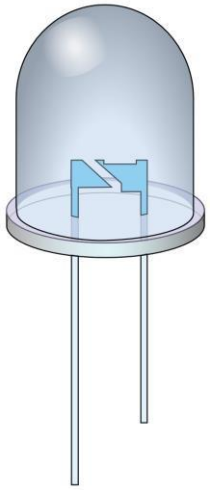
# Arduino: Hvordan?

## Kretser og strøm



# Vår første elektriske krets: Blinke ekstern LED






**Vi trenger:**

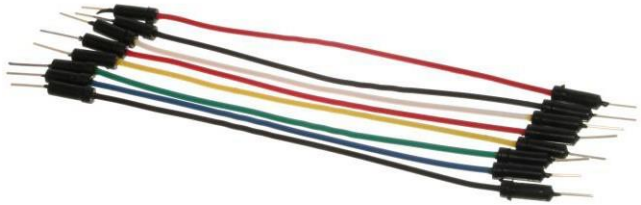
Breadboard LED

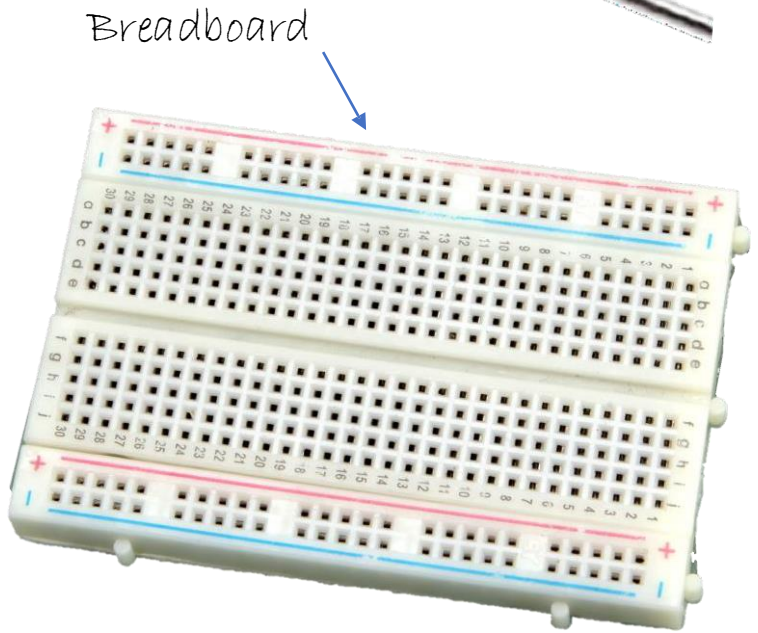
Resistor 220 Ohm

1 svart ledning

1 rød ledning

Skisse: modifisere Blink-skissen





# Bygge kretsen

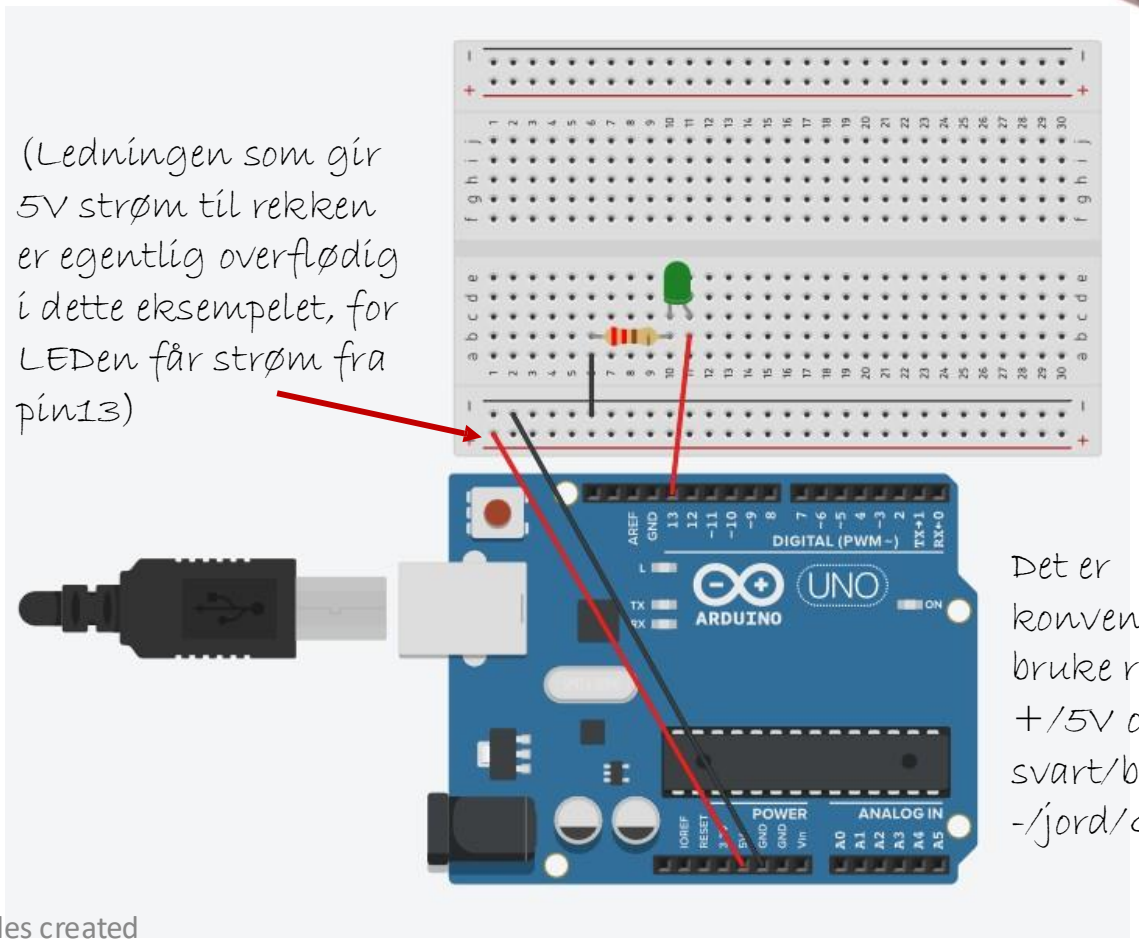
Start med
1: å sjekke at strømmen er koblet fra
2: rød ledning fra 5V til rød rekke på breadboardet
svart ledning fra GND til svart(blå) rekke

Rød ledning fra pin 13 på Arduinoen til breadboardet

Det lengste benet (+) til LED-en i samme rad som den røde ledningen (bøy det lengste benet litt)

220 ohms resistor fra det korteste benet (-) på LEDen (Resistoren kan settes begge veier)

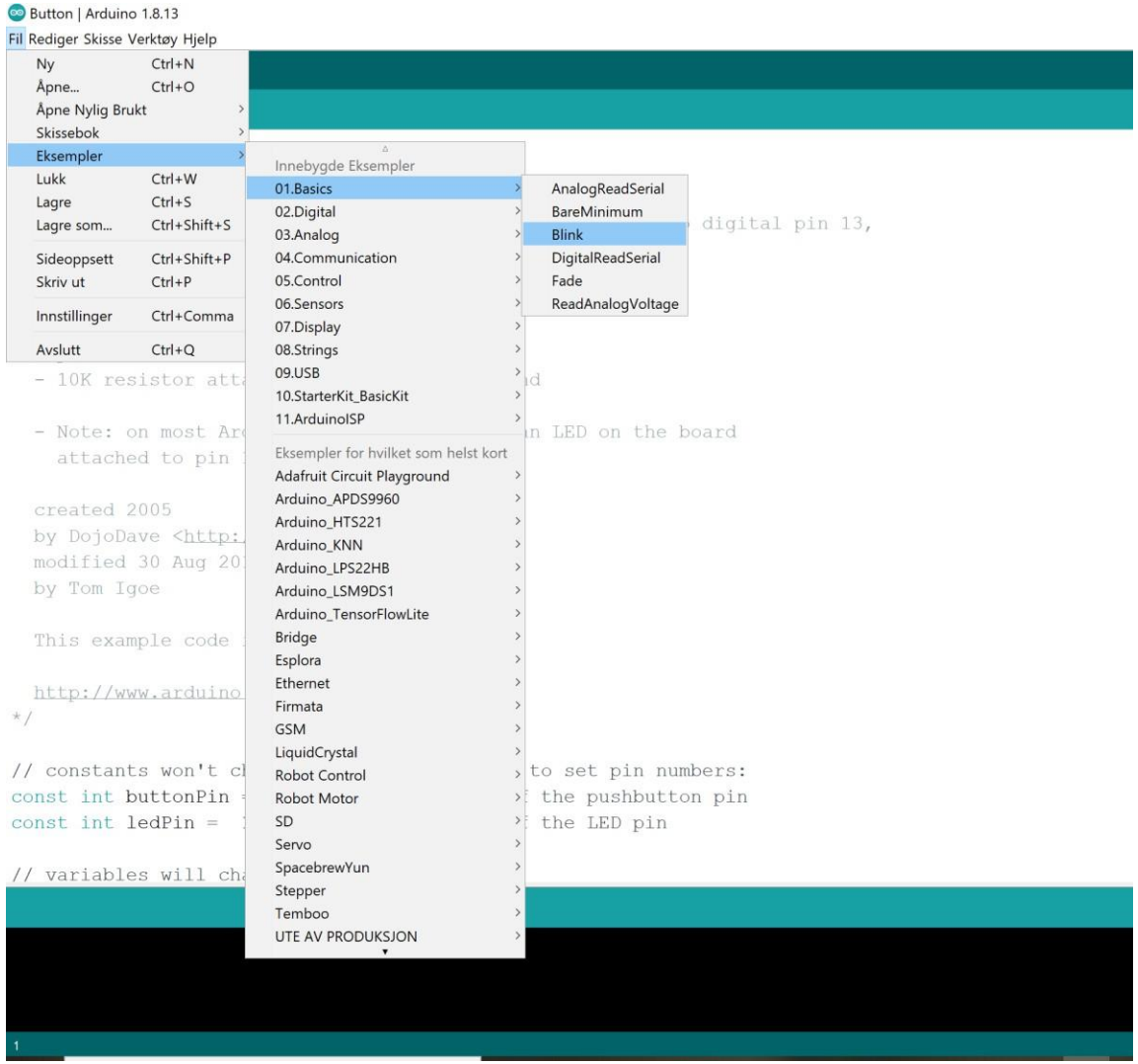
Svart ledning fra resistoren til GND på Arduinoen



# Åpne Arduino IDE

I Arduino IDE på PCen, åpne
Fil – Eksempler – 01 Basics – Blink

(samme som i forrige time)



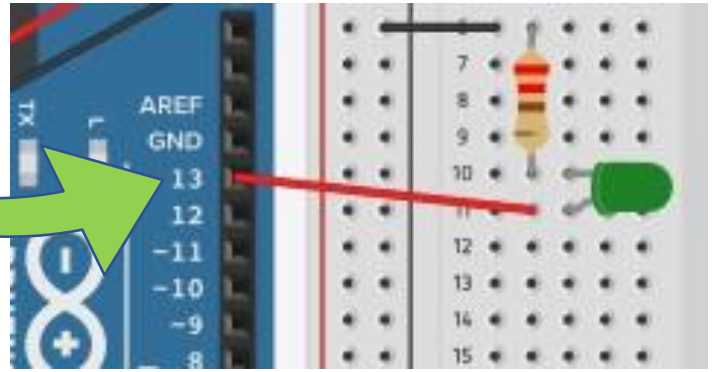
# Blink-skisse til egendefinert pin

```
void setup() {
int ledPin = 13; //Når vi ikke bruker den innebygde LED-en må vi initialisere (lage en variabel og gi den en verdi) en variabel for pin-en vi velger
pinMode(ledPin, OUTPUT); //vi bruker pinMode til å bestemme om det er signal ut eller inn på pin-en, som tidligere
•}

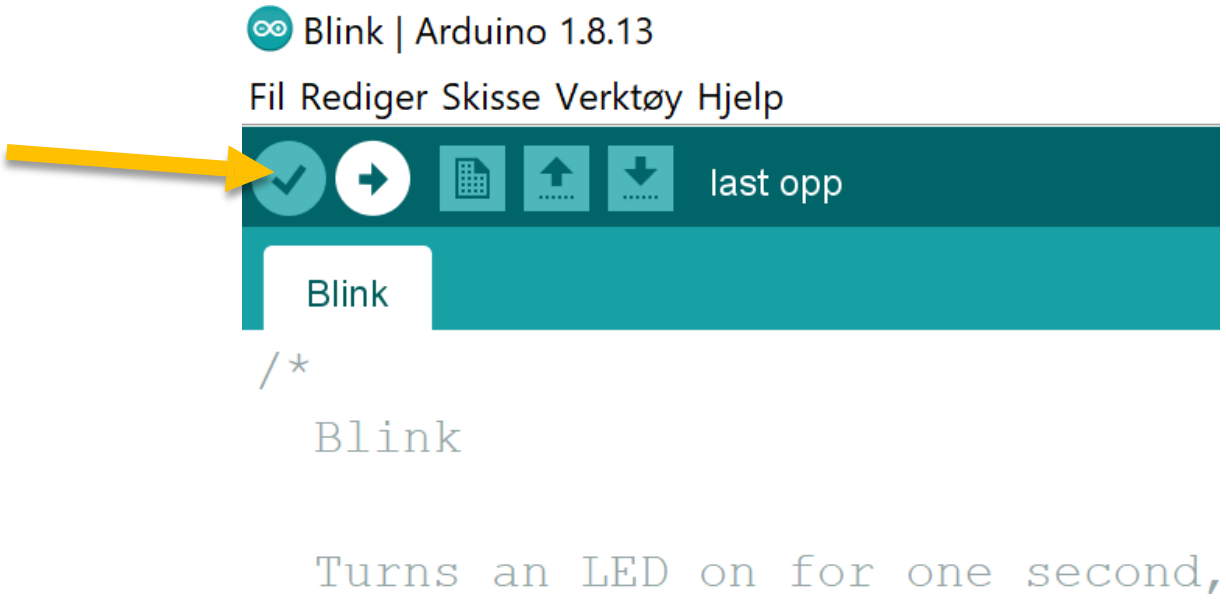
void loop() {
digitalWrite(ledPin, HIGH); //sender strøm til variabelen vi laget, ledPin, og slår på LED-en
delay(1000); //pauser programmet i ett sekund
digitalWrite(ledPin, LOW); //slår av strømmen til ledPin, og slår av LED-en
delay(1000); //pauser programmet i ett sekund
```

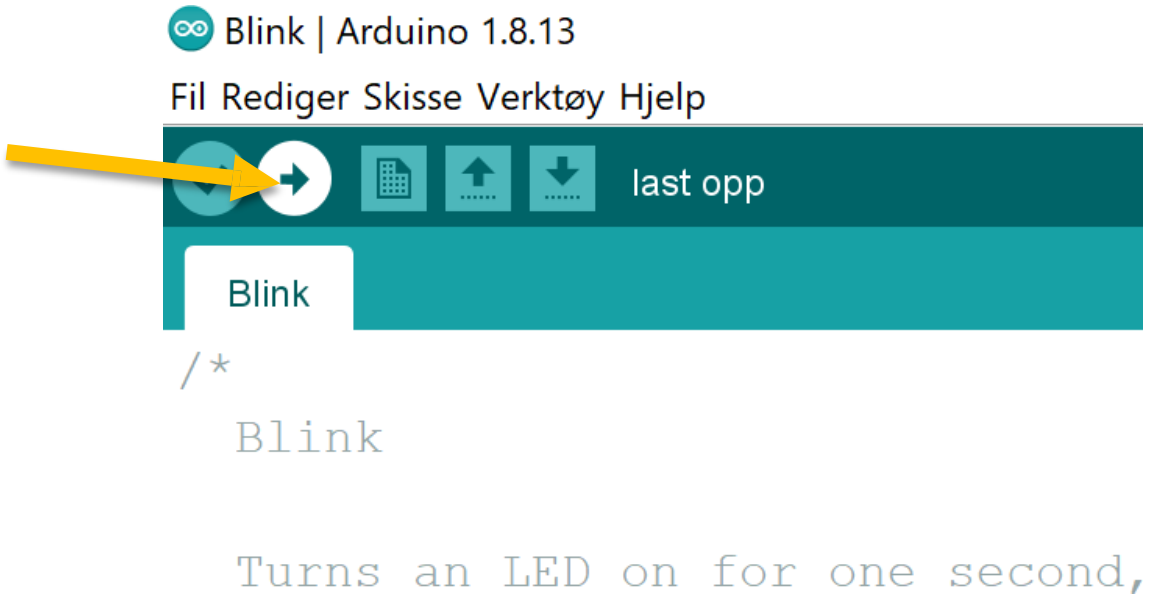
```
void setup() { int ledPin =
13;
pinMode(ledPin, OUTPUT);
}

void loop() {
digitalWrite(ledPin, HIGH);
delay(1000);
digitalWrite(ledPin, LOW);
delay(1000);
}
```



# Kompiler og last opp skissen til Arduino



# Elektrisk strøm

## Grunnleggende prinsipper



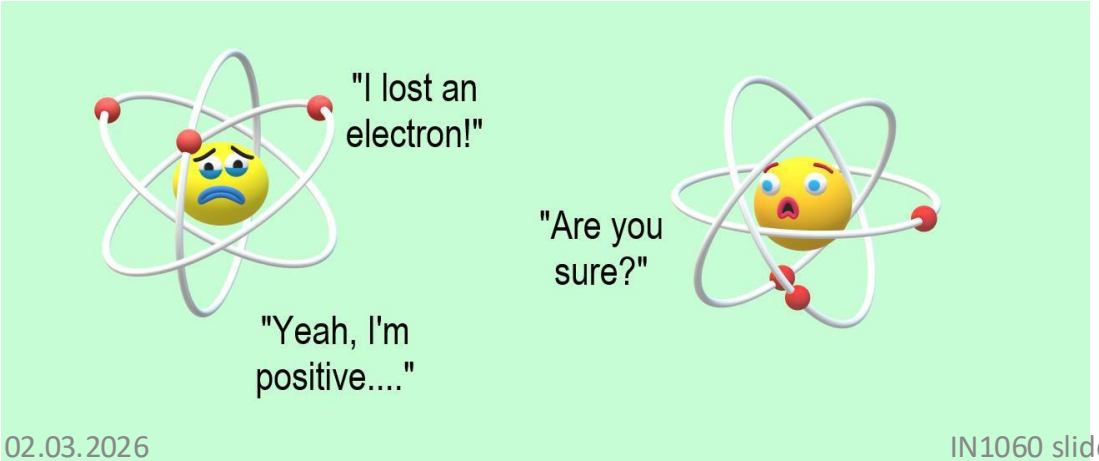
IN1060 slides created
by Heidi Bråthen

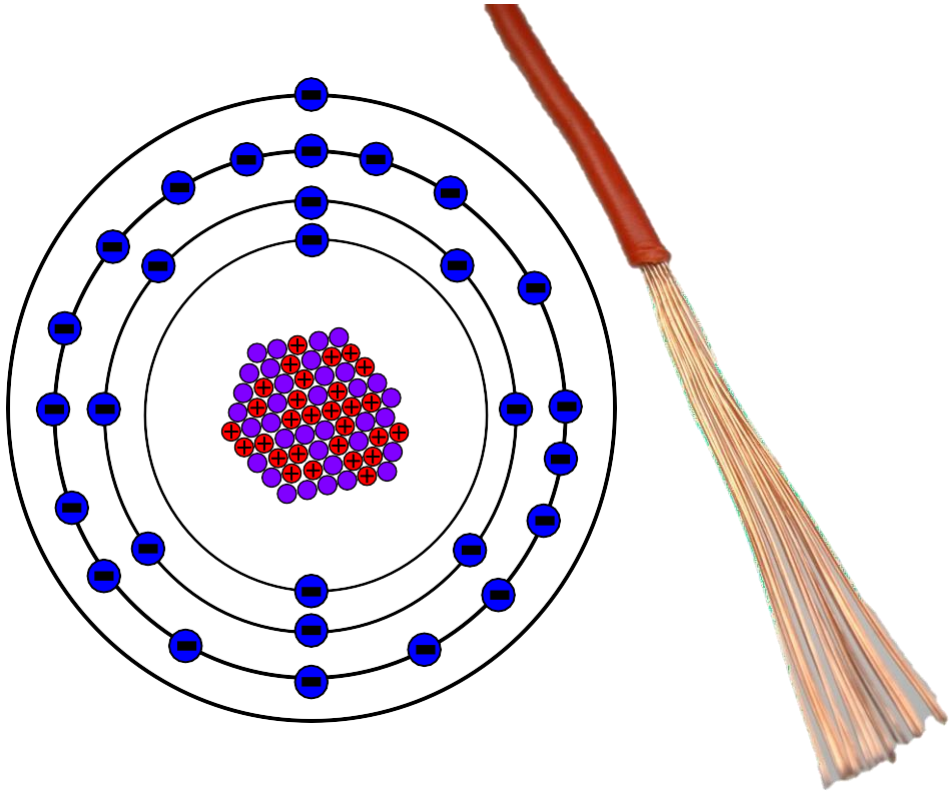
# Elektrisk strøm er ladede partikler i bevegelse

Atomer har positivt ladede kjerner (protoner+) og negativt ladede elektroner(-)

Både negativt og positivt ladede partikler i bevegelse kan skape strøm, avhengig av materialet strømmen går i

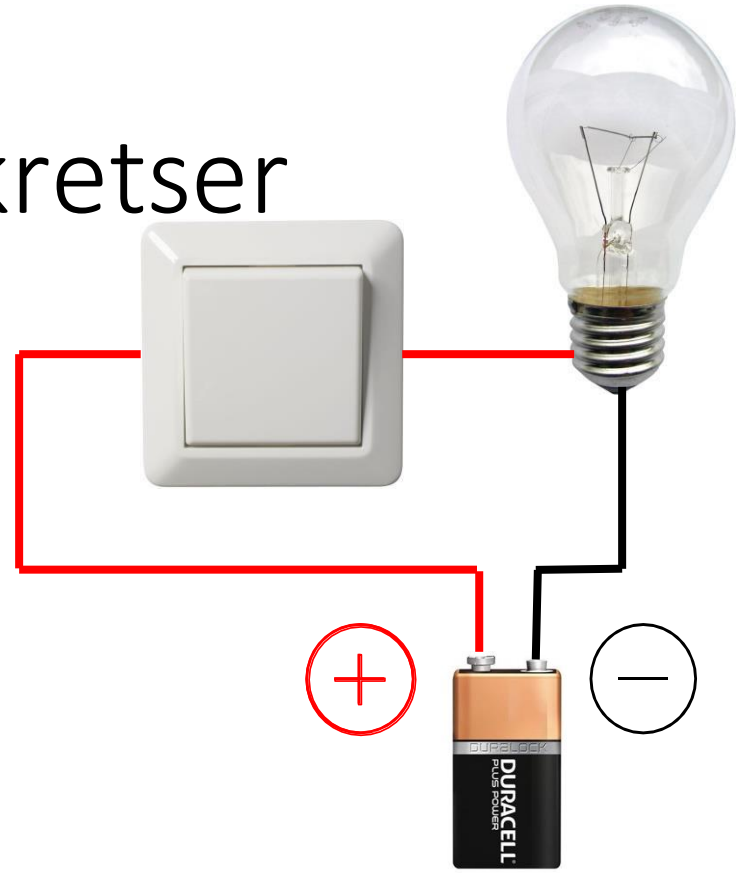
I metaller, som er gode elektriske ledere, er det lett for elektronene å bevege seg fra atom til atom



Kobber-atom CU (v) og kobbertråder i ledning (h)

# Elektrisk strøm går i kretser




**Krets**

En sluttet (komplett eller hel) bane som kontrollert strøm beveger seg rundt i

Består av ledninger, komponenter og en strømkilde (f.eks. batteri)

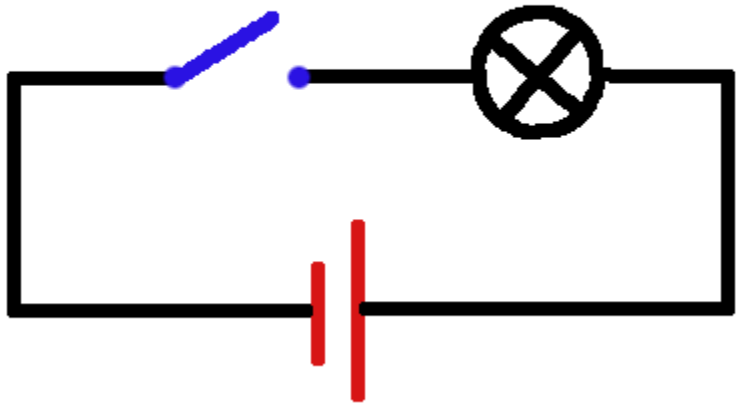
**Komponenter**

Grunnleggende bestanddeler i en krets (f.eks. lyspære eller LED og

bryter)

**Kretsdiagram**

en skjematisk tegning av en krets som beskriver hvordan kretsen skal bygges



# Tre prinsipper

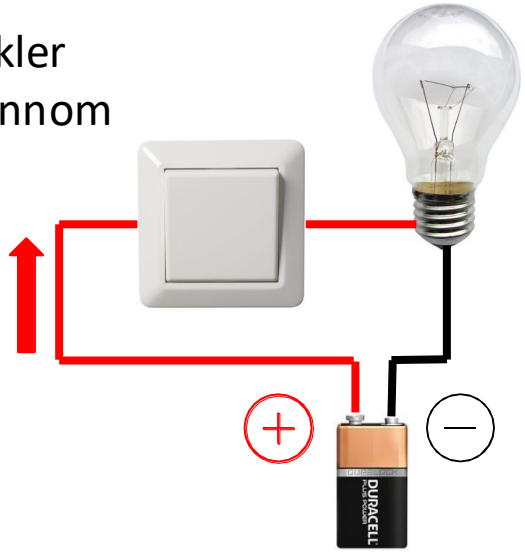
Tre prinsipper påvirker hverandre og hvor mye strøm som går gjennom kretsen

**Spenning (V)** tilfører energi til kretsen og «dytter» elektronene gjennom kretsen

**Strøm (I)** er elektronene som strømmer gjennom kretsen, måles i ampere (A)

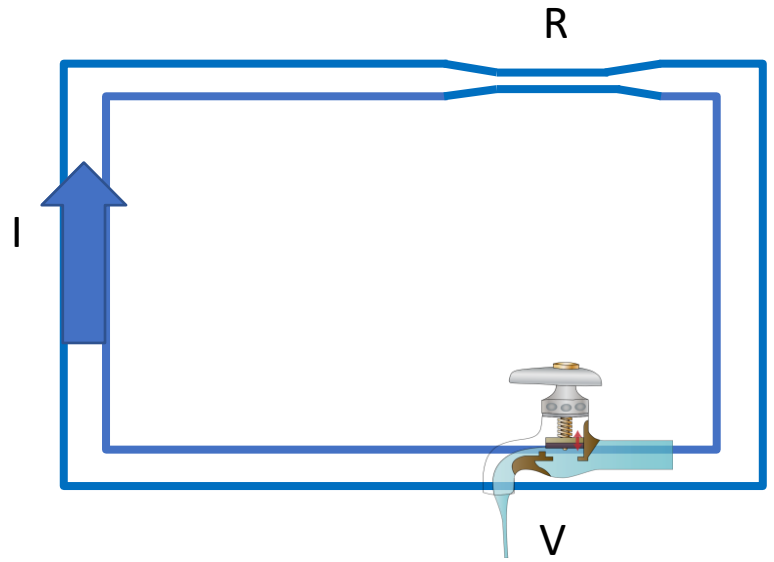
**Motstand (R)**: bremser og reduserer strømstyrken ved å gjøre det vanskeligere for strømmen å bevege seg gjennom kretsen, måles i ohm

I: Ladde partikler strømmer gjennom ledningen




R: Tråden i lyspærer har så høy motstand at den gløder (R)

V: Batterier er spenningskilder som «dytter» partikler gjennom kretsen




Det er vanlig å forklare strøm ved å sammenligne en krets med et vannrør-system

# Vann-analogien: kretser som vannrør

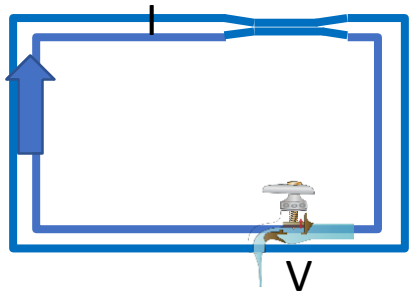
**Strømmen** er vannet som strømmer gjennom rørene

-(det står alltid vann I alle delene av rørsystemet)

**Batteriet** er som en vannpumpe, eller som en kran som man skrur opp og igjen

Rør som blir smalere slik at mindre vann passerer gjennom representerer **motstand**

R






# Spenning volt (V)

Spenning er forskjell i ladning mellom to punkter

Volt “faller” eller “dytter”

Volt avgjør hvor mye strøm vi kan få:

Jo høyere spenning (V), jo flere ladde partikler beveger seg i kretsen

Vannanalogien:

Spenning er som en vannpumpe som dytter vannet gjennom rørene

5V



5V



9V




230V



Batterier er spenningskilder



# Strømstyrke ampere (A)

Hvor sterk strømmen er

A = hvor mange partikler som beveger seg forbi et gitt punkt per sekund

Vann-analogien:

Strømstyrken er som hvor fort vannet strømmer gjennom rørene

- Styrken på strøm i en elv måles i hvor mange liter vann som strømmer forbi et gitt punkt pr sekund, ikke hvor mye vann som finnes i elven

40mA 9A 30 000A










by Heidi Bråthen

# Motstand (R)

Gjør det vanskeligere for de ladde partiklene å bevege seg I kretsen

Senker hastigheten i hele kretsen, ikke bare etter motstanden

Måles i ohm Vann-analogien

Motstanden er som rør som blir smalere, en hageslange med knekk eller en demning som gjør at mindre vann passerer gjennom



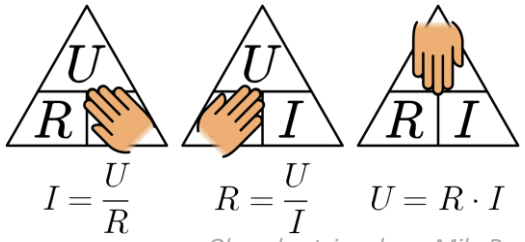
Tråden (filamentet) i lyspærer har så høy motstand (R) at den gløder



I Arduino-settet har dere motstander i form av disse komponentene



# Ohms lov



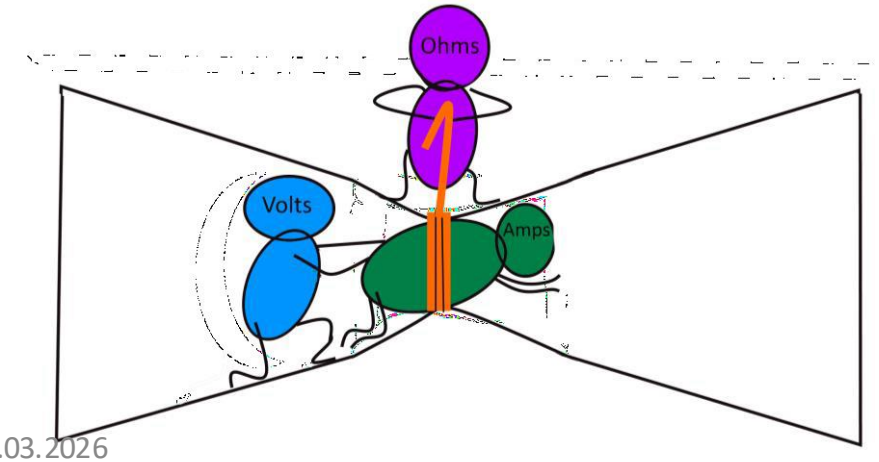

*Ohms lov triangle av MikeRun*

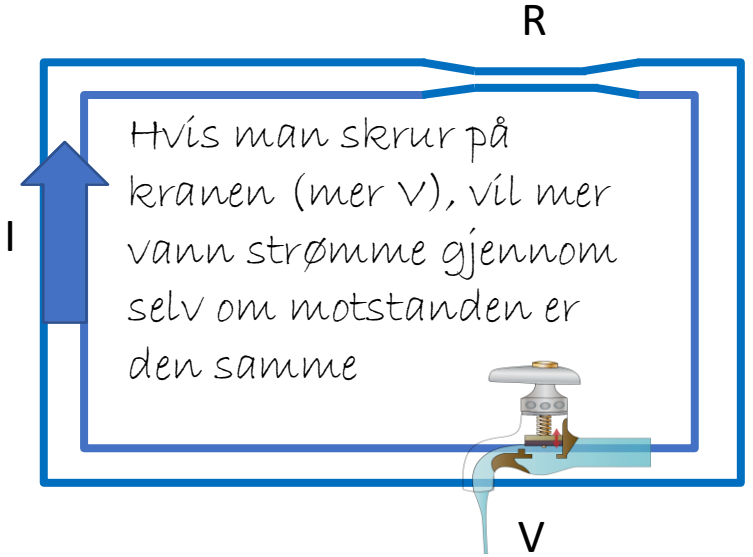
Ohms lov uttrykker forholdet mellom spenning, strøm og motstand i en krets (se s 30 i Arduinoboka)

V = volt, spenning, batteri

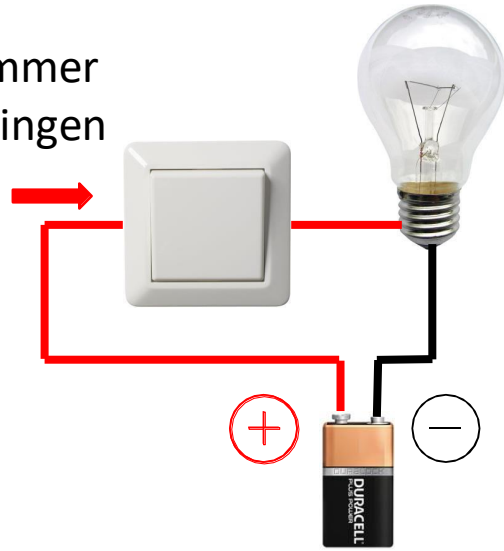
R = resistance, motstand, glødepære, resistorer

I = current, strømstyrke, ampere, ladde partikler som flyter i kretsen



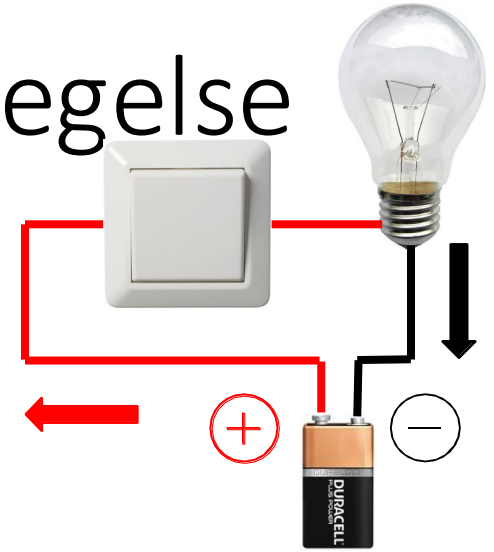

I/amp: Ladde partikler strømmer gjennom ledningen



R/ohm: Tråden i lyspærer har så høy motstand at den gløder (R)

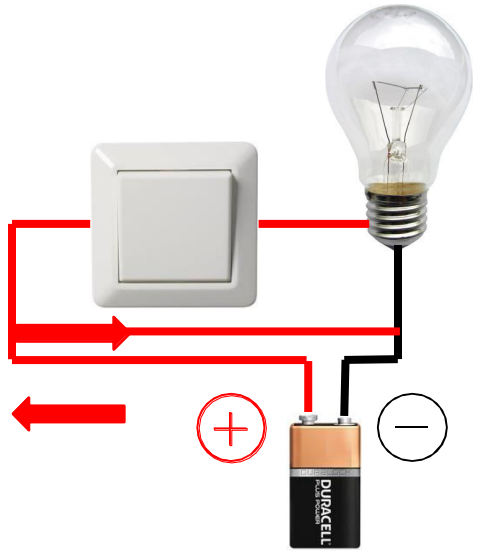
V/volt: Batterier er spenningskilder som «dytter» partikler gjennom kretsen

# Strøm er ladede partikler i bevegelse




Strømmen går fra pluss til minus

Hvis strømmen går rett fra pluss til minus uten noen komponenter imellom blir det kortslutning, noe som kan ødelegge komponenter og brettet



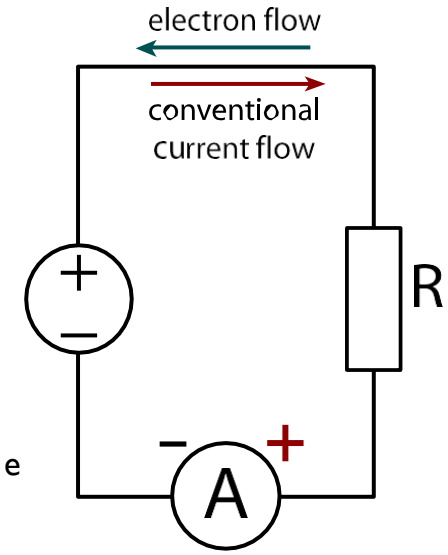

Må ha minst ett komponent i en krets, for eksempel en resistor

Strømmen tar raskeste vei (ikke korteste vei)

# Men hvilken vei går strømmen egentlig?

Strømmen kan gå begge veier avhengig av hva slags ladning partiklene har, + eller –

I metalledninger er det elektroner(-) som beveger seg. De beveger seg egentlig fra minus til pluss!




Den **konvensjonelle strøm-retningen** er den motsatte, fra pluss til minus.

Konvensjon: en regel eller ordning det er allmenn enighet om

Grunnen til at vi sier at strømmen går fra pluss til minus selv om vi vet at den går andre veien er at vi følger en **notasjonskonvensjon som følger den konvensjonelle strømretningen**

Notasjon: vi bruker bokstaver eller andre tegn for å sette noe i system

Alle som lager Arduino kretser, tegner kretsdiagrammer, selger komponenter, og bruker komponenter forholder alle seg til samme retning; fra pluss til minus.

Det motsatte er **elektronflytnotasjon**, notasjon basert på den veien elektroner flyter, men det er ganske sjeldent i bruk.

**NB! Vi kan ikke velge notasjon, Arduinoen er laget for å virke med den konvensjonelle notasjonen**

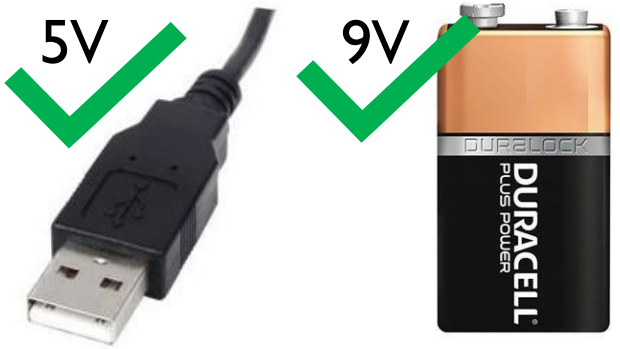
# Sikkerhet

Og hva er egentlig kortslutning?

# Sikkerhet #1: riktig strømkilde




Riktig strømkilde: 5-12 volt




Godkjente strømkilder: USB-porten på PC, 9V-batteri

**Viktigste sikkerhetshensyn: Ikke koble på veggstrøm**
**Arduinoen kan ikke håndtere 230 volt og strømmen vil gå gjennom og gi støt**
**Mengden strøm som kommer fra veggsokler setter liv i fare**




Unngå å bruke Lithiumbatterier som strømkilde til Arduino Uno.
Det selges Lithium-batterier til Arduinobrett som går på 3V, som brett som er ment for wearables.
Lithiumbatterier er følsomme for oppvarming, feil og eksploderer ved bruk.
Bruk kvalitetsprodukter, bruk kun for 3V-brett, ikke for Uno. Vær oppmerksom.

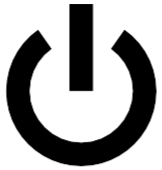


# Sikkerhet #2: Brannfare



**Sjekk kretsen**
Gjør det til en vane å alltid se over kretsen før du kobler til strømmen

- Kretsen er riktig satt opp (sjekk kretsen mot kretsdiagrammet)
- Koblingene er ikke løse
- Ingen uisolerte ledninger eller komponenter kommer borti hverandre
- Ha komponenter mellom 5V og GND
- Alle komponenter står riktig vei



**Koble fra**
Ha som hovedregel at strømmen kobles av når du ikke er til stede



**Ventilasjon**
Sørg for god ventilasjon slik at varmen slipper ut



**Kvalitetskomponenter**
Bytt alltid ut defekte komponenter og bruk kvalitetskomponenter



**Riktig spenningskilde**
5-12 volt



**Unngå overlast**
mange komponenter på ett brett, komponenter som trekker mye strøm, som motorer



**Lithiumbaterier**
Vær oppmerksom (ikke anbefalt for Uno, kun for mindre prosjekter)

# Sikkerhet #3: beskytt komponenter

Koble ut strømmen/USB-kabelen mens du bygger **og endrer** kretser

Minst ett komponent i kretsen før du kobler til USB (i tillegg til bryter)

Sjekk om komponentet tåler 5V, eller sett på riktig resistor

Sjekk om komponentet må kobles i en spesiell retning (gjelder mange)

Sjekk kretsen en gang til mot kretsdiagrammet før du kobler til USB-kabelen

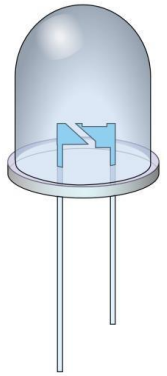
Vær oppmerksom på at enkelte komponenter trekker mye strøm, som motorer, og at disse må kobles med komponenter som beskytter Arduinoen mot overlast

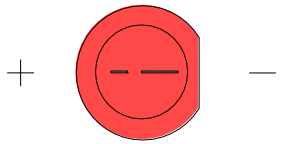
# Komponentene i kretsen vår

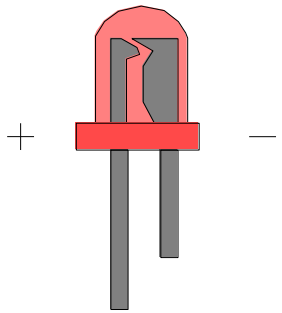
LED

Resistor

Breadboard







# LED

Light Emitting Diode

Tåler ikke 5V, vanlig å bruke en 220-330 ohm resistor

Det lengste benet er 5V/+ på LED i settet

Pæren er butt i jord-enden (-)

# Resistor

Bremser strømmen

Beskytter komponenter som ikke tåler 5V

Fargede bånd angir styrken på resistoren

- se fin oversikt i boka s. 41



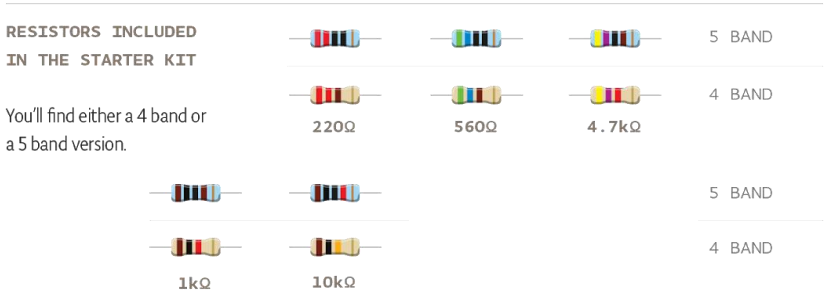
# Reistoren i mitt sett ser ikke sånn ut?

Det er vanlig å merke resistorens styrke med 4 bånd eller 5 bånd

Deres resistorer kan være merket med 4 eller 5 bånd om hverandre

Det er kun 5 resistorer i settet deres

Se denne oversikten og finn riktig resistor



# I settet vårt er det kun 5 resistorer

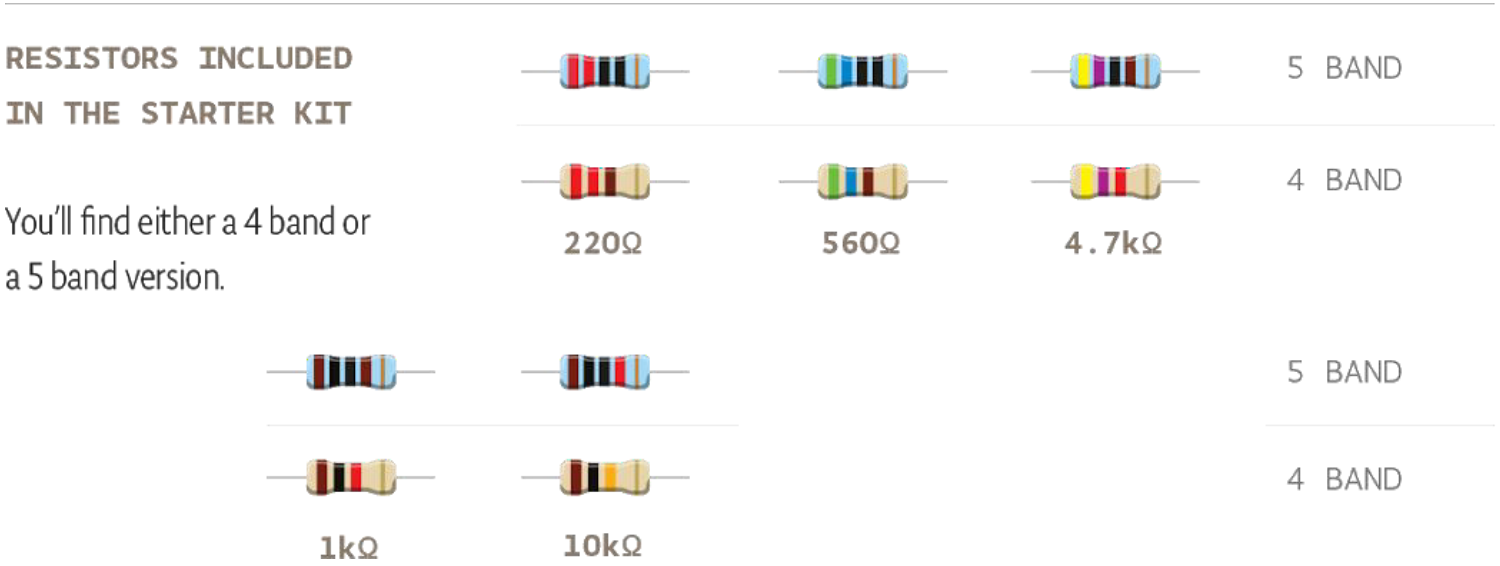
5 stk 560Ω

5stk 220Ω

1stk 1kΩ,

2stk modstande 4,7kΩ

1stk modstand 10kΩ



# Fargeblindhet

Resistorene er utfordrende for mange som har fargeblindhet

Tips til dere: Få tak i små zip-poser og merk dem med penn

Oversikten over resistorene i settet ligger på Arduino-siden

# Hva skjer hvis jeg bruker feil resistor?

Så lenge du har en resistor går det stort sett greit med komponentene

Komponenter som får mer strøm enn de skal (for liten resistor) kan ta skade

Komponenter som får for lite strøm (for stor resistor) kan virke dårligere eller ikke virke

# Breadboard

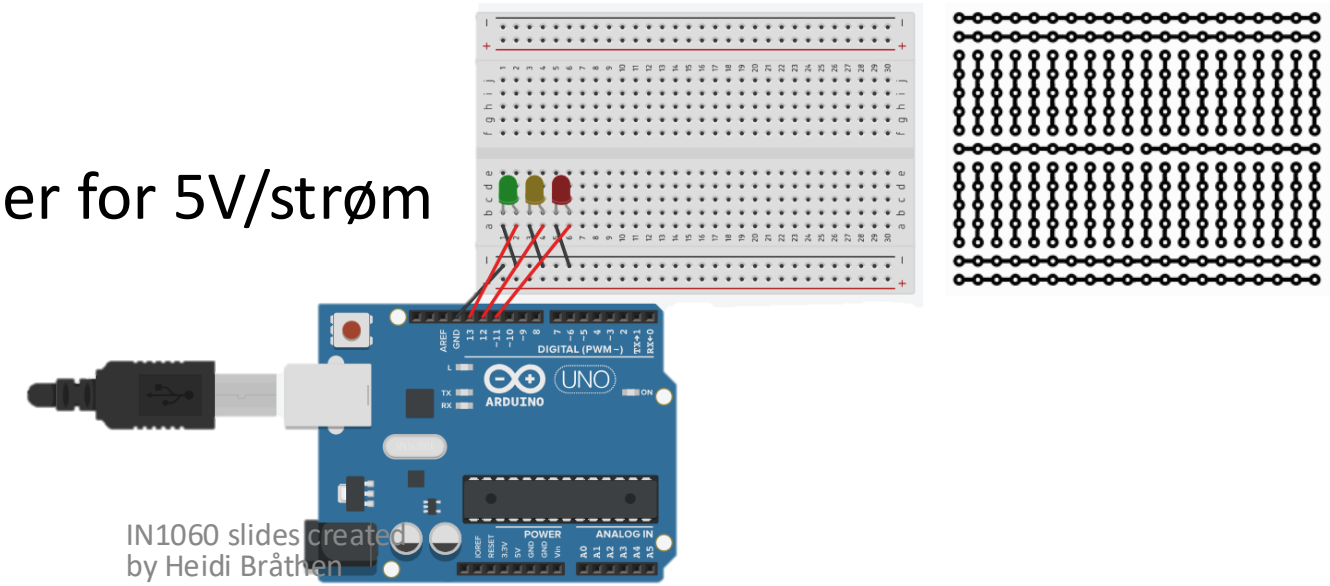


Gjør det enkelt å koble sammen komponenter

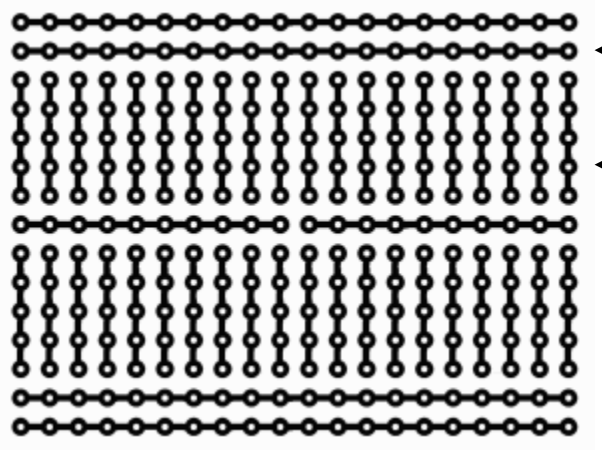
Trenger ikke å lodde («solderless breadboard»)

Hver rad deler en metallstripe som fungerer som ledning

Langs brettet går det to skinner for 5V/strøm (rød) og GND/jord (svart/blå)

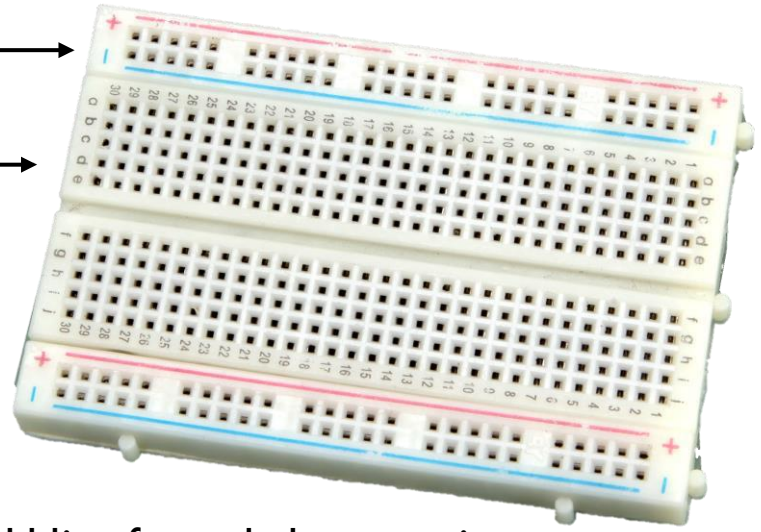


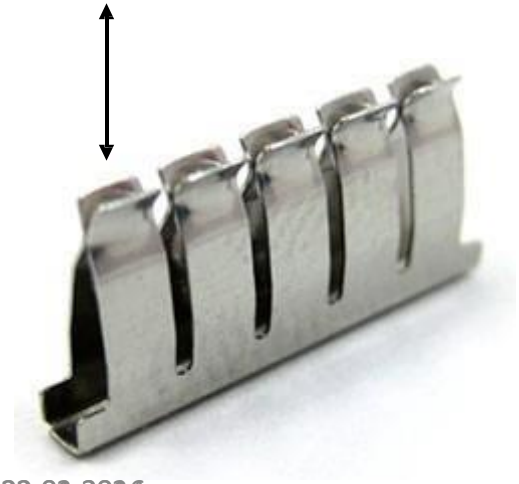
# Hvordan virker breadboardet?



Skinner

Rader





Breadboardet er et system av ledninger med klips for enkel montering av komponenter i en krets

De horisontale lange stripene merket med - og + kalles skinner.

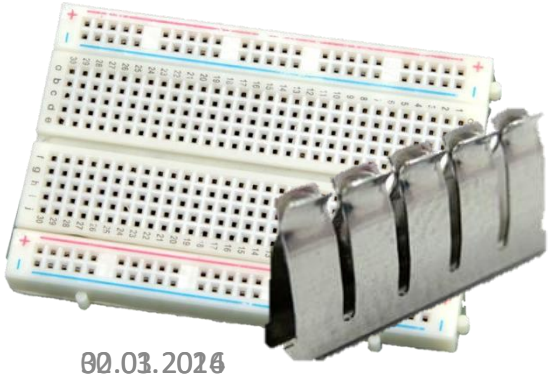
De vertikale stripene merket med tall kalles rader

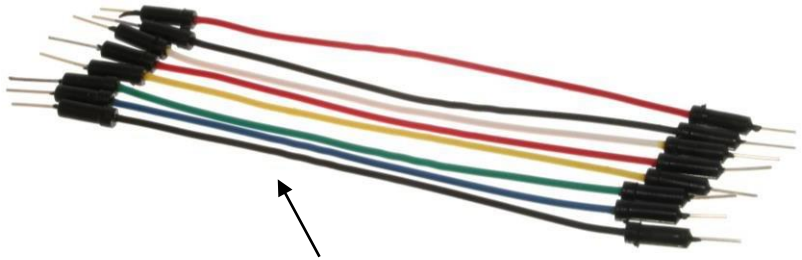
Det er laget av rekker med smaå klips I metal som holder fast komponenetene

# - Hva er egentlig en ledning?

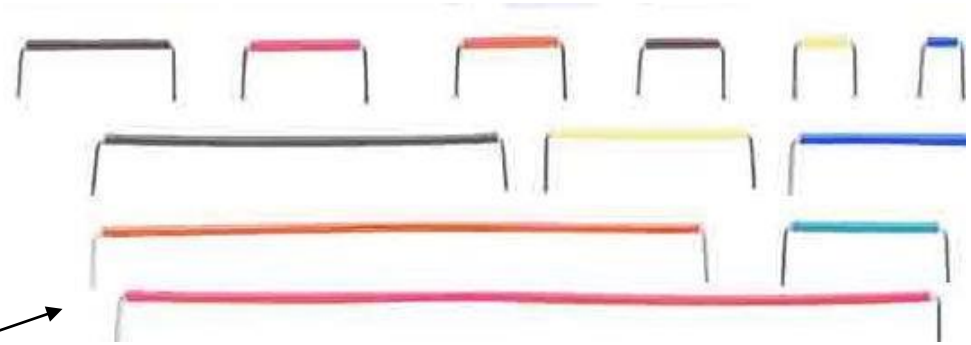



- En ledning består av et strømførende materiale, ofte et metall.
- For at ledninger til + og – ikke skal komme borti hverandre og danne kortslutning må vi isolere metallet. Plast og gummi leder strøm dårlig, og er derfor gode isolasjonsmaterialer.
- I settene deres har dere ulike typer ledninger. Noen ser litt uvanlige ut og de kan ha ulike farger, men alle virker på samme måte.





Alle disse er ledninger



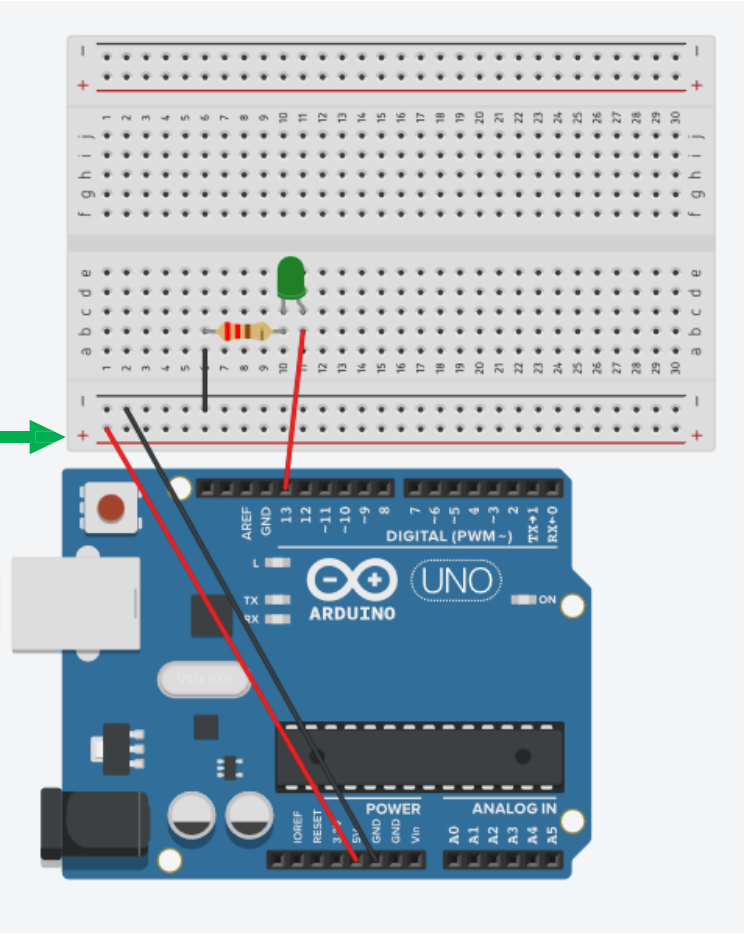
# Hvorfor trekke ledninger til rekkene?

I instruksjonen for den første kretsen anbefalte jeg å trekke ledninger fra Arduinoens 5V og GND til rekkene først

Men i eksempelet ble ikke 5V-rekken brukt til noe, den hadde ingen funksjon!

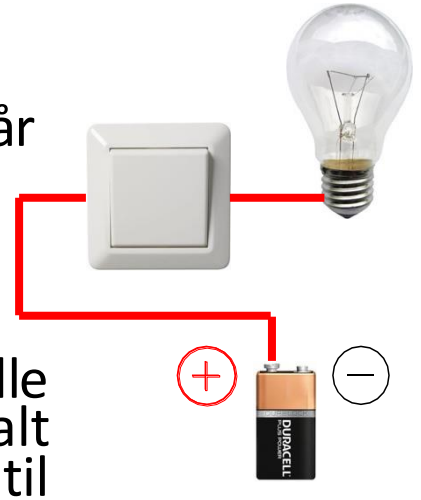
Grunnen til anbefalingen er at Arduinoen kun har to 5V-pinner og tre jord, og at man fort slipper opp for pin'er når det blir flere komponenter. Da er det nyttig å alltid ha strøm og jord tilgjengelig i rekkene

(Senere, hvis man har en kompleks krets med mange ledninger kan det derimot være lurt å kutte ut overflødige ledninger fordi de kan skape unødvendige forstyrrelser)



# Hvorfor må kretsen gå til jord/GND?

**Ingenting vil virke hvis ikke hele kretsen går til samme jord/GND-punkt**



Hvis ikke kretsen går til jord, vil ikke kretsen være sluttet (hel), fordi den ikke går fra pluss til minus. Da kan ikke elektronene strømme, for det blir ingen spenningsforskjell som dytter dem.

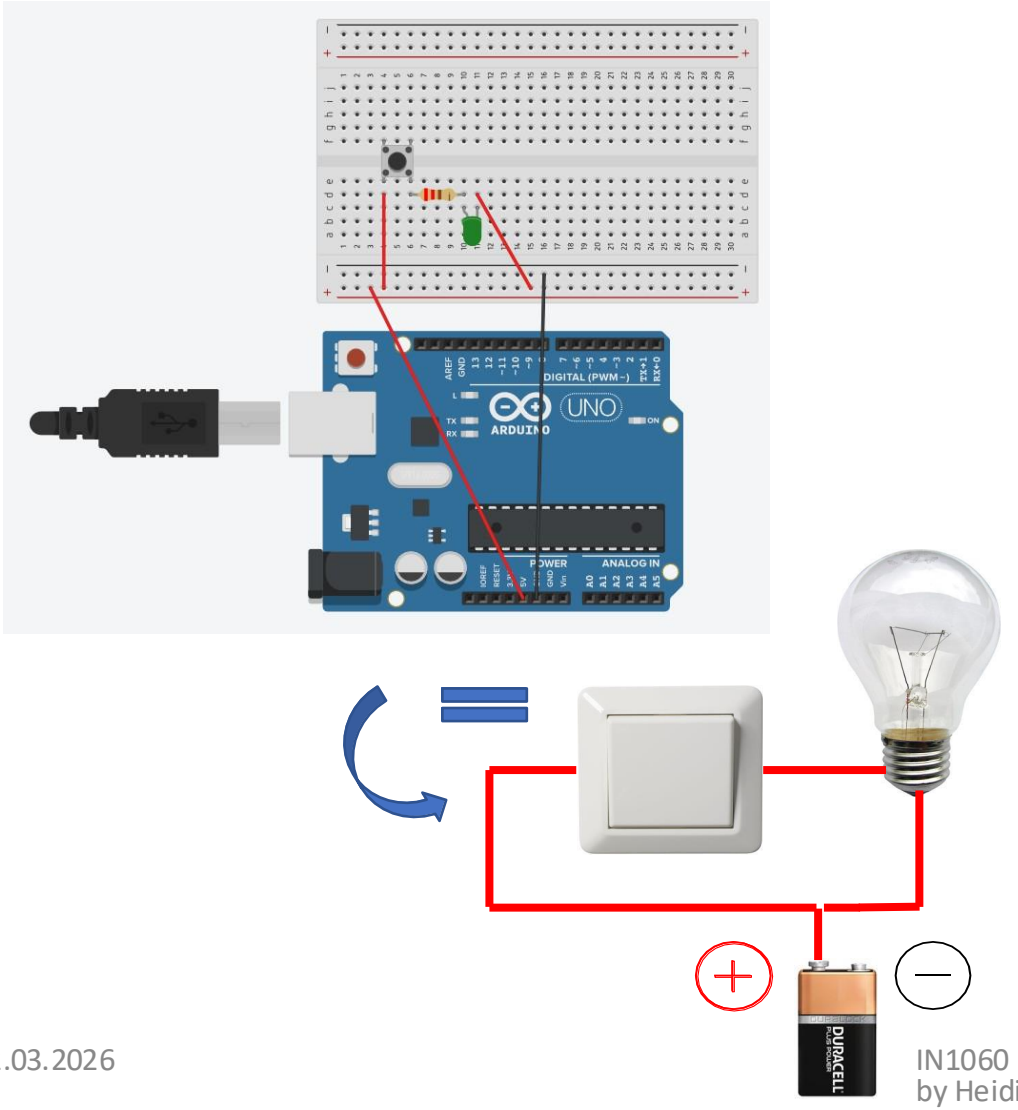
Jord/GND på Arduino er et *referansepunkt* hvor det er 0 volt. Det betyr at alle andre signaler blir målt mot dette punktet som representerer 0 V, og siden alt blir målt mot samme referansepunkt, kan man måle alle verdiene i forhold til hverandre slik at de gir mening i kretsen.

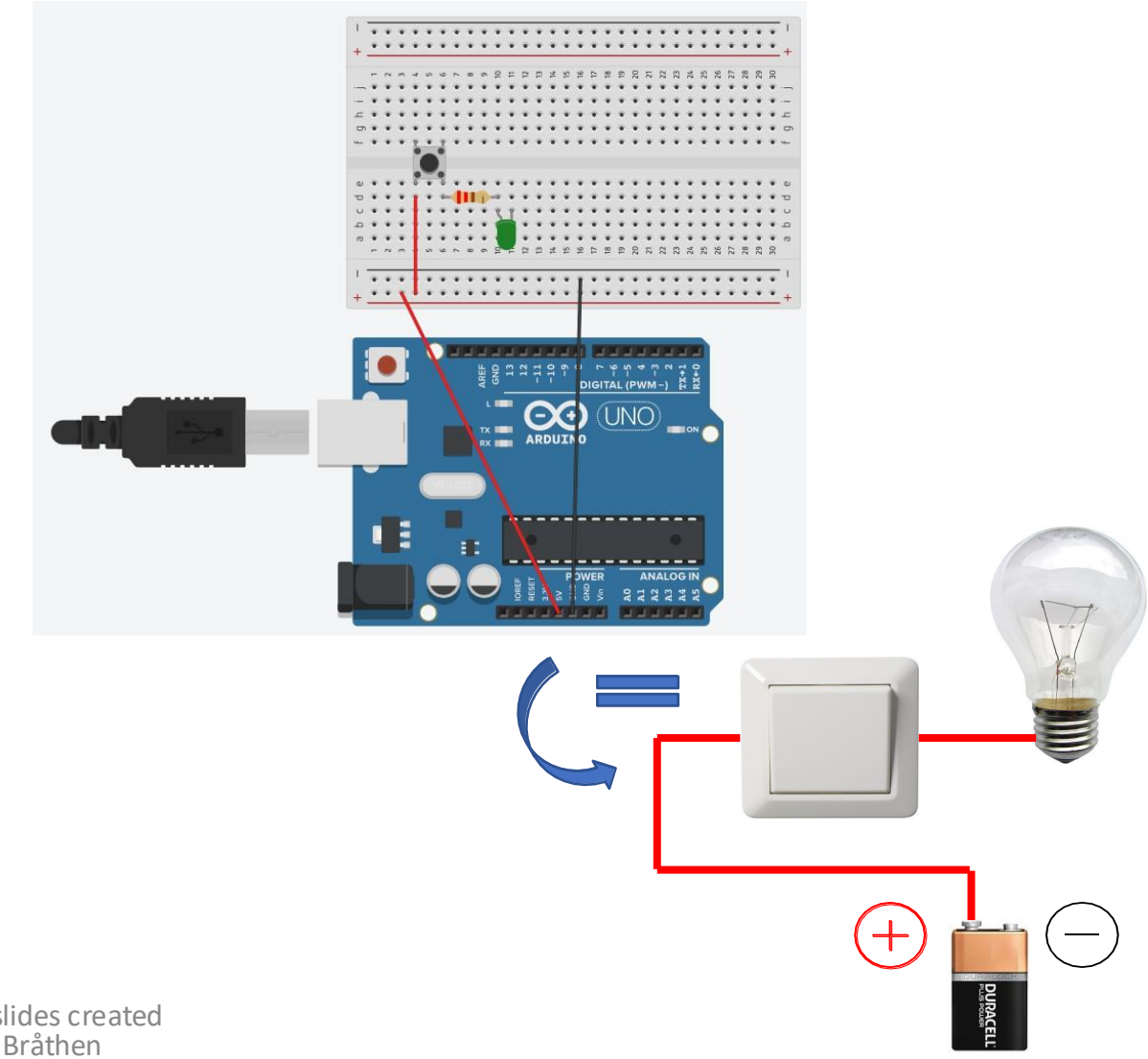


På store strømkraftverk og i elektriske systemer i hus er jord/GND ledet ned i den faktiske jorda/bakken. Det er vanlig praksis å jorde kretsen med et jordspyd, en meterlang metallbolt, som er drevet ned i den faktiske bakken

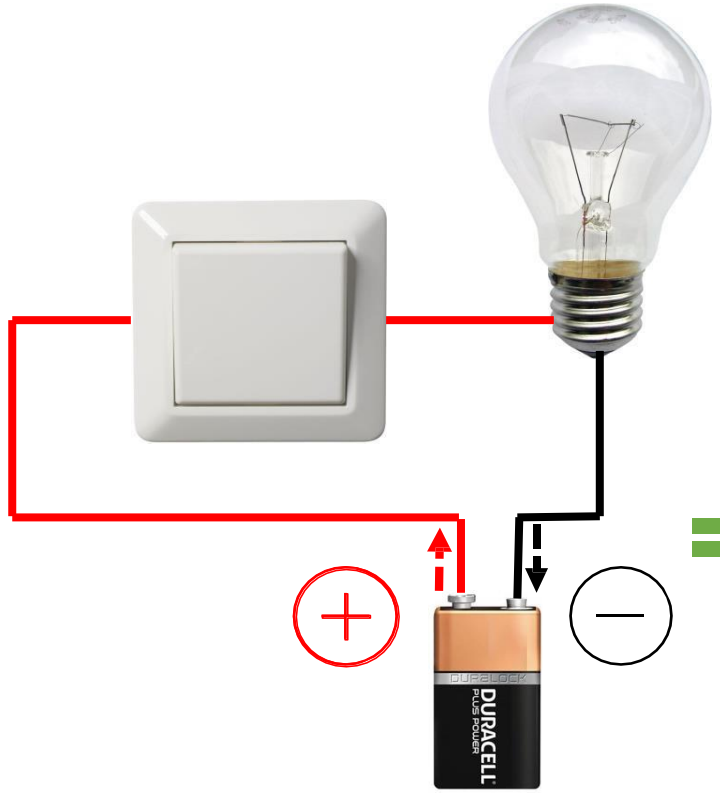
# Kretsen må kobles tilbake til jord/GND for at kretsen skal bli hel

Hvis vi ikke trekker kretsen tilbake til jord blir ikke kretsen hel, og strømmen strømmer ikke, ingenting virker.

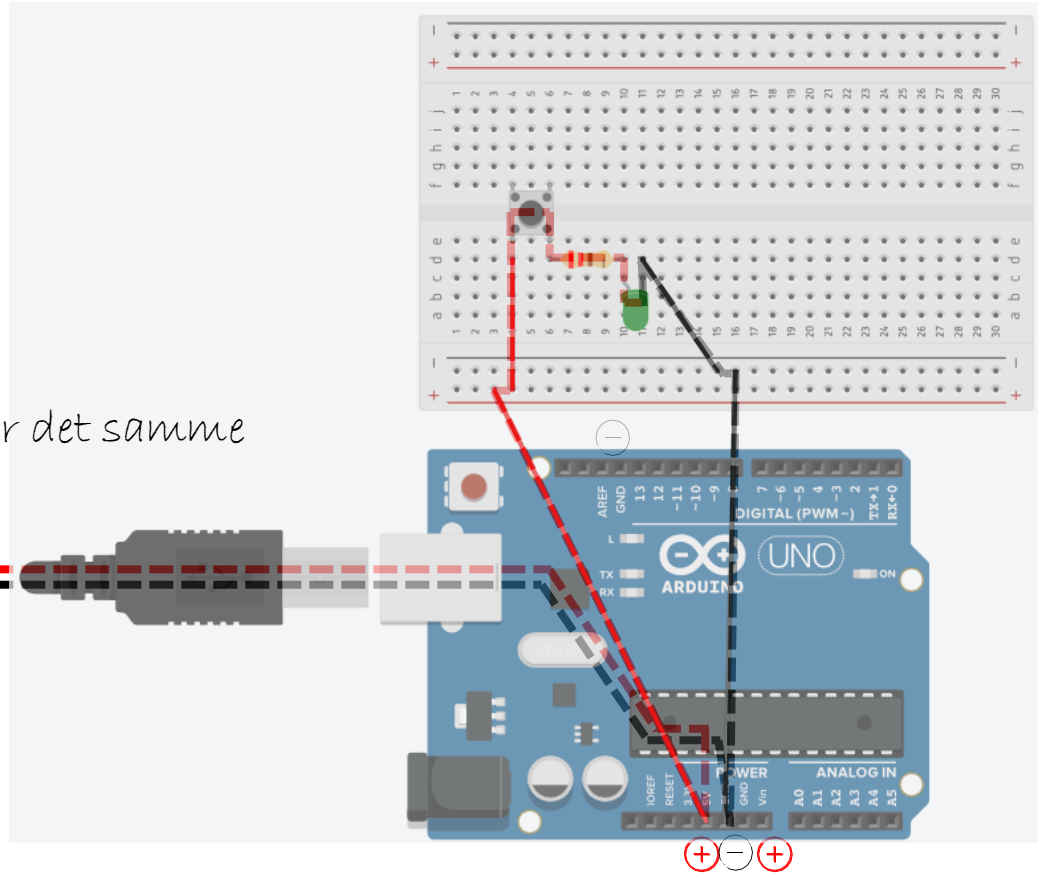




# Bygge kretser på Arduino og breadboard




Disse kretsene gjør det samme

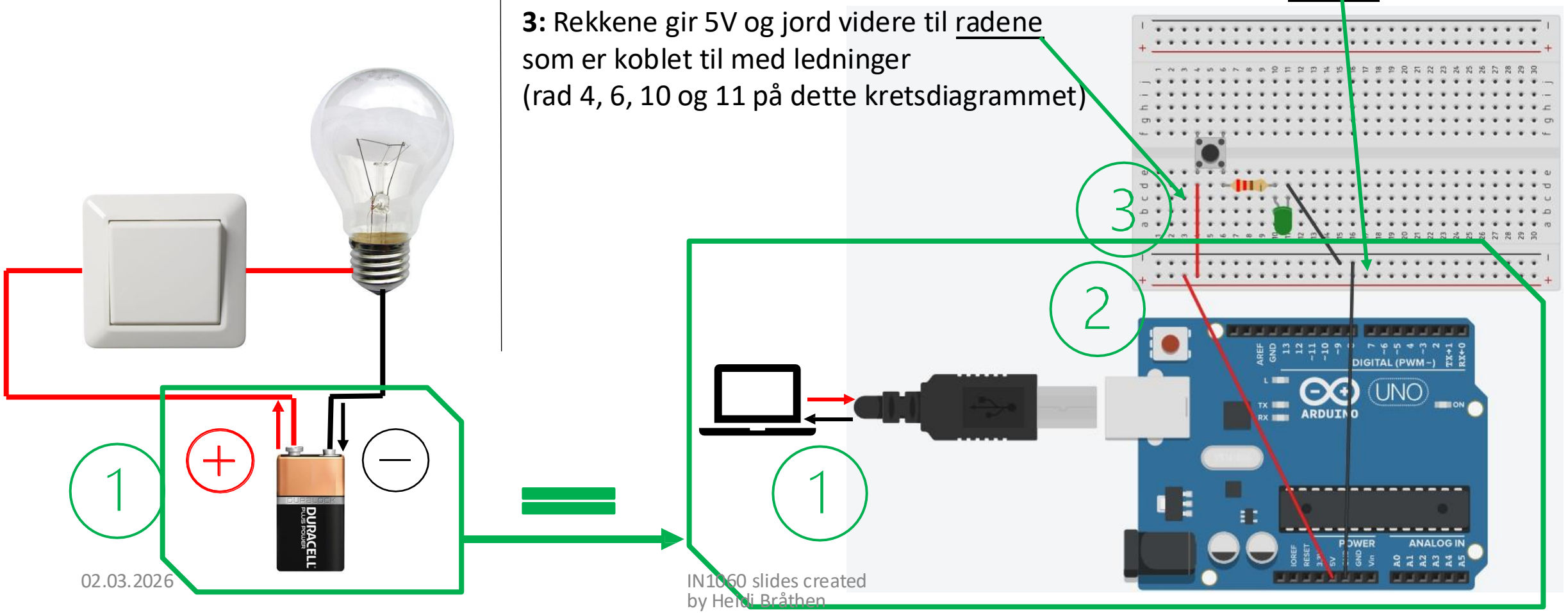



Det er kanskje ikke så lett å se at disse kretsene er like, for Arduinoen og breadboardet legger til litt flere elementer i den delen av kretsen som er spenningskilden (batteriet/ fra plusstrøm inn).

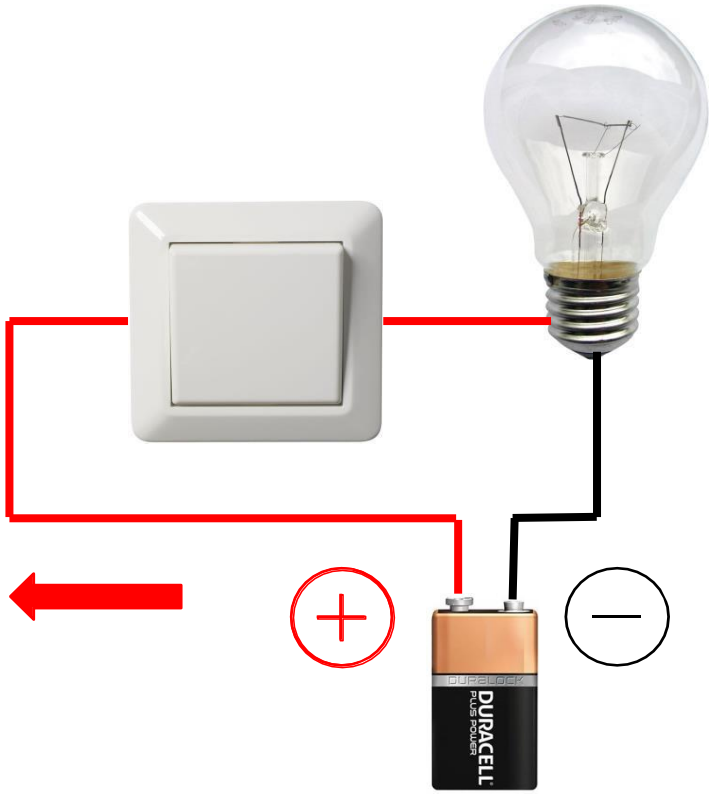
# Spenningskilden får litt flere ledd

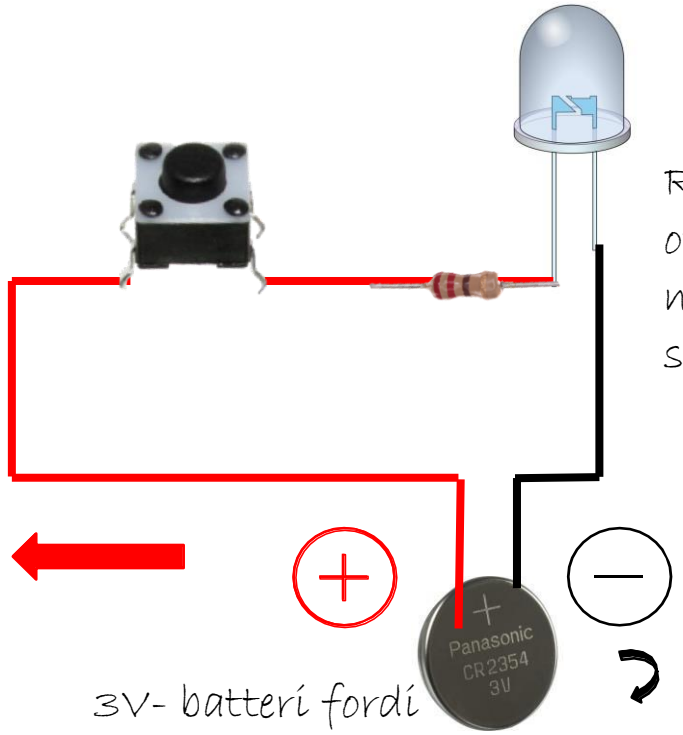
**Spenningskilden til denne kretsen:**
**1:** Batteriet gir 9 volt og jord til kretsen

**Spenningskilden til denne kretsen:**
**1:** PCen gir 5V og jord til Arduinoen
**2:** Arduinoen gir 5V og jord videre til breadboardets «+ og –» skinner
**3:** Rekkene gir 5V og jord videre til radene som er koblet til med ledninger
(rad 4, 6, 10 og 11 på dette kretsdiagrammet)



# Fra krets til breadboard steg for steg



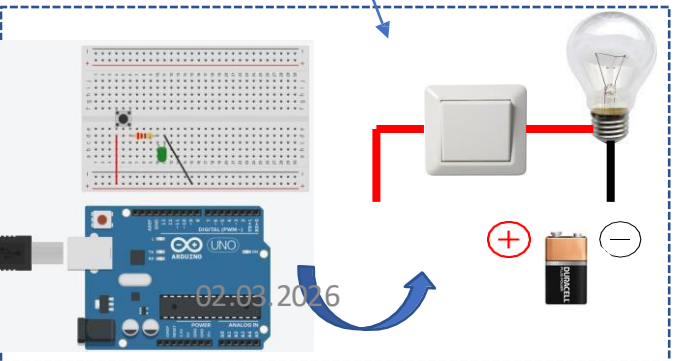

Resistor er egentlig overflødig med 3V-batteri, men er med for sammenligningens skyld

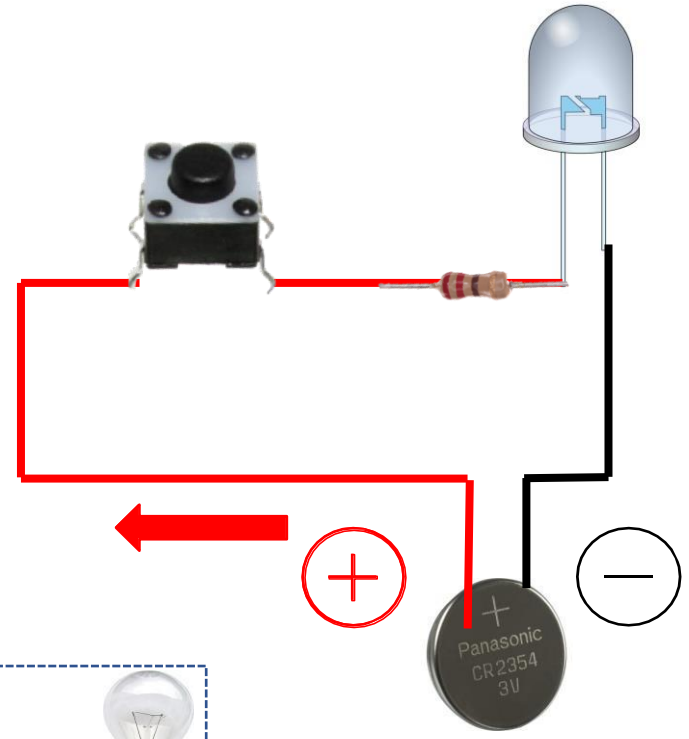
3V- batteri fordi LEDen ikke tåler mer strøm

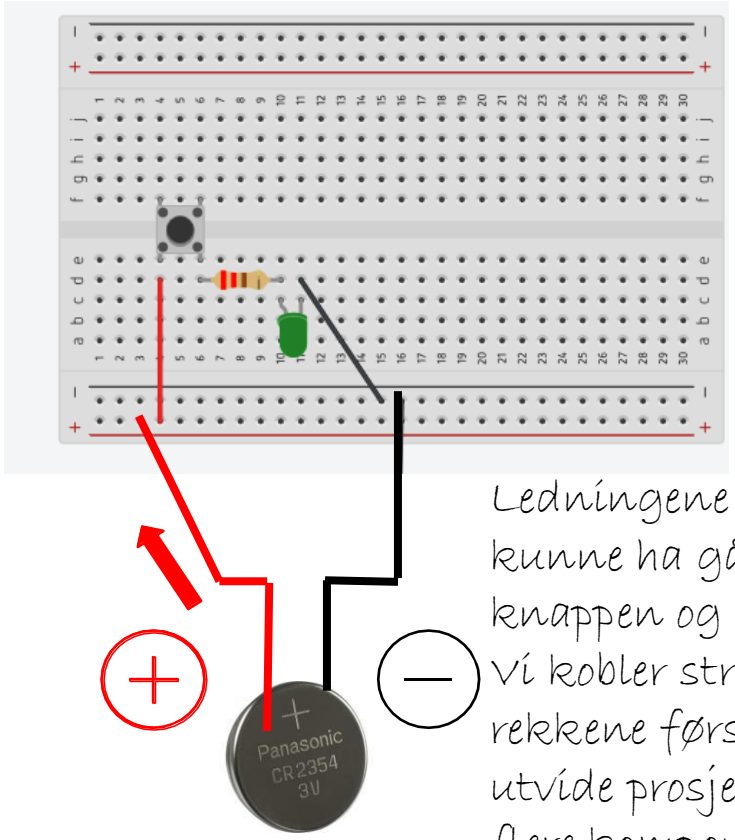
Klokkebatterier har plusspolen foran og minuspolen bak

# Breadboardet har ikke noen egen strømkilde

Hvis vi ikke trekker ledninger fra arduinoen til breadboardet finnes det ikke noe strøm tilgjengelig på breadboardet:



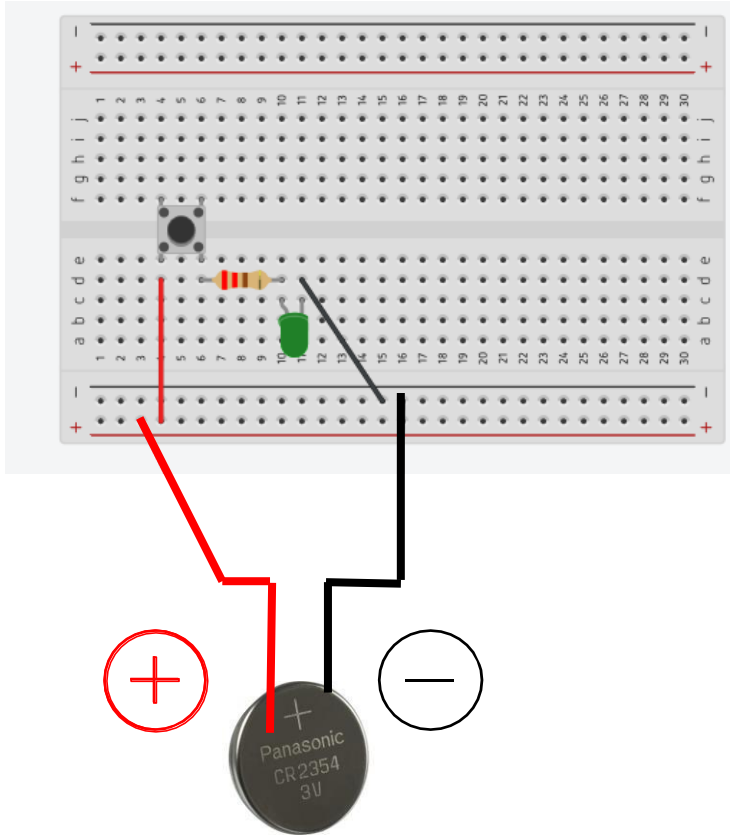


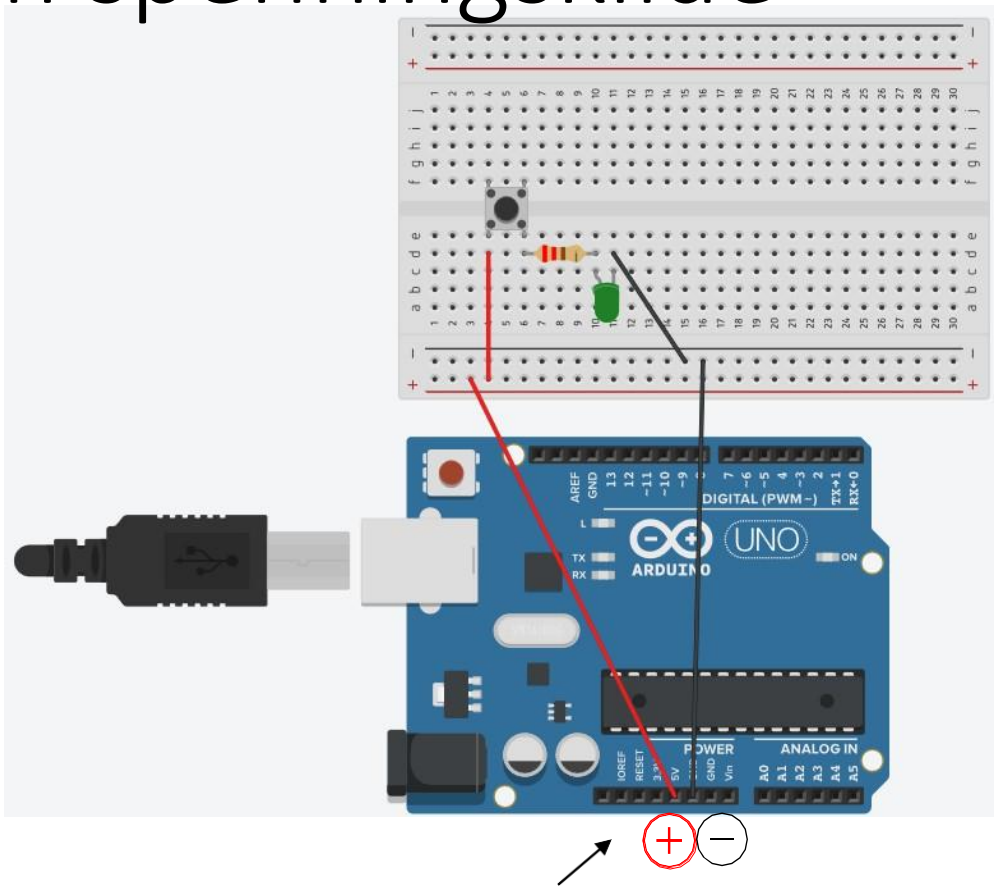




Ledningene fra batteriet kunne ha gått rett til knappen og rett fra LEDen. Vi kobler strøm og jord til rekkene først slik at vi kan utvide prosjektene våre med flere komponenter

Vi bruker breadboard fordi det gjør det enkelt å koble sammen mange komponenter på en gang. Breadboardet har ingen egen strømkilde, så vi må koble på batteri eller gi strøm fra Arduino.

# Arduinoen som spenningskilde




Hvis vi skal programmere knappen og LEDen til å gjøre mer enn å skru LEDen på når knappen trykkes ned, og skru av LEDen når knappen slippes opp, må vi programmere den. Da trenger vi Arduino.

Arduinoen har egne pin'er som gir ut 5V og går tilbake til jord. Disse pin'ene er spenningskilden slik som et batteri. Vi kan hente strøm fra Arduinoen over til breadboardet ved å koble ledninger fra 5V og GND til den lange røde og svarte rekken

# Utvide kretsen med en bryter

# Utvide kretsen vår med en bryt



**Trenger nå**

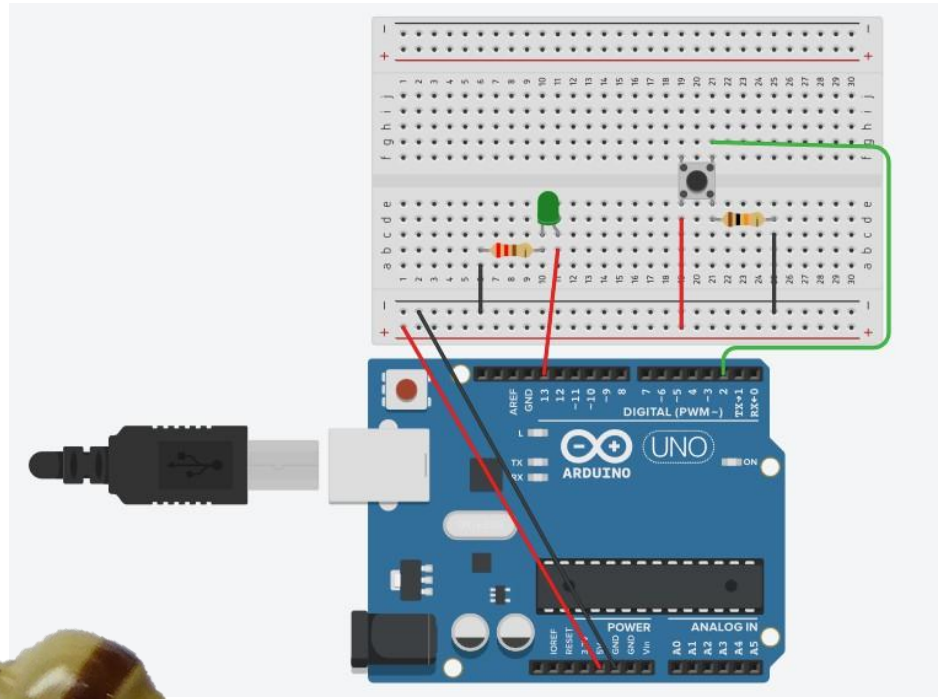
Kretsen fra i stad

Pushbutton (knapp/bryter) Ekstra ledning i

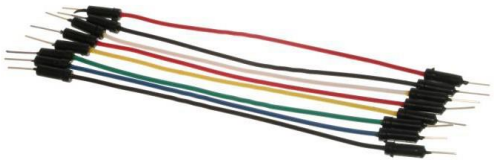
valgfri farge 10k resistor

Eksempelskisse fra Arduino:

last opp fra Fil – Eksempler – 02 Digital – Button





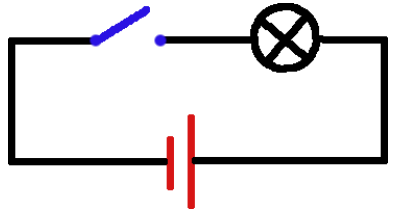
```
const int button =9;
const int led =2;
void setup()
{
  pinMode(button, INPUT);
  pinMode(led, OUTPUT);
}

void loop()
{
  if(digitalRead(button)){
  digitalWrite(led,HIGH);
  }
     else{digitalWrite(led,LOW);}
}
```

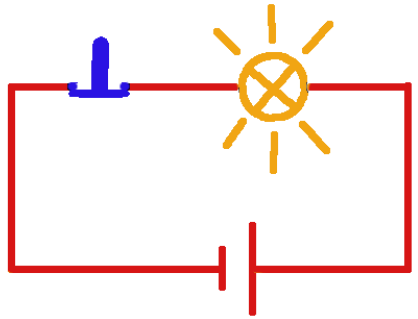
# Bryter



En bryter er et komponent som kobler sammen
to punkter i en krets (slutter kretsen) når den trykkes ned
og bryter kretsen når den slippes (eller omvendt)



Knapper (pushbuttons) er eksempler på brytere

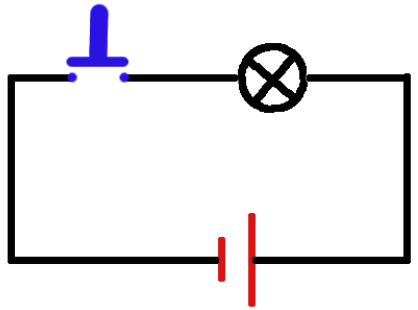


Enkle mekanismer: metall som kommer i kontakt for å slutte en krets

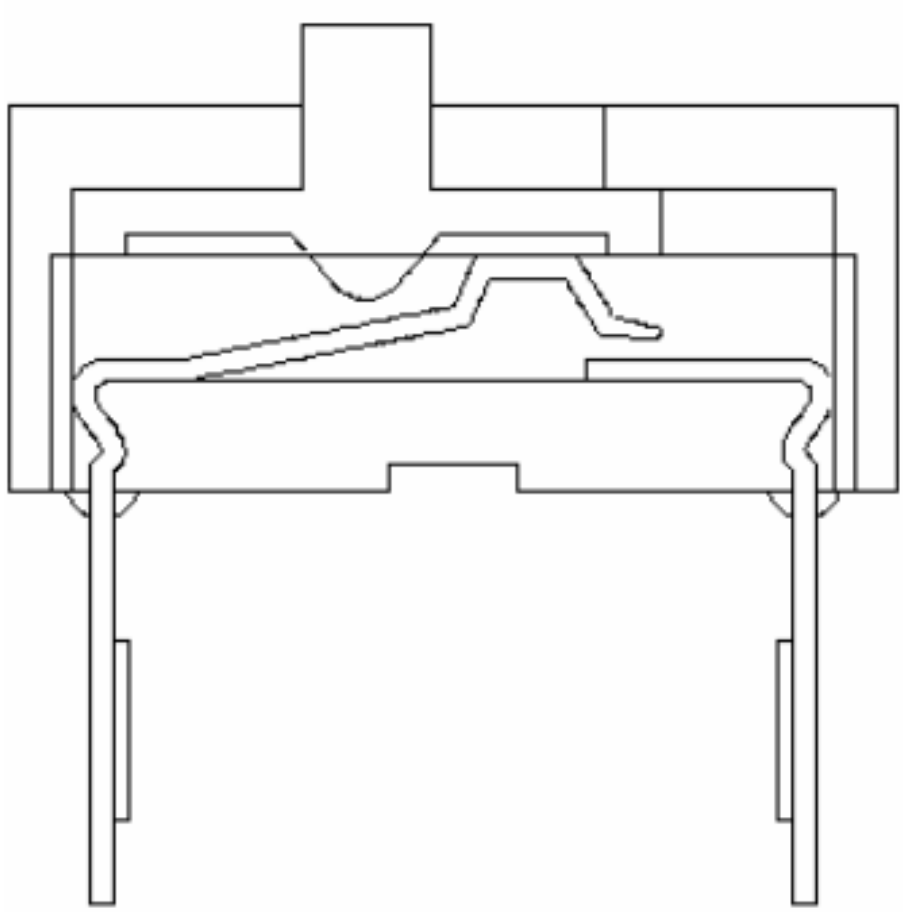
Ben 1 og 3 er koblet sammen og

ben 2 og 4 er koblet sammen





# Resistorer og knapper



Knappen er en bryter som setter to metallpinner i kontakt med hverandre når vi trykker den ned. Den er som en ledning i kretsen.

Hvis det ikke er noen andre komponenter i kretsen får vi en direkte linje fra pluss til minus og risikerer kortslutning

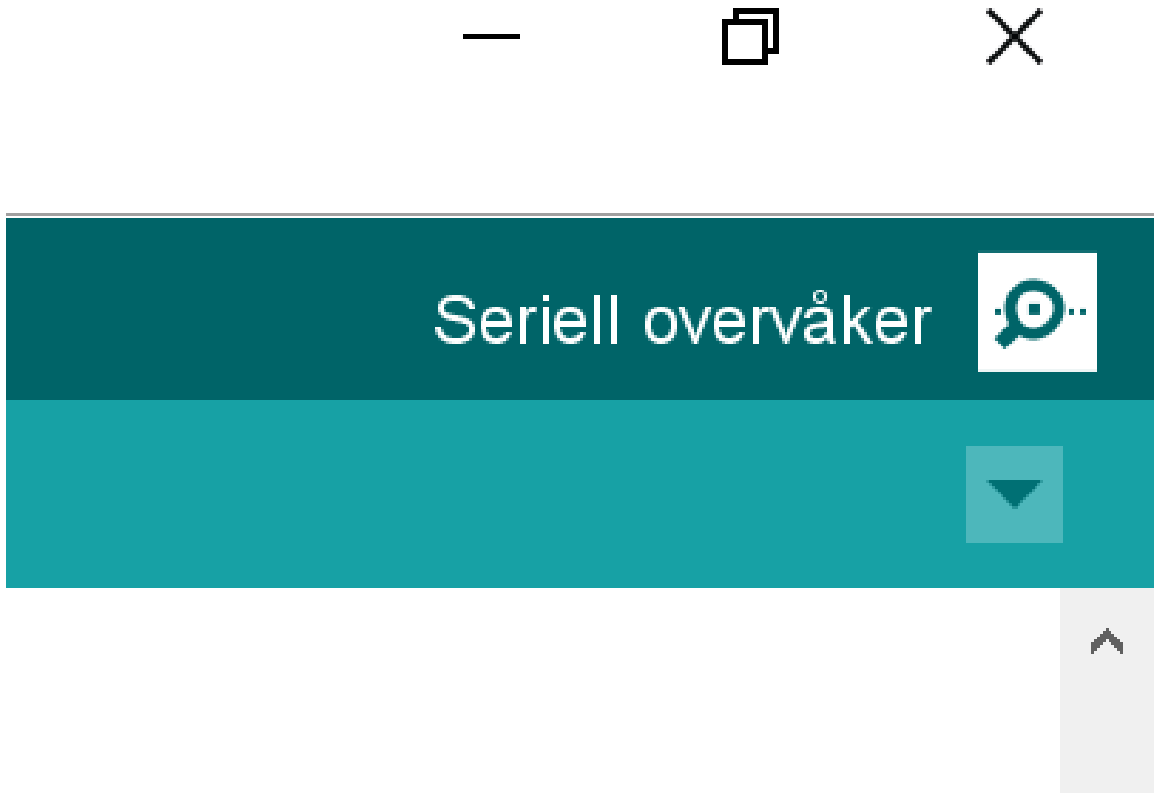
En resistor legger til ett komponent i kretsen, og motvirker en direkte krets fra pluss til minus

# Problemløsning og feilsøking

## + To vanlige problemer med knapper

# Serial: et verktøy for ~~problemløsning~~

Arduinoen har et verktøy som lar oss se signalene som går ut og inn i form av tekst




Gå til:

Verktøy – seriell overvåker eller ctrl+shift+m eller ikon øverst til høyre

Vises som popup-vindu (integrert i nye IDE 2.0)

Seriell overvåker (serial monitor)

Skriver ut returverdier fra funksjonene når programmet kjører i form av tall eller tekst

# Funksjoner for seriell kommunikasjon #1

For å starte seriell kommunikasjon må vi initialisere den med

Serial.begin(9600)

**Serial.begin()**-funksjonen tar inn en variabel for baud rate.

**Baud rate** refererer til hastigheten på kommunikasjonen over en datakanal. På Arduino er det vanlig å sette den til 9600.

**Serial.print()**- funksjonen lar oss skrive ut beskjeder til overvåkeren og er nyttig for å følge med på hva som egentlig skjer i programmet

**DigitalReadSerial eksempelskisse**

```
int pushButton = 2;

void setup() {
Serial.begin(9600); // start seriell
kommunikasjon ved 9600 bits per
sekund
pinMode(pushButton, INPUT);
}

void loop() {
int buttonState =
digitalRead(pushButton);
Serial.println(buttonState); //skriv
ut tilstanden til knappen
  delay(1); //pause mellom
avlesningene for stabilitet
}
```

# Vanlig problem med brytere #1: Sprettende input

**Switchbounce**

Når vi trykker ned knappen tar det noen millisekunder før metallbitene kommer i stabil kontakt med hverandre

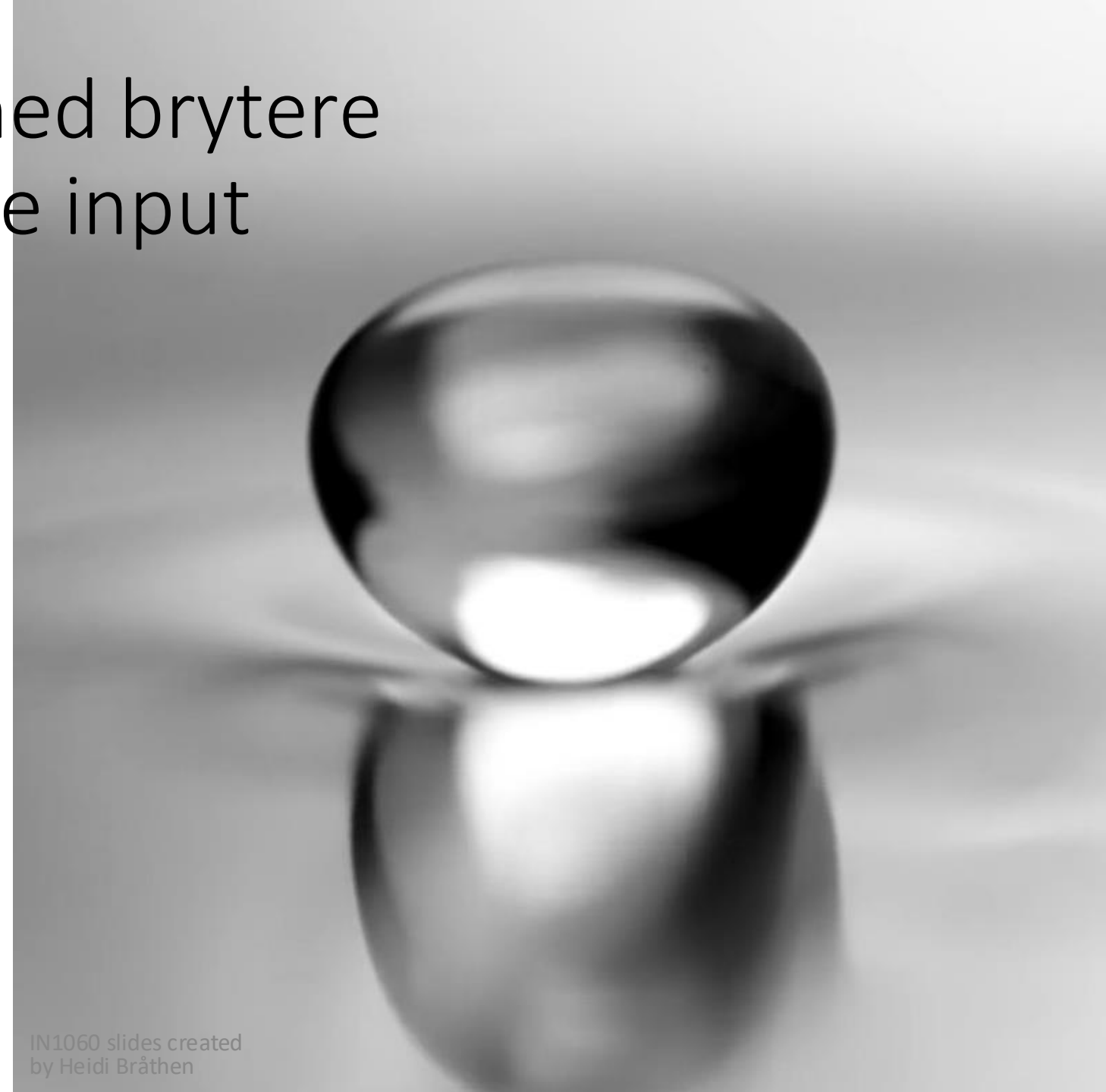
Fordi loop() kjører veldig fort klarer Arduinoen å registere flere knappetrykk idet vi trykker ned knappen

**Løsning: Debounce**

Vi kan programmere Arduinoen til å vente noen millisekunder og dermed kun registrere ett av trykkene

**Debounce ved å bruke delay()**

Delay() pauser programmet i millisekunder: delay(antall millisekunder)



# Debounce med delay()

Trykk på knappen, må du holde inn en liten stund for å få LEDen til å lyse stabilt?

Hva skjer når du trykker knappen raskt inn og slipper?

Se på seriell overvåker mens du trykker ned knappen

```
*/ Eksempelskisse: Button

const int buttonPin = 2;
const int ledPin =  13;
int buttonState = 0;

void setup() {
pinMode(ledPin, OUTPUT);
pinMode(buttonPin, INPUT);
}

void loop() {
buttonState = digitalRead(buttonPin);
if (buttonState == HIGH) {
          delay(500);
          digitalWrite(ledPin, HIGH);
          } else {
          delay(500); digitalWrite(ledPin,
          LOW);
          }
}
```

# Vanlig problem #2: Flaksende og flytende input

**«Flagrende» eller flytende input** er et vanlig problem når vi bruker brytere i Arduino-kretser. Signalet på porten knappen er koblet til settes vilkårlig til 5V/HIGH eller 0V/LOW, og det knappen skal kontrollere oppfører seg uforutsigbart eller virker ikke i det hele tatt

**Løsning: Pullup**

Arduino har en innebygget pullup i form av en innebygget konstant: INPUT_PULLUP som brukes i stedet for INPUT i pinMode().



# Pullup med resistor og innebygget PULLUP

**Pullups og pulldowns** setter signalet til høyt(-up) eller lavt(-down)

**Bruke den innebygde pull-upen:**

INPUT_PULLUP brukes i stedet for INPUT i pinMode()

**Bruke komponentet resistor som pull up:**

En resistor kan brukes som «pull up» som tilbakestiller signalet fra knappen til Arduinoen etter at den er trykket på ved å kobles mellom 5V og knappen

# Oppsummering: hva, hvorfor, hvordan

**hva** Arduino er

et verktøy for prototyping med elektronikk og programmering

*Arduino er en åpen prototypingsplattform som lar oss bygge elektroniske kretser og styre dem med enkel programmering*

en åpen mikrokontroller-plattform som vi bruker for å sanse input-signaler og sende output-signaler

**hvorfor** vi velger Arduino

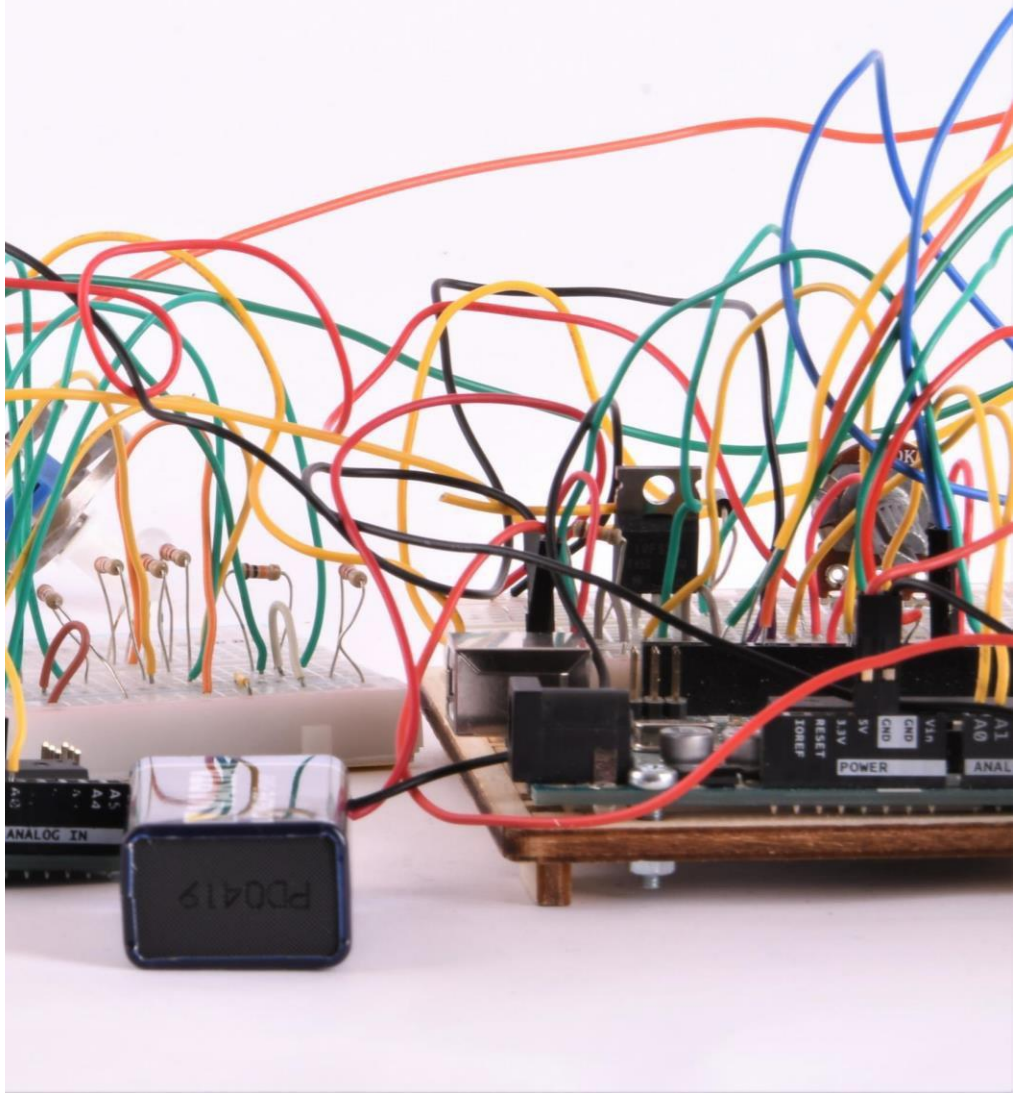
utforske andre designløsninger enn skjerm

åpent, krever lite forkunnskaper, aktivt community av brukere = mye tilgjengelige ressurser

**hvordan** vi skal bruke Arduino og hvilke ferdigheter vi trenger å lære

brukerundersøkelser og systemkrav først, deretter utforske designløsninger med prototyping

kjerne-ferdigheter er programmering – bygge elektriske kretser – problemløsning og feilsøking



# Oppsummering: strøm

**Strøm:**

Elektroner

Spenning

Strømstyrke

Motstand

Ohms lov

Kretser og kretsdiagrammer/koblingsskjema

Sikkerhet

Kortslutning

**Komponenter:**

Resistorer

LED

Knapper

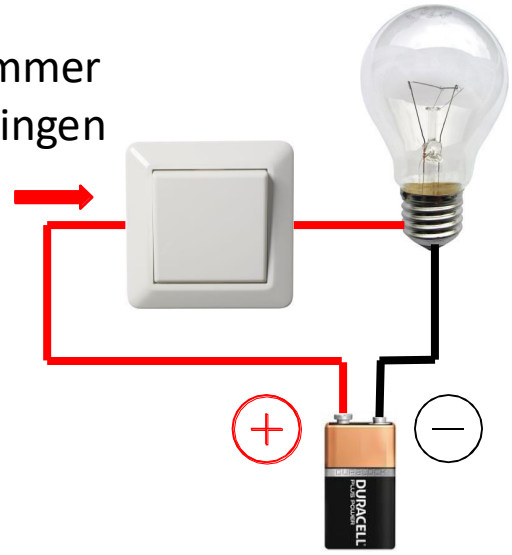
Ledninger

Breadboard

**2 vanlige problemer med brytere med løsninger**

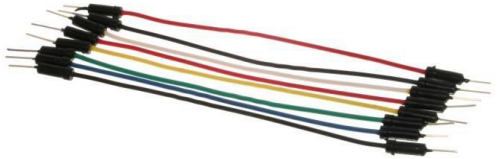
Debounce Pullup

I/amp: Ladde partikler strømmer gjennom ledningen

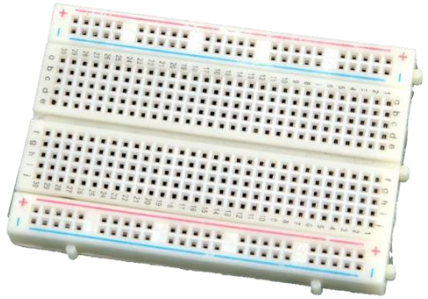



R/ohm: Tråden i lyspærer har så høy motstand at den gløder (R)

V/volt: Batterier er spenningskilder som «dytter» partikler gjennom kretsen











# Til neste forelesning

**Ukesoppgave 1 og Oblig 1 (merk frist 12 mars)**
Den første ukesoppgaven og obligen er /vil bli publisert på semestersidene, under Arduino i menyen til venstre

Lag gjerne en bruker på tinkercad.com (gratis og frivillig), neste gang skal vi se på hvordan vi kan tegne kretsdiagrammer. Tinkercad er veldig enkelt i bruk. Andre muligheter er Fritzing og EagleCad..

**Fellesøvelse/orakeltime i morgen 10.15-12.00:**

- Spørsmål om ukesoppgaver og oblig 1
- Knut og Mali gjennomgår oppsett og relevante eksempler
- Pusle med Arduino-oppgaver sammen med andre

# Kreditering av bilder i presentasjonen #1

**Creative commons-lisenser**

MikeRun, CC BY-SA 4.0 <https://creativecommons.org/licenses/by-sa/4.0>, via Wikimedia Commons

Breadboard schematics

en:User:Waveguy, CC BY-SA 3.0 <http://creativecommons.org/licenses/by-sa/3.0/>, via Wikimedia Commons

Breadboard og Arduino I Tinkercad

me, CC BY-SA 2.5 <https://creativecommons.org/licenses/by-sa/2.5>, via Wikimedia Commons

No breadboard

Mattruffoni, CC BY-SA 4.0 <https://creativecommons.org/licenses/by-sa/4.0>, via Wikimedia Commons

Brødrister

Pymouss, CC BY-SA 4.0 <https://creativecommons.org/licenses/by-sa/4.0>, via Wikimedia Commons

5 pushbuttons

oomlout, CC BY-SA 2.0 <https://creativecommons.org/licenses/by-sa/2.0>, via Wikimedia Commons

LED oversikt

Original: Adam850, Public domain, via Wikimedia Commons Breadboard-bilde

MAbW, CC BY-SA 4.0 <https://creativecommons.org/licenses/by-sa/4.0>, via Wikimedia Commons

Basic led

KDS4444, CC BY-SA 4.0 <https://creativecommons.org/licenses/by-sa/4.0>, via Wikimedia Commons

Resistor 220 ohm

Jumper wires

oomlout, CC BY-SA 2.0 <https://creativecommons.org/licenses/by-sa/2.0>, via Wikimedia Commons

Water circuit

MikeRun, CC BY-SA 4.0 <https://creativecommons.org/licenses/by-sa/4.0>, via Wikimedia Commons

Ohms law with voltage xource

GorillaWarfare, CC0, via Wikimedia Commons

Ohms Law triangle (må attribueres)

By MikeRun - Own work, using File:Hand-1028272.svg, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=88083579

Volt ohm og amp

Christinelmiller, CC BY-SA 4.0 <https://creativecommons.org/licenses/by-sa/4.0>, via Wikimedia Commons

10k resistor

© Nevit Dilmen, CC BY-SA 3.0 <https://creativecommons.org/licenses/by-sa/3.0>, via Wikimedia Commons

# Kreditering av bilder i presentasjonen #2

Switch kretssymbol

Esko Ranta & Matti Tiilikainen, CC BY-SA 4.0 <https://creativecommons.org/licenses/by-sa/4.0>, via Wikimedia Commons

Arduino Uno 006

oomlout, CC BY-SA 2.0 <https://creativecommons.org/licenses/by-sa/2.0>, via Wikimedia Commons

elv strøm

Avish18, CC BY-SA 4.0 <https://creativecommons.org/licenses/by-sa/4.0>, via Wikimedia Commons

demning

The Oaked Ridge, CC BY-SA 4.0 <https://creativecommons.org/licenses/by-sa/4.0>, via Wikimedia Commons

Vannpumpe I åker

ThomasLendt, CC BY-SA 4.0 <https://creativecommons.org/licenses/by-sa/4.0>, via Wikimedia Commons Vannpumpe som

pumper vann

Surajmondol, CC BY-SA 4.0 <https://creativecommons.org/licenses/by-sa/4.0>, via Wikimedia Commons Bouncing droplets

A7N8X, CC BY-SA 4.0 <https://creativecommons.org/licenses/by-sa/4.0>, via Wikimedia Commons

Flytende frø

PiccoloNamek at English Wikipedia, CC BY-SA 3.0 <http://creativecommons.org/licenses/by-sa/3.0/>, via Wikimedia Common

**Øvrige**

Tittelbilde: Heidi Bråthen