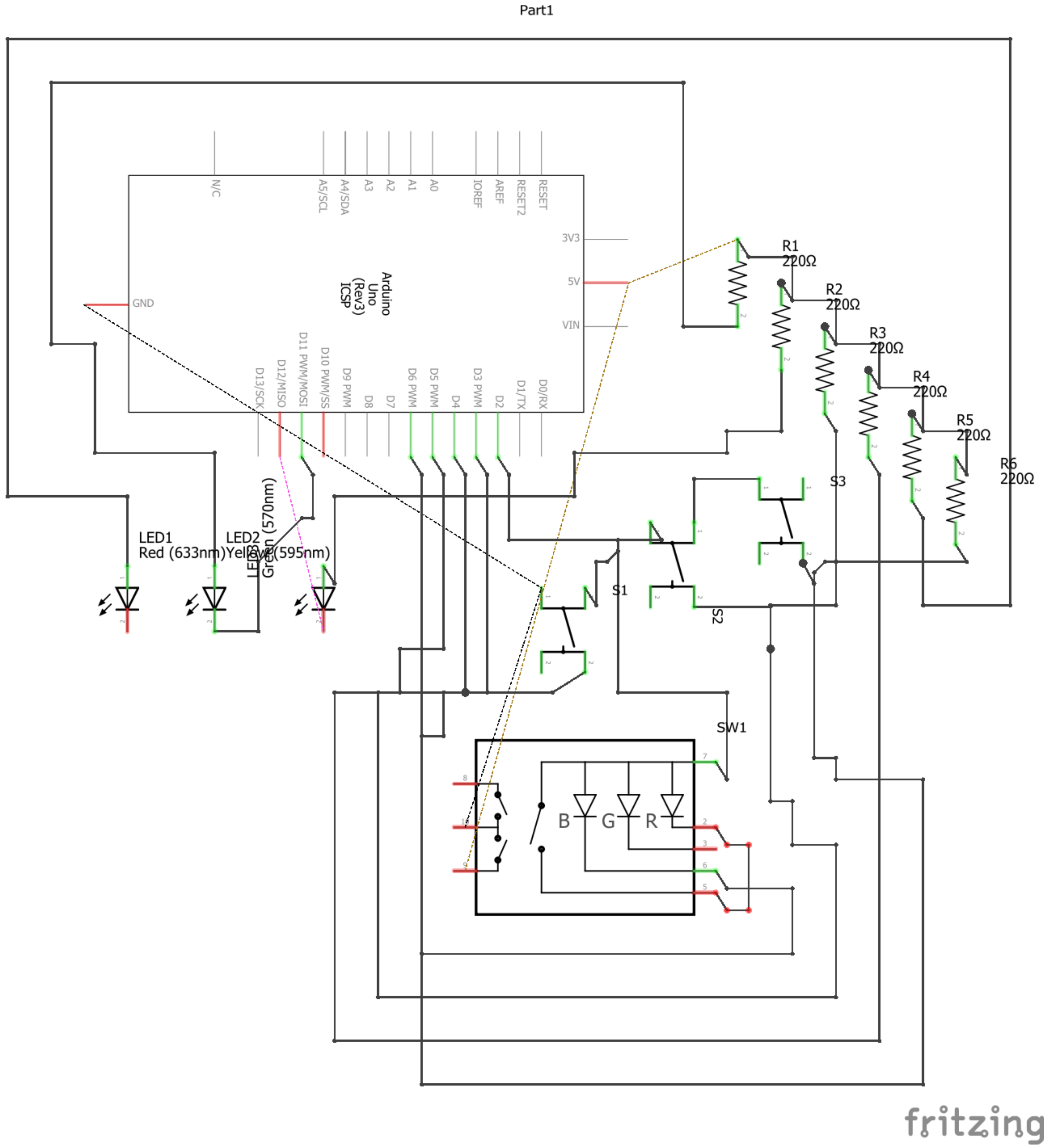
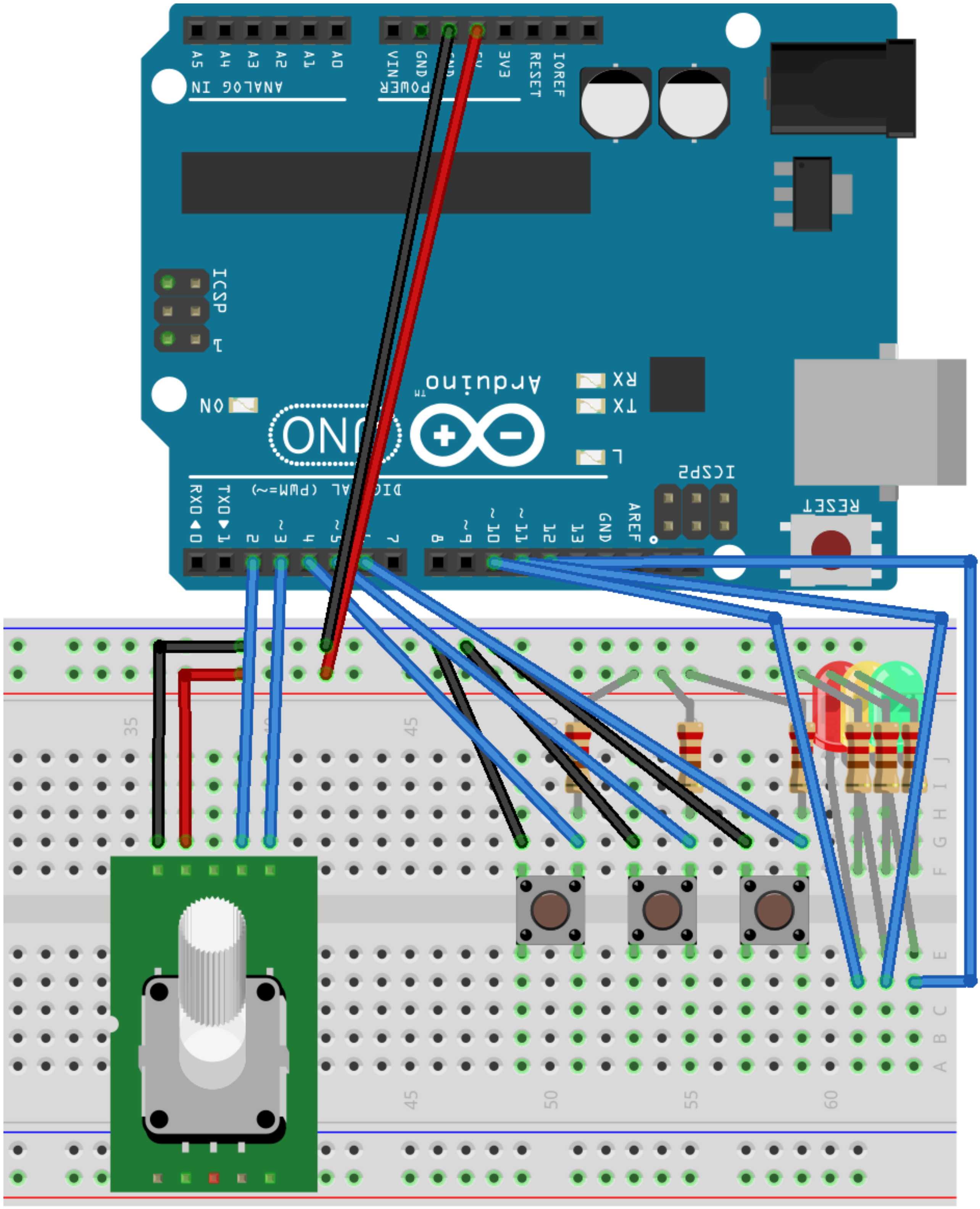
Jonne Kaajalahti

Lesson 7 raportti

Piirilevydiagrammi




A5
A4
A3
A2
A1
A0
ANALOG IN
VIN
GND
GND
5V
3V3
RESET
IOREF
POWER
ICSP
1
ON
UNO
Arduino™
RX
TX
L
ICSP2
RESET
RX0◄0
TX0►1
2
~3
4
~5
~6
7
DIGITAL (PWM=~)
8
~9
~10
~11
12
13
GND
AREF
35
40
45
50
55
60
J
I
H
G
F
E
D
C
B
A

Koodi, A

```
int currentState = 0;
int lastState = 0;
int counter = 0;
int outputA = 2;        // pin numbers for rotary encoder outputs A and B
int outputB = 3;
int x = 0;

// initialize button pins for all LED colors
int buttonPinRed = 4;
int buttonPinYellow = 5;
int buttonPinGreen = 6;

// initialize LED pin numbers
int ledPinRed = 10;
int ledPinYellow = 11;
int ledPinGreen = 12;

// initialize buttonstatuses for all colors
int buttonStatusRed = 0;
int buttonStatusYellow = 0;
int buttonStatusGreen = 0;


void setup() {
    // initialize serial for counter reading and pinmodes and
    // for buttons and LEDs
    Serial.begin(9600);
    pinMode(buttonPinRed, INPUT);
    pinMode(buttonPinYellow, INPUT);
    pinMode(buttonPinGreen, INPUT);
    pinMode(ledPinRed, OUTPUT);
    pinMode(ledPinYellow, OUTPUT);
    pinMode(ledPinGreen, OUTPUT);
    // reset the variable used in the rotary encoder
    lastState = digitalRead(outputA);
}

// function to shutdown all leds
void shutAllLeds(){
    digitalWrite(ledPinRed, HIGH);
    digitalWrite(ledPinYellow, HIGH);
    digitalWrite(ledPinGreen, HIGH);
}


void loop() {
    Serial.println(counter);    // print the counter value to serial console
    shutAllLeds();              // shut all LEDs in the beginning of each loop
    currentState = digitalRead(outputA);
    if(currentState != lastState){
        // if the rotary encoders position has changed increase or decrease the cou
nter variable
        if(digitalRead(outputB) != currentState){
            counter ++;
        }
        else{
            counter --;
        }
    }

    // prevents the counter variable from going negative
    if (counter < 0){
        counter = 0;
    }

    // depending which button is pressed, corresponding LED will light up
    if(digitalRead(buttonPinRed) == LOW){
        x = 1;
    }
    if(digitalRead(buttonPinYellow) == LOW){
        x = 2;
    }
    if(digitalRead(buttonPinGreen) == LOW){
        x = 3;
    }

    /*
    the value of the counter corresponds with the frequency of the LEDs flicker
    if the counters value is over 10 the led will be at its max brightness
    */
    if(x == 1){
        if(counter > 10){
            digitalWrite(ledPinRed, LOW);
        }
        else{
            digitalWrite(ledPinRed, LOW);
            delay(counter);
            digitalWrite(ledPinRed, HIGH);
            delay(counter);
        }
        digitalWrite(ledPinYellow, HIGH);
        digitalWrite(ledPinGreen, HIGH);
    }
    // same as previous but for the yellow led
    if(x == 2){
        if(counter > 10){
            digitalWrite(ledPinYellow, LOW);
        }
        else{
            digitalWrite(ledPinYellow, LOW);
            delay(counter);
            digitalWrite(ledPinYellow, HIGH);
            delay(counter);
        }
        digitalWrite(ledPinRed, HIGH);
        digitalWrite(ledPinGreen, HIGH);
    }
    // and for the green led
    if(x == 3){
        if(counter > 10){
            digitalWrite(ledPinGreen, LOW);
        }
        else{
            digitalWrite(ledPinGreen, LOW);
            delay(counter);
            digitalWrite(ledPinGreen, HIGH);
            delay(counter);
        }
        digitalWrite(ledPinYellow, HIGH);
        digitalWrite(ledPinRed, HIGH);
    }
    // set the lastState to currentState so that the code runs properly when the loop starts again
    lastState = currentState;
}
```

Loopin alussa pitää painaa nappia, jotta sitä vastaavan ledin kirkkautta pystyy säätämään (ledin kirkkaudessa kolme astetta: pois, himmeä, kirkas). Toista nappia painamalla ainoastaan sitä vastaava ledi syttyi. Kun säädintä kierrettiin, tietyn arvon ylitettyään ledi oli kirkkaimmillaan.