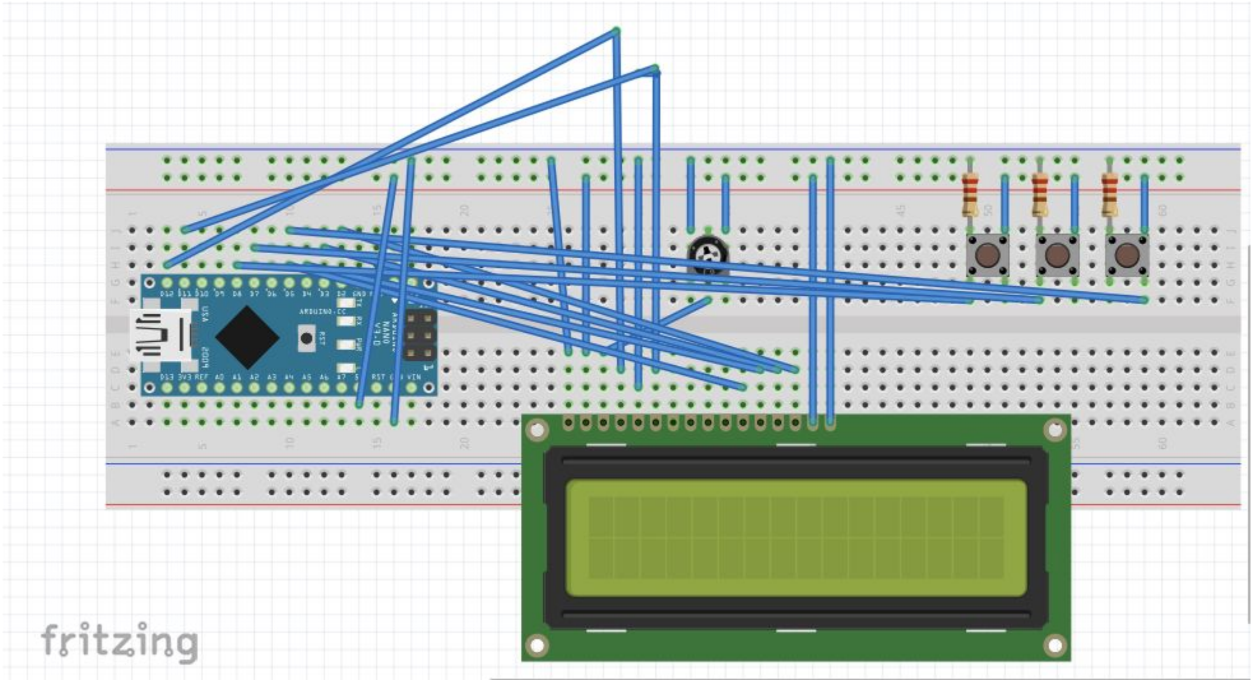
The super simple scoreboard doesn't have much to it. We track the score of two teams by using tactile switches, and then a reset line. Here's the circuit diagram:



I know you can't really see where the wires are going, but this is the layout of my circuit. The data pins of the lcd screen are routed to digital pins 2-5. The backlight controls are set so that A goes to 5V and K is grounded. V_0 is adjusted with the pot. V_{DD} and V_{SS} go to 5V and ground, respectively. Digital pins 6, 7, and 8 are connected to the switches such that the home score is controlled by 6, visitor by 7, and the reset option to 8. Here's the code:

[code]

```
// include the lcd library
#include <LiquidCrystal.h>

// initialize the library with the numbers of the arduino pins
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

// before the setup loop I'll define some global variables
int homeScore;
int visitorScore;
const int buttonHomePin = 6;
const int buttonVisitorPin = 7;
const int buttonResetPin = 8;
int buttonState = 0;
```

```
void setup() {
  // set up the LCD’s number of columns and rows:
  lcd.begin(16, 2);

  setupScreen();
}

//
void setupScreen()
{
  lcd.clear();
  // Make the home score on the left, and the visitor several spaces away
  lcd.setCursor(0,0);
  lcd.print("Home     Visitor");
  // start it at zeros
  homeScore = 0;
  visitorScore = 0;
}

// each of the following lines just set up the logic for the buttons
// if it sees a high level on the respective pins, home/visitor scores increase by 1.
// the reset line will take you back to the setupScreen() stage
void loop() {
  buttonState = digitalRead(buttonHomePin);
  if(buttonState == HIGH)
  {
    homeScore++;
    delay(500);
  }
    buttonState = digitalRead(buttonVisitorPin);
  if(buttonState == HIGH)
  {
    visitorScore++;
    delay(500);
  }

   buttonState = digitalRead(buttonResetPin);
  if(buttonState == HIGH)
  {
    setupScreen();
  }
  // These lines set the location of the scores and print them.
   lcd.setCursor(3, 1);
   lcd.print(homeScore);
   lcd.setCursor(12, 1);
   lcd.print(visitorScore);
}
[/code]
```

In the code that defines the button commands, I was able to get around the bouncing issue by using delay(500).