



# TTT4255 Elektronisk systemdesign, grunnkurs

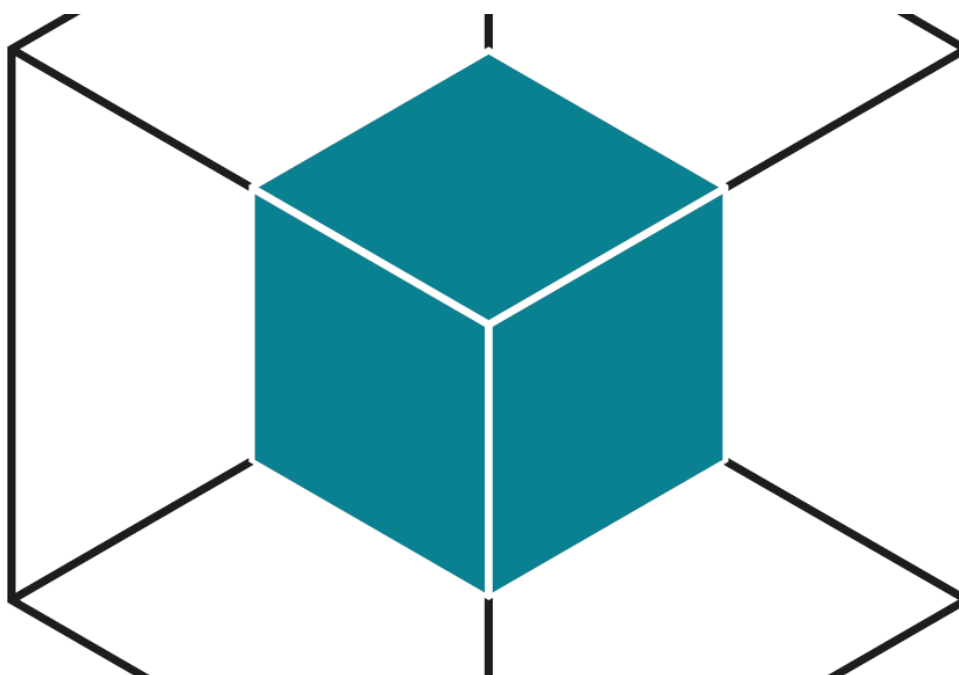
## S6: Tiltensor

Elektronisk systemdesign og innovasjon

---

Ida Bjørnevik, Sven Amberg, Amalie 28.06.2023  
Fridfeldt Hauge og Peter Magerøy

---



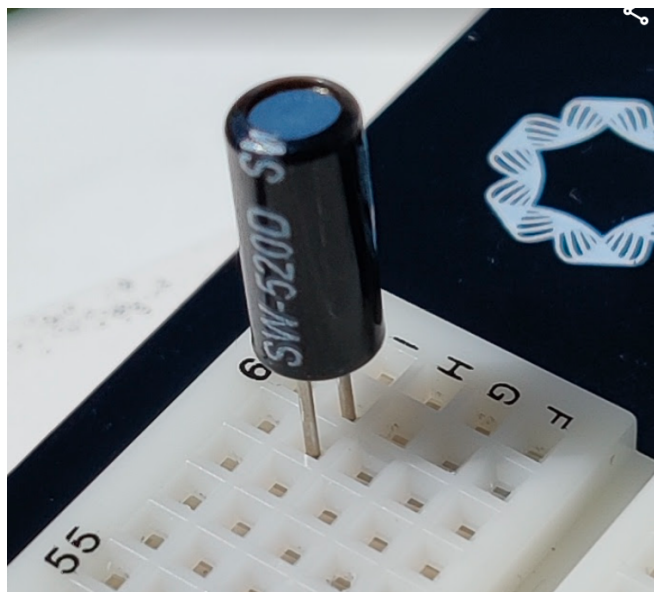
### Innhold

<b>Introduksjon</b>	<b>2</b>
Teori . . . . .	2
<b>Prosjektet</b>	<b>3</b>
Utstyrliste . . . . .	3
Steg 1: Oppkobling av krets. . . . .	3
Steg 2: Programmering . . . . .	4
Steg 3: Ekstra funksjonalitet til systemet. . . . .	5
<b>Konklusjon</b>	<b>5</b>

---

## Introduksjon

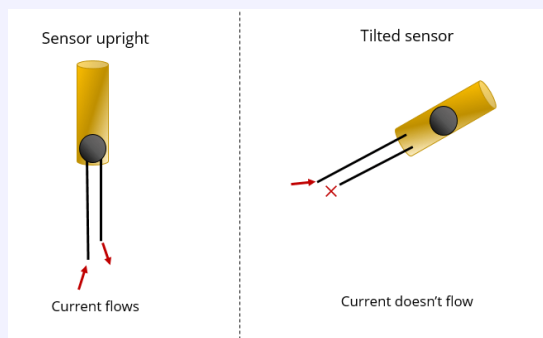
Passar for deg med **ingen** forkunnskapar.



**Figur 1:** Tiltensor.

### Teori for den interesserte

En tiltensor er en bryter som blir skrudd av eller på, avhengig om sensoren står vertikalt eller horisontalt. I denne modulen skal du få en LED til å lyse dersom sensoren ligger horisontalt.



**Figur 2:** Illustrasjon henta frå: <https://randomnerdtutorials.com/guide-for-the-tilt-sensor-inclinometer-with-arduino/>

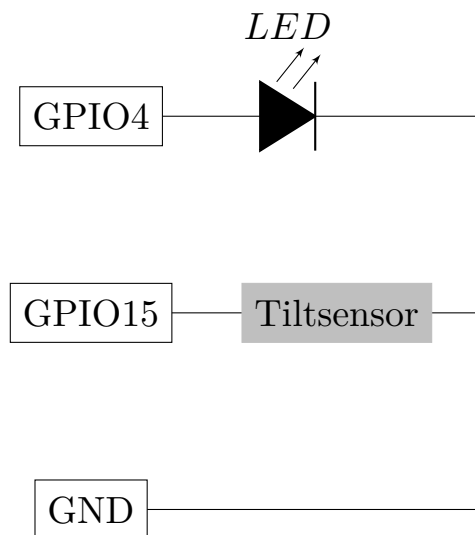
# Prosjektet

## Utstyrliste

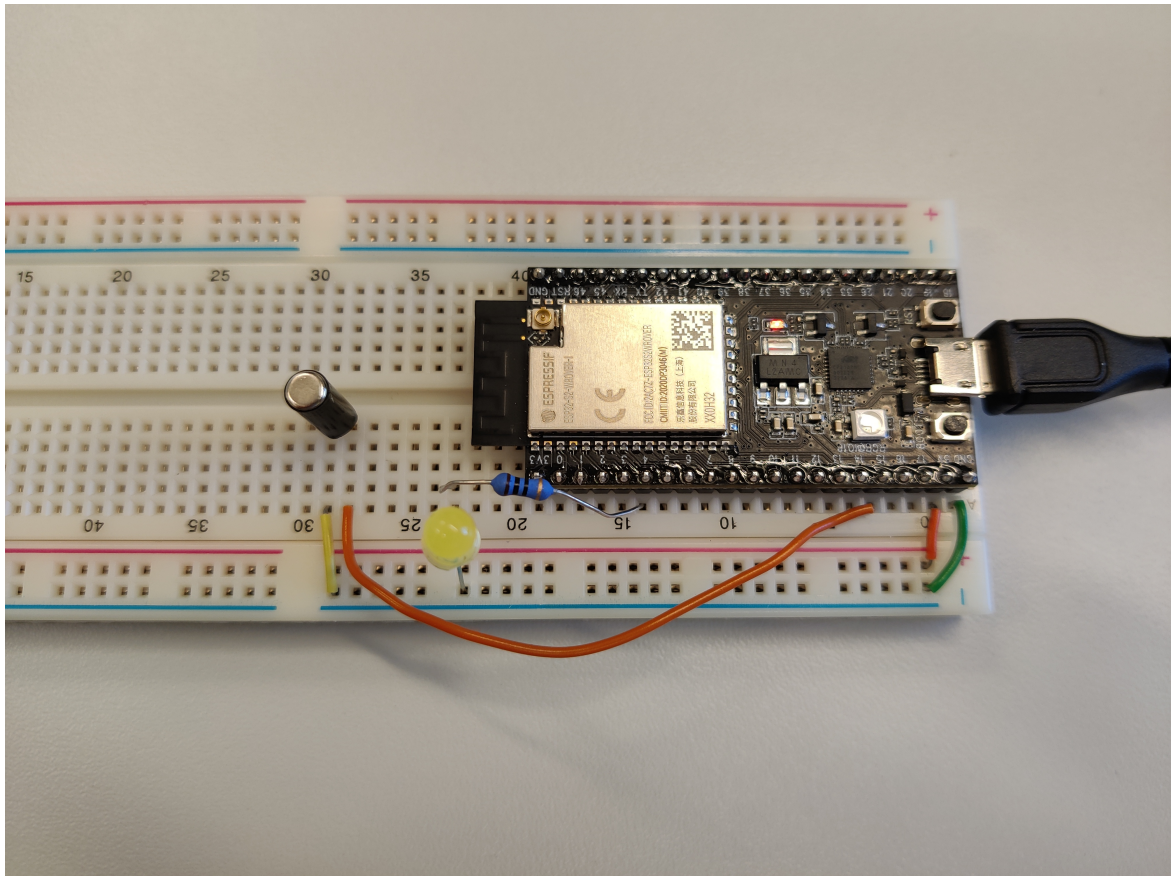
- ESP32-S2
- Tiltssensor
- LED

### Steg 1: Oppkobling av krets.

Koble opp kretsen som vist i figur 3. Fysisk oppkobling er vist i figur 4.



**Figur 3:** Kretsskjema for systemet.



**Figur 4:** System med en ESP32, en tiltsensor og en LED.

## Steg 2: Programmering

Koden for programmet er vist i figur 5, skriv av koden og fyll inn rett verdi for variabelene sensorPin og led. I definisjonsdelen (øverst) av koden definerer vi hvilke pinner vi har valgt å bruke til sensoren og LED-en. I setup-delen setter vi hvilke funksjoner de ulike pinnene har og starter seriell kobling. Seriell overvåker er der vi kan lese verdiene som sensoren registrerer. I loop()-funksjonen sjekker vi hvilken tilstand sensoren er i med jevne mellomrom, denne verdien blir skrevet til seriell overvåker.

```

int sensorPin = 9999; // Eksempel: 4 svarar til GPIO4 (D4 på
ESP32).
int led = 9999;
int sensorValue;      //Lagar variabel utan å gi verdi.

void setup(){
  pinMode(sensorPin, INPUT_PULLUP); //Set pin til INPUT med
  innebygde pullup-motstand aktivert.
  pinMode(led, OUTPUT);
  digitalWrite(sensorPin, HIGH);
  Serial.begin(9600);
}

void loop(){
  sensorValue = digitalRead(sensorPin); //Les verdien på
  sensoren (leiar den strøm eller ikkje?).
  Serial.print("Sensorverdi: "); //Skriv sensorverdi til
  seriellmonitor.
  Serial.println(sensorValue);
  digitalWrite(led, sensorValue);
  delay(1000);
}

```

**Figur 5:** Kode for å få LED til å lyse når sensoren er tiltet.

Koble ESP32-S2 til PC og last opp koden, deretter kan du åpne seriell overvåker og se hva som dukker opp. Prøv å holde sensoren loddrett, hva skjer med LEDen? Prøv så å holde den vannrett, hva skjer her?

### Steg 3: Ekstra funksjonalitet til systemet.

#### Tenke sjølv

Prøv deg på minst ett punkt:

- Legg inn delay slik at LEDen lyser i 10 s etter sensoren blir utløst.
- Få LED-en til å skru seg på når sensoren ligger horisontalt.
- Gjør om koden slik at å vippe sensoren veksler tilstanden til LED-en.

## Konklusjon

I denne modulen lærte du hvordan man kan bruke en tiltsensor for å merke når systemet vippes over.

### Refleksjonsspørsmål

- I hvilket prosjekt er det nyttig med en tiltsensor?
- Hvorfor bør vi ha med **delay()** i **loop()**?
- Når vi snur sensoren tar det litt tid før LED-en slutter å lyse. Hvordan kan vi gjøre slik at den oppdateres raskere?