Patrick Austin

CPE 301 – 1104

Assignment # 6

10/17/2016

Assignment description:

In this lab we built a circuit and wrote code to select values 0-F on a seven segment LED. First I created the circuit using the breadboard, the Arduino, an inverter chip, and a resistor chip. Then I wrote code that would send appropriate output bits to the LED for each possible input.

Problems encountered:

My group members had difficulty getting their circuits to show the expected results and we spent some time trying to troubleshoot, but my board/code worked successfully on the first shot. I did have to consult the datasheet to learn about how the pullup resistors worked, but once I knew the procedure there was no significant difficulty.

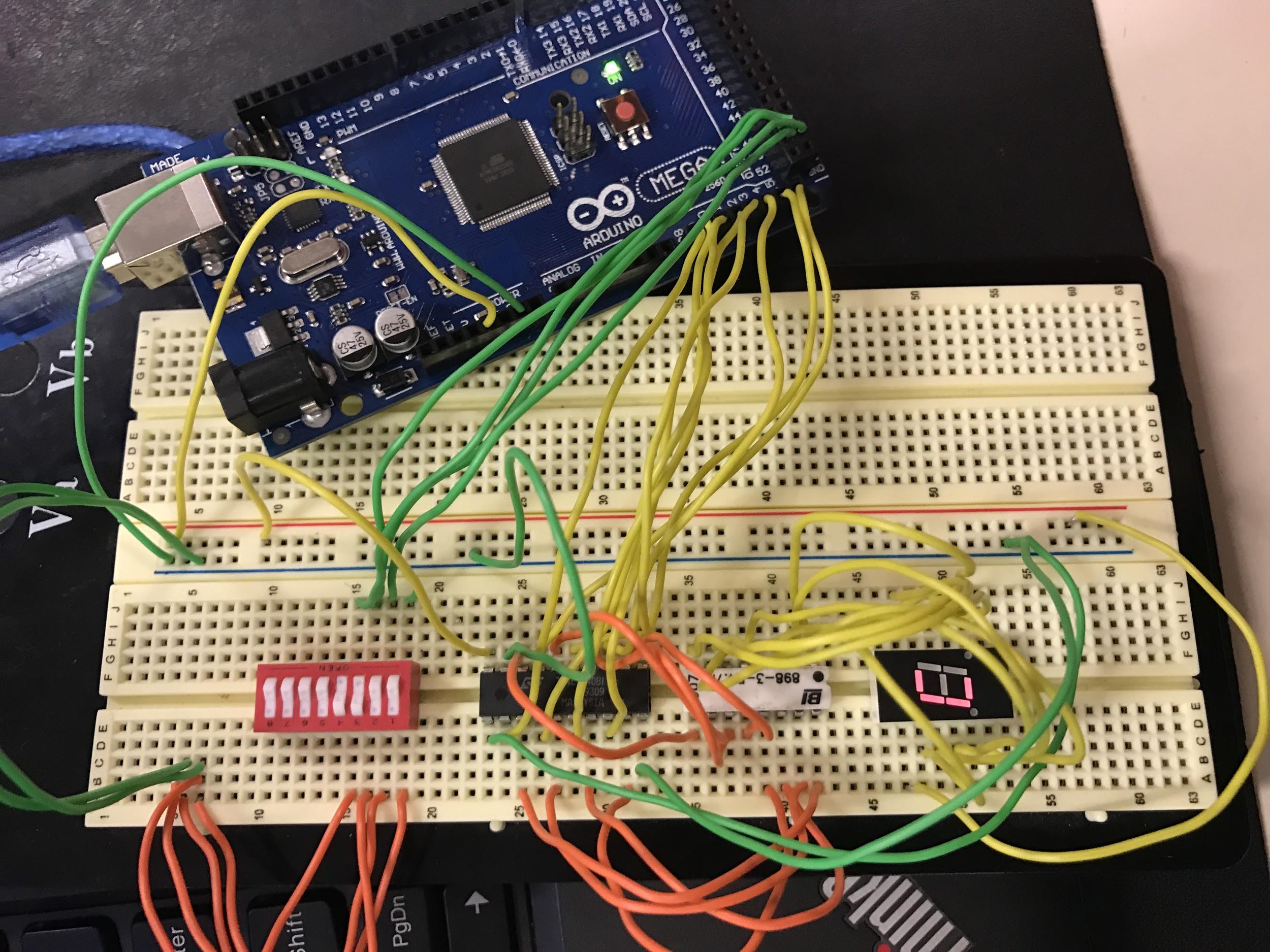
Lessons learned:

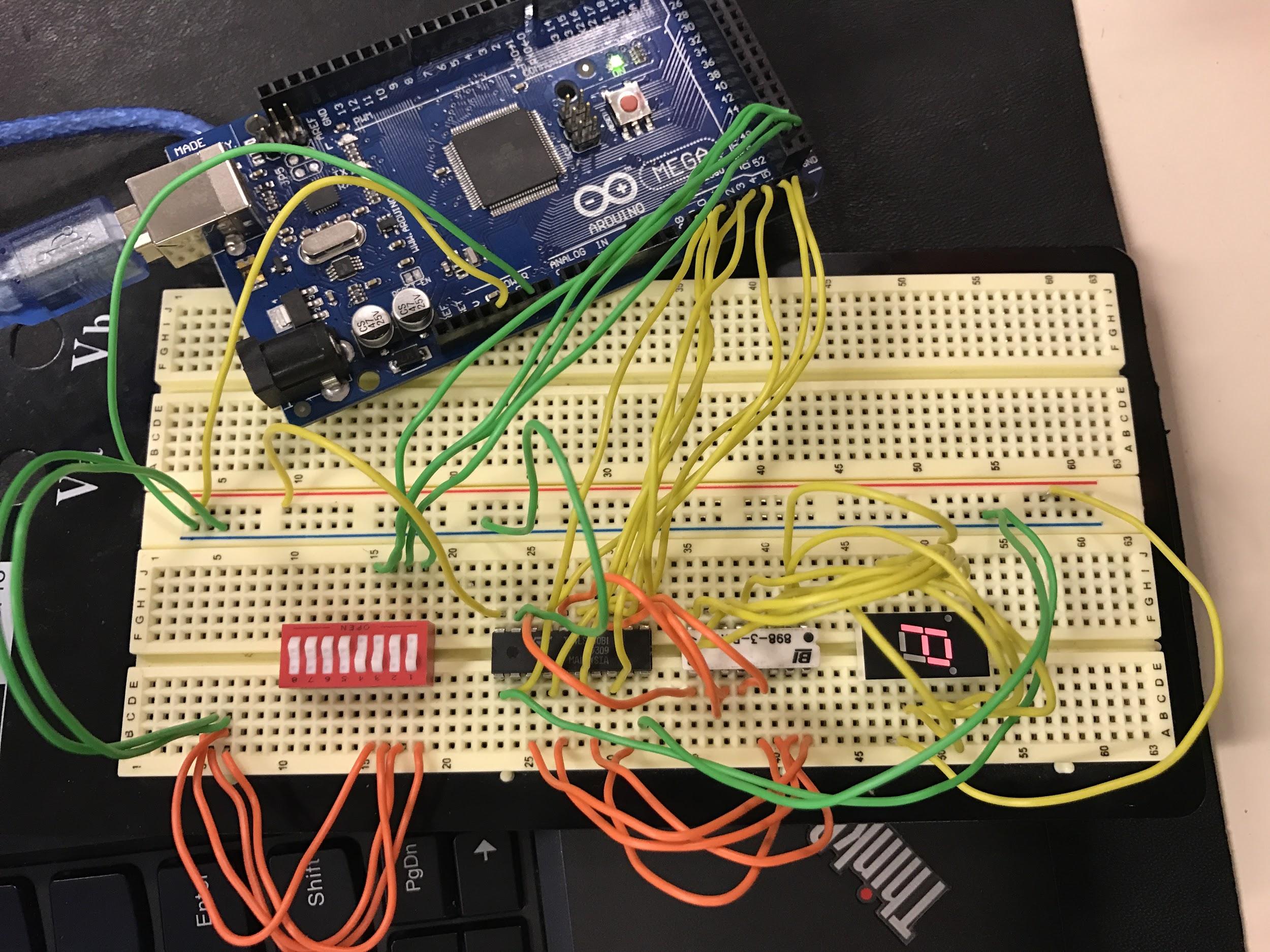
Learned how to use the pullup resistors on the Arduino to resist the inputs coming in to the board. Reviewed and reinforced the usual basics about board design, microcontroller GPIO, and coding in the Arduino environment.

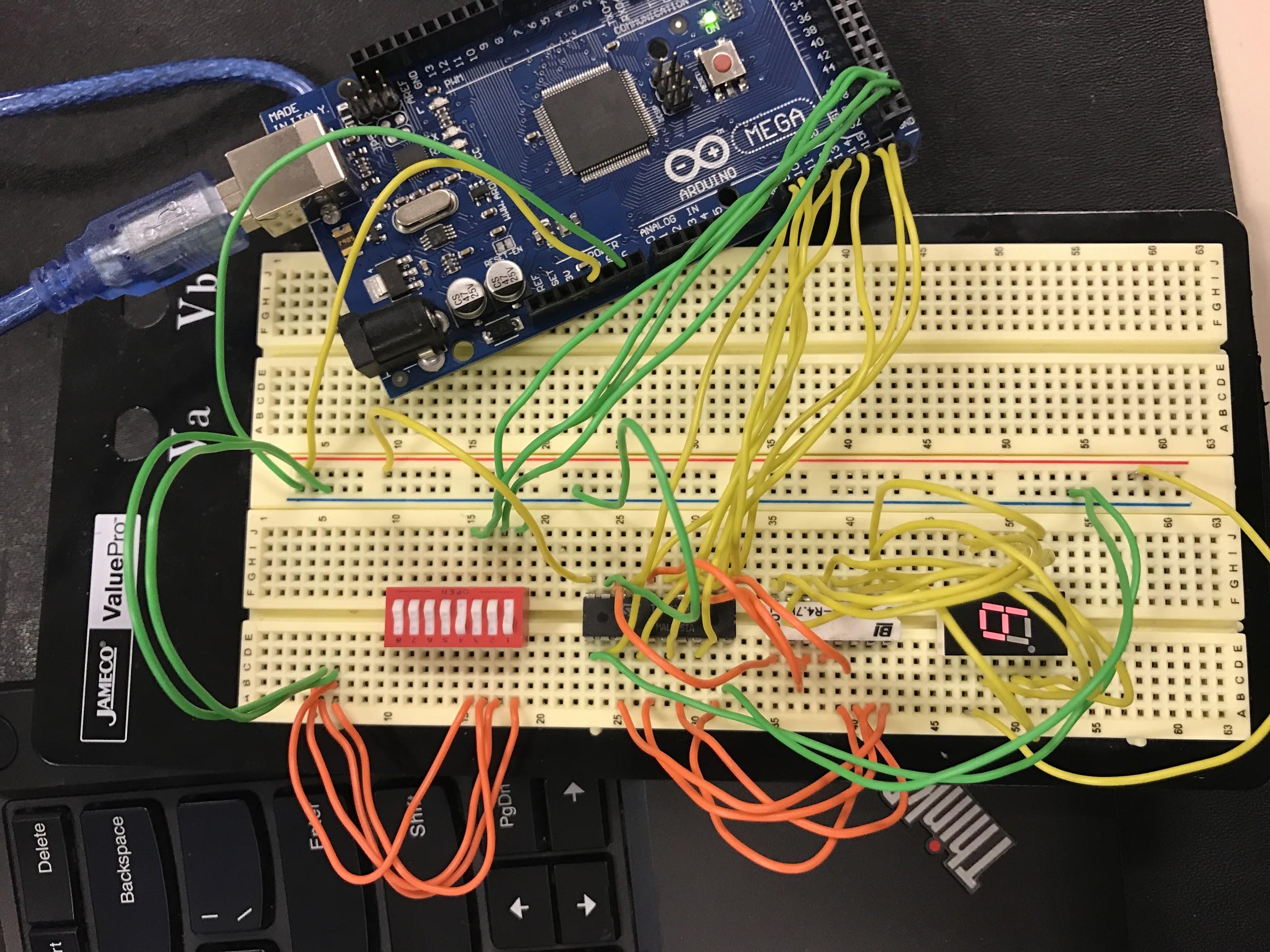
Description of completed lab:

The circuit worked as expected, showing values 0-f on the appropriate input. The inputs are resisted with pullups, and the outputs are resisted with the white resistor chip.

Circuit photos, showing a few inputs:







The circuit ran this program:

//Patrick Austin

//CPE 301 Lab 6

//Revision Number 1

//Revision date: 10/17/2016

//global hardware pointers

volatile unsigned char\* portDDRB = (unsigned char\*) 0x24;

volatile unsigned char\* pinB = (unsigned char\*) 0x23;

volatile unsigned char\* portB = (unsigned char\*) 0x25;

volatile unsigned char\* portDDRK = (unsigned char\*) 0x107;

volatile unsigned char\* portK = (unsigned char\*) 0x108;

/\* Lookup table for seven segment values: such that

segmentvalues[x] contains the segments to light

to display x. convention: bit 0 (rightmost) specifies

whether to light segment a, bit 1 b, ..., bit 6 f. Bit 7

is for the decimal point and is unused in this program,

so it is always set to 0. \*/

static const unsigned char segmentValues[] =

{ 0b00111111, 0b00000110, 0b01011011, 0b01001111,

0b01100110, 0b01101101, 0b01111101, 0b00000111,

0b01111111, 0b01100111, 0b01110111, 0b01111100,

0b00111001, 0b01011110, 0b01111001, 0b01110001 };

void setup()

{

//set DDR B 0-3 to 0 ie input. pin B3 will be the MSB

\*portDDRB = \*portDDRB & 0xF0;

//set port B 0-3 to 1 to enable pullup resistors.

\*portB = \*portB | 0x0F;

//set port K 0-7 to 1 ie output. pin K7 will be the MSB

\*portDDRK = 0xFF;

}

void loop()

{

//poll the input pins and send the corresponding table value to the LED segments

\*portK = segmentValues[ \*pinB & 0x0F ];

}

Picture of code compiling successfully:

