

Nils Kr. Rossing

Måling av lufttrykk og høyde med Arduino – ELEVARK



NTNU



Trondheim

Institutt for
fysikk

Skolelaboratoriet
for matematikk, naturfag
og teknologi

Februar 2020

Måling av lufttrykk og høyde med Arduino ELEVARK

Nils Kr. Rossing

Måling av lufttrykk og høyde med Arduino - ELEVARK

Trondheim 2020

Layout og redigering: Nils Kr. Rossing, Skolelaboratoriet ved NTNU

Trykk: NTNU Grafisk senter

Tekst og bilder: Nils Kr. Rossing, Skolelaboratoriet ved NTNU

Faglige spørsmål rettes til:

Skolelaboratoriet for matematikk, naturfag og teknologi

v/ Nils Kr. Rossing nils.rossing@ntnu.no

Skolelaboratoriet ved NTNU

Realfagbygget,

Høgskoleringen 5,

7491 Trondheim

Telefon: 73 55 11 43

<http://www.ntnu.no/skolelab/>

Rev 2.2 – 29.02.20



Innhold

1	Innledning	7
2	Montering og oppkobling for trykkmåling	7
2.1	Oppkobling	7
2.2	Kort innføring i bruk av Arduino editoren (IDE)	9
2.3	Programmering	12
2.4	Beregn høyden – legg inn formelen i programmet	17
2.5	Kalibrering	19
3	Oppdragene	21
3.1	Om du befinner deg på Bårdshaug i Orkanger:	21
3.1.1	Oppdrag 1 – Mål relative høyder inne på Bårdshaug konferansesenter	21
3.1.2	Oppdrag 2 – Kalibrering og finn absolutt høyde over havet	21
3.1.3	Oppdrag 4 – Mål lufttrykket i klasserommet	22
3.2	Om du befinner deg på Realfagbygget på Gløshaugen	23
3.2.1	Oppdrag 1 – Mål relative høyder i Realfagbygget	23
3.2.2	Oppdrag 2 – Kalibrering og absolutt høyde over havet	23
3.2.3	Oppdrag 3 – Mål høyden på et av høyhusene på Gløshaugen	24
3.2.4	Oppdrag 4 – Mål lufttrykket i klasserommet	24
4	Referanser	25

1 Innledning

I denne oppgaven skal dere bruke Arduino UNO til å lage et instrument som måler lufttrykket. På bakgrunn av lufttrykket skal dere beregne relativ høyde og etter kalibrering absolutt høyde over havet. Resultatene skal vises på et lite display.

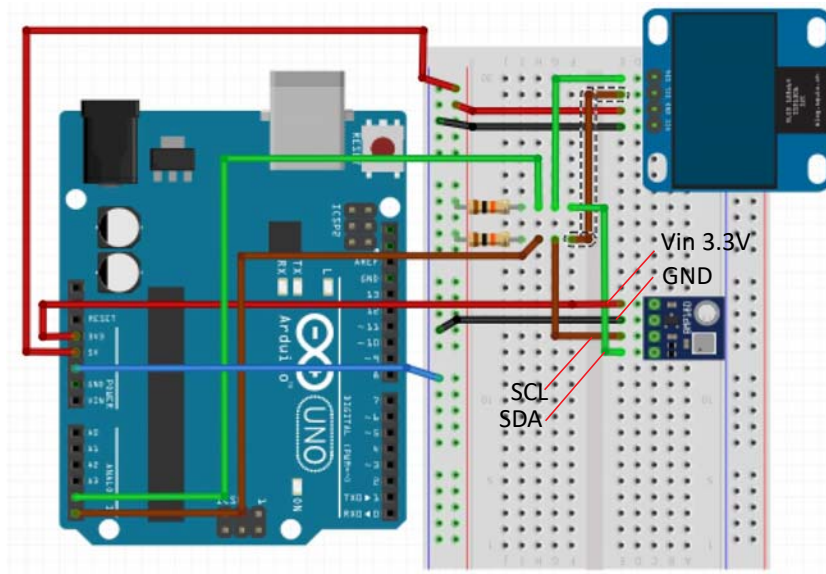
2 Montering og oppkobling for trykkmåling

I vårt måleoppsett har vi valgt å benytte en Arduino UNO og trykkmåleren/barometeret BMP180 som gir stor relativ nøyaktighet ved måling av lufttrykk. Vi bruker dessuten et SSD 1306 grafisk display og har valgt å benytte en batteriklemme for 9V batteri slik at vi kan ta med oss måleren uavhengig av PC'en.

2.1 Oppkobling

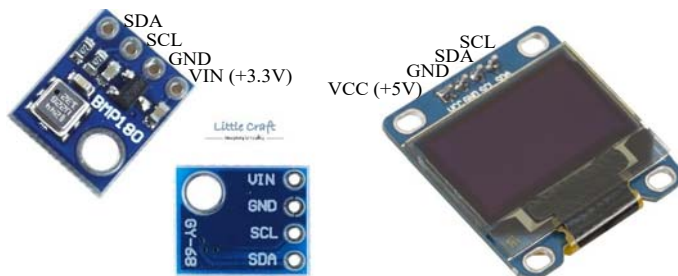
Figuren under viser oppkoblingen. Både BMP180 og SSD 1306 kommuniserer med Arduino'en via en I²C buss som er tilsluttet Arduino'en via de to analoge inngangene A4 (SDA - Data) og A5 (SCL - klokke). Kommunikasjonen skjer ved hjelp av to ledninger, en som overfører en klokkefrekvens og en som overfører data. For at denne skal fungere tilfredsstillende benyttes to motstander hver på 10 kΩ som er koblet til +5V.

OBS! Merk at displayet er koblet til +5V og GND, mens trykkmåleren **BMP180 er koblet til +3,3 V og GND**. Dette er særdeles viktig for ikke å ødelegge trykkmåleren.

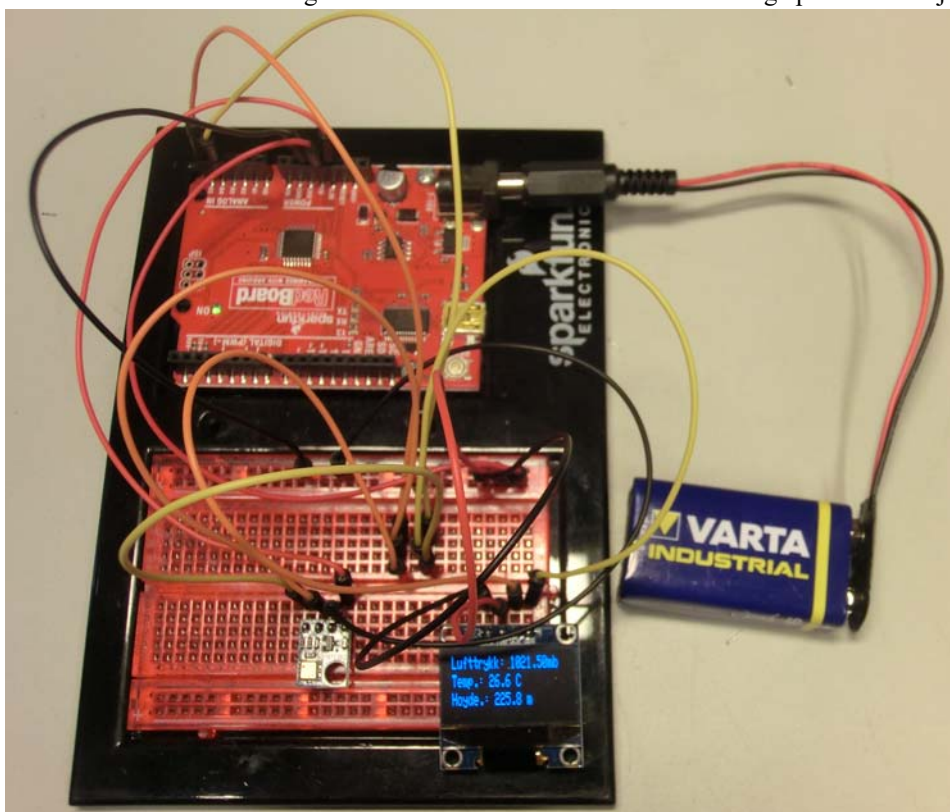


Merk at navnene på beina til BMP180 står på undersiden av kretsen.

Figuren under viser nærbilder av de to kretsene.



Oppkoblingen kan bli omtrent som på bildet under. Merk at man bruker kabler med de fargene man har. Det kan være vanskelig å være helt konsekvent mht til samme farge på samme linje.



Bruk overgangen fra batteriplugg til batterikontakt og et 9V batteri til å gjøre måleinstrumentet uavhengig av PC'en.



2.2 Kort innføring i bruk av Arduino editoren (IDE)

Dette er en meget kortfattet innføring i bruk av editoren til Arduino. Hopp over om dette er kjent.

1. Installasjon av programvaren

Det er normalt ikke nødvendig å installere en ny utgave dersom en gammel er installert i maskinen fra før. Eldre versjoner vil sannsynligvis fungere godt.

Programvaren hentes fra:

<https://www.arduino.cc/en/Main/Software>

Download the Arduino IDE



Velg: *Windows installer* (For Mac – *Mac OS X*)

Velg: *Just Download*

Velg: *Kjør*



Velg: *Følg prosedyren i installasjonen*

Velg: *Å installere drivere*

2. Start programmet og installer tre biblioteker

Klikk på ikonet til Arduino og start programmet.



Du vil forhåpentlig se følgende programvindu åpne seg etter en liten stund.



For å kunne kommunisere med displayet (SSD1306) og trykkmåleren (BMP180), må vi installere tre biblioteker. Disse bibliotekene er komprimert i zip-filer og kan lastes ned fra:

<https://www.ntnu.no/skolelab/bla-hefteserie>

under fanen: *Måling av lufttrykk, høyde og posisjon*

Adafruit_SSD1306-master.zip

Adafruit-GFX-Library-master.zip

BMP180_Breakout_Arduino_Library-master.zip

Lagre bibliotekene i nedlastingskatalogen eller et annet sted der du finner dem igjen.

Bibliotekene installeres på følgende måte:

Åpen Arduino-editoren



Velg: *Skisse*

Velg: *Inkluder bibliotek*

Velg: *Legg til .zip bibliotek* (øverst i nedtrekksmenyen)

Finn biblioteket i katalogen Nedlasting

Velg: *Open*

... og biblioteket installeres



Gjenta det samme for alle tre bibliotekene.

3. Start programmet og koble Arduino'en til en USB-kontakt

Se på menylinjen øverst til venstre.



Velg: *Verktøy*

Velg: *Kort*

Velg: *Arduino/Genuino UNO*

Velg: *Verktøy*

Velg: *Port*


Velg: Det høyeste COM nummeret, eller velg porten merket med Arduino/Genuino UNO


Det skal nå være opprettet kontakt mellom programeditoren (IDE) og Arduinokortet


4. Kort brukerveiledning for program-editoren

De viktigste kommandoene for å betjene bruken av editoren er oppsummert under:



 Sjekk at koden er fri for skrivefeil (syntaksfeil)

 Kompiler og send koden til kortet

 Åpne en ny skisse og start med blanke ark



Last opp en programskisse fra disken



Lagre programskisse på disken

2.3 Programmering

Vi har laget et forenklet program for måling av lufttrykk og temperatur og skriving til displayet. Dersom du legger inn programmet slik det er, vil du få opp lufttrykk og temperatur på displayet. Dette er det lufttrykket som sensoren leverer her og nå.

Programkoden kan lastes ned fra siden:

<https://www.ntnu.no/skolelab/bla-hefteserie>

Under fanen “*Måling av lufttrykk, høyde og posisjon med Arduino*”.

// Programmet er beregnet for å måle barometertrykk og temperatur ved hjelp av
// BMP180 og vise resultatet på et display av typen SSD1306.

```
// Inkludering av biblioteker
#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h> // Display
#include <SFE_BMP180.h>      // Trykkmåling

// Deklarasjon av klasser
Adafruit_SSD1306 display(4); // Deklarasjon av enheten "display" av klassen
                                Adafruit_SSD1306
SFE_BMP180 pressure;          // Deklarasjon av enheten "pressure" av klassen
                                SFE_BMP180

void setup()
{
    Serial.begin(9600);
    display.begin(SSD1306_SWITCHCAPVCC, 0x3C); // Initialiser display med adr.0x3C
    pressure.begin();
}

void loop()
{
    // Hent trykk- og høydeinformasjon
```



```
double P,T;                                // Deklarer variabler

P = getPressure();                          // Hent måleverdi av trykk
T = getTemperature();                      // Hent måleverdi av temperatur

// Deklarerer variable for beregning av høyden
float Hl=0;                                // Referansehøyde i meter
float Pl=1000.0; // Referansetrykk i mbar ved referansehøyde
float Tl=0.0; // Temperatur i grader C, husk at formelen krever grader Kelvin
float a=0.0065; // TMemperaturgradient i K/m
float R=287.06; // Den spesifikke gasskonstanten i J/kg K
float g0=9.81; // Tyngdeakselerasjonen i m/s2

// Legg inn referansetrykk, høyde og temperatur i formelen og beregn høyden

//   Beregning av høyden skrives inn her,
//   bruk parametrene som er deklart over

// Klargjør display for utskrift
display.clearDisplay();                    // Slett informasjon på display
display.setTextSize(1);                   // Sett størrelse på tekst
display.setTextColor(WHITE);              // Hvit tekst på sort bakgrunn

// Skriver ut lufttrykk med to desimaler
display.setCursor(0,0);                    // Plasser markør øverst til venstre
display.print("Lufttrykk: ");              // Skriv "Lufttrykk
display.print(P,2);                        // Skriv verdien til lufttrykk med to desimaler
display.println("mb");                     // Skriv benevning mbar (mb)

// Skriver ut temperatur med en desimal
display.setCursor(0,10);                   // Flytt markør til neste linje
display.print("Temp.: ");                  // Skriv "Temp.:", Temperatur
display.print(T,1);                        // Skriv den målte temperaturen i ...
display.println(" C");                     // ... grader C

// Skriv inn kode for å skrive ut høyden med en desimal på displayet
display.setCursor(0,20);                   // Flytt markør til neste linje

//   Utskrift av høyde til display skrives inn her
//   Bruk kodelinjene over som eksempler
```

```

    display.display();          // Overfør informasjonen til displayet og vis
    delay(500);                 // Vent i 500 msek.
}

double getPressure()
{
    char status;
    double T,P,p0,a;           // Deklarer lokale variable

    // For å foreta en trykkmåling så trengs temperaturen
    status = pressure.startTemperature(); // Forespørsel om temperaturmåling
    if (status != 0)             // Om forespørselen er vellykket, gå videre
    {
        delay(status);          // Vent til målingen er klar
        status = pressure.getTemperature(T); // Mål temperaturen
        if (status != 0)        // Om målingen er vellykket gå videre
        {
            status = pressure.startPressure(3); // Forespørsel om trykkmå-
ling                                     //
            Forespørsel om trykkmåling, (3) høyest presisjon
            if (status != 0)      // Om forespørselen er vellykket, gå videre
            {
                delay(status);    // Vent til målingen er klar
                status = pressure.getPressure(P,T); // Foreta trykk og temperaturmåling
                if (status != 0)
                {
                    return(P);
                }
                else Serial.println("Feil ved forespørsel av trykkmåling\n");
            }
            else Serial.println("Feil ved måling av trykk\n");
        }
        else Serial.println("Feil ved forespørsel av temperaturmåling\n");
    }
    else Serial.println("Feil ved måling av temperatur\n");
}

double getTemperature()
{
    char status;
    double T;

```



```
status = pressure.startTemperature(); // Forespørsel om temperaturmåling
if (status != 0)                      // Om forespørselen er vellykket, gå videre
{
    delay(status);                    // Vent til målingen er klar
    status = pressure.getTemperature(T); // Mål temperaturen
    if (status != 0)                  // Om målingen er vellykket gå videre
    {
        return(T);
    }
    else Serial.println("Feil ved forespørsel av temperaturmåling\n");
}
else Serial.println("Feil ved måling av temperatur\n");
}
```

Blokkdiagram for koden

Figuren til høyre gjengir et blokkdiagram av programmet slik at det skal være lettere å forstå koden.

Start

Programmet starter på nytt hver gang strømmen slås på eller man trykker på RESET-knappen på kortet. Den starter også på nytt når man slår på monitoren i program-editoren.

Biblioteker og deklarasjoner

De fleste av de kommandoene vi bruker når vi koder er funksjoner som kalles opp fra biblioteker. For å bruke display, trykksensor og kommunikasjon på linjene mellom kretsene (I²C), så kreves spesiallagde funksjoner, disse er samlet i biblioteker som må inkluderes i starten av programmet (`#include ...`).

Variabler som skal kunne brukes i alle funksjoner (globale variable) må deklarerer utenfor funksjonene. Disse deklarerer ofte her i starten.

Setup () – funksjonen

`Setup()` – funksjonen kjøres bare en gang, hver gang programmet starter. Her ligger initialisering som ikke trengs og gjentas mens programmet kjører.

Dette gjelder initialisering av kommunikasjon mellom kortet og monitoren i PC'en (`Serial.begin(9600);`).

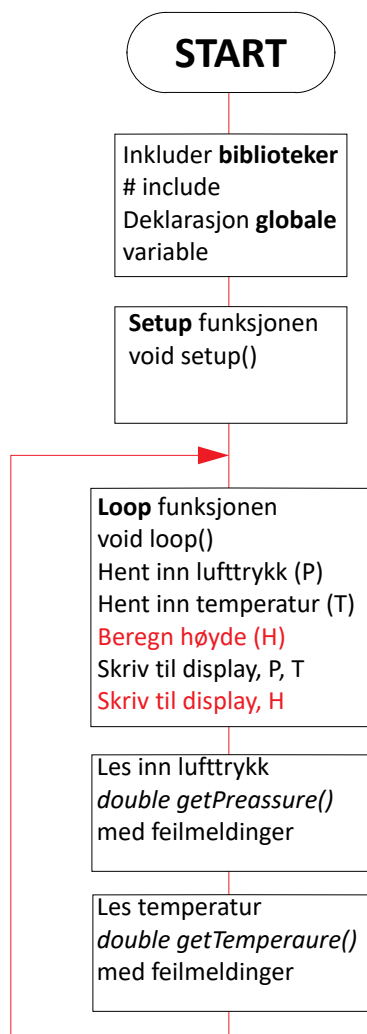
Videre gjelder det initialisering av displayet (`display.begin()`), og av trykkmåleren (`pressure.begin()`).

Loop () – funksjonen

Denne gjentar seg så lenge kortet har spenning. Når en kommer til slutten, så begynner programmet fra toppen av `loop()`-funksjonen igjen. For hver runde leses temperatur og trykk. Hvor ofte det gjøres målinger er hovedsakelig bestemt av hvor lang tid det tar å gå gjennom loopen. Kommandoen `delay(500);` på slutten av loopen legger inn en pause på 500 millisekunder. Det er hovedsakelig denne som bestemmer hvor ofte det tas målinger.

Avlesing av trykk og temperatur

Først leses trykk (`P = getPressure();`) og temperatur (`T = getTemperature();`). De avleste verdiene legges i variablene `P` og `T`.





Deklarasjon av variabler og konstanter

Dernest deklarerer en del lokale variabler (P_1 , H_1 , T_1)¹ og konstanter (a , R , g) som skal brukes når man legger inn formelen for beregning av høyden, H .

Beregning av høyden

I programskissen er det vist med en tekst hvor formelen for å beregne høyden kan legges inn.

Display resultater

De neste avsnittene inneholder kommandoer som skriver data til displayet. Dette er ulike kommandoer av typen, `display.print` og `display.println` o.l. Se forøvrig avsnittet Tips til programmering på side 18. Tanken er at når formelen for beregning av høyde legges inn så kan også den skrives ut på displayet.

double getPressure() – funksjon

Denne funksjonen leser av trykksensoren og sjekker om det oppnås kontakt med sensoren. Der-som det ikke oppnås kontakt, så skrives det ut en feilmelding på IDE-monitoren. For at funksjonen skal kunne beregne trykket må den også lese av temperaturen. Temperaturavlesningen er inkludert i funksjonen. Funksjonen returnerer verdien av lufttrykket som legges inn i variabelen P .

double getTemperature() – funksjon

Denne funksjonen leser av temperatursensoren som er inkludert i trykksensoren. Funksjonen returnerer verdien av temperaturen som legges inn i variabelen T .

2.4 Beregn høyden – legg inn formelen i programmet

Når vi har fått programmet til å oppføre seg som det skal, vil vi beregne høyden som funksjon av lufttrykket og temperaturen. Vi kan da bruke formelen²:

$$h = \frac{T_1}{a} \left(\left(\frac{p}{p_1} \right)^{\frac{aR}{g_0}} - 1 \right) + h_1 \quad (2.1)$$

Hvor:

h	Beregnet høyde i meter
p_1	Trykk i Pa ved referansehøyden h_1
h_1	Referansehøyden i meter
T_1	Temperatur ved referansehøyden h_1 i K(elvin)
p	Målt trykk i Pa

1. Ser at jeg har blandet store og små bokstaver når det gjelder programmet og formelen, men det er altså de samme parametrene.
2. Se lign. (4.2), side 28 for nærmere omtale.

a Temperaturgradient, foreslått verdi -0,0065 K/m
 g_0 Tyngdeakselerasjonen 9,81 m/s²
 R Den spesifikke gasskonstant 287,06 J/kg K

Legg formelen inn i programmet på det markerte stedet. Bruk variablene som er definert foran i programmet. Legg også inn noen kodelinjer som skriver høyden til displayet med riktig tekst og benevning.

Tips til programmeringen for å beregne høyden

Aritmetiske funksjoner:

+	Addisjon
-	Subtraksjon
*	Multiplikasjon
/	Divisjon
<i>pow (grunntall, eksponent);</i>	Eksponent
<i>double log(double x);</i>	Naturlig logaritme, definer argument og resultat som <i>double</i>

for-loop(){ }

Dersom man ønsker å midle verdier over flere målinger kan man bruke en for-loop:

```
for (int i=0; i <= 255; i++){ }           // Denne går 256 ganger i for-loopen.
                                           // Det er det som står mellom klamme-
                                           // parentesene som gjentas 255 ganger
```

Funksjoner for skriving til lokalt display

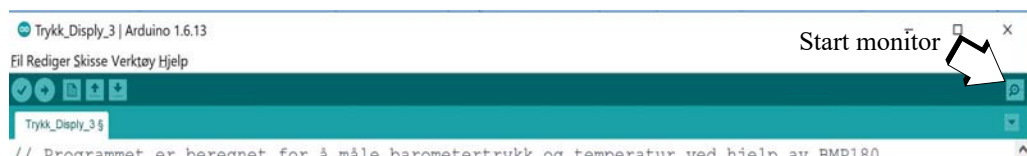
```
display.begin(SSD1306_SWITCHCAPVCC, 0x3C); // Initialiser display med adr.0x3C
display.clearDisplay();                     // Slett informasjon på display
display.setTextSize(1);                    // Sett størrelse på tekst
display.setTextColor(WHITE);               // Hvit tekst på sort bakgrunn
display.setCursor(0,10);                   // Flytt markør til neste linje
display.print("<tekst>");                   // Skriv ut tekst uten linjeskift
display.print(T,1);                        // Skriv den målte temperaturen m/en desimal
display.println("<tekst>");                 // Skriver ut tekst med linjeskift
display.display();                         // Overfør informasjonen til displayet og vis
```

Funksjoner for skriving til IDEs monitor

```
Serial.begin(9600); // Initialiser skriving til monitor m/9600baud
Serial.print("<tekst>");                     // Skriv ut tekst uten linjeskift
Serial.print(T,1);                          // Skriv den målte temperaturen m/en desimal
Serial.println("<tekst>");                   // Skriver ut tekst med linjeskift
```



Monitoren åpnes ved å trykke på:



2.5 Kalibrering

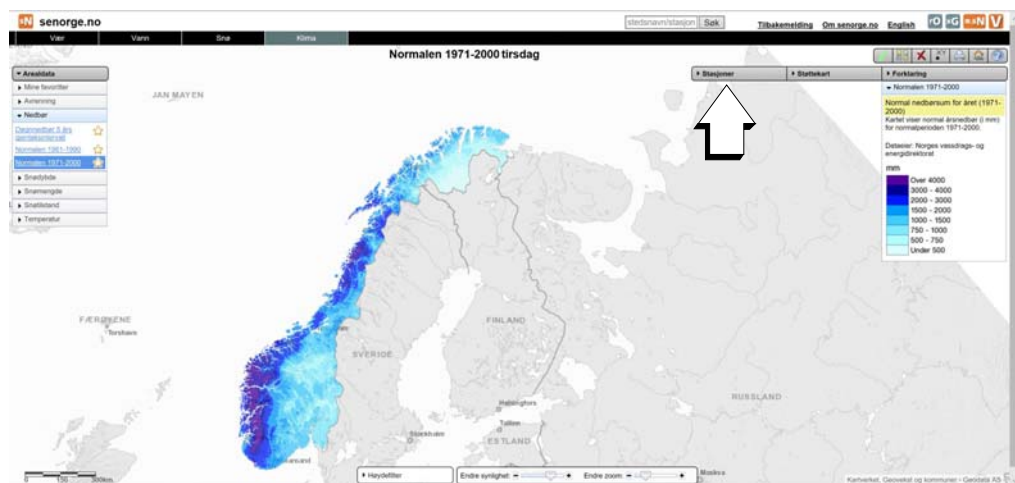
Dersom vi ønsker å måle absolutt høyde må vi kalibrere målingene. Dette kan vi gjøre ved å finne trykket (p_f) hos en nærliggende målestasjon med en kjent høyde (h_f). Her kan vi anta at målingene angir lufttrykket på stedet ved den angitte høyden og ikke er omregnet til trykket ved havnivå.

Undersøk målestasjoner i nærområdet. Et godt sted å begynne er:

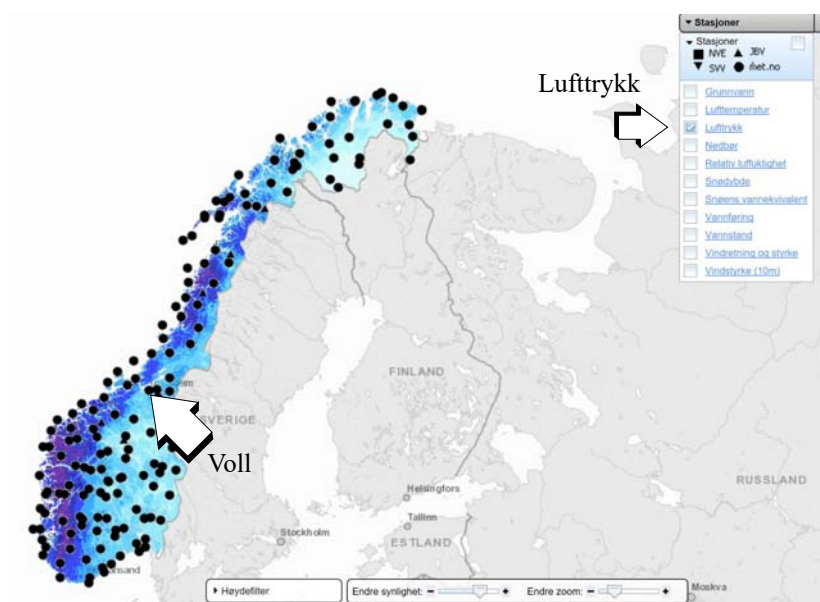
<http://www.xgeo.no/index.html?p=klima>

Offisiell portal for målestn. i Norge

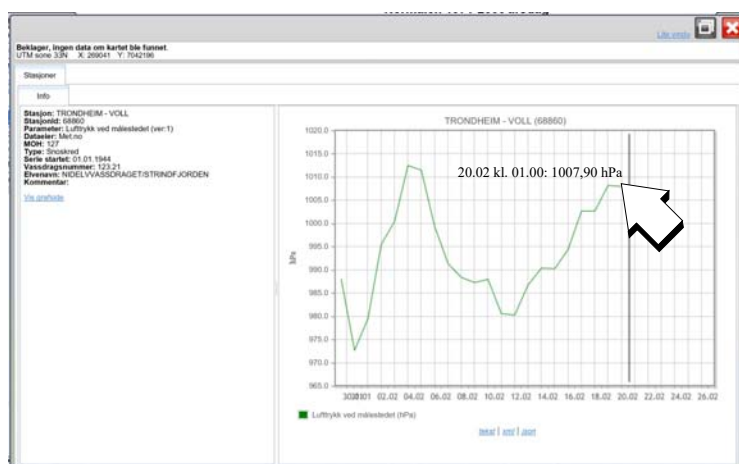
Dette er en åpen portal på Internett, som viser daglig oppdaterte kart over snø-, vær- og vannforhold og klima for Norge – og mye mer.



Velg så stasjoner og kryss av for “Lufttrykk” og velg en målestasjon nær der du befinner deg. Voll forsøksgård er et alternativ som er ganske nært Gløshaugen. Men Værnes eller Ørlandet flystasjon er også mulig å bruke. En kan også sammenligne målinger fra flere nærliggende stasjoner for å se om det er store lokale variasjoner.



Etter å ha valgt målestasjon, får man opp en graf som viser lufttrykket siste døgn. Ulempen er imidlertid at målingene kun oppdateres en gang i døgnet. Det kan derfor være store avvik mellom avlest verdi og virkelig verdi for lufttrykket på det ønsket tidspunkt. En kan se på hvordan trykket har varierte den seneste tiden og anslå om trykket endres fort eller langsomt.



Man har da funnet siste registrerte lufttrykk og stasjonens høyde over havet. Disse verdiene settes så inn i programmet som $P1$ og $H1$ (se side 13). Dersom det er en stund siden siste oppdatering eller at nærmeste målestasjon er et stykke unna, så må mann regne med at det kan være avvik.

En mer nøyaktig måte å kalibrere utstyret på er å gå inn i et kart over stedet der man befinner seg. Finne den eksakte høyden og legge inn denne ($H1$) for det målte trykket i øyeblikket ($P1 = p$).

3 Oppdragene

3.1 Om du befinner deg på Bårdshaug i Orkanger:

3.1.1 Oppdrag 1 – Mål relative høyder inne på Bårdshaug konferansesenter

Ta utgangspunkt i den bygningen dere befinner dere i og mål relativ høyde for hver etasje, fra kjeller til toppetasje så lavt og høyt dere kommer. Sammenlign målingene med hva andre har fått.



3.1.2 Oppdrag 2 – Kalibrering og finn absolutt høyde over havet

Forsøk å kalibrer instrumentet etter beste evne ut fra data hentet inn fra ulike vær-stasjoner funnet på nettet (se side 19). Sammenlign målinger utført av andre som befinner seg på samme sted og se på avvik. Hvordan kan vi ev. forklare avvik i målinger gjort av absolutt høyde?

Kartet under er et utsnitt fra området rundt Bårdshaug Herregård.



Kartet er hentet fra:

<https://norgeskart.no/>

#!?project=seeiendom&layers=1002,1013,1014,1015&zoom=16&lat=7029266.67&lon=241906.09&panel=searchOptionsPanel&sok=B%C3%A5rdshaug&markerLat=7029421.748763504&markerLon=241850.8050601559

Gå inn i kartdatabasen og finn høyden over havet utenfor bygningen.

Hvilke avvik registrerer dere mellom høyden hentet fra kartdata og den dere måler etter kalibrering?

3.1.3 Oppdrag 4 – Mål lufttrykket i klasserommet

I dette oppdraget er oppgaven å registrere lufttrykket i klasserommet når man åpner og lukker døra. For at dette skal være mulig må man gjøre følgende:

1. Sample trykket så raskt som mulig (responstiden er maks. 7,5 msek)
2. Legge inn kommandoer som skriver data til monitoren
3. Sørg for å legge ut data slik at det lett lar seg kopiere inn i f.eks. Excel eller at annet analyseprogram. Skill gjerne verdiene med komma eller semikolon. Lag ny linje for hver måling.

3.2 Om du befinner deg på Realfagbygget på Gløshaugen

3.2.1 Oppdrag 1 – Mål relative høyder i Realfagbygget

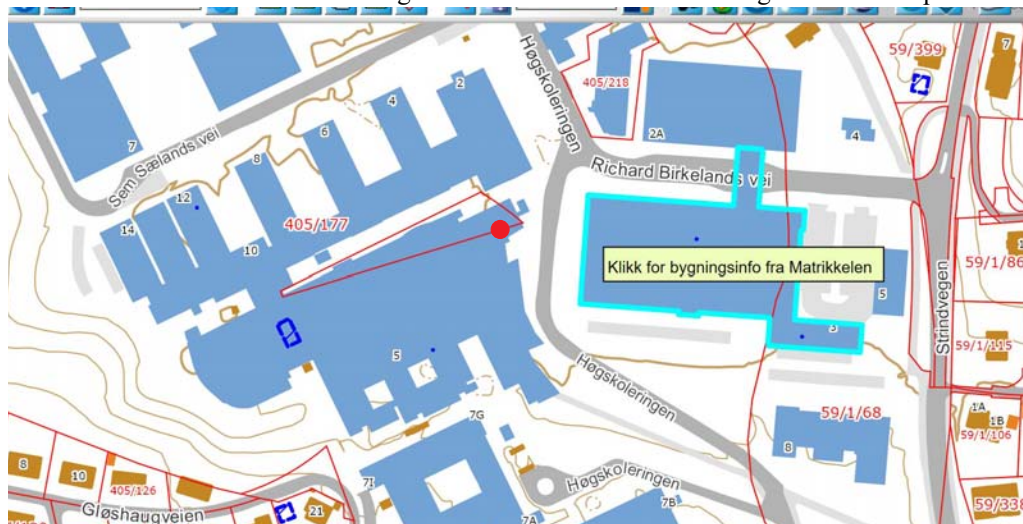
Ta utgangspunkt i den bygningen dere befinner dere i og mål relativ høyde for hver etasje, fra kjeller til toppetasje så lavt og høyt dere kommer. Sammenlign målingene med hva andre har fått. Bygget har 9 etasjer.



3.2.2 Oppdrag 2 – Kalibrering og absolutt høyde over havet

Forsøk å kalibrer instrumentet etter beste evne ut fra data hentet inn fra ulike vær-stasjoner funnet på nettet (se side 30). Sammenlign målinger utført av andre som befinner seg på samme sted og se på avvik. Hvordan kan vi ev. forklare avvik i absolutte målinger?

Kartet under er et utsnitt fra Gløshaugen. Skolelaboratoriet befinner seg nær den røde prikken.



Kartet er hentet fra:

<https://kart5.nois.no/trondheim/Content/Main.asp?layout=trondheim&time=1550437879&vwr=asv>

Gå inn i kartdatabasen og finn høyden over havet utenfor bygningen. Ev. på laboratoriet dere befinner dere. Hvilke avvik registrerer dere i forhold til høyden fra kartdata og den dere måler etter kalibrering?

3.2.3 Oppdrag 3 – Mål høyden på et av høyhusene på Gløshaugen

Det høyeste punktet her på Gløshaugen er sannsynligvis på toppen av en av høyblokkene. Ta med dere måleinstrumentet å finn ut følgende:

- Mål absolutt høyde ved foten av høyblokka.
- Ta heisen opp i 13. etasje og finn den absolutte høyden der. Estimer høyden fra der dere klarer å komme til å måle og til toppen.
- Finn høyden av blokka fra foten til toppen.
- Sammenlign med andre som har gjort det samme.



3.2.4 Oppdrag 4 – Mål lufttrykket i klasserommet

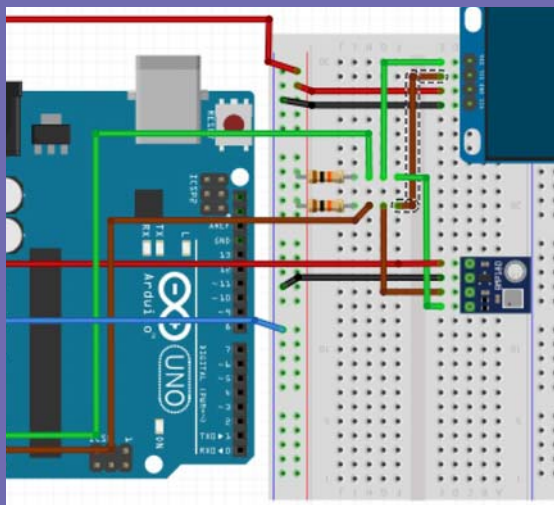
I dette oppdraget er oppgaven å registrere lufttrykket i klasserommet når man åpner og lukker døra. For at dette skal være mulig må man gjøre følgende:

1. Sample trykket så raskt som mulig (responstiden er maks. 7,5 msek)
2. Legge inn kommandoer som skriver data til monitoren
3. Sørg for å legge ut data slik at det lett lar seg kopiere inn i f.eks. Excel eller at annet analyseprogram. Skill gjerne verdiene med komma eller semikolon. Lag ny linje for hver måling.



4 Referanser

- [1] Gunnar Stette, *Romtekologi - Del av faget Teknologi og Forskningslære (CanSat)*, NTNU juli 2011.
- [2] Nils Kr. Rossing, *ROV - med trykk og temperaturmåling*, Rev 3.1 - 20.11.17
Heftet kan lestes ned i sin helhet fra: <https://www.ntnu.no/skolelab/bla-hefteserie>



Opplegget ble laget i forbindelse med en nettverksamling for ToF-lærere i Østlandsregionen mars 2018. I mars 2019 ble dette elevheftet laget. I tillegg finnes det et revidert lærerhefte som utdyper team som sensorteknologi og kalibrering.

Elevheftet beskriver hvordan man kan koble opp og måle lufttrykk ved hjelp av sensoren BMP180 og en Arduino UNO og vise resultatet på et lite display. Videre hvordan man kalibrerer og gjør høydeberegninger på bakgrunn av lufttrykk.

Nils Kr. Rossing

Dosent ved Skolelaboratoriet

E-post: nils.rossing@ntnu.no

Prosjektleder ved Vitensenteret

E-post: nkr@vitensenteret.com



Trondheim

Institutt for
fysikk

Skolelaboratoriet
for matematikk, naturfag
og teknologi

Tlf. 73 55 11 43

<https://www.ntnu.no/skolelab>