

## Serial PC-AT Keyboard to ASCII Interface

This interface converts serial keyboard data from a PC-AT keyboard into a parallel data plus strobe output simulating an ASCII keyboard. It also transmits data back to the PC-AT keyboard to illuminate the 3 keyboard LEDs (Num Lock, Caps Lock, Scroll Lock). It includes a Reset button which resets both its PIC16F84 and the Microtan.

There is an excellent article here which describes how the PC-AT keyboard works:

<http://www.philipstorr.id.au/pcbook/book3/keyboard.htm>

This interface employs 2 lookup tables; one for 'No Shift' key presses, one for 'Shift' key presses. Other special keys catered for are:

Backspace	Backspace (0x08)
Delete	Rubout (0x7F)
ESC	Escape (0x1B)
Space-bar	Space (0x20)
Num Pad Enter	Line Feed (0x0A)
Keypad Enter	Return (0x0D)

If the Caps Lock is set, the interface output is the same as if the Shift had been used. Caps Lock plus Shift reverts back to the No Shift state. So if Caps Lock is used, then the Numeric Pad has to be used to generate numbers (rather than the top row of the keypad). Also, the comma (,) is not available in this state (unless Shift is pressed at the same time).

There is a shortcoming in the code; the Keyboard LED states are not reset when the Reset button is pressed so can be out of sync until the appropriate Lock key is pressed again.

So as a substitute for a real ASCII Keyboard for the Microtan, this interface provides everything the Microtan Keypad does plus it gives the full range of ASCII codes for all the characters depicted on the PC-AT Keyboard. What it does not provide are 'Control' codes in the range 0x00 to 0x1F except for the special keys listed above and a true Caps Lock facility. These would require the employment of a further 2 lookup tables. I will have to look into this.

### **Further Modifications (Version 2.0) – June 2016**

The original code contained 2 Look-up tables; one for when only one key is pressed (No\_shift) and one for when the Shift key is pressed (or Caps Lock is set) as well. By removing the lower part of the tables, which is not used by the program anyway, it has been possible to squeeze in two extra

tables; one for when Caps Lock is set and one when the Control key is pressed. With these mods, the interface now behaves the same as the Tangerine ASCII Keyboard. Because the PC-AT keyboard key layout is different to that of the Tangerine Keyboard, ASCII codes for S1 (0x1B) – S5 (0x1F) are obtained by pressing the Control key along with Numeric Pad numbers 1 -5.

During the initialisation phase the program now transmits a Keyboard reset command to the keyboard. This clears the three keyboard LEDs after first flashing all three LEDs on. This ensures that when the interface Reset button is pressed, the Keyboard LEDs are always cleared. The program then goes on to clear its corresponding status flags. When the keyboard is powered up, the LEDs will flash twice; once when the keyboard receives power and again when the program sends the keyboard reset command.

### **Further Modification (Version 3.0) – October 2016**

The standard ASCII keyboard has a mechanical ‘CAPS LOCK’ switch which is mostly left in the ‘on’ position when used with the Microtan. Therefore, on power-up, the keyboard is already in the CAPS LOCK ‘on’ state. This is not the case with the Serial Keyboard Interface – on power-up the converter defaults to the CAPS LOCK ‘off’ state. A simple modification has been made in version 3 so that following a RESET, the interface sends a CAPS ON signal to the keyboard.