8. ALTIO

AltIO serves as an Alto Input-Output package which does all the input-output for Maxc except for the disks. Specifically it does input-output for the Alto monitor, the Diablo printer, the Ethernet and the IMP (ARPA Network).

8.1. Calling AltIO

AltIO may be called either from Midas or directly from the Alto Executive. In Midas, selecting "AltIO" with the mouse causes a menu of AltIO subcommands to appear. Selecting "Do-It" causes AltIO to be started with no options selected; in this case, AltIO performs no initialization but simply resumes the microprocessor at its current location. If other subcommands are selected before "Do-It", they modify the initial actions of AltIO in various ways.

When AltIO is called directly from the Alto Executive, the same options may be specified by switches. The general form of the AltIO command line is:

> AltIO/switches parameter/switch parameter/switch ...

The switches immediately following "AltIO" correspond to Midas "AltIO" menu items in the following way:

Switch	Menu Item	Description
/B	Boot-MExec	Boot Micro-Exec into Maxc memory and start it.
/H	Dont-Go	Leave Maxc halted rather than resuming it.
/P	Protect	Enter "protected" mode in AltIO. This means that AltIO will allow only those commands that examine (but do not write into) memory and that do not change machine state.
/R	Reset-Memory	Reset the memory system.
/S	Start-Tenex	Start Tenex, first booting Micro-Exec from disk as for /B and then executing an automatic "Go" command.

The optional additional parameters may be specified only in the Executive command line; there is no way to invoke them from Midas.

filename/L	Load a PDP-10 "save"-format file into Maxc memory (default extension .SAV). Used for loading PDP-10 diagnostics when Micro-Exec won't work.
n/U	Use disk unit n rather than unit 0 in Boot-MExec and Start-Tenex commands, where 0 corresponds to unit A and 7 to H.

8.2. AltIO Commands

When AltIO is running, the Alto monitor is divided by black lines into three windows. If Tenex is running, the top window continuously displays the addresses and contents of five interesting Maxc main memory variables. The middle window belongs to a command processor for controlling AltIO itself. The bottom window implements the Maxc controlling terminal used by Tenex and by stand-alone programs such as Micro-Exec and PDP-10 diagnostics.

Keyboard input is directed to whichever window currently contains the blinking cursor. The middle and bottom unmarked keys on the right side of the keyboard may be used to direct input to the middle and bottom windows respectively. Keyboard input is automatically switched to the bottom window when Maxc is started by commands such as "Boot" or "Go", and to the middle window when Maxc stops for any reason.

The AltIO command processor has two heralds, a number sign (#) which indicates that only commands that examine memory are legal, and a colon (:) which indicates all commands are legal.

AltIO commands consist of a single letter or other special character, optionally preceded by a numeric argument. The argument is an octal number of up to 40 bits, depending on the command. While typing an argument, one may use the editing characters control-A, control-W, and Delete to erase, respectively, a single character, a word, or the entire command.

Many AltIO commands request confirmation with "[confirm]". The confirming character is a Return or period. Most of AltIO's commands are intentionally the same as in NVIO; the ones for which this is true are marked with a superscript "1".

n/[1]
    Change the current Maxc memory location to n; and print out the contents of that location in the form:

        aa bbbbbb cccccc

    where:

        aa (0 to 17) is the last 4 bits (i.e., the "tag" bits) of the 40-bit Maxc memory word, it is not printed out if zero;

    and

        bbbbbb and cccccc are the left and right half 18 bits of the PDP-10 (Maxc) word.

/
    Print out the current location.

n<cr>[1]
    Put the 40-bit number n into the current location. The first 4 bits input are the last 4 bits of the memory. Thus one can think of Maxc as a 36-bit machine (like the PDP-10), right adjusted in a 40-bit field, rather than a 36-bit machine left adjusted in a 40-bit word as is actually the case.

    This command is only legal after a command which prints out a Maxc memory location.

<lf>[1]
    (lf means linefeed). Change the current location to the current location plus 1, and print out the (new) current location.

n<lf>[1]   Equivalent to n, <carriage return>, <linefeed>.

^[1]   Change the current location to the current location minus 1, and print out the
                 (new) current location.

n^[1]   Equivalent to n, <carriage return>, <up arrow>.

<tab>[1]   Change the current location to the location pointed to by the current location. Print
                 out its contents.

B[1]   Boot Micro-Exec from Save Area 1 of physical disk unit A.

nB[1]   Boot Micro-Exec from Save Area 1 of physical disk unit n (0=A, 1=B, etc.) The
                 parameter n is remembered for subsequent "B" commands without arguments.

D   Toggle on or off a switch causing typeout on the Maxc controlling terminal (the
                 bottom display window) to be copied to the Diablo printer. This switch is normally
                 off, but turning it on is useful when debugging Tenex or doing memory maintenance.

E   Print out the current state of Maxc's PDP-10 emulator. This includes the program
                 counter and all the accumulators. This command may be issued only when Maxc is
                 stopped.

nG[1]   Go to Maxc location n. Precisely: stop the Maxc microprocessor (cleanly if possible),
                 then tell it to begin executing PDP-10 instructions at Maxc main memory location n.

G   Go to the starting address of the currently loaded Maxc program, which is pointed to
                 by Maxc memory location 7.

H[1]   Halt Maxc (cleanly if possible).

L   Prompt for a filename (default extension .SAV), then interpret that file as a PDP-10
                 "save"-format (not "ssave") file and load it into Maxc memory. The program's
                 starting address is stored in location 7 so that a "Go" command without arguments
                 will start the program. The file is expected to be in the 36-bit binary format defined
                 by FTP. Retrieving a PDP-10 "save"-format file using the FTP program will cause
                 such a file to be created.

M[1]   Halt Maxc (cleanly if possible) and return control to   Midas. Midas is resumed in
                 whatever state it was in when control was last transferred to AltIO from Midas. This
                 is true even if AltIO has been exited and other Alto programs run in the meantime.

nM[1]   Go to   Microprocessor location n. (If you use this command, you had better know
                 what you are doing.)

P[1]   Protect NVIO. The herald becomes a number sign (#); only those commands that
                 examine memory (and other state varibles) are legal.

nP[1]   If the number typed is the   Pass(word) number: AltIO becomes un   protected; the herald
             becomes a colon (:) and all commands are legal. Currently the pass(word) number is
             3301; this should be easy to remember as it is the smallest number which is the sum
             of two different cubes in two different ways. $(14^3 + 1^3 = 11^3 + 12^3$; octal of course.)

Q   Quit: return control to the Alto Executive, after first stopping Maxc if it is running.

R[1]   Resume Maxc (the microprocessor is told to proceed). If a reset was previously done
             ("21M" to ODT or "21;G" to Midas), PDP-10 interpretation will begin at the starting
             address pointed to by main memory location 7; otherwise, Maxc will resume from
             where it was halted.

S[1]   Startup Tenex, booting Micro-Exec from disk unit A.

nS[1]   Startup Tenex, booting Micro-Exec from disk unit n (0=A, 1=B, etc.)

W   Print out the current state of Maxc (running, stopped, or at a micro breakpoint).

Z   Zap Maxc memory: reset and reconfigure the entire memory system. Useful when
             the memory has become hung up. This command may be issued only while Maxc is
             stopped.