

Purpose: More programming practice, using AVRmicrocontrollers, using your programming environment, simple binary inputs, software delays, using “enum” and “#define”.

Assignment: Make a simple “pong” game. We will simulate a ball moving back and forth by lighting the LEDs one at a time on the bar LEDs. We will add a keypad providing “left” and “right” buttons. When the “left button” is pushed and the light is in the leftmost position, the ball will move to the right. Similarly, when the “right button” is pushed and the light is in the rightmost position, the ball will move to the left. If a button is pushed at the wrong time, the opposite player will then “serve”.

**Prelab:** There will be a series of CodeLab problems that must be completed before coming to lab. This will be up to 50% of your lab grade. The completed CodeLab problems will constitute almost all of the code needed for this lab.

**Notes:**

Your code will first set the initial state to "left\_serve" (see starter code).

In an infinite loop you will 1) adjust the "state", 2) adjust the LED pattern, and 3) delay.

The state change is based on the buttons, the LED pattern, and the current state. Adjust it as follows:

- If the left button is pressed: the new state will be moving\_right if the ball is at the leftmost position and the new state will be right\_serve otherwise.
- If the right button is pressed: the new state will be moving\_left if the ball is at the rightmost position and the new state will be left\_serve otherwise.
- If neither button is pushed and the LED pattern is 0x00 (the ball has shifted off the end): set the state to left\_serve if the ball was last moving to the right, and set it to right\_serve otherwise.
- If none of the above: leave state unchanged.

Based on the new state adjust the LED pattern as follows: if the ball is moving right or left then shift accordingly. If the state is left or right serve, then place the ball accordingly. (Use `switch/case`.)

Delay each time around the main loop. Use a “for” loop with an empty body and count up to some value. Delay should be such that it takes one second for the ball to travel from one side to the other. HINT: use an `unsigned int` as your loop control variable (counter).

Doing the above will get you a “C” grade. Expanding the “court” from 8 LEDs to 10 LEDs will raise your grade by one letter. Adding scoring and displaying the score in the LCD will also raise your grade by one letter. For this you must also add a third button for reset. Reset the score when either player reaches a score of 15 or the reset button is pressed.

Have your working code with comments and proper formatting checked off by the TA.