

DIY Turbidimeter

Amund Midtgard Raniseth

11. september 2019

Sammendrag

Hvis du noen gang har sett en grumsete væske og lurt på hvor grumsete den egentlig er, så har du kommet til rett sted. Her lager jeg, og gir dere muligheten til å reproduusere, et turbidimeter som måler hvor skittent vannet egentlig er.

I løpet av gjennomføringen vil dere lære om Arduino, lysdioder, generell kobling, lodding, kalibrering med mer.

Innhold

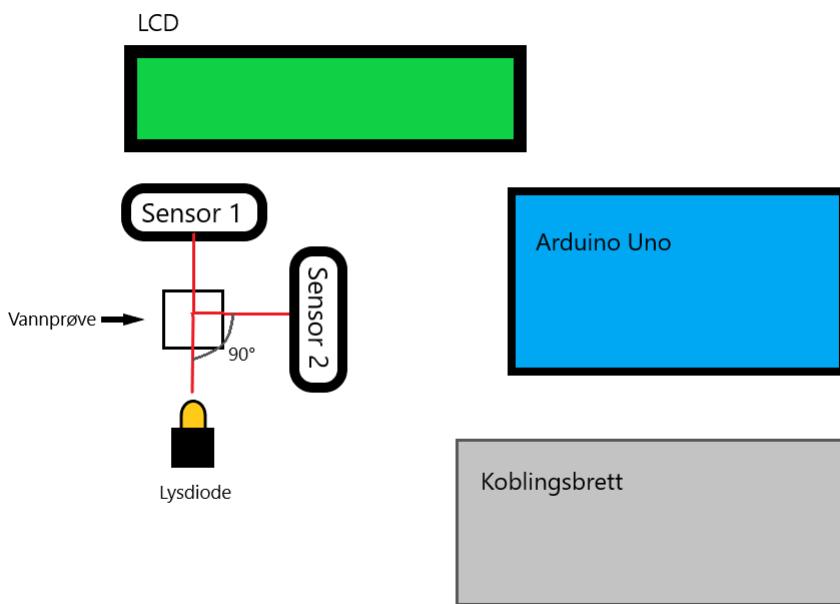
1	Introduksjon	2
2	Materialiste	3
3	Hardware	3
3.1	Lysdioden	3
3.2	Printing av boks og prøveholder	5
3.3	Oppkobling av lyssensorer (TSL2561)	6
3.4	Oppkobling av LCD-Display	7
3.5	Oppkobling av LED	8
3.6	Ferdiggjøring av Oppkobling	9
4	Software	11
4.1	Komme i gang med koding	11
4.2	Koden	14
5	Kalibrering	14
5.1	Lage kalibreringsløsninger	14
5.2	Måle data	14
5.3	Regresjon	15
5.4	Modifisere koden	17
6	Appendiks	18
6.1	Lysdiode	18
6.1.1	Motstand på felles elektrode	19

1 Introduksjon

Å måle antall partikler som er i vann er kritisk for å sikre kvalitet av blandt annet drikkevann. Et kommersielt turbidimeter koster ofte veldig mye, og ikke alle har bruk for et turbidimeter til den grad at de kan legge ut pengene som kreves. Målet med denne guiden er at du skal kunne produsere et turbidimeter som kan gjøre målinger av vannkvalitet uten at det koster skjorta. Det er også et poeng å lære litt om teknologier vi bruker underveis.

Det finnes flere typer turbidimeter, men felles for alle er at de sender en form for lys inn i en vannprøve, og bruker enten spredningen som lyset har i prøven, eller absorpsjonen som et mål på hvor mange partikler som er i vannet. Europerisk standard er ISO 7027 og bruker spredningen av infrarødt lys til å sette en verdi. Denne verdien har enheten FNU (Formazin Nephelometric Units).

I figur 1 ser du oversikten over hvordan turbidimeteret vil se ut. Det du vil se fra utsiden av boksen er kun LCD'en, og hullet til å plassere vannprøven. Av skissen ser man også at vi skal måle både absorpsjon(sensor 1) og spredning(sensor 2), for å mer nøyaktig kunne estimere prøvens turbiditet. Sensorene er koblet til koblingsbrettet. Det samme gjelder lysdioden, mens noen av LCD-ledningene går rett til Arduinoen.



Figur 1: Enkel skisse som gir en oversikt over turbidimeteret.

2 Materialliste

Før vi starter trenger vi alle delene. I tabellen under (tabell 1) finner du en liste over det du trenger.

Antall	Navn
2	Adafruit TSL2561 Sensor
1	RGB LED
1	Arduino Uno
1	16x2 LCD med I ² C grensesnitt
3	220Ω motstander
1	halv størrelse breadboard
230 gram	PLA filament
18	male-female jumper cables
6	male-male jumper cables

Tabell 1: Materialliste

Hvis du har en hvit LED fra før så trenger du bare én motstand, se appendiks 6.1 for mer informasjon.

Vi kommer også til å få bruk for følgende verktøy:

Verktøy	
Loddebolt	3D-Printer
loddetinn	Drill
Nebbtang	Borr
Limpistol	

Tabell 2: Verktøyliste

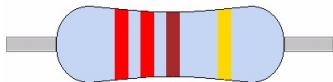
3 Hardware

I denne seksjonen skal vi 3D-printe, sette sammen og koble alt.

3.1 Lysdioden

Ettersom vi bare skal bruke dioden som en lyskilde og ikke har bruk for å skru den av eller på, vil vi koble den slik at den er på så lenge instrumentet er på. Spenningen som dioden behøver er litt lavere enn 5v (som arduinoen bruker), så vi må sette ned spenningen. Vi gjør dette litt enklere for å spare tid, men hvis du vil se den litt mer avanserte metoden så sjekk appendiks 6.1. Vi gjør følgende:

- Finn frem tre 220Ω motstander. De ser slik ut:



Figur 2: 220Ω Motstand

- Vi må identifisere hvilke av beina på lysdioden som er til fargene. Det er som regel slik at det lengste beinet på lysdioden er enten felles anode eller felles katode (enten + eller -). For å finne ut om du har felles anode eller katode kan du bare lodd opp motstandende, og så prøve å sette på spenningen begge veier, og se hvilken som gjør at dioden lyser. Se neste steg.



Figur 3: Lysdiode med enten felles katode eller anode. Bokstavene representerer den andre elektroden til de forskjellige fargene.

- Lodd hver av de tre motstandende til hver av fargene sin positive del, og lodd den andre enden av motstanden sammen, og til en ledning vi kan sette i breadboardet. Det er viktig at elektrodene til de forskjellige fargene ikke berører hverandre. Bruk gjerne litt elektrikertape i mellom. Loddingen kan se slik ut:



Figur 4: Motstander loddet til RGB-LED'en.

- **NB:** Vi loddet på alle tre beina for å få litt loddeerfaring. Man kan selvfølgelig også bruke formelen

$$\frac{1}{R} = \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3} + \dots$$

Og finne ut hvilken motstand man kan koble på den felles elektroden. Se appendix 6.1.1.

3.2 Printing av boks og prøveholder

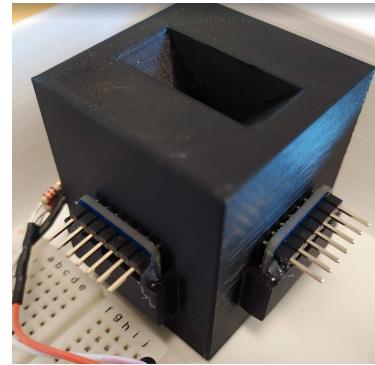
- Begynn med å laste ned .STL filene. Last disse inn i sliceren til din 3D-printer. Jeg brukte 15% infill og 0.15mm layer height samt alt med supports. Overfør deretter eksportert Gcode til 3D-printeren og skriv ut. Det kan være lurt å printe alt i svart for å unngå at for mye lys kommer gjennom veggene og forstyrrer lysstrålen som skal gå gjennom prøven. 3D-printene vil se slik ut:



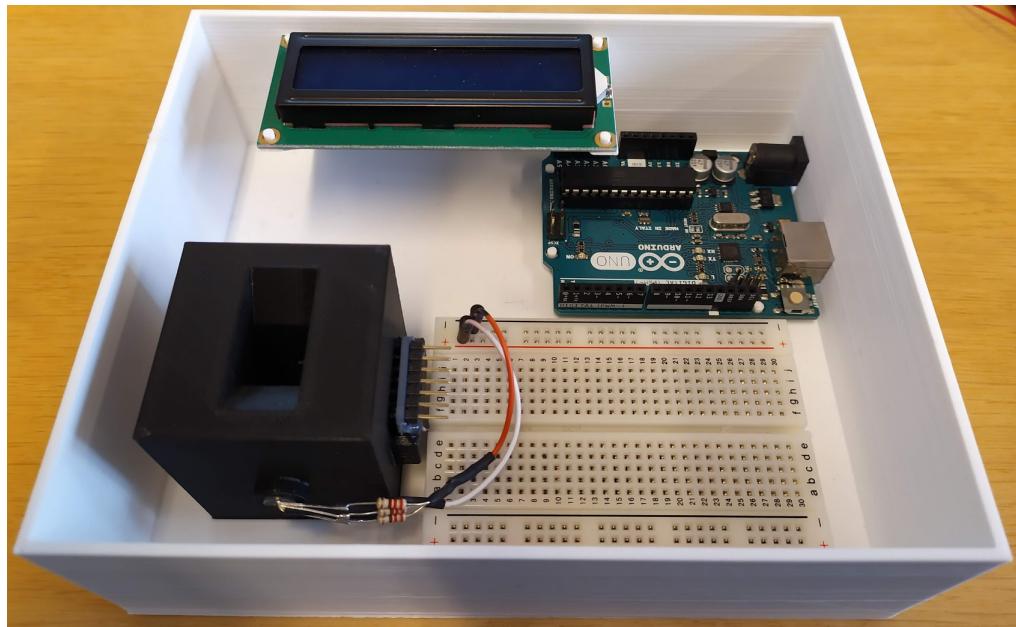
Figur 5: 3D-printet boks, Sensor-assembly og skyggekappe.

- Fjern all support på 3D-printene. Dette kan være vanskelig på de små delene, men med en nebbtang går det meste.

- Plasser lyssensorene og lysdioden i sensor-holderen og plasser hele assemblyen i sin plass i plast-boksen. Hvis dioden eller sensorene sitter litt løst, kan det være lurt å bruke limpistolen litt for å få det til å sitte.



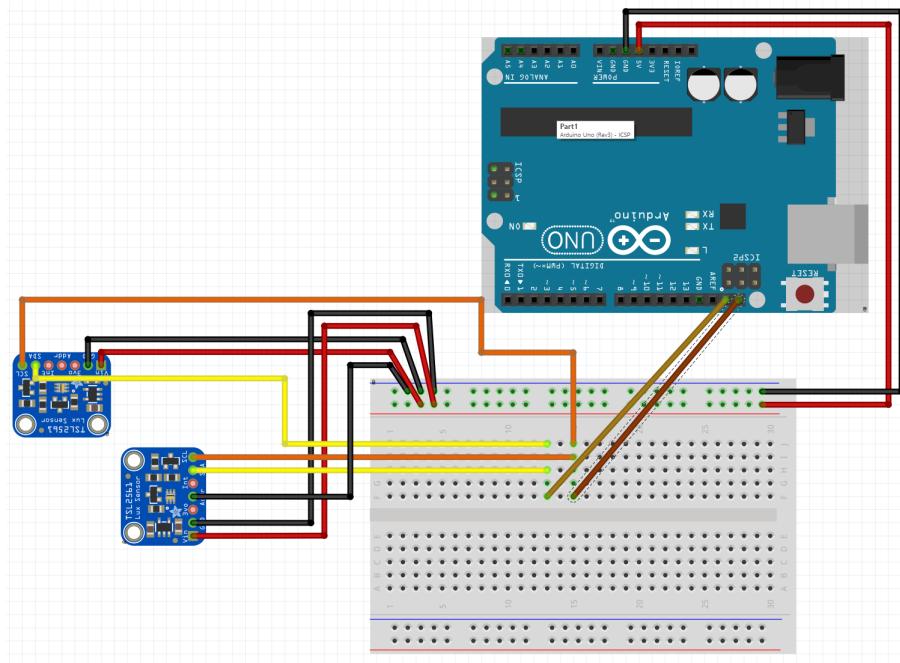
- Plasser så Arduino, Breadboard, og LCD på sin plass i boksen som i figur 6. Skyggekappen skal ikke på enda.



Figur 6: Alle delene i boksen

3.3 Oppkobling av lyssensorer (TSL2561)

- Se figur 7 for forslag til oppkobling.
- Koble 5v og jord fra Arduinoen til breadboardet.
- Koble SCL fra begge sensorene sammen og deretter koble de til Arduinoens SCL (Illustrert på tegningen)

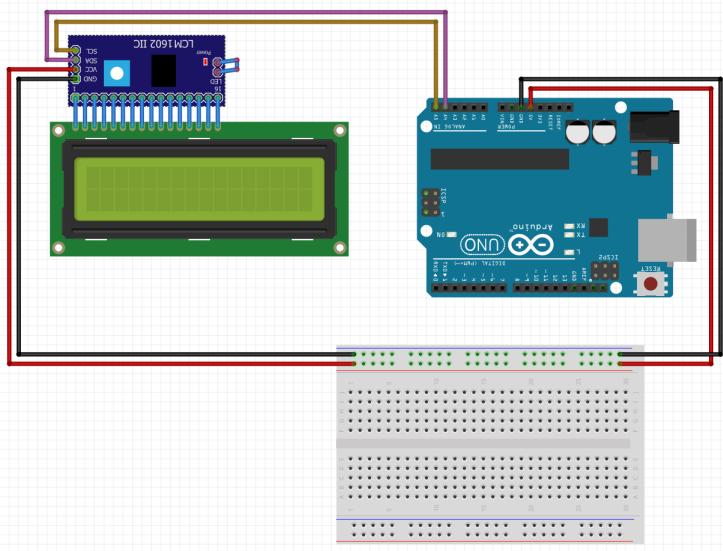


Figur 7: Koblingsskjema som viser hvordan lysensorene skal kobles opp

- Koble SDA fra begge sensorene sammen og deretter koble de til Arduino-ens SDA (Illustrert på tegningen).
- Koble begge sensorenes Vin til 5v, og GND til jord.
- Den sensoren som står på 90° vinkel fra lysdioden skal også ha en kobling fra Addr til jord. Dette gjør at vi senere, i software, kan skille mellom sensorene.

3.4 Oppkobling av LCD-Display

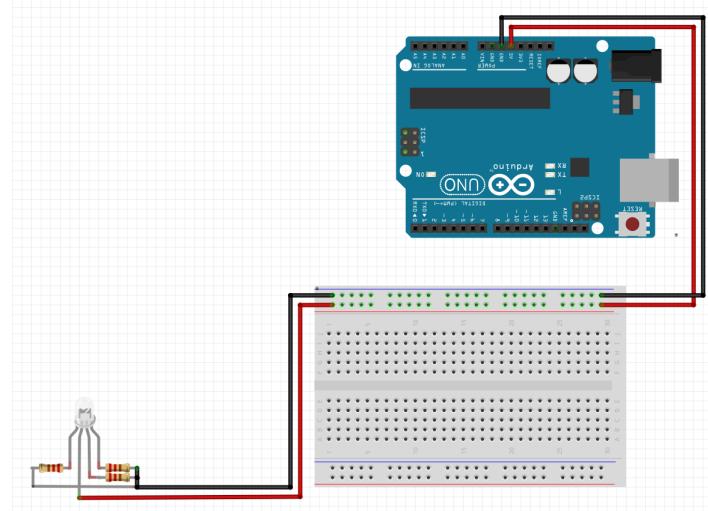
- Se figur 8 for forslag til oppkobling.
- Koble I^2C -grensesnittets GND til jord, og VCC til 5v.
- Koble SDA til arduino pin A4 og SCL til pin A5.



Figur 8: Koblingsskjema som viser hvordan LCD'en skal kobles opp

3.5 Oppkobling av LED

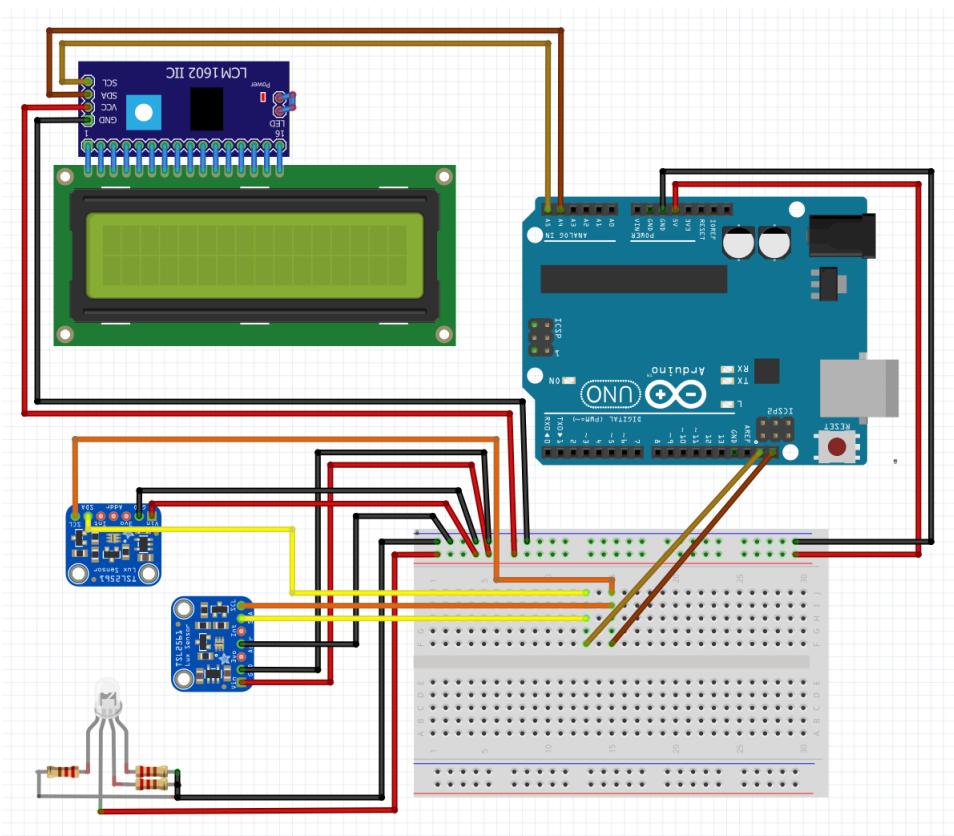
- Koble lysdiodens katode til jord og dens anode til 5v.



Figur 9: Koblingsskjema som viser hvordan lysdioden skal kobles opp

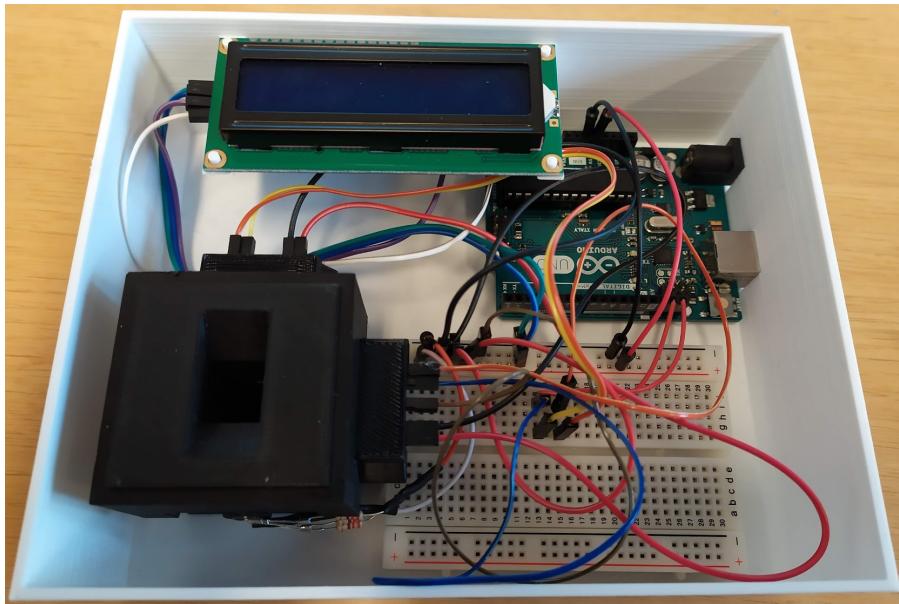
3.6 Ferdiggjøring av Oppkobling

Til slutt skal alt som er koblet opp ligne på figur 10. Du skal nå tre på skyggekappen over sensorasemblyen.



Figur 10: Koblingsskjema som viser hvordan alt kobles opp

Mitt spaghettimonster endte slik:



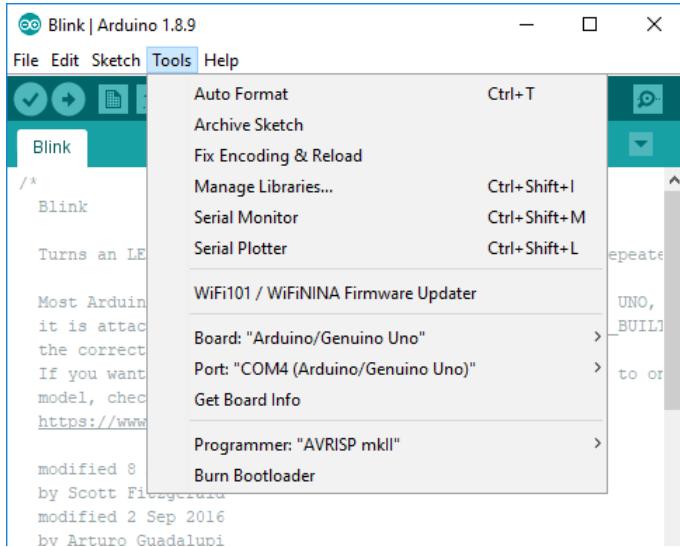
Figur 11: Eksempel av ferdig koblet hardware med skyggekappen satt på.

4 Software

4.1 Komme i gang med koding

Her skal vi sette opp Arduino IDE, og installere de bibliotekene vi trenger for vårt formål. Hvis stegene blir vanskelige å følge, eller noe uforutsett oppstår, konsulter google.

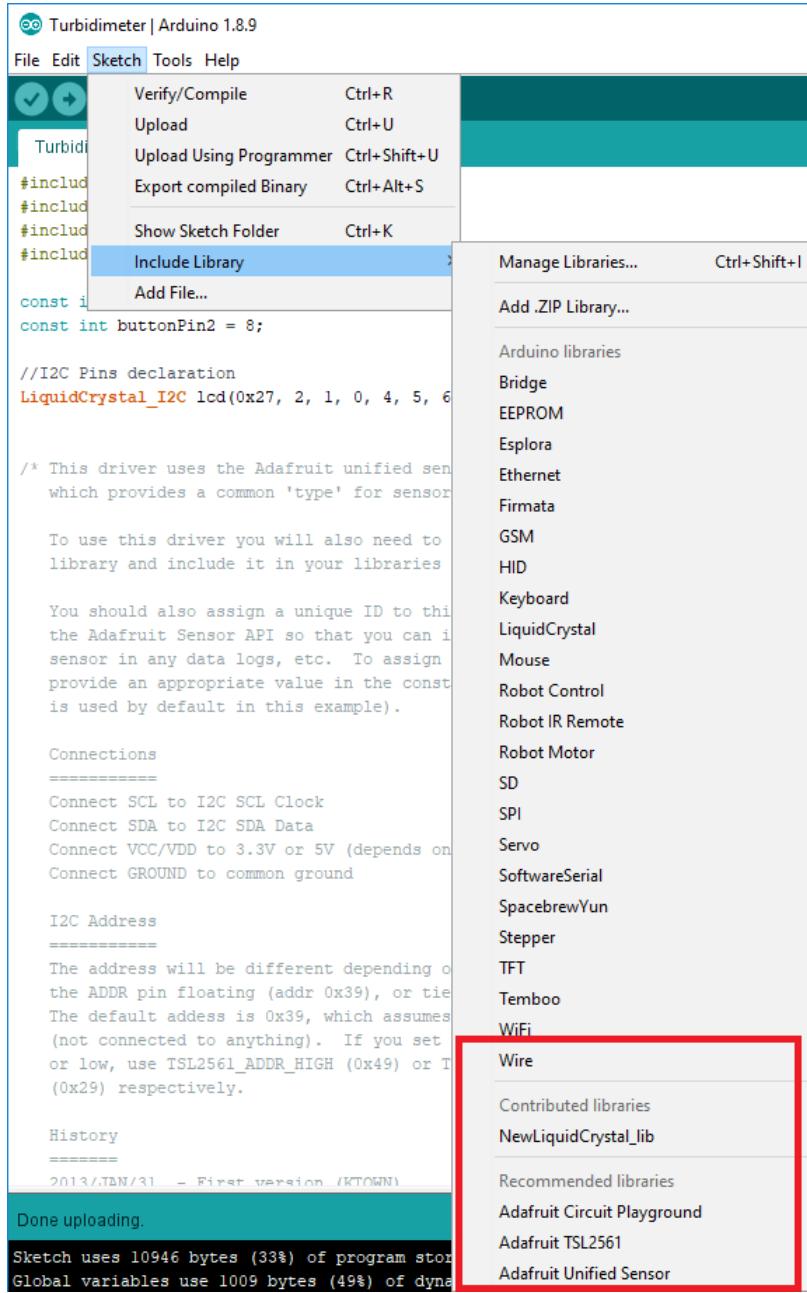
- Last ned Arduino IDE herfra: <https://www.arduino.cc/en/Main/software>. Jeg anbefaler å bare bruke ZIP-versjon (da den ikke behøver adminrettigheter), og å unngå Windows ”Appen”. Jeg kommer til å bruke ZIP-versjonen. Pakk den første mappen ut av ZIP, og legg den et sted du vil ha den.
- Plugg en USB-kabel inn i Arduinoen og i PC-en du skal bruke.
- Åpne Arduino IDE. Velg først arduinoen ved å trykke Tools → Board: → Arduino/Genuino Uno. Og deretter Tools → Port → og så den aktuelle USB-porten.



Figur 12: Hvordan det skal se ut i menyen når du har valgt type og port.

- Test deretter at det funker ved å laste inn en eksempel-sketch for så å laste opp til arduinoen. Dette gjør du ved å trykke File → Examples → 01.Basics → Blink. Et nytt vindu vil komme opp med koden. Trykk den blå pilen øverst til venstre for å laste opp og kjøre på Arduinoen. Hvis alt gikk bra skal lysdioden på selve arduinoen blinke.
- Nå behøver vi noen ekstra bibliotek for å kjøre både LCD og lyssensorene våre. Last ned ZIP-filen og pakk den ut i libraries mappen til Arduino IDE. Hvis du lastet ned ZIP-filen for Arduino IDE tidligere, kan du bare legge mappene i libraries.zip rett inn i libraries mappen inne i arduino mappen du fikk da du pakket den ut tidligere.
For flere detaljer, se avsnitt ”Manual Installation” her <https://www.arduino.cc/en/guide/libraries>.

Etter en restart av Arduino IDE burde du nå finne Adafruit Unified Sensor, Adafruit TSL2561, Wire og newLiquidCrystal.lib under Sketch → Include Library.



Figur 13: Hvordan det skal se ut i menyen når du har installert bibliotekene.

4.2 Koden

Da er allt klart for selve koden! I prosjektfilene ligger det en mappe som heter turbidimeter, med en fil som heter turbidimeter.ino. Dette er kodefilen som du kan åpne i Arduino IDE'en. En slik prosjektfil må alltid ligge inni en mappe med samme navn.

5 Kalibrering

5.1 Lage kalibreringsløsninger

For å kalibrere turbidimeteret trenger vi noen løsninger vi allerede vet turbiditeten på. Dette gjør vi ved å blande vann og melk. Marker først glassene fra 1 til 5, og drypp riktig antall melkedråper ned i hver glass. Fyll deretter vann til 50mL-merket på alle de fem begerglassene. Nå brude du ha 5 løsninger med

Glass nr.	Dråper melk	Teoretisk NTU-verdi
1	1	400NTU
2	2	555
3	3	666
4	4	777
5	5	888

Tabell 3: Tabell over kalibreringsløsningene

NTU-verdi som i tabell 3.

5.2 Måle data

- Begynn med å røre løsningen i hvert begerglass først, før du overfører den til kyvetter som passer i turbidimeteret. Husk å markere kyvettene med riktig tall, 1-5.
- Gå deretter i arduino-koden og fjern kommenteringen rundt den biten av koden som du skal bruke til kalibreringen, og legg til kommentering rundt koden som brukes senere. Dette finner du helt nederst i koden. Du kommenterer ut en del ved å sette /* og */ rundt koden. For eksempel: /* void loop */. Se figur 14.

```

//Kommenter ut seksjonen under når du kalibrerer turbidimeteret
/*
//Viser turbiditeten på LCD-displayet
lcd.clear();
lcd.setCursor(0,0);
lcd.print(NTU); lcd.print("NTU");
*/

//I kalibreringsmodus tar du vekk kommentaren av dette feltet,
// og kommenterer ut den vanlige turbiditetsvisningen over.
//Dette viser råverdiene verdiene fra sensorene på LCD'en slik at du kan kalibrere.
lcd.clear();
lcd.setCursor(0,0);
lcd.print(event1.light/1000);
lcd.setCursor(0,1);
lcd.print(event2.light/100);
delay(250);

```

Figur 14: Viser hvordan koden skal se ut når man kalibrerer

- Last deretter koden opp til arduinoen. Nå vil LCD'en vise rå sensordata.
- Vend kyvette 1 opp ned et par ganger for å blande, og sett den i turbidimeteret. Vent 10 sekunder og skriv ned sensorverdiene. Kall gjerne dataene fra den øverste linja sensordata 1, og fra den nederste sensordata 2. Sensor 1 er forøvrig sensoren rett ovenfor lysdioden hvis du lurte.
- Gjenta dette for de fire andre kyvettene.
- Du har nå fem målepunkter med rådata fra sensorene, som du kan matche med deres teoretiske NTU-verdi!

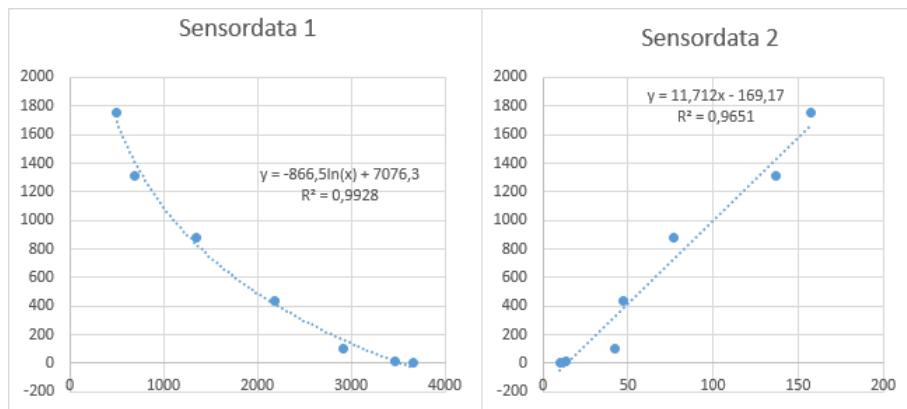
5.3 Regresjon

- Før opp dataene i regnearket som i figur 15.

Glass nr.	NTU-verdi	Sensordata 1	Sensordata 2
1	500	2186	47,41
2	1000	1131	98,41
3	1500	602	151,21
4	2000	470	211,13
5	2500	231	278,31

Figur 15: Eksempel på tabell i regnearket

- Marker deretter NTU-verdiene og sensordata 1, klikk Insert og finn de små prikkene i grafen under Charts. Trykk disse og sett inn et Scatterchart.
- Nå vil vi ha NTU-verdi på Y-aksen og måledata på X-aksen, så trykk trakten til høyre for plottet, og klikk "Select data.." nede til høyre. Under Legend entries (Series) trykker du Edit. Under Series X values Trykker du på knappen helt til høyre og markerer X-verdiene dine, altså måledataene fra sensordata 1. Under Series Y values trykker du på knappen og velger NTU-verdiene. Trykk OK. Du vil nå ha NTU-verdier på Y-aksen og sensordata 1 på X-aksen.
- Klikk på Plusstegnet ved det nye plottet, gå ned til trendline og trykk på pilen mot høyre. Trykk deretter "More options..". I innstillingene til høyre huker du av for Display R-squared value on chart og Display equation on chart. Bytt mellom regresjonsmodellene, ofte enten exponential, logarithmic eller linear, til du finner den som har minst R-squared verdi (feilverdi). Du burde nå ha en funksjon på grafen din som tilhører regresjonen din. Se figur 16.



Figur 16: Viser et eksempel på hvordan regresjonsplottene kan se ut.

- Gjør dette på nytt, bare for sensordata 2.
- Du skal nå ha 2 regresjonsfunksjoner.

5.4 Modifisere koden

- Nå går du tilbake til Arduino-IDE og kommenterer ut den koden du kommenterte inn i sta. Kommenter inn den koden som viser NTU-verdiene også.
- Vi må nå sette inn vår regresjonsmodell. Finn definisjonene av NTU1 og NTU2, og endre disse til deres regresjonsfunksjoner. Se figur 17 for eksempel.

```
//Her setter vi inn regresjonen for NTU-verdi for sensor 1 og sensor 2
NTU1 = -866.5*log(event1.light/1000)+7076.3;
NTU2 = 11.712*(event2.light/100) -169.17;

//Dette viser NTU=0 hvis beregnet NTU er negativ, da vi ikke kan få negativ NTU i virkeligheten.
if (NTU1<0){
    NTU1 = 0;
}
if (NTU2<0){
    NTU2 = 0;
}

//Total beregnet NTU er et snitt av beregningene fra begge sensorene.
NTU = (NTU1+NTU2)/2;

//Kommenter ut seksjonen under når du kalibrerer turbidimeteret

//Viser turbiditeten på LCD-displayet
lcd.clear();
lcd.setCursor(0,0);
lcd.print(NTU); lcd.print("NTU");

/*
//I kalibreringsmodus tar du vekk kommentaren av dette feltet,
// og kommenterer ut den vanlige turbiditetsvisningen over.
//Dette viser rawverdiene verdiene fra sensorene på LCD'en slik at du kan kalibrere.
lcd.clear();
lcd.setCursor(0,0);
lcd.print(event1.light/1000);
lcd.setCursor(0,1);
lcd.print(event2.light/100);
delay(250);
*/
}
```

Figur 17: Viser hvordan koden kan se ut når du har kommentert ut riktig, og satt inn egen regresjonsmodell for hver av sensorene.

- Last deretter opp den ferdige koden til arduinoen din.

Gratulerer, turbidimeteret er ferdig!

6 Appendiks

6.1 Lysdiode

Hvis du har en hvit LED, må du finne i dataarkene hvilket spenningsfall den behøver, og velge motstand deretter. Du kan følge fremgangsmåten under.

Det vi gjorde enklere over var å anta at alle fargene i dioden har samme spenningsfall. Det fungerer fint slik vi gjorde over, men hver farge vil ikke ha lik intensitet. For å fikse dette kan vi gjøre følgende:

- Sjekk databladet for spenningen til dioden. Dette beskrives ofte med ”Forward Voltage” eller V_f og er ofte forskjellig for de tre fargene i dioden. Hvis du ikke finner et datablad er det vanlig at det ligger på 1.9V for rød, 3.2V for grønn og 3.3V for blå. Finn også ut hvor mye strøm hver farge behøver ved hjelp av databladet. Det bruker å ligge på 20mA for hver farge.
- Siden vi vet at det totale spenningsfallet i arduinoen fra positiv til negativ er 5V må vi gjøre slik at også spenningsfallet over dioden er 5V. For å få til dette bruker vi motstander. For å regne ut hvilken motstand vi må bruke kan vi bruke Ohms lov: $V = I \cdot R$, Hvor V er spenning, I er strøm og R er motstand.
- Vi bruker at spenningsfallet over motstanden og over dioden må bli 5V til sammen. Da får vi for eksempel for rødt at $V = 5V - 1.9V = 20 \cdot 10^{-3}A \cdot R$ hvor vi har satt inn at spenningsfallet over motstanden må være $V = 5V - 1.9V$ og at strømmen er $I = 20mA = 20 \cdot 10^{-3}A$. (Bruken av symbolet V kan være forvirrende, men når V står alene betyr det spenning og når det har et tall forran, for eksempel 1.9V, så betyr det Volt.) Vi gjør om regnestykket slik at motstanden R blir alene og får:

$$R_r = \frac{5V - 1.9V}{20 \cdot 10^{-3}A} = 155\Omega$$

$$R_g = \frac{5V - 3.2V}{20 \cdot 10^{-3}A} = 90\Omega$$

$$R_b = \frac{5V - 3.3V}{20 \cdot 10^{-3}A} = 85\Omega$$

- Når vi da velger motstander som er veldig like disse utregnede verdiene vil vi få maksimal intensitet for hver av de tre fargene, og dermed også få et lys som er hvitere og sterkere enn det vi får når alle motstandene er lik. Grunnen til at vi unngår denne løsningen er fordi de fleste har en del 220Ω motstander liggende, mens de motstandene vi behøver her er litt vanskeligere å få tak i.

6.1.1 Motstand på felles elektrode

Hvis du vil gjøre det enklere ved å sette motstanden på den felles elektroden kan du gjøre følgende.

- Først finner vi hvilken motstand vi behøver. Dette finnes ved å sette inn motstandene i følgende formel:

$$\frac{1}{R} = \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3} + \dots$$

og får:

$$\begin{aligned}\frac{1}{R} &= \frac{1}{220} + \frac{1}{220} + \frac{1}{220} \\ \frac{1}{R} &= \frac{3}{220} \\ R &= \frac{220}{3} \\ R &\approx 73\end{aligned}$$

Vi ser at motstanden vi må ha på den felles elektroden er 73Ω .