

AUTOSTARTING

All Max-FORTH versions have autostarting capabilities. The autostart capability is the means by which the language or the designated user program is started at power-up or reset. It makes up the better part of the Max-FORTH's operating system. In Version 2.x and greater releases, there are three distinct actions in the autostart sequence.

In brief summary, the first autostart attempt allows a machine code routine to alter the INIT, OPTION and TMSK2 registers immediately after reset. The second autostart attempt checks the beginning of EEPROM and then all 1K memory map boundaries for a temporary autostart pattern. The third autostart attempt checks the beginning of EEPROM and all memory map 1K boundaries for a primary autostart pattern. Now each will be discussed in detail.

The first autostart occurs within a few cycles after reset. It looks at the EEPROM at \$B7FE for an primary autostart pattern, \$A55A. If that pattern is found, a JSR to location \$B7FB is executed. At that location three empty bytes allow the insertion of a jump instruction. Control can be there by passed to a designated machine code routine. The routine can alter the INIT (REG 3-0 & RAM 3-0 bits), OPTIONS (IRQE, DLY, CR1 & CR0 bits) and TMSK2 (PR1 & PR0 bits) registers which must be establish in the first 64 clock cycles following reset. Since they can only be written once and only in the 64 cycle period, any settings made in that routine will not be affected by later system initialization action. Normally, the machine code routine will return to the system initialization and autostart functions by performing an RTS. Of course, when the rest of the system initialization is not required, control could be maintained by user machine code programs thereafter.

The second autostart attempt checks the beginning of EEPROM, \$B600, and then searches all 1K memory map boundaries, starting at location \$0400 and continuing through \$FC00, for a temporary autostart pattern, \$A44A. If the temporary autostart pattern is found at a given location \$XX00, the next word location following the autostart pattern, \$XX02, is taken to be the CFA of the word to be run. This word could be anywhere in the memory map, although this program will typically be in the memory device positioned at the boundary. The autostarted routine can be a either a high level or code definition. The unique value of this autostart attempt and the temporary autostart pattern is that the autostarted definition need not be an endless loop. It can terminate normally with a ; for high level words or a JMP NEXT for code definitions. When the system turns control over to a temporary autostart word, the IP is set to return control to the third autostart attempt. In this manner, normal termination of a temporary autostart word will cause initiation of the primary autostart search.

The third autostart attempt checks the beginning of EEPROM, \$B600, and then searches all 1K memory map boundaries, starting at location \$0400 and continuing through \$FC00, for a primary autostart pattern, \$A55A. If the primary autostart pattern is found at a given location \$XX00, the next word location following the autostart pattern, \$XX02, is taken to be the CFA of the word to be run. This word could be anywhere in the memory map, although this program will typically be in the memory device positioned at the boundary. The autostarted routine can be a either a high level or code definition. The autostarted definition will normally be an endless loop. If it terminates, with a ; for high level words or a JMP NEXT for code definitions, the system again returns control to the beginning of the third autostart attempt. In this case, the same word will be found and autostarted again and again. If no user autostarts are found, however, a primary autostart pattern should be found at \$E000, the beginning of the Max-FORTH ROM. If by some circumstance it is not there, the search will begin again at the third step. It will so continue until an autostart pattern is found.

This three level autostart system gives a great deal of flexibility to the system user. With no interference, the Max-FORTH operating system will find and run itself. A user program can intercept the primary autostart search by having an autostart pattern at a lower memory location than the Max-FORTH ROM. A specific application to be started and the system thereby dedicated to its operation, instead of the running the Max-FORTH outer interpreter. On the other hand, a user may simply want to modify the normal start up procedure, without taking permanent system initialization responsibilities. The temporary autostart

option might allow linking of additional ROM's and command vocabularies into Max-FORTH or even multitasking without interfering with the primary autostart abilities. The temporary autostart option is also useful to modify baud rates, etc., when there is no opportunity to catch the primary autostart by being "lower" in memory (i.e.: primary autostart pattern located at \$B600 or \$0400). A ROM higher in the memory map can still do its thing and then allow the primary search to find its mark.