CS499 - Fundamentals of Robotics
Lab Assignment #3
Fall 2010

A seven-segment display is often used to display a digit in such devices as digital clocks or other readouts. Several separate displays are usually used together to display several digits. Each display has seven LEDs, labelled `'a'` thru `'g'`, and a decimal point (`dp`). (The displays we have appear to have two `dp`'s, but only the right-most one is connected.) Internally, the displays are connected such that one side of each LED is common (connected together with the other LED's, while the other end is brought out on a separate pin on the device. Thus there are a total of 9 pins used for the device - 7 segments, the decimal point, and a common lead. Our display actually has 10 pins - the middle pins on each side are connected together internally, so either pin (or both) can be used for the common. This particular type of display is called "common anode," since the anode (positive) sides of each LED are connected together.

If wired up separately, each display would require 9 pins total. However, to save pins, multiplexing is used. For each display, the pins connected to `'a'` are wired together, `'b'`s are connected together, etc. Each of these are connected to one of seven port pins on the Arduino. The `dp`'s can also be connected together and attached to an eighth pin. Then, the common pin of each display is connected to its own separate pin. In operation, the pins that make up a particular digit are pulled low, and then the common pin for that particular digit is made to go high. If this scheme is used for each digit, and the switching between each digit occurs quickly enough, it will appear that all the digits are displayed, each with their own number. The total number of pins that are necessary are 7 (8 with the `dp`) plus one common pin per digit: $n = 7 + d$, where d is the number of digits. Especially for a larger number of digits, this scheme reduces the total number of pins considerably.

For this exercise, you are to wire up at least two digits (we have a third digit, but the `e` segment is burned out), and create a digital counter. When a button is pushed, the display should count, one count per second. When the button is pushed a second time, the current count count should show in the display. A third push should reset and start the count over again.