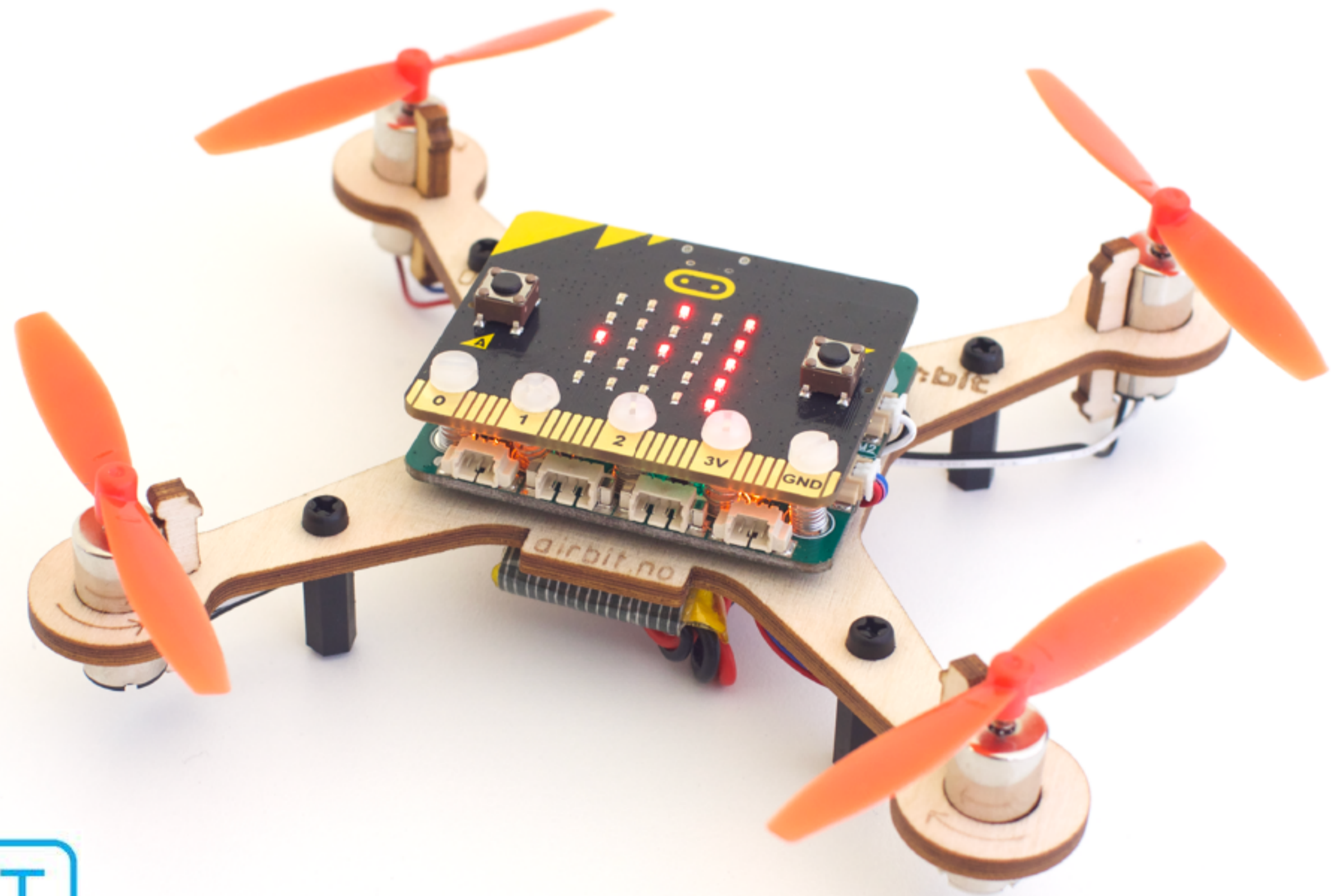


# Air:bit

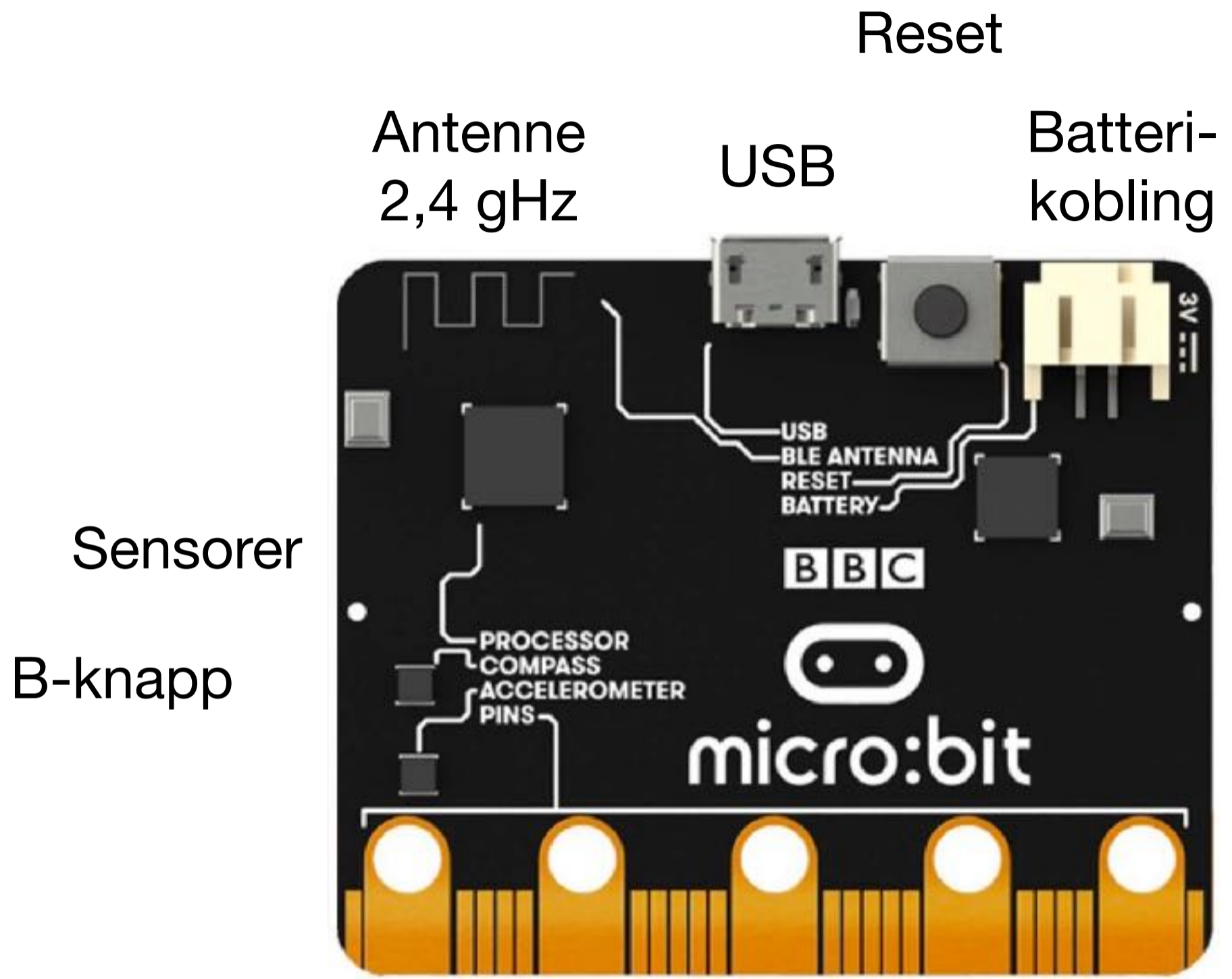
Verdens første micro:bit drone



# Hva er micro:bit?



**Front**



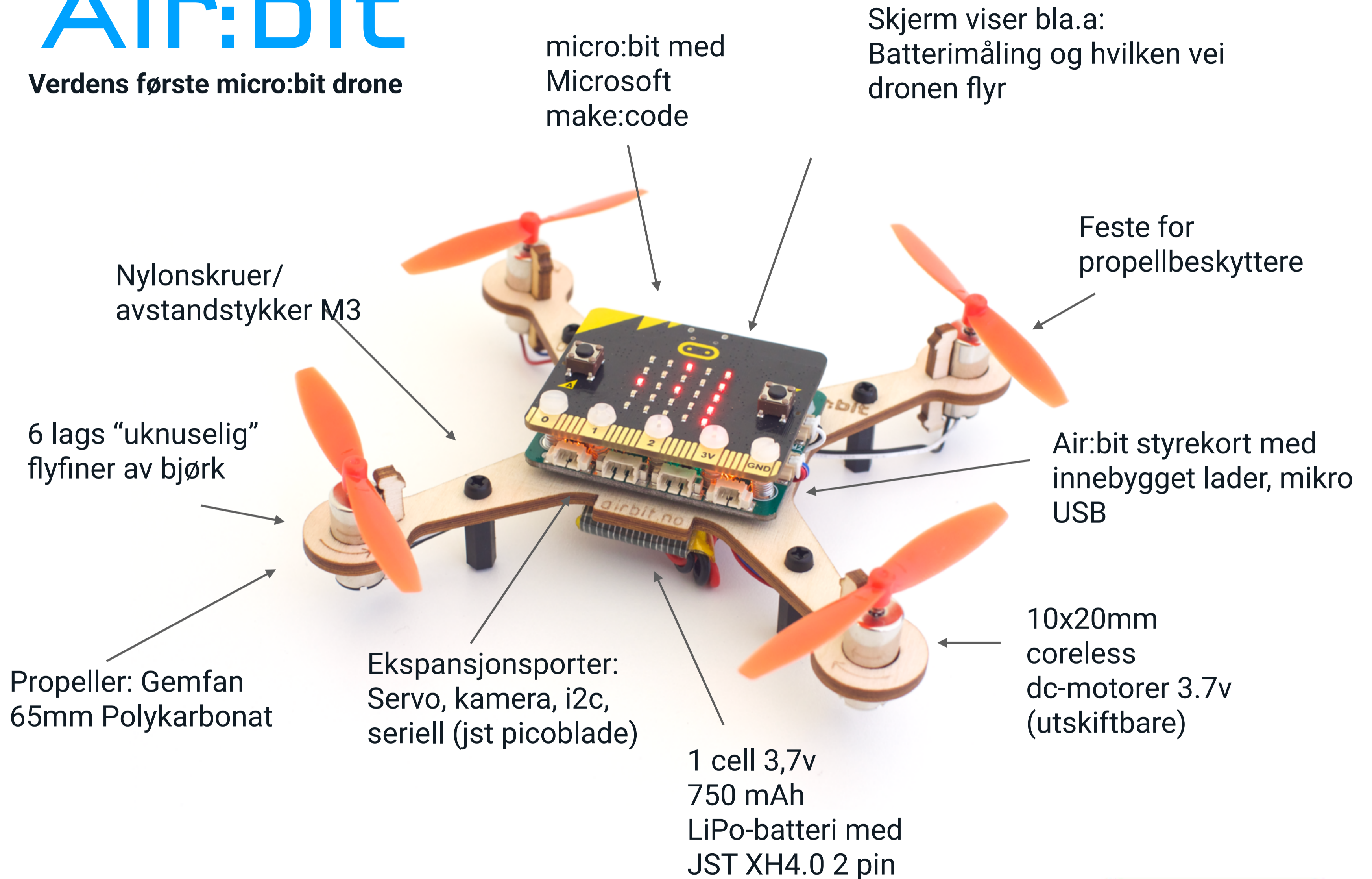
**Bakside**

micro:bit er en liten datamaskin med prosessor, sensorer, display og radio. Den har tilkoblinger hvor man kan koble til alt fra LED til sensorer og høyttalere.

Mer detaljert beskrivelse finner du her: <https://tech.microbit.org/hardware/>

# Air:bit

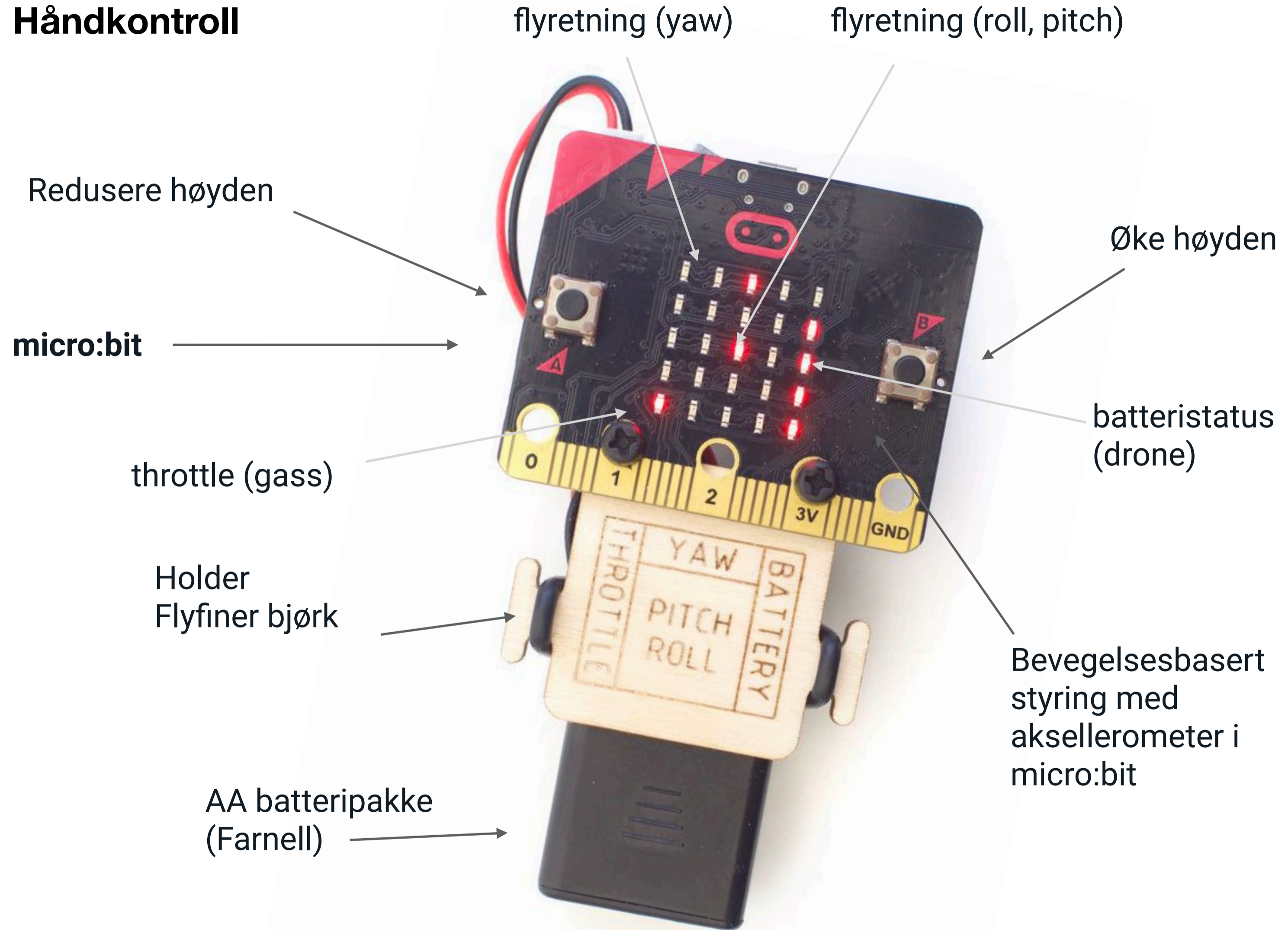
Verdens første micro:bit drone



Mer info: Besøk [airbit.no](http://airbit.no)



# Håndkontroll



# Styrekort

Utviklet av MakeKit og  
SPRacing

ST XH4.0  
batteriplugg

mikro USB:  
lading og firmware

Software debug

Hovedbryter

Motorplugger  
(Picoblade)

Servoplugg

Motorplugger  
(Picoblade)

Servoplugg

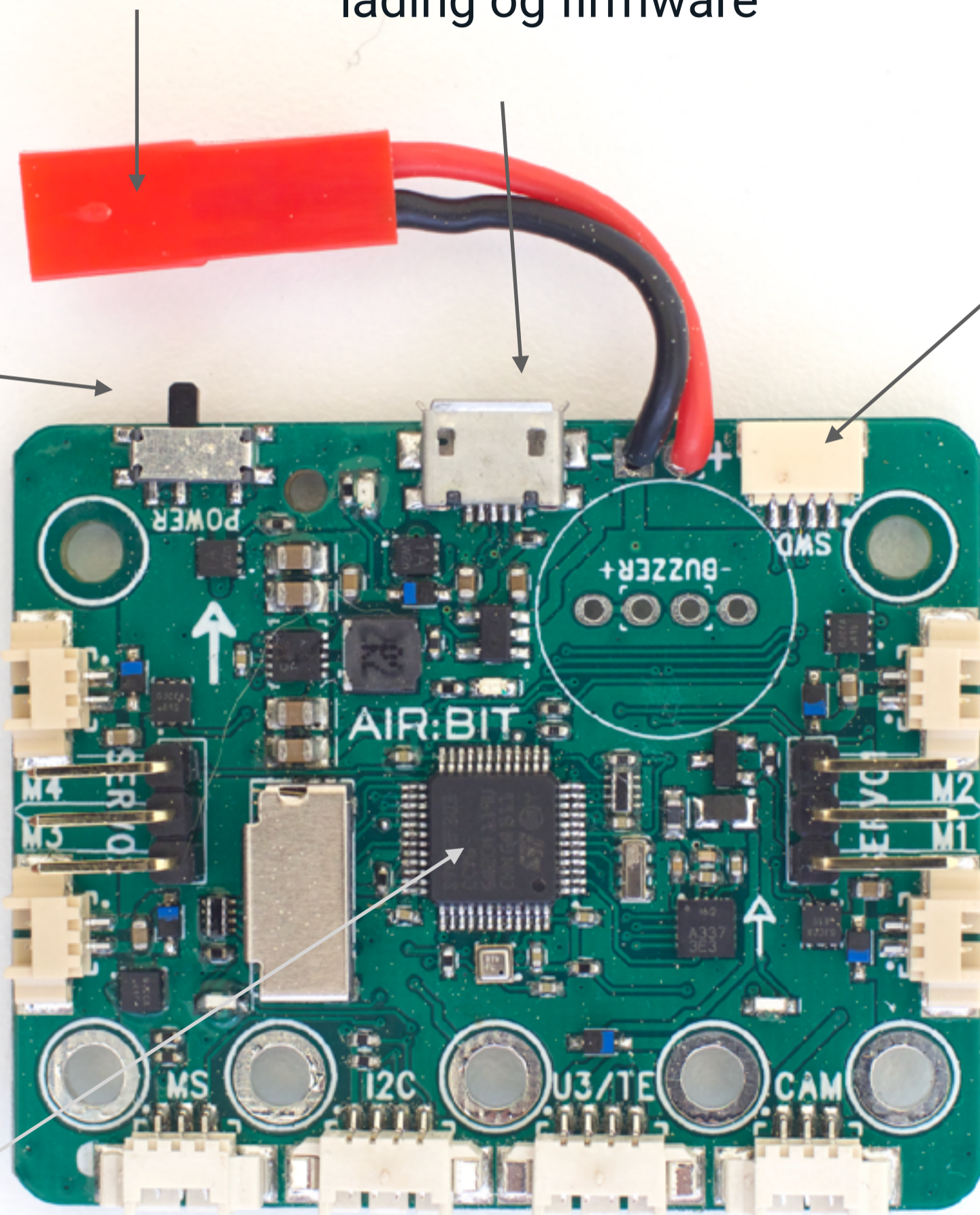
STM32 F303  
prosessor

Micro  
servo

i2c

seriell

kamera



# Deleliste: Med to micro:bit



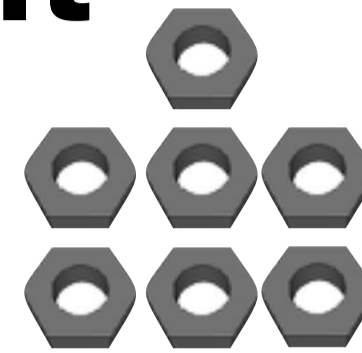
6 stk  
nylonskruer  
M3x8



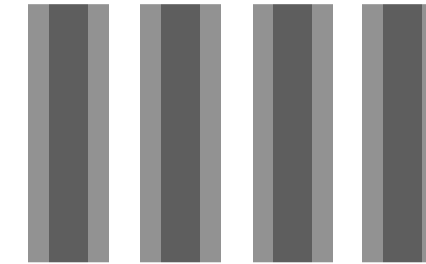
5 stk  
nylonskruer  
M3x12



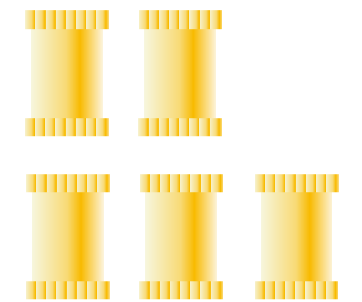
2 stk  
nylonskruer  
forsenket



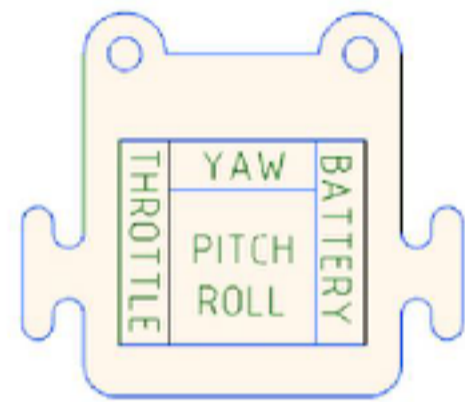
8 stk  
nylonmuttere  
m3



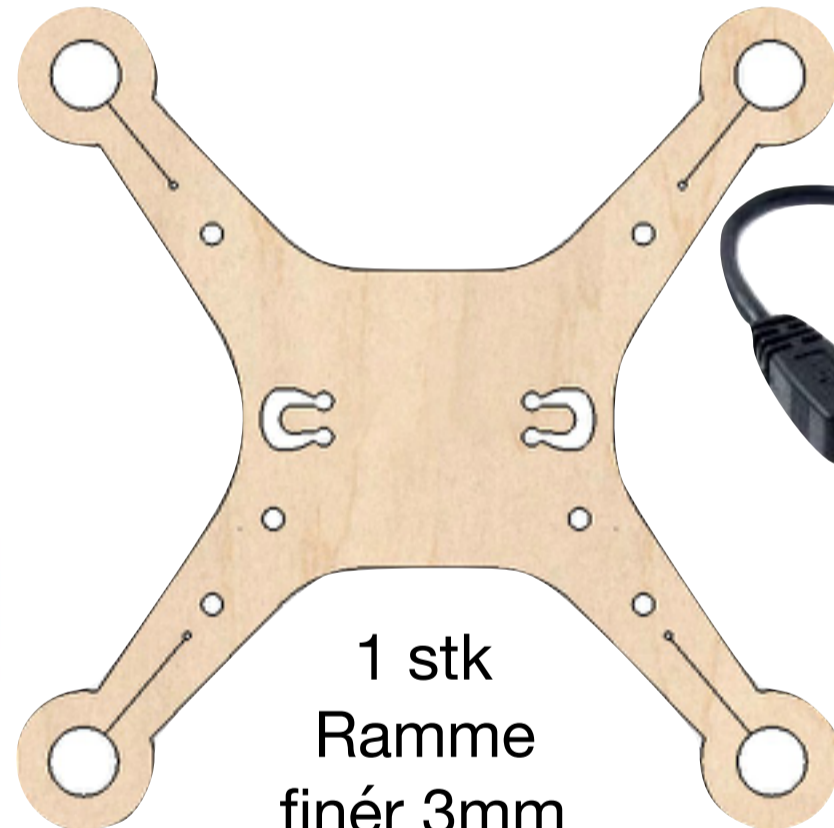
4 stk  
avstandstykker  
m3x20



5 stk  
ruglete  
tønnemuttere



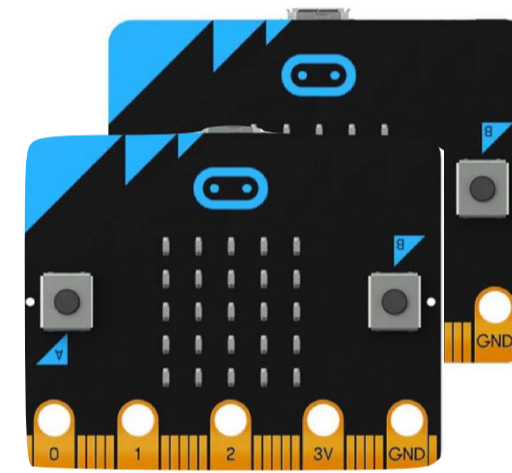
fjernkontroll-  
holder



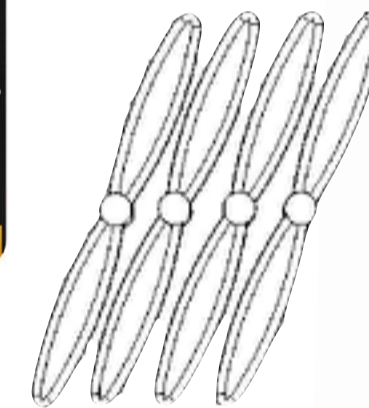
1 stk  
Ramme  
finér 3mm



1 micro usb  
kabel



2 micro:bit



4 stk  
propeller CW/CCW



2 stk CW (rød/blå)  
2 stk CCW (hvit/sort)  
motorer



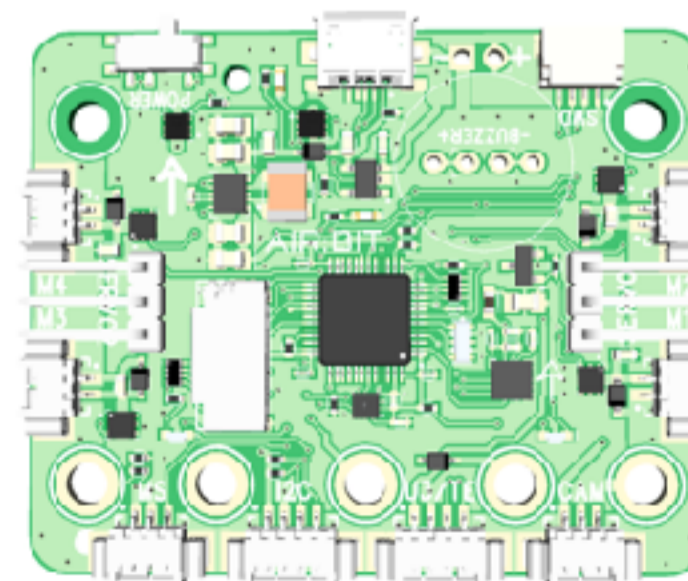
1 stk  
batteriboks med batteri



1 stk  
stor gummiring



1 stk Lithium  
(LiPo)-batteri



1 stk  
Styrekort



1 stk  
strikk



4 stk spiler

# Verktøy (pr. drone)



Pipenøkkel 5.5mm



Stjernetrekker philips

# Byggeveiledning



# Montering av fjernkontroll

Verktøy: Mellomstor stjernetrekker

## Deler:



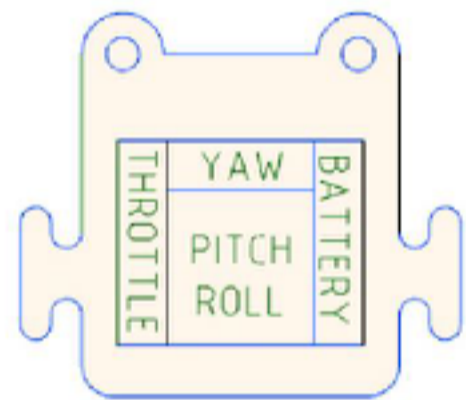
2 stk

nylonskruer m3x8



2 stk

nylonmuttere m3

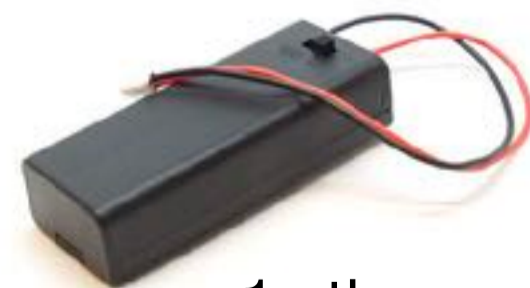


fjernkontroll-  
holder



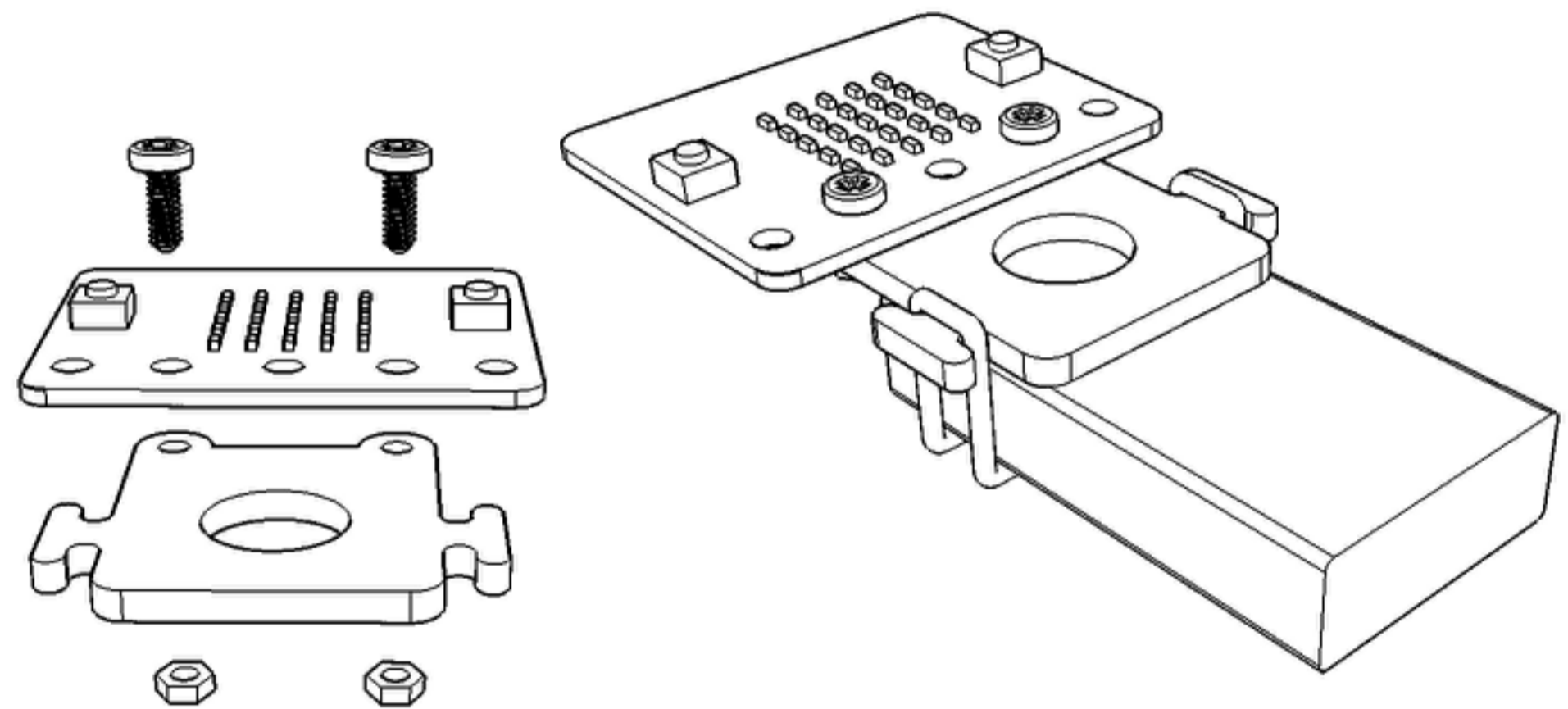
1 stk

stor gummiring



1 stk

batteriboks AAA (evt AA)



- Skru fast micro:bit til holderen med de lange skruene (1)
- Montér batteriboksen med gummiringen (2)
- Ulike batteribokser kan brukes

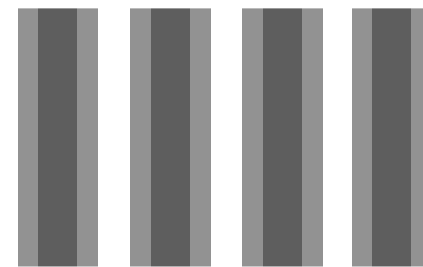
# Montering av ramme

Verktøy: Mellomstor stjernetrekker

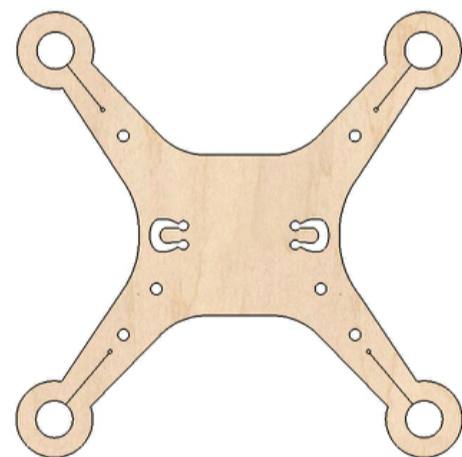
## Deler:



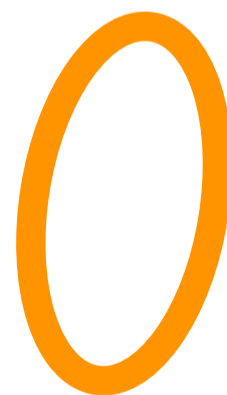
4 stk nygonskruer m3x8



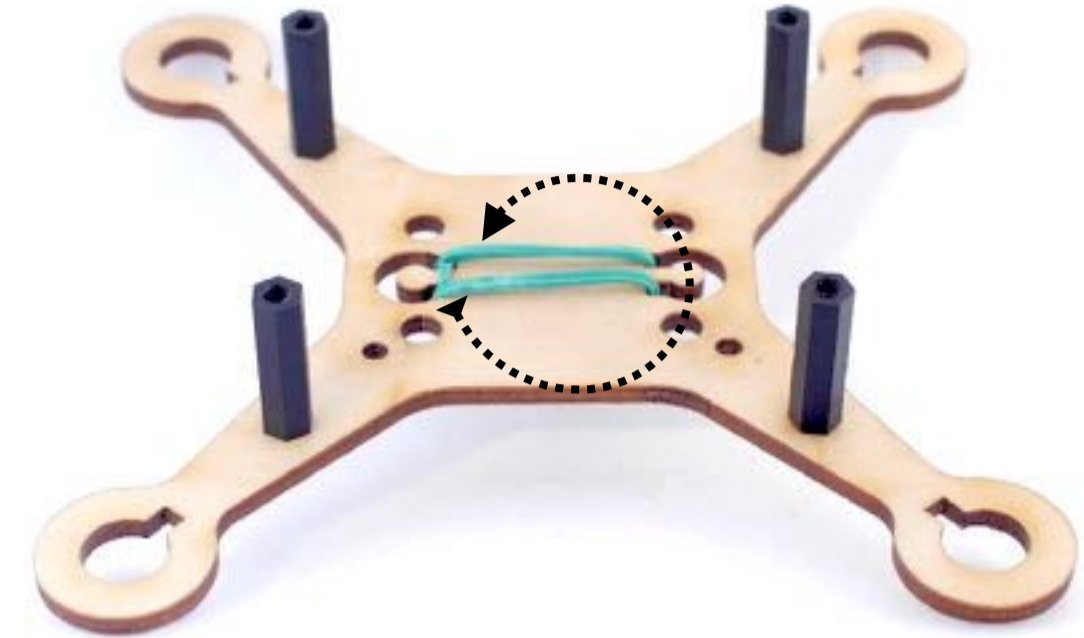
4 stk  
avstandstykker  
m3x20



Senterramme

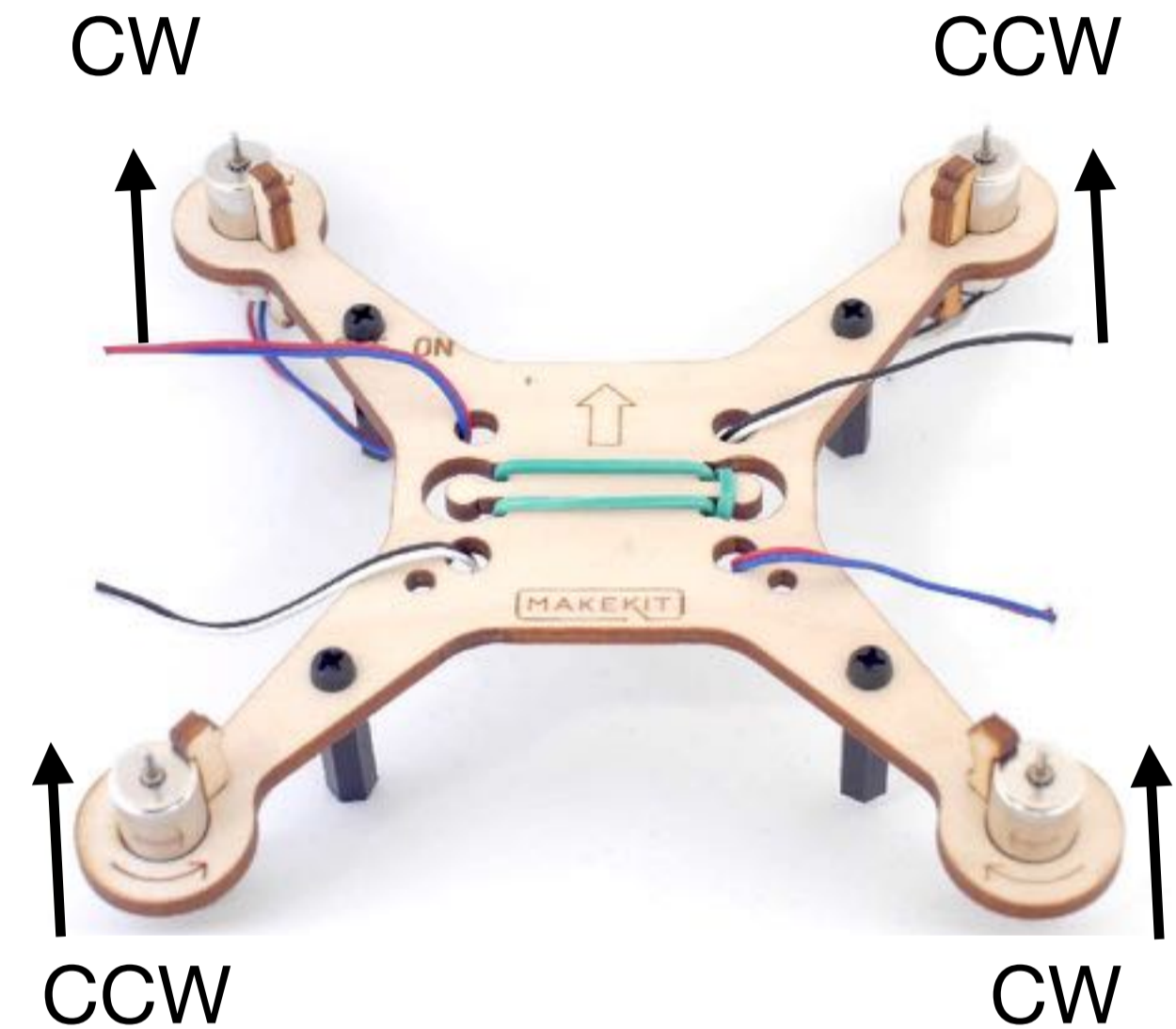
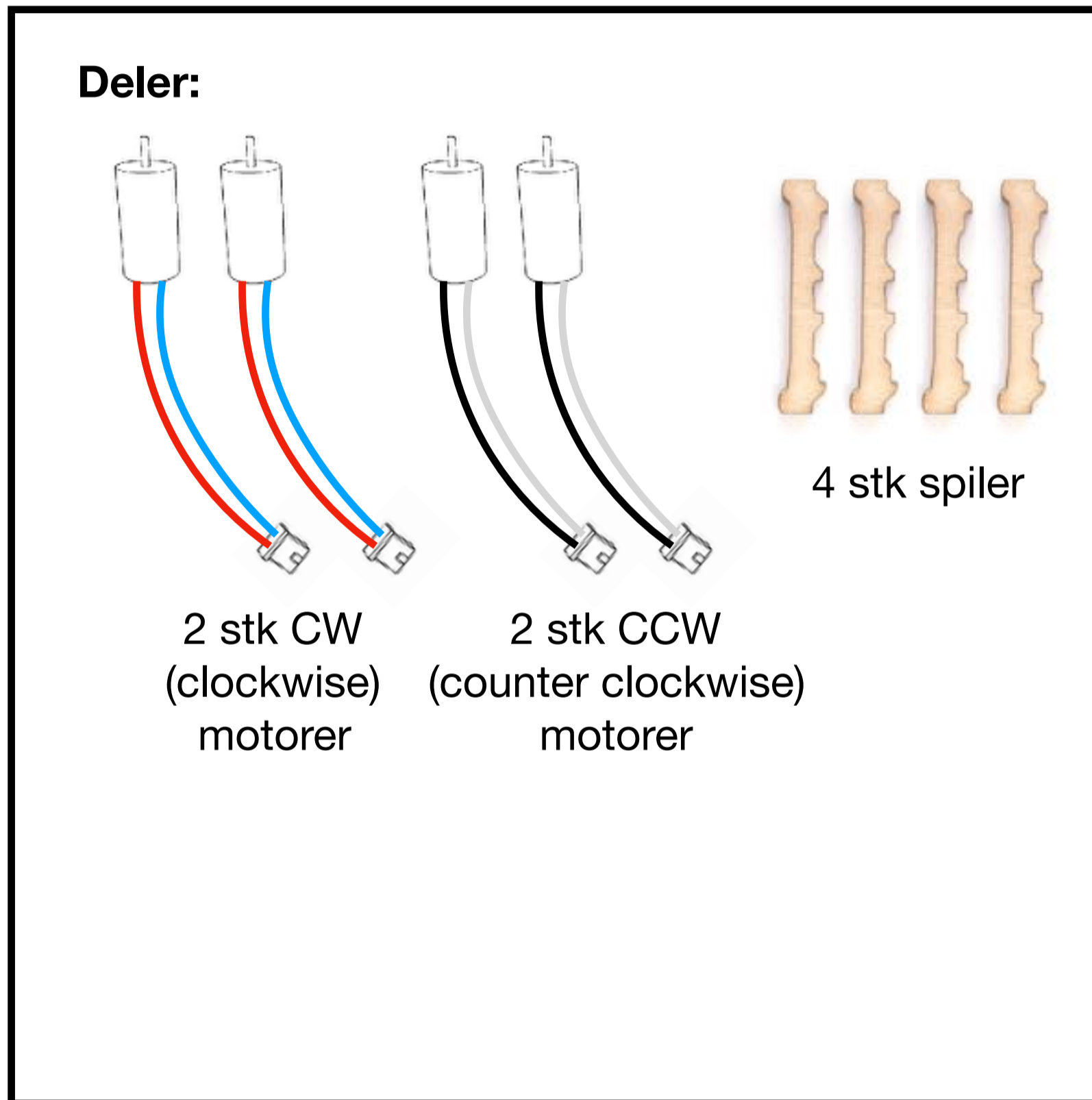


Strikk



- Skru fast avstandstykkene med skruene som vist på bildene
- Montér strikken fra den høyre kroken, over platen, ned forbi den venstre kroken, under platen, tilbake til den høyre kroken
- Strikken vil da

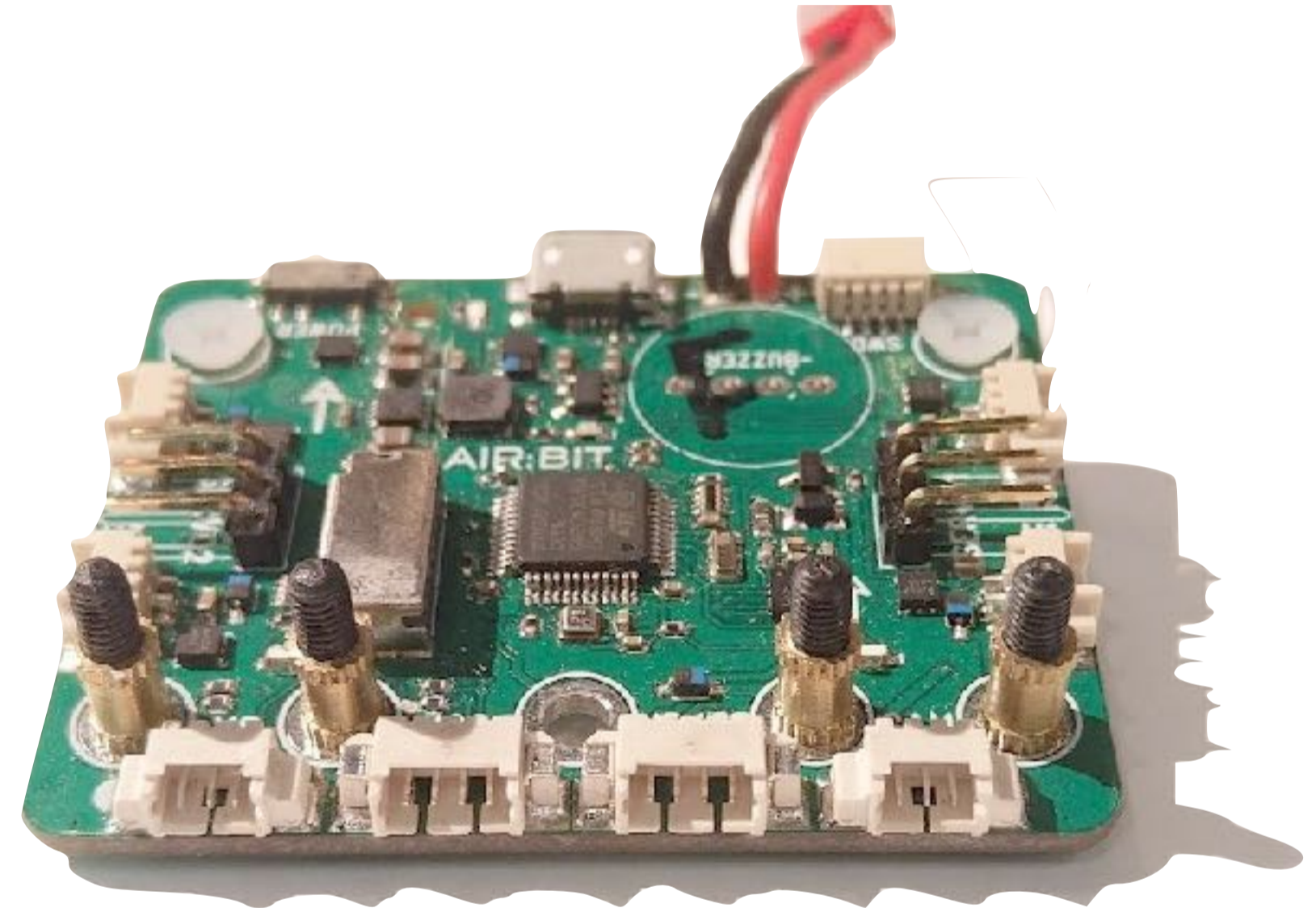
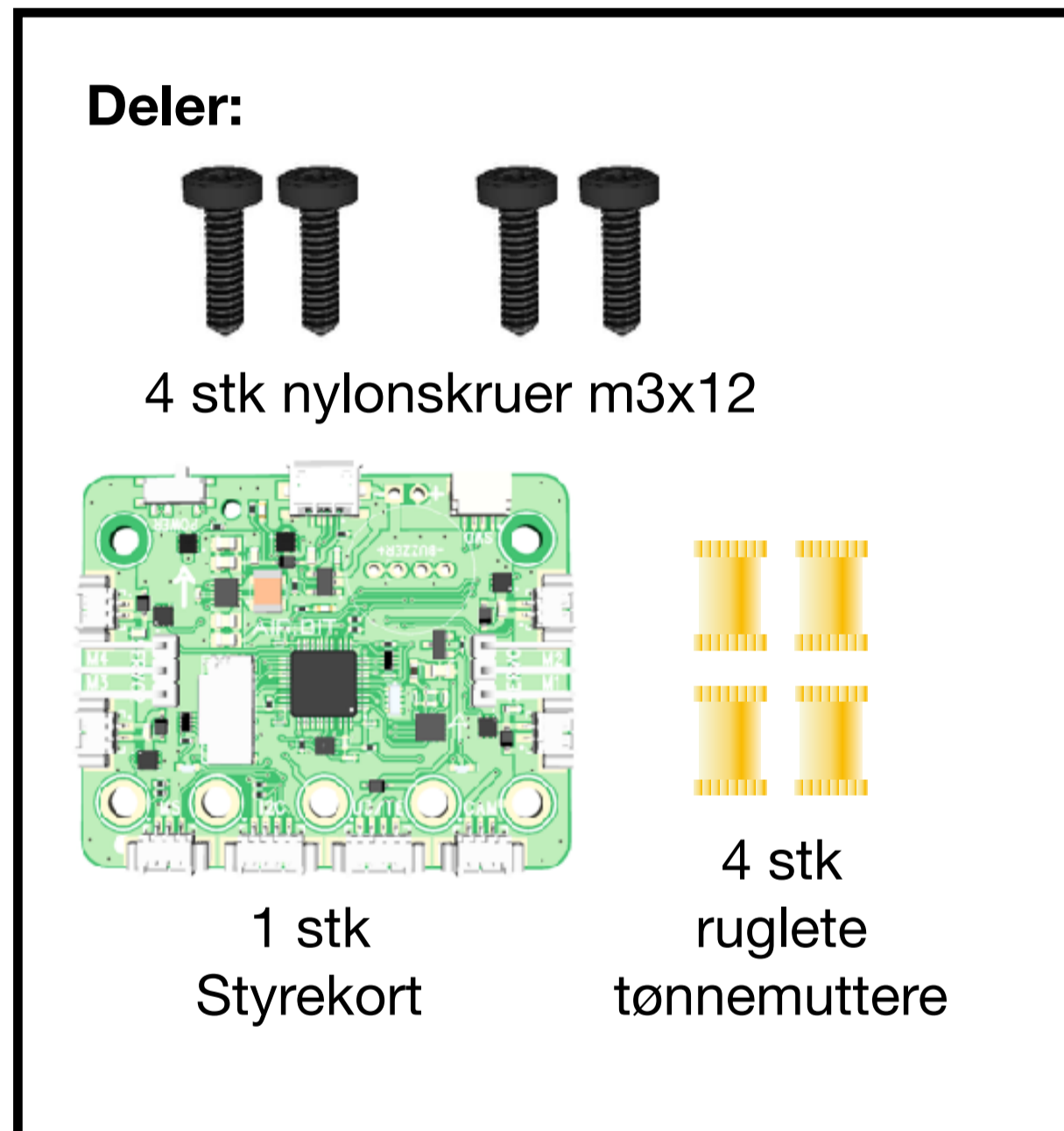
# Montering av motorer



- Motorene (CW og CCW) skal plasseres slik bildet viser, de føres oppover med spissen (ankeret) først og ledningene forblir på undersiden av dronen.
- Montér motorene på rammen med hver sin spile: Først settes spilen inn, deretter klikkes motoren på plass
- Trekk motorkablene opp gjennom hullene

# Styrekort og tønnemuttere

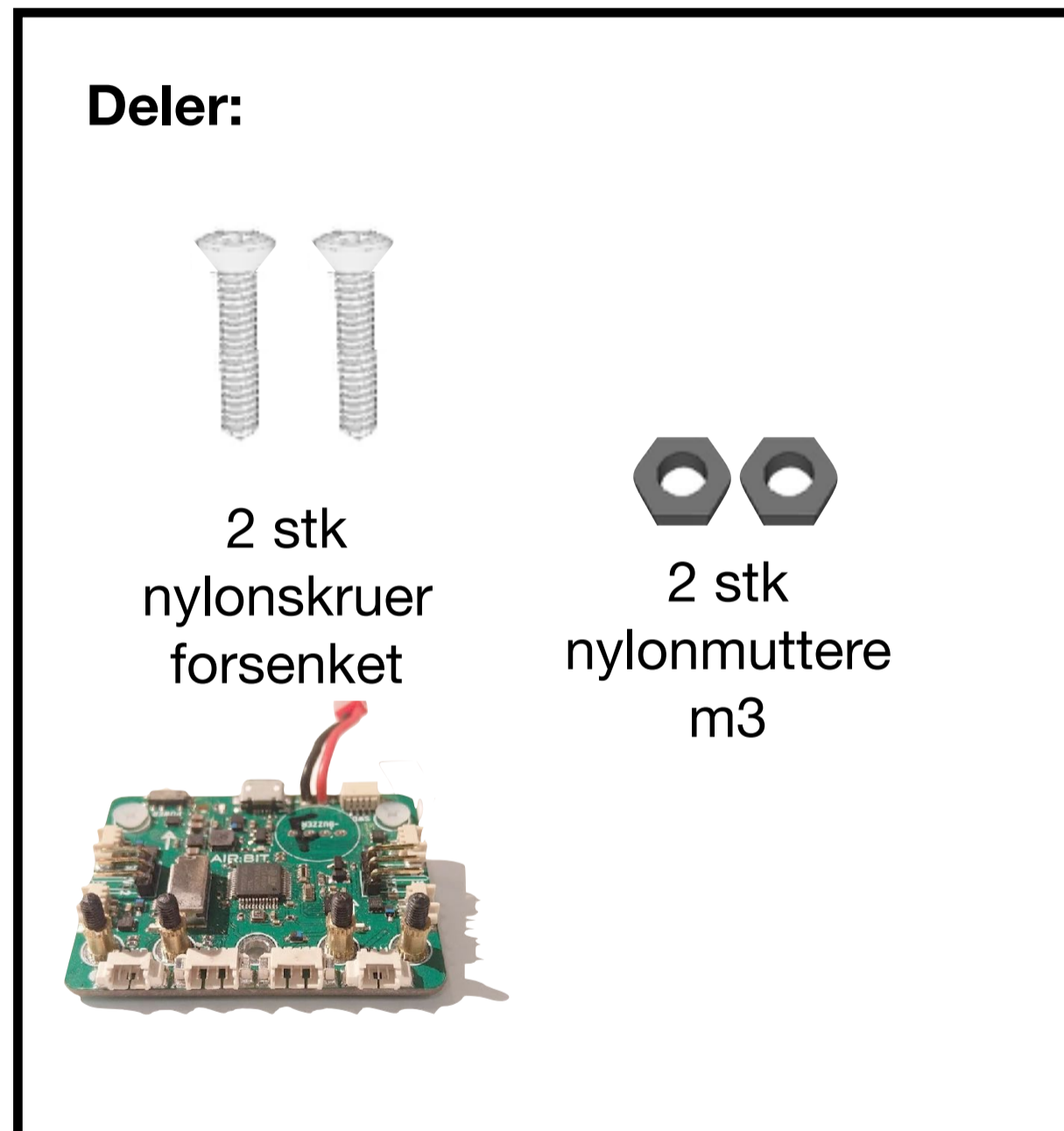
Verktøy: Mellomstor stjernetrekker



- Skru fast tønnemutterene med nylonskruene. De skal skrus passe hardt fast:
- Hardt nok til at det blir strømførende kontakt med styrekortet,
- Ikke så hardt at skruen eller styrekortet skades.

# Forsenkede skruer

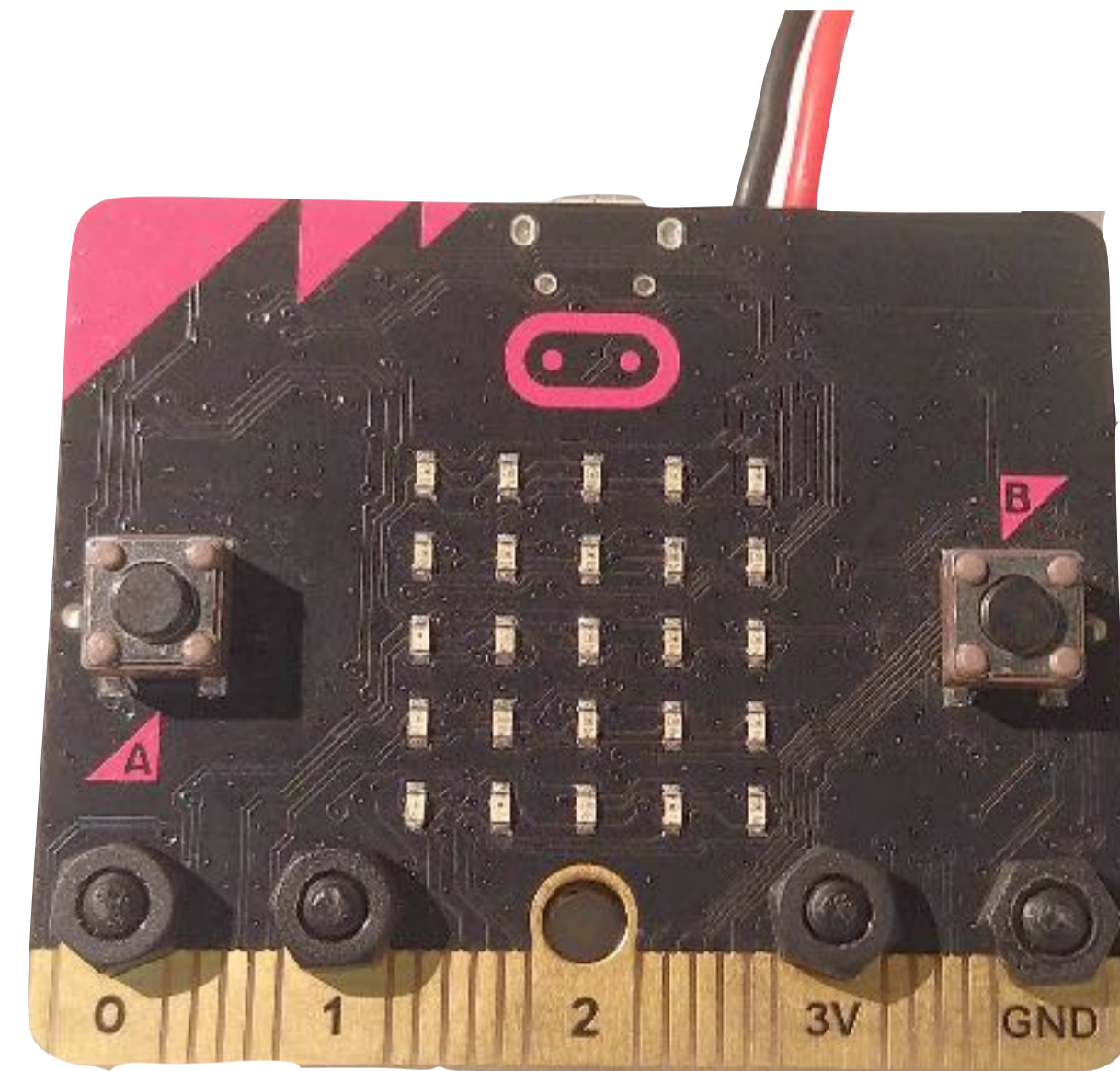
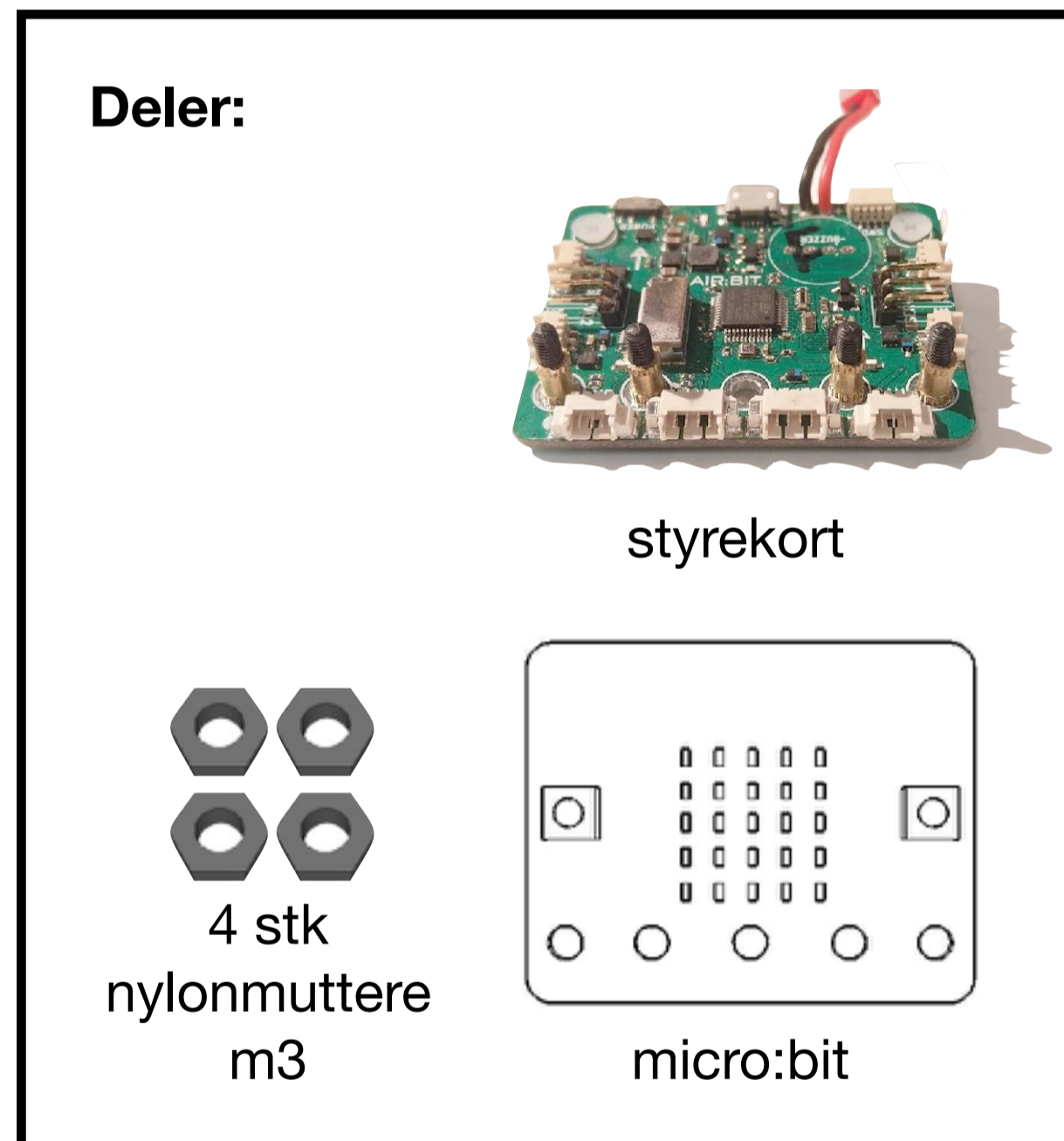
Verktøy: Mellomstor stjernetrekker



- Fest nylonskruene med muttere som vist på bildet.

# Styrekort og micro:bit

Verktøy: Mellomstor stjernetrekker, pipenøkkel 5.5mm



- Plasser styrekortet på skruene som vist på bildet.
- Skru fast en mutter på hver skru
- Hardt nok til at det blir strømførende kontakt
- Ikke så hardt at mutter eller styrekort skades.

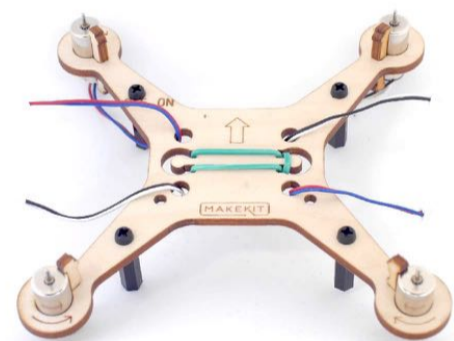
# Montere styrekort og motorer

Verktøy: Pipenøkkel 5.5mm

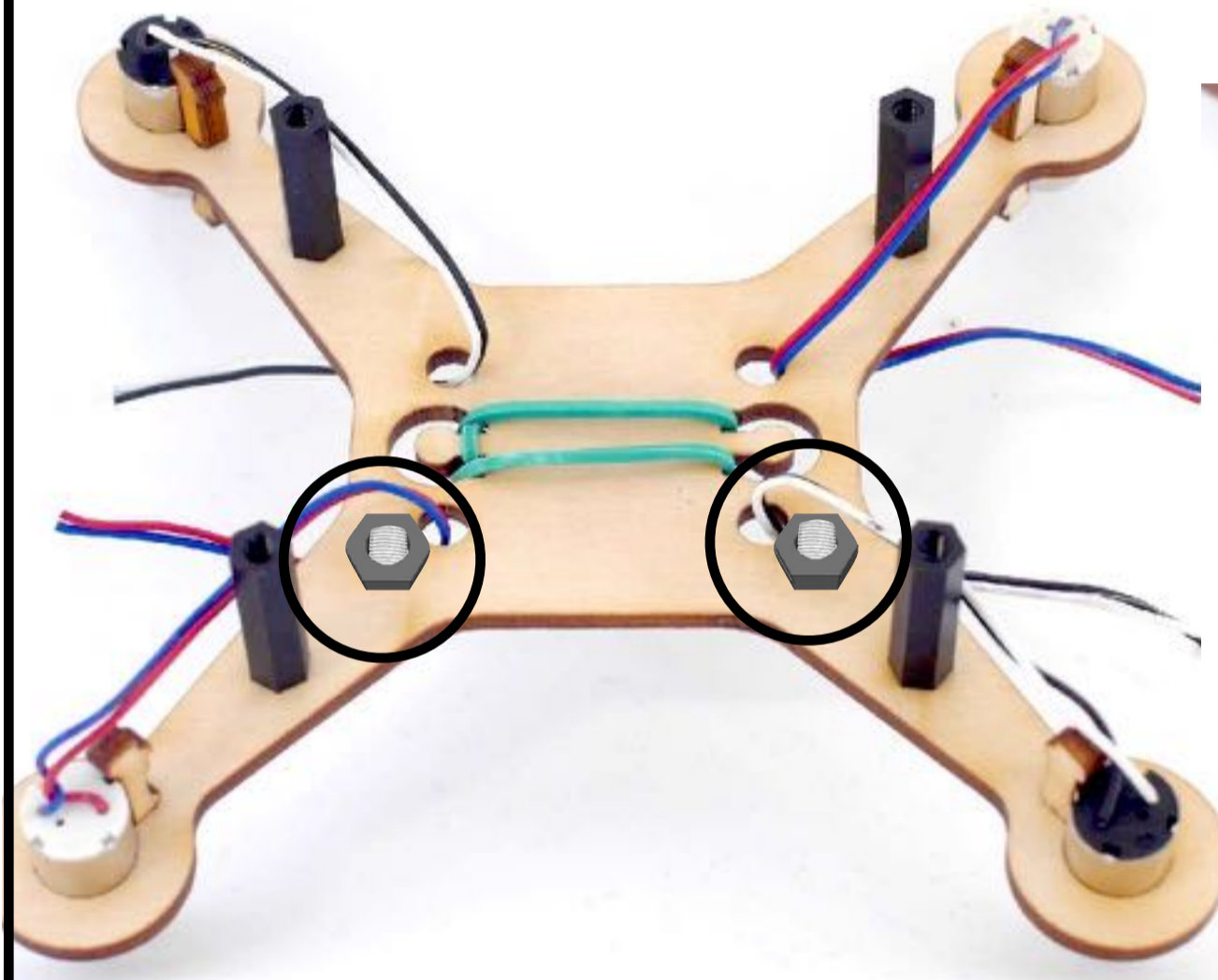
Deler:



2 stk  
nylonmuttere  
m3



Ramme med  
motorer



- Skru fast styrekort med to nylonmuttere på undersiden



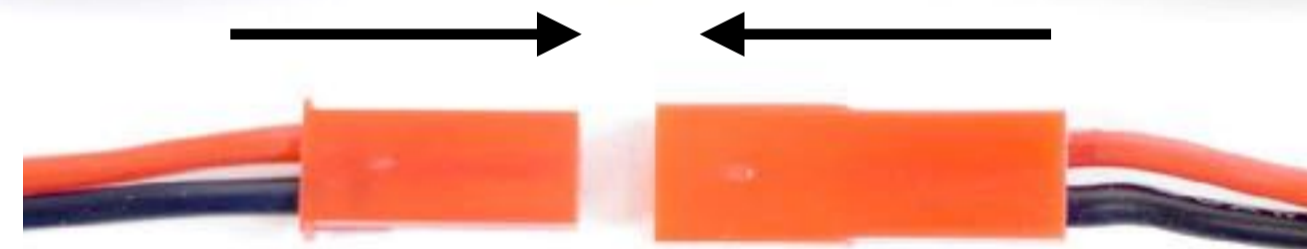
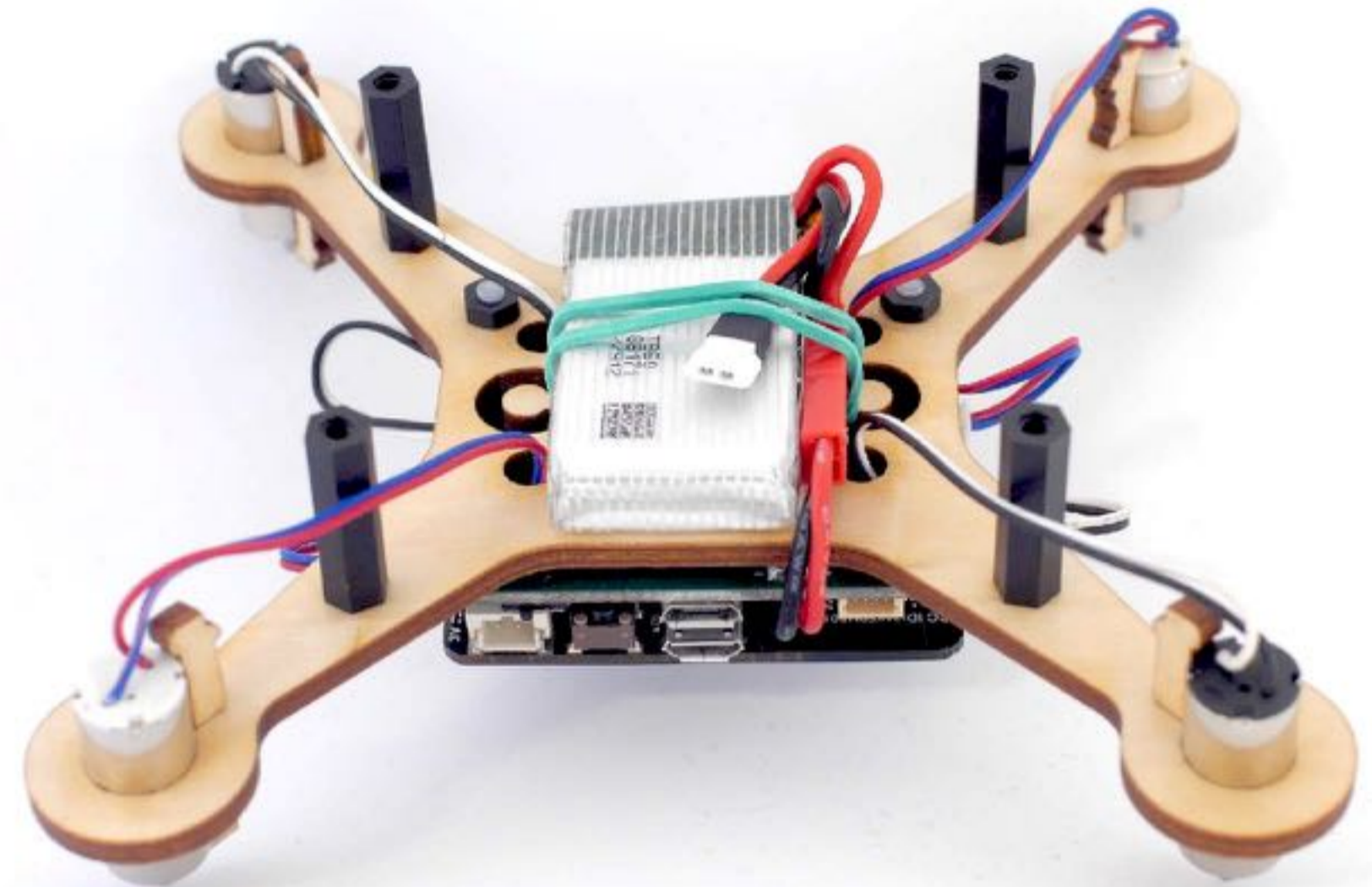
- Snu quadkopteret rett vei
- Plugg i de fire motorene på siden av kortet

# Batteri

Deler:



1 stk Lithium  
(LiPo)-batteri



Rett



Feil

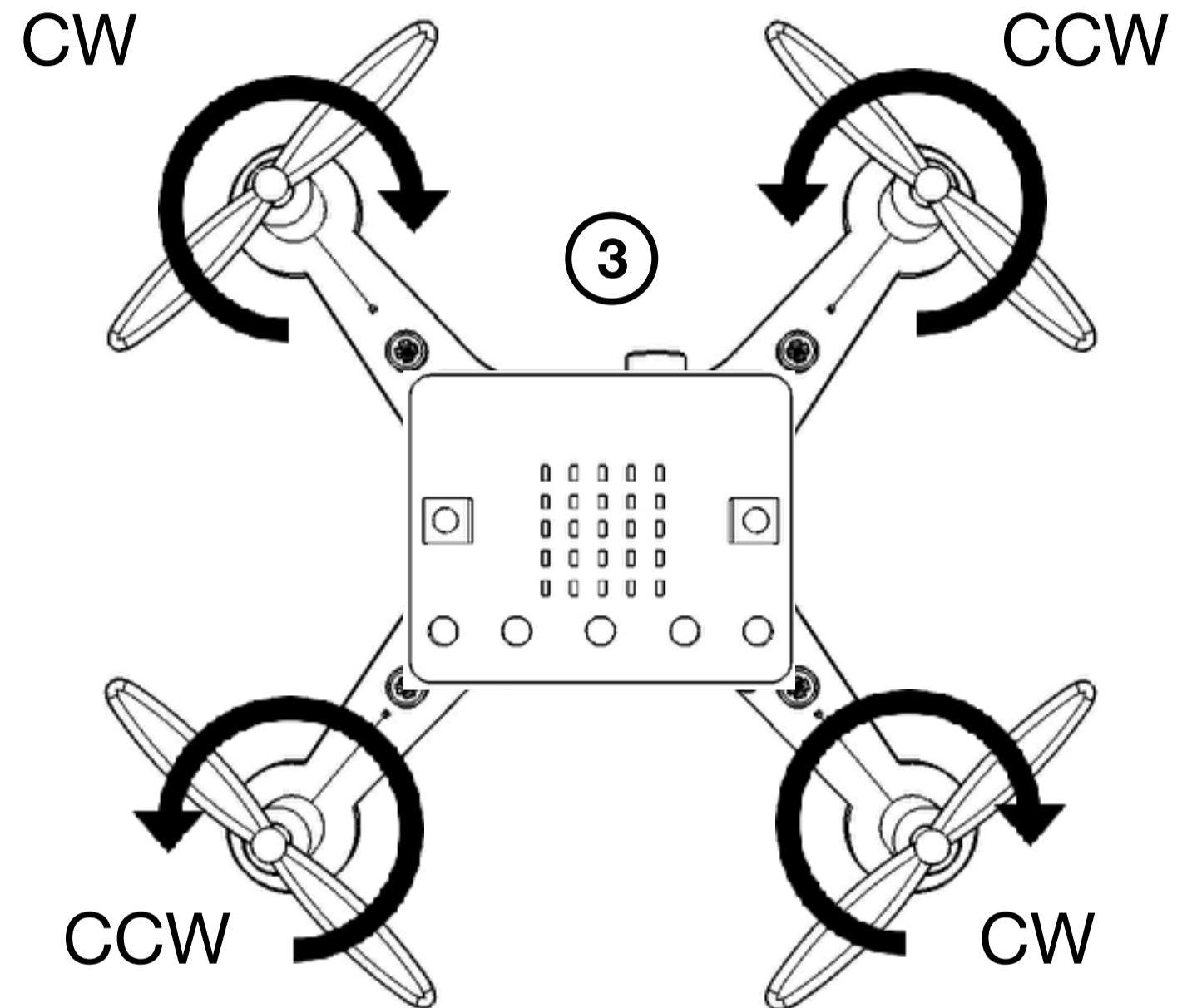
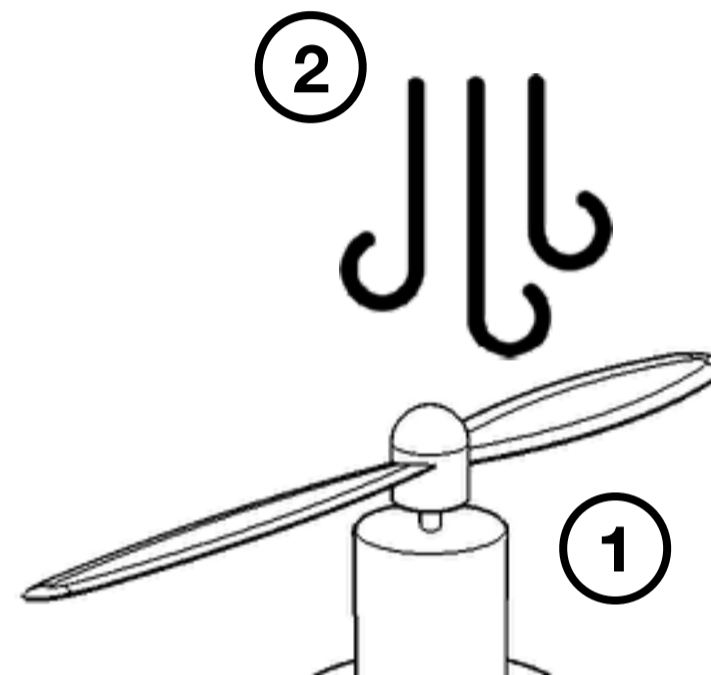
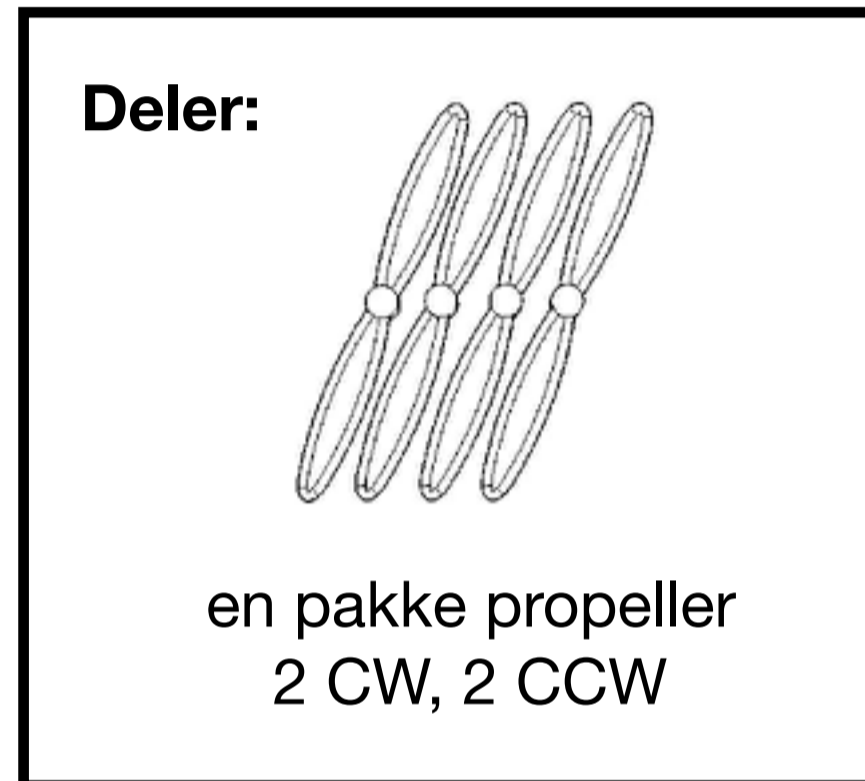


- Fest batteriet under stikken
- OBS! Rød kabel til rød kabel, sort til sort.





# Propeller



- Det finnes det 2 CW og 2 CCW propeller
- Finn fram den første propellen og plasser den på et bord eller på en motor (1)
- Blås nedover propellen (2)
- Om den roterer **med klokka**, er det en **CW** propell.
- Om den roterer **mot klokka**, er det en **CCW** propell.
- Plassér alle fire propellene korrekt i henhold til oversikten (3)

Feb. 16, 1926.

G. DE BOTHEZAT

1,573,228

HELICOPTER

Filed June 27, 1923

3 Sheets-Sheet 1

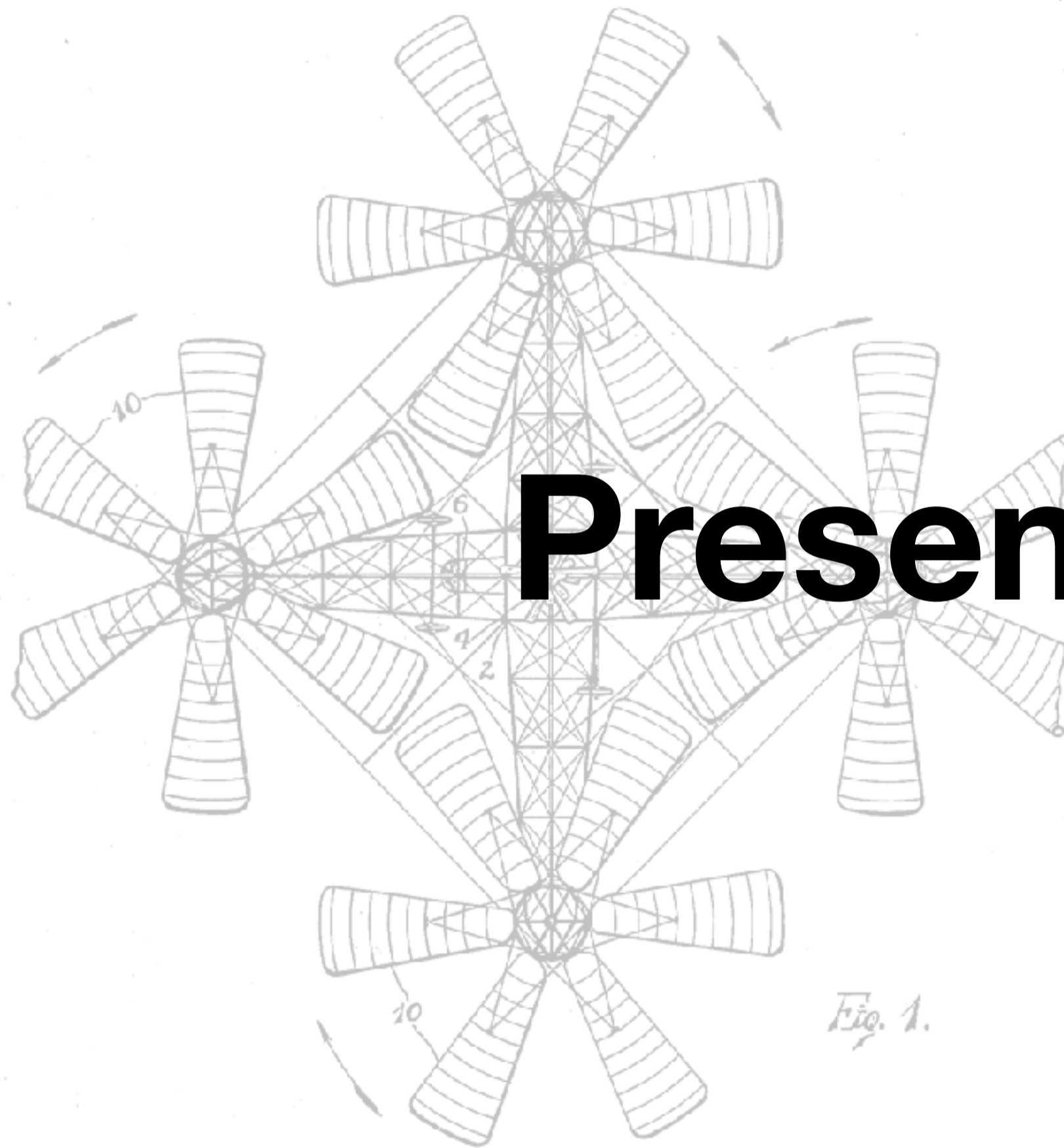


Fig. 1.

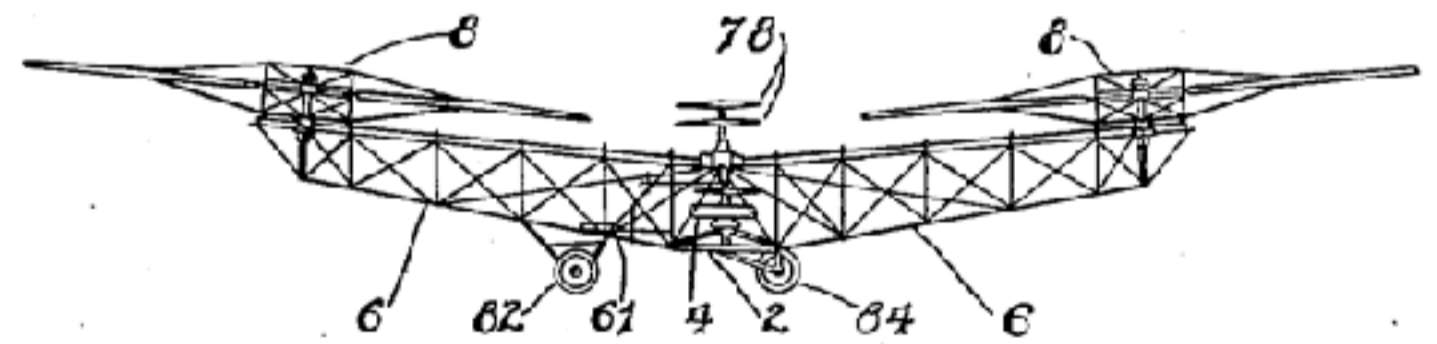


Fig. 2.

# Presentasjon

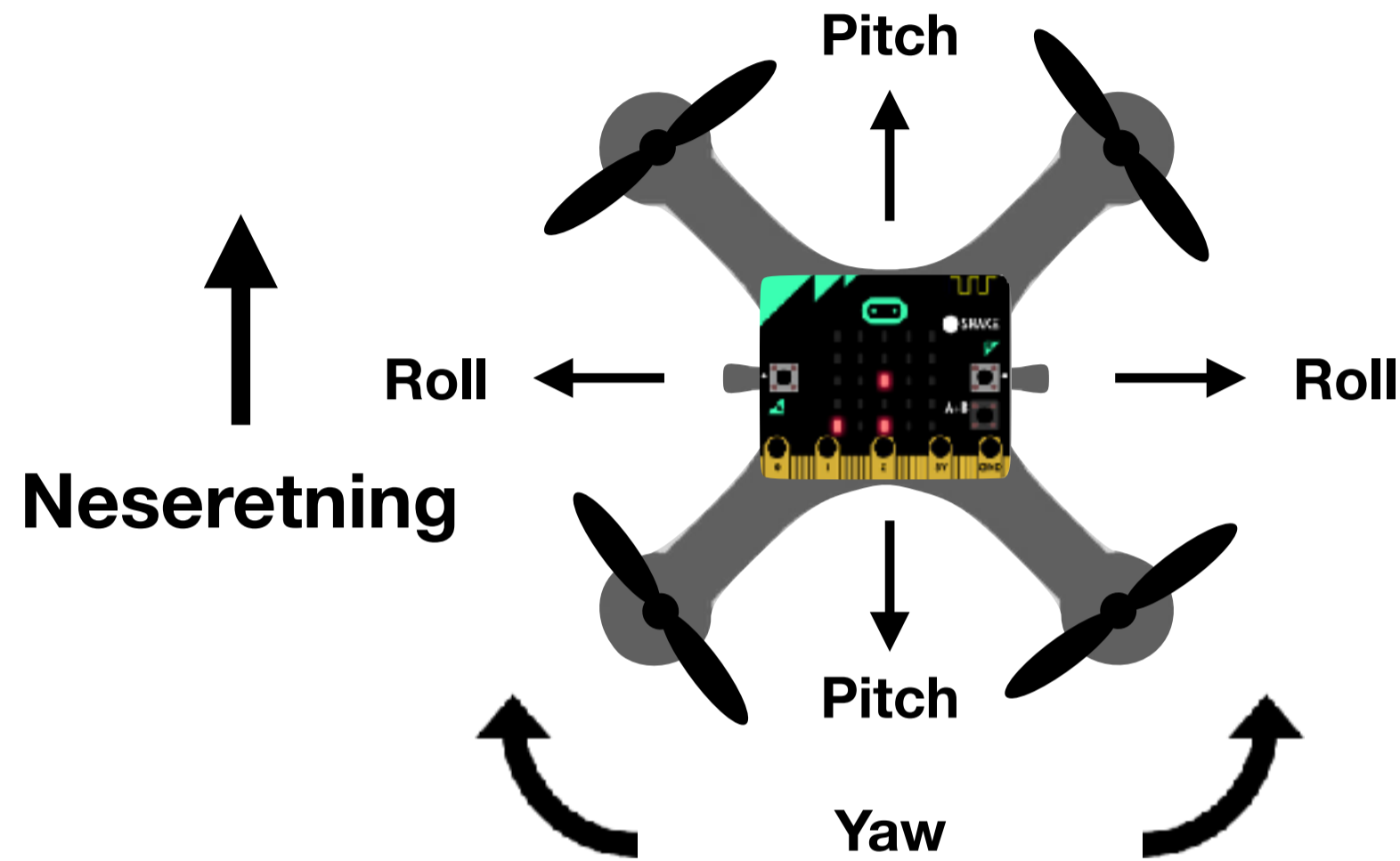
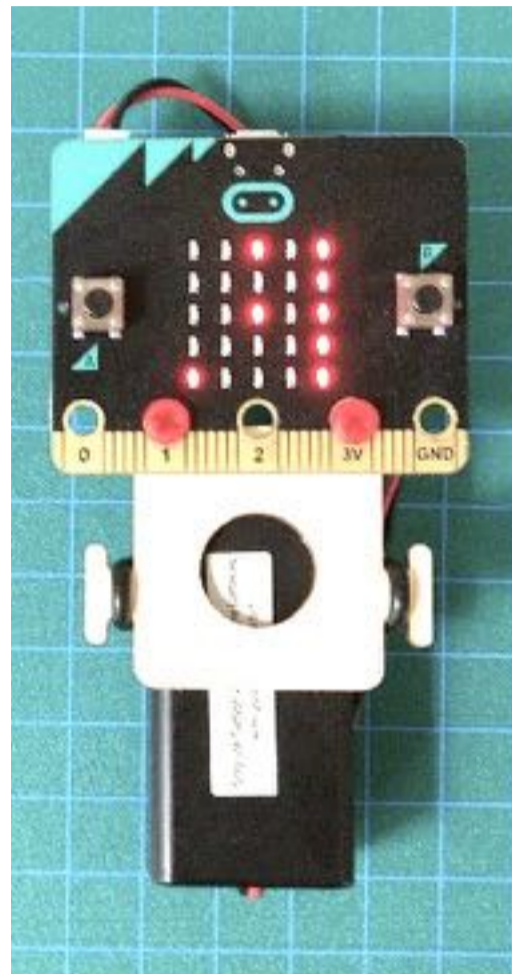
INVENTOR

BY George de Bohezat

Robert H. Young ATTORNEY

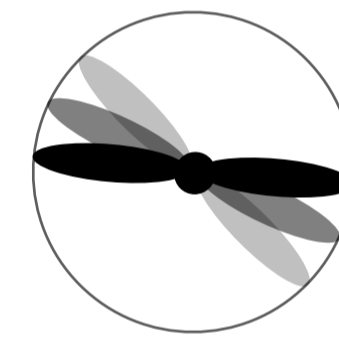
**PARTY**

# Flyretninger



**PARTY**  
pitch, arm, roll, throttle, yaw

**Arm:**  
Starte propeller



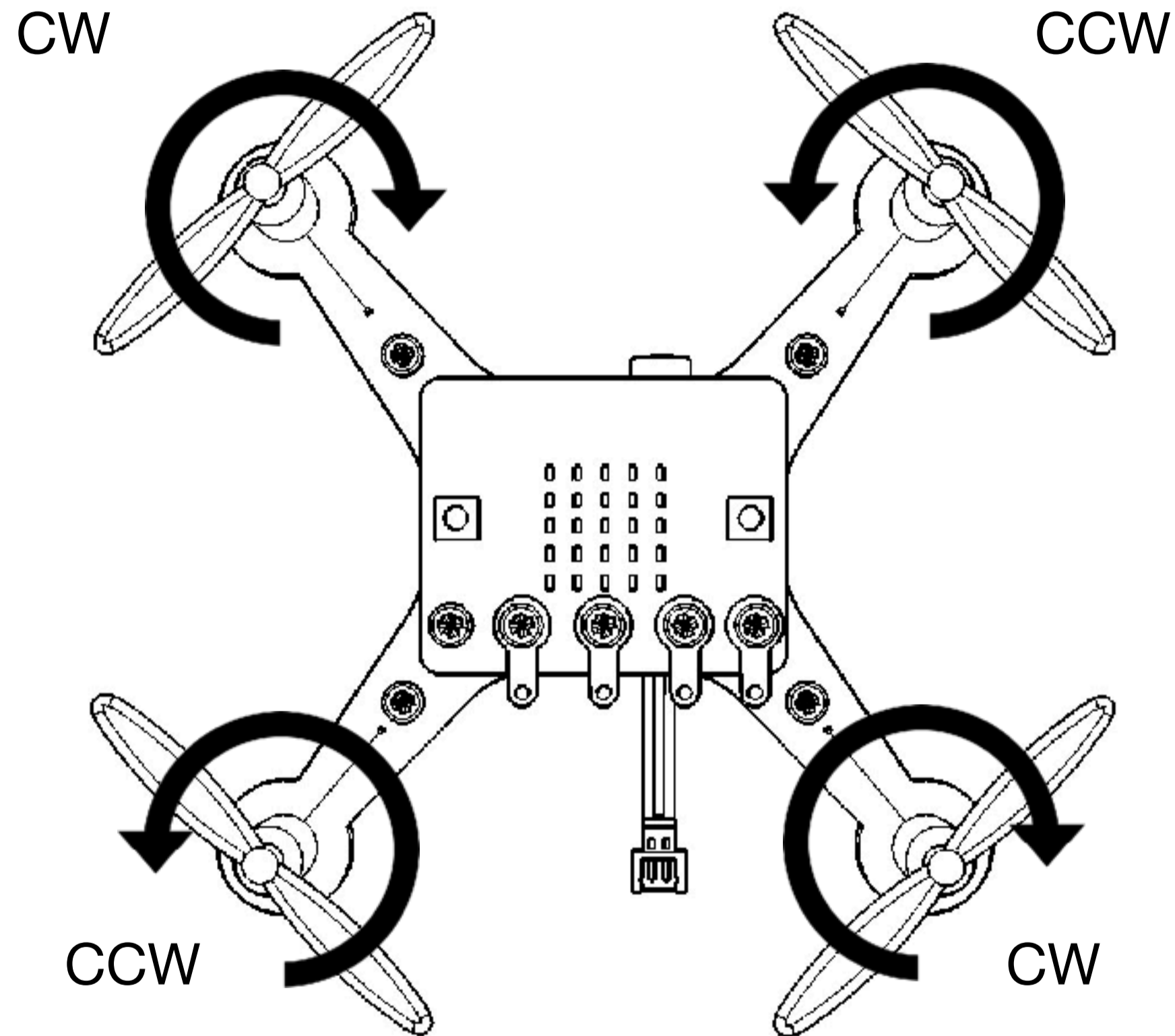
Minimum

Medium

Maximum

Arm	0	1	
Throttle	0	50	100
Pitch	-45	0	45
Roll	-45	0	45
Yaw	-30	0	30

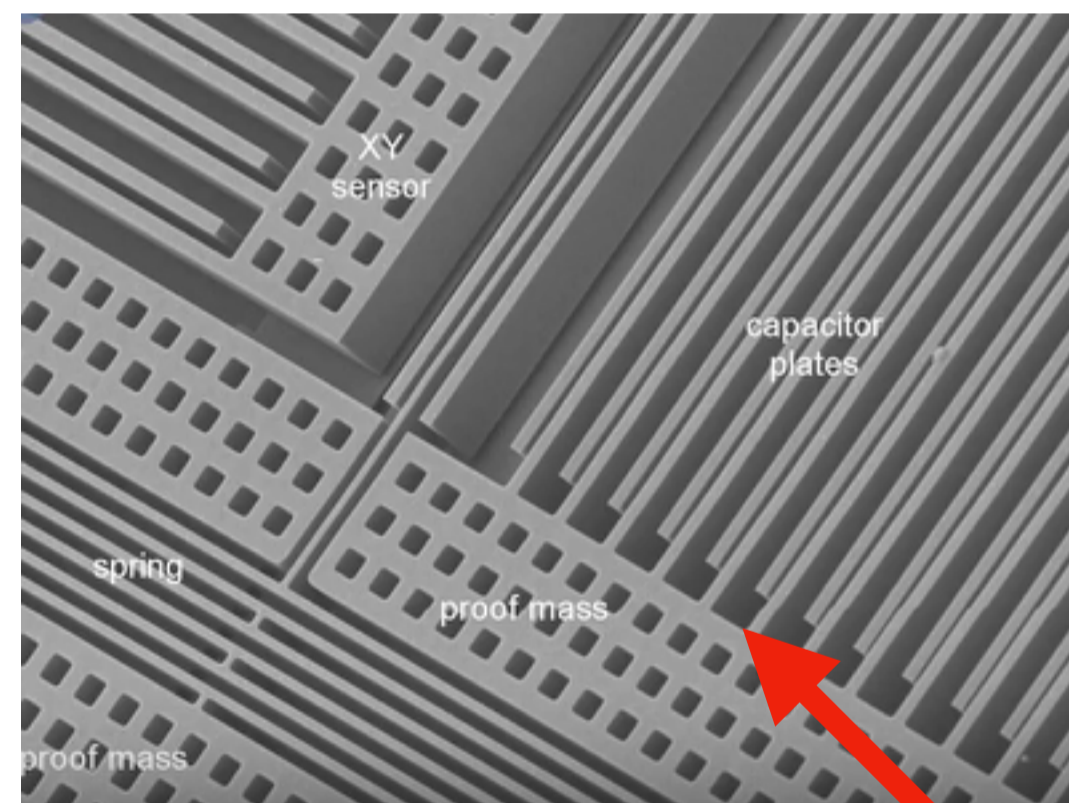
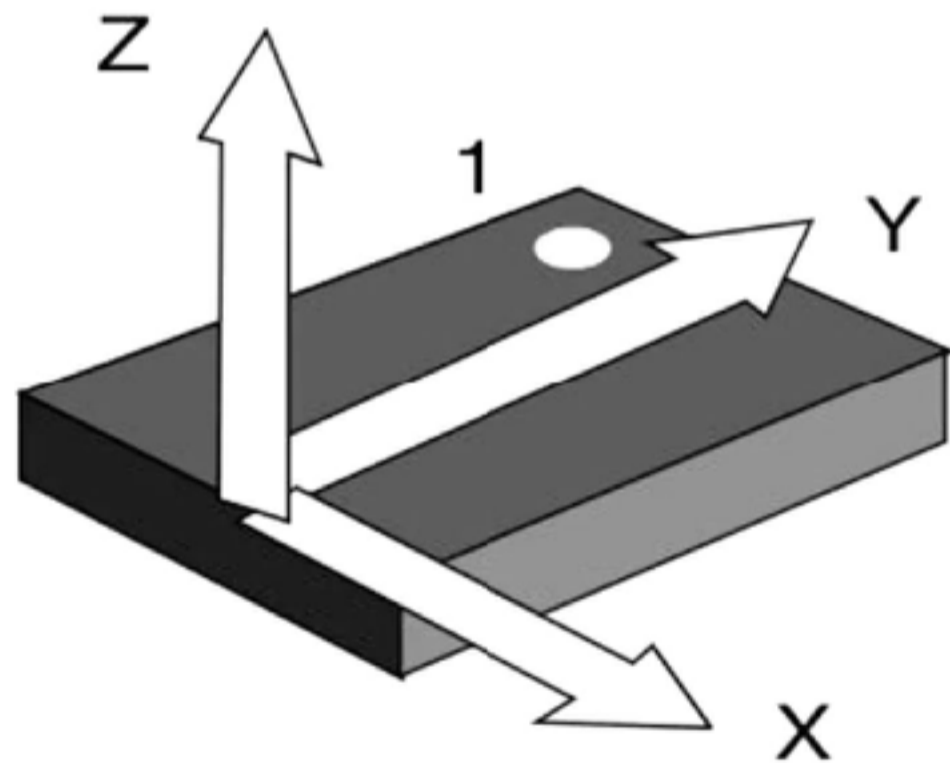
# Motorenes virkemåte



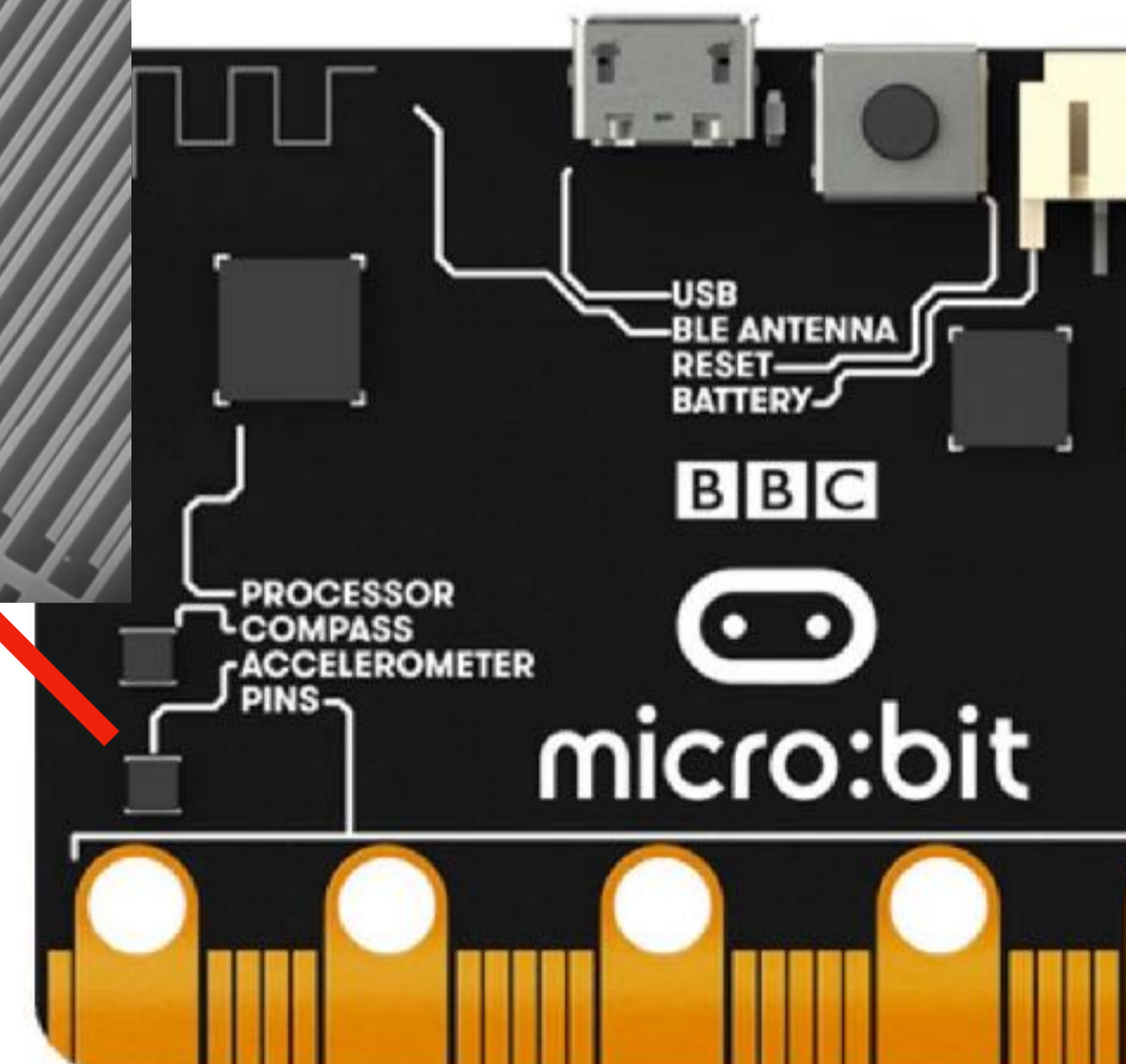
- To motorer går med klokka (Clock Wise), og to går mot klokka (Counter Clock Wise).
- Dronen går oppover (**Throttle**) ved at alle motorene går litt fortere
- Dronen går forover (**Pitch**) ved at de to bakerste motorene går litt raskere, mens to forreste går litt saktere
- Dronen går til venstre (**Roll**) ved at to motorer til høyre går litt fortere, mens to motorer til venstre går litt saktere.
- Dronen kan rotere langs **Yaw**-aksen ved at to CW-motorer går litt fortere mens to CCW går litt saktere.
- Styrekortet ordner dette for oss.

# Aksellerometer

- Micro:bit har et aksellerometer, også kalt MEMS (Micro Electro Mechanical System)
- Aksellerasjon er endring av fart.
- Sensoren kan måle i tre retninger, X Y og Z-retning
- Aksellerasjon og tyngdekraft oppleves som den samme kraften. Derfor kan aksellerometeret også måle hvilken vei bakken er.

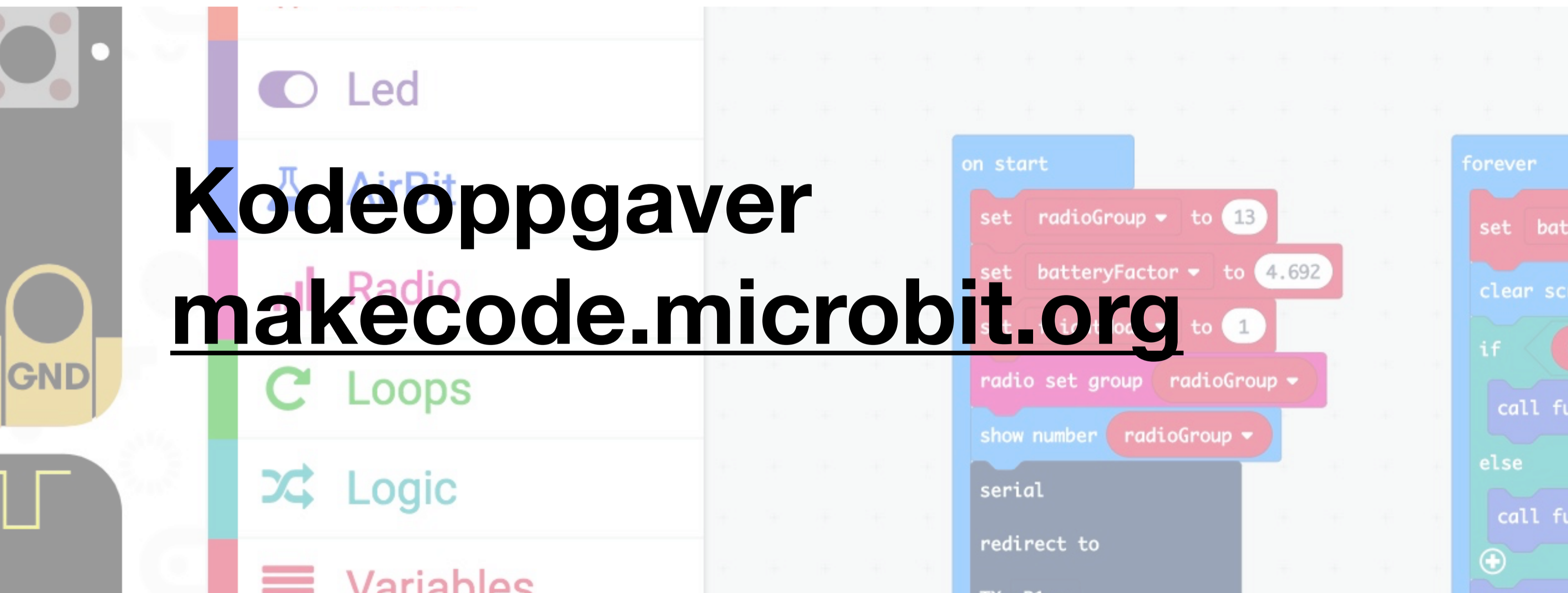


mikroskopiske fjærer flytter seg ved bevegelse



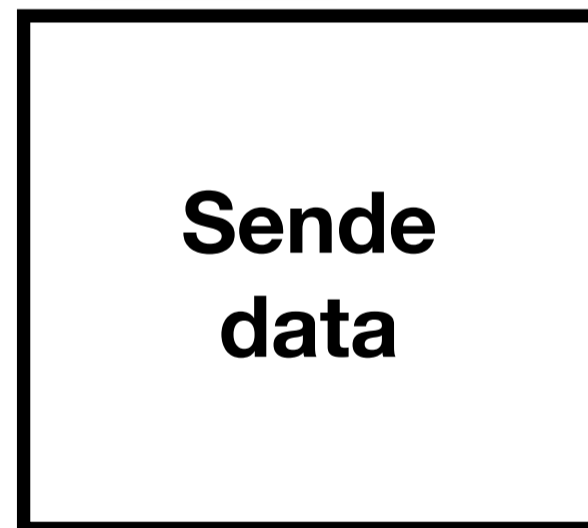
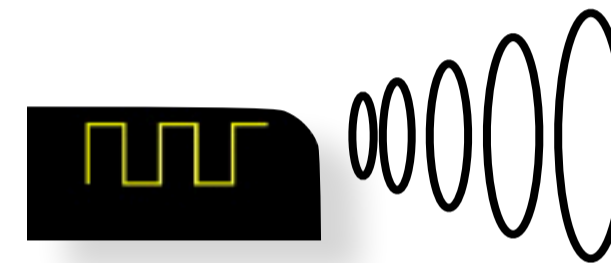
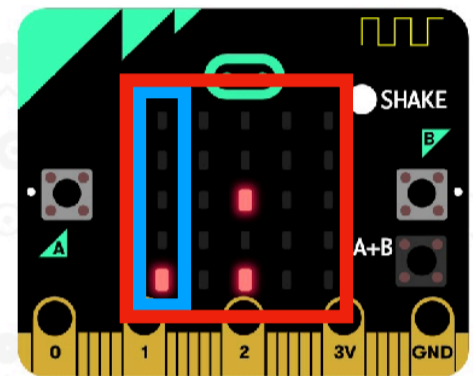
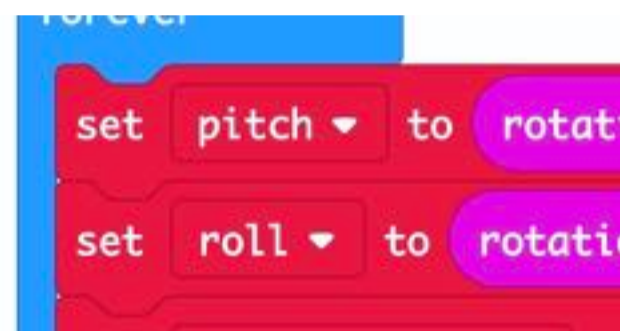


# Kodeoppgaver [makecode.microbit.org](https://makecode.microbit.org)



Dette dokumentet er under utvikling og skal ikke publiseres eller deles uten skriftlig tillatelse fra Dronebygg Henning Pedersen

# Dette skal vi kode:



Vi samler inn data fra sensorer og knapper og lagrer dem

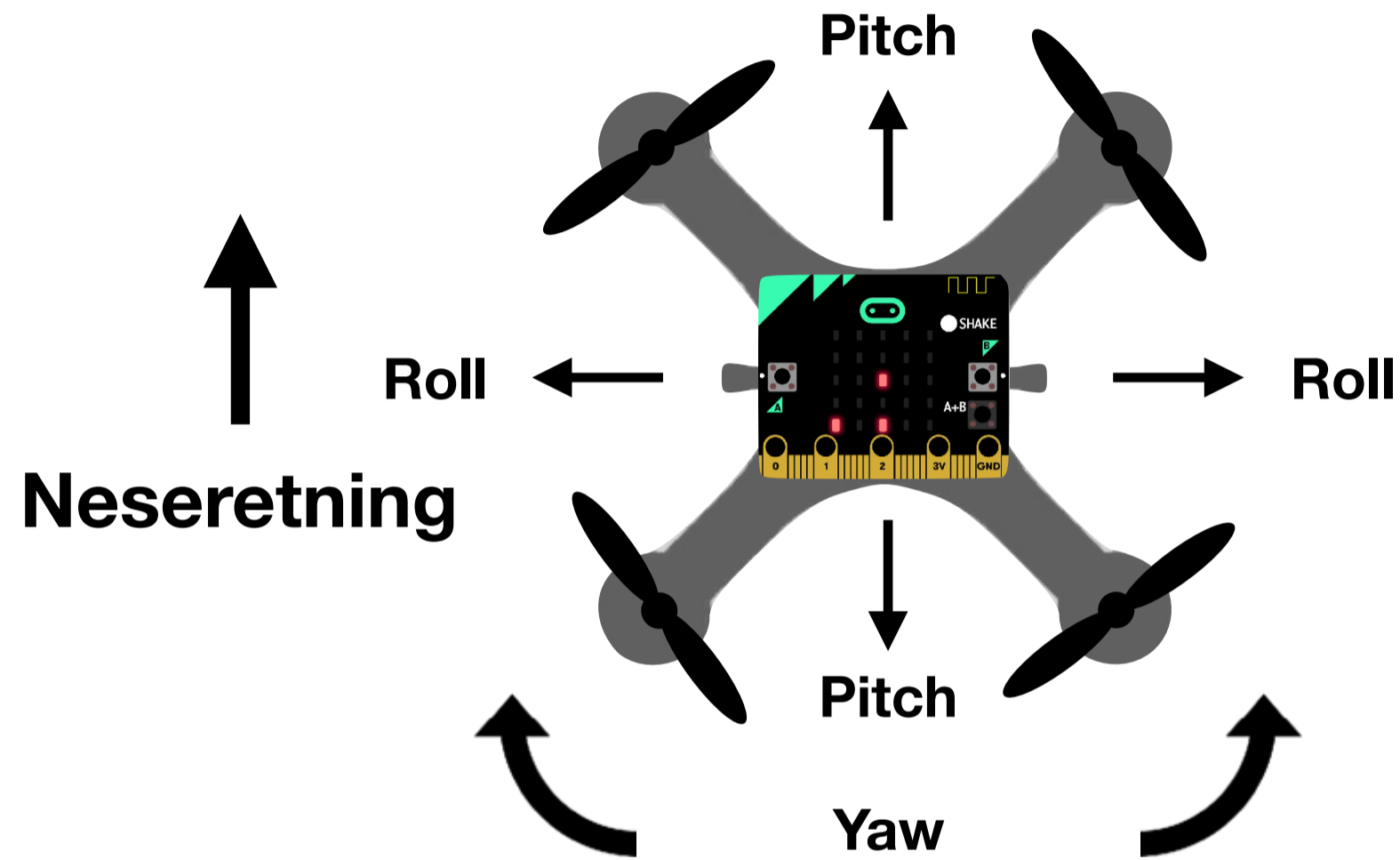
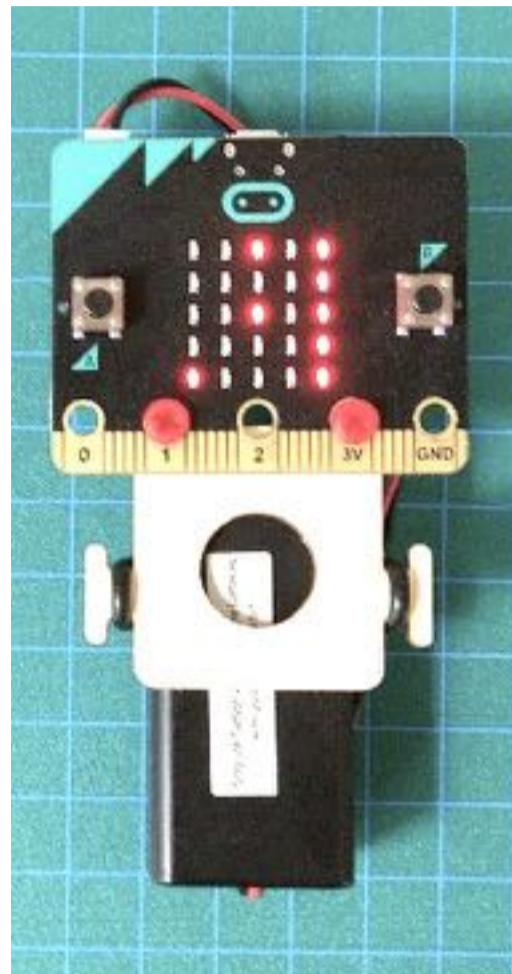
Vi regner om til koordinater slik at vi kan vise disse dataene på skjermen

Vi sender dataen over radioen

Vi må motta dataen på dronen på rett kanal slik at vi kan styre dronen. Vi må legge inn ferdig kode der vi må bytte til rett kanal.



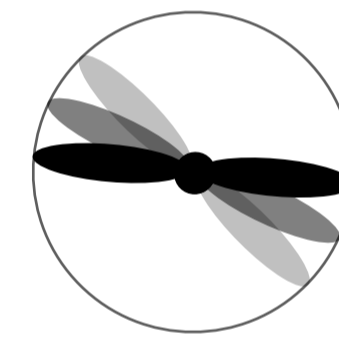
# Flyretninger



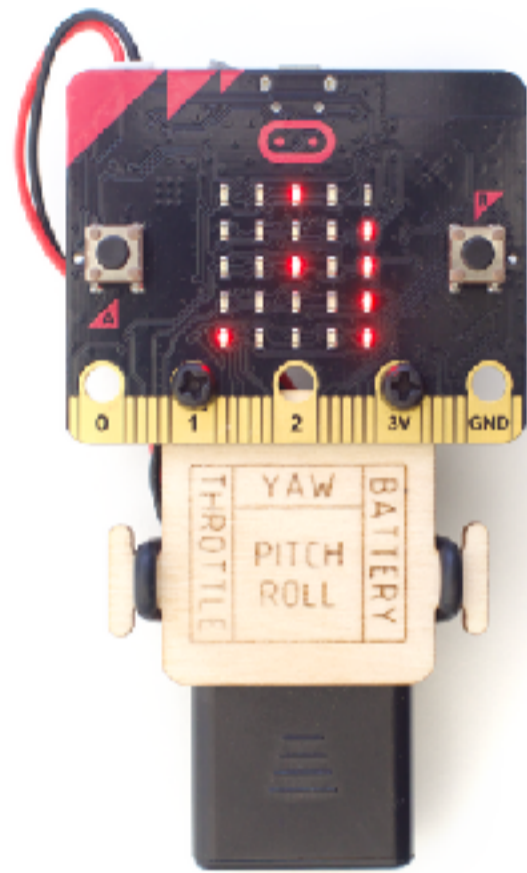
# PARTY

pitch, arm, roll, throttle, yaw

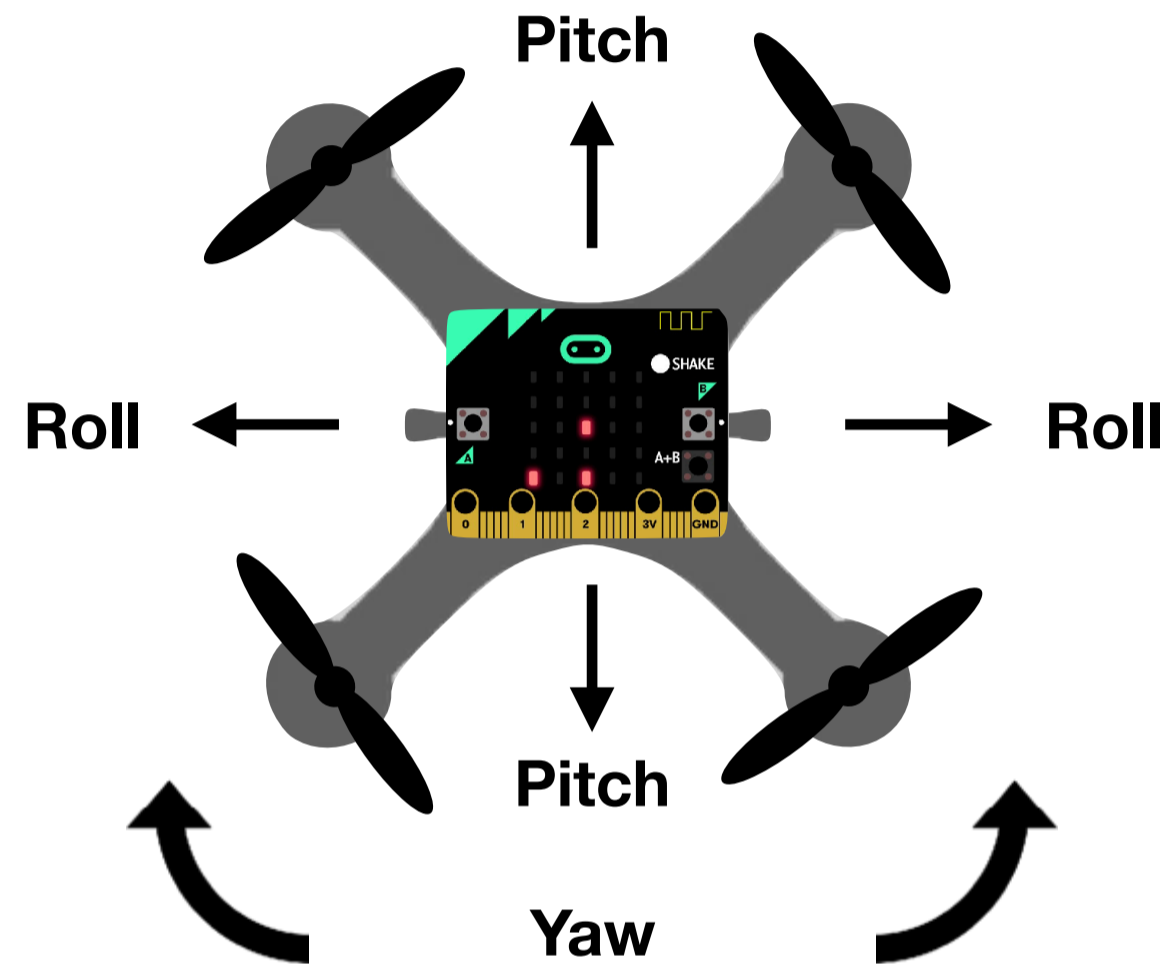
**Arm:**  
Starte propeller



Variabel:	Minimum	Nøytral	Maximum
Arm	0		1
Throttle	0	50	100
Pitch	-45	0	45
Roll	-45	0	45
Yaw	-30	0	30

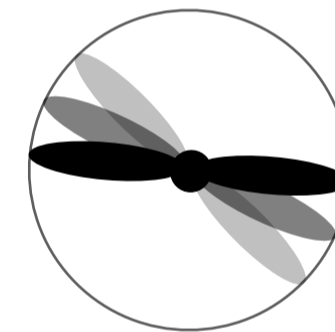


**Ergonomic hand controller**



**PARTY**  
pitch, arm, roll, throttle, yaw

**Arm:**  
**Spin propellers**



# Starte å kode

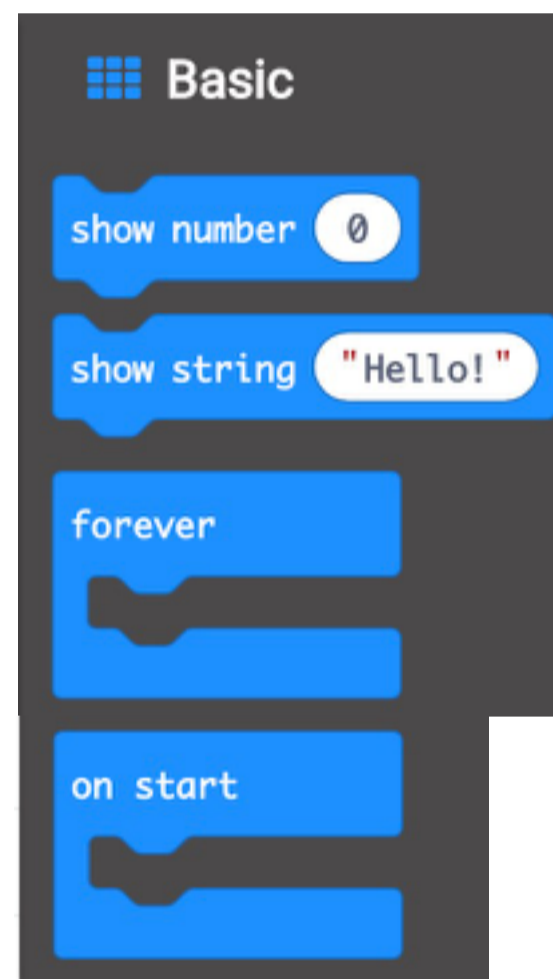
- Gå til [makecode.microbit.org](https://makecode.microbit.org) (Chrome nettleser)
- Velg New Project
- Gi prosjektet et navn
  
- Bruk CTRL C CTRL V for å kopiere / lime inn
- Bruk Backspace for å slette
- Husk å laste over rett versjon til micro:bit dersom du ikke laster over direkte.
- Gjør det enkelt: Løs problemene med så lite kode som mulig
- Hold det oversiktlig på skjermen
- Ikke ha flere forever-løkker (du vet ikke hvem som kjører først)
- Ikke ha flere show number (du ser ikke forskjell på de)
- Ikke ha forsinkelser (blokker som starter med Show) i forever-løkken når du skal fly da dette forsinker styringen og gjør det vanskelig å fly.

**[makecode.microbit.org](https://makecode.microbit.org)**

# Kodeblokker

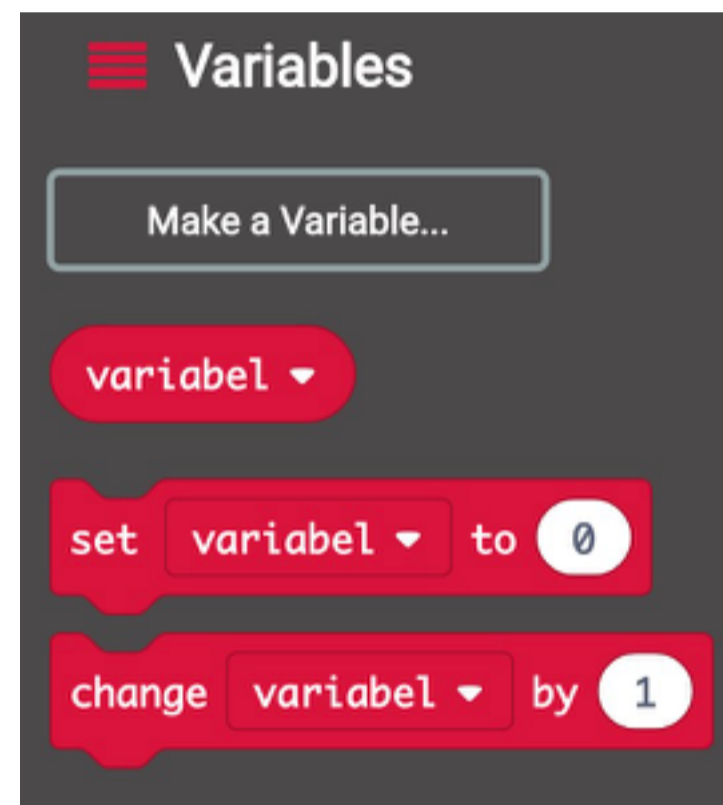
[makecode.microbit.org](http://makecode.microbit.org)

Disse blokkene skal vi bruke:



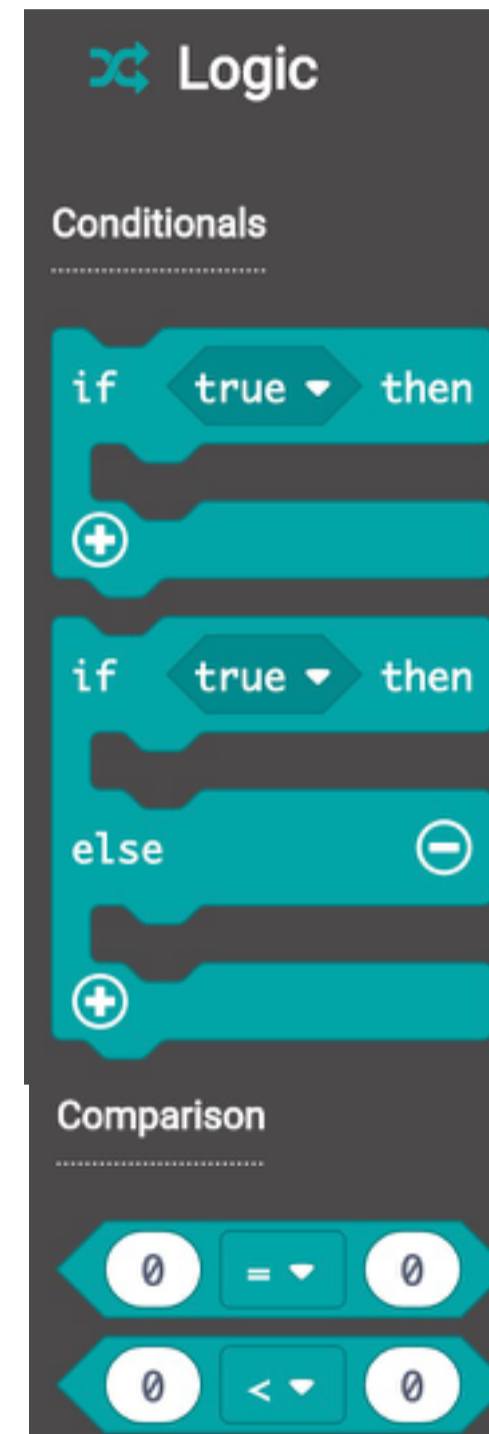
## Basic:

Vis tall (show number),  
Vis tekst (string)  
Forever-løkke  
On start



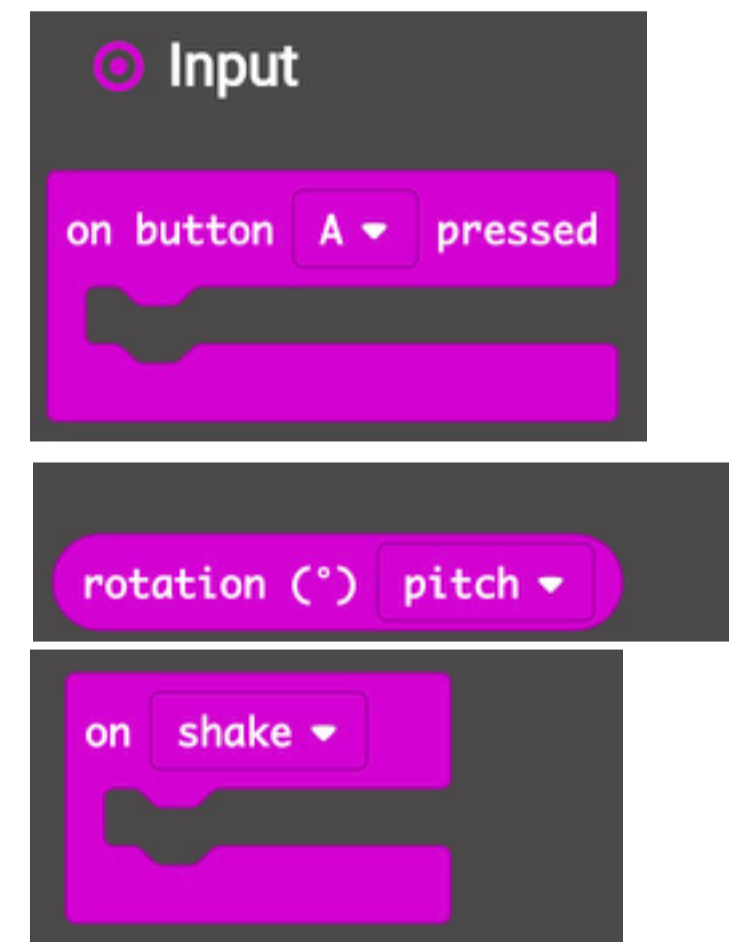
## Variables:

Lage variabler, sette og  
forandre variabler



## Logic:

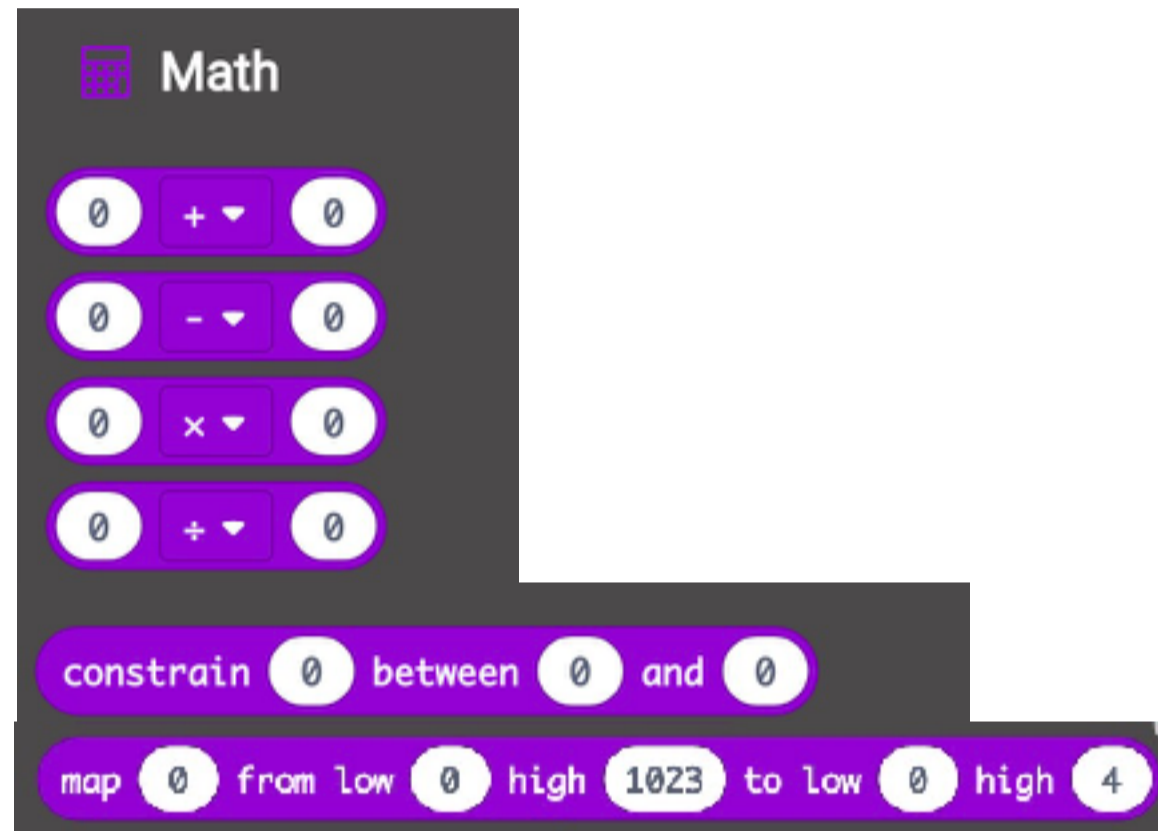
IF (hvis- parameter)  
Else (hvis ikke)  
Sammenligning  
(comparison)



## Input:

Knappene A, B, og A+B  
Rotation Pitch og Roll  
(vinkel)  
On Shake (risting)

# Flere blokker



## Math

Regne om variabler.

Bruke paranteser for å

bestemme rekkefølge

Constrain (begrense tall)

Map (omregning og skalering)

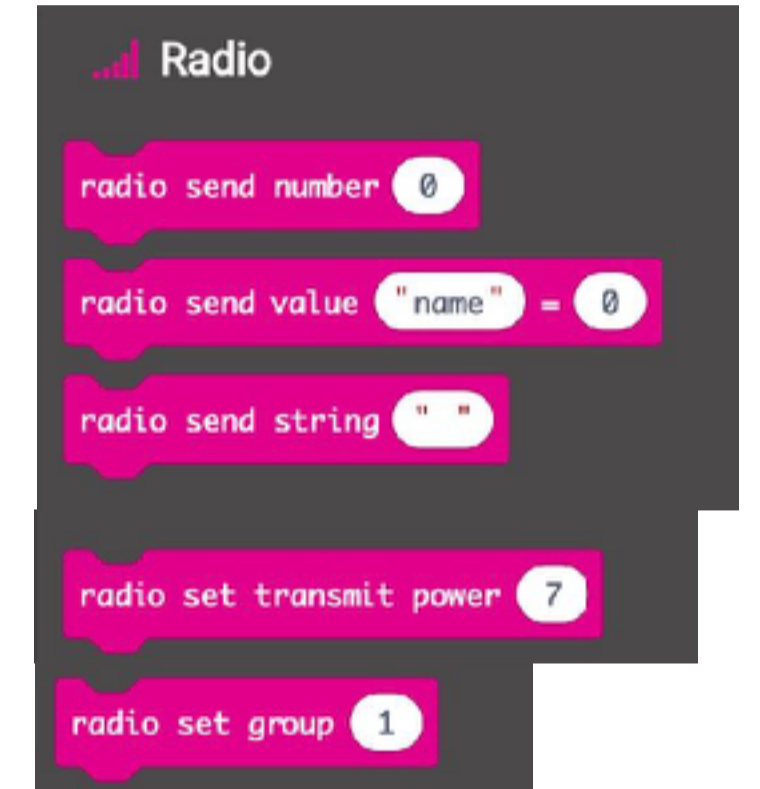


## Pins:

Analog read, analog write:

Styre støm (spenning) ut fra pinnene på micro:bit.

Måle spenning inn. Bruke pinnene 0-2



## Radio:

Sende tekststreng

kombinert med tall

Sette sendestyrke

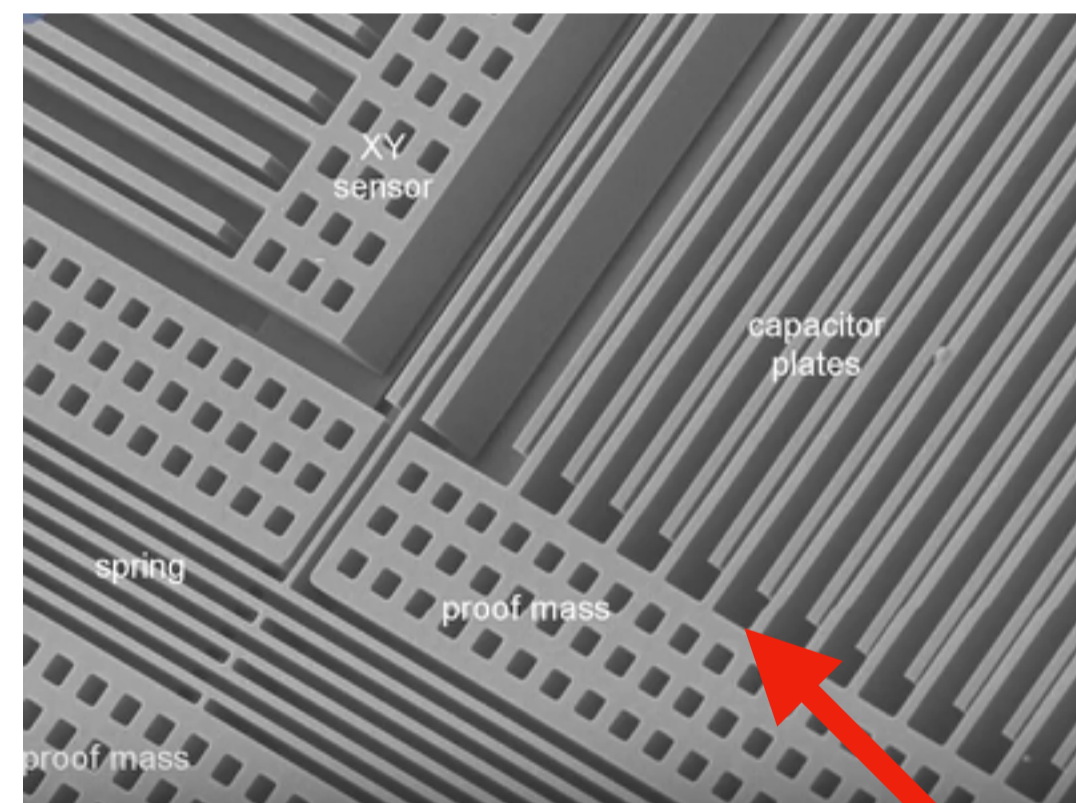
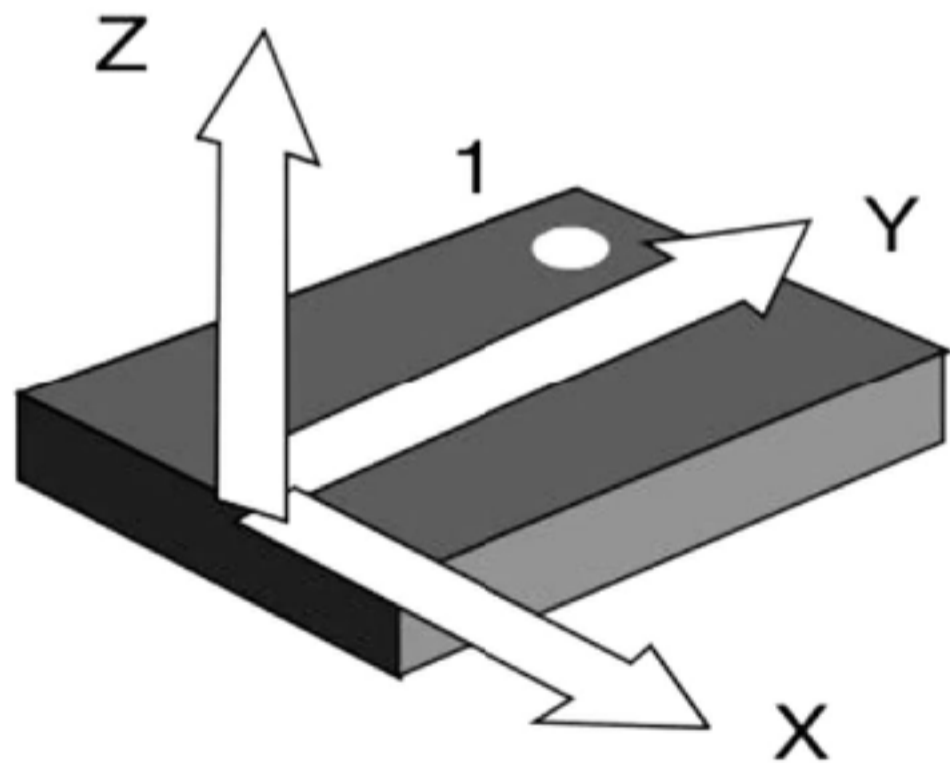
(transmit power)

Velge radiogruppe

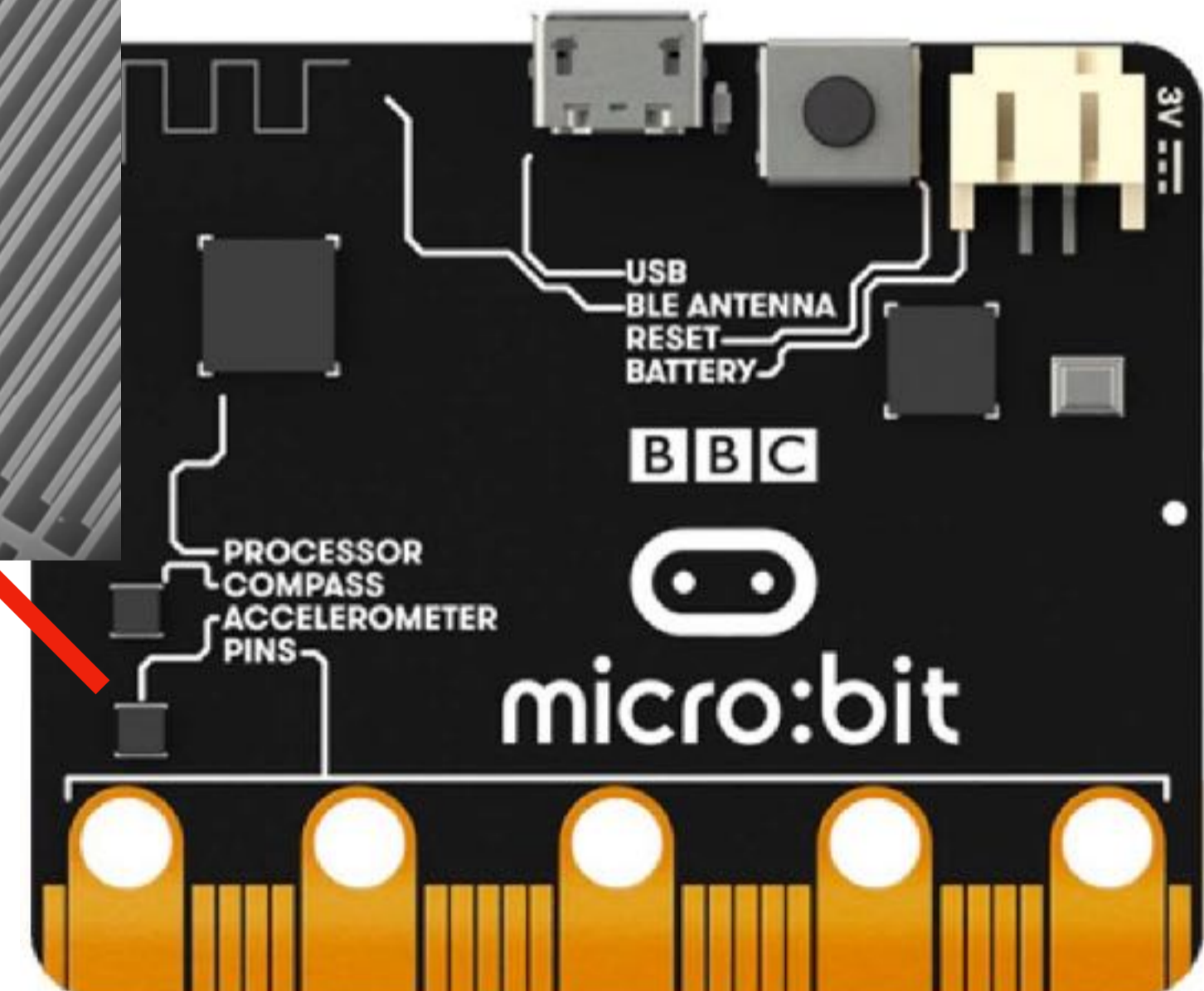
(radio set group)

# Aksellerometer

- Micro:bit har et aksellerometer, også kalt MEMS (Micro Electro Mechanical System)
- Aksellerasjon er endring av fart.
- Sensoren kan måle i tre retninger, X Y og Z-retning
- Aksellerasjon og tyngdekraft oppleves som den samme kraften. Derfor kan aksellerometeret også måle hvilken vinkel micro:bit har i forhold til bakken.



mikroskopiske fjærer flytter seg ved bevegelse



# Kode til fjernkontroll: Variabler

## PARTY

pitch, arm, **roll**, **throttle**, yaw

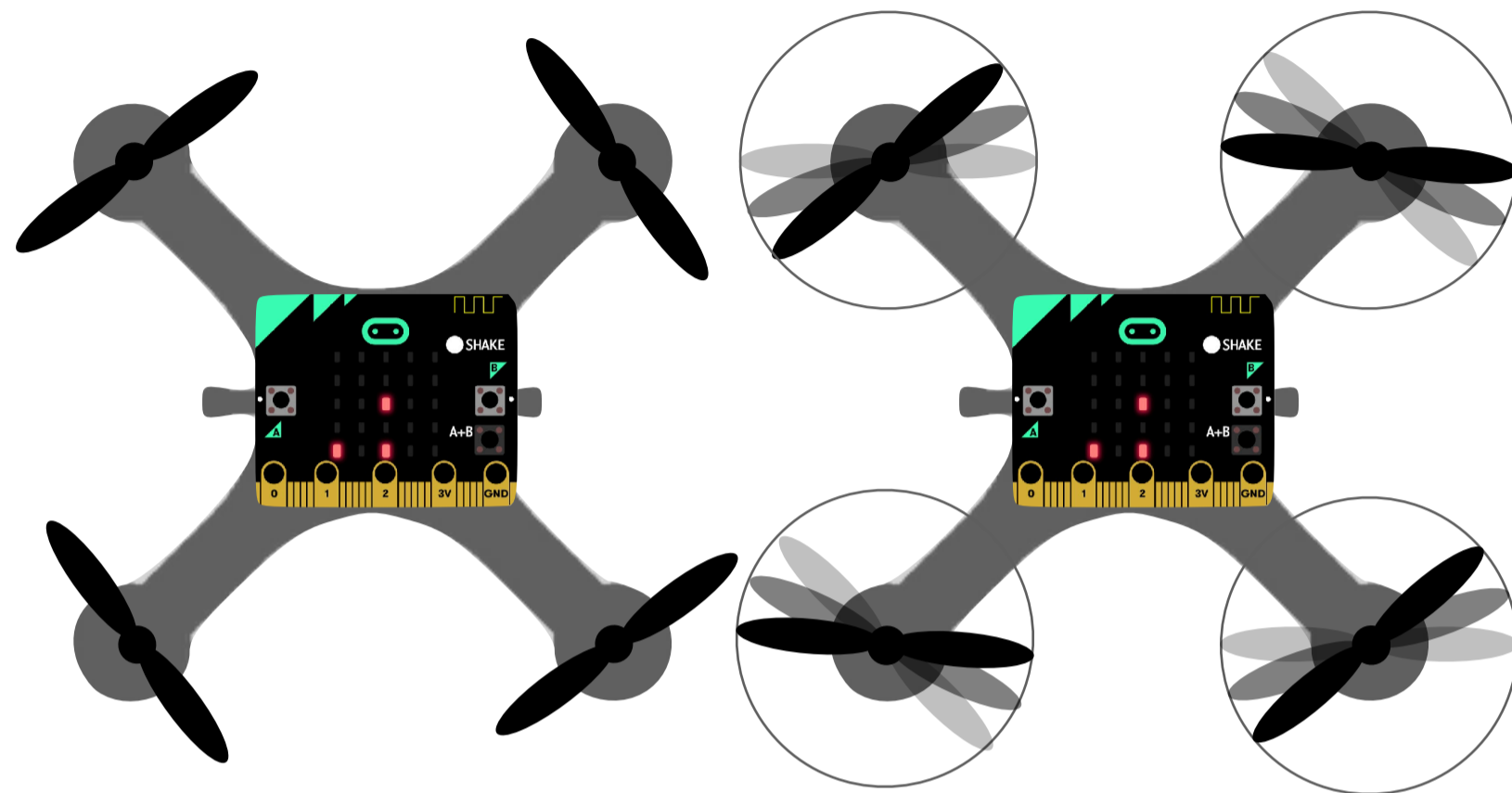
### Throttle

- Lag en variabel som heter throttle.
- Bruk knappene A og B for å endre throttle. A skal minke med 5, B, skal øke med 5
- Throttle skal ikke øke mer enn en gang pr. klikk
- Bruk Show Number ( i forever-løkken) for å kontrollere at throttle endres riktig og at den aldri kan være under 0 eller over 100
- Gjør slik at throttelen kun økes med 1 når den er over 40. Dette gjør det lettere å finstille mengden fart som skal til for å holde dronen helt i ro vertikalt.

### Roll og Pitch

- Lag en variabel som heter Roll og en som heter Pitch.
- I Forever: Sett variablene roll og pitch til å hente roll og pitch fra input-blokkene
- Bruk Show Number for å vise hva som blir lest. Les av en variabel av gangen. Husk at tallet må ha tid til å scrolle over skjermen før det blir oppdatert.

# Armering



**Disarmert**

**Armert**

## Armering

- Armering er en sikkerhetssperre, omtrent som startnøkkelen på en bil.
- Propellene kan ikke starte før variabelen er satt til 1.

## Oppgave:

- Lag en variabel som heter Arm
- Lag en start/stopp knapp: Når du trykker på A+B skal arm skiftes mellom 0 og 1 annenhver gang
- Altså hvis den er 0, skal den bli 1, og motsatt
- Bruk show number for å teste at den virker
- Når vi armerer, skal throttle alltid være 0, for at ikke dronen skal kunne ta av for tidlig. Legg inn en funksjon som gjør at throttle alltid settes til 0 i det vi armerer.
- Det skal gå an å øke thottelen etter at vi har armert.

## Ekstraoppgave

- Lag en “crash detection” som stopper propellene når dronen ligger sidelengs eller opp ned.

Bryteren setter Arm til 0 og 1 annenhver gang



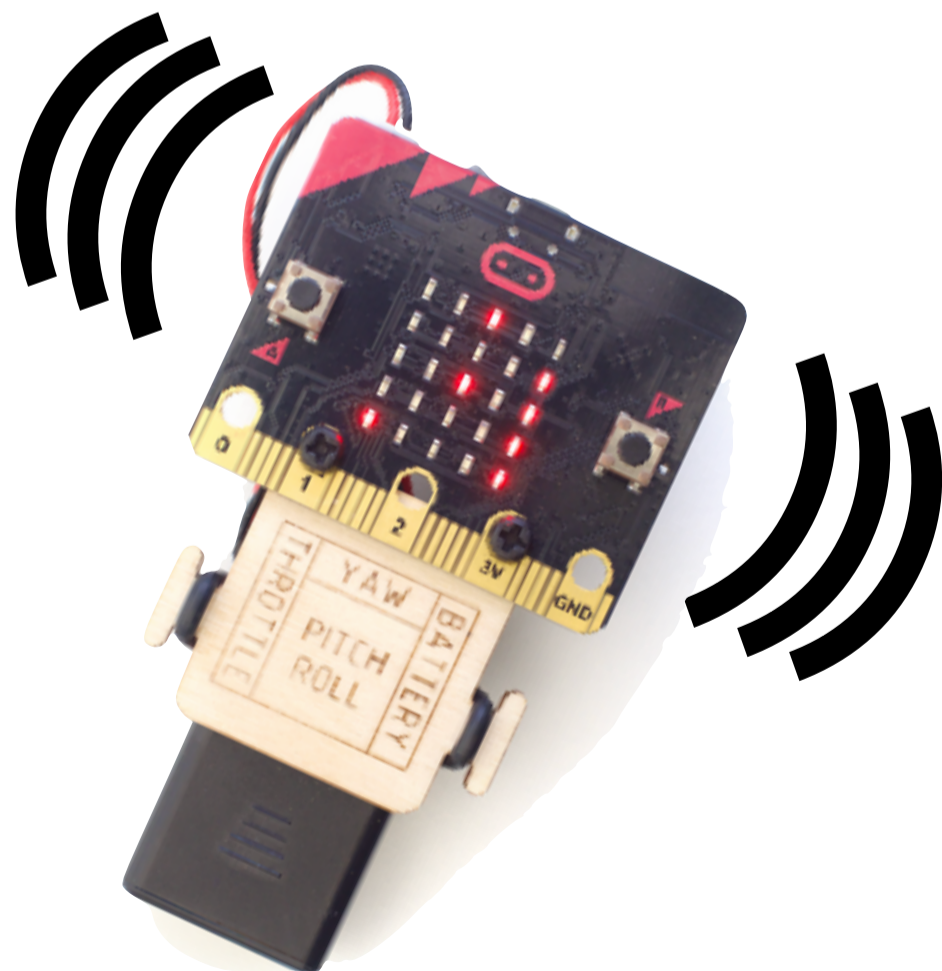
**PARTY**  
pitch, **arm**, roll, throttle, yaw



# Kræsjs-deteksjon og panikkfunksjon

Om vi mister kontroll over dronen glemmer vi fort å trykke A+B. Da er det enklere om vi kan riste på fjernkontrollen. Om dronen krasjer og havner på siden eller opp ned er det praktisk om den selv stanser motorene slik at de ikke fortsetter å slå bort i ting.

- Bruk funksjonen On Shake til å desarmere dronen
- Finn en måte å oppdage om dronen ligger på siden eller opp ned, og gjør at den da desarmerer dronen umiddelbart.



# Visualisere verdiene

## Clear Screen

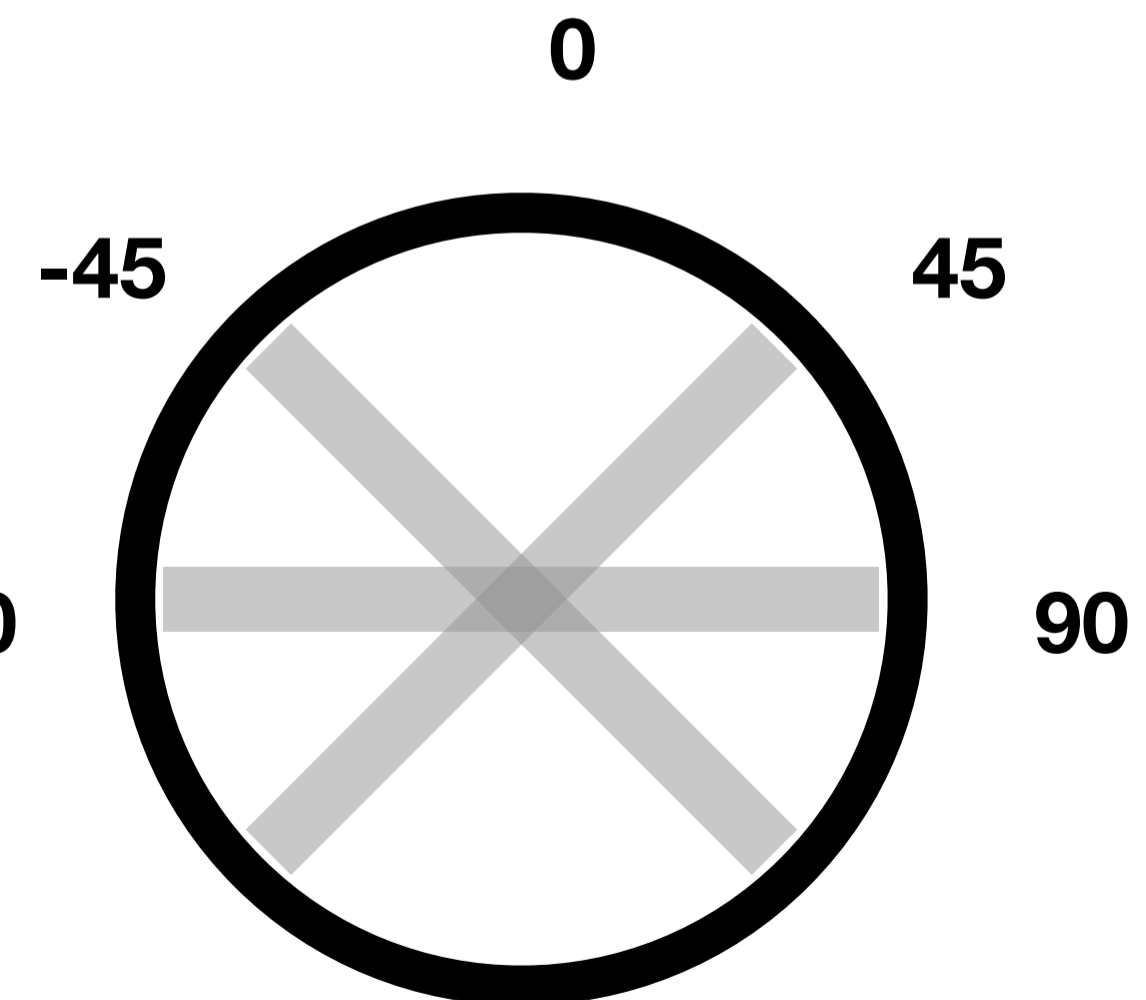
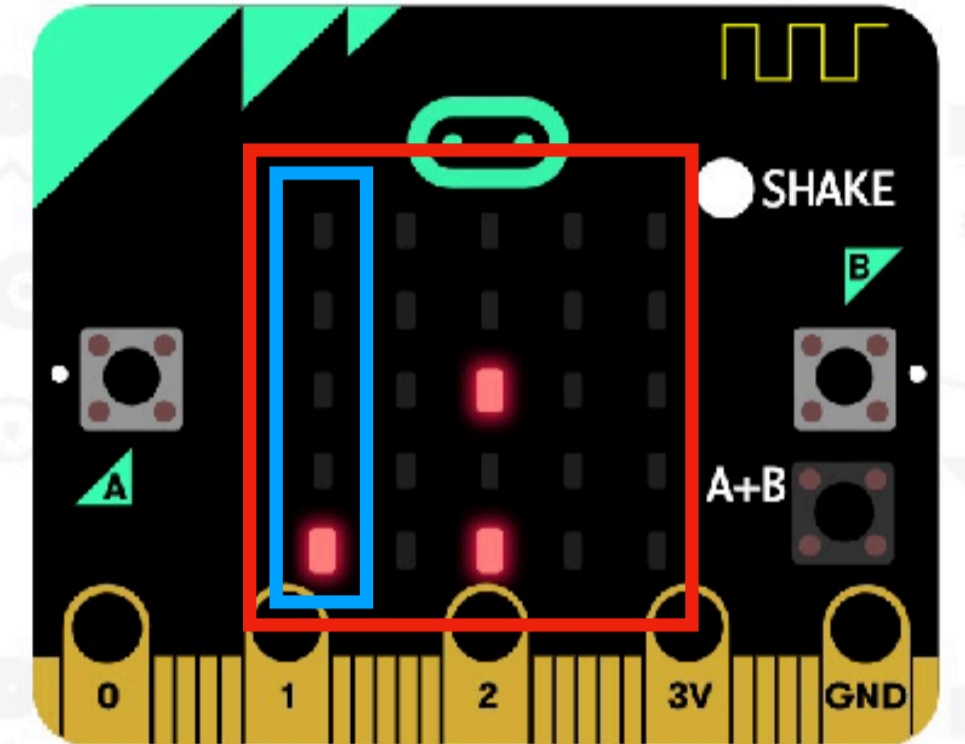
Før vi plotter, må vi tømme skjermen for tidligere piksler. Bruk Basic: More og Clear Screen, som skal kjøres rett før plottingen.

## Throttle

- Vi kan indikere hvor mye throttle vi har med pikslene til venstre (blå ramme). Utfordringen er at Y-aksen har null-punktet på toppen, mens vi ønsker at prikken går oppover ved høyere throttle.
- Lag et regnestykke der et av tallene i regnestykket kan byttes ut. Hvis du putter inn 100 i dette regnestykket, skal du få 4 til svar, hvis du putter 50 inn i dette regnestykket skal du få 2 til svar, hvis du putter 0 inn i regnestykket skal du få 0 til svar.
- Lag et nytt regnestykke. Hvis du putter inn 4 i dette regnestykket, skal du få 0 til svar. Tilsvarende skal 3 gi 1, 2 gi 2, 1 gi 3 og 0 gi 4.
- Sett sammen oppgave a og b til et nytt regnestykke slik at når du putter 100 inn i dette regnestykket, får du 0 til svar, 50 skal gi 2 og 0 skal gi 4 til svar.
- Plott svaret du får med  $X=0$  og  $Y=funksjonen\ din$  med Throttle som input. Test resultatet i simulator ved å øke/senke Throttle.

## Roll / Pitch

- For å styre Roll og Pitch på dronen bruker vi vinkelmåleren på micro:bit. Vi begrenser oss til -45 grader til venstre over mot 45 grader til høyre.
- Lag et regnestykke der et av tallene kan byttes ut. Hvis du putter inn -45 i dette regnestykket, skal du få 0 til svar, hvis du putter 0 inn i dette regnestykket skal du få 2 til svar, hvis du putter 45 inn i regnestykket skal du få 4 til svar.
- Bruk plot-funksjonen til å plote svaret du får når du legger variabelen Roll inn i regnestykket. Hele skjermen kan brukes. (Rød ramme)
- Gjør det samme med Pitch. Her er det y-delen av Plot som skal regnes ut.



# Visualisere verdiene

## Oppgavene formulert som funksjoner

- Lag en funksjon med følgende verditabell for **throttle**

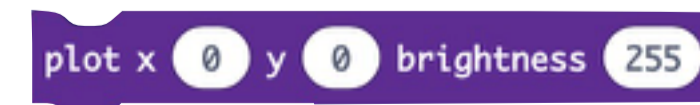
Inn	100	50	0
	4	2	0
Ut	0	2	4

- Lag en funksjon med følgende verditabell for **pitch og roll**

Inn	-45	0	45
45	0	45	90
/22.5	0	2	4

## Armering

- Bruk skjermen til å vise om dronen er armert eller ikke. Du kan plote en eller flere nye piksler eller forandre på de eksisterende ved å erstatte Plot med Plot Brightness (under Led - More)
- **Bonusoppgave:** Klarer du å få en Lysdiode til å endre styrke eller blinke når den er armert?



# Visualisere verdiene

## Oppgavene formulert som funksjoner

- Lag en funksjon med følgende verditabell for **throttle**

Inn	100	50	0
	4	2	0
Ut	0	2	4

- Lag en funksjon med følgende verditabell for **pitch og roll**

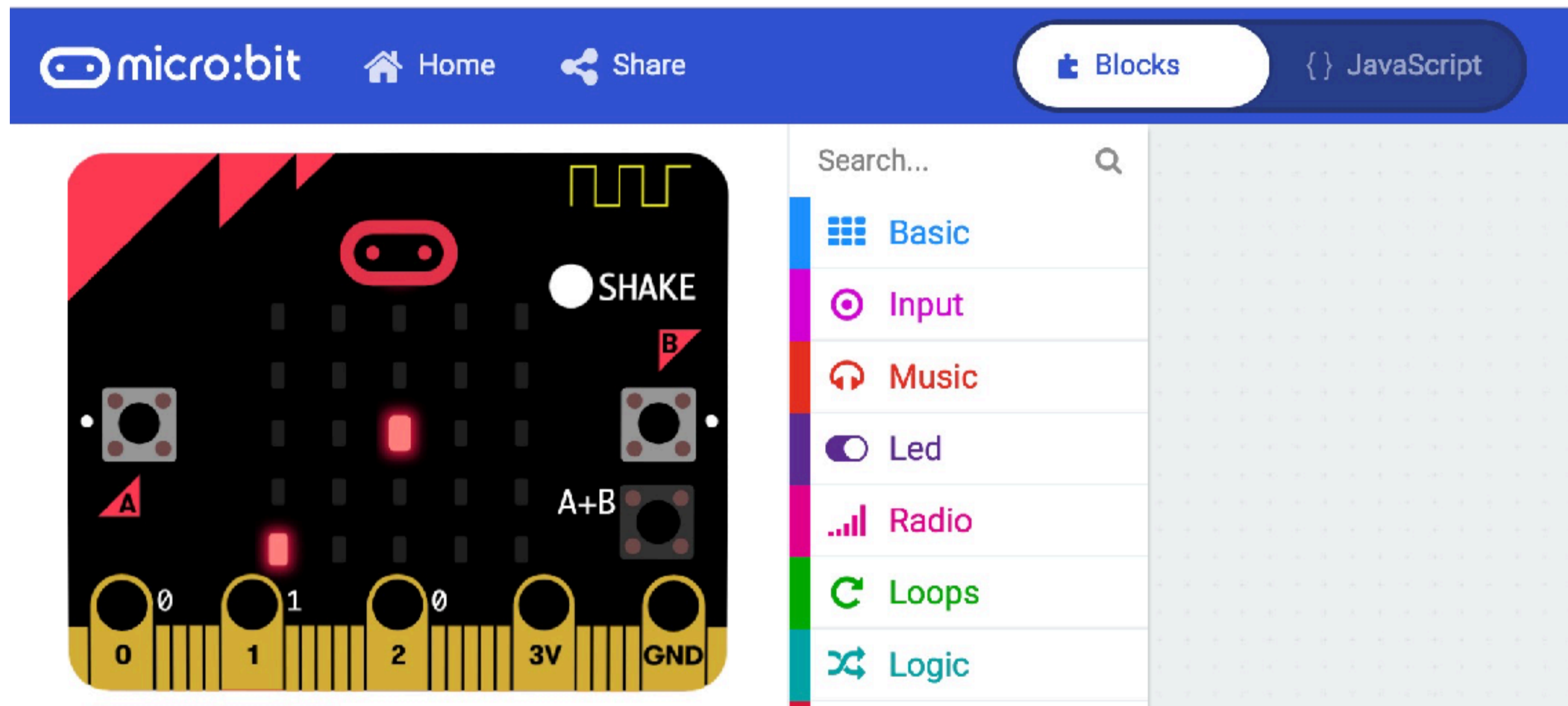
Inn	-45	0	45
	0	45	90
	0	2	4

## Armering

- Bruk skjermen til å vise om dronen er armert eller ikke. Du kan plote en eller flere nye piksler eller forandre på de eksisterende ved å erstatte Plot med Plot Brightness (under Led - More)
- **Bonusoppgave:** Klarer du å få en Lysdiode til å endre styrke eller blinke når den er armert?



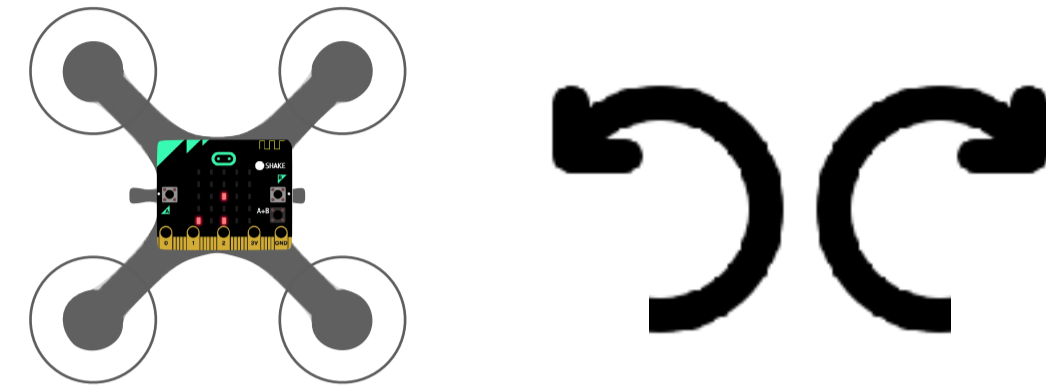
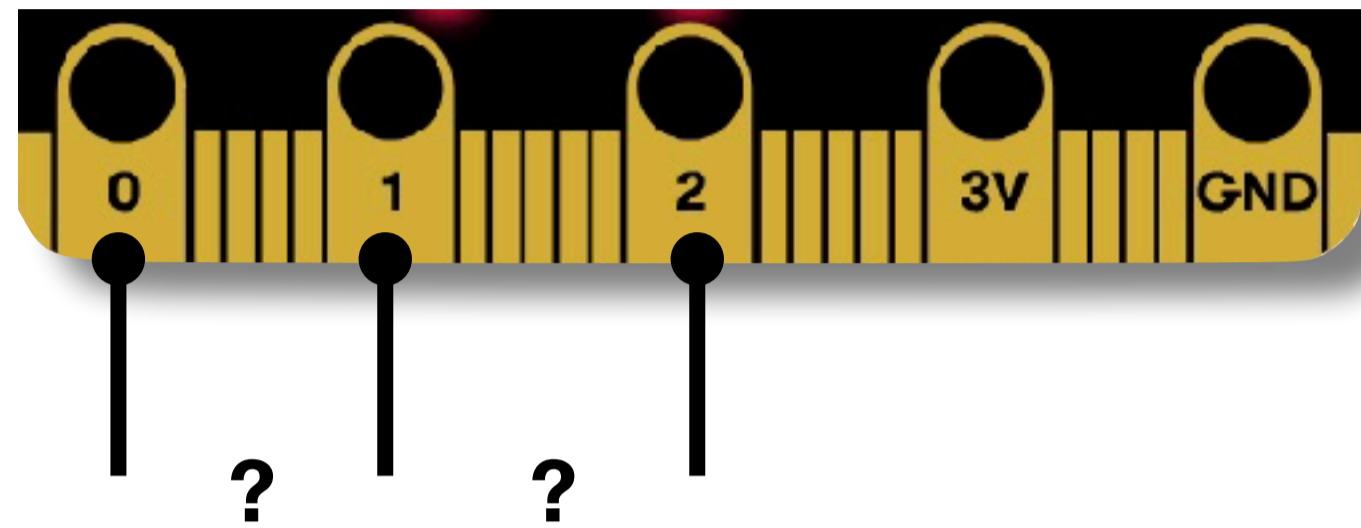
# Test om det virker!



**Har du kodet rett, skal følgende virke:**

- Prikken i midten av skjermen følge musepila i simulatoren.
- Throttle-pikselen på venstre side skal stige

# Ekstraoppgave: Yaw



Yaw er det som får dronen til å rotere sidelengs.

## Yaw

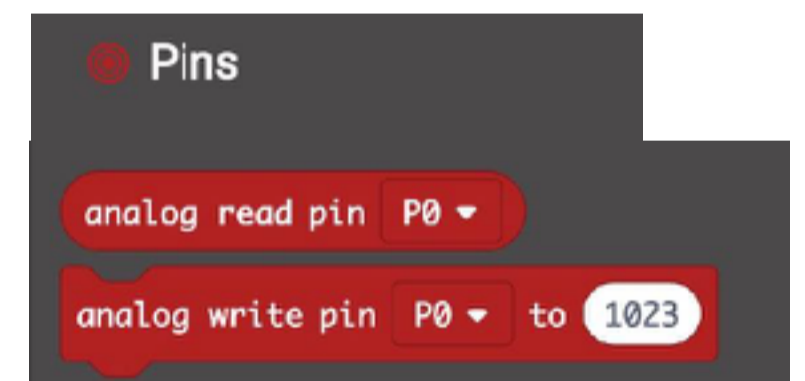
Yaw får dronen til å rotere sidelengs. Hvordan kan vi klare å styre dette uten å sette på noen ekstra knapper?

- Bruk analog write for å sende strøm ut av en av pinnene 0-2
- Bruk analog read og show number for å lese av de to andre pinnene 0-2
- Last over koden til en micro:bit (simulator kan ikke brukes)
- Les først av tallet på micro:bit som kommer når vi ikke gjør noen ting og skriv det ned.
- Hvordan kan vi så lede strøm mellom pinnene på enklest mulige måte?
- Prøv å lage en kobling mellom P1 og P2 for å få tallet til å bli høyere, (over 500)

Hint: Huden vår leder strøm

## Oppgave: Lag en kode som:

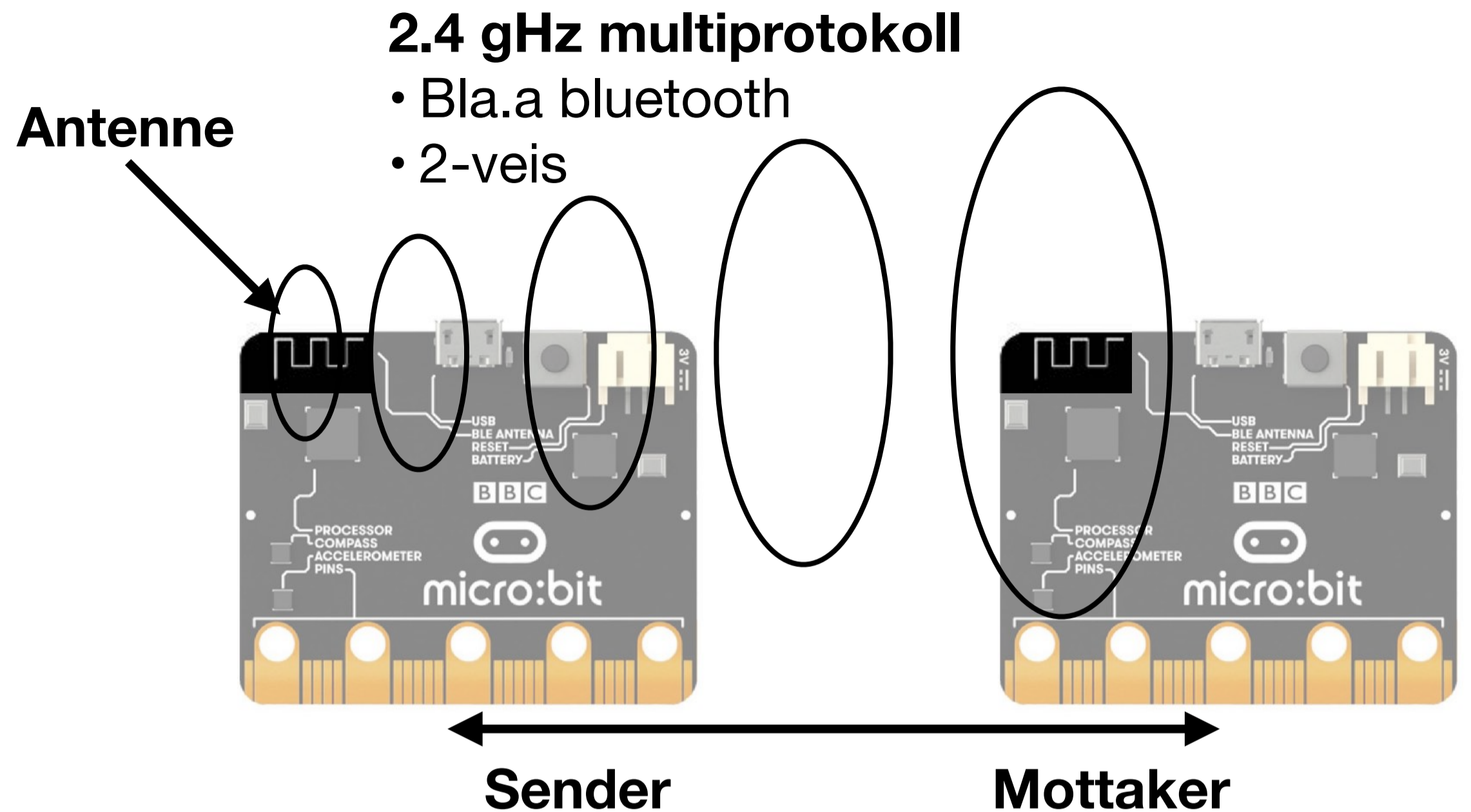
- setter yaw til -30 dersom man trykker til venstre (0 og 1)
- setter yaw til 30 om vi trykker til høyre (1 og 2)
- setter yaw til 0 om ingen strøm registreres (ikke noe trykk)
- Tips: Sett yaw to 0 først



## Bonusoppgave:

- Lag en trinnvis øking av yaw basert på hvor hardt vi trykker (hvor mye strøm som leses på pinnene). Utnytt at vi har 5 piksler langs x-aksen i displayet.

# Radio



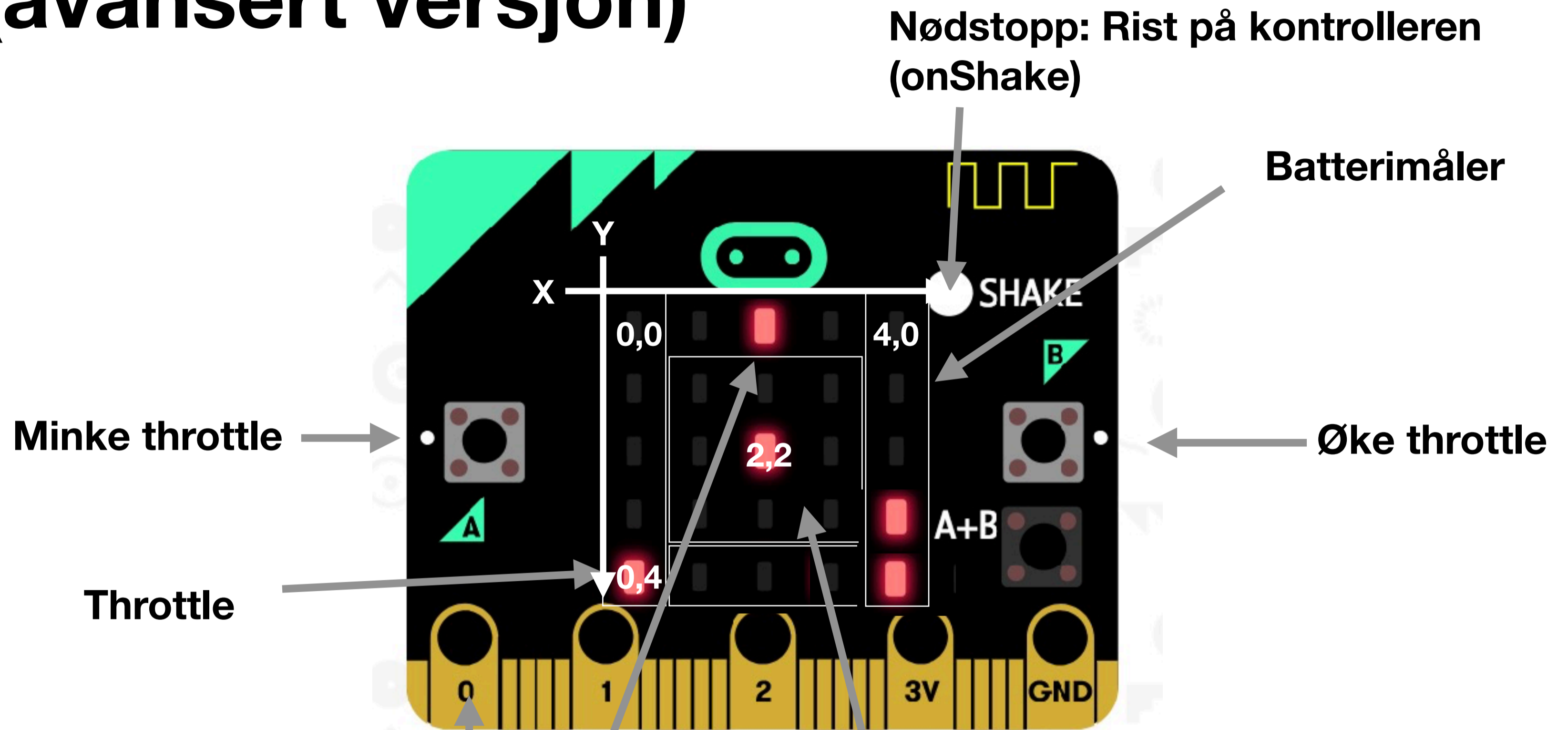
## Oppsett av radio

- Vi trenger normalt to micro:bit for å bruke radio, men micro:bit kan også kommunisere med mange bluetooth-enheter.
- Vi starter med å velge en radiogruppe på både sender og mottaker. Dette gjøres normalt kun i oppstarten av koden.
- Vi kan velge sendestyrke. 7 er kraftigst og gir oss 50-100 meter rekkevidde.
- For at mottaker skal vite hvilket tall vi sender, bruker vi en kombinasjon av tall og streng. Strengen (string) fungerer som en merkelapp på tallet vi sender.
- Mottakeren tolker tallene vi sender etter merkelappene **P,A,R,T,Y** (pitch, arm, roll, throttle, yaw)
- Vi må derfor bruke disse merkelappene sammen med de respektive verdiene.

## Sende radio:

- I on start, sett en unik radiogruppe og sendestyrken til 7 (maks)
- I bunnen av forever, send kombinasjon av bokstaver (string) og tall for verdiene. Vi må bruke merkelapper sammen med de respektive verdiene, f.eks "P" sammen med Pitch-verdien.
- Gjør dette for alle de 5 PARTY-variablene.

# Senderens funksjon (avansert versjon)



Nødstopp: Rist på kontrolleren  
(onShake)

Batterimåler

Minke throttle

Øke throttle

Throttle

Rotasjon (YAW):  
Legg tommelen mellom  
P0 og P1 (venstre), eller P1 og P2  
(høyre)

Forover, bakover, høyre, venstre  
(PITCH, ROLL)

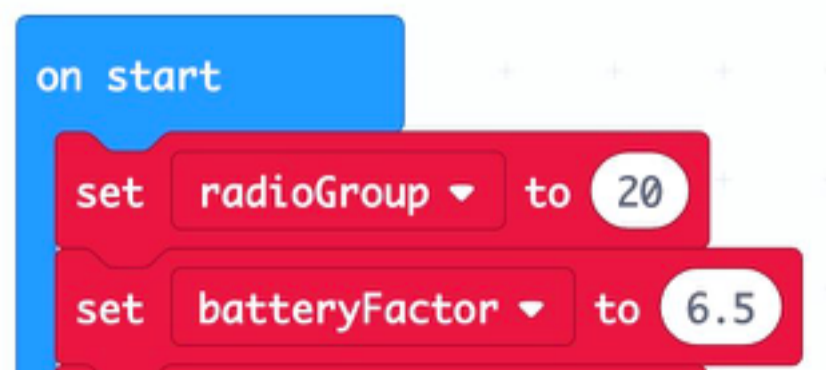


# Oppsummering av kode

- Vi har innhentet data med variablene Pitch, Arm, Roll, Throttle, Yaw (PARTY)
- Throttle er programmert til å økes og senkes med A / B
- Roll og pitch overføres fra input til variablene roll og pitch
- Yaw kan leses via strømpulser f.eks på pin 0 og 2
- Arm aktiveres og deaktiveres med A + B
- Når vi armerer og disarmerer bør throttle nullstilles
- Vi har visualisert variablene på skjerm med piksler
- Alle variablene sendes over radio i kombinasjon av forbokstav f.eks "P", og variabelen til tilhører.

## Kode til mottaker/quadkopter

- Vi må laste ned en egen kode til dronen. Denne mottar radiosignalene, styrer dronen, samt sørger for å lande ved lavt batteri. Du må sette rett radiogruppe og batterifaktor i denne koden før den lastes over.
- Batterifaktoren 6,5 om ikke annet er oppgitt. Denne brukes til å beregne batterispennning og landing ved lavt batteri.



Spørsmål? Ikke nøl med å kontakte oss!  
Bruk gjerne vår facebook-chat



[www.makekit.no](http://www.makekit.no)



henning@makekit.no



makekit



gomakekit (også twitter)