

📌 Lag ditt eget Arduino-prosjekt

LØSNINGSFORSLAG

8.-10. trinn · 60-120 min · SkapLab – Arduino

Hva du skal gjøre

Kombiner sensorer, aktuatorer og logikk fra de forrige oppgavene til et eget produkt. Planlegg grundig på papir før du begynner å kode.

Planlegg først

- Hva skal produktet gjøre? Skriv én setning
- Hvilke innganger trenger du? (sensorer, knapper, potensiometer ...)
- Hvilke utganger trenger du? (LED, servo, LCD, summer ...)
- Tegn koblingsskjema på papir og skriv pseudokode
- Del koden inn i funksjoner fra starten – lettere å feilsøke

Minimumskrav

- | |
|--|
| 1. Minst 2 ulike komponenter koblet til (sensor + aktuator). |
| 2. Koden bruker minst én funksjon du har skrevet selv. |
| 3. Seriell monitor viser nyttig debuginfo underveis. |
| 4. Kommentarer forklarer hva koden gjør. |
| 5. Prosjektet starter i en kjent tilstand når det skrur på. |

Prosjektideer

- Sikkerhetssystem: ultralydsensor oppdager inntrenger, sirene og LCD viser 'ALARM!'
- Værstasjon: temperatur og lysnivå på LCD, logger til seriell port
- Automatisk gardin: servo åpner/lukker etter lysnivå fra LDR-sensor
- Reaksjonsspill: LED blinker, trykk knapp raskt mulig – vis responstid på LCD
- Piano: 8 knapper spiller ulike toner via summer, LCD viser note-navn

Løsningsforslag - reaksjonsspill

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
```

```

LiquidCrystal_I2C lcd(0x27, 16, 2);
const int LED = 13, KNAPP = 2;

void setup() {
  pinMode(LED, OUTPUT);
  pinMode(KNAPP, INPUT_PULLUP);
  lcd.init(); lcd.backlight();
  randomSeed(analogRead(A5));
  Serial.begin(9600);
}

void loop() {
  lcd.clear(); lcd.print("Klar? Vent...");
  delay(random(2000, 5000));

  if (digitalRead(KNAPP) == LOW) {
    lcd.clear(); lcd.print("For tidlig! :P");
    delay(2000); return;
  }

  digitalWrite(LED, HIGH);
  lcd.clear(); lcd.print("NÅ! Trykk!");
  long start = millis();
  while (digitalRead(KNAPP) == HIGH) {}
  long tid = millis() - start;
  digitalWrite(LED, LOW);

  lcd.clear();
  lcd.print("Tid: "); lcd.print(tid); lcd.print(" ms");
  lcd.setCursor(0, 1);
  if (tid < 200)      lcd.print("Lynrask!");
  else if (tid < 350) lcd.print("Bra!");
  else               lcd.print("Øv mer...");

  Serial.println(tid);
  delay(3000);
}

```

millis() returnerer millisekunder siden Arduino startet. Differansen mellom to kall gir nøyaktig responstid – mye bedre enn delay()-basert timing.