908           DEFB    '@'
01909           DEFB    '['
01910 ;
01911 ZR        DEFM    'Z<!>!='        ;
01912           DEFB    ZZ
01913           DEFB    El
01914           DEFB    Y
01915           DEFB    '@'
01916           DEFM    'Z='            ;
01917           DEFB    ZZ
01918           DEFB    '@'
01919           DEFB    '['
01920 ;
01921 BLANK     DEFM    ' ='            ;375
01922           DEFB    PA0
01923           DEFB    '@'
01924           DEFB    '['
01925 ;
01926 PERIOD    DEFM    '.='            ;
01927           DEFB    PA1
01928           DEFB    '@'
01929           DEFB    '['
01930 ;
01931 APOST     DEFB    39              ;
01932           DEFM    '<!>!='
01933           DEFB    PA0
01934           DEFB    '@'
01935           DEFB    39              ;
01936           DEFM    'S>!='
01937           DEFB    II
01938           DEFB    ZZ
01939           DEFB    '@'
01940           DEFB    39              ;
01941           DEFM    'S<!.='
01942           DEFB    ZZ
01943           DEFB    '@'
01944           DEFB    39              ;380
01945           DEFM    'S<E.:#='
01946           DEFB    ZZ
01947           DEFB    '@'
01948           DEFB    39              ;
01949           DEFM    'S<!#='
01950           DEFB    ZZ
01951           DEFB    '@'
01952           DEFB    39              ;
01953           DEFM    ' ='
01954           DEFB    PA0
01955           DEFB    '@'
01956           DEFB    '['
01957 ;
01958 COMMA     DEFM    ',='            ;
01959           DEFB    PA1
01960           DEFB    '@'
01961           DEFB    '['
01962 ;
01963 QUEST     DEFM    '?='            ;
01964           DEFB    PA1
01965           DEFB    '@'
01966           DEFB    '['
01967 ;
01968 ZERO      DEFM    '0='            ;385
01969           DEFB    ZZ
01970           DEFB    AY
01971           DEFB    Il
01972           DEFB    RR
01973           DEFB    Ol
01974           DEFB    Ul
01975           DEFB    '@'
01976           DEFB    '['
01977 ;
01978 R1        DEFM    '1='            ;
01979           DEFB    W
01980           DEFB    UH1
01981           DEFB    UH2
01982           DEFB    N
01983           DEFB    '@'
01984           DEFB    '['
01985 ;
01986 R2        DEFM    '2='            ;
01987           DEFB    T
01988           DEFB    IU
01989           DEFB    Ul
01990           DEFB    Ul
01991           DEFB    '@'
01992           DEFB    '['
01993 ;
01994 R3        DEFM    '3='            ;
01995           DEFB    TH
01996           DEFB    RR
01997           DEFB    El
01998           DEFB    Y
01999           DEFB    '@'
02000           DEFB    '['
02001 ;
02002 R4        DEFM    '4='            ;
02003           DEFB    F
02004           DEFB    Ol
02005           DEFB    O2
02006           DEFB    RR
02007           DEFB    '@'
02008           DEFB    '['
02009 ;
02010 R5        DEFM    '5='            ;390
02011           DEFB    F
02012           DEFB    AH1
02013           DEFB    EH3
02014           DEFB    Y
02015           DEFB    V
02016           DEFB    '@'
02017           DEFB    '['
02018 ;
02019 R6        DEFM    '6='            ;
02020           DEFB    S
02021           DEFB    Il
02022           DEFB    I3
02023           DEFB    K
02024           DEFB    PA0
02025           DEFB    S
02026           DEFB    '@'
02027           DEFB    '['
02028 ;
02029 R7        DEFM    '7='            ;
02030           DEFB    S
02031           DEFB    EH1
02032           DEFB    EH3
02033           DEFB    V
02034           DEFB    I2
02035           DEFB    N
02036           DEFB    '@'
02037           DEFB    '['
02038 ;
02039 R8        DEFM    '8='            ;
02040           DEFB    A2
02041           DEFB    A2
02042           DEFB    Y
02043           DEFB    T
```

Listing 3 continued

Listing 3 continued

```
02044           DEFB    '@'
02045           DEFB    '['
02046 ;
02047 R9        DEFM    '9='            ;
02048           DEFB    N
02049           DEFB    AH1
02050           DEFB    EH3
02051           DEFB    Y
02052           DEFB    N
02053           DEFB    '@'
02054           DEFB    '['
02055 ;
02056 MINUS     DEFM    '-='            ;395
02057           DEFB    MM
02058           DEFB    AH1
02059           DEFB    Y
02060           DEFB    N
02061           DEFB    EH3
02062           DEFB    S
02063           DEFB    '@'
02064           DEFB    '['
02065 ;
02066 PLUS      DEFM    '+='            ;
02067           DEFB    PP
02068           DEFB    LL
02069           DEFB    UH1
02070           DEFB    UH2
02071           DEFB    S
02072           DEFB    '@'
02073           DEFB    '['
02074 ;
02075 EQUALS    DEFB    '='             ;
02076           DEFB    '='
02077           DEFB    Y
02078           DEFB    K
02079           DEFB    W
02080           DEFB    UH3
02081           DEFB    LL
02082           DEFB    ZZ
02083           DEFB    '@'
02084           DEFB    '['
02085 ;
02086 MULT      DEFM    '*='            ;
02087           DEFB    T
02088           DEFB    AH1
02089           DEFB    EH3
02090           DEFB    Y
02091           DEFB    MM
02092           DEFB    ZZ
02093           DEFB    '@'
02094           DEFB    '['
02095 ;
02096 DIVIDE    DEFM    '/='            ;
02097           DEFB    DD
02098           DEFB    I1
02099           DEFB    V
02100           DEFB    AH1
02101           DEFB    EH3
02102           DEFB    Y
02103           DEFB    DD
02104           DEFB    I2
02105           DEFB    DD
02106           DEFB    BB
02107           DEFB    AH1
02108           DEFB    EH3
02109           DEFB    I3
02110           DEFB    Y
02111           DEFB    '@'
02112           DEFB    '['
02113 ;
02114 LARROW    DEFB    8               ;400
02115           DEFB    '='
02116           DEFB    BB
02117           DEFB    AE1
02118           DEFB    AE1
02119           DEFB    K
02120           DEFB    S
02121           DEFB    PP
02122           DEFB    A1
02123           DEFB    AY
02124           DEFB    Y
02125           DEFB    S
02126           DEFB    '@'
02127           DEFB    '['
02128 ;
02129 RARROW    DEFB    9               ;
02130           DEFB    '='
02131           DEFB    RR
02132           DEFB    UH3
02133           DEFB    AH2
02134           DEFB    Y
02135           DEFB    T
02136           DEFB    PA1
02137           DEFB    EH1
02138           DEFB    EH3
02139           DEFB    RR
02140           DEFB    O1
02141           DEFB    U1
02142           DEFB    '@'
02143           DEFB    '['
02144 ;
02145 UARROW    DEFB    91              ;
02146           DEFB    '='
02147           DEFB    UH1
02148           DEFB    UH2
02149           DEFB    PP
02150           DEFB    EH1
02151           DEFB    EH3
02152           DEFB    RR
02153           DEFB    O1
02154           DEFB    U1
02155           DEFB    '@'
02156           DEFB    '['
02157 ;
02158 DARROW    DEFB    10              ;
02159           DEFB    '='
02160           DEFB    DD
02161           DEFB    AH1
02162           DEFB    UH3
02163           DEFB    U1
02164           DEFB    N
02165           DEFB    EH1
02166           DEFB    EH3
02167           DEFB    RR
02168           DEFB    O1
02169           DEFB    U1
02170           DEFB    '@'
02171           DEFB    '['
02172 ;
02173 CLEAR     DEFB    31              ;
02174           DEFB    '='
02175           DEFB    K
02176           DEFB    LL
02177           DEFB    AY
02178           DEFB    I3
02179           DEFB    RR
02180           DEFB    '@'
02181           DEFB    '['
02182 ;
02183 LTHAN     DEFM    '<='            ;405
02184           DEFB    LL
02185           DEFB    EH1
02186           DEFB    EH3
02187           DEFB    S
02188           DEFB    THV
02189           DEFB    EH1
02190           DEFB    EH3
02191           DEFB    N
02192           DEFB    '@'
02193           DEFB    '['
02194 ;
02195 GTHAN     DEFM    '>='            ;
02196           DEFB    G
02197           DEFB    RR
02198           DEFB    A1
02199           DEFB    Y
02200           DEFB    T
02201           DEFB    ER
02202           DEFB    THV
02203           DEFB    EH1
02204           DEFB    EH3
02205           DEFB    N
02206           DEFB    '@'
02207           DEFB    '['
02208 ;
02209 COLON     DEFM    ':='            ;
02210           DEFB    PA0
02211           DEFB    '@'
02212           DEFB    '['
02213 ;
02214 SEMICO    DEFM    ';='            ;
02215           DEFB    PA0
02216           DEFB    '@'
02217           DEFB    '['
02218 ;
02219 EXCLA     DEFM    '!='            ;
02220           DEFB    PA0
02221           DEFB    '@'
02222           DEFB    '['
02223 ;
02224 QUOTE     DEFM    '"='            ;410
02225           DEFB    PA0
02226           DEFB    '@'
02227           DEFB    '['
02228 ;
02229 DOLLAR    DEFM    '$='            ;
02230           DEFB    DD
02231           DEFB    AH1
02232           DEFB    UH3
02233           DEFB    LL
02234           DEFB    ER
02235           DEFB    '@'
02236           DEFB    '['
02237 ;
02238 PERCNT    DEFM    '%='            ;
02239           DEFB    PP
02240           DEFB    ER
02241           DEFB    S
02242           DEFB    EH1
02243           DEFB    EH3
02244           DEFB    N
02245           DEFB    T
02246           DEFB    '@'
02247           DEFB    '['
02248 ;
02249 ANDD      DEFM    '&='            ;
02250           DEFB    AE1
02251           DEFB    EH3
02252           DEFB    N
02253           DEFB    DD
02254           DEFB    '@'
02255           DEFB    '['
02256 ;
02257 LBRACK    DEFM    '(='            ;
02258           DEFB    PA0
02259           DEFB    '@'
02260           DEFB    '['
02261 ;
02262 RBRACK    DEFM    ')='            ;415
02263           DEFB    PA0
02264           DEFB    '@'
02265           DEFB    '['
02266 ;
02267 ATSYM     DEFM    '@='            ;416
02268           DEFB    PA0
02269           DEFB    '@'
02270           DEFB    '['
02271 ;
02272           END     CHTBLE
```