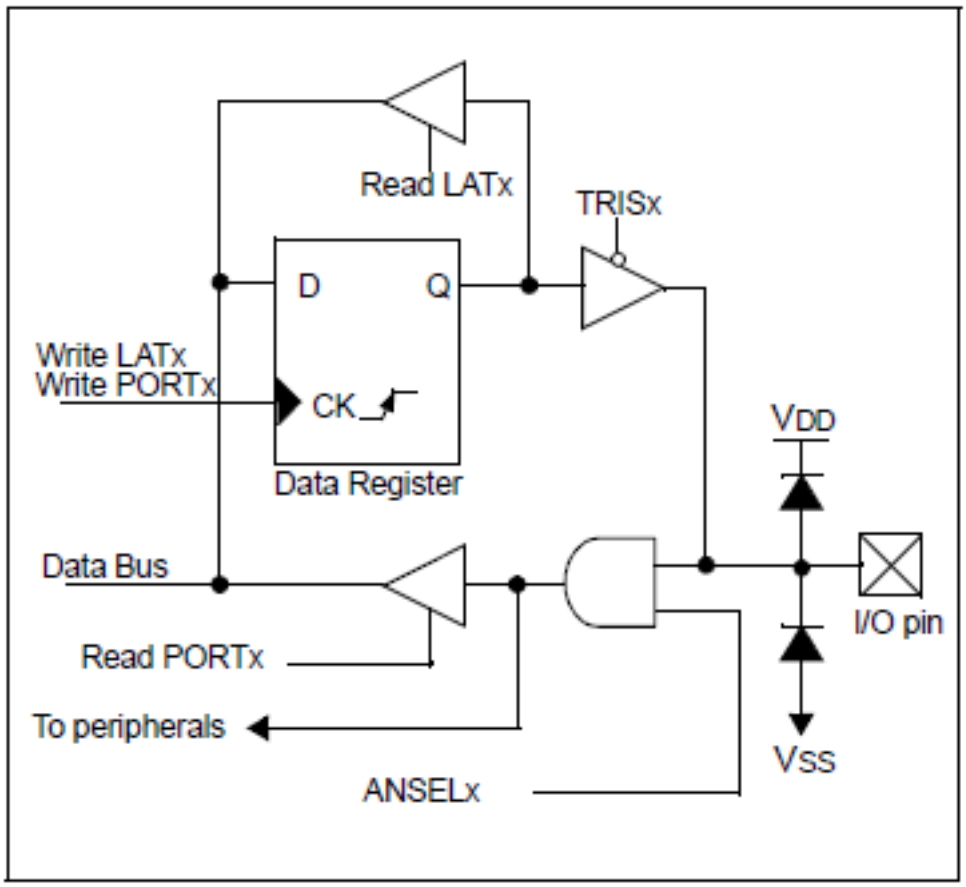
La información está sacada de esta web y del datasheet del PIC18F2550.
http://stackoverflow.com/questions/2623428/difference-between-port-and-latch-on-pic-18f

Each port has three registers for its operation. These registers are:

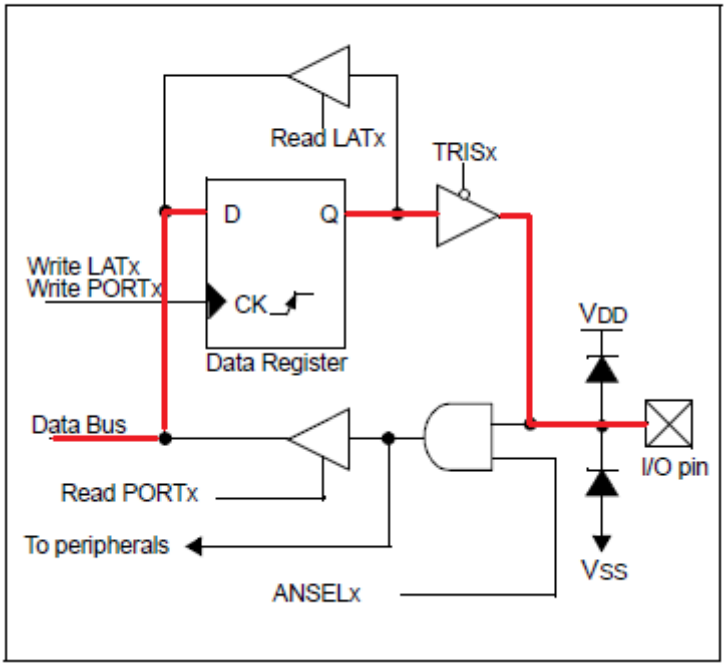
- TRIS register (data direction register)
- PORT register (reads the levels on the pins of the device)
- LAT register (output latch)

FIGURE 12-1: GENERIC I/O PORT OPERATION



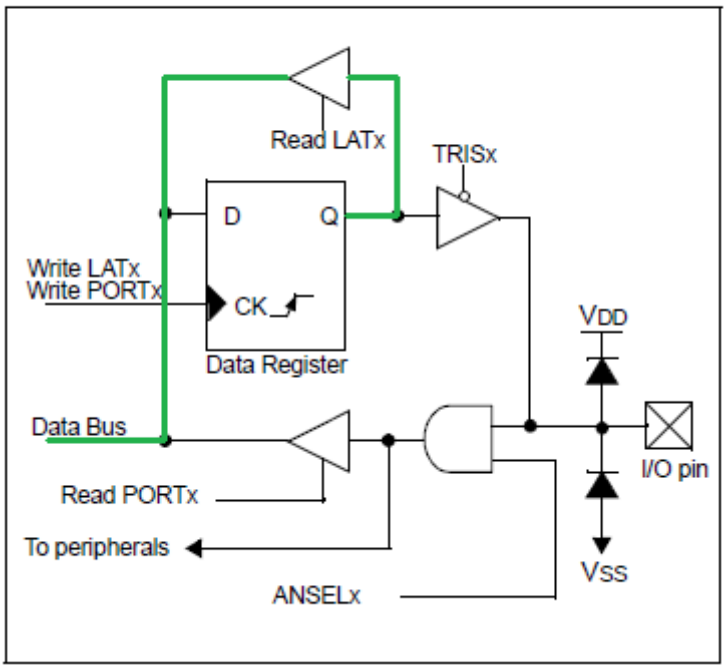
When you write a bit in a I/O pin, you're storing this bit from Data Bus to the Data Register (D-FlipFlop). If TRISx of this bit is 0, so data from Q of the Data Register will be in the I/O pin. Write in LATx or PORTx is the same. See below in red:

FIGURE 12-1: GENERIC I/O PORT OPERATION



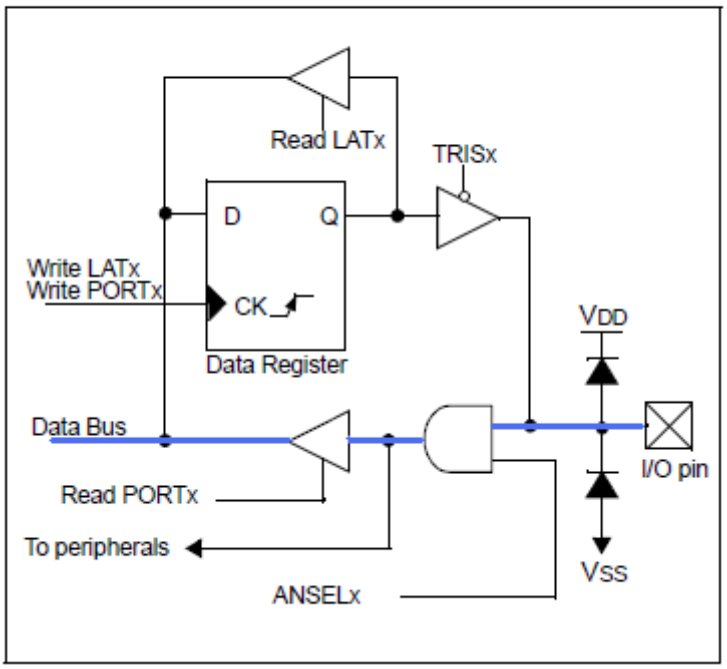
On the other hand, read from LATx is different of read from PORTx.
When you're reading from LATx, you're reading what is in the Data Register (D-FlipFlop). See picture below in green:

FIGURE 12-1: GENERIC I/O PORT OPERATION



And when you read from PORTx, you're reading the actual I/O pin value. See below in blue:

FIGURE 12-1: GENERIC I/O PORT OPERATION



PIC uses read-modify-write to write operations and this can be a problem, so they use this shadow register to avoid it.

My recommendation is to regard the PORT values as read-only. The LAT values may be read or written, but the value read will be the last value written, not the input value of the pin.

When you want to read whether some external hardware is driving a pin high or low, you must set the pin to input mode (with TRIS or the DIR register), and you must read PORTx. That read tells you if the actual voltage at the pin is high or low.
When you want to drive a pin high or low, you must set the pin to output (with TRIS or the DIR register); you should write the bit to the LATx register.
(Writing that bit to the PORTx register may *seem* to do the right thing: that pin will -- eventually -- go high or low as commanded. But there are many cases -- such as when some other pin on that port is connected to an open-collector bus -- that writing to one bit of the the PORTx register will mess up the state of the other pins on that port, leading to difficult-to-debug problems).