

Opgavesæt - Arduino

Arduino starter opgaver:

Opg. 1

Lav et program, der får LED L til at tænde.

Hints

- Brug den indbyggede LED: LED_BUILTIN.
- I setup() sættes pinnen som OUTPUT.
- Tænd LED'en med digitalWrite(..., HIGH).

Opg. 2

Udvid programmet, så LED L først tænder og derefter slukker igen.

Hints

- Tænd → kort pause → sluk → kort pause.
- Brug delay(...) til pauser.
- Husk at loop() kører i ring.

Opg. 3

Lav et program, hvor LED L blinker med 100 millisekunders interval i al uendelighed.

Hints

- Ét blink = tænd → delay(100) → sluk → delay(100).
- Koden ligger i loop() for gentagelse.

Opg. 4

Lav et program, hvor LED L blinker med 100 millisekunders interval, men kun 10 gange.

Hints

Aalborghus Sensor Besøg

- Brug en tæller (fx for-løkke).
- Efter 10 blink: stop (enten tom `loop()` eller “parkér” programmet).
- Alternativt: tæller i `loop()` og stop, når den rammer 10.

Opg. 5

Lav et program, hvor LED L blinker. Tilføj derefter en funktion, så LED L kan stoppes og startes igen, når du trykker på en knap på computeren (via serielmonitoren).

Hints

- Start seriel: `Serial.begin(9600)` og åbn Serielmonitor (samme baud).
- Tjek input med `Serial.available()` og læs tegn med `Serial.read()`.
- Brug en bool variabel (fx `koerer`) som “start/stop”-tilstand.
- Vælg et kommando-tegn (fx `t`) til at toggle tilstanden.
- Vær opmærksom på Serielmonitorens line ending-indstilling.

Opg. 6

Udvid programmet fra Opg. 5: Når LED L stopper og starter med at blinke, skal programmet samtidig printe “stop” og “start” i serielmonitoren.

Hints

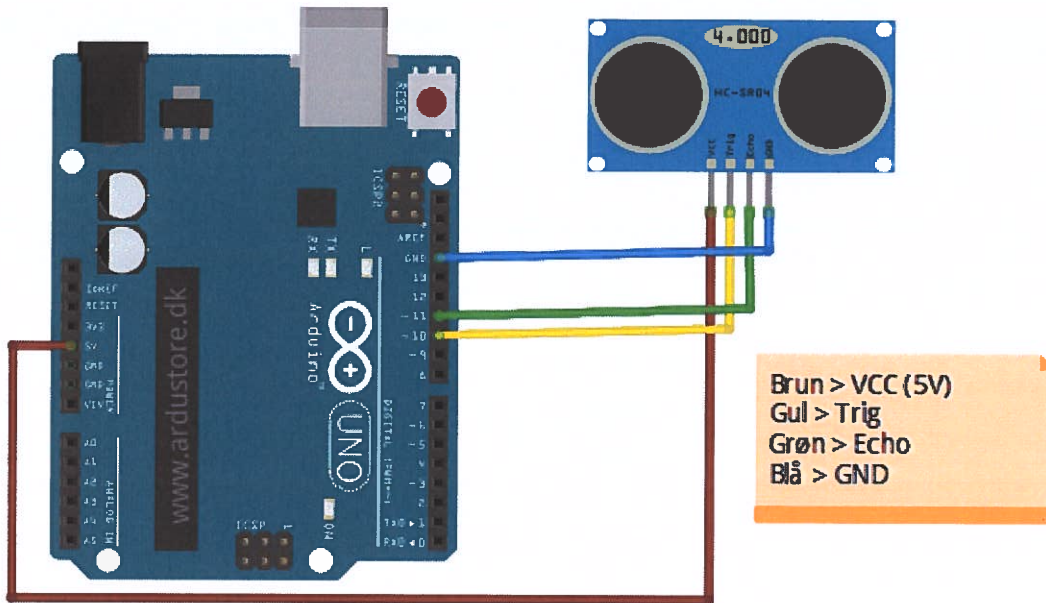
- Når du skifter `koerer`, så print “start” eller “stop” med `Serial.println(...)`.
- Udskriv en kort hjælpetekst i `setup()` om, hvad man skal trykke på.

Opgave Ultralydssensor

I denne øvelse skal I bruge en ultralydssensor (HC-SR04) til at måle afstanden til en genstand. Resultatet skal vises i **Serial Monitor** på computeren.

Udstyr:	Kobling af sensoren:
<ul style="list-style-type: none">• Arduino UNO• Computer• Ultralydssensor• Breadboard og jumperkabler	<ul style="list-style-type: none">• VCC → 5V• GND → GND• Trig → Pin 10• Echo → Pin 11

Aalborghus Sensor Besøg

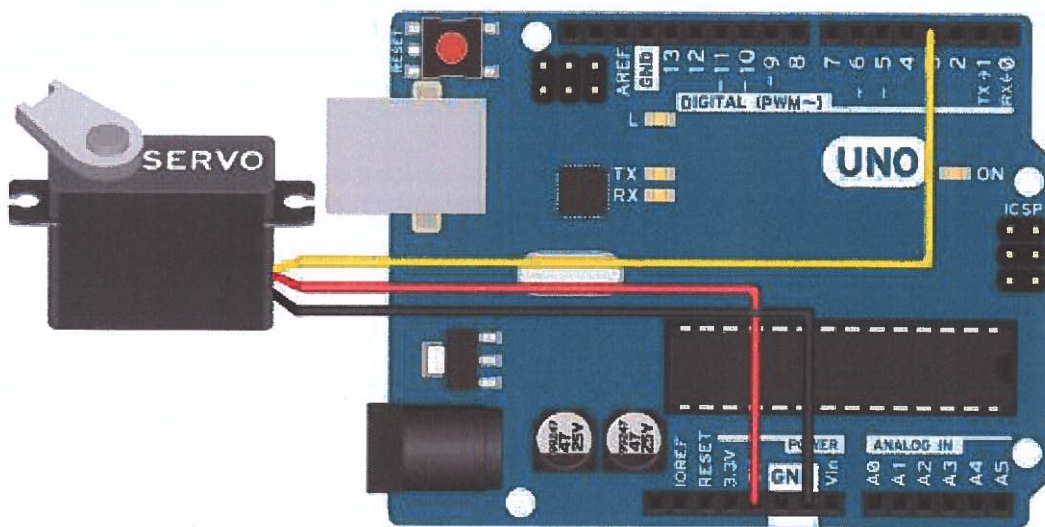


Skeletkode:

```
1 // Definér pins
2 const int trigPin = 10;
3 const int echoPin = 11;
4
5 void setup() {
6   Serial.begin(9600);
7   pinMode(trigPin, OUTPUT);
8   pinMode(echoPin, INPUT);
9 }
10
11 void loop() {
12   // 1. Send en kort puls til trigPin
13   // TODO: Brug digitalWrite() og delayMicroseconds() her
14   // 2. Læs hvor længe echoPin er HIGH
15   // TODO: Brug pulseIn() og gem resultatet i en variabel "duration"
16   // 3. Regn tiden om til afstand
17   // TODO: Lav udregningen og gem i en variabel "distance"
18   // 4. Print resultatet i Serial Monitor
19   // 5. Vent lidt før næste måling (0.5 sec)
20 }
```

Opgave Servomotor:

I denne øvelse skal I bruge en servomotor til at dreje rundt. Servomotoren skal kobles til således:



Servomotor

Servomotoren kan dreje til en bestemt vinkel mellem 0 og 180 grader.

Man styrer vinklen med kommandoen:

```
Servo1.write(vinkel);
```

Eksempel:

```
Servo1.write(90);
```

drejer servoen til 90 grader.

Tilslutning

- Rød ledning → 5V
- Sort/brun ledning → GND
- Gul/orange ledning (signal) → pin 3

Opgave 1

Få motoren til at:

- dreje til 0 grader
- vente 1 sekund
- dreje til 90 grader
- vente 2 sekunder

Aalborgarhus Sensor Besøg

- dreje til 180 grader
- dreje tilbage til 0 grader

Tips

- Inkluder biblioteket med `#include <Servo.h>`
- Lav et servo-objekt: `Servo Servo1;`
- Angiv hvilken pin servoen er koblet til: `int servoPin = 3;`
- I `setup()` kobles servoen til pinnen med `Servo1.attach(servoPin);`
- Brug `delay(...)` til at vente. Tallet angives i millisekunder

Opgave 2

Få motoren til at bevæge sig jævnt fra 0 til 180 grader og tilbage igen.

Tips

- Brug en for-løkke
- Lad fx en variabel med navnet `position` starte ved 0 og tælle op til 180
- Flyt servoen én grad ad gangen med `Servo1.write(position);`
- Brug en lille `delay(...)` mellem hvert trin

Opgave 3

Prøv selv:

Få servoen til at vælge en tilfældig vinkel mellem 0 og 180 grader, vente et øjeblik og derefter vælge en ny tilfældig vinkel.

Tips

- Du kan bruge `random(0, 181)` til at lave en tilfældig vinkel.
- Husk at bruge `randomSeed(analogRead(A0))` for mere tilfældige tal.
- Gem vinklen i en variabel og send den til servoen med `Servo1.write(...)`