

Informasjon

Oppgave	Tittel	Maks poeng	Oppgavetype
i	Om eksamen		Informasjon eller ressurser

Elektronikk

Oppgave	Tittel	Maks poeng	Oppgavetype
1	Kirchhoff's spenningslov	3	Flervalg (flere svar)
2	LED	3	Flervalg (flere svar)
3	Motstandsnettverk	3	Fyll inn tall
4	Norton / Thévenin	3	Flervalg (flere svar)
5	Kondensatorer	3	Flervalg (flere svar)
6	Spenningskilde	3	Flervalg (flere svar)
7	Elektrisk ladning	3	Flervalg (flere svar)
8	Naturlig respons	3	Flervalg (flere svar)
9	Opamp	3	Fyll inn tall
10	Kirchhoff's strømlov	3	Flervalg (flere svar)
11	Spenningsdeler	3	Flervalg (flere svar)

Motorer og drivekretser

Oppgave	Tittel	Maks poeng	Oppgavetype
12	H-bru	3	Flervalg (flere svar)
13	H-bru - kode	3	Flervalg (flere svar)
14	Steppermotor	3	Flervalg (flere svar)

15	DC-motor	3	Flervalg
16	Kommutering 1	3	Flervalg (flere svar)
17	BLDC motor	3	Flervalg (flere svar)
18	Kommutering - 2	3	Flervalg (flere svar)

Sensorer og AD-konvertering

Oppgave	Tittel	Maks poeng	Oppgavetype
19	Wheatstone-bro	3	Fyll inn tall
20	Sensor - dimensjonering	3	Flervalg (flere svar)
21	Sensor og støy	3	Flervalg (flere svar)
22	Sensor - 2 stk ADC	3	Fyll inn tall
23	AD konvertering	12	Fyll inn tall

Innebygde systemer

Oppgave	Tittel	Maks poeng	Oppgavetype
24	Frekvens og pulsbredde	9	Fyll inn tall
25	Innebygde systemer	3	Flervalg (flere svar)
26	Busser	3	Flervalg (flere svar)
27	Blokkerende kode	3	Flervalg (flere svar)
28	Regulering	3	Flervalg (flere svar)

Ressurser

Oppgave	Tittel	Maks poeng	Oppgavetype
i	Formelark		Informasjon eller ressurser

i Om eksamen

Skriftlig eksamen i IN1080

2022 Vår

Tid: 10. juni kl. **15:00** (4 timer)

Det er viktig at du leser denne forsiden nøye før du starter.

Hjelpemidler:

Ingen, unntatt kalkulator i Inspira og vedlagt formelsamling (til sist i oppgavesettet).

Om oppgavene:

I dette eksamenssettet er det flervalgsoppgaver og oppgaver der man kan fylle inn ett eller flere tall. Det å svare feil gir ikke negativ uttelling, så det vil alltid lønne seg å velge et alternativ, eller skrive inn et tall, selv om man er usikker. For desimaltall vil både punktum og komma fungere som desimalskilletegn. Det er ikke nødvendig å taste inn nuller som desimal (Inspira ser ikke forskjell på f.eks 9,0 og bare 9).

Tips:

Det kan være lurt å kladde og skissere ved siden av. (Kladd og skisser skal ikke leveres inn).

Lykke til!

1 Kirchhoff's spenningslov

Kirchhoff's spenningslov

Hva er riktig?

- ☐ Alle spenninger i alle løkkene i kretsen må tas med i samme sum for å få 0
- ☐ Løkker som inneholder strømkilder kan ikke tas med i beregningene
- ☐ Summen av spenninger blir 0 når man summerer spenninger i bare en av løkkene i en krets, selv om kretsen har flere andre løkker
- ☐ Man summerer bare spenningskilder og strømkilder, ikke spenningen over passive komponenter slik som motstander
- ☐ Man summerer bare spenningskilder, ikke spenningen over passive komponenter slik som motstander

Maks poeng: 3

2 LED

LED

Hva er riktig for en Arduino mikrokontroller?

Velg ett svar

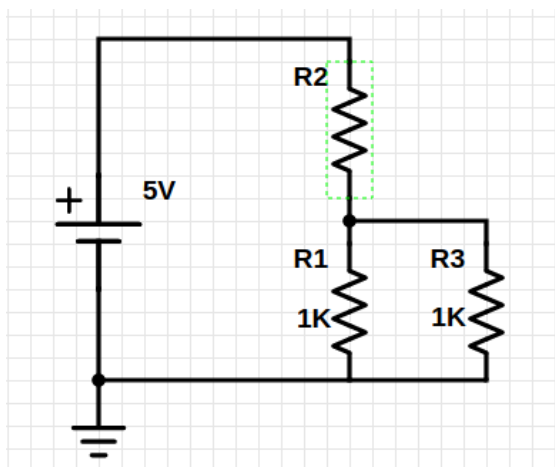
- ☐ En LED vil lyse med omtrent samme lysstyrke uansett retning på strømmen gjennom den
- ☐ LED's bør ikke kobles direkte til vanlige digitale utganger, men man kan koble dem direkte mellom en utgang av type PWM og jord
- ☐ Det er greit å styre lyset i en LED ved å koble den direkte mellom en digital utgang og jord
- ☐ Man kobler ofte motstander i serie med LED's for å kunne drive dem med firkantpuls signaler
- ☐ Det er greit å styre lyset i en LED ved å koble den direkte mellom en analog utgang og jord

Maks poeng: 3

3 Motstandsnettverk

Motstandsnettverk

Hva må verdien på R2 være for at vi skal få 2.5V over R3? (1K betyr 1 kOhm)



Skriv inn et tall: Ohm

Maks poeng: 3

4 Norton / Thévenin

Norton / Thévenin

Hva er riktig?

- ☐ Ikke alle lineære to-terminalers nettverk bestående av strømkilder, spenningskilder og resistorer kan representeres med Thévenin-forenklinger - derfor har vi Norton-forenklinger
- ☐ En Thévenin-forenklet krets er å betrakte som ideell slik at spenningen ut er uavhengig av strømmen ut
- ☐ Enhver Thévenin-forenklet krets kan representeres med en Norton-forenklet krets
- ☐ En Thévenin-type spenningskilde er en spesiell type spenningskilde som kan brukes i enhver praktisk sammenheng, men ulempen er vanligvis høyere pris enn for vanlige spenningskilder
- ☐ Har man en Thévenin-forenklet krets og bytter ut spenningskilden med en strømkilde får man en Norton-forenklet krets

Maks poeng: 3

5 Kondensatorer

Seriekoblede kondensatorer

Hva er riktig for ideelle kondensatorer?

Velg ett svar

- ☐ Dobler man arealet på platene i en gitt kondensator får man samme reaktans som i en seriekobling av to kondensatorer med den opprinnelige kapasitansen
- ☐ Tidskonstanten i en enkel RC-krets med 2 like kondensatorer koblet i serie etter hverandre er lik tidskonstanten for en tilsvarende krets som bare inneholder den ene kondensatoren
- ☐ Total reaktans fra to like seriekoblede kondensatorer er den doble av reaktansen i en av kondensatorene
- ☐ Halverer man arealet på platene i en gitt kondensator får man samme reaktans som i en seriekobling av to kondensatorer med den opprinnelige kapasitansen
- ☐ Man unngår vanligvis seriekobling av kondensatorer på grunn av ikke-ideelle effekter

Maks poeng: 3

6 Spenningskilde

Ideell spenningskilde

Hva er riktig?

Velg ett svar

- ☐ Spenningen ut er alltid 1 joule når strømmen ut er 1 coulomb
- ☐ Strømmen ut er alltid konstant
- ☐ Spenningen ut er alltid konstant selv om man kobler til kondensatorer og spoler
- ☐ Spenningen ut er avhengig av strømmen
- ☐ Spenningen ut er konstant kun når man kobler til et resistivt nettverk

Maks poeng: 3

7 Elektrisk ladning

Elektrisk ladning

Hvor stor elektrisk ladning representerer 3 elektroner til sammen

Velg ett svar

- ☐ -3.28×10^{-19} Volt
- ☐ -3.28 Coloumb
- ☐ -4.87×10^{-19} C
- ☐ 0C
- ☐ -3.28 Ampere

Maks poeng: 3

8 Naturlig respons

Naturlig respons

Hva er naturlig respons for en enkel RC krets?

Velg ett svar

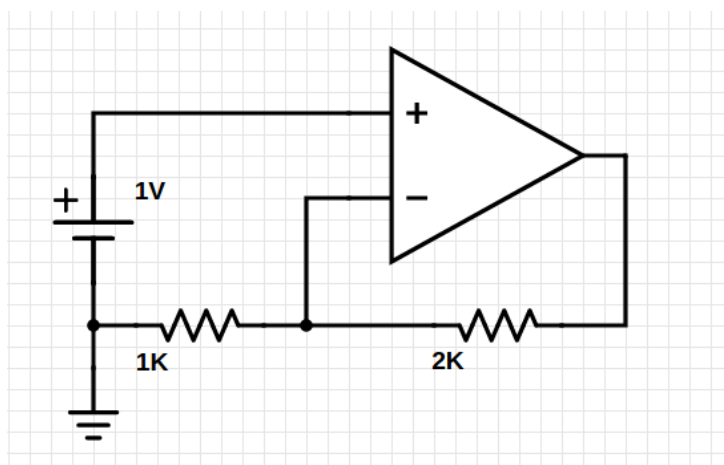
- ☐ Et mål på kun resistansen i kretsen
- ☐ Resistansen + kapasitansen i kretsen
- ☐ Spenningsforløpet over en av komponentene når man momentant fjerner påvirkningen fra alle ytre strøm eller spenningkilder
- ☐ Resistansen multiplisert med kapasitansen i kretsen
- ☐ Et mål på kun reaktansen i kretsen

Maks poeng: 3

9 Opamp

Ideell opamp

Hva blir spenningen ut av opampen? (1K betyr 1 kOhm)



Skriv et tall: volt

Maks poeng: 3

10 Kirchhoff's strømlov

Kirchhoff's strømlov

Hva er riktig?

- ☐ Kirchhoff's strømlov kan også brukes på en node i en løkke der man tidligere har brukt Kirchhoffs spenningslov
- ☐ Kirchhoff's strømlov gjelder ikke for kretser med spoler og kondensatorer
- ☐ Kirchhoff's strømlov gjelder ikke for kretser med åpne stier
- ☐ Bytter man strøm mot spenning i Kirchhoff's strømlov får man Kirchhoff's spenningslov
- ☐ Kirchhoff's strømlov gjelder ikke for kretser med spenningkilder

Maks poeng: 3

11 Spenningsdeler

Spenningsdeler

Hva er riktig?

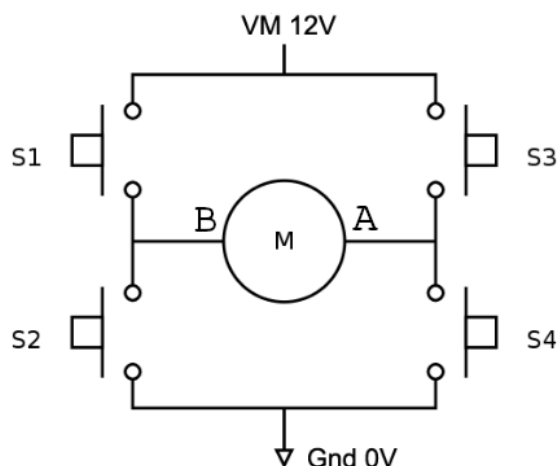
- ☐ En spenningsdeler vil ikke trekke noe strøm ut fra den kilden som leverer inngangsspenningen
- ☐ En spenningsdeler kan kobles til utgangen av en ideell strømkilde og vil da fungere som en strømdeler
- ☐ Spenningsdelere kan også brukes til å øke spenning
- ☐ Spenningen ut av en spenningdeler som består av to like motstander med verdi R blir inngangsspenningen multiplisert med 0.5 hvis man ikke har koblet til noen lastmotstand
- ☐ Spenningen ut av en spenningsdeler med lastmotstand øker med minkende verdi på lastmotstanden

Maks poeng: 3

12 H-bru

H-bru

I figuren under er det vist en H-bru.



Anta at transistor S1 og S4 er lukket (leder strøm) og at S3 og S2 er åpne (leder ikke strøm). M representerer en vanlig DC motor som kan rotere begge veier. Motoren roterer med klokka (CW) når den får høyere spenning på terminal A enn på terminal B.

Hva er riktig for tilstanden beskrevet over?

Velg ett alternativ

- ☐ Motoren er koblet i fra (free running state)
- ☐ Motoren roterer med klokka
- ☐ Dette er en ulovlig tilstand
- ☐ Motoren er i en bremsetilstand uansett tidligere tilstand
- ☐ Motoren roterer mot klokka

Maks poeng: 3

13 H-bru - kode

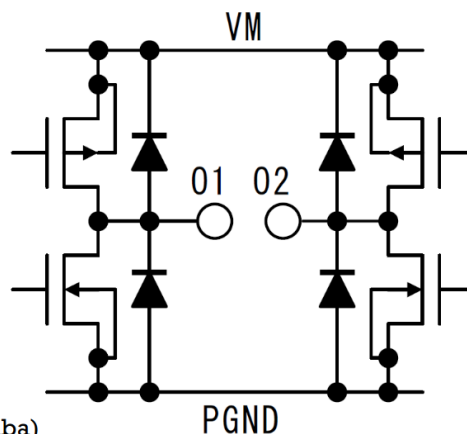
H-bru - kode

I figuren under er det vist et datablad for en H-bru.

H BRU : TB6612FNG

Input				Output		
IN1	IN2	PWM	STBY	OUT1	OUT2	Mode
H	H	H/L	H	L	L	Short brake
L	H	H	H	L	H	CCW
		L	H	L	L	Short brake
H	L	H	H	H	L	CW
		L	H	L	L	Short brake
L	L	H	H	OFF (High impedance)		Stop
H/L	H/L	H/L	L	OFF (High impedance)		Standby

Kilde: Datablad TB6612FNG (Toshiba)



I tabellen er det vist hvordan en DC motor oppfører seg når den kobles til utgangene OUT1 og OUT2 på driveren, i kretsen vist ved terminalene 01 og 02. Dette gjelder for en gitt motoroppkobling, snur man motoren får man en annen oppførsel.

Vi ønsker å kjøre motoren i en repeterende sekvens der vi kjører med klokka (CW) i 1 sekund, så stoppe motoren i 1 sekund, så med klokka (CW) i 1 sekund osv. Hvilken kode kan brukes til dette?

Velg ett alternativ

```
// -----

const int in1Pin = 3; // kobles til IN1
const int in2Pin = 4; // kobles til IN2
const int stbyPin = 5; // kobles til STBY
const int pwmPin = 6; // kobles til PWM

void setup() {
  pinMode(in1Pin, OUTPUT);
  pinMode(in2Pin, OUTPUT);
  pinMode(stbyPin, OUTPUT);
  pinMode(pwmPin, OUTPUT);
☐  digitalWrite(stbyPin, HIGH);
}

void loop() {
  digitalWrite(pwmPin, HIGH);
  digitalWrite(in1Pin, HIGH);
  digitalWrite(in2Pin, LOW);
  delay(1000);
  digitalWrite(pwmPin, LOW);
  digitalWrite(in1Pin, LOW);
  digitalWrite(in2Pin, HIGH);
}

// -----

const int in1Pin = 3; // kobles til IN1
const int in2Pin = 4; // kobles til IN2
const int stbyPin = 5; // kobles til STBY
const int pwmPin = 6; // kobles til PWM

void setup() {
  pinMode(in1Pin, OUTPUT);
  pinMode(in2Pin, OUTPUT);
  pinMode(stbyPin, OUTPUT);
☐  pinMode(pwmPin, OUTPUT);

  digitalWrite(in1Pin, LOW);
  digitalWrite(in2Pin, HIGH);
  digitalWrite(pwmPin, HIGH);
}

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

```
// -----
const int in1Pin = 3; // kobles til IN1
const int in2Pin = 4; // kobles til IN2
const int stbyPin = 5; // kobles til STBY
const int pwmPin = 6; // kobles til PWM
```

```
void setup() {
  pinMode(in1Pin, OUTPUT);
  pinMode(in2Pin, OUTPUT);
  pinMode(stbyPin, OUTPUT);
  pinMode(pwmPin, OUTPUT);
```

```

☐   digitalWrite(pwmPin, HIGH);
                digitalWrite(stbyPin, HIGH);
}
```

```
void loop() {
  delay(1000);
  digitalWrite(in1Pin, HIGH);
  digitalWrite(in2Pin, LOW);
  delay(1000);
  digitalWrite(in1Pin, HIGH);
  digitalWrite(in2Pin, HIGH);
}
```

```
// -----
const int in1Pin = 3; // kobles til IN1
const int in2Pin = 4; // kobles til IN2
const int stbyPin = 5; // kobles til STBY
const int pwmPin = 6; // kobles til PWM
```

```
void setup() {
  pinMode(in1Pin, OUTPUT);
  pinMode(in2Pin, OUTPUT);
  pinMode(stbyPin, OUTPUT);
  pinMode(pwmPin, OUTPUT);
```

```

☐   digitalWrite(stbyPin, HIGH);
                digitalWrite(in1Pin, HIGH);
                digitalWrite(in2Pin, LOW);
}
```

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

```
// -----

const int in1Pin = 3; // kobles til IN1
const int in2Pin = 4; // kobles til IN2
const int stbyPin = 5; // kobles til STBY
const int pwmPin = 6; // kobles til PWM

void setup() {
  pinMode(in1Pin, OUTPUT);
  pinMode(in2Pin, OUTPUT);
  pinMode(stbyPin, OUTPUT);
  pinMode(pwmPin, OUTPUT);

☐   digitalWrite(stbyPin, HIGH);
  digitalWrite(pwmPin, LOW);
}

void loop() {
  delay(1000);
  digitalWrite(in1Pin, HIGH);
  digitalWrite(in2Pin, LOW);
  delay(1000);
  digitalWrite(in1Pin, LOW);
  digitalWrite(in2Pin, HIGH);
}
```

Maks poeng: 3

14 Steppermotor

Steppermotor

Hva er riktig?

Velg ett svar

- ☐ Hvordan man kommuterer påvirker ikke antall mulige vinkler som rotoren kan settes i
- ☐ Wave-kommutering gir færre mulige vinkler som rotoren kan settes i enn ved fullstepp kommutering
- ☐ Halvstepp-kommutering gir færre mulige vinkler som rotoren kan settes i enn ved fullstepp kommutering
- ☐ Wave-kommutering gir mange flere mulige vinkler som rotoren kan settes i enn ved microstepp kommutering
- ☐ Wave-kommutering gir like mange mulige vinkler som rotoren kan settes i som ved fullstepp kommutering

Maks poeng: 3

15 DC-motor

Ideell DC motor

For en DC motor, hvilket utsagn er riktig for en steady state situasjon? Med belastning mener vi dreiemomentet lasten yter på motoren, og med hastighet mener vi vinkelhastighet. Effekt inn er gitt av spenning ganger strøm. Effekt ut (uten tap) er gitt av dreiemoment*vinkelhastighet.

Velg ett svar

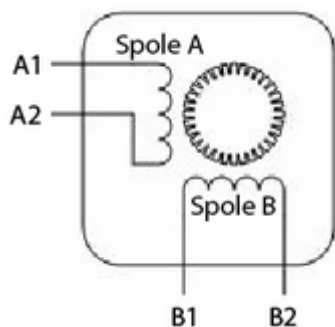
- ☐ Belastningen er omtrent proporsjonal med strømmen. Effekt inn og ut er størst ved minimal hastighet, derfor må aldri en DC motor belastes for fullt før den kommer opp i en viss hastighet
- ☐ Når belastningen er konstant, er hastigheten omtrent lineært avhengig av påført spenning. Når påført spenning er konstant er hastigheten omtrent lineært avhengig av belastningen
- ☐ Belastningen er omtrent proporsjonal med strømmen. Effekt inn er størst ved maksimal hastighet
- ☐ Så lenge motoren ikke overbelastes er hastighet for en gitt spenning uavhengig av belastningen. Hastigheten er gitt av motorkonstantene
- ☐ Den induserte spenningen øker med belastningen. Maksimal hastighet er omtrent gitt av spenning delt på motorkonstant

Maks poeng: 3

16 Kommutering 1

Kommutering av steppermotor

Vi har en bipolar steppermotor med 2 spoler A og B. Vi tenker oss et arrangement der vi styrer spenningene over spolene fra vår Arduino mikrokontroller. Vi tenker oss at "A1pin" og "A2pin" direkte setter spenningene over spole A (se figuren under) via en motordriver. En HØY verdi på pinne "A1pin" setter 5V på ledning A1, en LAV verdi på pinne "A1pin" setter 0V på ledning A1 osv. Tilsvarende for B1pin og B2pin. Vi trenger ikke tenke på hvordan motordriveren virker. Motoren skal rotere med en jevn stepphastighet i en retning.



Under er det vist en liten sekvens av koden som styrer motoren.

```
const int A1pin = 3; // styrer A1 direkte via driver
const int A2pin = 4; // styrer A2 direkte via driver
const int B1pin = 5; // styrer B1 direkte via driver
const int B2pin = 6; // styrer B2 direkte via driver

int ms; // forsinkelse

void setup() {
  pinMode(A1pin, OUTPUT);
  pinMode(A2pin, OUTPUT);
  pinMode(B1pin, OUTPUT);
  pinMode(B2pin, OUTPUT);
  // ms settes;
}

void loop() {
  digitalWrite(B1pin, HIGH);
  digitalWrite(B2pin, LOW);
  delay(ms);
  digitalWrite(A1pin, LOW);
  digitalWrite(A2pin, LOW);
  delay(ms);
  digitalWrite(A1pin, HIGH);
  digitalWrite(A2pin, LOW);
  delay(ms);
  digitalWrite(B1pin, LOW);
  digitalWrite(B2pin, LOW);
  delay(ms);
  digitalWrite(B1pin, LOW);
  digitalWrite(B2pin, HIGH);
  delay(ms);
  digitalWrite(A1pin, LOW);
  digitalWrite(A2pin, LOW);
  delay(ms);
}
```

Hvilken kommuteringstype produserer Arduinokoden?

Velg ett alternativ

- ☐ Wave commutation
- ☐ Mikrostepp
- ☐ Fullstepp
- ☐ Ingen av delene
- ☐ Halvstepp

Maks poeng: 3

17 BLDC motor

BLDC motor

Hva er riktig?

Velg ett svar

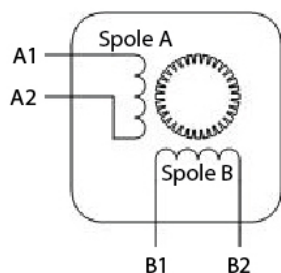
- ☐ Man vil i prinsippet kunne bruke H-bruer til å generere strømmen inn til spolene på en BLDC motor
- ☐ BLDC motor kommuteres på samme måte som en steppermotor under halvstepp kommutering
- ☐ En BLDC motor får strømmen til spolene direkte fra utgangene til hall sensorene
- ☐ Man trenger ikke å kommutere en BLDC motor
- ☐ BLDC motor kommuteres på samme måte som en steppermotor under fullstepp kommutering

Maks poeng: 3

18 Kommutering - 2

Kommutering av steppermotor - 2

Vi har en bipolar steppermotor med 2 spoler A og B. Vi tenker oss et arrangement der vi styrer spenningene over spolene fra vår Arduino mikrokontroller. Vi tenker oss at "A1pin" og "A2pin" direkte setter spenningene over spole A (se figuren under) via en motordriver. En HØY verdi på pinne "A1pin" setter 5V på ledning A1, en LAV verdi på pinne "A1pin" setter 0V på ledning A1 osv. Tilsvarende for B1pin og B2pin. Vi trenger ikke tenke på hvordan motordriveren virker. Motoren skal rotere med en jevn stepphastighet i en retning i flere runder.



Under er det vist **kun en del** av en sekvens av koden som styrer motoren i en vanlig kommuteringsform. Motoren bytter ikke kommuteringsform under bruk.

```
const int A1pin = 3; // styrer A1 direkte via driver
const int A2pin = 4; // styrer A2 direkte via driver
const int B1pin = 5; // styrer B1 direkte via driver
const int B2pin = 6; // styrer B2 direkte via driver

int ms; // forsinkelse

void setup() {
  pinMode(A1pin, OUTPUT);
  pinMode(A2pin, OUTPUT);
  pinMode(B1pin, OUTPUT);
  pinMode(B2pin, OUTPUT);
  // ms settes;
}

void loop() {
  digitalWrite(A1pin, LOW);
  digitalWrite(A2pin, LOW);
  digitalWrite(B1pin, LOW);
  digitalWrite(B2pin, HIGH);
  delay(ms);
  digitalWrite(A1pin, HIGH);
  digitalWrite(A2pin, LOW);
  digitalWrite(B1pin, LOW);
  digitalWrite(B2pin, LOW);
  delay(ms);
  digitalWrite(A1pin, LOW);
  digitalWrite(A2pin, LOW);
  digitalWrite(B1pin, HIGH);
  digitalWrite(B2pin, LOW);
  delay(ms);

  // ...
}
```

Vi vil at motoren skal rotere med 1 omdreining/sekund, og antar at den har 200 fullsteppposisjoner/omdreining, hva må verdien på ms være i millisekund?

Velg ett svar

- ☐ 1s
- ☐ 2.5ms
- ☐ 15ms
- ☐ 5ms
- ☐ Hvis man skal ha 200step/omdreining må man ha en type steppermotor med flere ledninger

Maks poeng: 3

19 Wheatstone-bro

Wheatstone-bro

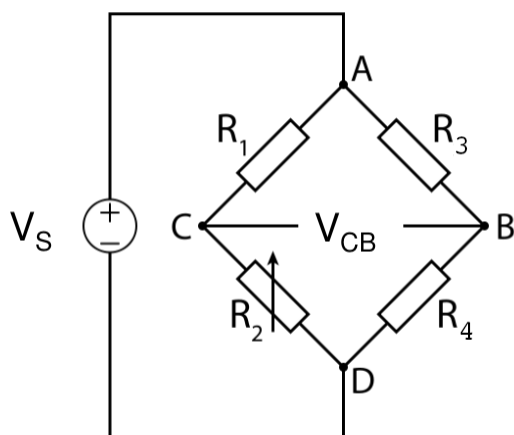
I kretsen under har vi følgende info:

$$V_S = 12V$$

$$R_1 = 1k\Omega$$

$$R_3 = 1k\Omega$$

$$R_4 = 1k\Omega$$



Anta at R_2 representerer en sensor som måler temperatur, og at verdien for øyeblikket er 990Ω . Hva blir spenningen V_{CB} ut?

Skriv inn en verdi. $V_{cb} =$ V

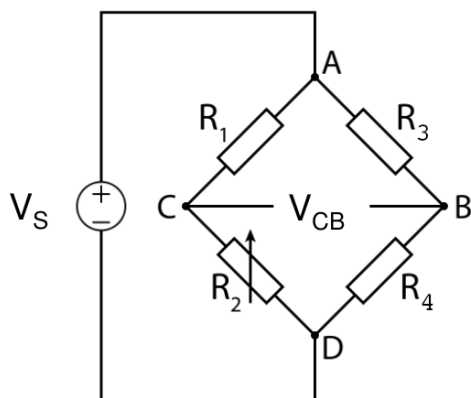
Maks poeng: 3

20 Sensor - dimensjonering

Temperatursensor - dimensjonering av motstander

Vi har en temperatursensor som forandrer resistans med temperatur. Denne har vi koblet opp i en Wheatstone-bro og representert med R_2 slik som vist i figuren under. De 3 andre motstandene er like og har en verdi på $1\text{k}\Omega$. Spenningen V_{CB} ut av broen ønsker vi å lese av med en ADC innebygd i en Arduino. Arduinoens ADC tar inn analoge spenninger som ligger i intervallet $0\text{V} - 5\text{V}$.

Vi kobler noden C til Arduinoens jord/GND og kobler en ledning fra node B til Arduinoens analoge inngang A0.



Vi vil kreve at den analoge spenningen inn på Arduinoens analoge inngang alltid er positiv i forhold til jord/GND. Hva må til for å garantere dette?

Velg ett eller flere alternativer

- ☐ Uansett verdier på alle motstandene oppfylles kravet
- ☐ Uansett verdi på R_2 og kan vi ikke garantere dette
- ☐ R_2 må alltid være mindre enn $1\text{k}\Omega$
- ☐ R_2 må alltid være større enn $1\text{k}\Omega$
- ☐ Spenningen V_S må være større enn 5V

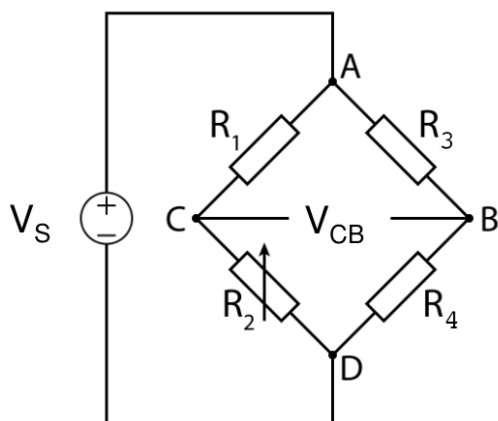
Maks poeng: 3

21 Sensor og støy

Temperatursensor med støy

Vi har en temperatursensor som forandrer resistans med temperatur. Denne har vi koblet opp i en Wheatstone-bro og representert med R_2 slik som vist i figuren under. De andre motstandene er konstante og har lik verdi. Spenningen V_{CB} ut av broen ønsker vi å lese av med en ADC innebygd i en Arduino. Arduinoens ADC tar inn analoge spenninger som ligger i intervallet 0V - 5V.

Vi kobler noden C til Arduinoens jord/GND og kobler en ledning fra node B til Arduinoens analoge inngang A0. Vi antar at motstandsverdiene for øyeblikket er slik at den analoge spenningen inn til Arduinoens ADC er litt over 0V i forhold til jord/GND.



Hvis vi får spenningsstøy på V_S , slik spenningen V_S blir litt forandret, hva skjer med den analoge spenningen inn til Arduinoens ADC?

Velg ett eller flere alternativer

- ☐ Arduinoens ADC vil registrere en støyspenning som er like stor som støyspenningen på V_S
- ☐ Arduinoens ADC vil registrere en støyspenning som er like stor som minus støyspenningen på V_S
- ☐ Arduinoens ADC vil nesten ikke registrere noen støyspenning på V_S
- ☐ Arduinoens ADC vil registrere en støyspenning som er minus halvparten av støyspenningen på V_S
- ☐ Arduinoens ADC vil registrere en støyspenning som er halvparten av støyspenningen på V_S

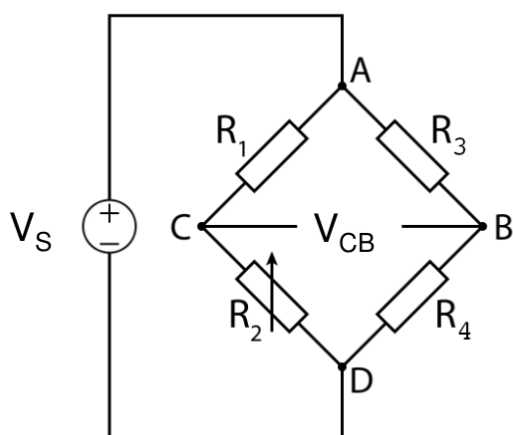
Maks poeng: 3

22 Sensor - 2 stk ADC

Temperatursensor - bruk av 2 stk ADC

Vi har en temperatursensor som forandrer resistans med temperatur. Denne har vi koblet opp i en Wheatstone-bro og representert med R_2 slik som vist i figuren under. De 3 andre motstandene er like og har en verdi på 1kOhm. Vi vil bruke en Arduino med 2 stk ADC til å konvertere spenningene på node C og node B til digital. Vi kobler derfor node D til Arduinos jord/GND. Vi kobler node B til Arduinoens A0 inngang og node C til Arduinoens A1 inngang. Arduinoens ADC tar inn analoge verdier i intervallet 0V-5V.

Vi ønsker å beregne spenningen V_{CB} digitalt i vår egen programkode. Det er greit å merke seg at kommandoen `analogRead()` tar en viss tid å utføre, slik at vi får ikke samplet spenningene på node C og B samtidig. Dette medfører høyfrekvent spenningsstøy på V_S får negative konsekvenser.



Anta at vi sampler 3V på node C, og sampler 2.5V på node B. Arduinoen har en ADC på 10bit. Vi bruker følgende kode for å finne V_{CB} .

```
int vCB;
int sensB;
int sensC;

void loop() {
    sensB = analogRead(A0);
    sensC = analogRead(A1);

    vCB = sensC - sensB;
}
```

Hva blir verdien på vCB?

Svar:

Maks poeng: 3

23 AD konvertering

AD-konvertering

Vi har en 3,3V AD-konverter med 10 bits presisjon. Det dynamiske området er "Rail-to-rail", det vil si fra 0 til 3,3V. Samplingshastigheten til ADkonverteren er 1MHz.

a) Hvor mange nivåer kan AD-konverteren skille?

Skriv inn et tall: nivåer.

b) Hvor stor er minste spenningsforskjell er minste forskjell mellom to nivå?
(Vi forutsetter at AD-konverteren virker linjært i hele området)

Skriv inn et tall: mV

AD konverteren sitter inne i en mikrokontroller, og den kan brukes på to måter. Enten så sampler den en inngang på raskeste frekvens, ellers så sampler den alle de åtte inngangene A0 til A7 etter tur i fast rekkefølge. Selve samplingen foregår alltid med samme frekvens.

Vi har fem sensorer vi ønsker å koble til AD-konverteren.

c) Hva er den høyeste frekvensen vi kan ha på inngangssignalene til de fem sensorene, uten at vi får aliasing?

Skriv inn et tall: kHz

Et Inngangssignal har -40dB støy. Vi antar signalet bruker hele det dynamiske området.

d) Hvor mange nivåer for AD-konverteren tilsvarer støyen (maksimalt)?

Skriv inn et tall: nivåer

Maks poeng: 12

24 Frekvens og pulsbredde

Frekvens og pulsbredde

I en kontrollsløyfe som utfører pulsbreddemodulering brukes arduinofunksjonen **analogRead()** som tar 100 mikrosekunder å gjennomføre.

Avlesing av tid med **micros()** og øvrige beregninger tar så liten tid at det regnes som ubetydelig i denne oppgaven.

Vi har definert en konstant som følger

```
const int VARIGHET = 500; //mikrosekunder
```

Til å lagre tid har vi variabelen

```
unsigned long nåtid;
```

I sløyfen så utføres følgende i sekvens:

- 1 lesing av en måleverdi med **analogRead()**
- 2 nåtid settes til **micros()**
- 3 beregning av pulsbredde med måleverdien
- 4 periodeslutt blir satt til (nåtid + VARIGHET)
- 5 Pulsen settes høy
- 6 Programmet venter til pulsen har fått hele bredden sin
- 7 Pulsen settes lav
- 8 Programmet venter til periodeslutt.

Etter at sekvensen er fullført starter den på ny (om og om igjen).

Hvor lang tid tar en gjennomkjøring av kontrollsløyfen?

Skriv inn et tall: mikrosekunder.

Hva blir frekvensen til kontrollsløyfen?

Skriv inn et tall: Hz

Hva blir maksimal pulsbredde (andel tid pulsen er på) med denne kontrollsløyfen?

Skriv inn et tall mellom 0 og 100: %

Maks poeng: 9

25 Innebygde systemer

Innebygde systemer (embedded systems)

Kryss av for riktig påstand

Velg ett svar

- ☐ Innebygde systemer trenger ikke å ha operativsystem
- ☐ Innebygde systemer må alltid ha et operativsystem
- ☐ Innebygde systemer er ikke optimale å bruke i sanntid da Linux ikke uten videre kjører i 100% sanntid
- ☐ Innebygde systemer er alltid basert på Linux eller varianter av Linux
- ☐ Innebygde systemer med Windows operativsystem vil vanligvis kunne kjøre i 100% sanntid

Maks poeng: 3

26 Busser

Busser

Vi har en robot med femten servomotorer, syv sensorkort, og to kontrollerkort. Begge kontrollerkortene må kunne ta kontroll over bussen, de andre kortene og servoene er slaver. Disse skal alle kobles til samme buss, og vi ønsker oss så få ledninger som mulig, siden det er trangt om plassen i roboten.

Hva slags buss er mest egnet for jobben?

Velg ett alternativ

- ☐ SPI
- ☐ P2P
- ☐ USB
- ☐ UART
- ☐ I2C

Maks poeng: 3

27 Blokkerende kode

Blokkerende kode

Vi har to definisjoner av blokkerende kode: Kode som forhindrer kjøring av all etterfølgende kode, eller kode som forhindrer kjøring av annen kode mens den venter på en hendelse.

Vi definerer kode som ikke-blokkerende eller dersom det ikke blokkerer etter noen av disse definisjonene.

I et innebygd system med arduino har vi kode for pulsbreddemodulering som skal brukes sammen med følgende kode:

```
const int POTMETER_PIN = A3;
const int PWM_PIN = 13;
const int PERIOD_DURATION = 500;
const int ANALOG_MAX = 1024;

int potmeterValue = 0;
unsigned long pulseEnd = 0;
unsigned long periodStart = 0;
unsigned long periodEnd = 0;

void setup(){
  pinMode(POTMETER_PIN, INPUT); //
  digitalWrite(PWM_PIN, LOW); // low before enabling output
  pinMode(PWM_PIN, OUTPUT); //
}

void updatePeriod(){
  potmeterValue = analogRead(POTMETER_PIN);
  periodStart = micros();
  periodEnd = periodStart + PERIOD_DURATION;
  pulseEnd = periodStart + (potmeterValue*PERIOD_DURATION)/ANALOG_MAX;
}

void otherStuff(){
  // ... other stuff that our embedded program does ...
}
```

Hvilket av alternativene under er minst blokkerende?

Velg ett alternativ:



```
void setPWM() {  
    updatePeriod();  
    digitalWrite(PWM_PIN, HIGH);  
    while (currentTime < pulseEnd) {  
    }  
    digitalWrite(PWM_PIN, LOW);  
    while (currentTime < periodEnd) {  
    }  
}  
  
void loop() {  
    setPWM();  
    otherStuff();  
}
```



```
void setPWM() {  
    while (currentTime < pulseEnd) {  
        digitalWrite(PWM_PIN, HIGH);  
    }  
    while (currentTime < periodEnd) {  
        digitalWrite(PWM_PIN, LOW);  
    }  
}  
  
void loop() {  
    updatePeriod();  
    setPWM();  
    otherStuff();  
}
```

```

void setPWM(){
    unsigned long currentTime = micros();
    if (currentTime < pulseEnd) then{
        digitalWrite(PWM_PIN, HIGH);
    }
    else{
        digitalWrite(PWM_PIN, LOW);
    }
    if (currentTime < periodEnd){
        setPWM();
    }
}

void loop(){
    otherStuff();
    updatePeriod();
    setPWM();
}

```

```

void setPWM(){
    digitalWrite(PWM_PIN, HIGH);
    delayMicroseconds(pulseEnd-periodStart);
    digitalWrite(PWM_PIN, LOW);
    delayMicroseconds(periodEnd-pulseEnd);
}

void loop(){
    setPWM();
    updatePeriod();
    otherStuff();
}

```



```
void setPWM(){
  unsigned long currentTime = micros();
  if (currentTime < pulseEnd) then{
    digitalWrite(PWM_PIN, HIGH);
  }
  else{
    digitalWrite(PWM_PIN, LOW);
  }
  if (currentTime > periodEnd){
    updatePeriod();
  }
}

void loop(){
  setPWM();
  otherStuff();
}
```

Maks poeng: 3

28 Regulering

Regulering av vannstand

Vi har et system bestående av en elektrisk styrt ventil og en nivåsensor. Ventilen kan slås på via en motordriver som er styrt fra Arduino. Når ventilen er på fylles det opp vann i en tank, når den er av stenger den for vanntilførselen. Nivåsensoren gir ut verdien logisk HIGH (5V) når vannet i tanken har et ønsket nivå, og logisk LOW (0V) når vannivået er for lavt. Tanken har et utløp som vi ikke har kontroll på, der en mindre mengde vann kan renne ut.

Hvilken kode vil regulere vannstanden korrekt?

Velg ett alternativ

```

// -----

const int ventilPin = 3; // HIGH: slår på ventil, LOW: slår av ventil
const int sensorPin = 4; // HIGH: for mye vann, LOW: for lite vann
int sensorState = 0;

void setup() {
  pinMode(ventilPin, OUTPUT);
  pinMode(sensorPin, INPUT);
}

void loop() {
  sensorState = digitalRead(sensorPin);

  digitalWrite(ventilPin, HIGH);

  while (sensorState == HIGH)
    digitalWrite(ventilPin, LOW);
}

// -----

const int ventilPin = 3; // HIGH: slår på ventil, LOW: slår av ventil
const int sensorPin = 4; // HIGH: for mye vann, LOW: for lite vann
int sensorState = 0;

void setup() {
  pinMode(ventilPin, OUTPUT);
  pinMode(sensorPin, INPUT);
  digitalWrite(ventilPin, LOW);
}

void loop() {
  sensorState = digitalRead(sensorPin);

  if (sensorState == HIGH)
    digitalWrite(ventilPin, LOW);
  if (sensorState == LOW)
    digitalWrite(ventilPin, HIGH);
}

// -----

const int ventilPin = 3; // HIGH: slår på ventil, LOW: slår av ventil
const int sensorPin = 4; // HIGH: for mye vann, LOW: for lite vann
int sensorState = 0;

void setup() {
  pinMode(ventilPin, OUTPUT);
  pinMode(sensorPin, INPUT);
}

void loop() {
  sensorState = digitalRead(sensorPin);

  if (sensorState == HIGH)
    digitalWrite(ventilPin, HIGH);
  else
    digitalWrite(ventilPin, LOW);
}

```

```
// -----

const int ventilPin = 3; // HIGH: slår på ventil, LOW: slår av ventil
const int sensorPin = 4; // HIGH: for mye vann, LOW: for lite vann
int sensorState = 0;

void setup() {
  pinMode(ventilPin, OUTPUT);
  pinMode(sensorPin, INPUT);
  digitalWrite(ventilPin, LOW);
}

void loop() {
  sensorState = digitalRead(sensorPin);

  for (sensorState == HIGH)
    digitalWrite(ventilPin, LOW);
  end (sensorState == LOW)
    digitalWrite(ventilPin, HIGH);
}

// -----

const int ventilPin = 3; // HIGH: slår på ventil, LOW: slår av ventil
const int sensorPin = 4; // HIGH: for mye vann, LOW: for lite vann
int sensorState = 0;

void setup() {
  pinMode(ventilPin, OUTPUT);
  pinMode(sensorPin, INPUT);
}

void loop() {
  sensorState = digitalRead(sensorPin);

  digitalWrite(ventilPin, LOW);

  while (sensorState == LOW)
    digitalWrite(ventilPin, HIGH);
}
```

Maks poeng: 3

i Formelark

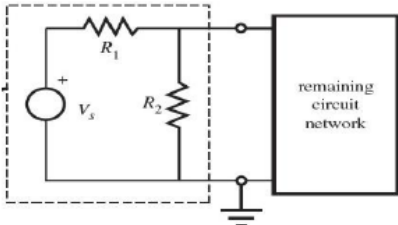
Formelark

Document 2

Attached



Bevegelsesligninger $s(t) = \int v \cdot dt$ $v(t) = ds/dt$ $a(t) = dv/dt$ Konstant hastighet: $s = s_0 + v \cdot t$ Konstant akselerasjon: $a = v/t$ $s = s_0 + v_0 t + at/2$	Sirkelbevegelse $a = v^2/r$ $F = mv^2/r$ $\omega = d\theta/dt$ $\omega = v/r$ $\alpha = \omega/t$ $\omega = 2\pi/T$ $f = \omega/2\pi$
Bevegelsesmengde $p = m \cdot v$ (enkelt partikkel) $p = \sum m_i v_i$ (mange-partikkel) Kraft $F = dp/dt = ma$ $G = mg$, der $g \approx 9,81 \text{ m/s}^2$	Arbeid $W = F \cdot s$ $W = P \cdot t$ (konstant effekt) $P = dW/dt$ $P = V \cdot I$ $P = E/t$
Motorer $k_T = 1/k_v$ $k_v = \omega_{no-load}/V_{peak}$ $\tau = k_T I$ $V_{EMF} = k_v \omega$ $V_{motor} = I \cdot R + L \cdot dI/dt + V_{EMF}$	Desibel notasjon $dB = 10 \log \left(\frac{P_2}{P_1} \right)$ $L_{dB} = 20 \log \left(\frac{L_2}{L_1} \right)$
Sampling $f_{Nyquist} = \frac{f_{sample}}{2}$	Elektronikk $V = RI$ $V_{RMS} = \sqrt{2} V_p / 2$ $X_C = 1/(2\pi f C)$ Utladning kondensator og spole $\tau = RC$ og $\tau = L/R$ $V = V_0 e^{(-t/\tau)}$ Thevenin og Norton $R_{th} = R_{no}$ $V_{th} = I_{no} R_{no}$ $V_{th}/R_{th} = I_{no}$

Gjennomsnittsverdien V_{avg} til et sinussignal (målt over en halv periode) der V_p er lik amplituden	$V_{avg} = \frac{2}{\pi} V_p$
RMS-verdien til et sinussignal der V_p er amplituden	$V_{RMS} = \frac{1}{\sqrt{2}} V_p$
Vinkelfrekvens (=vinkelhastighet)	$\omega = \frac{2\pi}{T} = 2\pi f$
Sammenheng mellom radianer og grader	$2\pi \text{ radianer} = 360^\circ$
Kapazitiv reaktans	$X_C = \frac{1}{2\pi f C}$
Sammenheng mellom ladning, kapasitans og spenning over en kondensator	$Q = VC$
Tidskonstant	$\tau = RC$
<p>Den ekvivalente Thévenin-spenningen V_{th} for en krets med to resistorer i serie med en spenningskilde</p>  <p>Den ekvivalente Thévenin-resistansen R_{th}</p>	$V_{th} = \frac{R_2}{R_1 + R_2} V_s$ $R_{th} = \frac{R_1 R_2}{R_1 + R_2}$
Sammenhengen mellom Norton- og Thévenin-ekvivalenter	$R_{th} = R_{no}$ $V_{th} = I_{no} R_{no}$ $\frac{V_{th}}{R_{th}} = I_{no}$
Spenningen V_C over en kondensator med kapasitans C når den lades ut gjennom en motstand R fra V_0 til $0V$	$V_C = V_0 e^{-\frac{t}{\tau}}$
Spenningen V_C over en kondensator med kapasitans C når den lades opp gjennom en motstand R fra $0V$ til V_s	$V_C = V_s (1 - e^{-\frac{t}{\tau}})$
Sammenheng mellom strømmen gjennom og spenningen over en kondensator	$i_c = \frac{dv_c}{dt}$