LEO-1 Homebrew Computer

Register Board Implementation

5th December 2016

The LEO-1 contains eight general-purpose 16-bit registers, r0 to r7. These registers are all identical and can be used interchangeably in all instructions that utilize registers. Each register is implemented by using eight integrated circuits; two 74HC273 octal D-type flip-flops and six 74HC244 octal line drivers. The register board therefore contains 64 ICs just for the registers, with the high 8 bits running down the left side of the board and the low 8 bits running down the right. The centre of the board contains the control logic, and the bottom contains the LED drivers.

There are three sets of inputs to the register board; the 16-bit Register Input bus (RIN), three 3-bit selectors (REGSA[3], REGSB[3] and REGSC[3]), and three miscellaneous control signals, /RESET, /REGSE and /REGWRI. There are also three output buses designated A bus, B bus and C bus. All of these inputs and outputs connect to the Control Board, and the A bus also connects directly to the ALU Board where it is permanently connected to the A-input of the ALU. The B bus can be routed to the B-input of the ALU by the Control Board, and the C bus is used by the Control Board for performing conditional branches and memory writes.

/RESET is connected to the /MR input of all the register ICs and resets them to zero when the system reset button is pressed. /REGSE is low when registers are being driven onto the A, B and C buses, and when /REGWRI goes low, one register will be written to.

The three 3-bit selector groups correspond to the three output buses. These selectors drive three 74HC138 decoders each of which selects exactly one register onto a bus. The values present on each 3-bit selector therefore determine which register will be driven onto the corresponding bus while /REGSE is low.

The C Bus selector has a second function; it also selects the register that will be written to when /REGSE is low and /REGWRI goes low. Therefore the C-selected register is also the register that will accept the data present on the RIN Bus at the moment of the write. This arrangement allows three registers to be accessed in one instruction. For example, during a memory read, the data written to the C-selected register could be the result of a memory load from an address determined by the sum of two other registers (A and B). Also, during a memory write, the address being written to could be the sum of the A and B selected registers while the source of the data is the C-selected register. This last example is the main reason for the existence of the C bus, although it was also convenient to use it for checking conditional branches.