

Elektroniikan työpaja
Programming microcontrollers
Sulautetut ohjelmistot & IoT
PTIVIS17S

Jonne Kaajalahti, 1601968

Anton Leinonen, 1703154

LESSON 8

– Motors



TURUN AMMATTIKORKEAKOULU
TURKU UNIVERSITY OF APPLIED SCIENCES

RAPORTTI | TIIVISTELMÄ

TURUN AMMATTIKORKEAKOULU

Elektroniikan työpaja | Sulautetut ohjelmistot & IoT

10.12.2019 | Sivumäärä

Antti Haarto

Jonne Kaajalahti & Anton Leinonen

LESSON 8 - MOTORS

Tehtävän ensimmäisessä osiossa kontrolloitiin DC-tuuletinta Arduinon "serial" -konsolin avulla. Toisessa osiossa samaa tuuletinta kontrolloitiin kulma-anturin avulla. Lisätehtävänä kytkimme myös LCD-näytön, jossa luki tuulettimen reaaliaikainen nopeus prosentteina.

SISÄLTÖ

| | |
|----------------------------|---|
| 1 JOHDANTO | 4 |
| 2 LESSON 8, PART A, TASK 1 | 5 |
| 3 LESSON 8, PART A, TASK 2 | 7 |
| 4 PÄÄTÄNTÖ | 9 |

1 JOHDANTO

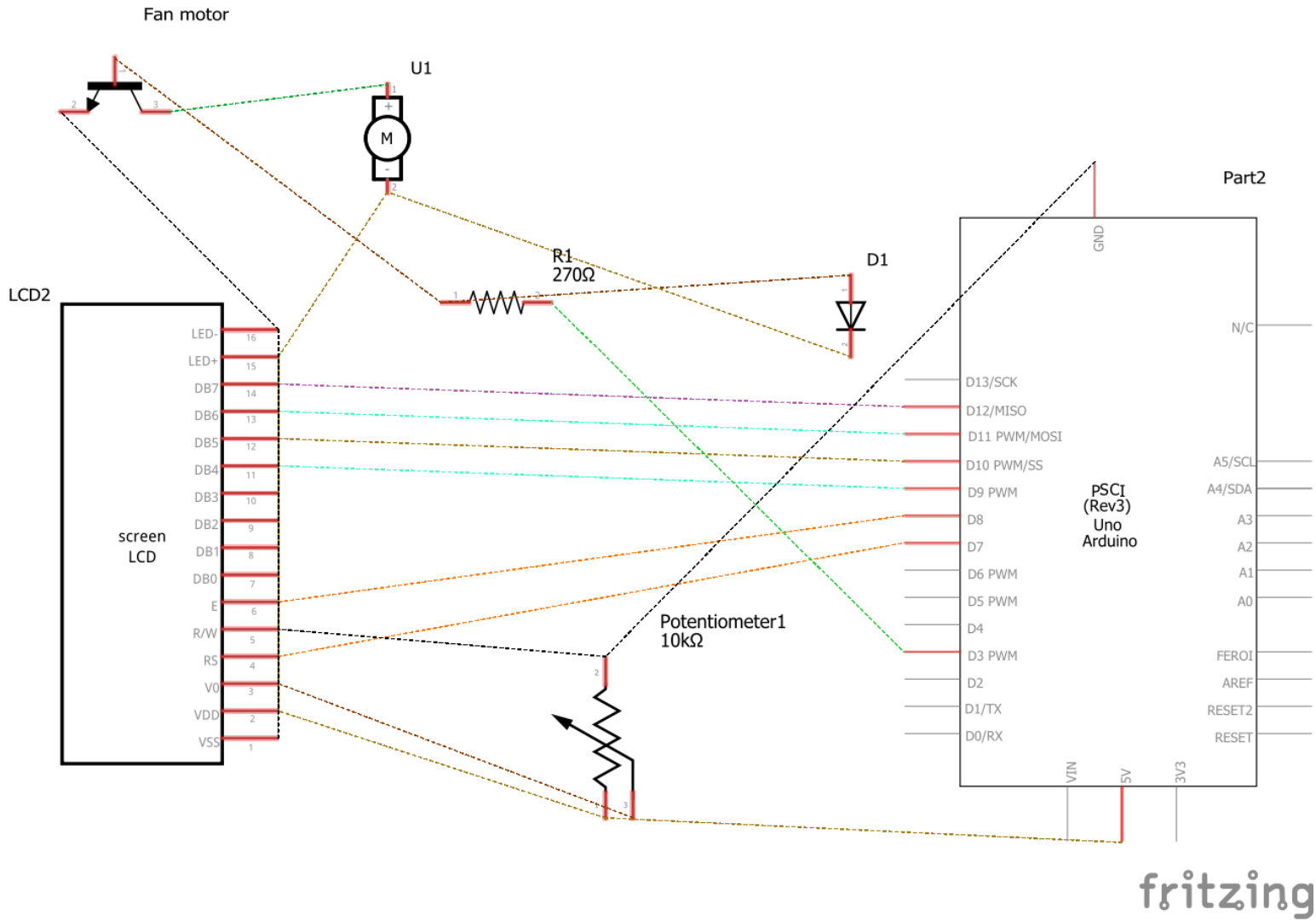
Tehtävän ensimmäisessä osiossa käytimme Arduinon konsolia kontrolloimaan DC-tuuletinta. Toisessa osiossa käytimme kulma-anturia kontrolloidaksemme samaa tuuletinta. Lisätehtävänä kytkimme piiriin LCD-näytön, jossa luki tuulettimen reaaliaikainen nopeus prosentteina.

2 LESSON 8, PART A, TASK 1

```

1.      /* Task 1
2.      Jonne Kaajalahti, Anton Leinonen
3.      */
4.
5.      int motorPin = 3;
6.      int speed = 0;
7.      int currentState = 0;
8.      int lastState = 0;
9.      // int counter = 0;
10.     int outputA = 5;           // pin numbers for rotary encoder outputs A and B
11.     int outputB = 6;
12.
13.
14.     void setup() {
15.         pinMode(motorPin, OUTPUT);
16.         Serial.begin(9600);
17.         lastState = digitalRead(outputA);
18.     }
19.
20.     void loop() {
21.
22.         Serial.println("Speed 0 to 255A");
23.         speed = Serial.parseInt();
24.         while(!Serial.available()){
25.             speed = Serial.parseInt();
26.         }
27.         if (speed >= 0 && speed <= 255){
28.             analogWrite(motorPin, speed);
29.         }
30.
31.
32.         currentState = digitalRead(outputA);           //Code to change the value
        depending on which way the rotary encoder was turned
33.         if(currentState != lastState){
34.             if(digitalRead(outputB) != currentState){ //If the state has changed
        we raise or lower the counter variable
35.                 counter ++;
36.             }
37.             else{
38.                 counter --;
39.             }
40.         }
41.
42.         if (counter < 0){                               //We stop the value from
        going into negative because that would break the code
43.             counter = 0;
44.         }
45.     }

```



Kuva 1. Kytentäkaavio tehtävään 1.

3 LESSON 8, PART A, TASK 2

```

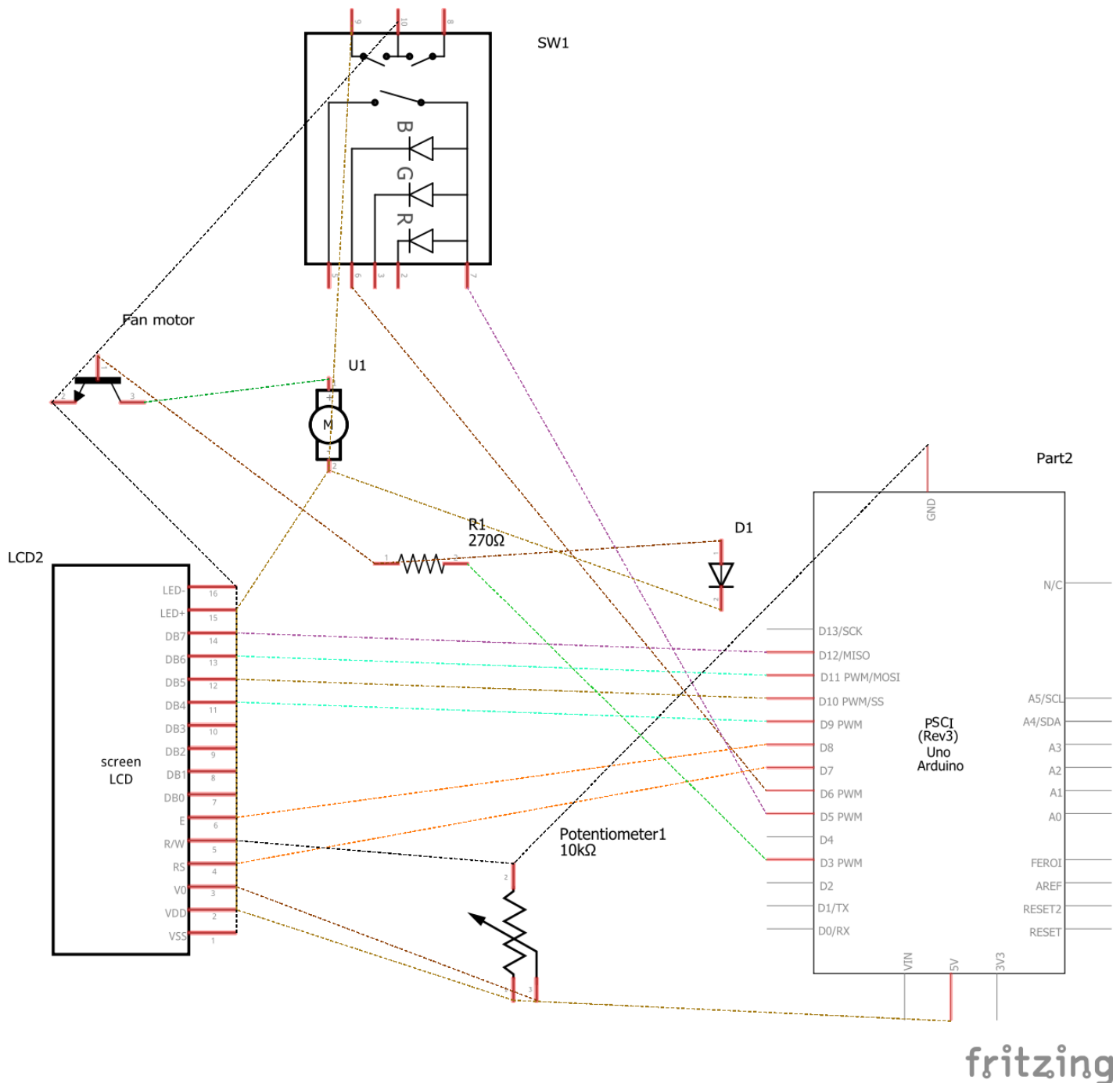
1.      /* Task 2
2.      Jonne Kaajalahti, Anton Leinonen
3.      */
4.
5.      #include <LiquidCrystal.h>
6.
7.      //          BS  E  D4  D5  D6  D7
8.      LiquidCrystal lcd(7, 8, 9, 10, 11, 12); // initialize LCD displays pins
9.
10.     int motorPin = 3;
11.     int speed = 0;
12.     float speedPercentage = 0;
13.     int currentState = 0;
14.     int lastState = 0;
15.     int counter = 0;
16.     int outputA = 5;           // pin numbers for rotary encoder outputs A and B
17.     int outputB = 6;
18.
19.
20.     void setup() {
21.         pinMode(motorPin, OUTPUT);
22.         Serial.begin(9600);
23.         lastState = digitalRead(outputA);
24.         lcd.begin(16,2);
25.     }
26.
27.     void loop() {
28.         speedPercentage = (speed / 255.0) * 100.0;
29.         // Display speed (%) on LCD
30.         lcd.setCursor(0,0);
31.         lcd.print("Speed      %");
32.         lcd.setCursor(9,0);
33.         lcd.print(speedPercentage, 1);
34.         delay(50);
35.
36.
37.         if (speed >= 0 && speed <= 255){
38.             analogWrite(motorPin, speed);
39.         }
40.
41.         Serial.println(speed); // print the counter value to serial console
42.         currentState = digitalRead(outputA);
43.         if(currentState > lastState){
44.             // if the rotary encoders position has changed increase or decrease
the counter variable
45.             if(digitalRead(outputB) != currentState){
46.                 counter --;
47.             }
48.             else{
49.                 counter ++;
50.             }
51.             speed = counter * 10;
52.         }
53.
54.         if (counter < 0){ // stop the counter from
going negative in order for the code to run properly
55.             counter = 0;
56.         }

```

```

57.
58.     if (speed >= 255) {
59.         speed = 255;
60.     }
61.
62.         // set the lastState to currentState so that the code runs properly when
the loop starts again
63.         lastState = currentState;
64.     }

```



Kuva 2. Kytentäkaavio tehtävään 2.

4 PÄÄTÄNTÖ

Molemmat tehtävät suoritettu onnistuneesti. Koodit ja piirit toimivat.