



Måling og styring. Sensorer og Arduino

Af Karsten Refsgaard og Ole Sørensen

I disse noter vil vi vise, hvordan man kan arbejde med områderne sensorer og styring. Vi lader eleverne konstruere et simpelt måleapparat, for eksempel et elektrisk termometer. Ingredienserne kan være en temperaturfølsom resistor i en spændingsdeler.

Sensorerne kan betragtes som en "black box", eller man kan se på sensorens fysik: Hvordan virker en temperaturfølsom resistor, en transistor, en trykmåler etc. Styring illustreres ved hjælp af Arduino. Eleverne konstruerer en alarm, hvor en rød lampe lyser, hvis temperaturen bliver for høj. Ingredienserne er det elektriske termometer samt en lysdiode. Vi vil præsentere et lille udvalg af de mange billige sensorer, der findes til Arduinoen, bl.a. afstandsmåler og bevægelsesføler, som kan anvendes i forløb, hvor elevernes innovative evner udfordres. Hvad skal til for at kunne gå fra laboratiemodellen til et færdigt produkt?

Side 4 til 30 er skrevet af Ole Sørensen, Fredericia Gymnasium.

Side 31 til 39 er skrevet af Karsten Refsgaard, Vejen Gymnasium

Vi har ikke gjort meget for at få bidragene til at hænge sammen. Der vil være gentagelser. Til gengæld kan de enkelte dele bruges uafhængigt af hinanden

Forudsætninger

For at få udbytte af disse noter, skal eleverne kende til følgende begreber og lovmæssigheder fra indledende ellære:

Strømstyrke, spændingsforskel, resistans, Ohms lov, serieforbindelse, parallelforbindelse, (U,I)-karakteristik

Enkelte rettelser 16. april 2018.

Indholdsfortegnelse

1. Spændingsdeler	4
Eksempel 1.1	4
Opgave 1.2	4
Opgave 1.3	4
Eksempel 1.4. Beskyttelsesresistor	5
Opgave 1.5 Rød lysdiode.....	5
Eksempel 1.6. Arbejds punkt	6
Eksempel 1.7 Temperaturfølsom modstand.....	8
Opgave 1.8	8
Eksperimenter med spændingsdeling	9
Eksperiment 1.1	9
Eksperiment 1.2 Temperaturfølsom modstand 1.....	9
Eksperiment 1.3 Temperaturfølsom modstand 2.....	9
Eksperiment 1.4	11
Efterbehandling af forsøg med temperaturmåling	12
2. Fysikken bag sensorerne	14
Lidt faststoffysik	14
Termometer med resistor	15
Termometer med halvlederkomponent	16
Lysfølsom komponent	16
Kraftmåler.....	16
Opgave (med udfordringer!).....	18
3. ARDUINO	18
Beskrivelse af Arduino.....	18
Hvorfor arbejde med Arduino?	19
At programmere eller kode	19
I gang med ARDUINO	20
0. Installation.....	20
1. Opsæt kommunikationsport.....	20
2. Skriv eller vælg et program.....	21
3. Opbyg et kredsløb	21
4. Upload program til Arduino	21
Eksperiment 4.1. Dit første Arduino-program	21
Eksperiment 4.2	22
Programmere Arduino	22
Arduinos ind- og udgange	24
Eksperiment 4.3	25
Eksperiment 4.4. Lysdiode på breadbord	25

Måling med Arduino.....	27
Opgave 4.5.....	27
Eksperiment 4.6. Analog måling.....	27
Eksperiment 4.7. Forbedret analog måling.....	28
Eksempel 4.8 Bygge en alarm.....	30
4. Små Projekter.....	32
1. Trafiklys 1.....	32
2. Trafiklys 2.....	32
3. Trafiklys 3.....	32
4. Temperatur-alarm.....	32
5. Belysnings-styring.....	34
6. Vandhøjde-alarm.....	34
7. Bevægelses-sensor.....	34
8. Pumpe.....	35
Større Arduino-eksperimenter.....	37
Arduino-eksperiment 1. Blinkende dioder og løbelys.....	37
Arduino-eksperiment 2. Ringeklokke, fodgængersignal og bevægelsesføler.....	41
Et fodgængerlys med lyd.....	42
Arduino-eksperiment nr. 3. Afstandsmåler.....	45

1. Spændingsdeler

På figur 1 er vist en spændingsdeler

Kredsløbet er forsynet med en spændingskilde, der afgiver en fast spænding U_0

Desuden indeholder kredsløbet to resistorer R_1 og R_2 i serieforbindelse. Spændingsforskellen over R_2 måles med voltmeteret U . Der løber ikke strøm gennem voltmeteret, da voltmeteret har meget høj indre modstand.

Resistansen af seriekoblingen er som bekendt $R_s = R_1 + R_2$.

Strømstyrken beregnes ved hjælp af Ohms lov til $I = \frac{U_0}{R_1 + R_2}$

Derfor er spændingsforskellen over R_2 : $U = R_2 \cdot I = R_2 \cdot \frac{U_0}{R_1 + R_2}$

Hvilket også kan skrives ("spændingsdeler-ligningen")

$$U = \frac{R_2}{R_1 + R_2} U_0$$

Spændingen U_0 deles mellem de to resistorer i samme forhold som resistanserne.

Eksempel 1.1

Hvis $U_0=5,0\text{V}$, $R_1=400\Omega$, $R_2=100\Omega$ er $U=1,0\text{V}$

Opgave 1.2

Beregn U hvis $U_0=12,0\text{V}$, $R_1=1000\Omega$, $R_2=100\Omega$

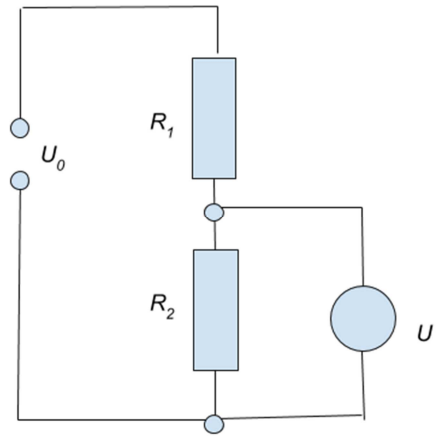
Opgave 1.3

I en bestemt opstilling ønsker vi at måle spændingen U_0 .

Spændingskilden er indrettet, så U_0 kan varieres mellem 0 V og 24 V.

Vores voltmeter kan desværre højst måle 6 V, derfor opbygger vi en spændingsdeler.

- Tegn et kredsløbsdiagram
- Foreslå værdier for R_1 og R_2 , så vi kan måle det vi ønsker.
- Skriv en instruktion til den medarbejder der skal aflæse voltmeteret og beregne U_0
- Beregn strømstyrken gennem serieforbindelsen med disse valg af R_1 og R_2 .
- Spændingskilden U_0 må højst afgive strømstyrken 10mA.
Hvilke værdier af R_1 og R_2 giver en strømstyrke på højst 10mA?



Figur 1. Spændingsdeler

Eksempel 1.4. Beskyttelsesresistor

En diode har en typisk (U,I) -karakteristik som vist på figuren.

Det ses at karakteristikken er stærkt u-lineær, i modsætning til karakteristikken for en resistor, der som bekendt er en proportionalitet.

Bliver strømstyrken for stor, ødelægges dioden.

I elektroniske kredsløb bruges ofte lysdioder.

Lysdioderne beskyttes ved at sidde i serieforbindelse med en passende resistor.

Vi vil se et eksempel.

En diode sidder i serie med en resistor R .

En spændingskilde sørger for at spændingen over serieforbindelsen er konstant $U_0=5,00V$

Tegn kredsløbsdiagrammet.

Vi ønsker at strømmen gennem dioden skal være 17mA.

På grafen ses at det svarer til en spændingsforskel over dioden på 0,70V.

Det betyder at spændingsforskellen over beskyttelsesresistoren skal være

$$U=5,00V-0,70V=4,30V$$

Samtidig er strømmen gennem resistoren 17mA. Dens resistans skal så være $R = \frac{U}{I} = \frac{4,30V}{0,017A} = 253\Omega$

Opgave 1.5 Rød lysdiode

Hvis du har lavet eksperiment 1.1 side 8 kan du bruge dine måleresultater i denne opgave.

Ellers benyt følgende måleresultater:

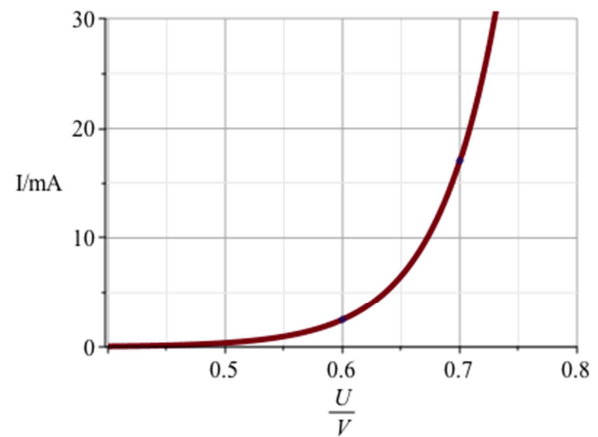
U / V	1,50	1,60	1,70	1,80	1,90	2,00
I / mA	0,06	0,17	0,52	1,56	4,71	14,22

- Tegn (U,I) -karakteristikken for dioden
- Udfør eksponentiel regression på resultaterne og find forskriften for I som funktion af U .

Dioden anbringes nu i serie med en resistor med resistansen $R=220\Omega$.

Resistor og diode tilsluttes en spændingskilde på $U_0=5,0V$.

- Tegn et kredsløbsdiagram.
- Opskriv et udtryk for spændingsforskellen U_R over resistoren udtrykt ved U_0 og U .
- Opskriv ved hjælp af Ohms lov samt udtrykket fra d. et udtryk for strømstyrken gennem resistoren.



Figur 2. (U,I) -karakteristik for diode

- f. Beregn strømstyrken gennem serieforbindelsen samt spændingsforskellen over dioden.

Tips: Strømstyrken gennem resistor og diode er jo den samme (hvorfor?)

Brug dette samt resultaterne fra b. og e. til at opstille en ligning, hvor U er den ubekendte.

Løs ligningen med CAS og find U (dvs. spændingsforskellen over dioden)

Indsæt til sidst i udtrykket fra b. eller e. for at finde strømstyrken.

Eksempel 1.6. Arbejdspunkt

Diagrammet viser en del af et kredsløb med en diode i serie med en resistor med resistans $R=220\Omega =0,22V/mA$.

Vi vil bestemme spændingen U ved en grafisk metode.

(Husk at når vi siger "spændingen U " mener vi egentlig "spændingsforskellen mellem U og nulpunktet).

Spændingsforskellen over resistoren er i følge Ohms lov og med I målt i mA:

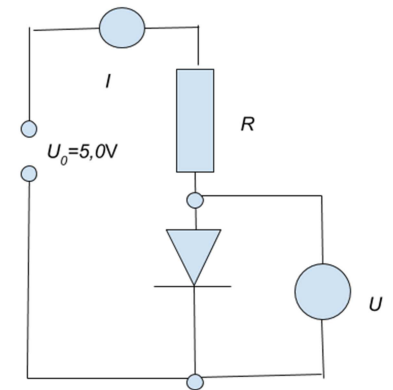
$$U_R = R \cdot I = 0,22V/mA \cdot I$$

Derfor er spændingen $U = 5,0V - U_R$

$$U = 5,0V - 0,22V/mA \cdot I$$

Samtidig kan spændingen U som funktion af I findes af diodens (I,U) -karakteristik.

(I,U) -karakteristikken findes ved at bytte om på akserne i forhold til (U,I) -karakteristikken



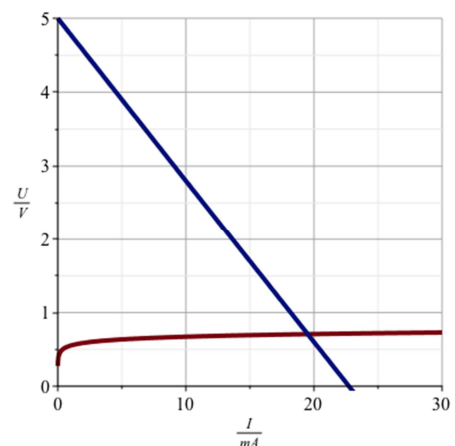
Figur 3. Diode i serie med resistor

På figuren er de to tegnet grafer for U i samme koordinatsystem.

Skæringspunktet svarer til at der går samme strøm gennem diode og resistor samtidig med at U er den samme. Det ses at svare til $I=19,5mA$, $U= 0,82V$.

Dette punkt kaldes kredsløbets *arbejdspunkt*.

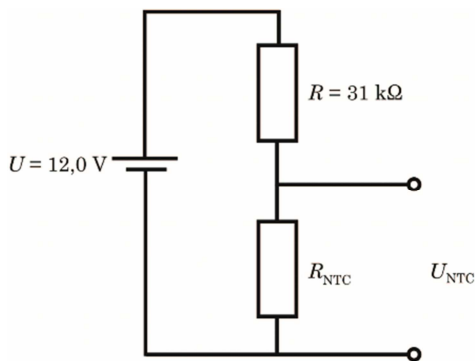
Tilsluttes et voltmeter til spændingskildens minus-pol og punktet mellem resistoren og dioden (se figur 3), vil det vise 0,82 V.



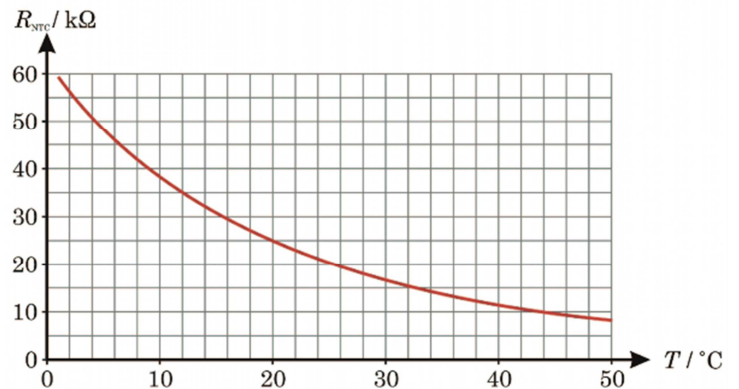
Figur 4. Arbejds punkt

Eksempel 1.7 Temperaturfølsom modstand (Inspireret af stx fysik 25.maj 2016)

Termoføler i bil



Figur 5. Spændingsdeler med NTC-resistor



Figur 6. (T,R)-graf for NTC-resistor

I en bil måles temperaturen mange steder. Ofte måles temperaturen med en NTC-resistor (NTC: negativ temperatur coefficient) hvor resistansen er en aftagende funktion af temperaturen, modsat dypperesistoren i Eksperiment 1.2.

Ved hjælp af formlen for spændingsdeling samt omsætningsfunktionen mellem resistans og temperatur kan temperaturen bestemmes.

Opgave 1.8

En bestemt termoføler er opbygget som figuren i eksempel 1.7.

- Hvad er temperaturen, når $R_{NTC} = 25 \text{ k}\Omega$?
En dag måles spændingen U_{NTC} til 3,9 V
- Beregn R_{NTC} ved hjælp af "spændingsdeler-ligningen"
- Bestem temperaturen

Eksperimenter med spændingsdeling

Eksperiment 1.1

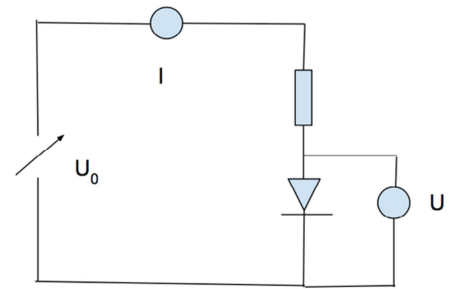
Mål (U,I)- karakteristikker for lysdioder. Vælg for eksempel skiftevis en rød, grøn og gul lysdiode.

Husk beskyttelsesresistor, f.eks. 220Ω

Ved ca. hvilke værdier af strøm og spænding begynder dioderne at lyse kraftigt?

Tegn (U,I)-karakteristikkerne i samme koordinatsystem.

Kommenter eventuelle forskelle.



Figur 4. Kredsløbsdiagram med diode

Eksperiment 1.2 Temperaturfølsom modstand 1

Indledende kvalitativt forsøg.

Spænd en tynd ståltråd op mellem to holdere. Forbind med krokodillenæb enderne til et ohmmeter. Opvarm ståltråden med en gasbrænder og hold øje med ohmmeteret.

Hvad sker? Forklaring?

Eksperiment 1.3 Temperaturfølsom modstand 2.

I dette forsøg udnyttes at resistansen i en metaltråd vokser med temperaturen.

Når vi kender sammenhængen mellem resistans og temperatur, kan vi bruge metaltråden som termometer.

I skal vælge ét af følgende to typer udstyr:

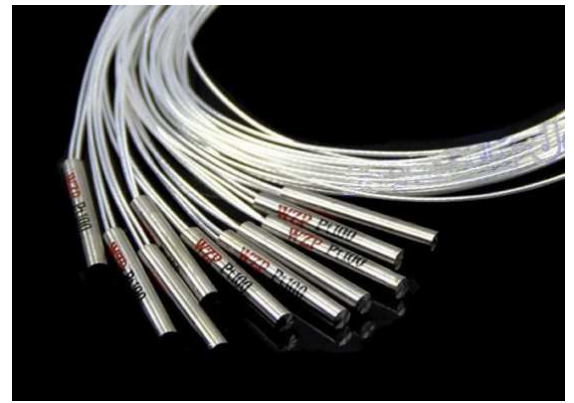
- Dypperresistor af kobber
- Platin-resistor Pt100.

Udstyr:

- Kobber eller platin-resistor
- Beskyttelsesresistor, (100 Ohm til Cu-resistoren, 500Ω til Pt)
- Elkedel
- Spændingskilde
- Voltmeter, gerne tilsluttet LoggerPro eller andet dataopsamlingsudstyr
- Termometer, gerne tilsluttet LoggerPro.



Figur 8. Dypperesistor med kobbertråd



Figur 9. Platin- resistorer

Opbyg et kredsløb som vist på figur 10.

Opbygning af måleopstilling

Den temperaturfølsomme resistor R_2 anbringes i elkedlen, der fyldes med vand.

Anbring et termometer i elkedlen.

Opbyg et kredsløb som vist på diagrammet.

R_1 er en resistor, der skal beskytte R_2 mod for høj strømstyrke og dermed ødelæggelse. Vælg 100Ω til

kobber og 500Ω til Platin.

Spændingsforskellen på spændingskilden indstilles på $10,0V$. Spændingen må ikke ændres under forsøget.

Tilslut et voltmeter til at måle spændingsforskellen U over R_2 .

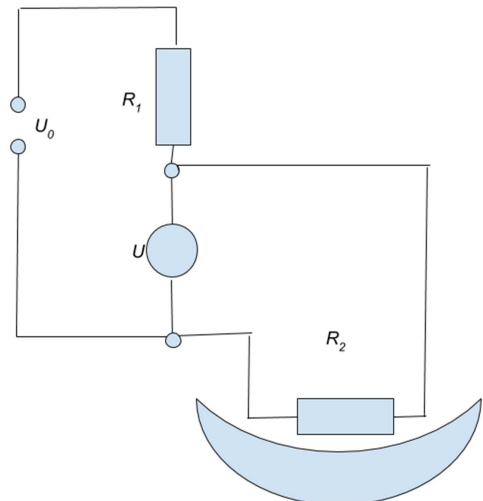
I kan vælge at måle U og temperaturen T med computer (f.eks. LoggerPro). Det er ikke nødvendigt at måle oftere end med 5-10 sekunders interval.

Måling

Mens vandets temperatur langsomt øges, måles og noteres sammenhørende værdier af temperatur T og spændingsforskel U .

Databehandling

Beregn for hver måling dypperesistorens resistans R_2



Figur 10. Kredsløb med temperaturfølsom resistor. "Bananen" symboliserer elkedlen.

Lav en (T,U) og en (T, R_2) -graf.

Find en forskrift for T som funktion af U .

Fremgangsmåden ligner den fra kemi velkendte(?) metode med at lave en standardkurve og derefter finde koncentrationen af en ukendt opløsning ved at aflæse på standardkurven.

Anvendelse

Tilbered et bæger med varmt vand, hvis temperatur I kender.

Giv bægeret til en anden gruppe, og bed dem fortælle jer vandets temperatur.

Ekstra

Inden for et "rimeligt" temperaturområde afhænger metaltrådens resistans R_2 lineært af temperaturen T .

Man har indført begrebet resistansens temperaturkoefficient α

$$R_2 = R_0 \cdot (1 + \alpha \cdot T)$$

R_0 er resistansen ved 0°C .

Udregn R_0 og temperaturkoefficienten for jeres dypperesistor. Sammenlign med en tabelværdi (Databog).

Eksperiment 1.4

Opbyg et kredsløb som vist på figur 3.

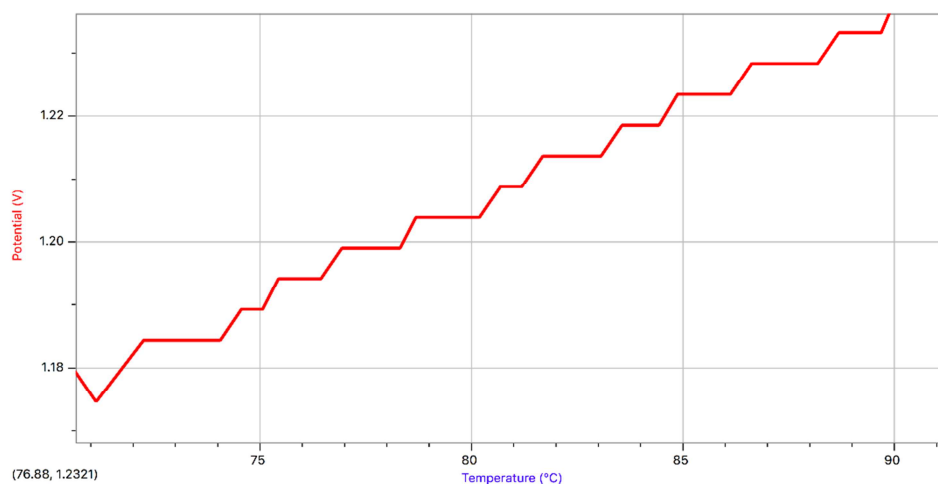
Mål spændingen U (dvs. spændingsforskellen mellem U og 0)

Indsæt et amperemeter og mål strømstyrken gennem resistoren.

Efterbehandling af forsøg med temperaturmåling

1. Lav en (T,U) -graf, hvis I ikke allerede har gjort det.
I LoggerPro: Insert-> Graph. GraphOptions-> Axes Options-> X-axis sættes til Temperature
2. Lav lineær regression på den retlinjede del af grafen.
Angiv regneforskrift for spændingen som funktion af temperatur $U(T)$
3. Realitetstjek:
Hvad burde værdien af U være, med spændingskilden indstillet på $U_0=10,0V$, beskyttelsesresistans på $R=500\Omega$ og platinresistorens lovede resistans på $R_T=100,0\Omega$?
Tip: Brug spændingsdelerligningen: $U = \frac{R_T}{R+R_T} \cdot U_0$
4. Hvis jeres værdi for U afviger fra den forventede, prøv da at angive nogle årsager.
5. Vi har nu bygget et simpelt elektrisk termometer.
Hvor langt er vi nået; dvs. hvad kan vores termometer?
Hvad mangler før termometeret er brugbart og klar til produktion?
6. Omskriv spændingsdelerligningen, så R_T står alene, dvs. så R_T bliver en funktion af U
7. Lav en ny kolonne i LoggerPro eller regneark, hvor I beregner R_T ved hjælp af jeres nye formel.
LoggerPro: Data->New Calculated Column.
8. Lav en (T, R_T) -graf og lav lineær regression. find en forskrift for R_T
Sammenlign med udtrykket for R_T nævnt under "ekstra" side 10.
9. Find en værdi for resistansens temperaturkoefficient α
Sammenlign med en tabelværdi for platin (Databog)
10. Ser vi nærmere på tabellen eller grafen, ses nogle tydelige spring i måleværdierne for spændingsforskellen U .
Det er ikke fordi spændingen i virkeligheden ændres i spring.
Prøv at give forslag til en forklaring inden du læser videre.
Forklaring: Spændingen varierer i virkeligheden glat (kontinuert).
Men når computeren læser spændingen, omsættes den til et helt tal, for eksempel mellem 0 og 1023. Det kaldes Analog til digital konvertering (ADC). På billedet har vi zoomet meget ind på grafen, der derfor ser "takket" ud.

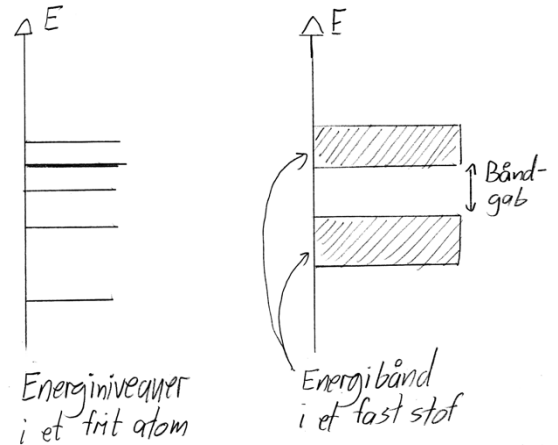
	Time (s)	Potential (V)
1	0	1.209
2	2	1.209
3	4	1.204
4	6	1.033
5	8	1.033
6	10	1.033
7	12	1.033
8	14	1.033
9	16	1.033
10	18	1.033
11	20	1.033
12	22	1.033
13	24	1.038
14	26	1.038
15	28	1.038
16	30	1.038
17	32	1.043
18	34	1.043
19	36	1.043
20	38	1.048
21	40	1.048
22	42	1.052
23	44	1.056



2. Fysikken bag sensorerne

Lidt faststoffysik

I et atom findes elektronerne som bekendt i nogle bestemte tilstande, ofte kaldet "skaller". Hver skal svarer til en bestemt energi. Dette afbildes ofte i et såkaldt *energiniveaudiagram*. For et frit atom er energiniveaudiagrammet kendetegnet ved helt bestemte energier som vist i figur 12.



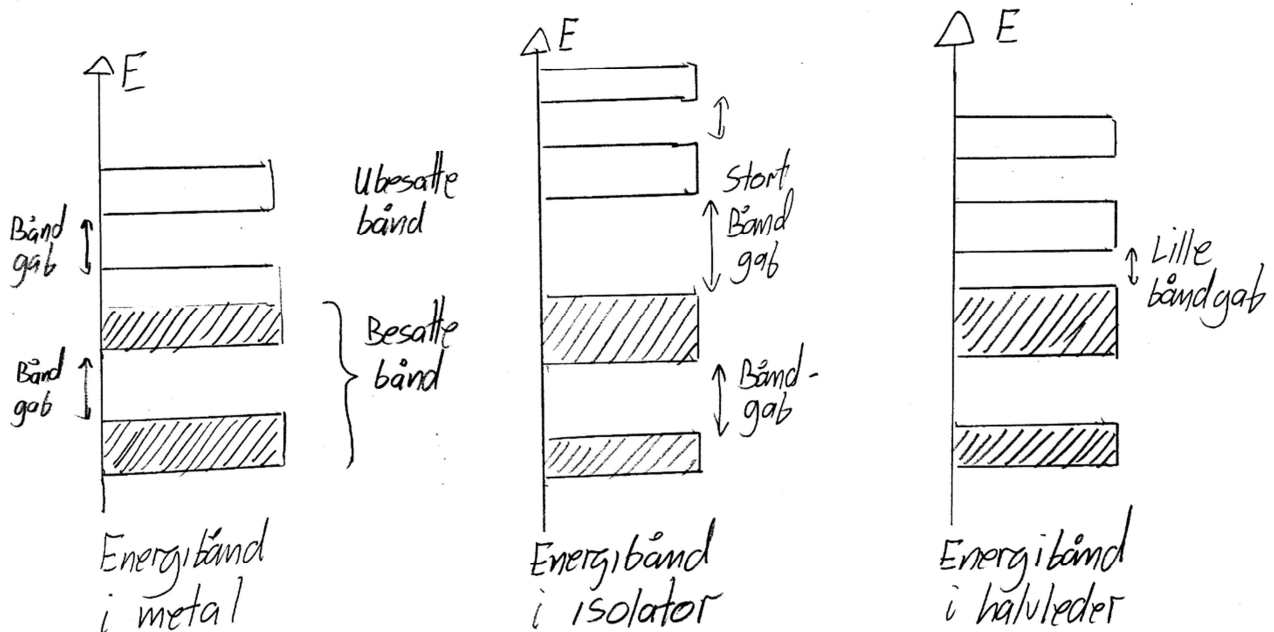
I et fast stof sidder mange atomer tæt sammen i en gitter-struktur. Elektronerne i de enkelte atomer påvirker hinanden, og energiniveauerne flyder sammen til mange tætliggende energiniveauer, der tilsammen danner et *energibånd*. Se figur 12.

Figur 12

Elektronerne i et energibånd indgår i en pulje fælles for mange atomer. Det er ikke knyttet til et bestemt atom, men kan bevæge sig næsten frit gennem gitterstrukturen.

Mellem energibåndene er der *båndgab*, her kan der ikke være elektroner.

Energibåndene fyldes med de tilgængelige elektroner, således at bånd med lavest energi fyldes først.



Figur 13

I et *metal* er det højest besatte energibånd kun delvist fyldt.

En lille elektrisk påvirkning kan derfor løfte elektronerne lidt energimæssigt, de kan bevæge sig, og dermed skabe en elektrisk strøm.

Det energibånd, hvor elektronerne kan bevæge sig og skabe strøm, kaldes *ledningsbåndet*.

I en *isolator* er et energibånd helt fyldt. Det kræver meget energi at ”løfte” elektroner over båndgabets til det næste, tomme energibånd. Det er vanskeligt at skabe en strøm i dette materiale, derfor kaldes det en isolator.

I nogle materialer er der et meget lille båndgab, eller der er nogle få elektroner i det øverste besatte bånd. I dette tilfælde leder materialet ikke strømmen særlig godt. Vi taler om en *halvleder*.

Det vigtigste eksempel på et halvledermateriale er *silicium*.

I helt rent silicium er der kun få elektroner, der kan bevæge sig. Typisk under én fri elektron pr million atomer. Derfor er ledningsevnen lav.

Hvis man tilsætter små mængder af andre atomer til silicium, kan ledningsevnen forøges væsentligt. Hermed kan man styre de elektriske egenskaber af sin silicium-krystal. For eksempel så nogle områder af krystallen leder godt, mens naboområder isolerer, eller kun tillader strøm i én retning – en *diode*.¹

Termometer med resistor

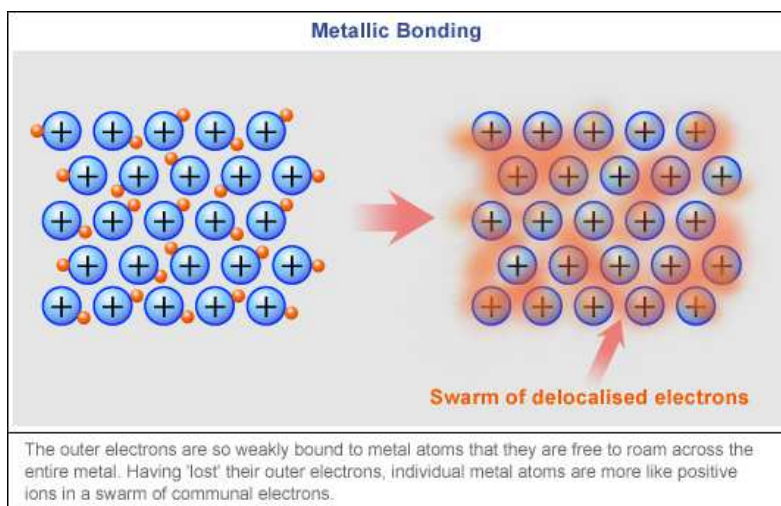
Nogle elektriske termometre udnytter at resistansen i en metaltråd vokser med temperaturen.

I et metal på fast form sidder atomerne på faste pladser i et mere eller mindre regelmæssigt gitter.

Atomerne vibrerer en lille smule i forhold til deres faste pladser. Jo højere temperatur desto kraftigere bevægelse.

Når der går en elektrisk strøm gennem metallet, betyder det at de yderste løst bundne elektroner i metalatomerne bevæger sig gennem metallet.

Hvis metalatomerne på grund af høj temperatur vibrerer i forhold til deres faste pladser, øges sandsynligheden for at elektronerne støder ind i metalatomerne. Det bliver lidt vanskeligere for elektronerne at passere. Resultatet er at resistansen stiger.



¹ F. Elvekjær og B.D.Nielsen: *Stråling og atomer*, Gads forlag, 1984

Inden for et "rimeligt" temperaturområde afhænger metaltrådens resistans R_T lineært af temperaturen T . Se også side 10.

Termometer med halvlederkomponent

I omtalen af halvledere nævnte vi at der kun var ganske få elektroner, der havde mulighed for at bevæge sig og dermed skabe en elektrisk strøm. Hvis halvledermaterialet varmes op, tilføres energi til elektronerne. Herved får flere elektroner mulighed for at bevæge sig. Den elektriske strøm vokser. Man kan også udtrykke det ved at resistansen falder.

For en diode afhænger resistansen kraftigt af temperaturen. Men i modsætning til metaller aftager resistansen med temperaturen.

Kender man sammenhængen mellem temperatur og resistans, kan dioden bruges som termometer.

Lysfølsom komponent

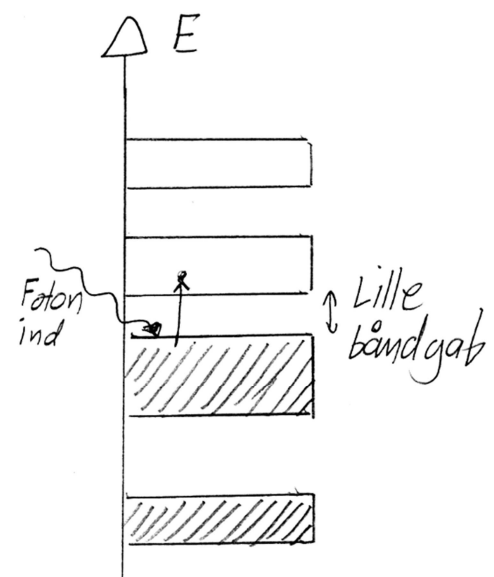
I en halvleder sidder elektronerne som nævnt i energibånd, hvor det højeste fyldte bånd for eksempel er næsten fyldt. Der er kun få elektroner, der kan bevæge sig.

Men hvis halvlederen belyses, kan lysets fotoner overføre deres energi til elektroner i det fyldte bånd, så elektronerne løftes til et bånd over båndgabet. Her kan de skabe en elektrisk strøm.

Lyser vi på halvlederen, falder dens resistans. Vi har således en lysfølsom elektrisk komponent.

Fotens energi er som bekendt proportional med lysets frekvens: $E = h \cdot f$, hvor f er lysets frekvens.

Det er derfor kun lys med tilpas høj frekvens (og dermed tilpas lav bølgelængde, der er i stand til at excitere elektroner til ledningsbåndet og dermed gøre komponenten til en elektrisk leder.



Figur 15. En foton med passende energi kan excitere en elektron til ledningsbåndet.

Kraftmåler

I en elektrisk kraftmåler benyttes en lang tynd tråd af metal eller halvledermateriale.

Når man trækker i tråden, bliver den længere og tyndere. Herved øges resistansen.

Det viser sig at resistansen kun vokser lidt, men nok til at måle. Kender vi sammenhængen mellem kraft og resistansændring, kan kraften bestemmes.

Der gælder følgende for resistansen af en tråd:

$$R = \rho \cdot \frac{L}{A}$$

L er længden, A tværsnitsarealet, mens resistiviteten ρ er en materialekonstant

For kobber er $\rho = 0,0168 \cdot 10^{-6} \Omega\text{m}$ ved 20°C .

Eksempel

Vi vil beregne resistansen af en 1,0m lang kobbertråd med diameter $0,25\text{mm} = 2,5 \cdot 10^{-4}\text{m}$.

Tværsnitsarealet A beregnes:

$$A = \pi \cdot r^2 = \pi \cdot (1,25 \cdot 10^{-4}\text{m})^2 = 4,91 \cdot 10^{-8}\text{m}^2$$

$$R = 0,0168 \cdot 10^{-6} \Omega\text{m} \cdot 1,0\text{m} / 4,91 \cdot 10^{-8}\text{m}^2 = 0,34\Omega$$

Trækker man i tråden med en vis kraft F, vil snoren strækkes et stykke ΔL

Sammenhængen er givet ved *Hookes lov*:

$$F = k \cdot \Delta L$$

k kaldes fjederkonstanten og afhænger normalt af flere ting: fjedermateriale, fjederens tykkelse og længde mm.

Er der tale om en tråd eller en stang af et bestemt materiale, kan man foretage nogle omskrivninger:

$$F = k \cdot \Delta L \Leftrightarrow F = \frac{k \cdot L}{A} \cdot A \cdot \frac{\Delta L}{L}$$

I mange situationer er $\frac{k \cdot L}{A}$ en materialekonstant, der kaldes *elasticitetsmodul* eller *Youngs modul*. Dvs. $E = \frac{k \cdot L}{A}$

$$F = EA \frac{\Delta L}{L} \Leftrightarrow \Delta L = \frac{F \cdot L}{E \cdot A}$$

For kobber er elasticitetsmodul $E = 130 \cdot 10^9 \text{N/m}^2$

Hvis vi belaster vores kobbertråd med 10 N, svarende til at vi hænger et lod på ca. 1 kg op i tråden,

vil trådens længde blive forøget med 1,6 mm:

$$\Delta L = \frac{F \cdot L}{E \cdot A} = \frac{10\text{N} \cdot 1\text{m}}{130 \cdot 10^9\text{N/m}^2 \cdot 4,91 \cdot 10^{-8}\text{m}^2} = 0,0016\text{m}$$

Vi kan vurdere hvor meget arealet formindskes.

Vi gør den (rimelige) antagelse at kobberets volumen V er uændret. Volumen af tråden kan skrives $V = L \cdot A = (L + \Delta L)(A - \Delta A)$

Løser vi ligningen med hensyn til ΔA fås

$$\Delta A \approx -A \frac{\Delta L}{L}$$

Vi har smidt et led væk: Da ΔA og ΔL er små (størrelsesorden 0,16%) vil $\Delta A \cdot \Delta L$ være meget lille (størrelsesorden $2,5 \cdot 10^{-4} \%$).

Sættes de nye værdier af længde og tværsnitsareal ind i resistansformlen, beregnes den nye resistans.

Resultatet bliver at resistansen forøges med ca. 0,32%

Det er ikke meget, men heldigvis er det muligt at måle.

Opgave (med udfordringer!)

- Gennemregn detaljerne i udledning af udtrykket $\Delta A \approx -A \frac{\Delta L}{L}$
- Vis at når tråden er strakt et lille stykke ΔL kan resistansen udtrykkes $R = \rho \cdot \frac{L+\Delta L}{A+\Delta A} = \rho \frac{L(1+\frac{\Delta L}{L})}{A(1-\frac{\Delta L}{L})}$

Vi kalder resistansen af den ubelastede tråd R_0 $R_0 = \rho \cdot \frac{L}{A}$

Vi indfører den relative længdeforøgelse ved variabelen $x = \frac{\Delta L}{L}$.

- Vis at forholdet mellem resistansen af den belastede og ubelastede tråd kan skrives

$$f(x) = \frac{R}{R_0} = \frac{1+x}{1-x}$$

Vi har nu omformuleret opgaven til at undersøge funktionen $f(x)$ for x tæt på 0.

- Beregn $f'(0)$, f.eks. med dit CAS-værktøj, og find en ligning for tangenten til grafen for f i $x_0=0$
- Tegn grafen for f og den fundne tangent i intervallet $0 \leq x \leq 0,1$
Hvor lille er den største afvigelse mellem den "rigtige" værdi $f(x)$ og den tilnærmede værdi fundet ved hjælp af tangenten?
- Forklar påstanden ovenfor: "Når længden forøges med 0,16% forøges resistansen med ca. 0,32%"
- Opstil en formel (gældende for små længdeforøgelser) der for den pågældende tråd giver trækraften F udtrykt ved resistansforholdet $f = \frac{R}{R_0}$ Tip: Der er ligefrem proportionalitet mellem f og F .
- Gennemfør tilsvarende beregninger for andre tråde. Resistivitet og elasticitetsmodul kan f.eks. slås op i Databogen²

3. ARDUINO

Beskrivelse af Arduino

Arduino er et lille stykke elektronik, der kan

- styre ting i dens omverden: lamper, motorer, ventiler, højttalere mm.
- måle på variable i dens omverden. Tryk på knap, elektriske spændinger mm.
Signalerne kommer ofte fra en sensor, der omsætter f.eks. en temperatur til en spænding

² E. Strandgaard Andersen, P. Jespersgaard og O. Grønbæk Østergaard: *DATABOG Fysik Kemi*, F&K Forlaget

- programmeres. Dvs. instrueres i at foretage bestemte handlinger



Billedkilde: www.arduino.cc

Arduino kan arbejde når den er forbundet til en computer. Når den er programmeret og får strøm fra f.eks. et batteri, kan den arbejde på egen hånd.

Der findes mange typer Arduino, der er specialiserede til bestemte formål.

Vejledning og programmer er skrevet til basismodellen Arduino UNO.

Hvorfor arbejde med Arduino?

Et af de mange nye begreber der høres er *Internet of Things*. At en lang række ting lige fra biler og køleskabe til løbesko og tøj forbindes til internet. Tingene kan udveksle informationer.

For eksempel kan dine løbesko registrere mulig overbelastning af anklerne og sende advarslen videre til dit løbe-ur.

Som almindelig borger er en vis indsigt i programmering og styring vigtig for at følge med i udviklingen inden for it.

Sidst men ikke mindst er it et område hvor beskæftigelsesmulighederne er særdeles gode.

At programmere eller kode

At skrive kode kræver en systematisk tankegang. Der er flere steder på nettet, hvor man kan lære kodning. Tag for eksempel et kig på hourofcode.com/dk

Her kan man blandt andet lære kodning mens man hjælper Disney-figuren Moana med at fange fisk og bekæmpe pirater: hourofcode.com/moana

I gang med ARDUINO

I denne øvelse skal du skrive dit første Arduino-program og få Arduino til at blinke

0. Installation

Hvis du ikke har ARDUINO softwaren på din computer skal softwaren først hentes og installeres:

www.arduino.cc/en/Main/Software

Under *Download the Arduino IDE* vælges relevant version (Windows, MAC etc.)

Følg anvisningerne.

1. Opsæt kommunikationsport

Åben Arduino programmet

Tilslut Arduino til en USB port på computeren.

Vælg menupunkt Tools (eller Værktøjer, hvis computeren taler dansk)

Vælg Port

På *MAC* vælges f.eks. `/dev/cu.usbmodem1421` (Arduino/Genuino Uno)

I *Windows* vælges f.eks. COM3 eller COM4

Windows 8 kan drille.

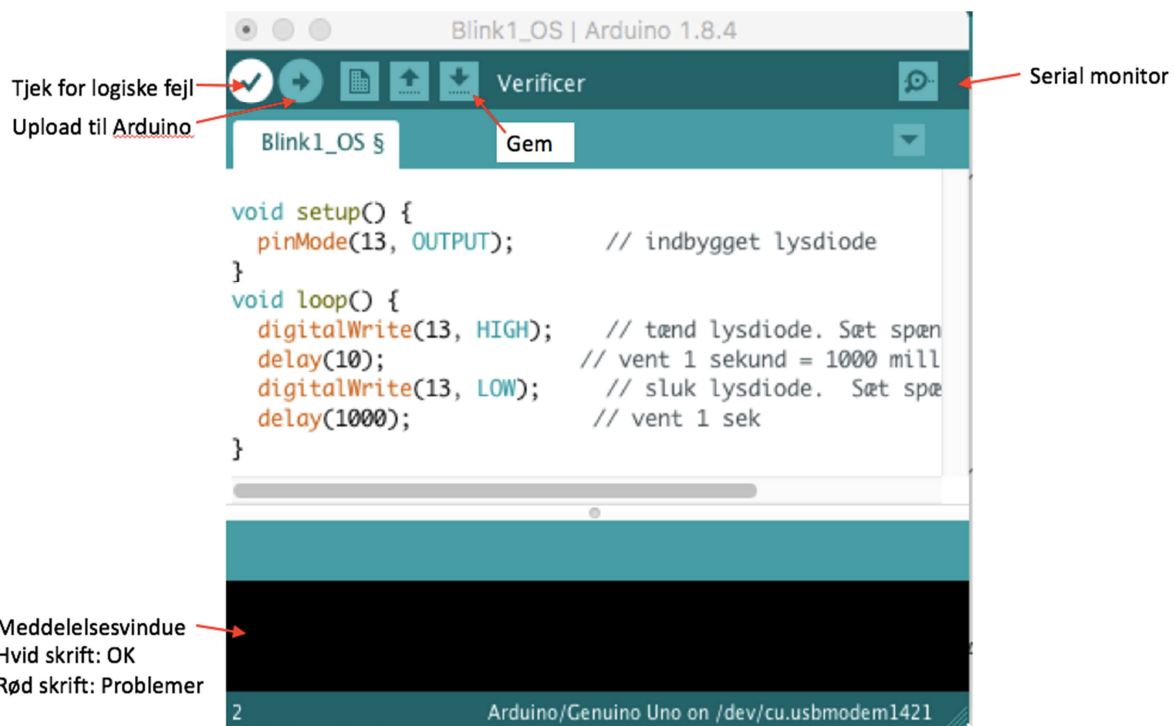
Normalt genkendes Arduino automatisk lige som andre USB-enheder.

Nogle Arduino bruger en ikke-standard, men billig kommunikationschip. Den kan for eksempel hedde R3 CH340. Computeren kan have svært ved at genkende en sådan Arduino.

Løsningen er at hente en *driver* på producentens (kinesiske) hjemmeside. Der er hjælp til dette på internet. Søg på chip-navnet.

Tjek evt. kommunikationen ved klik på Serial monitor øverst til højre i programvinduet.

```
(1000);
```



2. Skriv eller vælg et program

Der er mange gode eksempler i menupunkt *Fil -> Eksempler*

Du lærer hurtigt at kopiere eksemplerne og lave dine egne programmer

Kontroller programmet for logiske fejl (f.eks. manglende } -parenteser) ved at klikke på fluebenet øverst til venstre i programvinduet.

Hvis du ændrer programmet, så husk at gemme.

3. Opbyg et kredsløb

Arduino kan ikke meget i sig selv (ud over at blinke med en lille lysdiode).

Alt afhængigt af programmet skal Arduino tilsluttes *sensorer* (så den kan måle noget i dens omverden) eller *aktuatorer* (så den kan styre noget i dens omverden)

En aktuator kan f.eks. være en lysdiode. Men også højttaler, relæ, motor mm.

Mere om dette i eksemplerne herunder.

4. Upload program til Arduino

Klik på højrepilen øverst til venstre i programvinduet.

Arduino begynder at udføre programmet straks efter det er uploaded.

Eksperiment 4.1. Dit første Arduino-program

Skriv følgende program. Skriv krøllede parenteser { og } samt semikolon ; præcis som der står i teksten.

Eventuelt kan du undlade "//" og al tekst efter "//"

Upload til Arduino og få en indbygget lysdiode til at blinke.

```
void setup() {                                // Tekster efter "//" er kommentarer til læseren
  pinMode(13, OUTPUT);                        // indbygget lysdiode
}
void loop() {
  digitalWrite(13, HIGH);                    // tænd lysdiode. Sæt spænding på pin 13 HØJ
  delay(1000);                               // vent 1 sekund = 1000 millisekunder
  digitalWrite(13, LOW);                     // sluk lysdiode. Sæt spænding på pin 13 LAV
  delay(1000);                               // vent 1 sek
}
```

Tillykke. du har nu skrevet dit første Arduino-program til at styre noget i omverden.

Måske ikke så ophidsende, men nu er du i gang.

Eksperiment 4.2

Prøv at ændre på den ene eller begge delay-tider og se at du faktisk kan kontrollere blink-frekvensen. Sæt for eksempel delay til 100 (dvs 100ms=0,1s).

Prøv at slutte et 9-volt batteri til Arduino. Nu kan du fjerne USB-kablet til computeren og overbevise dig om, at Arduino kan selv. Det er ikke din computer der styrer blinket.

Programmere Arduino

Et program indeholder tre dele

1. definitioner
2. setup
3. loop

I **definitions-delen** defineres konstanter og variable.

Vi kan skrive

```
int led = 13; // Det svarer til kommandoen led:=13 i et CAS-program.
```

int betyder heltal (integer)

Hvis der ikke er variable at definere, kan definitions-delen undværes.

Setup-delen udføres kun én gang. Her defineres bl.a. konstanter

Loop-delen gentages igen og igen til en evt. slutbetingelse er opfyldt eller til Arduino slukkes eller resettes.

Nøgleordet *void* betyder at programmet eller funktionen ikke forventes at give nogen information tilbage.

De krøllede parenteser { } afgrænser det der skal gøres.

Grundstruktur i program:

```
// Definitioner
// Her kan du definere konstanter og variable
//

void setup() {
  // Her skriver du setup-kode, der kun afvikles
  én gang
}

void loop() {
  // Her skriver du hovedprogrammet der gentages
}
```

Læs (meget) mere på www.arduino.cc Begynd under punktet Learning.

Arduinos ind- og udgange



Arduinos kommunikation med omverden sker gennem stikforbindelserne (*pins*), der ses øverst og nederst på billedet.

Øverst findes pins til digitale ind- og udgange. Man kan selv bestemme om den enkelte pin skal være indgang eller udgang.

I programmet i Eksperiment 2.1 bestemte vi at pin 13 skal være udgang med kommandoen `pinMode(13, OUTPUT)`

Når pin 13 sættes høj med kommandoen

```
digitalWrite(13, HIGH)
```

sættes en spænding på 5 V på pin 13.

Når vi taler om at vi måler en elektrisk spænding i et digitalt system, mener vi i virkeligheden at vi måler en spændingsforskel i forhold til et fastlagt nulpunkt. Dette nulpunkt betegnes GND (ground, jordforbindelse) og kan findes flere steder på Arduino. Det svarer til minus-polen på et batteri.

Find på billedet, eller på din egen Arduino, 3 punkter med spændingen GND

Et digitalt signal kan antage netop 2 værdier af spændingen, nemlig 0 V og 5 V.

Underforstået, spændingsforskel i forhold til GND.

Andre pins har en fast spænding, 5V eller 3,3V

Find på billedet, eller på din egen Arduino, et punkt med spændingen 5V.

Når vi i praksis skal have fat i signalet på en pin, bruger vi tynde farvede kabler (ofte kaldet jumpers eller duPont kabler), der stikkes ned i pinnen. Den anden ende af kablet kan forbindes til en prøveplade (*breadboard*), hvor vi kan eksperimentere.



Vi kan også forbinde kablet til et voltmeter eller andet apparat.

Det anbefales kraftigt at holde en "farvejustits", så alle kabler, der er forbundet til GND har samme farve, f.eks. sort, og alle kabler forbundet til 5V har samme farve, for eks. rød.

Eksperiment 4.3

- a) Tilslut to kabler til et voltmeter med krokodillenæb. Det sorte kabel til 0 på voltmeteret (kaldes også COM, GND eller -) det røde kabel til V (kaldes også +).
- b) Sæt det sorte kabel til GND på Arduino og det røde kabel på 5V
- c) Nu skulle voltmeteret gerne vise spændingsforskellen 5,0V.
- d) Forbind derefter det røde kabel til pin 13, mens blinkprogrammet fra Eksperiment 2.1 kører.
- e) Nu skulle voltmeteret gerne skifte mellem 0 V og 5,0V.
Den røde ledning kan nu bruges som en "probe", der måler spændingen på en ønsket pin.
- f) Kontroller at spændingen på pin mærket 3,3V er hvad den giver sig ud for.
- g) Hvad er spændingen på en pin mærket GND ?

Eksperiment 4.4. Lysdiode på breadbord

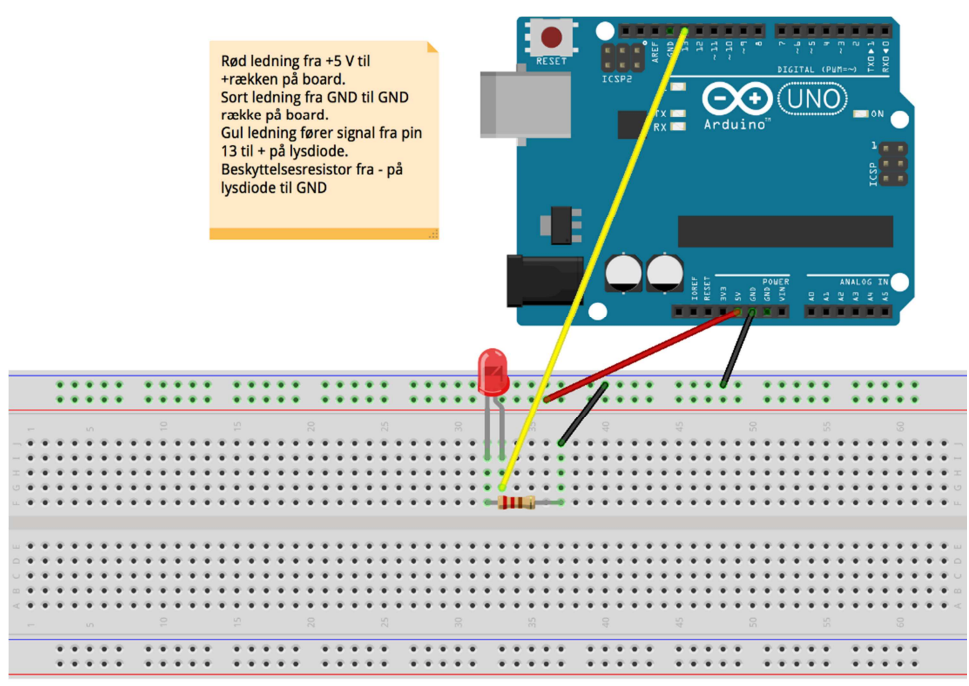
Vi går frem i små skridt. Næste skridt er at få en lysdiode til at lyse.

Som bekendt leder en diode kun strøm i den ene retning.

Det længste ben på en lysdiode er + og det korte er - .

- a) Opbyg et kredsløb, der ser ud som på tegningen.

Rød ledning fra +5 V til +rækken på board.
 Sort ledning fra GND til GND række på board.
 Gul ledning fører signal fra pin 13 til + på lysdiode.
 Beskyttelsesresistor fra - på lysdiode til GND



fritzing

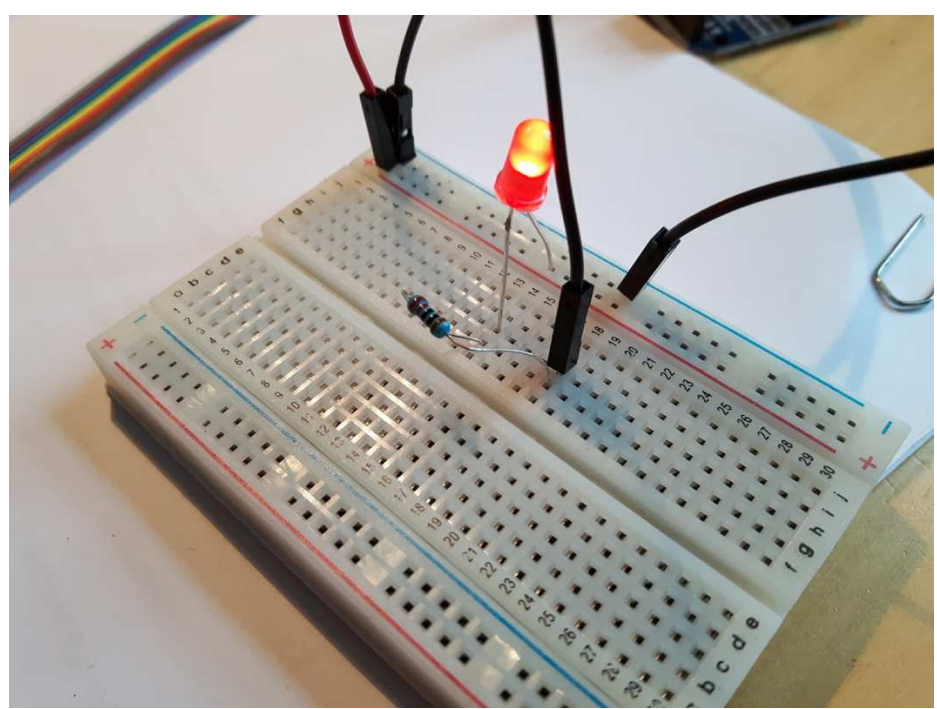
Bemærk den meget vigtige beskyttelsesresistor, som her er 220Ω

Det anvendte breadboard er bygget således at pins øverst og nederst er forbundet (markeret med rød eller blå strek).

Symbolerne + og - tolker vi som henholdsvis 5V og 0 V (GND).

Længere inde mod midten af breadboard er forbindelserne derimod lodret, således at fem pins mærket a, b, c, d og e er forbundet.

Tilsvarende er pins mærket f, g, h, i og j forbundet



Vi kan referere til en bestemt pin med et tal og et bogstav lige som i et regneark.

På billedet er lysdiodens - ben forbundet til pin h14 mens + benet er på 5V

Mellem g14 og g18 er indsat en 220Ω resistor. h18 er kabelforbundet til 0 V.

Med opstilling som på billedet vil dioden lyse konstant.

Måling med Arduino

Nederst til højre på billedet af Arduino ses 6 pins mærket ANALOG IN.

Det er som navnet antyder analoge indgange, hvor spændinger mellem 0 og 5 V kan måles.

Spændingen kan for eksempel komme fra et elektronisk termometer eller en anden sensor.

Spændingen konverteres til et tal mellem 0 og 1023.

Processen kaldes Analog til digital konvertering (Analog-Digital Converting ADC).

Her omsættes en spænding, der kan variere kontinuert mellem 0 og 5 V til ét ud af 1024 tal. For eksempel omsættes spændingen 2,5 V til tallet 512.

Det nytter selvfølgelig ikke meget at måle, hvis vi ikke kan se resultaterne.

For at se resultaterne, skal de sendes til computeren via USB-forbindelsen.

I næste eksperiment arbejdes med et lille program, der måler spændingen på pin A0 og sender det til computeren.

Opgave 4.5

Skitser en opstilling, så spændinger vises med lysdioder. Jo højere spænding, desto flere dioder skal lyse.

Eksperiment 4.6. Analog måling

Formål: Vi vil måle en spænding med Arduino

Indtast og kørs måleprogrammet.

Forbind pin A0 til GND, 5V og 3,3V og forklar resultatet.

Mål nogle andre spændinger.

Husk at forbinde minus på spændingskilden til GND på Arduino

```
int count=0;           // nulstil tæller

void setup() {

  Serial.begin(9600);  // sæt op til seriel kommunikation via USB
}
```

```

void loop() {

  Serial.print(count);           // send tællerens visning
  Serial.print(" ");           // mellemrum
  Serial.println(analogRead(A0)); // send tal målt på analog indgang 0:
  delay(1000);                 // vent 1 sekund mellem hver måling
  count = count + 1;           // læg 1 til tælleren
}

```

I Arduino-vinduet klikkes på Serial Monitor øverst til højre. Nu vises en skærm med måleresultater.

Eksperiment 4.7. Forbedret analog måling

Formål: Vi vil gerne have vist spændingen i volt:

```

int count=0;                    // nulstil tæller
float U;                        // Definer U som decimaltal

void setup() {
  Serial.begin(9600);           // start seriel kommunikation:
}

void loop() {
  // send the value of analog input 0:
  U=5.00*analogRead(A0)/1023;   // Omregn fra heltal til spændingsværdi
  Serial.print(count);
  Serial.print("\t");           // Tabulering
  Serial.print(analogRead(A0)); // Skriv den "rå" værdi læst af A0
  Serial.print("\t");
  Serial.print(U,3);            // Skriv spænding med 3 decimaler
  Serial.println();             // Linjeskift

  delay(1000);                 // 1 sekund mellem hver måling
  count = count + 1;           // Forøg tæller med 1
}

```

Indtast og kød måleprogrammet.

Forbind pin A0 til GND, 5V og 3,3V og forklar resultatet.

Mål nogle andre spændinger.

Husk at forbinde minus på spændingskilden til GND på Arduino

Eksempel 4.8 Bygge en alarm

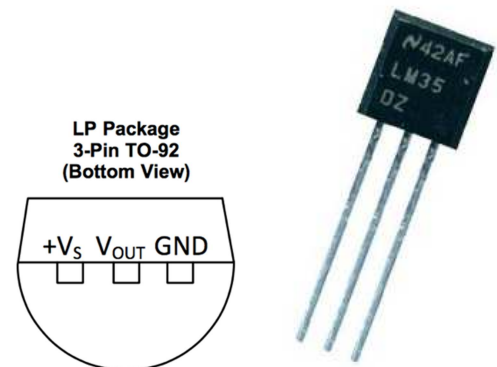
Vi vil bygge et apparat, der reagerer på en ændring i omgivelserne.

Ændringen i omgivelserne kan være en temperatur eller en vandhøjde, der overstiger en kritisk værdi.

Reaktionen kan være at en rød alarmlampe tændes.

I dette eksempel måles temperaturen med en komponent ved navn LM35.

LM35 tilsluttes 5V til +Vs og afgiver da en spænding, Vout, der er tilnærmelsesvis proportional med temperaturen. Leverandøren oplyser at Vout vokser med $10\text{mV}/^\circ\text{C}$. Tegningen viser LM35 fra neden. Det midterste ben er Vout.



Herunder ses et program, der udfører det ønskede.

Bemærk kommentarerne.

```
int Critical=50;           //Den kritiske værdi findes ved at prøve sig lidt frem

void setup() {
  pinMode(2,OUTPUT);      // Pin 2 defineres til output. Skal styre lysdiode
}

void loop() {
  if (analogRead(A0) > Critical) { // Læs signal fra sensor på pin A0
                                // Hvis (IF) signal større end kritisk værdi aktiver alarm
    digitalWrite(2,HIGH);      // dvs. sæt pin 2 HIGH og tænd lysdiode
  }                            // slut på IF- kommando
  else {
    digitalWrite(2,LOW);      // Udføres hvis betingelse i IF ikke er opfyldt,
  }                            // dvs hvis temperatur er ok, så sluk lysdiode
}
```


5. Belysnings-styring

Formål: Bygge en controller der tænder lyset, når det bliver for mørkt.

Brug en lysfølsom resistor (LDR) i serie med fast resistor (spændingsde-
ler)



Forberedelse:

Kortlæg i grove træk LDR's lysrespons. Hvis lysstyrken vokser, hvad sker da med R_{LDR} ?

Tips: Mål spændingsforskelle over LDR med analog indgang på Arduino. F.eks A0.

Vælg en "tænd-værdi" for A0. Når værdien overskrides, skal lys tændes.

Vælg en "sluk-værdi" for A0. Når værdien overskrides, skal lys slukkes.

Lyset kan i første omgang være en lysdiode (husk beskyttelsesresistor)

I næste version kan det være en kraftigere lampe styret af et relæ.

6. Vandhøjde-alarm

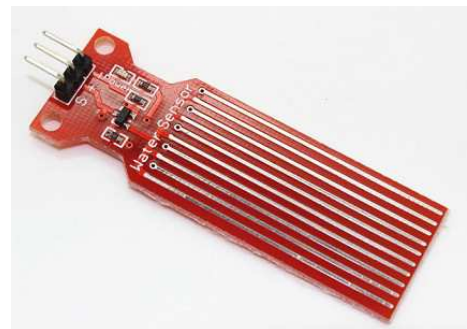
Formål: bygge en alarm, der lyser rødt når vandhøjden bliver for høj eller for lav.

Brug Water Sensor

Forberedelse:

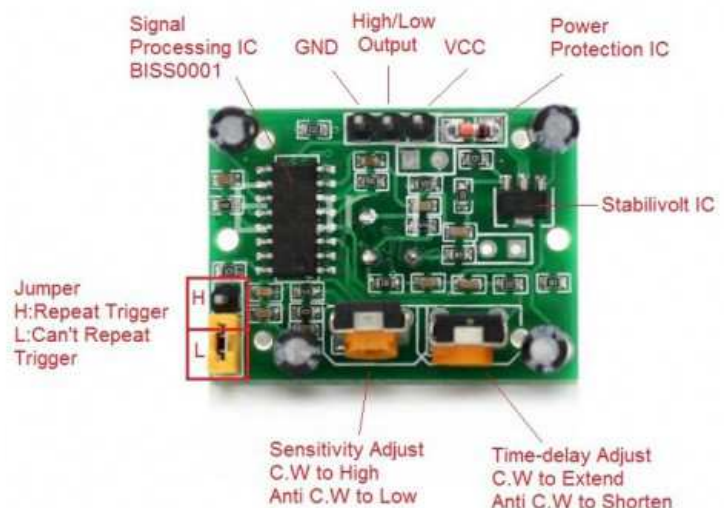
Kortlæg i grove træk sensorens respons på vandhøjde ved at anbringe den i et bæger med vand. - til GND, + til 5V. S er udgangssignalet. Mål i første omgang med voltmeter

Tip: Signalet S føres til analog indgang, f.eks. A0 på Arduino. Ved et valgt niveau tændes lysdiode



7. Bevægelses-sensor

Formål: Bygge en controller, der tænder lys eller alarm, når bevægelses-sensor aktiveres.



Fotos af komponenterne er fundet på www.arduinootech.dk

8. Pumpe

På sygehuse får patienterne ofte tilført væske eller medicin gennem en slange.

Det er vigtigt at kontrollere væsketilførslen, derfor kan væsken pumpes ind i patienten med en lille pumpe. Vi skal her skitsere en model af situationen med en pumpe, der styres af Arduino.

Du skal bruge:

- Peristaltisk doseringspumpe
- Relæ, f.eks. Songle SRD-05VDC-SL-C
- Spændingskilde 6V. F.eks. laboratoriestrømforsyning.
- Arduino
- Diverse ledninger og krokodillenæb



I programeksemplet sættes pumpen til at køre i 2 sekunder (2000 millisekunder). Herefter pause i 2 sekunder, pumpe i 2 sek osv.

```
void setup()
{
  pinMode(10,OUTPUT); // pin 10 sættes til output
}

void loop() {
  digitalWrite(10,HIGH); // Sæt pin 10 høj. Start pumpe
  delay(2000);           // Vent 2000 ms
  digitalWrite(10,LOW); // Sæt pin 10 lav. Sluk pumpe
  delay(2000);           // Vent 2000 ms
}
```

Der er imidlertid det problem, at pumpen kræver 6V og en strømstyrke på ca. 0,1 A. Det er mere end Arduino kan levere.

Vi løser problemet med et relæ, der kan tænde og slukke for strømmen til pumpen.

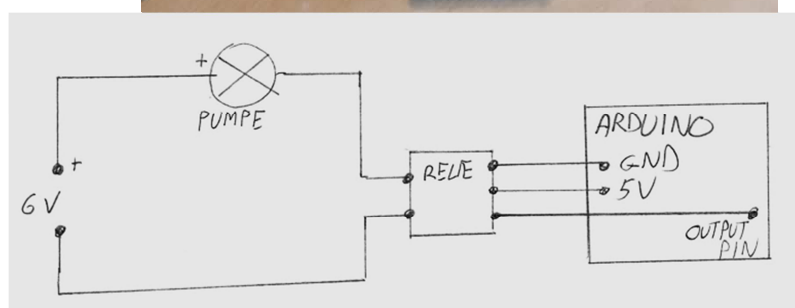


Højre side forbindes til Arduino

Sort: **GND**

Rød: **5 V**

Måling og styring. Sensorer og Arduino



Gul: **Signal** til pin 10

Venstre side

Modstand mellem **blå** og **gul** reguleres af spænding på Signal

Når Signal er HØJ er modstanden 0

Når signal er LAV er modstanden meget stor, dvs. mindst 1,8MΩ

Hvid kan sættes på GND, men det er ikke nødvendigt

Hvis pumpen skal bruges til noget, skal den kalibreres. Dvs. man skal vide hvor meget væske, der kommer igennem pr. tid.

Tilrettelæg selv et projekt hvor I kalibrerer pumpen. Lav eventuelt programmet, så man kan bestille en bestemt væskemængde, f.eks. 10 mL. Når væsken er leveret, skal programmet give besked ved at en lampe lyser eller ved anden form for signal.

Hvis I har kemi, kan I bygge en pumpe til titreringsforsøg.

Større Arduino-eksperimenter

Arduino-eksperiment 1. Blinkende dioder og løbelys.

Hvad lærer du her

Vi lærer om grundlæggende arduino programmering, om digitalt output og om opbygningen af prøvepladen.

Hvad skal du bruge

En computer med arduino programmet, et usb-kabel til arduinoen, en prøveplade, nogle dioder og nogle ledninger.

Mit første arduinoprogram.

Vores allerførste program til arduinoen får en diode på selve arduinoen til at blinke. Tilslut computeren til arduinoen med usb-kablet, tænd arduino-programmet og indtast følgende:

```
int led = 13; // her giver vi variabelen led størrelsen 13. Det svarer til at
skrive led:=13 i dit CAS-program

// Den følgende programstump afvikles kun en gang, når programmet uploades til
arduinoen
void setup()
{
    pinMode(led, OUTPUT); // klargør den digitale pin 13 til output.
}

//loop programstumpen kører igen og igen, når programmet er uploaded til arduinoen
void loop()
{
    digitalWrite(led, HIGH); // tænd dioden på pin 13, ved at skrive
                             // HIGH til den
    delay(1000); // vent i 1000 millisekunder
    digitalWrite(led, LOW); // sluk for dioden ved at skrive
                             //LOW til den
    delay(1000); // vent i et sekund
}
```

Læg mærke til, at programmet er delt op i tre dele:

- En del, hvor forskellige konstanter og variable defineres
- En programstump, void setup(), som afvikles en gang ved upload

- En programstump, void loop(), som afvikles igen og igen

Læg også mærke til, at alle programlinjer afsluttes med et semikolon ; og at man med // kan tilføje tekst til programmet. Det er god skik, at man fortæller om hensigten med de forskellige kommandoer i sit program.

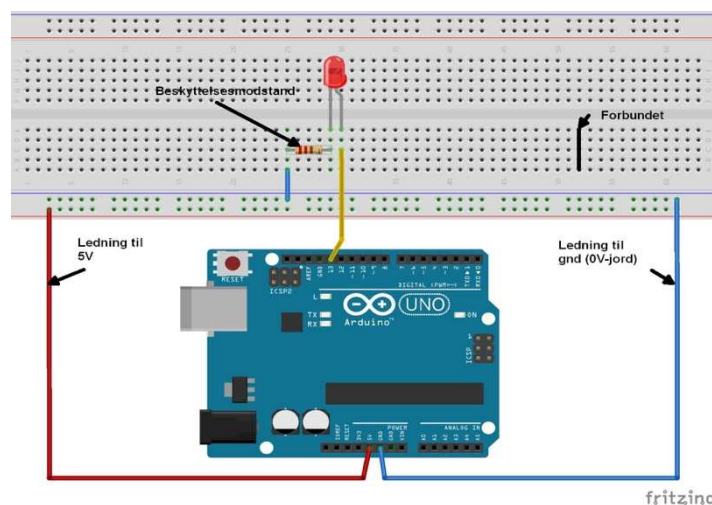
Upload nu programmet til din arduino. Hvis det ikke virker, kan det skyldes, at computeren forsøger at uploade programmet til en arduino, der ikke er der. Det kan man undersøge, ved at klikke på >Værktøjer >Seriel Port og evt. ændre på vælge en anden port (Det vil føre for vidt, at beskrive begrebet port her) Virker det, har du lavet dit første program til arduino, tillykke!

Øvelse

Prøv at ændre i dit program, så hvert blink kun varer et halvt sekund.

Brug af prøveplade

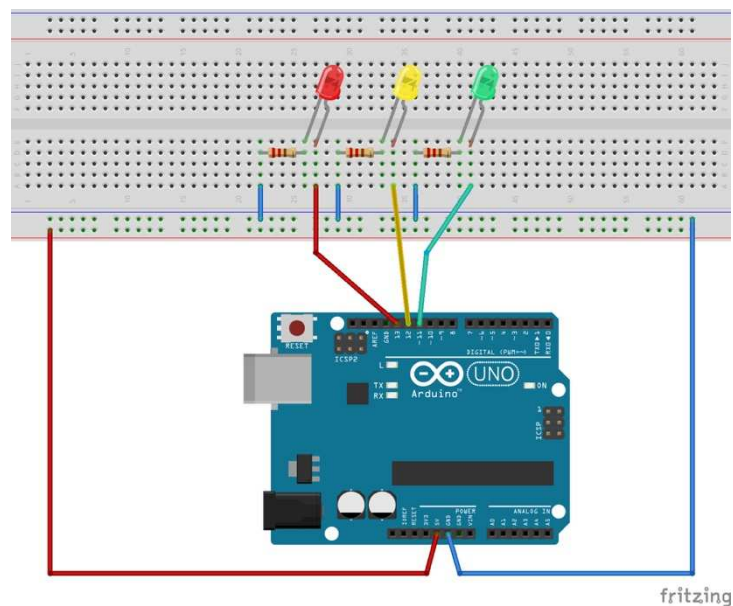
Vi vil bruge ovenstående program sammen med en eksterne diode anbragt på en prøveplade. En prøveplade består af en masse huller, der er indbyrdes elektrisk forbundet. I de to øverste og de to nederste rækker er hullerne forbundet vandret, mens de øvrige er forbundet lodret 5 og 5. Vi anbringer en diode i en prøveplade og forbinder den til den digitale pin 13 og jord (0V), kaldet gnd (=ground) på arduinoen. Jord kan sidestilles med minuspolen på et batteri, og alle spændinger er spændingsforskelle i forhold til gnd.



Prøv at upload dit program igen og se, om du har kontakt til dioden. Hvis det ikke virker, kan det skyldes, at dioden vender forkert.

Løbelys

For at gøre eksperimentet lidt mere avanceret kan vi lave et løbelys med tre lysdioder. Hvis du har mulighed for det, kan du vælge en grøn, en gul og en rød diode. Forbind ledningerne som vist på figuren. Programmet skal skrives en lille smule om, og der er indført en konstant, der hedder vent. Den bestemmer, hvor lang tid dioderne er tændt, og hvor lang tid,



der går mellem, at der er lys i en diode.

```
int led1 = 13;           // her giver vi variabelen led1 størrelsen 13. Det svarer
                        //til at skrive led:=13 i dit CAS-program

int led2 = 12;
int led3 = 11;
int vent=250;
// Den følgende programstump afvikles kun en gang, når programmet uploades til
//arduinoen
void setup()
{
  pinMode(led1, OUTPUT); // klargør den digitale pin 13 til output.
  pinMode(led2, OUTPUT);
  pinMode(led3, OUTPUT);
}

//loop programstumpen kører igen og igen, når programmet er uploadet til ardui-
//noen
void loop()
{
  digitalWrite(led1, HIGH); // tænd dioden på pin 13, ved at skrive HIGH til den
  delay(vent);              // vent i "vent" millisekunder, hvilket er det
                          //samme som et kvart sekund.

  digitalWrite(led1, LOW); // sluk for dioden ved at skrive LOW til den
  delay(vent);             // vent i "vent" milisekunder

  digitalWrite(led2, HIGH);
  delay(vent);
  digitalWrite(led2, LOW);
  delay(vent);
}
```

```
digitalWrite(led3, HIGH);  
delay(vent);  
digitalWrite(led3, LOW);  
delay(vent);  
}
```

Øvelse

Den viste programstump får lyset til at løbe fra rød til grøn og så starte forfra. Prøv at få lyset til at løbe frem og tilbage.

Øvelse

De tre farver på dioderne er de samme som på et trafiklys. Prøv at lave et program, der får dioderne til at skifte som et trafiklys.

Arduino-eksperiment 2. Ringeklokke, fodgængersignal og bevægelsesføler.

Hvad lærer du her

Vi lærer om digitale og analoge indgange og afprøver en buzzer, som er en lille piezoelektrisk højttaler.

Hvad skal du bruge

En computer med arduino programmet, et usb-kabel til arduinoen, en prøveplade, en trykknop, en buzzer, en bevægelsesføler, lysdioder), beskyttelsesmodstand 220 ohm og nogle lysdioder.

En ringeklokke.

Vi vil først lave en ringeklokke. Her skal vi for første gang bruge en af de digitale pins til input.

Som output vil vi bruge en buzzer, der er en lille højttaler, der virker ved hjælp af en lille piezoelektrisk krystal.

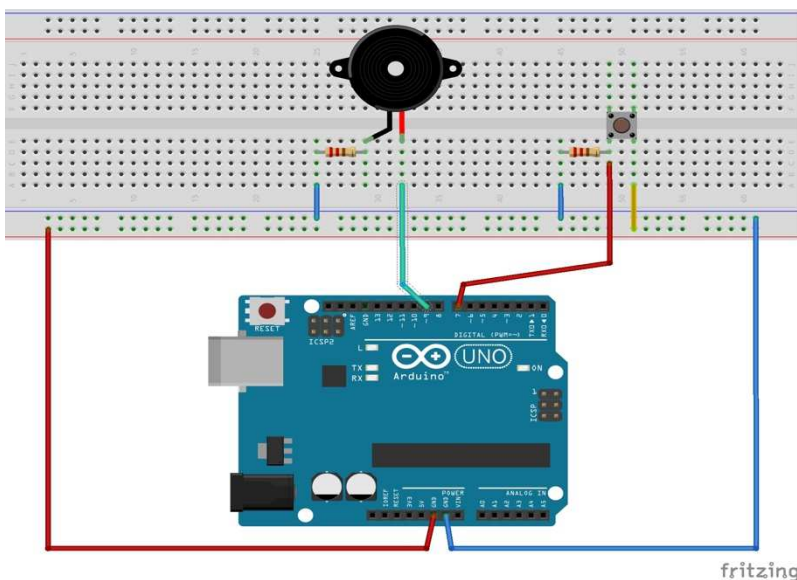
Vi skal have lavet et kredsløb, så et tryk på knappen, gør en af tilslutningerne høje.

Det kunne man forestille sig, at man kunne lave ved blot at sætte den ene side af trykknappen til høj (5V) og den anden side til pin 7, som vi vil bruge som input.

Det virker bare ikke, for når vi slipper knappen, er der ikke en mekanisme til at gøre vores tilslutning lav igen.

Vi leder derfor den samme side som tilslutningen til pin 7 ned til jord gennem en modstand.

Når vi slipper knappen vil der derfor være lav spænding i den side, som tilslutningen er.



```
//konstanter defineres:
const int knapPin = 7;           // trykknappen sluttes til pin 7
const int buzzerPin = 9;        // buzzeren sættes til pin 9

// variable defineres:
int knap = 0;                   // variable til at gemme knapstatus i

void setup()
{
  pinMode(buzzerPin, OUTPUT); // klargør buzzerPin til output:
  pinMode(knapPin, INPUT);    // klargør knapPin til input:
}
void loop()
{
```

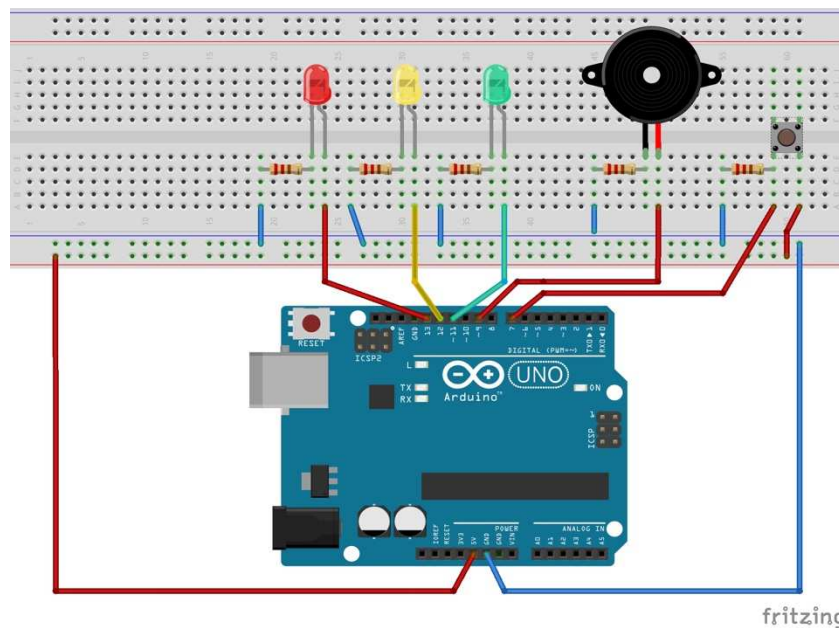
```

knap = digitalRead(knapPin); // læser knapstatus:
if (knap == HIGH)
{
  // hvis knappen er nedtrykket, så::
  tone(buzzerPin, 1000); // Send 1000 Hz signal til buzzeren
}
else
{
  noTone(buzzerPin);      // sluk buzzeren
}
}

```

Et fodgængerlys med lyd.

Et lidt mere komplekst eksempel på en anvendelse er opbygningen af et fodgængerlys. Nogle steder fungerer det sådan, at man kan trykke på en knap, og efter et stykke tid får man grønt lys som fodgænger. Det vil vi prøve at efterligne med arduinoen, nogle lysdioder og buzzeren til at fortælle, om du kan gå over. Der skal trækkes lidt flere ledninger og monteres lidt flere komponenter end før, og så skal man lige prøve sig frem, hvilken vej dioderne skal vende, det kan være svært at se, hvis dioderne er lidt medtagede. Først et billede at opstillingen:



fritzing

Som det ses, er grøn, gul og rød diode sluttet til de digitale porte 11,12 og 13. De kan både virke som input og output, og skal i programmet klargøres til den aktuelle anvendelse.

Knappen er tilsluttet port 7 og buzzeren port 13.

Programmeringen af dette er lidt mere krævende end de forrige eksempler. Hvis man vil, kan man forsøge sig med en opstilling uden buzzer, det gør programmeringen lidt nemmere; men man kan jo også bare vælge at kopiere programmet direkte herfra.

```

//konstanter defineres:
const int knapPin = 7;           // trykknappen sluttet til pin 7
const int buzzerPin = 9;        // buzzeren sættes til pin 9
int led1 = 11;                  // her giver vi variabelen led1 størrelsen 11. Det
                                // svarer til at skrive led:=11 i dit CAS-program

int led2 = 12;
int led3 = 13;

int staa=1000;                  //når forgængereren står og venter, er der

```

```

//langsomme toner, her varer de 1 sekund
int gaa=250; //når forgængerer må gå er der hurtige toner, her
1/4 sekund
// variable defineres:
int knap = 0; // variable til at gemme knapstatus i

void setup()
{
  pinMode(buzzerPin, OUTPUT); // klargør buzzerPin til output:
  pinMode(led1,OUTPUT); // klargør led1 til output:
  pinMode(led2,OUTPUT); // klargør led2 til output:
  pinMode(led3,OUTPUT); // klargør led3 til output:
  pinMode(knapPin, INPUT); // klargør knapPin til input:
}

void loop()
{
  knap = digitalRead(knapPin); // læser knapstatus:
  if (knap == HIGH) // hvis knappen er nedtrykket, så start
  //grønt lys sekvensen
  {
    for (int i=0; i<5;i++) //Vent 5 sekunder på, at der bliver rødt for
    //bilerne ved at gentage staa tonen 5 gange.
    {
      tone(buzzerPin, 440); //Der sendes 440Hz til buzzeren
      delay(staa); //Der ventes staa millisekunder
      noTone(buzzerPin); //Buzzeren slukkes
      delay(staa); //Der ventes staa millisekunder
    }
    digitalWrite(led3,HIGH); //Tænd både gult og rødt lys
    digitalWrite(led2,HIGH);
    for (int i=0; i<3;i++) //Vent ca. 6 sekunder mens der er rødt og gult
    //lys ved at gentage staa tonen 3 gange
    {
      tone(buzzerPin, 440); //der skal stadig være langsomme toner som
      //advarsel

      delay(staa);
      noTone(buzzerPin);
      delay(staa);
    }
    for (int i = 0; i < 20; i++) //Der er grønnt lys i ca. 10 sekunder. Det
    //styres ved at gentage gaa tonen 20 gange
    {
      digitalWrite(led3,LOW); //Det røde og gule lys slukkes, og det
      //grønne tændes

      digitalWrite(led2,LOW);
      digitalWrite(led1,HIGH);
      tone(buzzerPin, 440); //Der udsendes hurtige signaler, som
      //fortæller, at du kan gå over fodgængerfeltet

      delay(gaa);
      noTone(buzzerPin);
      delay(gaa);
    }
    digitalWrite(led1,LOW); //Det grønne lys slukkes
    digitalWrite(led2,HIGH); //Der skiftes til gult lys
    delay(1000);
  }
  else
  {
    digitalWrite(led3,HIGH); //Når knappen ikke er trykket på, vises rødt
    //lys, og staa-tonen sendes til buzzeeren

    digitalWrite(led2,LOW);
    digitalWrite(led1,LOW);
  }
}

```

```
tone(buzzerPin, 440);  
delay(staa);  
noTone(buzzerPin);  
delay(staa);  
}  
}
```

Øvelse

Du kan prøve at skifte frekvens og ændre tiderne for gaa og staa.

Hvis du vil have det til at være grønt i længere tid, kan du ændre på antallet af gentagelser.

Øvelse

Det kan være irriterende, at man ikke ved, om der er trykket på knappen eller ej. Tilføj en lysdiode ekstra, der tænder, straks knappen er trykket på.

Øvelse

Man kan få lyset til at skifte automatisk, når man nærmer sig ved i stedet for knappen at bruge en bevægelses-sensor. Den venstre pin på billedet skal have jord, den højre 5V og signalet kommer fra midterste pin. Signalet er højt, når detektoren føler bevægelse. Se også side 28

Giver det mening med en sådan føler som fodgænger?

Giver det mening, at have en føler, når du kommer kørende i bil mod et lyskryds?



Arduino-eksperiment nr. 3. Afstandsmåler

Hvad lærer du her

Vi lærer at læse tal ud fra arduinoen via seriel kommunikation.

Hvad skal du bruge

- En computer med arduino programmet,
- et usb-kabel til arduinoen, en prøveplade,
- en ultralydsafstandsmåler HC-SR04
- samt evt. nogle lysdioder.

En afstandsmåler

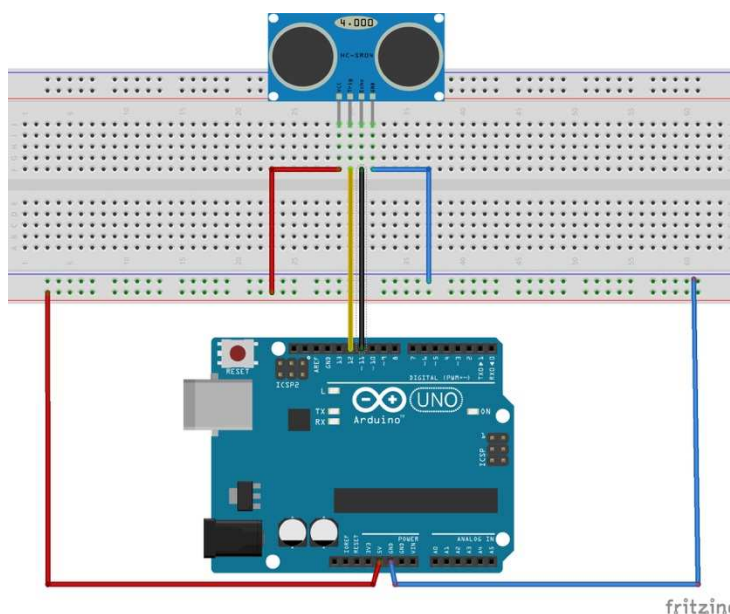
I de forrige eksperimenter har vi set, hvordan arduinoen kan give signaler med lysdioder og buzzeren.

Der er imidlertid mange situationer, hvor man får brug for at udlæse et tal.

Man kan forestille sig, at arduinoen skal bruges til spændingsmåler for spændinger op til 5V, som de analoge indgange A0-A5 kan håndtere, eller som i dette tilfælde, hvor arduinoen skal konvertere en tid til en værdi, som vi gerne vil kende.

I dette tilfælde vil vi bruge muligheden for at få arduinoen til serielt at sende værdier til arduino-programmet, som så har en mulighed for at vise hvilket tal, der bliver sendt via den serielle port.

Afstandsmåleren virker ved at sende en kort puls ultralyd ud og registrere, hvornår pulsen



kommer tilbage. Når afstanden skal beregnes, anvender man så en værdi for lydens fart i luft, som i eksemplet er sat til 342 m/s eller 0.0342 cm/ μ s.

Man skal huske på, at lyden både skal frem og tilbage, derfor divideres den beregnede afstand med 2.

```
//Definerer konstanter og variable
int trigPin = 12;
int echoPin = 11;
int tid=1000;           //Tiden mellem afstandsmålingerne
```

```

long forsinkelse;
float afstand;
float vlyd=0.0342;

void setup()
{
  Serial.begin (9600);           // Her klargøres den serielle port
  pinMode(trigPin, OUTPUT);     // triggerpin, den der sender signalet,
                                //klargøres til output
  pinMode(echoPin, INPUT);      // echopin, som modtager signalet,
                                //klargøres til input
}

void loop()
{
  digitalWrite(trigPin, LOW);   // Klargør triggeren
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH); // Sætter triggeren på HIGH i 10
                                //mikrosekunder

  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  forsinkelse = pulseIn(echoPin, HIGH); // Læser på echo-pin forsinkelsen i
                                         //mikrosekunder

  afstand= forsinkelse*vlyd/2; // Beregner afstanden i cm.
  Serial.print("Afstand: ");   // Printer afstanden til seriel-
                                //overvågning

  Serial.println(afstand);
  delay(tid);
}

```

Øvelse

I det foregående program er hastigheden af lyden indbygget som 342 m/s. Det er jo ikke sikkert, at lydets hastighed har lige netop denne værdi, der hvor I skal måle. Lav en lille ændring i programmet, så det er tiden, der bliver skrevet ud i stedet for afstanden og brug instrumentet til at bestemme hastigheden af ultralyden. Hold en plade af et eller andet op foran afstandsmåleren og noter sammenhørende værdier af tid (målt i mikrosekunder) og afstanden, som lyden skal bevæge sig, altså den dobbelte afstand fra måleren til pladen.

Øvelse

Med arduinoen og afstandsmålerne kan man altså måle afstande, og da man samtidig også har mulighed for at styre tiden mellem afstandsmålingerne kan man naturligvis bruge opstillingen til at måle hastighed. Det kræver jo blot, at man måler to afstande og den tid, der går mellem de to målinger.

Lav om på programmet, så det kan bruges til at måle hastighed, og check eventuel målingen med en anden form for afstandsmåling.

Øvelse

Prøv at finde ud af, hvor langt ud afstandsmåleren virker, og hvor stor en vinkel, den kan måle i.

Øvelse

Prøv at lave en "bak-sensor" som får en grøn diode til at lyse, når der er 50 cm til en mur, en gul, når der er 30 og en rød, når der er 15 cm til muren. Brug if-sætninger i programmet.