Conversion to other Basics

Radio Shack BASIC: Approximately 25 of the games which do not use strings will run under Level 1 BASIC; another 20 or so may be converted relatively easily by converting to all numeric input (1 instead of YES, 0 for NO, etc.). In Level 2, change RND (1) to RND (0). Also function definitions, when used, must be expanded.

Apple II BASIC: All programs will run directly in Applesoft Basic with little or no modification. In integer Basic, strings are similar to HF Basic. Also, multiple statements on one line work differently than with Microsoft Basic.

DEC BASIC PLUS: Programs written in Altair BASIC should be completely compatible with DEC BASIC PLUS.

HP BASIC: Some conversion will be required to get certain programs to run in HP BASIC. First, HP BASIC does not have a zero subscript for arrays. So if, in a program, the zero subscript is used, HP BASIC will report a subscript out of bounds error. The solution is to add one to all the array subscripts if it is determined that the program does use the zero subscript.

Second, HP BASIC character strings differ from Altair BASIC. There are no string arrays. So if string arrays are used in some program, a conversion must be made. Possibly the string array could be converted into a single string, or a numeric array, or a series of data statements, or possibly a data file. Also, HP BASIC does not have RIGHT$, LEFT$ or MID$ functions. Instead substitute the normal HP string subscripting conventions [for instance, RIGHTS$(X$,2) becomes X$(LEN(X$)-1, LEN(X$)].

Third, HP BASIC does not have multiple statements on one line. Convert multiple statement lines to several separate lines.

SWTPC 6800 BASIC: Versions 2.0 and above of this BASIC should be nearly compatible. Like HP BASIC, SWTPC BASIC does not allow the zero subscript. Also, character strings may not contain more than 32 characters. Note that in Altair BASIC, RND(1) is used to get random numbers, rather than RND(0) as in SWTPC BASIC. These comments also apply to **MSI Disk BASIC.**

TDL ZAPPLE 8K and 12K BASIC: All the programs are executable in both TDL BASICs. However TDL ZAPPLE BASIC has a RANDOMIZE verb which should be inserted at the beginning of a program to get true pseudorandom numbers.

Sol BASIC: The programs will work directly in Sol Extended Basic. Many programs will run in 5K Basic but the majority will require considerable conversion.

PET BASIC: Commodore PET BASIC is identical to the BASIC used in this book. Thus, all the games will run without any conversion.

Cromemco 16K Extended BASIC: Cromemco BASIC allows multiple statements, but certain statements must be the last statement on a line, so some rearrangement may be necessary. Two-dimensional arrays must be dimensioned explicitly (there is no default to a 10 by 10 dimension). Also, strings in Cromemco BASIC are implemented like HP BASIC, not Altair BASIC.

Ohio Scientific BASIC: Most programs will run directly on Challenger 2P and larger systems. The 32-characters per line display of the Challenger 1P and Superboard will require conversion of the program output routines.

IMSAI 8K BASIC: Programs are executable with little or no modification.

North Star Disk BASIC: North Star BASIC has character strings like HP BASIC, so some conversion will be required. Also note that the function SQRT is used in place of SQR.

PolyMorphic 11K BASIC: The multiple statement separator is a backslash (/), not the colon. Like North Star BASIC, SQRT is used instead of SQR. In Poly BASIC, you're not allowed to exit a FOR loop prematurely, except by using the EXIT verb. So whenever you see an IF...THEN out of a FOR loop, use the EXIT verb. Also note that in order to get random numbers, RND must be used with a "seed" value between 0 and 1, which determines at what point in the sequence of random numbers the RND function starts. This seed value is usually based on the time of day. After initializing the RND function, use RND(0) to get random numbers.

BASIC-E (runs under CP/M Disk Operating System): All arrays must be dimensioned (there is no default). The RANDOMIZE verb should be used, at the beginning of the program, to get random numbers. Also, multiple statements are not allowed, so conversion to several single statement lines will be necessary.