

# ● Lærerveiledning - Buzzer

*Skrevet av: Adrian Helle og Susanne Rynning Seip*

*Kurs: Arduino*

*Tema: Elektronikk, Tekstbasert*

*Fag: Musikk, Naturfag, Teknologi*

*Klassetrinn: 5.-7. klasse, 8.-10. klasse, Videregående skole*

## Om oppgaven

I denne oppgaven skal elevene programmere en Arduino til å spille av en lyd med en buzzer.

## ✓ Oppgaven passer til:

**Fag:** Musikk, Naturfag, Teknologi og Design, Teknologi og forskningslære

**Anbefalte trinn:** 5. trinn - VG3

**Tema:** Arduino, Kretser, Buzzer, Lyd

**Tidsbruk:** Dobbelttime

## Kompetansemål

- ☐ **Naturfag, 7. trinn:** utforske, lage og programmere teknologiske systemer som består av deler som virker sammen
- ☐ **Musikk, 10. trinn:** skape og programmere musikalske forløp ved å eksperimentere med lyd fra ulike kilder
- ☐ **Teknologi og design, 10. trinn:** velge og bruke ulike materialer og verktøy i arbeidet med prototyper for teknologiske produkt
- ☐ **Teknologi og design, 10. trinn:** grunngi valg av design, materialer, verktøy og framgangsmåter, og reflektere over hvor trygge og bærekraftige valgene er

- ☐ **Teknologi og forskningslære X og 1, VG2:** anvende kravspesifikasjon i arbeid med å utvikle og teste funksjonelle produkter og vurdere produktet opp mot kravspesifikasjonen
- ☐ **Teknologi og forskningslære 1, VG2:** bruke og programmere mikrokontroller for å utvikle et produkt

## Forslag til læringsmål

- ☐ Elevene kan bruke en buzzer for å spille lyd.
- ☐ Elevene kan bruke en lysfølsom motstand.
- ☐ Elevene kan skrive program til Arduino som utnytter `tone` , `analogRead` og `map` .

## Forslag til vurderingskriterier

Oppgaven er grunnleggende, og kan ikke brukes alene for vurdering av kompetansemålet.

## Forutsetninger og utstyr

- ☐ **Forutsetninger:** Oppgaven er en introduksjon til Arduino, og krever ingen forkunnskaper eller erfaring. Det er en fordel å kunne litt engelsk.
- ☐ **Utstyr:**
  - ☐ 1 Arduino Uno

- ☐ Datamaskin med Arduino software installert (se *Eksterne ressurser*) og USB-kabel
- ☐ 1 breadboard
- ☐ 2 ledninger
- ☐ 1 buzzer
- ☐ 1 motstand 270 Ohm (rød-lilla-brun-gull)



## Fremgangsmåte

Her kommer tips, erfaring og utfordringer til de ulike stegene i den faktiske oppgaven. Klikk her for å se oppgaveteksten. ([../buzzer/buzzer.html](#))

## Lysfølsom motstand

En lysfølsom motstand endrer motstanden sin basert på lyset den får inn. Jo høyere lysstyrke den leser, jo lavere motstand yter den. Når vi bruker `INPUT_PULLUP` setter vi en positiv spenning ut fra pinnen. Jo høyere motstand i kretsen, jo høyere vil tallet vi leser ut være.

Dvs. at jo mer lys det er i rommet, jo lavere tall leser vi ut.

## PWM

PWM står for "Pulse With Modulation". PWM gjør at vi kan sende et "analogt" signal på en digital pin. På en ikke-PWM pin vil en sende enten HIGH eller LOW kontinuerlig, som vil si å enten sende 5v eller 0v, til en velger å sende det motsatte. Det PWM gjør er å sende 5v i deler av en periode, for så å slå av strømmen resten av perioden. Jo høyere verdi en setter utgangen til, jo større del av perioden vil vi sende 5v.

Du kan lese mer om PWM på Arduino sine sider (<https://www.arduino.cc/en/Tutorial/PWM>).

## analogWrite

`analogWrite` bruker PWM for å kunne sende "analoge" signaler over en pin. Dette lar oss blant annet styre lysstyrken på en lysdiode, hastigheten på en motor, eller fargeintensiteten på en RGB diode.

En verdi på 0 vil sende 0v ut, en verdi på 255 vil sende 5v hele perioden.

## analogRead

`analogRead` er en digital utlesning av den analoge spenningen på en pinne.

## tone

`tone` spiller av en frekvens en PWM kapabel pin. Du kan få den til å spille av en frekvens kontinuerlig med `tone(pin, frekvens)`, som så kan stoppes med `noTone(pin)`. Du kan også generere en tone for en gitt tid med `tone(pin, frekvens, millisekunder)`, som vil sende en tone i gitt antall millisekunder.

## Variasjoner



*Vi har dessverre ikke noen variasjoner tilknyttet denne oppgaven enda.*

## Eksterne ressurser

- ☐ Installasjonsinstruksjoner:
  - ☐ Windows (<https://arduino.cc/en/Guide/Windows>)
  - ☐ Mac OS X (<https://arduino.cc/en/Guide/MacOSX>)
  - ☐ Linux (<https://arduino.cc/en/Guide/Linux>)

Lisens: CC BY-SA 4.0 (<http://creativecommons.org/licenses/by-sa/4.0/deed>)