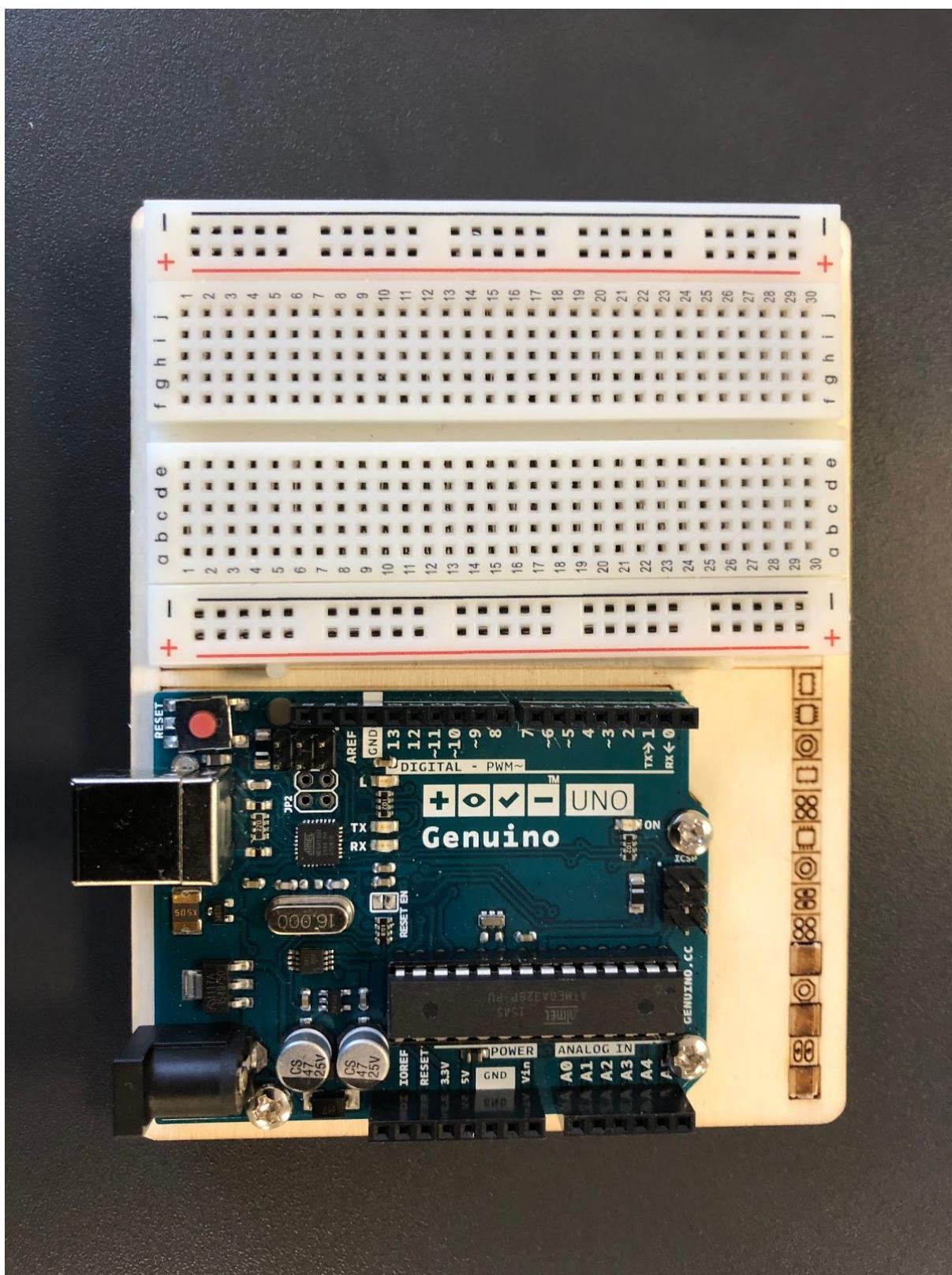


Arduino Projekt

Frederik Nielsen



Indhold

Indhold	2
Opgavebeskrivelse	5
Intro øvelsen:	5
Øvelse 1. Trafiklys:	5
Øvelse 2. Robot	5
Øvelser i "Arduino Project Book"	5
Project 01	5
Project 02	5
Project 03	5
Project 04	5
Project 05	6
Project 06	6
Project 07	6
Project 08	6
Project 09	6
Project 10	6
Project 11	6
Project 12	6
Project 13	6
Project 14	7
Project 15	7
Problemformulering	7
Øvelse 1. Trafiklys:	7
Øvelse 2. Robot	7
Øvelser i "Arduino Project Book"	7
Project 01	7
Project 02	7
Project 03	8
Project 04	8
Project 05	8
Project 06	8
Project 07	8
Project 08	9
Project 09	9
Project 10	9
Project 11	9
Project 12	10

Project 13	10
Project 14	10
Project 15	10
Design	11
Arduino	11
Arduino Robot	11
C++	11
Teori	11
Øvelsen 1. Trafiklys:	11
Øvelsen 2. Robot:	11
Dokumentation	12
Konklusion	12
Kilder	12
Kode:	13
Øvelse 1. Trafiklys:	13
Øvelse 2. Robot	14
Øvelser i "Arduino Project Book"	16
Project 02	16
Project 03	17
Project 04	18
Project 05	19
Project 06	20
Project 07	21
Project 08	22
Project 11	23
Project 12	25
Billeder:	27
Øvelse 1. Trafiklys:	27
Øvelse 2. Robot	27
Øvelser i "Arduino Project Book"	28
Project 01	28
Project 02	28
Project 03	29
Project 04	29
Project 05	30
Project 06	30
Project 07	31
Project 08	31
Project 11	32
Project 12	32

Opgavebeskrivelse

Intro øvelsen:

I skal nu programmere en Arduino for første gang, der skal installeres et Arduino IDE program på computeren. Det er en god ide at starte med at lave simple projekter som f.eks. at få en LED diode til at blinke. Øvelserne i "Arduino Projects Book" skal udføres og dokumenteres.

Øvelse 1. Trafiklys:

Øvelsen er nu at lave et trafiklys. hvor der også er mulighed for at trykke på en knap som illustrerer en fodgængerovergang.

Øvelse 2. Robot

Øvelsen går ud på at i skal programmere en robot, så den kan hente en kop kaffe for enden af en planlagt rute og returnere til start positionen.

Øvelser i "Arduino Project Book"

Alle opgaverne skal programmeres i Arduino IDE pånær Project 01

Project 01

I opgaven skal man få en LED pære til at lyse ved at klikke på en knap, i den ene del af opgaven skal man få pæren til at lyse når to knapper er trykket ned og i den anden del behøver kun en af knapperne at blive trykket på for at tænde pæren.

Project 02

I opgaven skal man få 3 LED pærer til at lyse skiftevis når man klikker på en knap, og det skal gøres ved at skrive en kode i Arduino IDE og så flashe det til Arduino pladen.

Project 03

I opgaven skal man bruge en temperaturmåler(tmp) til at få 3 LED pærer til at lyse op forskelligt efter hvor varmt tmp læser.

Project 04

I opgaven skal man ved at bruge 3 fototransistorer få en RGB LED pære til at lyse i forskellige farver ud fra det lys de 3 fototransistorer opfanger.

Project 05

I opgaven skal man styre en servomotor ved brug af et potentiometer

Project 06

I opgaven skal man få en højtalder til at afspille musik ud fra det lys en fototransistor opfanger.

Project 07

I opgaven skal man få 4 individuelle knapper til at virke på 1 pin port hvor alle knapperne afspiller deres egen lyd i højtaleren. Man skal bruge forskellige resistorer så Arduinoen kan finde ud af hvilken knap der bliver trykket på.

Project 08

I opgaven skal man lave et timeglas ved at bruge en tilt sensor og 6 LED pærer. der skal tændes en pære hver 10 minut og hvis man flipper boardet skal den starte forfra.

Project 09

I opgaven skal man bruge motoren til at få en LED pære til at lyse når man drejer på motoren og man skal kunne få motoren til at dreje når man klikker på en knap.

Denne opgave er ikke blevet lavet da vi ikke havde et batteri.

Project 10

I opgaven skal man bruge motoren til at bygge en billedfremviser.

Denne opgave er ikke blevet lavet da vi ikke havde et batteri.

Project 11

I denne opgave skal man få LCD skærmen til at vise nogen svar til et spørgsmål når man flipper boardet da der er også skal bruges en tilt sensor

Project 12

I denne opgave skal man få en servomotor til at virke som en lås, den skal virke ved at man banker på piezo'en der opfanger lyden og så åbner låsen. der skal også være en knap som kan åbne og lukke låsen.

Project 13

I denne opgave skal man ved at forbinde noget staniol til to ledninger opfange når man rør det med fingrene og så få en LED pære til at lyse op.

Denne opgave kunne ikke laves da vi ikke havde noget staniol

Project 14

I denne opgave skal man få Arduinoen til at snakke med pc'en og så tegne nogen billeder.
Denne opgave kunne ikke laves da vi ikke havde det rigtige program.

Project 15

I denne opgave skal man kunne optage lyd og så afspille det på pc'en.
Denne opgave kunne ikke laves da mikrofonen er en ting der ikke medfølger i sættet.

Problemformulering

Opgaverne Består af følgende ting

Øvelse 1. Trafiklys:

- Arduino
- Breadboard
- Ledninger
- LED Pærer
- Knapper

Øvelse 2. Robot

- Arduino Robot
- Tape til gulvet

Øvelser i "Arduino Project Book"

Project 01

- Arduino
- Breadboard
- Ledninger
- Knapper
- LED Pære

Project 02

- Arduino
- Breadboard
- Ledninger
- Knap
- LED Pærer

Project 03

- Arduino
- Breadboard
- Ledninger
- LED Pærer
- TMP måler

Project 04

- Arduino
- Breadboard
- Ledninger
- RGB Pære
- Fototransistorer

Project 05

- Arduino
- Breadboard
- Ledninger
- Potentiometer
- Servomotor
- Motor arm
- Capacitor

Project 06

- Arduino
- Breadboard
- Ledninger
- Piezo
- Fototransistor

Project 07

- Arduino
- Breadboard
- Ledninger
- Piezo
- Knapper

Project 08

- Arduino
- Breadboard
- Ledninger
- Tilt switch
- LED Pærer

Project 09

- Arduino
- Breadboard
- Ledninger
- Motor
- Knap
- MOSFET
- Batteri
- Diode

Project 10

- Arduino
- Breadboard
- Ledninger
- Motor
- H-Bro
- Knapper
- Potentiometer
- Batteri

Project 11

- Arduino
- Breadboard
- Ledninger
- LCD skærm
- Tilt switch
- Potentiometer

Project 12

- Arduino
- Breadboard
- Ledninger
- Servomotor
- LED pærer
- Piezo
- Knap
- Capacitor

Project 13

- Arduino
- Breadboard
- Ledninger
- LED Pære
- Staniol

Project 14

- Arduino
- Breadboard
- Ledninger
- Potentiometer

Project 15

- Arduino
- Breadboard
- Ledninger
- Optocoupler
- Mikrofon

Design

Arduino

Til opgaverne bruges der en Arduino som er en micro computer. En Arduino kan bruges til små iot projekter og man har adgang til I/O Pins som man kan koble op med elektroniske enheder og på den måde styre dem.

Arduino Robot

Arduino Robotten er en robot som er lavet til at efterligne de robotter som der bruges på f.eks Amazon's lager. den har 5 Ir sensorer som man kan bruge til at scanne gulvet med og så bestemme hvilken vej den skal køre, den har også et kompas som man ville kunne bruge til at styre hvilken vej den skal køre if f.eks et kryds af linjer. Den har også et LCD display som kan bruges til at udskrive data eller som en menu der styres af de 5 knapper.

C++

C++ er det sprog man bruger i Arduino IDE til at få Arduino'en til at gøre de ting man vil have den til. C++ bruges til andet end at programmere Arduino, det bruges nemlig også til at lave spil og desktop programmer.

Teori

Øvelsen 1. Trafiklys:

Min ide var at sætte 12 LED lys op og 2 knapper som skulle illustrere et lyskryds, de 6 af dem var lysene til bilerne og de andre 6 var til fodgængere. De 2 knapper skulle bruges til at aktivere fodgænger lyset. Trafiklyset skulle programmeres i c++ og så uploades til Arduinoen.

Øvelsen 2. Robot:

Min ide var at lave et program til robotten som skulle køre på main boardet og som så skulle bruge Ir sensorer til at bestemme om den er til venstre eller højre fra stregen. dens mål var så at stoppe ved et objekt.

Dokumentation

Jeg startede med at lave de 15 opgaver i bogen så jeg kunne lære Arduinoen bedre at kende. der var ikke de største problemer med opgaverne da de var godt beskrevet i bogen.

Der er Billeder af alle opgaverne nede i bilagene.

Efter det lavede jeg trafiklyset, da jeg havde lavet det på gf1 skulle jeg ikke rigtig gøre andet end at sætte ledningerne op og så bare uploadede den kode jeg lavede dengang, dog lavede jeg lidt ændringer så der ikke var så meget kode der gik igen.

Den sidste opgave jeg lavede var Robotten som skulle følge en linje. jeg indstillede den ud fra de instruktioner der stod på deres dokumentation og da den var kalibreret kørte den som den skulle et par omgange, efter jeg havde sat gul tape rundt i kanten. Efter det gad den ikke rigtigt at virke igen, men tror det har noget at gøre med den kalibrering som den laver lige inde programmet starter og da det ikke var en sort linje på hvid baggrund men i stedet på det mørkeblå gulv tror jeg nemt den har kunne forveksle den sorte linje med det blå gulv. Nu hvor den gule tape har sat sig fast til gulvet lyser den mørke farve lige igennem og har på fornemmelsen at det var derfor den ikke gad virke.

Konklusion

Jeg fik hurtigt lavet alle opgaverne da jeg havde prøvet at lege med arduino før og pga. det havde jeg ikke brug for at læse hvad den kunne og hvordan den virkede. Det gjorde desværre også at det hurtigt blev kedeligt da jeg på gf1 havde lavet alle opgaver som vi fik på nær den med robotten.

Lyskryds opgaven var som sagt hurtigt lavet da jeg kiggede efter et billede fra gf1 hvor der var vist hvad man skulle gøre. Så det fjernede delen hvor man planlægger sin udførelse af opgaven. Men i stedet for bare at tage alt koden og genbruge den fra dengang valgte jeg at skrive den om så den er på mit niveau indenfor programmering.

Robot opgaven virkede som en fed opgave i starten, men da jeg fandt ud af at de sensorer der sad på den ikke ligefrem var de bedste. gjorde det opgaven irriterende da det betød at for at få robotten til at klare ruten ville kræve at man var heldig da den kalibrerede dens sensorer. måden jeg så fik den til at virke på var det at sætte gul tape på begge sider af den sorte streg, og det virkede indtil den gule tape streg klistede sig helt fast til gulvet og så blev næsten helt gennemsigtig. Men jeg tro at hvis man bruger mere tape kan man få den til at virke hver gang.

Kilder

Kode:

Øvelse 1. Trafiklys:

```
const int lights[4][3] = {{A0,4,5},{6,1,7}, {13,12,11}, {9,10,8}};
const int delays[2] = {10000, 2500};

void setup(){
    for(int item = 0; item < 4; item++){
        for(int obj = 0; obj < 3; obj++){
            if(obj == 0) digitalWrite(lights[item][obj], HIGH);
            pinMode(lights[item][obj], OUTPUT);
        }
    }
    attachInterrupt(digitalPinToInterrupt(2), left_btn_pressed, CHANGE);
    attachInterrupt(digitalPinToInterrupt(3), right_btn_pressed, CHANGE);
}

void loop(){
    turn_green(lights[0], lights[1]);
    turn_red(lights[0], lights[1]);
    turn_green(lights[2], lights[3]);
    turn_red(lights[2], lights[3]);
}

void turn_green(int lights[3], int pedlights[3]){
    digitalWrite(lights[1], HIGH);
    delay(delays[1]);
    if(digitalRead(pedlights[1]) == HIGH){
        digitalWrite(pedlights[0], LOW);
        digitalWrite(pedlights[1], LOW);
        digitalWrite(pedlights[2], HIGH);
    }
    digitalWrite(lights[0], LOW);
    digitalWrite(lights[1], LOW);
    digitalWrite(lights[2], HIGH);
    delay(delays[0]);
}

void turn_red(int lights[3], int pedlights[3]){
    digitalWrite(lights[2], LOW);
```

```

digitalWrite(pedlights[2], LOW);
digitalWrite(pedlights[0], HIGH);
digitalWrite(lights[1], HIGH);
delay(delays[1]);
digitalWrite(lights[1], LOW);
digitalWrite(lights[0], HIGH);
digitalWrite(lights[2], LOW);
}

void left_btn_pressed(){
    digitalWrite(1, HIGH);
}

void right_btn_pressed(){
    digitalWrite(10, HIGH);
}

```

Øvelse 2. Robot

```

#include <ArduinoRobot.h>
#include <Wire.h>

long timerOrigin;
void setup() {
    // initialize the Robot, SD card, display, and speaker
    Robot.begin();
    Robot.beginTFT();

    // show the logots on the TFT screen

    Robot.fill(0, 0, 0);
    Robot.stroke(0,255, 0);
    // add the instructions
    Robot.text("Line Following\n\n place the robot on\n the track and \n see it run", 5, 5);
    Robot.text("Press the middle\n button to start...", 5, 61);
    Robot.waitContinue();

    Robot.lineFollowConfig(11, 7, 60, 5);

    Robot.setMode(MODE_LINE_FOLLOW);

```

```
Robot.fill(255, 255, 255);
Robot.stroke(255, 255, 255);
Robot.rect(0, 0, 128, 80);
Robot.stroke(0, 0, 0);
Robot.text("Start", 5, 5);

Robot.stroke(0, 0, 0);
Robot.text("Time passed:", 5, 21);

timerOrigin = millis(); // keep track of the elapsed time

while (!Robot.isActionDone()) {
    Robot.debugPrint(millis() - timerOrigin, 5, 29);
}

Robot.stroke(0, 0, 0);
Robot.text("Done!", 5, 45);
}

void loop() {
```

Øvelser i “Arduino Project Book”

Project 02

```
int switchState = 0;
void setup() {
pinMode (3, OUTPUT);
pinMode (4, OUTPUT);
pinMode (5, OUTPUT);
pinMode (2, INPUT);
}

void loop() {
switchState = digitalRead(2);

if(switchState == LOW){
    digitalWrite(3, HIGH);
    digitalWrite(4,LOW);
    digitalWrite(5,LOW);
}

else
{
    digitalWrite(3, LOW);
    digitalWrite(4,LOW);
    digitalWrite(5,HIGH);

    delay(250);
digitalWrite(4,HIGH);
digitalWrite(5,LOW);
delay(250);
}
}
```

Project 03

```
const int sensorPin = A0;
const float baselineTemp = 20.0;
void setup() {
    Serial.begin(9600);
    for(int pinNumber = 2; pinNumber<5; pinNumber++){
        pinMode(pinNumber, OUTPUT);
        digitalWrite(pinNumber, LOW);
    }
}
void loop() {
    int sensorVal = analogRead(sensorPin);
    Serial.print("Sensor Value: ");
    Serial.print(sensorVal);
    float voltage = (sensorVal/1024.0) * 5.0;
    Serial.print(", Volts: ");
    Serial.print(voltage);
    Serial.print(", degrees C: ");
    float temperature = (voltage - .5) * 100;
    Serial.println(temperature);
    if(temperature < baselineTemp + 2){
        digitalWrite(2, LOW);
        digitalWrite(3, LOW);
        digitalWrite(4, LOW);
    }
    else if (temperature >= baselineTemp+2 && temperature < baselineTemp+ 4){
        digitalWrite(2, HIGH);
        digitalWrite(3, LOW);
        digitalWrite(4, LOW);
    }
    else if (temperature >= baselineTemp+4&& temperature < baselineTemp+ 6){
        digitalWrite(2, HIGH);
        digitalWrite(3, HIGH);
        digitalWrite(4, LOW);
    }
    else if (temperature >= baselineTemp+6){
        digitalWrite(2, HIGH);
        digitalWrite(3, HIGH);
        digitalWrite(4, HIGH);
    }
    delay(1000);
}
```

Project 04

```
const int greenLEDPin = 9;
const int redLEDPin = 10;
const int blueLEDPin = 11;
const int redSensorPin = A0;
const int greenSensorPin = A1;
const int blueSensorPin = A2;
int redValue=0;
int greenValue = 0;
int blueValue = 0;
int redSensorValue = 0;
int greenSensorValue = 0;
int blueSensorValue = 0;
void setup() {
  Serial.begin(9600);
  pinMode(greenLEDPin, OUTPUT);
  pinMode(redLEDPin, OUTPUT);
  pinMode(blueLEDPin, OUTPUT);
}
void loop() {
  redSensorValue = analogRead(redSensorPin);
  delay(5);
  greenSensorValue = analogRead(greenSensorPin);
  delay(5);
  blueSensorValue = analogRead(blueSensorPin);
  Serial.print("Raw Sensor Values \t red: ");
  Serial.print(redSensorValue);
  Serial.print("\t green: ");
  Serial.print(greenSensorValue);
  Serial.print("\t blue: ");
  Serial.println(blueSensorValue);
  redValue = redSensorValue/4;
  greenValue = greenSensorValue/4;
  blueValue = blueSensorValue/4;
  Serial.print("Mapped Sensor Values \t ");
  Serial.print(redValue);
  Serial.print("\t green: ");
  Serial.print(greenValue);
  Serial.print("\t blue: ");
  Serial.println(blueValue);
  analogWrite(redLEDPin, redValue);
  analogWrite(greenLEDPin, greenValue);
  analogWrite(blueLEDPin, blueValue);
}
```

Project 05

```
#include <Servo.h>

Servo myServo;

int const potPin = A0;
int potVal;
int angle;

void setup() {
myServo.attach(9);

Serial.begin(9600);

}

void loop() {
potVal = analogRead(potPin);
Serial.print("potVal: ");
Serial.print(potVal);
angle = map(potVal, 0, 1023, 0, 179);
Serial.print(", angle: ");
Serial.println(angle);
myServo.write(angle);
delay(15);
}
```

Project 06

```
int sensorValue;
int sensorLow = 1023;
int sensorHigh = 0;

const int ledPin = 13;
void setup() {
pinMode(ledPin, OUTPUT);
digitalWrite(ledPin, HIGH);

while(millis()< 5000){
    sensorValue = analogRead(A0);
    if(sensorValue < sensorHigh){
        sensorHigh = sensorValue;
    }
    if(sensorValue< sensorLow){
        sensorLow = sensorValue;

    }
}

digitalWrite(ledPin, LOW);
}

void loop() {
sensorValue = analogRead(A0);
int pitch = map(sensorValue,sensorLow, sensorHigh,50,4000);
tone(8,pitch,20);
delay(10);
}
```

Project 07

```
int buttons[6];
int notes[] = {262,294,330,349};
void setup() {
    buttons[0] = 2;
    Serial.begin(9600);
}

void loop() {
    int keyVal = analogRead(A0);
    Serial.println(keyVal);
    if(keyVal == 1023){
        tone(8, notes[0]);
    }
    else if(keyVal >= 990 && keyVal <= 1010){
        tone(8, notes[1]);
    }
    else if (keyVal >= 505 && keyVal <= 515){
        tone(8, notes[2]);
    }
    else if (keyVal >= 5 && keyVal <= 10){
        tone(8, notes[3]);
    }
    else{
        noTone(8);
    }
}
```

Project 08

```
const int switchPin = 8;

unsigned long previousTime = 0;
int switchState= 0;
int prevSwitchState = 0;

int led = 2;

long interval = 600000;

void setup() {
for(int x = 2; x<8; x++){
    pinMode(x, OUTPUT);
}

pinMode(switchPin, INPUT);
}

void loop() {
unsigned long currentTime = millis();
if(currentTime - previousTime > interval) {
    previousTime = currentTime;
    digitalWrite(led, HIGH);
    led++;
}
if(led == 7){

}
switchState = digitalRead(switchPin);
if(switchState != prevSwitchState){
    for(int x = 2; x<8;x++){
        digitalWrite(x,LOW);
    }
    led = 2;
    previousTime = currentTime;
}
prevSwitchState = switchState;
}
```

Project 11

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

const int switchPin = 6;
int switchState = 0;
int prevSwitchState = 0;
int reply;

void setup() {
  lcd.begin(16,2);
  pinMode(switchPin, INPUT);
  lcd.print("Ask the");
  lcd.setCursor(0,1);
  lcd.print("Crystal Ball!");
}

void loop() {
  switchState = digitalRead(switchPin);

  if(switchState != prevSwitchState){
    if(switchState == LOW){
      reply = random(8);
      lcd.clear();
      lcd.setCursor(0,0);
      lcd.print("The ball says:");
      lcd.setCursor(0,1);

      switch(reply){
        case 0:
          lcd.print("Yes");
          break;
        case 1:
          lcd.print("Most Likely");
          break;
        case 2:
          lcd.print("Certainly");
          break;
        case 3:
          lcd.print("Outlook good");
          break;
        case 4:
```

```
lcd.print("Unsure");
break;
case 5:
lcd.print("Ask again");
break;
case 6:
lcd.print("Doubtful");
break;
case 7:
lcd.print("No");
break;
}
}
}
prevSwitchState = switchState;
}
```

Project 12

```
#include <Servo.h>
Servo myServo;
const int piezo = A0;
const int switchPin = 2;
const int yellowLed = 3;
const int greenLed = 4;
const int redLed = 5;
int knockVal;
int switchVal;
const int quietKnock = 10;
const int loudKnock = 100;
boolean locked = false;
int number_of_Knocks = 0;

void setup(){
myServo.attach(9);
pinMode(yellowLed, OUTPUT);
pinMode(redLed, OUTPUT);
pinMode(greenLed, OUTPUT);
pinMode(switchPin, INPUT);
Serial.begin(9600);
digitalWrite(greenLed, HIGH);
myServo.write(0);
Serial.println("The box is unlocked!");
}

void loop(){
if(locked == false){
switchVal = digitalRead(switchPin);
if(switchVal == HIGH){
locked = true;
digitalWrite(greenLed,LOW);
digitalWrite(redLed,HIGH);
myServo.write(90);
Serial.println("The box is locked!");
delay (1000);
}
}
if(locked == true){
knockVal = analogRead(piezo);
if(number_of_Knocks < 3 && knockVal > 0){
if(checkForKnock(knockVal) == true){
number_of_Knocks++;
}
}
}
}
```

```
}

Serial.print(3-numberOfKnocks);
Serial.println(" more knocks to go");
}

if(numberOfKnocks >= 3){
locked = false;
myServo.write(0);
delay(20);
digitalWrite(greenLed,HIGH);
digitalWrite(redLed,LOW);
Serial.println("The box is unlocked!");
}
}
}

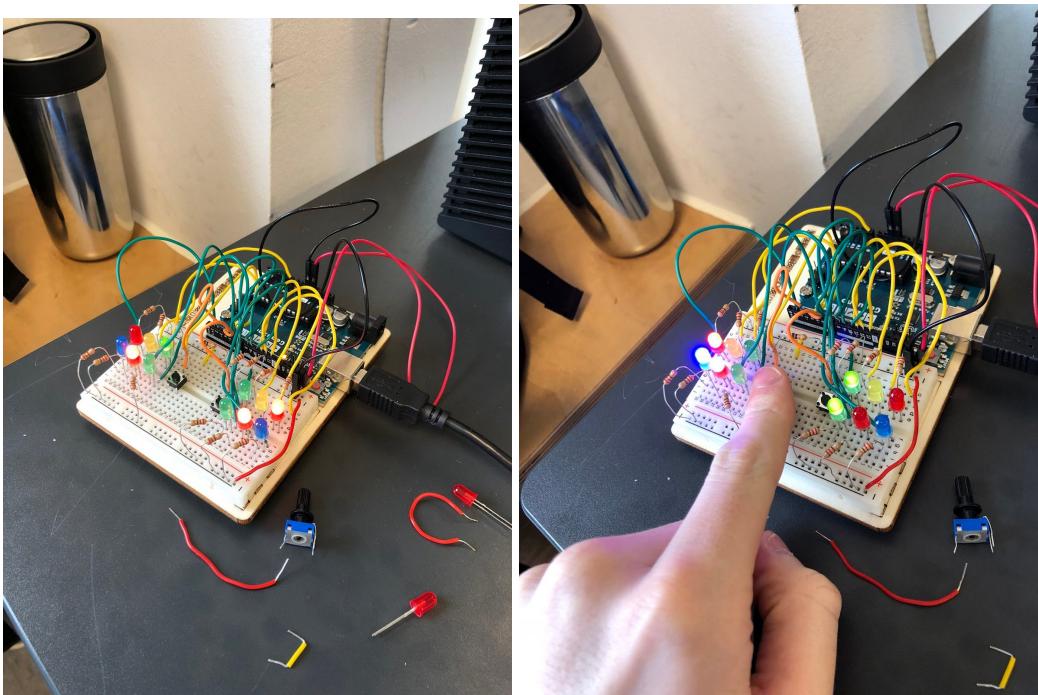
boolean checkForKnock(int value){
if(value > quietKnock && value < loudKnock){

digitalWrite(yellowLed, HIGH);
delay(50);
digitalWrite(yellowLed, LOW);
Serial.print("Valid knock of value ");
Serial.println(value);
return true;
}

else {
Serial.print("Bad knock value ");
Serial.println(value);
return false;
}
}
```

Billeder:

Øvelse 1. Trafiklys:

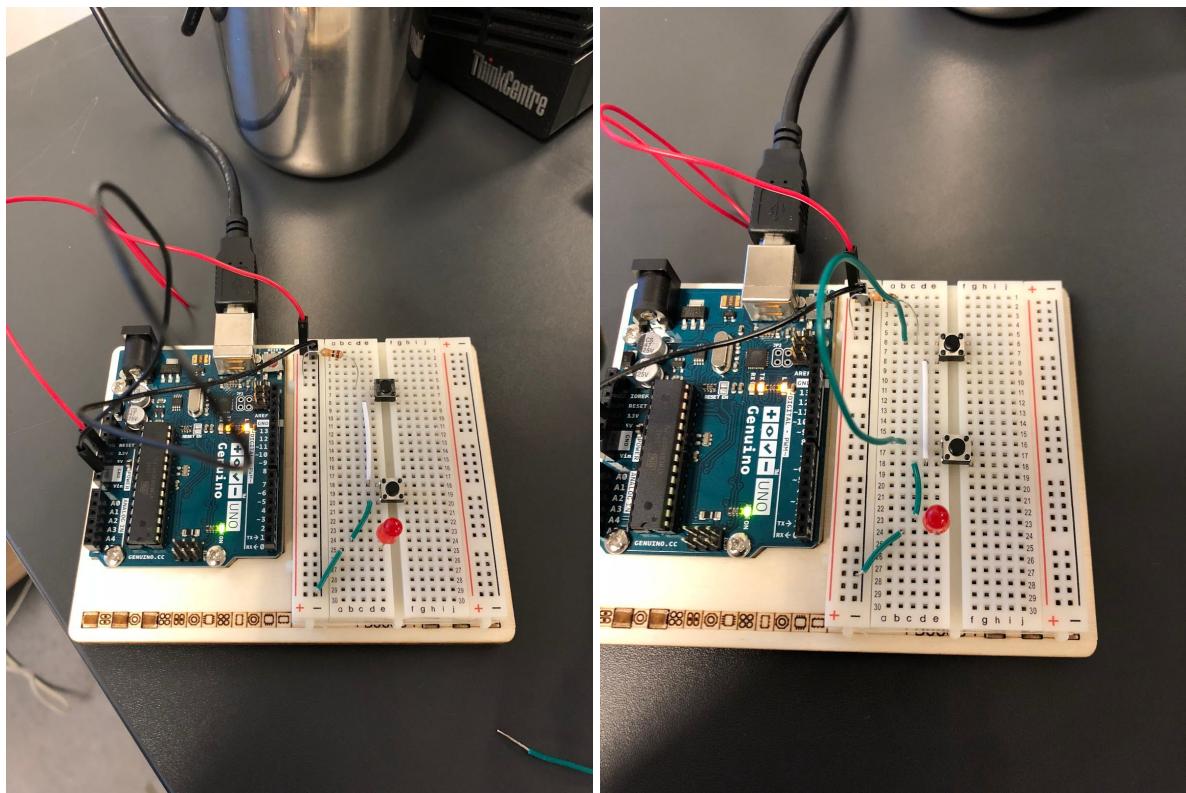


Øvelse 2. Robot

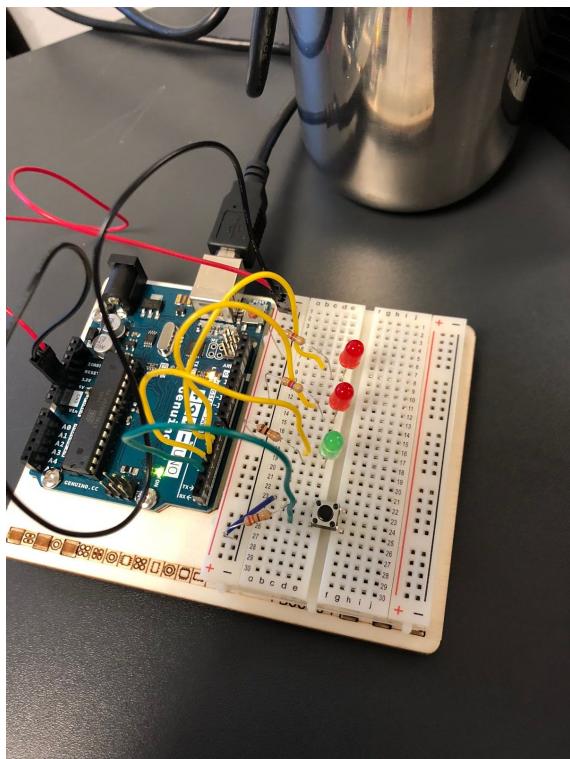


Øvelser i “Arduino Project Book”

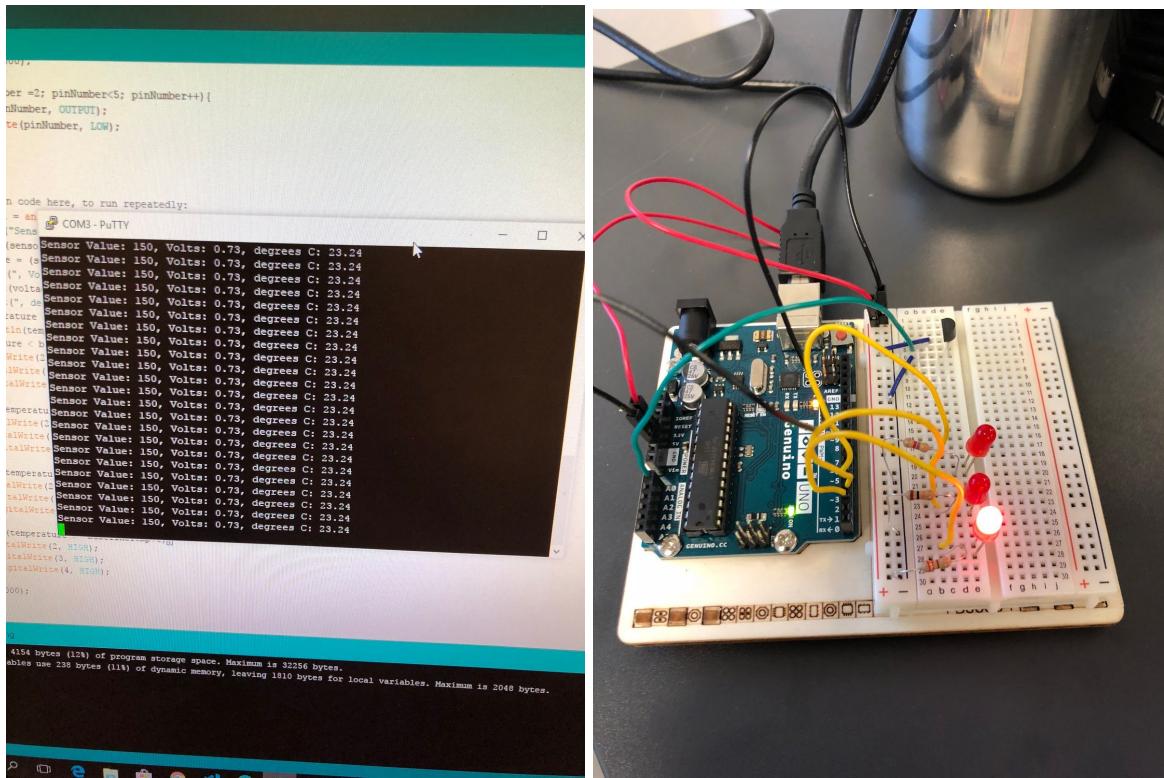
Project 01



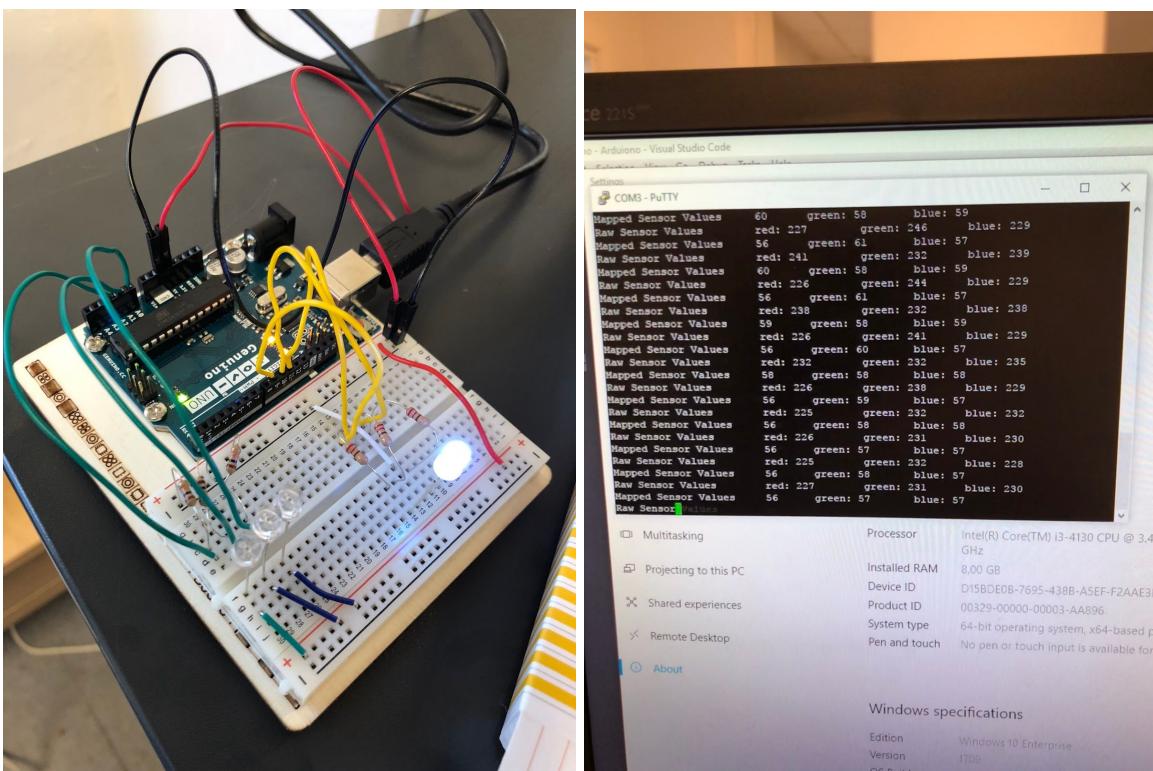
Project 02



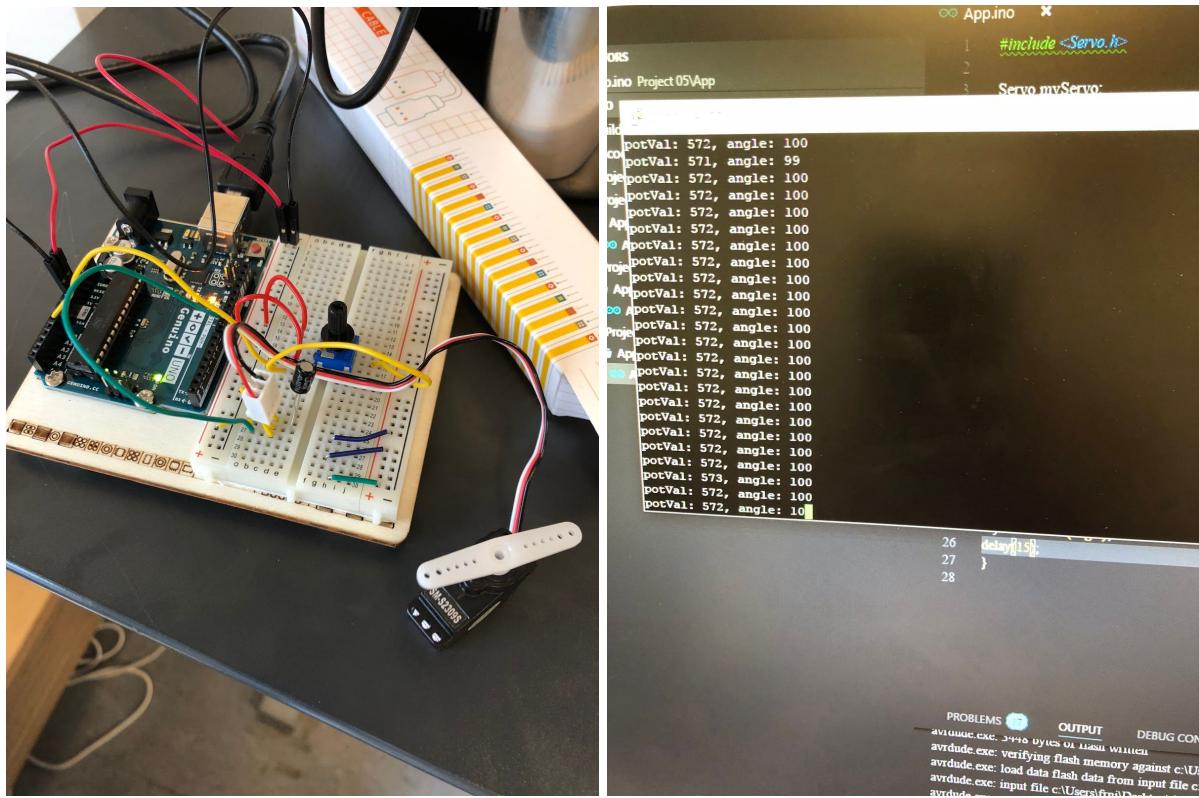
Project 03



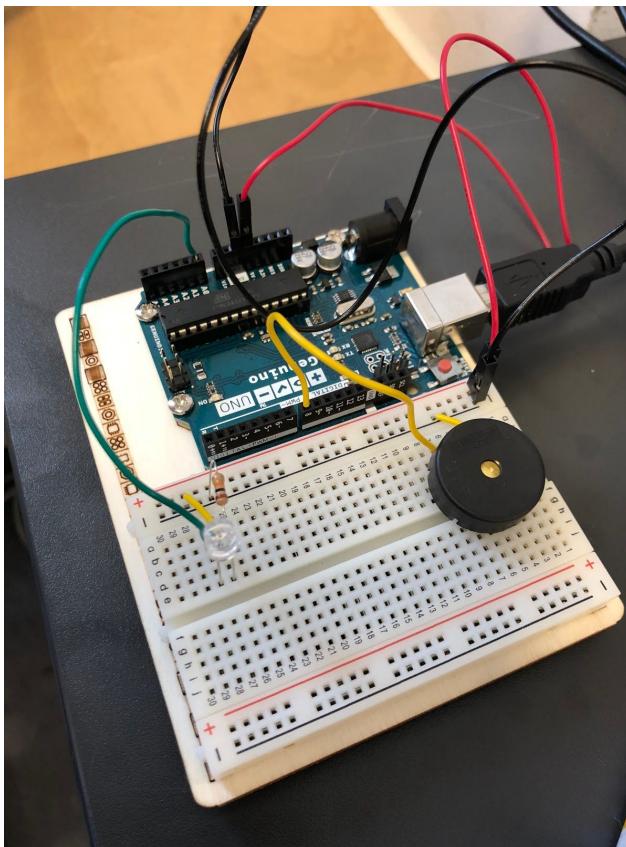
Project 04



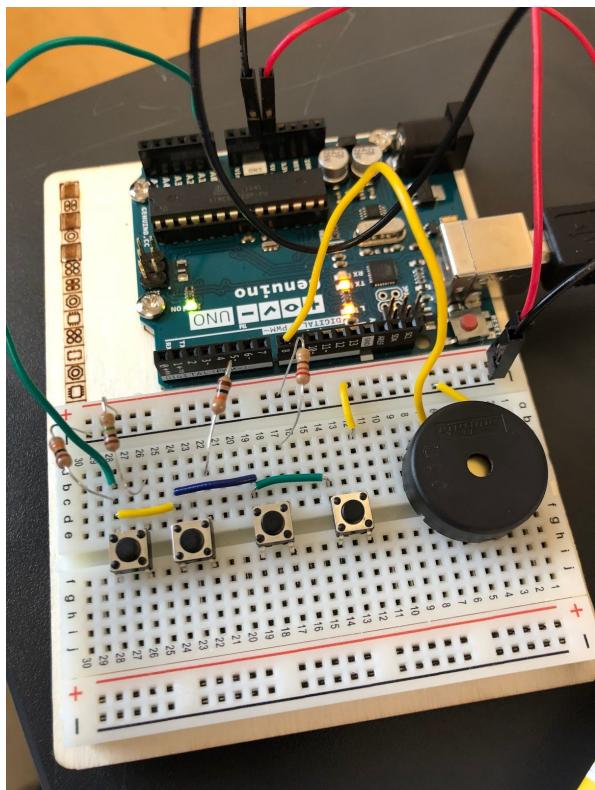
Project 05



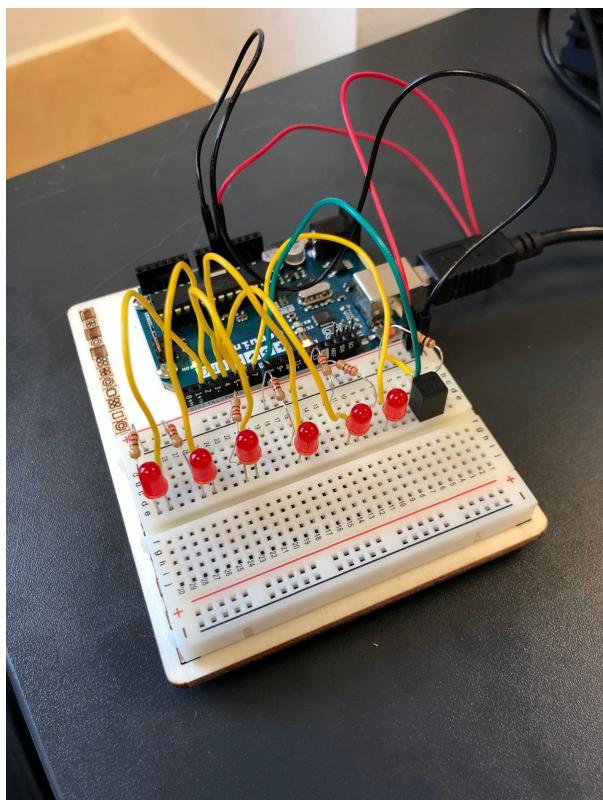
Project 06



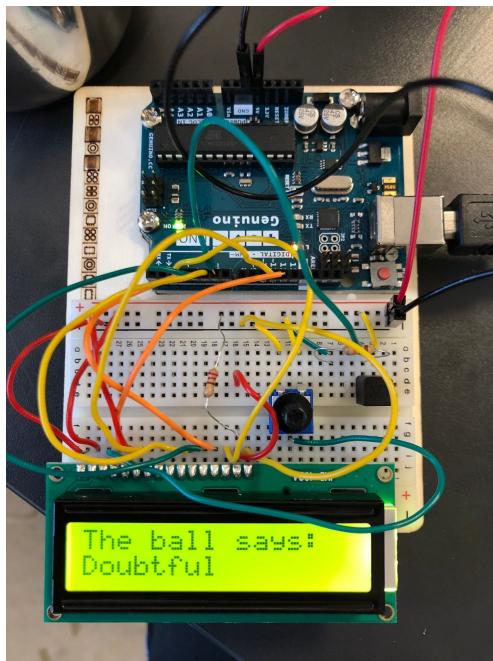
Project 07



Project 08



Project 11



Project 12

