which version was selected, each version was written as a separate program. Each of these programs turned out to be no more than a list of subroutines that control that mode of play. The subs are all the same for each version, but which subs where determine the version to be played. Actually, when a program is subroutined to this extent, each subroutine call may be thought of as one instruction in a specialized language designed to play the game you are programming. You only need to order the subroutines as if you were programming in a new language of your own design to arrive at the final controller. As the Version Controllers are well documented, you should be able to write your own versions with beepers where you want them, and styles of play that suit you, without having to dig into the subroutines themselves or understand how they work. (If you do want to go deeper, those subroutines are equally well documented, too.) Please note also the simplicity of each controller. The computer/computer version is only 20 instructions long occupying 40 bytes of memory! The human/human game controller is even shorter -- 15 instructions for 30 bytes of space run the entire game.

As a suggestion for a new controller, you may consider writing a version controller that keeps a