

address in the monitor to end a program. This will cause problems. It is always safe to jump to address 0, which restarts the monitor program correctly.

If you don't want to clear the screen

Reset the stack to OC33H  
then jump to PARSE

#### Notes on PIO Operation.

The Nascom I has on board two totally uncommitted 8 bit parallel I/O ports complete with handshake lines, in the shape of an MK3881 Z80 - PIO. The PIO is, in itself, a fairly complicated processor, which needs programming before it will operate in any of its 4 modes:

OUTPUT	MODE 0
INPUT	MODE 1 (automatically set on PIO Reset)
BIDIRECTIONAL	MODE 2
CONTROL	MODE 3

It is not the purpose of these notes to describe in detail these operational modes, but to help clear up a few common problems encountered in controlling the PIO.

One very important fact to note is that the PIO is not reset by the RESET button on the keyboard. This resets the CPU only, NOT the PIO. It may be reset in two ways. The simplest is to switch the power off and on again; a bit drastic but the PIO does have automatic power on reset. The second method (shown in fig.3 ) is to apply an M1 without either RD or IORQ. It should, however, be pointed out that, since the CPU can be reset, it is always possible to regain control of the PIO in software, by simply reprogramming it.

Now to 'interrupts'. Don't forget that the PIO is designed to operate in the Z-80 Interrupt Mode 2, so before doing anything put the CPU into this mode by executing 'IM 2' (HEX code ED 5E). Remember that a CPU reset puts the Z-80 back to Interrupt Mode 0 clears the I register, and dissables CPU interrupts (having no effect on the PIO).

In Interrupt mode 2, the CPU finds the address of the interrupt routine, by loading the Program Counter (P.C.) with the contents of the memory address. This is formed by the I register (high byte), and the interrupt vector sent from the interrupting port (low byte).

/ . . .