

Another form of the EXEC command allows execution of chains of XYBASIC programs without user intervention. For example, in the command

```
EXEC #A000, G
```

the address #A000 specifies the location of a XYBASIC program, as usual. The suffix ,G indicates that XYBASIC should execute the specified program immediately (by doing a GOTO to its first line), rather than returning to direct mode after performing the EXEC command. The EXEC <location>, G command does not clear variables. However, it does destroy any GOSUB/RETURN and FOR/NEXT context, making it illegal to have a FOR statement in one program and the matching NEXT in another. Of course, a CLEAR command can be included as the first line of the specified program; in this case the command acts like an EXEC immediately followed by a RUN.

The command

```
EXEC ,G
```

with the address omitted but the suffix ,G remaining, tells XYBASIC to execute the program currently in its RAM workspace.

You can use EXEC <location>, G to build complex systems containing several independent programs. A master control program can type a list of available programs (called a menu), and allow the user to choose which program he wishes by typing a character. Then by executing an appropriate EXEC command depending on the character, the master can transfer control directly to the chosen program.

One additional type of R0 error may occur in conjunction with EXEC. If the location specified in the EXEC command is ROM (EXEC is especially useful for executing programs stored on PROMs), line breakpoints may not be used. When a line BREAK or UNBREAK command is executed in this situation, an R0 error occurs and the command is ignored.

FIRST and LAST

You may sometimes want to know the location of a XYBASIC program, for example to burn it into PROM. Therefore the functions FIRST and LAST return the locations of the first and last bytes of the current program. The command

```
PRINT FIRST, LAST, LAST-FIRST+1
```

prints the location and length of the current program, whether in working space or not.

Suppose that you have a ROM based XYBASIC system, including a PROM burning routine at location 0A000H. Assume also that the PROM burning routine operates by burning the bytes starting at RAM location 0C000H, and that registers BC tell the routine how many bytes are to be burned. If the routine is successful, it returns 0 in BC; if unsuccessful, BC returns the first location at which the burned PROM disagrees with the desired value. Then you can burn a XYBASIC program into PROM by executing the following subroutine, which uses the SCALL command explained in Section 11 above.