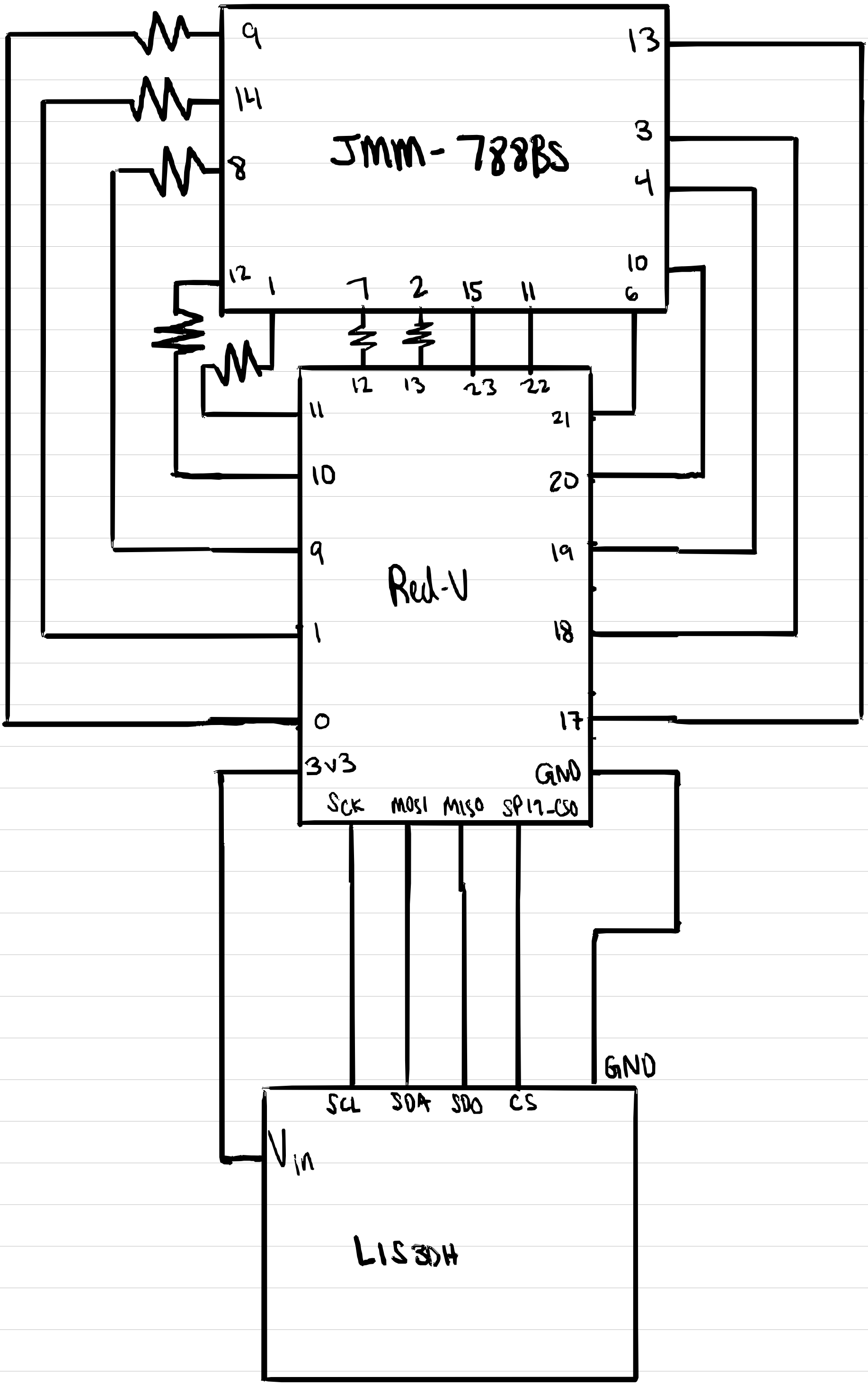
Lab 8: Digital Level

1. I spent 6 hours on this lab.

2. Circuit Schematic   All resistors are 330 Ω



```c
// lab8_CJ.c
// Christian Johnson
// chrjohnson@hmc.edu
// 11/9/24

#include "EasyREDVIO_ThingPlus.h"

// Constant Definitions

#define COL1 17
#define COL2 18
#define COL3 19
#define COL4 20
#define COL5 21
#define COL6 22
#define COL7 23

#define ROW1 0
#define ROW2 1
#define ROW3 9
#define ROW4 10
#define ROW5 11
#define ROW6 12
#define ROW7 13


/*
  Helper function that sets all the LEDs in the matrix to 0
*/
void turn_off(void) {
  // Initilalizing X (COL) to be 1
  digitalWrite(COL1, 1);
  digitalWrite(COL2, 1);
  digitalWrite(COL3, 1);
  digitalWrite(COL4, 1);
  digitalWrite(COL5, 1);
  digitalWrite(COL6, 1);
  digitalWrite(COL7, 1);

  // Initilalizing Y (ROW) to be 0
  digitalWrite(ROW1, 0);
  digitalWrite(ROW2, 0);
  digitalWrite(ROW3, 0);
  digitalWrite(ROW4, 0);
  digitalWrite(ROW5, 0);
  digitalWrite(ROW6, 0);
  digitalWrite(ROW7, 0);
}

/*
  Helper function which turns on an LED given an int x and int y then turns off all LEDs
*/
void led_helper(int x, int y) {
  // Set the x column to 0
  digitalWrite(x, 0);

  // Set the y row to 1
  digitalWrite(y, 1);

  delayLoop(500);

  turn_off();
}

int main(void) {
  volatile uint8_t debug;
volatile int16_t x, y, disx, disy;
int row, col;

spiInit(10, 1, 1);    // Initialize SPI pins

// Set up the GPIO pins as outputs (LEDs)

// // X axis LEDs: GPIO 18:23 and 17
pinMode(COL1, OUTPUT);
pinMode(COL2, OUTPUT);
pinMode(COL3, OUTPUT);
pinMode(COL4, OUTPUT);
pinMode(COL5, OUTPUT);
pinMode(COL6, OUTPUT);
pinMode(COL7, OUTPUT);

// Initilalizing X (COL) to be 0
digitalWrite(COL1, 1);
digitalWrite(COL2, 1);
digitalWrite(COL3, 1);
digitalWrite(COL4, 1);
digitalWrite(COL5, 1);
digitalWrite(COL6, 1);
digitalWrite(COL7, 1);

// // Y axis LEDs: GPIO 9:13, 1, and 0
pinMode(ROW1, OUTPUT);
pinMode(ROW2, OUTPUT);
pinMode(ROW3, OUTPUT);
pinMode(ROW4, OUTPUT);
pinMode(ROW5, OUTPUT);
pinMode(ROW6, OUTPUT);
pinMode(ROW7, OUTPUT);

// Initilalizing Y (ROW) to be 1
digitalWrite(ROW1, 0);
digitalWrite(ROW2, 0);
digitalWrite(ROW3, 0);
digitalWrite(ROW4, 0);
digitalWrite(ROW5, 0);
digitalWrite(ROW6, 0);
digitalWrite(ROW7, 0);

// Setup the LIS3DH
spiWrite(0x20, 0x77);    // highest conversion rate, all axis on
spiWrite(0x23, 0x88);    // block update, and high resolution

// Check WHO_AM_I register. should return 0x33
debug = spiRead(0x0F);

while (1) {
  // Collect the X and Y values from the LIS3DH
  x = spiRead(0x28) | (spiRead(0x29) << 8);
  y = spiRead(0x2A) | (spiRead(0x2B) << 8);

  // Pseudocode

  if (x > 7142) {
    col = COL7;
  } else if (x > 4284) {
    col = COL6;
  } else if (x > 1426) {
    col = COL5;
  } else if (x > -1426) {
    col = COL4;
  } else if (x > -4284) {
      col = COL3;
    } else if (x > -7142) {
      col = COL2;
    } else {
      col = COL1;
    }

     if (y > 7142) {
      row = ROW1;
    } else if (y > 4284) {
      row = ROW2;
    } else if (y > 1426) {
      row = ROW3;
    } else if (y > -1426) {
      row = ROW4;
    } else if (y > -4284) {
      row = ROW5;
    } else if (y > -7142) {
      row = ROW6;
    } else {
      row = ROW7;
    }

    // call LED helper on the designated LED
    led_helper(col, row);

    delayLoop(500);
  }
}
```

Accelerometer Calibration

X/Y axis -3 -2 -1 0 1 2 3

- Flat (0): x= -272 y= -192
- Max tilt left (-3): y=-10000
- max tilt right (-3): y= 10000
- max tilt left (3): y= -10000

accelerometer output range: -10000 to 10000 = 20000

-3 to 3 = 6

4. Yes, my program works as intended.

I'm able to reach all dots in the 7x7 array.