

# Bruksanvisning Bikubemonitor

## Innhold

1. Introduksjon .....	2
2. Komponentliste .....	2
3. Raspberry pi og Waveshare 4G-HAT (Ganske avansert) .....	2
3.1 Montering: .....	3
3.2 Sett opp kommunikasjon mellom Raspberry pi og Waveshare 4G-HAT: .....	4
3.3 Grunnleggende konfigurasjon .....	6
3.4 Konfigurer 4G-tilkoblingen med ModemManager .....	6
3.5 Konfigurer Wi-Fi-hotspot med hostapd og dnsmasq .....	7
3.5.1 Hostapd (Access Point) .....	7
3.5.2 dnsmasq (DHCP-server) .....	8
3.6 Ruting og NAT .....	9
3.7 Start tjenester automatisk ved oppstart .....	9
3.8 Test systemet .....	10
3.9 Feilsøking .....	10
4. Loding på kretskort eller bestilling av ferdig loddet kretskort .....	11
4.1 Alternativ 1. Lodde på komponenter selv .....	11
4.1.1 Bestilling av kretskort .....	12
4.2 Alternativ 2. Bestilling av ferdig loddet kretskort. ....	19
4.3 Innstilling av spenningsregulatorer .....	27
4.4 Påkobling og oppsett av sensorer .....	27
4.4.1 Temperaturføler .....	27
4.4.2 Vektcellene .....	28
5. Opplasting av kode til ESP (Wemos D1 Mini) .....	31
6. Koble på batteri og Raspberry Pi .....	50
7. Systemvedlikehold .....	53
7.1 Opplading og bytting av batteri .....	53
7.2 Omstart av system .....	53

# 1. Introduksjon

Dette er en detaljert bruksanvisning for systemet til bikuben. Her vil vi gå inn i hvert punkt å forklare hva som må gjøres og hvordan alt må settes opp for å få et fungerende system. Les hvert punkt nøye før du starter oppsettet. I bruksanvisningen finner du nødvendige filer som må lastes ned, men disse kan også finnes [HER](#) om det skulle være nødvendig.

## 2. Komponentliste

Komponentnavn	Beskrivelse	Antall	Link	
Raspberry pi 4B	Datamaskin	1	<a href="https://www.kjell.com/no/produkter/data/ettkortsdatamaskin/raspt">https://www.kjell.com/no/produkter/data/ettkortsdatamaskin/raspt</a>	
Waveshare 4G-HAT	4G-modul	1	<a href="https://www.aliexpress.com/item/1005002997084354.html?spm=a2g">https://www.aliexpress.com/item/1005002997084354.html?spm=a2g</a>	
ESP8266 D1 Mini	Mikrokontroller	1	<a href="https://www.kjell.com/no/produkter/elektro-og-verktoy/elektronikk/">https://www.kjell.com/no/produkter/elektro-og-verktoy/elektronikk/</a>	
DS18B20	Temp.sensor	1	<a href="https://www.kjell.com/no/produkter/elektro-og-verktoy/elektronikk/">https://www.kjell.com/no/produkter/elektro-og-verktoy/elektronikk/</a>	
Vektcelle + hx711	Vektcelle	1	<a href="https://www.aliexpress.com/item/4000215823944.html?spm=a2g0o.c">https://www.aliexpress.com/item/4000215823944.html?spm=a2g0o.c</a>	
Relé	Strømstyring	1	<a href="https://www.kjell.com/no/produkter/elektro-og-verktoy/elektronikk/">https://www.kjell.com/no/produkter/elektro-og-verktoy/elektronikk/</a>	
Batteri	Strøm	1	<a href="https://www.biltema.no/bil---mc/bilbatterier/blyakkumulatorer/blyb">https://www.biltema.no/bil---mc/bilbatterier/blyakkumulatorer/blyb</a>	
LM2596S	Step-down converter	2	<a href="https://www.aliexpress.com/item/1005006365697021.html?spm=a2g">https://www.aliexpress.com/item/1005006365697021.html?spm=a2g</a>	
Multimeter	Test	1	<a href="https://www.clasohlson.com/no/UNI-T-UT131B-multimeter/p/41-159">https://www.clasohlson.com/no/UNI-T-UT131B-multimeter/p/41-159</a>	
Koblingskabel Hann-hunn	Kabel	2	<a href="https://www.kjell.com/no/produkter/elektro-og-verktoy/elektronikk/">https://www.kjell.com/no/produkter/elektro-og-verktoy/elektronikk/</a>	
Kabel 1,5mm <sup>2</sup> 10m	Kabel	1	<a href="https://www.clasohlson.com/no/p/49-439-5">https://www.clasohlson.com/no/p/49-439-5</a>	

Komponentene kan finnes til lavere pris andre steder og noen kan komme med kretskortet, dette er bare for å vise mulige steder de kan kjøpes. (dobbeltklikk for lenke).

## 3. Raspberry pi og Waveshare 4G-HAT (Ganske avansert)

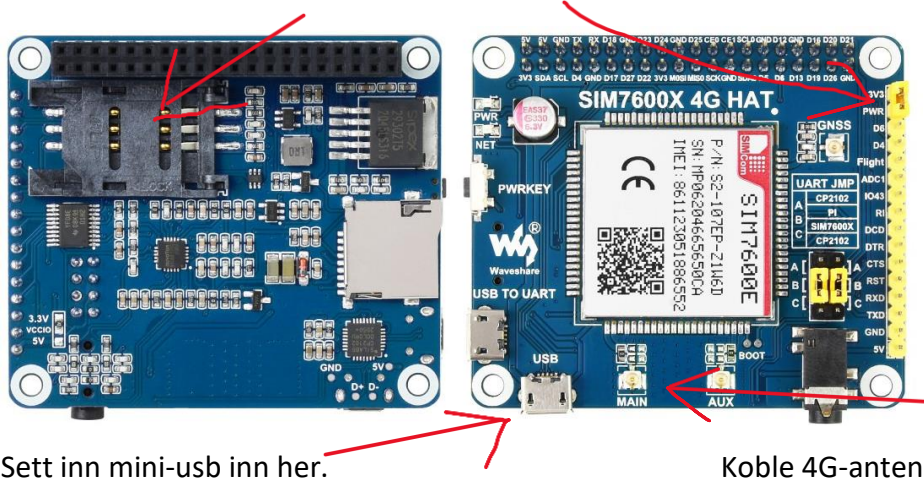
For å sette opp en Raspberry pi som et WiFi hotspot så må du skaffe dette utstyret.

- Raspberry Pi 4B (4GB eller 8GB RAM)
- Waveshare 4G-HAT (antenne og ledninger følger med)
- Strømforsyning til Raspberry pi (minst 5V/3A). Denne blir solgt utenom Raspberry pi.
- MicroSD-kort (minimum 16GB)
- HDMI til MINI-HDMI kabel eller overgang.
- SIM-kort (helst Telia eller Ice) hvis andre så sjekk opp APN til mobiloperatør.

- For SIM-kort så er det mulig å bruke tvilling-SIM men da må man være obs på at det kan være konflikter mellom kortene fordi bare et kort kan være aktivt om gangen for bruk av data. Vi ser ikke på dette som et stort problem siden ruterer ikke er aktiv over lengre perioder.

### 3.1 Montering:

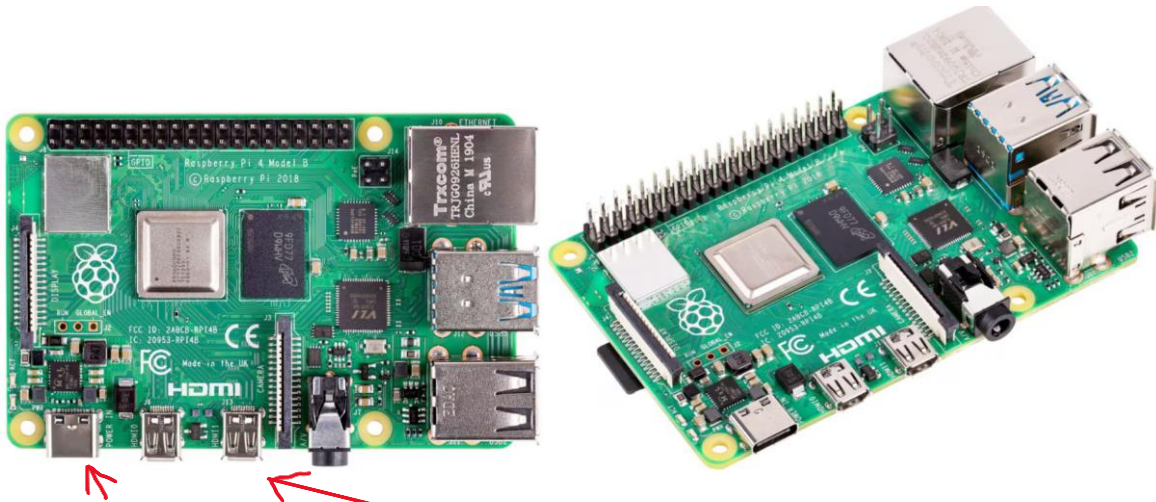
1. Sett først inn ett sim-kort uten pin kode og flytt denne tappen til PWR og D6 for at den skal starte automatisk.



2. Sett den på GPIO pinnene til Raspberry pi sånn at den ser slik ut. Og koble mini-usb-til-usb ledningen i usb-porten.



3. Koble MicroSD-kortet i en pc og laste ned Raspberry Pi Imager fra raspberrypi.com og flashe minnekortet via imager for å laste inn operativsystemet: Raspberry Pi OS. Når det er gjort så tar du minnekortet og setter det inn i Raspberry pi'en og kobler til tastatur og mus til usb-portene og mini HDMI-HDMI i en skjerm og koble til strøm via stikkontakt og usb-c porten.




Dette er strømninggang(usb-c) og dette er mini-HDMI.

Koble til hjemnettverket for å kunne oppdatere og laste ned nødvendige pakker.

### 3.2 Sett opp kommunikasjon mellom Raspberry pi og Waveshare 4G-HAT:

1. Når hjemskjermen kommer opp så åpner du et terminalvindu (svart firkant oppe i venstre

hjørnet). 

2. Skriv inn i terminalvinduet: `sudo raspi-config` (trykk enter).

Da popper det opp et nytt vindu hvor du får flere alternativer, velg alternativ 3 Interface Options. Deretter velger du I6 Serial Port (trykk enter).

I det nye vinduet som popper opp: Would you like a login shell to be accessible over serial? Velg NO (trykk enter).

I det neste vinduet som popper opp: Would you like serial port hardware to be enabled? Velg YES (trykk enter).

Da får du opp et nytt vindu: Would you like to reboot now? Velg YES (trykk enter).

3. Når Raspberry pi starter opp igjen, åpne et nytt terminalvindu og skriv inn:

`sudo apt-get install minicom`, hvis det kommer opp [Y/N] eller [Y/N] når den laster ned trykk: J eller Y utifra hvilket språk du har den på.

4. Fortsett i terminalvinduet og skriv inn:

`wget https://www.waveshare.com/w/upload/2/29/SIM7600X-4G-HAT-Demo.7z`

5. Når den er ferdig skriv inn:

`sudo apt-get install p7zip-full`

OBS! sjekk filstien, den kan variere hvor jeg har lagt inn ???, så sjekk over at den blir lastet ned til riktig sti: `/home/???/pi/...`

Sørg for at disse er riktig på alle under.

6. Skriv inn:

`7z x SIM7600X-4G-HAT-Demo.7z -r -o/home/pi`

7. Deretter dette:

`sudo chmod 777 -R /home/pi/SIM7600X-4G-HAT-Demo`

8. Deretter skriv inn:

`sudo nano /etc/rc.local`

9. Inne i denne filen legger du inn følgende:

```
_IP=$(hostname -I) || true
```

```
if [ «$_IP» ]; then
```

```
    printf «My IP address is %s\n» «$_IP»
```

```
fi
```

```
sh /home/pi/SIM7600X-4G-HAT-Demo/Raspberry/c/sim7600_4G_hat_init
```

```
exit 0
```

for å komme ut av dette vindu trykker du `ctrl + x`, da popper det opp om du ønsker å lagre, trykk lagre med Y og trykk enter.

Lukk deretter terminalvinduet.

OBS! sjekk filstien, den kan variere hvor jeg har lagt inn ???, så sjekk over at den blir lastet ned til riktig sti: /home/???/pi/...

Sørg for at disse er riktig på alle under.

10. Åpne et nytt terminalvindu og skriv inn:

```
cd /home/pi/SIM7600X-4G-HAT-Demo/Raspberry/c/bcm2835
```

også skriv inn:

```
chmod +x configure && ./configure && sudo make && sudo make install
```

Lukk terminalvinduet.

Når den er ferdig da er waveshare 4G-HAT og raspberry pi satt opp sånn at de kan kommunisere sammen.

### 3.3 Grunnleggende konfigurasjon

1. Åpne et nytt terminalvindu og skriv inn følgende for å oppdatere systemet:

```
sudo apt update && sudo apt upgrade -y
```

2. Når den er ferdig skriv inn for å installere nødvendig verktøy:

```
sudo apt install -y vim git curl dnsmasq hostapd iptables
```

Lukk terminalvinduet

### 3.4 Konfigurer 4G-tilkoblingen med ModemManager

1. Åpne et nytt terminalvindu og installer ModemManager:

```
sudo apt install -y modemmanager
```

2. Sjekk om modemmet blir gjenkjent (forventer noe som

```
«/org/freedesktop/ModemManager1/Modem/0»):
```

```
mmcli -L
```

3. Opprett en 4G-tilkobling med NetworkManager (etter gsm.apn må du legge inn APN til mobiloperatøren din) (Telia og Ice har «internet»):

```
sudo nmcli connection add type gsm ifname «*» con-name «4G-forbindelse» \ apn «internet»
```

4. Koble fra WiFi tilkoblingen:

```
nmcli radio wifi off
```

5. Aktiver 4G-tilkoblingen:

```
sudo nmcli connection up «4G-forbindelse»
```

6. Sjekk om den er aktiv (se etter «4G-forbindelse»):

```
nmcli connection show --active
```

7. Test tilkoblingen (trykk ctrl+c for å stoppe):

```
ping -c 4 8.8.8.8
```

Lukk terminalvinduet.

### 3.5 Konfigurer Wi-Fi-hotspot med hostapd og dnsmasq

#### 3.5.1 Hostapd (Access Point)

1. Åpne et nytt terminalvindu for å opprette en konfigurasjonsfil (her kan du velge SSID selv og passord):

```
sudo nano /etc/hostapd/hostapd.conf
```

2. Legg inn følgende i denne filen:

```
interface=wlan0
```

```
driver=nl80211
```

```
ssid=BiKubeWifi
```

```
hw_mode=g
```

```
channel=6
```

```
wmm_enabled=0
```

```
macaddr_acl=0
```

```
auth_algs=1
```

```
ignore_broadcast_ssid=0
```

```
wpa=2
```

```
wpa_passphrase=B1kubem0nitorering
```

```
wpa_key_mgmt=WPA-PSK
```

```
wpa_pairwise=TKIP
```

```
rsn_pairwise=CCMP
```

trykk ctrl+x så y også enter

3. Når du kommer ut igjen til terminalvinduet så endrer du rettigheter:

```
sudo chmod 600 /etc/hostapd/hostapd.conf
```

Fortell systemet hvor konfigurasjonen ligger (bla med piltastene i filen til du finner

```
#DAEMON_CONF=):
```

```
sudo nano /etc/default/hostapd
```

endre denne til (fjern # foran):

```
DAEMON_CONF= «/etc/default/hostapd/hostapd.conf»
```

trykk ctrl+x så y også enter

### 3.5.2 dnsmasq (DHCP-server)

1. Lag en sikkerhetskopi av originalen:

```
sudo mv /etc/dnsmasq.conf /etc/dnsmasq.conf.original
```

2. Opprett ny konfigurasjon:

```
sudo nano /etc/dnsmasq.conf
```

3. Legg inn dette i filen som åpner seg:

```
interface=wlan0
```

```
dhcp-range=192.168.50.10,192.168.50.100,255.255.255.0,24h
```

```
dhcp-option=3,192.168.50.1 # Gateway
```

```
dhcp-option=6,8.8.8.8,8.8.4.4 # DNS-servere
```

```
server=8.8.8.8
```

```
no-resolv
```

```
log-queries
```

```
log-dhcp
```

trykk ctrl+x så y også enter

4. Start tjenesten på nytt :

```
sudo systemctl restart dnsmasq
```



### 3.6 Ruting og NAT

1. Aktiver IP-forwarding (bla til du finner (net.ipv4.ip\_forward=1) og fjern # foran):

```
sudo nano /etc/sysctl.conf
```

trykk ctrl+x så y også enter

2. For å lagre endringene:

```
sudo sysctl -p
```

3. Legg til NAT-regler med iptables:

```
sudo iptables -t nat -A POSTROUTING -o wwan0 -j MASQUERADE
```

```
sudo iptables -A FORWARD -i wlan0 -o wwan0 -j ACCEPT
```

4. Lagre reglene permanent:

```
sudo apt install iptables-persistent -y
```

```
sudo iptables-save | sudo tee /etc/iptables/rules.v4
```

### 3.7 Start tjenester automatisk ved oppstart

1. Aktiver hostapd og dnsmasq:

```
sudo systemctl unmask hostapd
```

```
sudo systemctl enable hostapd dnsmasq
```

2. Opprett en system-tjeneste for å sikre riktig oppstartsrekkefølge:

```
sudo nano /etc/systemd/system/hotspot.service
```

3. Legg inn følgende i denne filen:

```
[Unit]
```

```
Description=Start Wi-Fi Hotspot
```

```
After=network.target
```

```
[Service]
```

```
Type=oneshot
```

```
ExecStart=/usr/sbin/hostapd -B /etc/hostapd/hostapd.conf
```

```
ExecStart=/usr/sbin/dnsmasq -C /etc/dnsmasq.conf
```

```
[Install]
```

```
WantedBy=multi-user.target
```

trykk ctrl+x så y også enter

#### 4. Aktiver tjenesten:

```
sudo systemctl daemon-reload
```

```
sudo systemctl enable hotspot
```

### 3.8 Test systemet

#### 1. Sørg for at den ikke kobler seg til WiFi(ditt WiFi-navn) automatisk:

```
nmcli connection modify «WiFi-navn» connection.autoconnect no
```

#### 2. Reboot Raspberry pi:

```
sudo reboot
```

Den skal nå starte alt opp automatisk og WiFi hotspotet skal være klart. Du skal nå kunne finne BiKubeWifi med andre enheter.

### 3.9 Feilsøking

For feilsøking hvis det er noe som feiler, kan du skrive inn følgende i terminalvinduet:

#### 1. for modem (waveshare):

```
mmcli -m 0
```

Se etter «signal quality»

#### 2. For DHCP-feil skriv inn i terminalvinduet:

```
sudo systemctl status dnsmasq
```

Se etter status (active) og running

#### 3. For hostapd, skriv inn i terminalvinduet:

```
sudo systemctl status hostapd
```

Se etter status (active) og running

## 4. Lodding på kretskort eller bestilling av ferdig loddet kretskort

Vi har vurdert to ulike løsninger for oppsettet av kretskortet, hver med sine egne fordeler og ulemper:

### 1. Bestille kretskort og komponenter separat:

Denne løsningen innebærer at man bestiller selve kretskortet og komponentene hver for seg, og deretter lodder komponentene på manuelt. Dette gir stor fleksibilitet og kontroll over både produksjonsprosessen og valg av komponenter. En utfordring med denne metoden er at mange komponenter ofte må kjøpes i bulk (f.eks. 10 eller 100 stk), noe som kan føre til overskudd dersom man kun skal lage et fåtall kort. Samtidig gir dette også en fordel i form av reservekomponenter, som kan være nyttige dersom man mister eller skader komponenter under montering.

### 2. Bestille ferdig loddet kretskort (montert PCB):

Den andre løsningen er å bestille et ferdig loddet kretskort fra en leverandør. Denne prosessen er mer omfattende og krever god innsikt i produksjonsflyten. Det er nødvendig med tett kommunikasjon med produsenten, da det må utveksles produksjonsfiler (som Gerber-filer og BOM-lister), samt godkjenning av eventuelle alternative komponenter, forslag til omplasseringer og andre produksjonsrelaterte avklaringer. Det er viktig å merke seg at selv om kretskortet blir levert delvis montert, er det ingen garanti for at *alle* komponenter loddas på automatisk. Derfor må man være forberedt på å ettermontere enkelte komponenter selv.

Du må selv vurdere valget du vil ta her.

### 4.1 Alternativ 1. Lodde på komponenter selv.

Loddeprosessen begynner etter at du har mottatt kretskortet og samlet alle nødvendige komponenter. Det er viktig at du merker deg plasseringen av hver komponent før du begynner å lodde. Feil plassering kan føre til at kretskortet ikke fungerer som det skal.

For å sette opp kretskortet trenger du følgende:

## Komponentliste

- Kretskort (bestilles via EasyEDA, se fremgangsmåte under)
- Elektroniske komponenter (motstander, kondensatorer, pinner etc.)
- Loddebolt og loddetinn

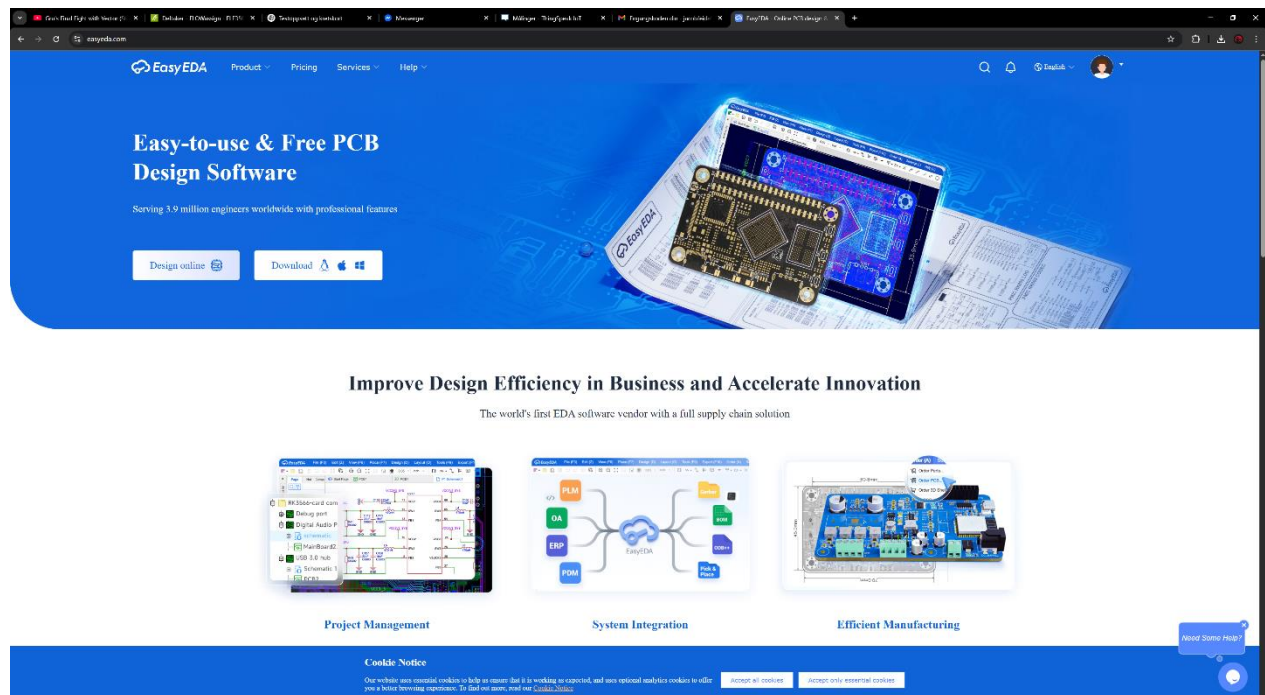
**Viktig:** Det er kritisk å ha alle komponentene tilgjengelig før du bestiller og setter opp kretskortet. Loddearbeidet krever presisjon og tålmodighet, men prosessen er overkommelig også for nybegynnere.

### 4.1.1 Bestilling av kretskort

Her er steg-for-steg hvordan du importerer og bestiller kretskortet: <https://easyeda.com/>

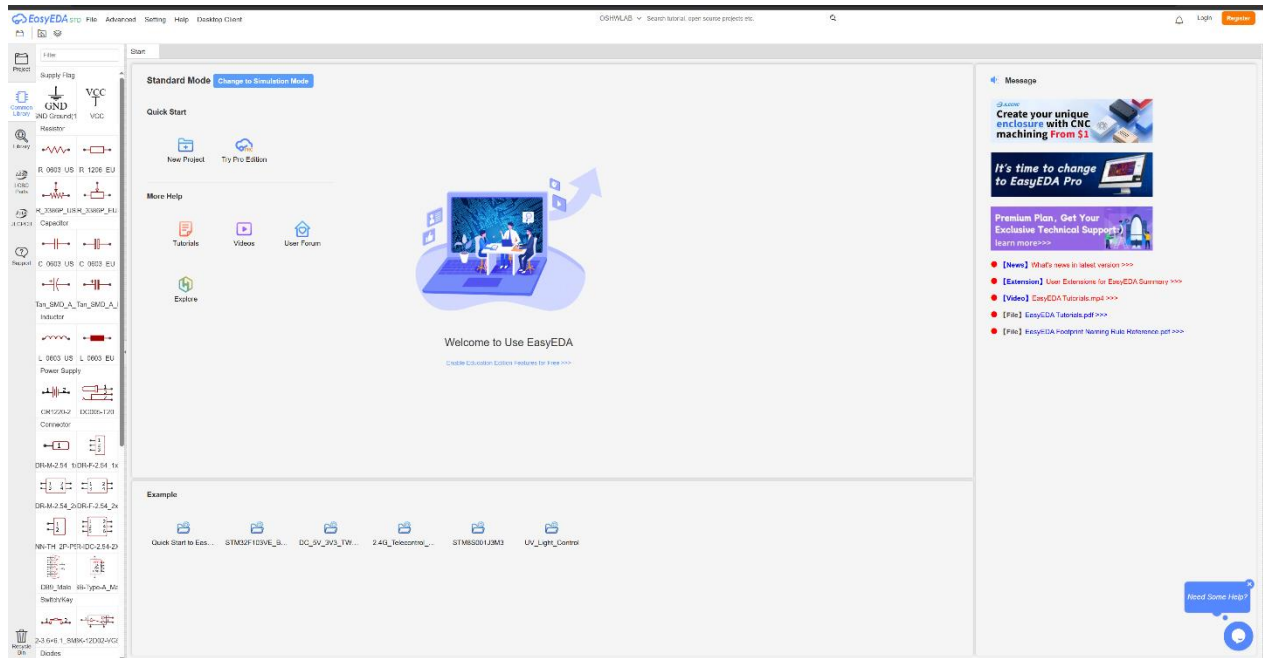
#### 1. Gå til EasyEDA

Her kommer du til en nettside som ser slik ut



#### 1. Trykk på Design Online knappen.

2. Trykk på Std Edition. Etter dette skal det se slik ut



### 3. Logg inn

- Klikk på "PROJECT" oppe i venstre hjørne.
- Deretter på "Logg inn" lag en gratis konto om du ikke har en.

### 4. Last ned prosjektfilen



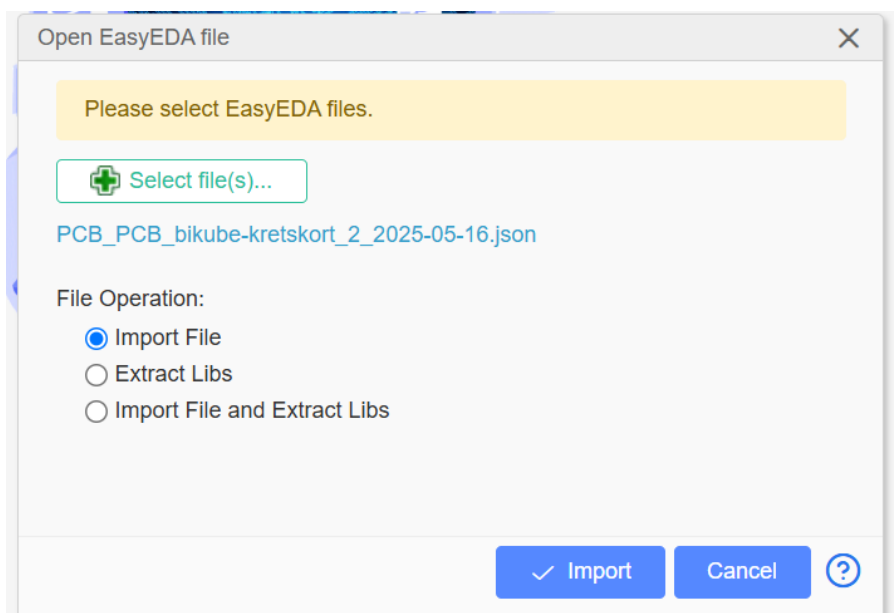
PCB\_PCB\_bikube-kr

- Last ned kretskort-filen vi har laget. (etskort\_2\_2025-05-1)
- Husk hvor du lagrer den på PC-en din.

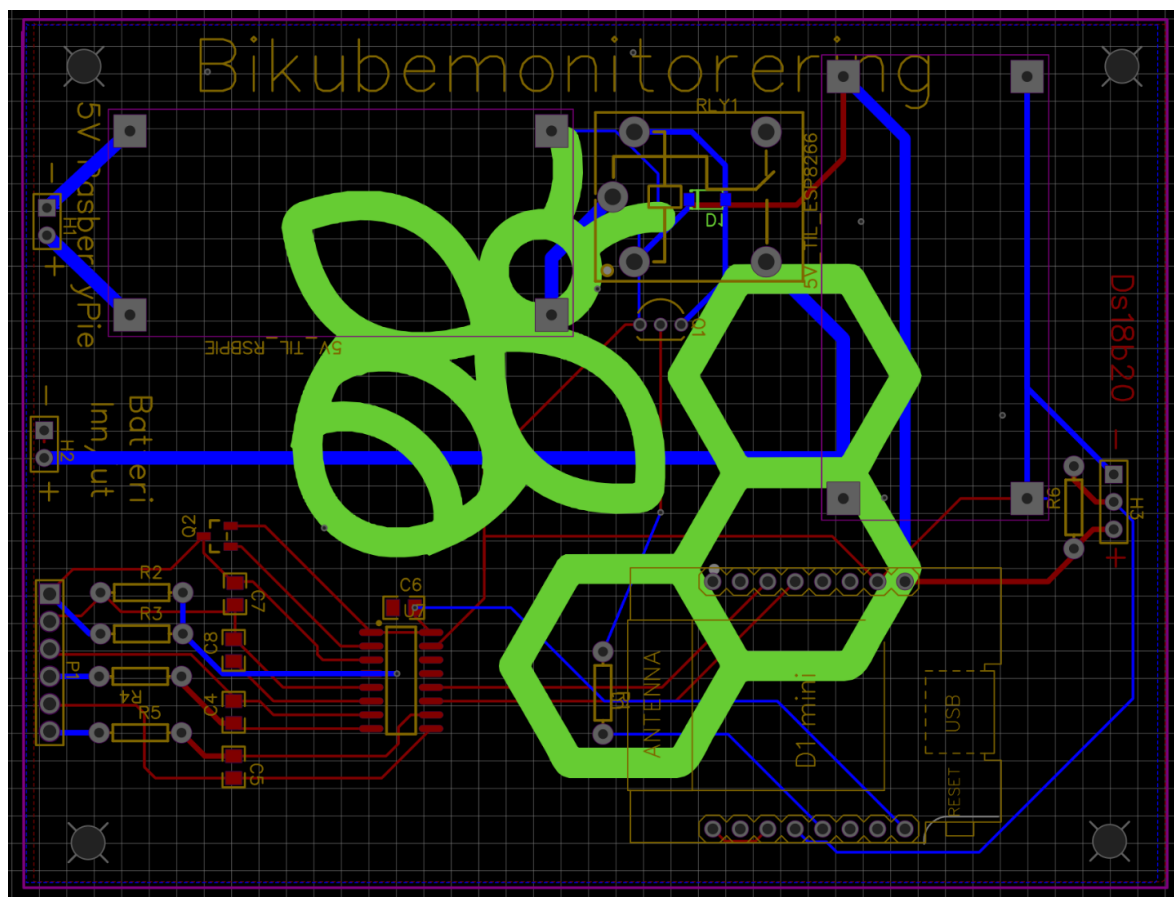
### 5. Importer filen i EasyEDA

- I EasyEDA, gå til File > Open > EasyEDA File.
- Velg filen du nettopp lastet ned.

c. Et nytt vindu vil åpnes – trykk "Import".



d. Du skal nå se kretskortdesignet på skjermen.



## 6. Lagre prosjektet

- Trykk Ctrl + S på tastaturet.
- Gi prosjektet et navn, f.eks. "New Project", og lagre.

## 7. Bestill kretskortet

- Klikk Fabrication > One-click Order PCB/SMT.
- Trykk "No, Continue to Order", deretter "OK".
- Når bestillingssiden åpner seg, trykk på "Save to cart".

The screenshot displays the JLCPCB website's PCB configuration interface. At the top, there are navigation tabs for 'Standard PCB/PCBA', 'Advanced PCB/PCBA', 'SMT-Stencil', and '3D Printing/CNC'. The main area shows a 3D model of a PCB and a list of specifications. On the right, a 'Charge Details' sidebar lists costs for engineering, surface finish, and build time. A 'Calculated Price' of \$7.30 is shown, along with a 'Shipping Estimate' of \$21.53. A 'SAVE TO CART' button is prominently displayed. The bottom right corner features a chatbot icon and a welcome message.

- Fullfør bestillingen.

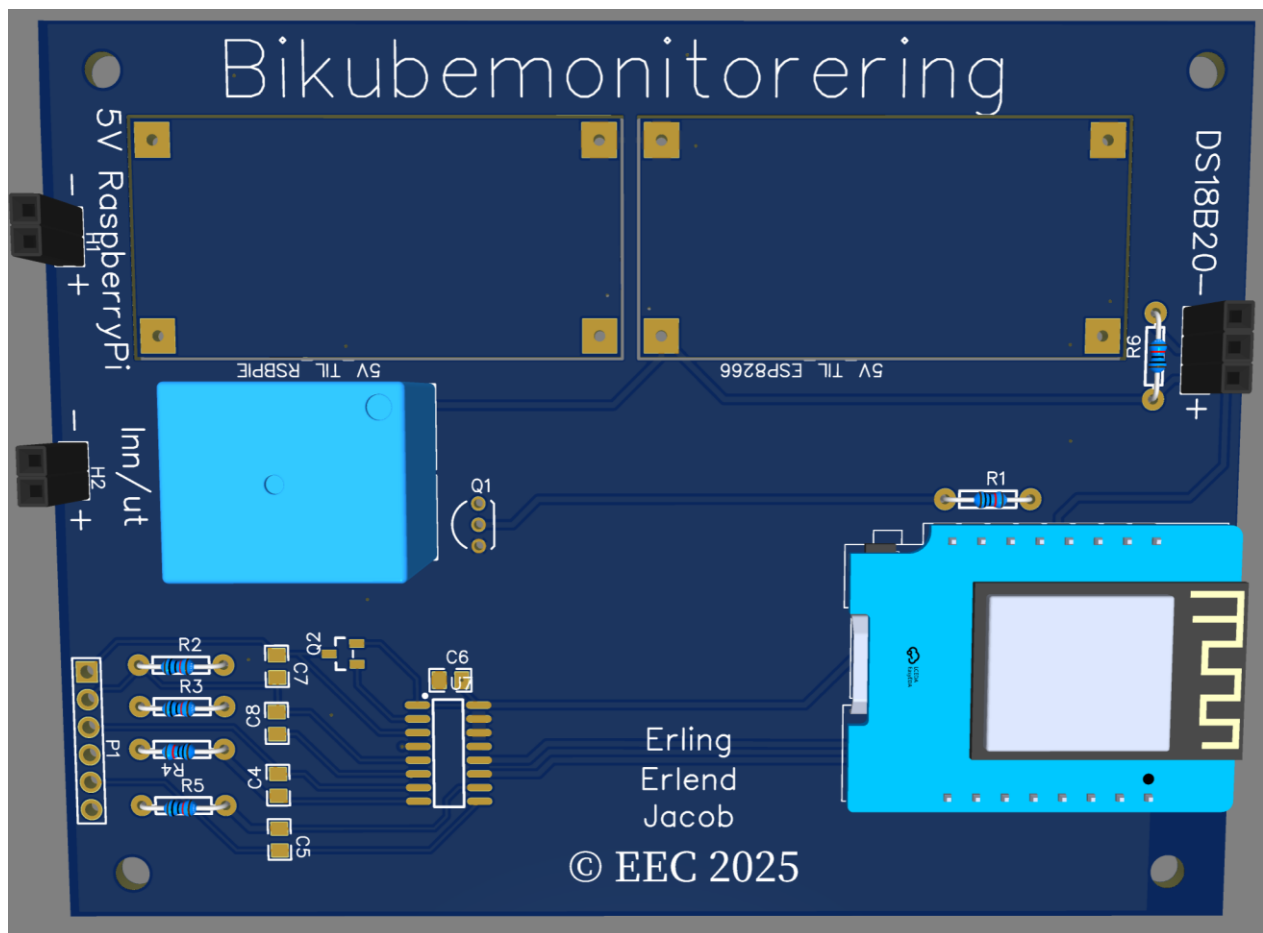
## Viktige punkter før du lodder

- Finn frem oversikten over komponentene – enten som en tabell eller et skjema.
- Sammenlign komponentnavn og plassering med merkingen på kretskortet (f.eks. "R1", "Q1", "D2", etc.).
- Tørrplasser komponentene på kretskortet først for å dobbeltsjekke at alt passer.

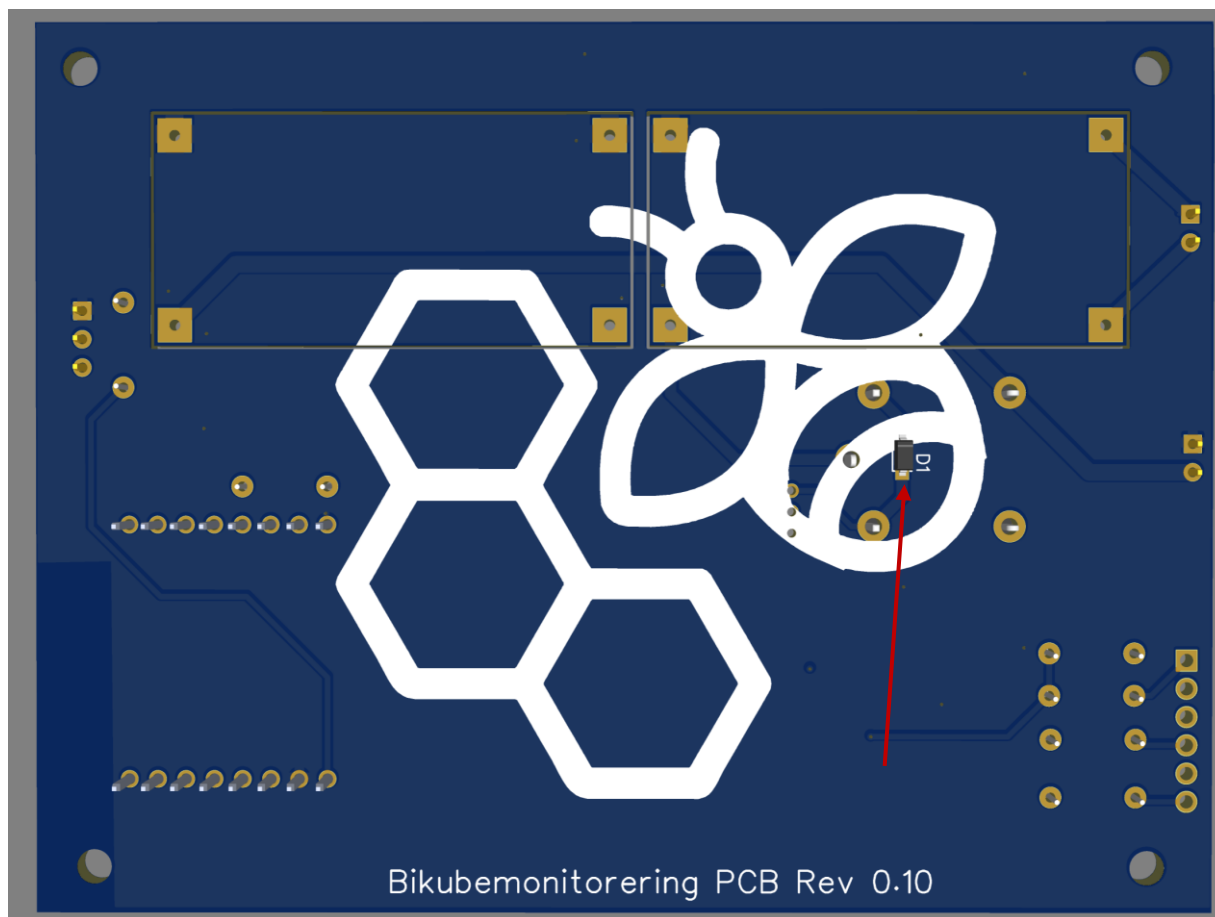
4. Dobbeltsjekk polaritet på komponenter som dioder, kondensatorer og buck converters – disse må loddess i riktig retning.
5. Se modellen/illustrasjonen under for å visualisere hvordan sluttresultatet skal se ut.

### Modell og oversikt

Under er en modell som viser den fysiske plasseringen av komponentene på kretskortet. De store rektanglene representerer plasseringen til Buck Converterne.







(merk D1 den eneste komponenten som skal loddess på baksiden av kretskortet.)

B	C	D	E
Name	Designator	Footprint	Quantity
4.7k	R6	R_AXIAL-0.3	1
8.2k	R2	R_AXIAL-0.3	1
20k	R3	R_AXIAL-0.3	1
1.2k	R4,R5	R_AXIAL-0.3	2
0.1uF	C4,C5,C8	0805	3
10uF	C6,C7	0805	2
1k	R1	R_AXIAL-0.3	1
LM2596DC-DC	5V_TIL_ESP8266,5V_TIL_RSBPiE	LM2596 DC-DC	2
Inn UT til Pie	H1	HDR-F-2.54_1X2	1
Inn UT Bat	H2	HDR-F-2.54_1X2	1
DS18B20 innganger	H3	HDR-F-2.54_1X3	1
Header-Female-2.54_1x6	P1	HDR-6X1/2.54	1
2N7000	Q1	2N7000	1
S9015	Q2	SOT-23(SOT-23-3)	1
5V-12V	RLY1	RELAY-TH_SRD-XXVDC-XL-C	1
WEMOS_D1_MINI_PRO4	U1	DIP-16_WEMOS_D1_MINI_PRO4	1
HX711	U7	SOP-16_150MIL	1
1N4007W	D1	SOD-123_L2.8-W1.8-LS3.7-RD	1

**Eksempel:** Dersom det i tabellen står Q1 i kolonne C, skal du lodde komponenten Q1 på det området på kretskortet som er merket med Q1.

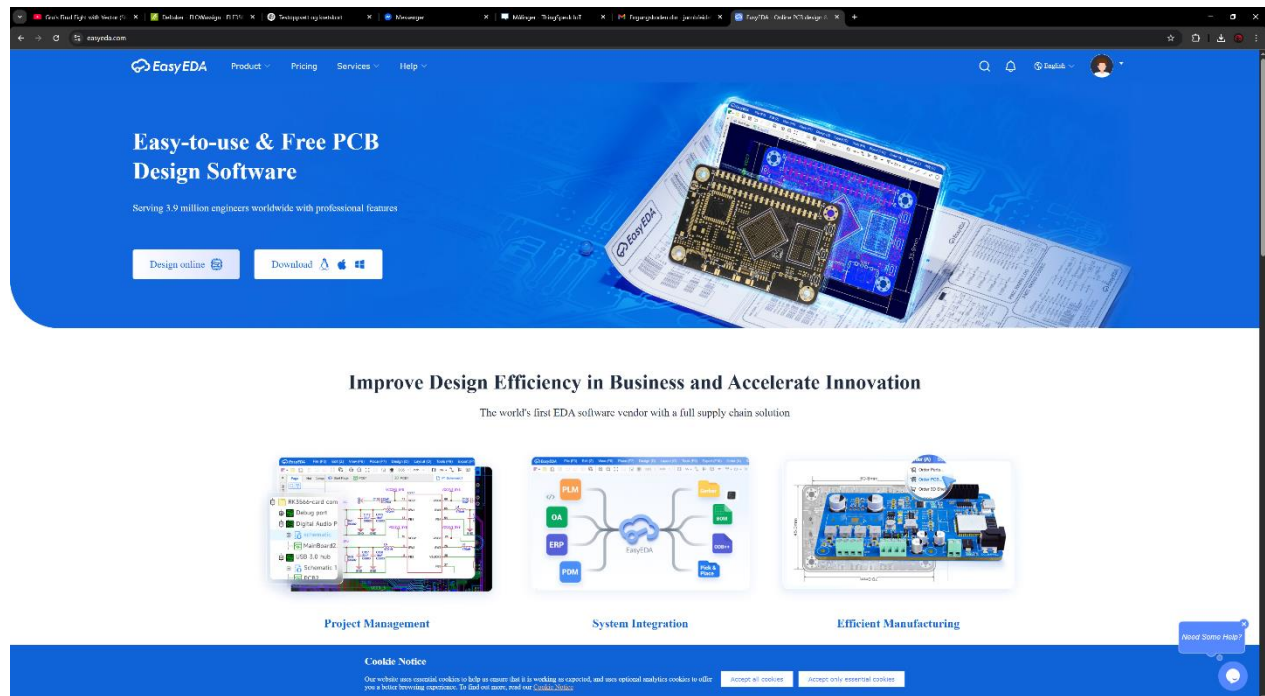
#### Tips:

- Bruk gjerne en komponentholder eller tape for å holde små komponenter på plass mens du lodder.
- Lodde en og en komponent, og ta deg god tid. Dette sikrer et stabilt og fungerende kretskort.

## 4.2 Alternativ 2. Bestilling av ferdig loddet kretskort.

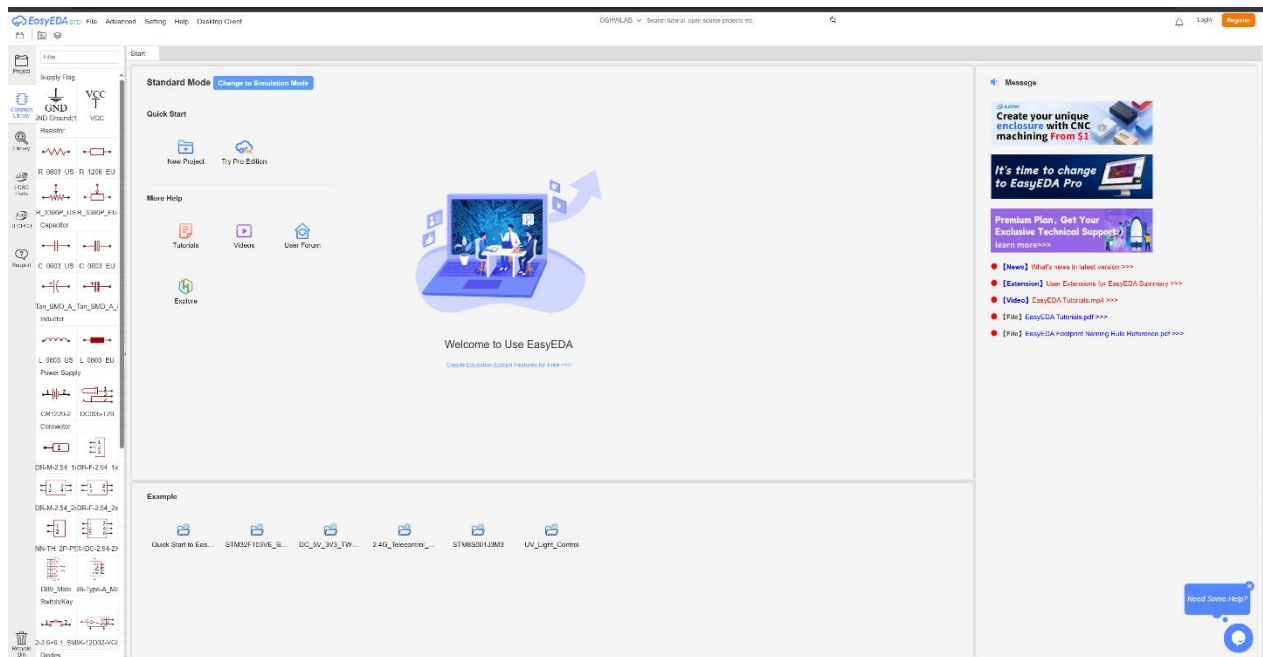
Gå til EasyEDA: <https://easyeda.com/>

Her kommer du til en nettside som ser slik ut



1. Trykk på Design Online knappen.

2. Trykk på Std Edition. Etter dette skal det se slik ut



### 3. Logg inn

- Klikk på "PROJECT" oppe i venstre hjørne.
- Deretter på "Logg inn" lag en gratis konto om du ikke har en.

### 4. Last ned prosjektfilen



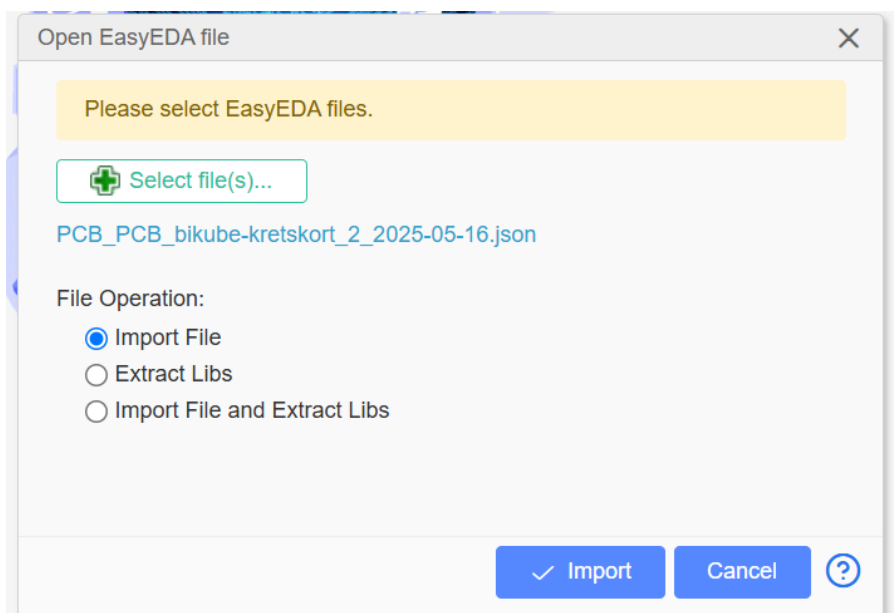
PCB\_PCB\_bikube-kr  
etskort\_2\_2025-05-1)

- Last ned kretskort-filen vi har laget. (
- Husk hvor du lagrer den på PC-en din.

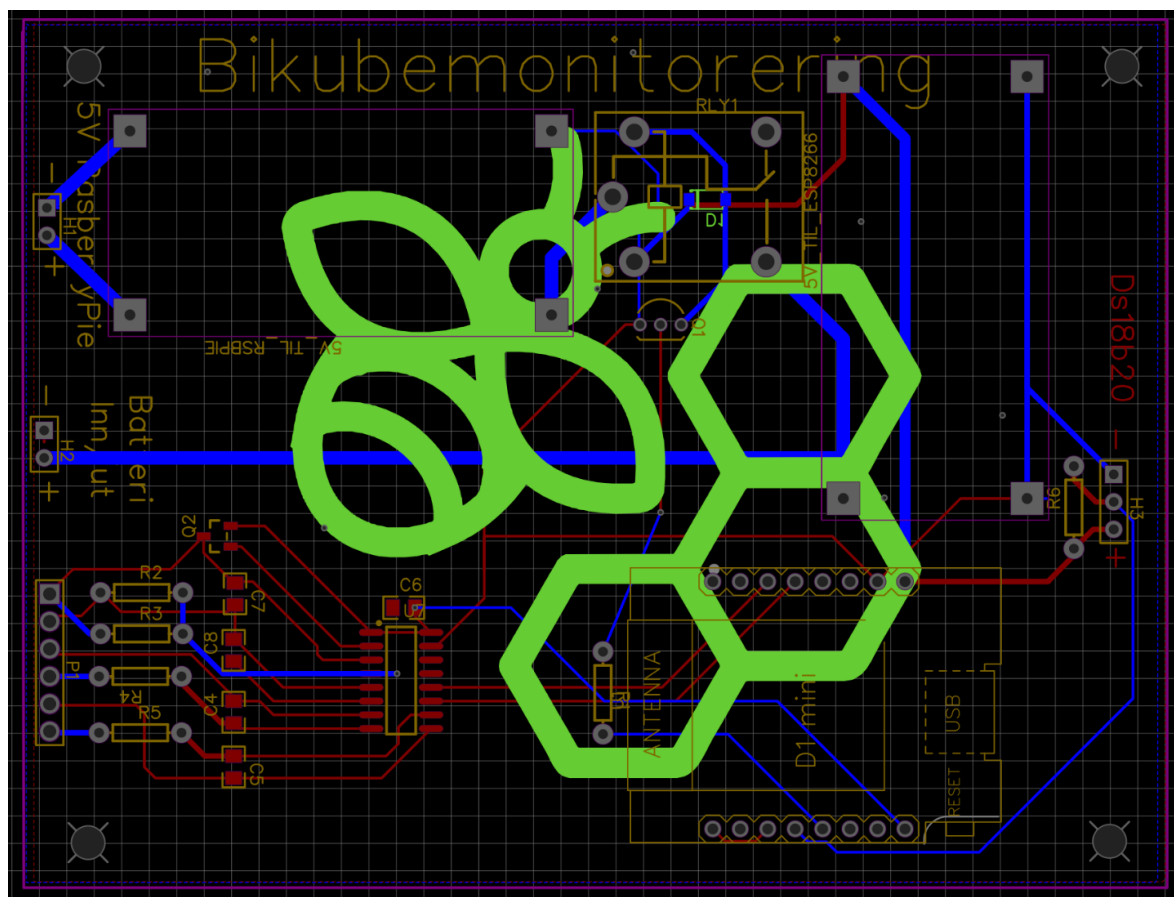
### 5. Importer filen i EasyEDA

- I EasyEDA, gå til File > Open > EasyEDA File.
- Velg filen du nettopp lastet ned.

c. Et nytt vindu vil åpnes – trykk "Import".



d. Du skal nå se kretskortdesignet på skjermen.




## 6. Lagre prosjektet

- a. Trykk Ctrl + S på tastaturet.
- b. Gi prosjektet et navn, f.eks. "New Project", og lagre.

## 7. Generering av filer

- a. Oppe på midten av skjermen trykk på filikonet med en B på.



- b. Klikk på «eksport BOM».
- c. Klikk på «eksport».
- d. Husk filplasseringen denne må tas i bruk senere.
- e. Gå tilbake til de tre filikonene.
- f. Klikk på nå på dette dette ikonet  .
- g. Klikk på «Export»
- h. Husk filplasseringen denne må tas i bruk senere.

## 8. Bestill kretskortet

- a. Klikk Fabrication > One-click Order PCB/SMT.
- b. Trykk "No, Continue to Order", deretter "OK". Nettsiden vil nå se slik ut:


c. Gå lengre ned på siden og huk av denne:



(den skal bli blå)


Du vil nå få en rekke valg.

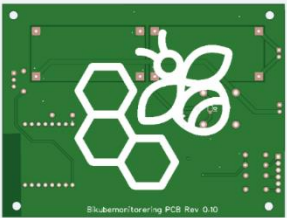
- På PCBA Type Klikk Standard.
- Assembly Side : Both sides.
- Edge Rails/Fiducials: Added by JLCPCB.
- Confirm Part Placement: No
- Parts Selection: Bt JLCPCB.

Det skal se slik ut når du er ferdig:


**PCB Assembly**


**Assembly cost starting from \$0 with coupon**








☒ Assemble top side

☒ Assemble bottom side


**PCBA Type**



Economic



Standard

[What's the difference?](#)


**Assembly Side**


Top Side


Bottom Side


Both Sides


**PCBA Qty**



5

Custom Qty

Qty from 2 to 5


**Edge Rails/Fiducials**



Added by JLCPCB


Added by Customer


The board size is modified to be 105.91mm\*90.24mm due to adding two 5mm edge rails on the shorter sides. [Learn More>](#)


**Confirm Parts Placement**


No


Yes

**Parts Selection**


By Customer


By JLCPCB

- Klikk på «Next»
- Klikk på «Add BOM File». Her legger du til filen du genererte tidligere som skal være tydelig merket med BOM i filnavnet.
- Klikk på «Add CPL File» her legger du til filen du genererte tidligere som skal være tydelig merket med Pick\_And\_Place i filnavnet.
- I produkt Description finn og klikk «Sensor\Controller\Precision Instrument / Temperature Sensor - HS Code 902590».



Det skal slik ut når du er ferdig:

#### Upload BOM & CPL files



You will receive an email with the part selection results within 1 - 2 business days. Please ensure that the selected parts meet your requirements.  
Note: We will use your private inventory to match parts. If certain parts are out of stock and no alternatives are available in the JLCPCB Parts Library, they will remain unmatched. You will need to purchase these parts and wait for their arrival before they can be matched to your order.



✓ BOM\_bikube-kretskort\_2025-05-16.csv

Only accept XLS,XLSX,CSV. [View Sample BOM](#)



✓ PickAndPlace\_PCB\_bikube-kretskort\_2\_202...

Pick&Place File, Only accept XLS,XLSX,CSV. [View Sample CPL](#)

#### Product Description ?

Sensor\Controller\Precision Instrument / Temperature Sensor -....

#### Assembly Notes

Leave a remark if you have any request

0/500



Upload file ?

☒ I agree to [the Terms and Conditions of JLCPCB Parts Selection Service](#)

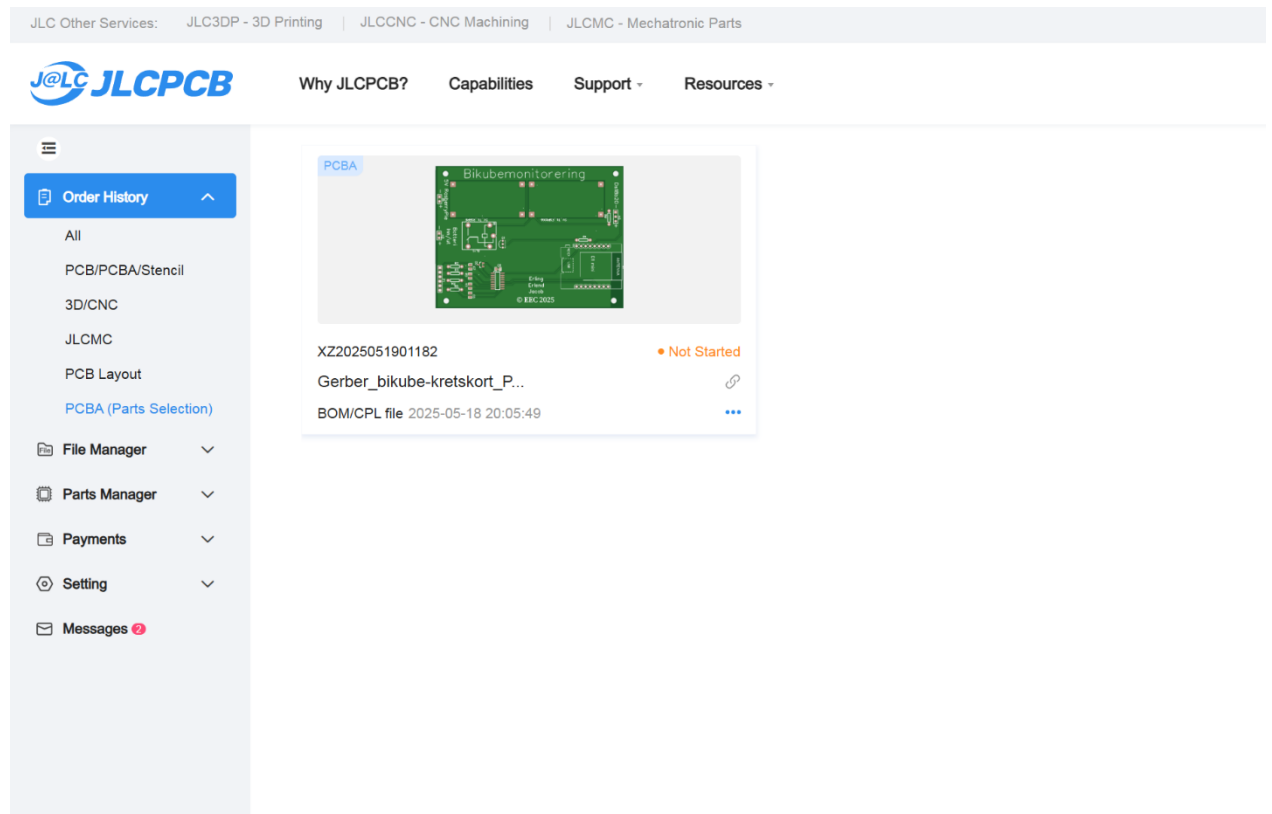
Parts selection by JLCPCB requires \$10 service fee.

Submit

Cancel

e. Klikk på «Submit».

- f. Fortsett nå med betalingsmetode og adresse for levering. Når du er ferdig skal det se slik ut:



Nå er bestillingen fullført, og du må følge jevnlig med på oppdatering fra leverandør med følge spørsmål. Denne prosessen skal ikke ta mer enn en uke. Det er viktig å merke seg hvilke komponenter som er blitt tatt med og hvilke du må lodde på selv senere. Dette skal være med i bekreftelsen av sending på kretskortet. Disse delene må du selv bestille i etterkant. Du kan finne de nødvendige komponentene i komponent listen for kretskortet.

**Viktig:** Vi anbefaler på det sterkeste at du enten spesifiserer at du ønsker ferdigjusterte spenningsregulatorer (Buck Converter/LM2596S) ved bestilling. Dette kan enkelt oppgis når du kommuniserer med leverandøren. Alternativt kan du bestille regulatorene separat og justere spenningen selv.

Begge regulatorene skal justeres til 5V.

I neste steg viser vi deg hvordan du trygt og enkelt kan stille inn spenningen selv.

### 4.3 Innstilling av spenningsregulatorer

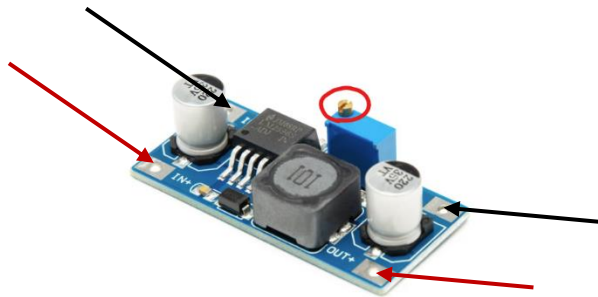
Spenningsregulatorene som skal festes til kretskortet må justeres til korrekt spenning før de loddas fast. Dette er et **viktig steg**, da feil spenningstilførsel kan føre til **ødeleggelse av komponentene** på kretskortet.

Begge spenningsregulatorene må justeres til **5V**. For å gjøre dette trenger du:

- Et multimeter
- Et lite flatt skrujern
- En 12V strømkilde (Batteriet)

#### Fremgangsmåte:

1. Koble batteriet inn på regulatorens inngang (IN+ og IN-).
2. Koble multimeterets måleprober til regulatorens utgang (OUT+ og OUT-).
3. Les av spenningen på multimeteret.
4. Bruk det flate skrujernet til å justere potensiometeret (den lille skruen på toppen av regulatoren). Drei forsiktig noen runder til du leser av nøyaktig 5V.



5. Når spenningen er korrekt innstilt, er regulatoren klar til å loddas fast på kretskortet.

### 4.4 Påkobling og oppsett av sensorer

Det er to sensorer som skal kobles på kretskortet. Temperaturføleren (Ds18b20) og de røde ledningene fra vektcellene.

#### 4.4.1 Temperaturføler

Temperaturføleren har tre ledninger: rød, svart og gul. Disse skal kobles til kretskortet på følgende måte:

Rød ledning kobles til inngangen nærmest pluss (+) tegnet på kretskortet.

Gul ledning kobles til midterste av de tre hullene.

Svart ledning kobles til inngangen nærmest minus (-) tegnet.



#### 4.4.2 Vektcellene

Dette steget innebærer å sette sammen det fysiske oppsettet av vektsystemet.

##### Verktøyliste:

- Loddebolt

##### Komponentliste:

- 4 vektceller
- En plate til å montere cellene på
- 4 vektcelle-bein (se beskrivelse under)
- 8 skruer som passer til den valgte platen

##### Vektcelle-bein:

Vektcelle-beina er spesialdesignet for å holde cellene stabile og sikre nøyaktig måling.

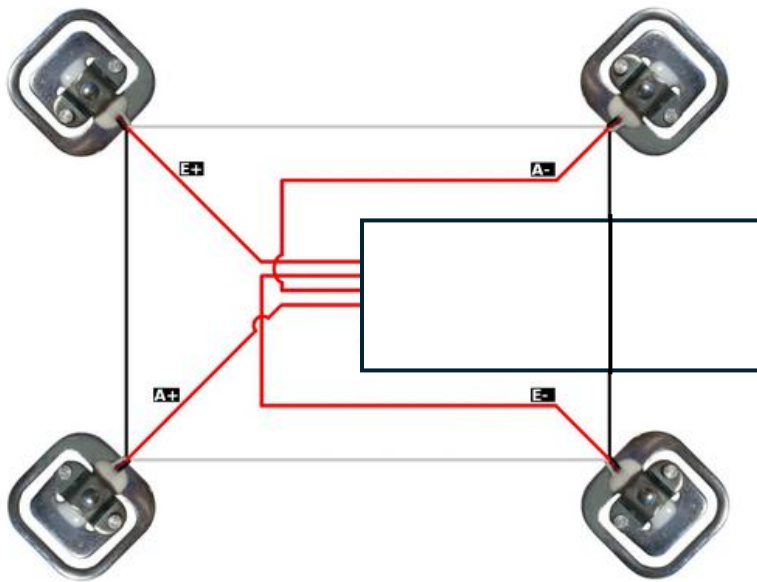
Disse må 3D-printes.

1. Finn en 3D-printer (på skolen, et bibliotek eller en “makerspace”).
2. Vis denne nettsiden til personen som hjelper deg med 3D-printingen:  
<https://www.thingiverse.com/thing:2624188>
3. Be dem importere og tilpasse filen til deres printer.
4. Når beina er ferdig printet, er du klar for neste steg.

## Oppsett og tilkobling

Når alle komponentene er klare, kan du begynne monteringen:

1. Fest vektcellene til platen med skruer.
2. Koble ledningene fra hver celle sammen i korrekt konfigurasjon (se fargekode på bildet under).



(Merk deg navnene på de fire røde ledningene(E+,E-,A-,A+) ettersom disse er viktig i ett senere steg.

3. Klips vektcelle-beina fast på hver celle.

Sluttresultatet skal ligne på bildet nedenfor:



## Tilkobling til kretskort

De røde ledningene fra vektcellene skal loddet fast på kretskortet.



Bruk kun de **fire øverste** inngangene, der rekkefølgen av hullene er:

- **E+**
- **E-**
- **A-**
- **A+**

Tilkoblingen skal være slik (fra topp til bunn):

1. Øverst: **E+**
2. Nest øverst: **E-**
3. Deretter: **A-**
4. Nederst (av de fire): **A+**

Lodd disse fire ledningene godt fast. Kontroller at alt sitter riktig før videre bruk.

## 5. Opplasting av kode til ESP (Wemos D1 Mini)

Nå gjenstår det bare å lase opp et spesifikt program til mikrokontrolleren.

Får å få fullført dette steget trenger du å laste ned Arduino IDE, ett program som lastes ned på pc-en din. Lenken til nedlastingsnettsiden : <https://www.arduino.cc/en/software/>

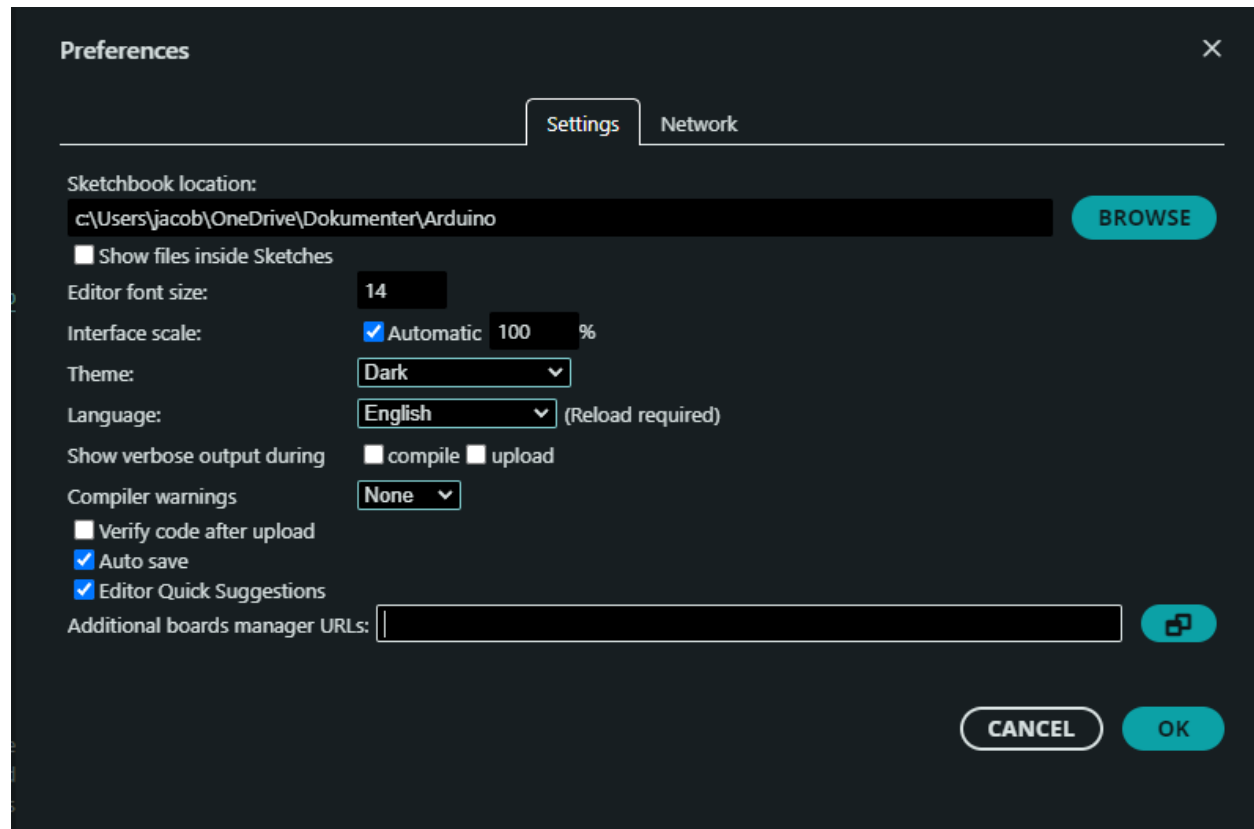
Videre må du laste ned programmet vi har laget spesielt for bruk av det helhetlige oppsettet. Filen kan du laste ned i GitHub lenken i introduksjon, men du kan også finne koden i tekst format lengre nede(anbefaler at du klipper og limer inn koden):

For at du skal kunne ta i bruk mikrokontrolleren må du laste den ned i Arduino IDE.

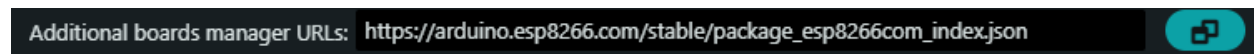
Stegene for dette er.

1. Start Arduino IDE på PC-en din.
2. Legg til ESP8266 URL

- a. Gå til **Fil > Innstillinger** (eller **Arduino > Preferences** på Mac). Du skal da få dette opp:



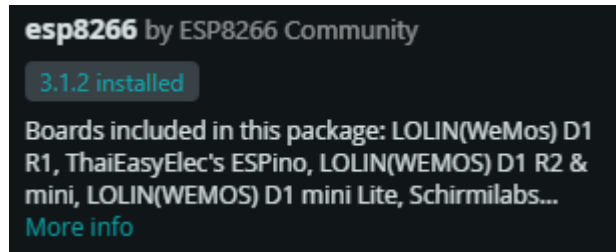
- b. I feltet **"Additional boards manager URLs:"**, lim inn følgende URL:  
[https://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](https://arduino.esp8266.com/stable/package_esp8266com_index.json). Det skal da se slik ut:



### 3. Installer ESP8266 Board Support

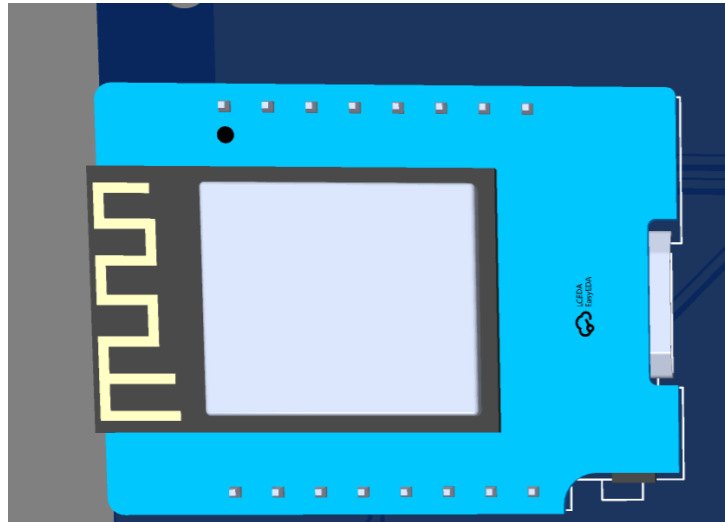
- Finn Verktøy > Kort > Kortstyrer...
- I søkefeltet skriver du: esp8266
- Finn **"esp8266 by ESP8266 Community"** og klikk **Installer**. Den skal se slik ut:





4. Koble til Wemos D1 mini

- a. Koble Wemos D1 mini til PC med en micro USB-kabel.

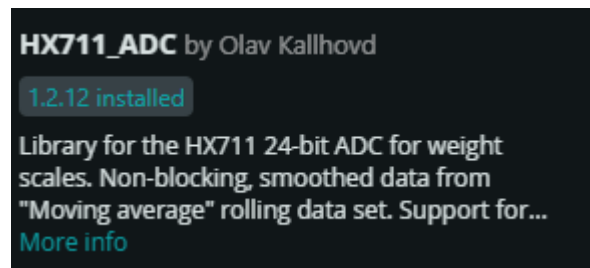


- b. Gå til **Verktøy > Port** og velg riktig COM-port (den som vises når du kobler til).

5. Nedlasting av nødvendige bibliotek.

- a. På tastaturet ditt trykk Ctrl + Shift + I.
- b. I søkefeltet skriv **HX711\_ADC** og finn den laget av Olav Kallhovd og klikk **Installer**.

Skal se slik ut:



- c. Gå tilbake til søkefeltet og skriv **OneWire** og finn den laget av Jim Studt og klikk **Installer**.

Skal se slik ut:

**OneWire** by Jim Studt, Tom Pollard, Robin James, Glenn ...  
Trewitt, Jason Dangel, Guillermo Lovato, Paul Stoffregen,...  
2.3.8 installed  
Access 1-wire temperature sensors, memory and other chips.  
[More info](#)

6. Kopier koden under og lim den inn i Arduino IDE. Klikk på Opplasting (en pil oppe i høyre gjørne). Nå skal koden være opplastet til mikrokontrolleren.

Kildekode:

```
////////// Start på kode
```

```
#include <ESP8266HTTPClient.h>
```

```
#include <ESP8266WiFi.h>
```

```
#include <HX711_ADC.h>
```

```
#if defined(ESP8266) || defined(ESP32) || defined(AVR)
```

```
#include <EEPROM.h>
```

```
#include <OneWire.h>
```

```
#endif
```

```
//////////
```

```
// WiFi Credentials
```

```
//////////
```

```

#define WIFI_SSID "BiKubeWifi" //

#define WIFI_PASSWORD "B1kubeM0nitorering"

////////////////////////////////////

// ThingSpeak Credentials

////////////////////////////////////

const char* THINGSPEAK_API_KEY = "REHEU1FGRAGQ9FHU";

const char* THINGSPEAK_URL = "http://api.thingspeak.com/update";

////////////////////////////////////

// OneWire Setup for DS18B20 Temperature Sensor

////////////////////////////////////

#define ONE_WIRE_BUS 14

OneWire ds(ONE_WIRE_BUS);

////////////////////////////////////

```

```

// HX711 Load Cell Setup

////////////////////////////////////

const int HX711_dout = 4;

const int HX711_sck = 5;

HX711_ADC LoadCell(HX711_dout, HX711_sck);

const int addr_magic = 0; // EEPROM address for magic number

const int addr_calibration = 4; // EEPROM address for calibration factor

const int addr_offset = 8; // EEPROM address for tare offset

const int addr_wake_counter = 12; // EEPROM address for wake counter

const int max_wake_count = 12; // Number of wakeups before sending

////////////////////////////////////

// Relay Pin

////////////////////////////////////

const int relayPin = D6; // GPIO12

```

```
////////////////////////////////////
```

```
// Function Prototypes
```

```
////////////////////////////////////
```

```
void connectToWiFi();
```

```
float readWeight();
```

```
float readTemperature();
```

```
void sendToThingSpeak(float weight);
```

```
void EEPROMWriteFloat(int address, float value);
```

```
float EEPROMReadFloat(int address);
```

```
////////////////////////////////////
```

```
// Setup Function
```

```
////////////////////////////////////
```

```
void setup() {
```

```
Serial.begin(115200);
```

```
pinMode(relayPin, OUTPUT);
```

```
digitalWrite(relayPin, HIGH); // Start with relay on
```

```
connectToWiFi();
```

```
LoadCell.begin();
```

```
unsigned long stabilizingtime = 2000;
```

```
LoadCell.start(stabilizingtime, false);
```

```
if (LoadCell.getTareTimeoutFlag() || LoadCell.getSignalTimeoutFlag()) {
```

```
    Serial.println("HX711 Timeout! Check wiring.");
```

```
    while (1)
```

```
    ;
```

```
}
```

```
EEPROM.begin(512);
```

```

const byte MAGIC_NUMBER = 0x42;

byte storedFlag = EEPROM.read(addr_magic);

if (storedFlag != MAGIC_NUMBER) {

    Serial.println("First-time startup detected. Performing tare...");

    while (!LoadCell.update())
        ;

    LoadCell.setCalFactor(1.0); // Temporary, real calibration later

    LoadCell.tare(); // Tare with current load

    float tareOffset = LoadCell.getTareOffset();

    float calFactor = 20.62; // <-- Change this to your actual calibration factor

    EEPROM.write(addr_magic, MAGIC_NUMBER);

    EEPROMWriteFloat(addr_offset, tareOffset);

    EEPROMWriteFloat(addr_calibration, calFactor);

```

```

EEPROM.commit();

Serial.println("Tare and calibration saved to EEPROM.");

} else {

float tareOffset = EEPROMReadFloat(addr_offset);

float calFactor = EEPROMReadFloat(addr_calibration);

LoadCell.setCalFactor(calFactor);

LoadCell.setTareOffset(tareOffset);

Serial.println("Tare and calibration loaded from EEPROM.");
}

Serial.println("Load cell initialized.");

while (!LoadCell.update())
;
}

////////////////////////////////////

```



```

// Main Loop Function

////////////////////////////////////

void loop() {

    // Les teller fra EEPROM

    byte wakeCounter = EEPROM.read(addr_wake_counter);

    if (wakeCounter >= max_wake_count) {

        digitalWrite(relayPin, HIGH);

        delay(200);

        float temperature = readTemperature();

        float weight = readWeight();

        Serial.println("Weight: " + String(weight) + " g");

        if (temperature != -127.0) {

            Serial.println("Temp: " + String(temperature) + " °C");

```

```

    sendToThingSpeak(temperature, weight);

} else {

    Serial.println("Ugyldig temperatursensor-avlesning.");
}

digitalWrite(relayPin, LOW); // Slå av releet

// Nullstill telleren etter sending

wakeCounter = 0;

EEPROM.write(addr_wake_counter, wakeCounter);

EEPROM.commit();

} else {

    Serial.println("Sender ikke data denne gangen. Teller: " + String(wakeCounter));

    wakeCounter++;

    EEPROM.write(addr_wake_counter, wakeCounter);

    EEPROM.commit();

```

```
}
```

```
WiFi.disconnect();
```

```
delay(1000);
```

```
Serial.println("Sover i 60 minutter...");
```

```
ESP.deepSleep(3600000000); // 60 min i mikrosekunder
```

```
}
```

```
////////////////////////////////////
```

```
// Connect to WiFi
```

```
////////////////////////////////////
```

```
void connectToWiFi() {
```

```
    Serial.print("Connecting to WiFi");
```

```
    WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
```

```
    while (WiFi.status() != WL_CONNECTED) {
```

```

Serial.print(".");

delay(500);
}

Serial.println("\nWiFi connected!");
}

////////////////////////////////////

// Les temperatur fra DS18B20

////////////////////////////////////

float readTemperature() {

byte data[9], addr[8];

float celsius;

if (!ds.search(addr)) {

ds.reset_search();

return -127.0;
}

ds.reset();

```

```

ds.select(addr);

ds.write(0x44, 1); // Start conversion

delay(1000);

ds.reset();

ds.select(addr);

ds.write(0xBE); // Read Scratchpad

for (int i = 0; i < 9; i++) {

    data[i] = ds.read();
}

int16_t raw = (data[1] << 8) | data[0];

celsius = (float)raw / 16.0;

return celsius;
}

```

```

////////////////////////////////////

```

```

// Read weight from HX711

////////////////////////////////////

float readWeight() {
    //Read initial weight some times to stabilice HX711

    int count = 30;
    while (count > 0) // repeat until count is no longer greater than zero
    {

        if (LoadCell.update()) {

            Serial.println("Stabelizing : " + String(LoadCell.getData()) + " g"); // Return weight
data from HX711
            count = count - 1;
        }
    }

    LoadCell.update();

    return LoadCell.getData(); // Return weight data from HX711
}

////////////////////////////////////

// Send data til ThingSpeak

```

```
////////////////////////////////////
```

```
void sendToThingSpeak(float temperature, float weight) {
```

```
    if (WiFi.status() == WL_CONNECTED) {
```

```
        HTTPClient http;
```

```
        WiFiClient client;
```

```
        String url = String(THINGSPEAK_URL) + "?api_key=" + THINGSPEAK_API_KEY +
```

```
            "&field1=" + String(temperature) + "&field2=" + String(weight);
```

```
        http.begin(client, url);
```

```
        int httpCode = http.GET();
```

```
        if (httpCode == 200) {
```

```
            Serial.println("✅ Data sendt til ThingSpeak.");
```

```
        } else {
```

```
            Serial.println("❌ Feil ved sending til ThingSpeak.");
```

```
        }
```

```

    http.end();

} else {

    Serial.println(" ✖ Ikke koblet til WiFi.");
}
}

////////////////////////////////////

// Save float value to EEPROM

////////////////////////////////////

void EEPROMWriteFloat(int address, float value) {

    byte* data = (byte*)(void*)&value;

    for (int i = 0; i < 4; i++) {

        EEPROM.write(address + i, data[i]);
    }
}

////////////////////////////////////

```



```

// Read float value from EEPROM

////////////////////////////////////

float EEPROMReadFloat(int address) {

    byte data[4];

    for (int i = 0; i < 4; i++) {

        data[i] = EEPROM.read(address + i);
    }

    float value;

    memcpy(&value, data, sizeof(value));

    return value;
}

////////// slutt på kode

```

7. Du skal få opp dette om det er blitt gjort riktig.

```
Output  Serial Monitor
Auto-detected Flash size: 4MB
Compressed 293856 bytes to 214871...
Writing at 0x00000000... (7 %)
Writing at 0x00004000... (14 %)
Writing at 0x00008000... (21 %)
Writing at 0x0000c000... (28 %)
Writing at 0x00010000... (35 %)
Writing at 0x00014000... (42 %)
Writing at 0x00018000... (50 %)
Writing at 0x0001c000... (57 %)
Writing at 0x00020000... (64 %)
Writing at 0x00024000... (71 %)
Writing at 0x00028000... (78 %)
Writing at 0x0002c000... (85 %)
Writing at 0x00030000... (92 %)
Writing at 0x00034000... (100 %)
Wrote 293856 bytes (214871 compressed)
Hash of data verified.

Leaving...
Hard resetting via RTS pin...
```

## 6. Koble på batteri og Raspberry Pi

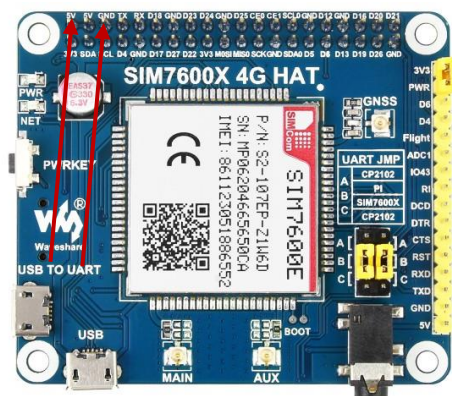
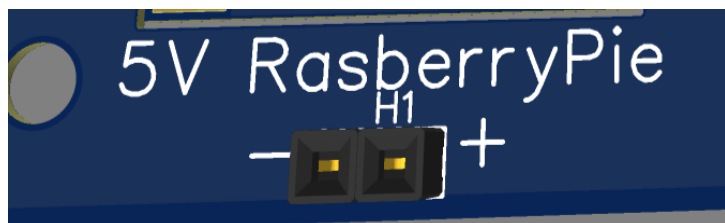
Nå er du klar for det siste steget i oppsettet: å koble kretskortet til Raspberry Pi og deretter koble til batteriet.

### Tilkobling til Raspberry Pi

For kobling av raspberry pi til kretskort bruk «hun til han ledninger».

På kretskortet finner du to utganger merket **(+)** og **(-)**. Disse skal kobles til Raspberry Pi som følger:

1. Koble en ledning fra (+) på kretskortet til 5V på Raspberry Pi.
2. Koble en ledning fra (-) på kretskortet til GND (jord) på Raspberry Pi.



## Tilkobling av batteri

Før du kobler til batteriet, må du dobbeltsjekke at alle tidligere steg er fullført, og at alle ledninger og komponenter er korrekt tilkoblet.

For tilkobling fra batteri til kretskort må du bruke 1.5 kvadrat ledninger.

Når alt annet er klart, kobler du til batteriet slik:

1. Batteriets røde (+) ledning kobles til (+) på kretskortet.
2. Batteriets svarte (-) ledning kobles til (-) på kretskortet.



### Systemstart og overvåking

Nå som oppsettet er ferdig, skal systemet starte automatisk når strømmen kobles til.

#### Viktig ved oppstart:

- Sørg for at det ikke er noe vekt på vektsensorene ved oppstart.  
Dette er nødvendig for at systemet skal kunne kalibrere vekten og sette riktig nullpunkt.
- Vent noen sekunder, deretter kan du plassere bikuben på vekten.

### Overvåking via ThingSpeak

Du kan nå følge med på temperatur og vektmålinger på nettsiden:

<https://thingspeak.mathworks.com/>

Vi har laget en konto for deg. Logg inn med følgende informasjon:

- **Brukernavn / E-post:** bikubemonitorering@gmail.com
- **Passord:** B1kubem0nitorering

### Hvordan se målingene:

1. Etter innlogging, gå til menyen og klikk på **Channels > Watched Channels**.
2. Trykk på "**Målinger**" for å se dataene.

## 7. Systemvedlikehold

### 7.1 Opplading og bytting av batteri

For opplading av batteriet anbefaler vi at du benytter en bilbatterilader. Slike ladere er lett tilgjengelige, for eksempel hos Biltema: [Batterilader](#)

Før opplading må batteriet kobles fra systemet, og lades separat før det tas i bruk igjen. Vi anbefaler at du bruker to batterier og veksler mellom dem, ett i bruk og ett til oppladning. Dette gjør det enklere å opprettholde kontinuerlig drift uten nedetid.

**Viktig:** Før du kobler til batteriet og starter opp systemet igjen, må du sørge for at bikuben ikke står på vekten. Dette er nødvendig for å sikre korrekt nullstilling og nøyaktig måling ved oppstart.

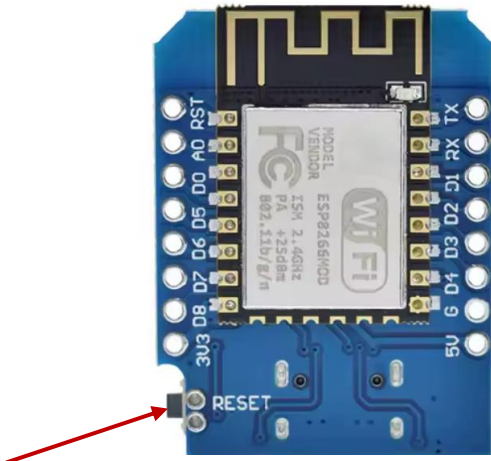
### 7.2 Omstart av system

Av og til kan det være nødvendig å restarte systemet, enten som følge av tekniske problemer eller for å tvinge frem en ny datasending.

Systemet restartes enkelt ved å trykke på reset-knappen som finnes på mikrokontrolleren på kretskortet.

Fremgangsmåte:

1. Fjern bikuben fra vekten.
2. Trykk og hold inne reset-knappen i 5 sekunder.



3. Sett bikuben tilbake på vekten.

Dette kan også benyttes for å trigge umiddelbar datasending. Merk at det i noen tilfeller kan kreve flere forsøk (maks 12) før sendingen skjer, avhengig av hvor lang tid det er siden forrige overføring.

### Trenger du hjelp?

Dersom du trenger hjelp til oppsett eller bruk, kan du kontakte oss.