

Sprengt potet? Overraskende lysmåler (håper jeg)

Målet mitt for denne oppgaven er å utforske forholdet mellom lysintensitet (målt i lux) og kamerainnstillinger (blender, lukkertid, iso) som gir et riktig eksponert bilde. For å ha et konkret og gøy mål vil jeg bruke det jeg lærer til å lage en enkel hjemmelaget lysmåler. (For meg er dette gøyere enn poteten din Nome, men du har lov til å være uenig om du vil=))

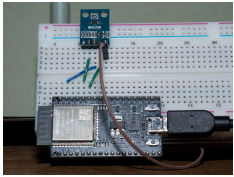
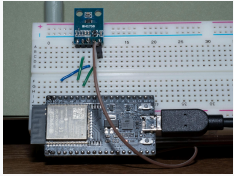
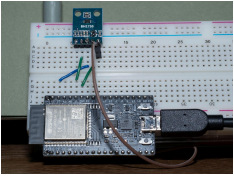
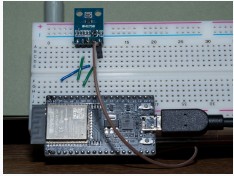
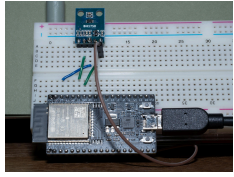
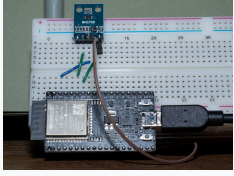
For å gjøre prosjektet gjennomførbart har jeg valgt å gjøre noen forenklinger:

1. Av praktiske grunner er alle målinger er tatt med blits. Det ga meg mye variasjon av lysintensitet på kortest mulig tid og uten å måtte lete etter en haug med ulike lyskilder. Men det betyr også at den sannsynligvis ikke vil være veldig hjelpsom i andre lysforhold.
2. Alle målinger er tatt på samme avstand (ca 40 cm fra sensor til kamerahus) avstand for å utelukke variabelen om hvordan lyset avtar etter avstand.
3. Den mest entydige måten å måle lys for foto er å måle lyset som treffer motivet, da slepper man for eksempel å ta hensyn til for eksempel hvilke farger motivet har osv og hvordan det påvirker lyset i bildet. Fordelen med dette er at det også er den enkleste formen for lysmåler å lage selv, siden man bare trenger en enkel lyssensor som kan måle i lux og som kan plasseres der motivet skal være. Oftest bruker man da en hvit dome over sensoren, dette skal gi den mest nøyaktige målingen av lys som treffer motivet -og som treffer fra ulike vinkler, men denne har jeg valgt å droppe for denne gang for å gjøre oppsettet så raskt og enkelt som mulig. Jeg bruker uansett bare en blits rett forfra i et relativt mørkt rom (målt til oppunder 25 lux) så det går fint.

Gjennomføring

1. Målinger med BH1750 i sammenheng med eksponering og kamerainnstillinger:

Jeg koblet en lyssensor BH1750 til en esp32 og målte hvor mange lux som traff sensoren for ulike styrker av blitsen og noterte meg hvilke kamerainnstillinger jeg brukte for å få en riktig eksponering.

Blender	Lukkertid	ISO	Blits	Luxmålinger	Lux snitt	Foto for å se eksponering
22	1/200	3200	1/256	334, 331, 343, 336, 347, 328, 318, 346, 350, 330, 348, 357, 344, 315, 363	339.3333333	
22	1/200	1600	1/128	395, 393, 405, 403, 406, 388, 382, 406, 408, 412, 393, 394, 411, 399, 391	399.0666667	
22	1/200	800	1/64	469, 483, 478, 477, 472, 484, 476, 479, 469, 486, 481, 493, 466, 465, 483	477.4	
22	1/200	400	1/32	578, 563, 574, 569, 582, 580, 587, 582, 578, 587, 578, 589, 573, 569, 576	577.6666667	
22	1/200	200	1/16	761, 757, 734, 743, 742, 756, 748, 746, 759, 753, 749, 740, 734, 742, 745	747.2666667	
22	1/200	100	1/8	1025, 1018, 1033, 1014, 1033, 1029, 1025, 1027, 1024, 1037, 1025, 1032, 1018, 1019, 1029	1025.866667	

2. Den matematiske sammenhengen mellom innstillinger, eksponeringsverdi (benevnes EV - kall det lysintensiteten i et bilde) og lysintensiteten i en scene (lux):

EV er trikket her. Eksponeringsverdien kan finnes både utifra kombinasjonen av innstillinger:

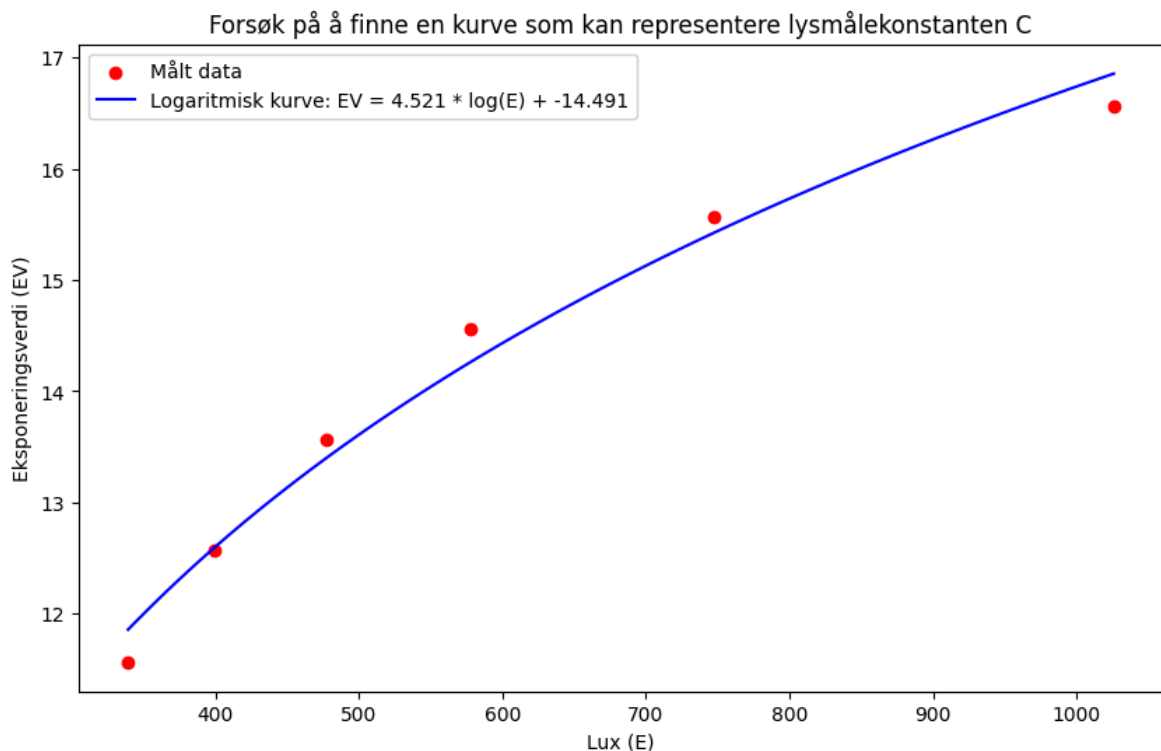
$EV = \log_2 (N^2 / (t * (S/100)))$, der N er blender, t er lukkertid og S er ISO.

...og utifra lysintensiteten:

$EV = \log_2 (E / C)$, der E er lysintensitet i lux, og C er en konstant for måling av incident (usikker på norsk, innfallende kanskje?) lys.

Den første av disse formlene har jeg sett et par ganger før, men den andre har jeg letet godt og lenge etter, men jeg fant den til slutt på wikipedia¹ (med henvisning til en bok kalt "The Manual of Photography" som tydeligvis har nok teknisk snacks til at den nå er temmelig langt oppe på "vil lese"-listen min).

Selv om Wikipedia kunne fortelle meg at en vanlig verdi av C er 250 hadde jeg lyst til å gjøre noen tester for å se om det stemte for lyssensoren jeg har, en BH1750. Det viste seg å være en god idé for den var mer lik en logaritmisk funksjon enn en konstant. Jeg så helt bort ifra formelen jeg hadde lett internett rundt etter og brukte heller Python for å finne en kurve som kunne stemme ganske godt med målte verdier av lux og EV regnet ut fra kamerainnstillingene.



Resten av disse utregningene ligger i en Jupiter notebook i github.

¹ https://en.wikipedia.org/wiki/Exposure_value

Eksponeringskalkulator:

Den gøyeste delen av dette prosjektet, da formlene først var funnet og jeg endelig kunne sette opp en utregning hele veien fra luxmåling til blenderinnstilling og se at det fungerte.

```
# 3. Finne blenderinnstillinger fra valgt lukker, iso og målt Lux:

import numpy as np

# 12 ulike verdier for hver innstilling, f = blender, l = lukkertid, i = iso
N = np.array([45, 32, 22, 16, 11, 8, 5.6, 4, 2.8, 2, 1.4, 1])
t = np.array([1/2000, 1/1000, 1/500, 1/250, 1/125, 1/60, 1/30, 1/15, 1/8, 1/4, 1/2, 1])
S = np.array([25, 50, 100, 200, 400, 800, 1600, 3200, 6400, 12800, 25600, 51200])

# EV fra innstillinger (blender = N, lukker = t, ISO = S):
def EVs(N, t, S):
    return np.log2((N**2)/(t*(S/100)))

# EV fra Lux (E):
def EVc(E):
    return 4.521185 * np.log(E) + -14.490967

# toleranse for å regnes som ekvivalent EV
tolerance = 0.3

#inputverdier
velgt = 1/100
velgS = 200
luxmaal = 400
EVlux = EVc(luxmaal)

#bruker en for løkke til å finne riktig blender, for å få faktiske blenderverdier fra listen over. Kunne skrevet form
for Nv in N:
    expcurr = EVs(Nv, velgt, velgS)
    if abs(EVlux - expcurr) < tolerance:
        blenderout = Nv

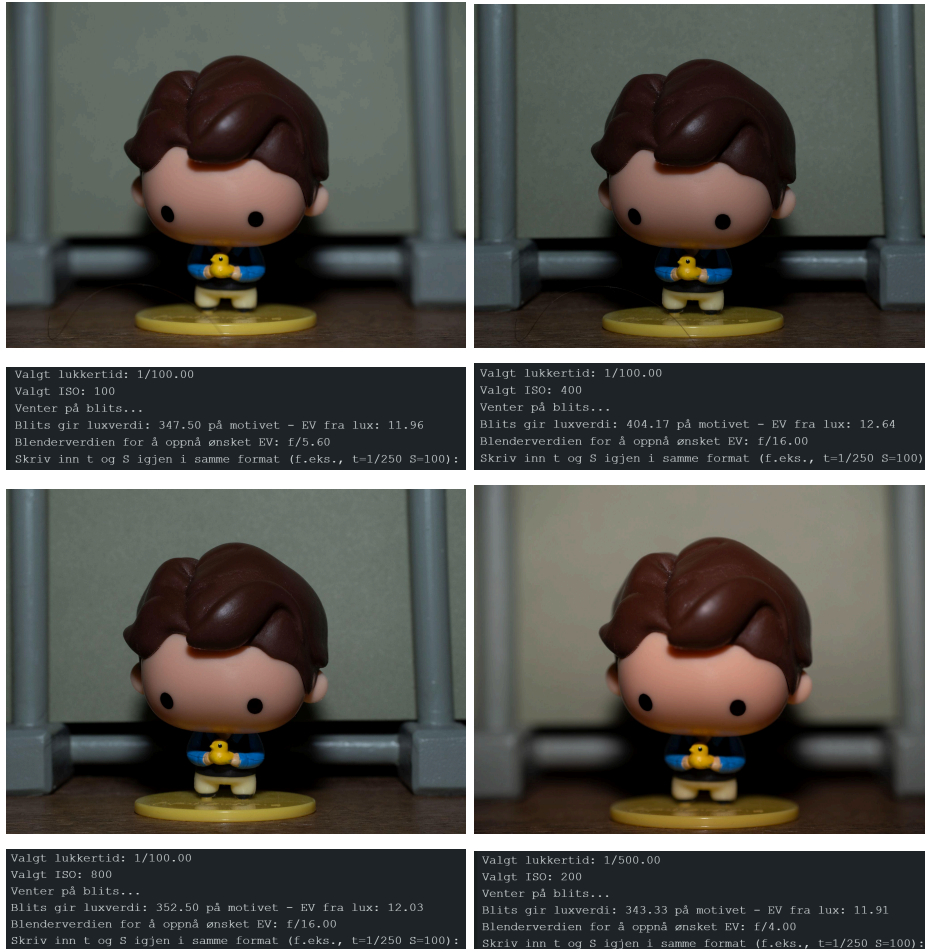
print(f"{luxmaal} Lux gir en EV på {EVlux:.2f}")
print(f"I kombinasjon med lukkertid 1/{round(1/velgt)} og iso {velgS} kan dette oppnås med blender: {blenderout}")
```

3. Arduino-kode som tar inn valgfri lukker og iso, måler lux og gir ut anbefalt blender:

Neste steg var å få de samme formlene inn i Arduino. Koden her ber først bruker om verdier for lukker og iso som input, og venter deretter på et blitsblink, før den gjør utregningene på samme måte som Python-koden og spytter ut en passende verdi for blenderen.

4. Testing:

Med kode på esp32en måtte den selvsagt testes, så her har du noen bilder der jeg har latt den bestemme blenderverdi for meg (sammen med outputen fra Arduino):



Jeg vil nok kanskje si at disse er en smule undereksponert, men når jeg nå ser tilbake på bildene jeg tok under målingene ser jeg jo sammenhengen. Det er nok en vane som kommer av at det er lettere å redde inn det som er litt for mørkt enn litt for lyst i etterarbeid på bilder tatt med moderne kameraer, og så her har jeg kanskje jobbet litt for mye på autopilot med fotobiten. Dersom jeg skal faktisk ender med å lage en frittstående lysmåler tror jeg nok jeg må ta nye målinger og være litt mer nøyaktig.

Skal sies at dette stemmer godt med resten av erfaringene her, jeg trodde blant annet flere ganger at jeg hadde funnet en feil, en eller annen eksponering som ga feil output, for jeg fikk noen bilder som dette ->

Og så viste det seg at jeg bare hadde glemt å justere en av innstillingene på kamera etter det jeg skrev inn i Arduino. Så først og fremst har jeg lært å ikke stole på meg selv.



5. **Github:**
<https://github.com/arifjell/OBLIG---TMA4101>