

FSK-demodulator

Tittel: Frequency shift keying demodulator

Forfattere: Markus Søvik Gunnarsson

Versjon: 1.0

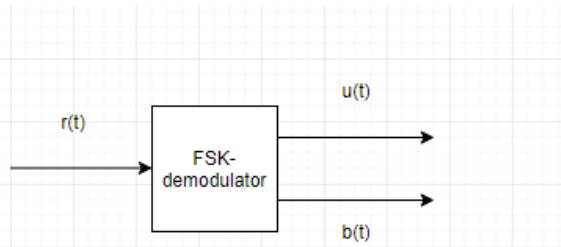
Dato: 2. desember 2018

Innhold

1	Problembeskrivelse	1
2	Prinsipiell løsning	2
2.1	A/D-omforming	3
2.2	Digital filtrering	3
2.3	Absoluttverdi og midling	3
2.4	Indikasjon	3
3	Realisering og test	3
3.1	A/D-omforming	3
3.2	Digital filtrering	4
3.3	Absoluttverdi og midling	5
3.4	Indikasjon	5
3.5	Test	6
4	Konklusjon	10
Referanser		10

1 Problembeskrivelse

Dette notatet tar for seg et design av et system som vist i Figur 1.



Figur 1: Blokkdiagram av en FSK-modulator. Figur hentet fra [1]

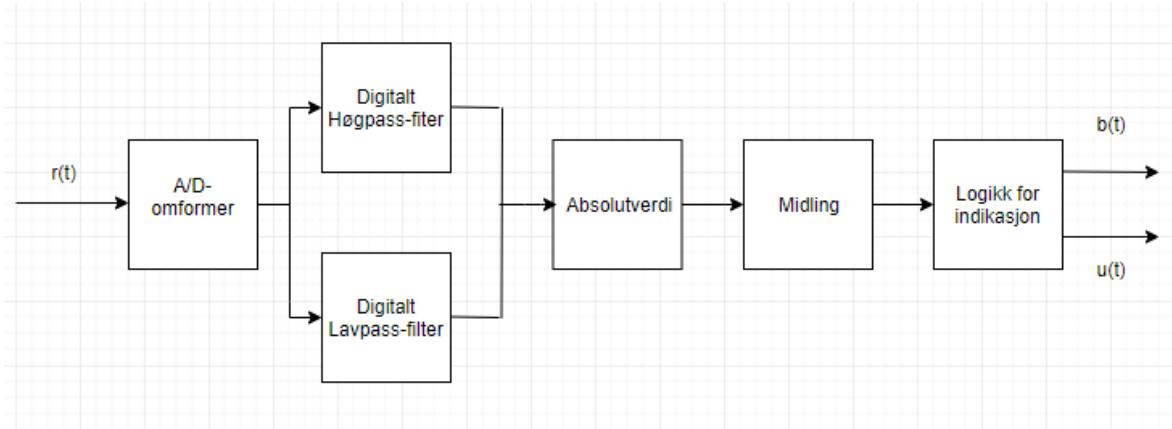
Figur 1. viser et blokkdiagram av en FSK-demodulator med inngangssignal $r(t)$ og utgangssignal $b(t)$ og $u(t)$. En FSK-demodulator blir brukt som mottaker til et FSK-signal. Et FSK-signal er et sinussignal med konstant amplitude A . Sinussignalet har en varierende frekvens mellom to frekvenser f_0 og f_1 , derav navn frekvensskift-modulasjon. Engelsk - Frequency shift keying.[2]

I dette notatet blir det laget en FSK-demodulator med følgende krav:

- Inngangssignalet $r(t)$ skal være sinusformet med en frekvens som varierer mellom f_0 og f_1 .
- Utgangssignalet $u(t)$ skal ha verdien HØY dersom f_0 eller f_1 er tilstede i inngangssignalet $r(t)$.
- Utgangssignalet $u(t)$ skal ha verdien LAV ellers.
- Utgangssignalet $b(t)$ skal ha verdien LAV dersom f_0 er tilstede i inngangssignalet $r(t)$.
- Utgangssignalet $b(t)$ skal ha verdien HØY dersom f_1 er tilstede i inngangssignalet $r(t)$.
- Demodulatoren skal ha et areal mindre enn $4cm^2$

2 Prinsipiell løsning

En FSK-demodulator kan implementeres på mange forskjellige måter og kompleksiteter. En mulig løsning er vist i figur 2.



Figur 2: Et blokkdiagram av en mulig FSK-demodulator.

Kretstopologien til FSK-demodulatoren er vist i Figur 2. Inspirasjon til løsningen er hentet fra et teknisk notat om digital kommunikasjon med FSK [2]. Fra figuren kan man se at demodulatoren inneholder ulike seksjoner. En mikrokontroller kan brukes til å implementere det meste av seksjonene i figur 2. Mikrokontrollere er som oftest ikke større enn $4cm^2$, slik at kravet om arealet blir overholdt.

2.1 A/D-omforming

For å gjøre om det analoge signalet $r(t)$ til et digitalt signal som mikrokontrolleren kan arbeide med kan man bruke en analog-til-digital omformer. Da er det viktig at frekvensene man skal sample (omforme) skal være mindre enn $\frac{f_s}{2}$, der f_s er samplingsfrekvensen. Ved å finne de to frekvensene f_2 og f_1 ved hjelp av et oscilloskop kan man bestemme en passende samplingsfrekvens f_s .

2.2 Digital filtrering

Digitale filter er algoritmer realisert med datamaskiner. I dette notatet brukes det for å skille de to frekvensene f_0 og f_1 . Den ene frekvensen vil ha større amplitude enn den andre i lavpassfilteret og motsatt i høypassfilteret. En av to hovedkategorier innen digitale filter er IIR-filter. Dette står for infinte impulse response. Disse filtertypene kan være rekursive og ha uendelige mange register med tidlige verdier. En av de mange gode egenskapene til digitale filter er at de kan implementeres på mange ulike måter.

2.3 Absoluttverdi og midling

For å kunne estimere verdien av ingangssignalet(sinussignal) implementeres logikk for absoluttverdi og middelverdi. Sinussignalet får da en tilnærmet DC-verdi for de to ulike frekvensene f_0 og f_1 . Dette gjør at implementeringen av indikasjon er betydelig lettere.

2.4 Indikasjon

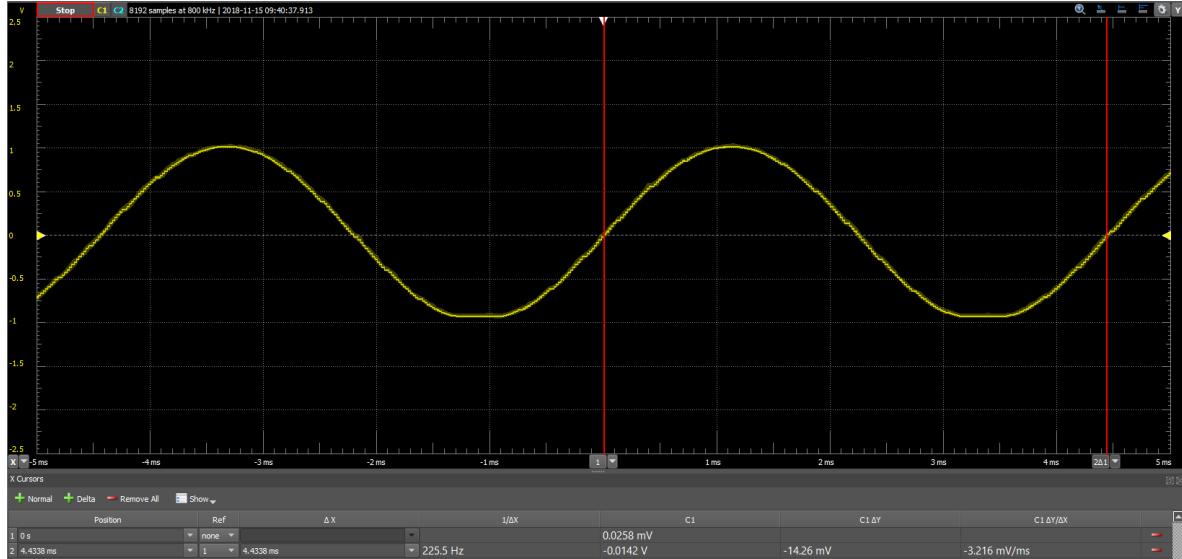
Indikasjonen blir tilpasset kravene til FSK-demodulatoren. Ved å programmere en mikrokontroller kan man bruke digital logikk til å styre eventuelle aktuatorer.

3 Realisering og test

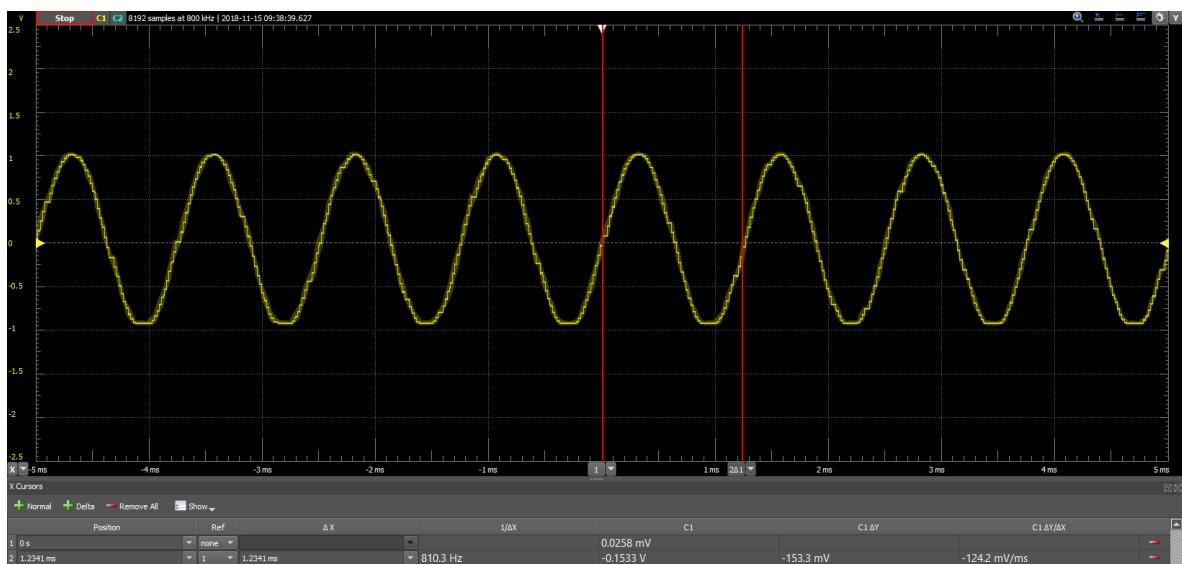
Mikrokontrolleren som bruke i dette notatet er en ATmega328P[3] som er programmert i utviklingsmiljøet Arduino IDE.[4].

3.1 A/D-omforming

Signalet $r(t)$ ble koblet opp til et oscilloskop og de to frekvensene f_0 og f_1 er bestemt fra skjermbildet i figur 3. og 4.



Figur 3: Skjermbilde av signalet $r(t)$ med frekvens $f_0 = 225$ Hz



Figur 4: Skjermbilde av signalet $r(t)$ med frekvens $f_1 = 810$ Hz

Figur 3. og 4. viser at $f_0 = 225$ Hz og $f_1 = 810$ Hz. Punktprøvingsfrekvensen til A/D-omformeren i mikrokontrolleren blir valgt til å være $f_s = 2000$ Hz, da dette tilfredsstiller Nyquists samplingsteorem for begge frekvensene f_0 og f_1 .

3.2 Digital filtrering

De digitale filterene blir implementert som vist i figur 5.

```

//Rekursivt lavpassfilter
lavpass = (x-forrigex) + 0.5*lavpass;

//Rekursivt Høypassfilter
hoypass = (x-forrigex) + (-0.5)*hoypass;

```

Figur 5: kodesnutt fra Arduino IDE: digitale filter.

Figur 5. viser at de digitale filterene er implementert som rekursive filter. Det vil si at de kaller på sin tidligere verdi. Den eneste forskjellen på det digitale lavpassfilteret og det digitale høypassfilter er fortegnet på dempningsleddet 0.5.

3.3 Absoluttverdi og midling

I utviklingsmiljøet Arduino IDE finnes det en funksjon for absoluttverdier, denne ble brukt i dette notatet. Midlingen ble implementert ved å lage kode for summering av alle punktprøvingene og kode for tellingen av antall punktprøver. Middelverdien av punktprøvene er: sum av punktprøver delt på antall punktprøver.

3.4 Indikasjon

Logikk i henhold til kravene for indikasjon er vist i figur 6.

```

//Indikasjon
if (midlethoypass < terskel && midletlavpass < terskel) {
    u = 0;
    digitalWrite(8, LOW);
    digitalWrite(7, LOW);
} else {
    u = 1;
    digitalWrite(8, HIGH);
    if (midlethoypass > midletlavpass) {
        b = 1;
        digitalWrite(7, HIGH);
    }
    else if (midlethoypass < midletlavpass) {
        b = 0;
        digitalWrite(7, LOW);
    }
}

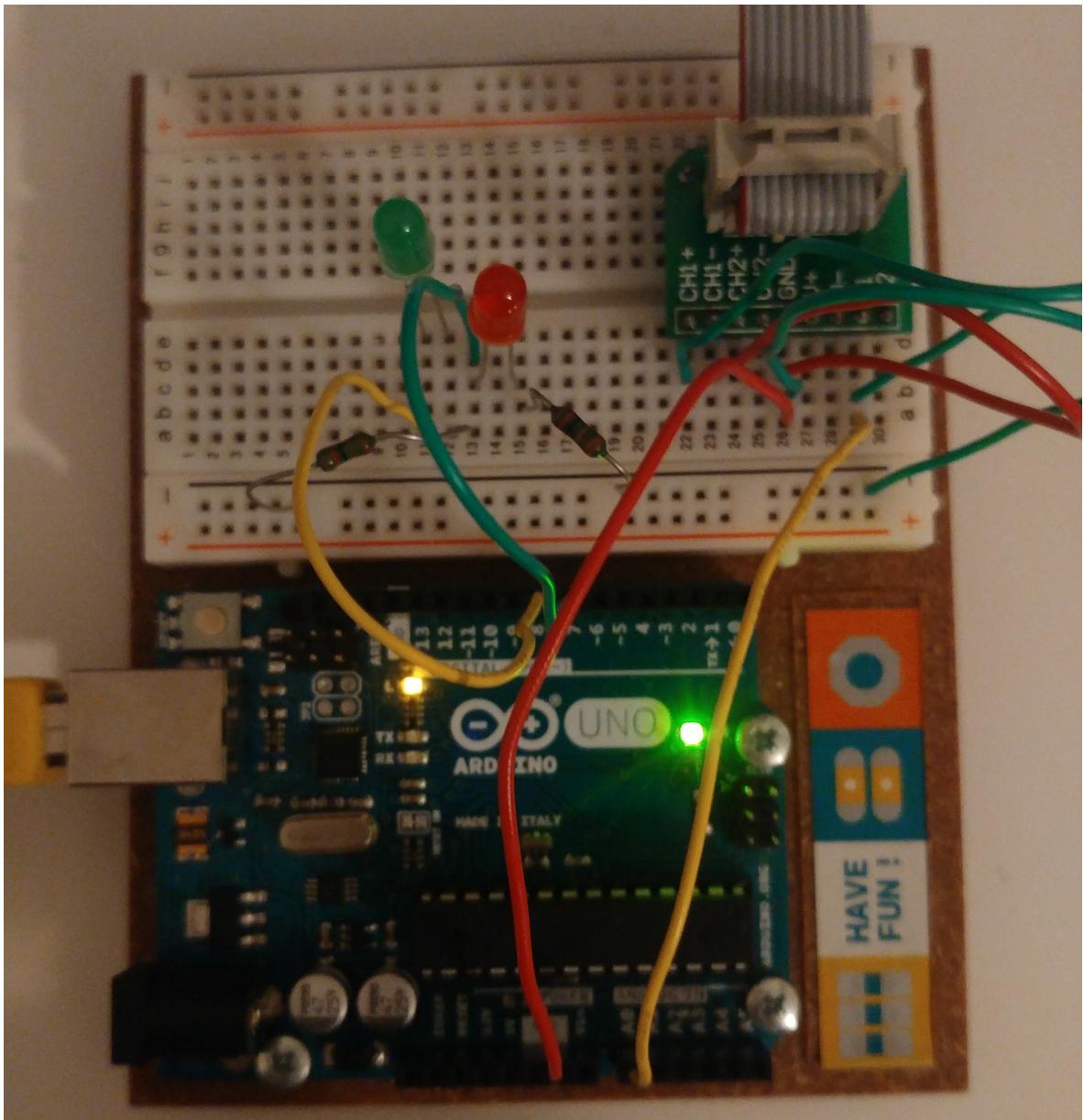
```

Figur 6: kodesnutt fra Arduino IDE: indikasjon. For utgangsignalene $u(t)$ og $b(t)$.

Logikken vist i figur 6. skrur av og på to LED som er koblet til mikrokontrolleren.

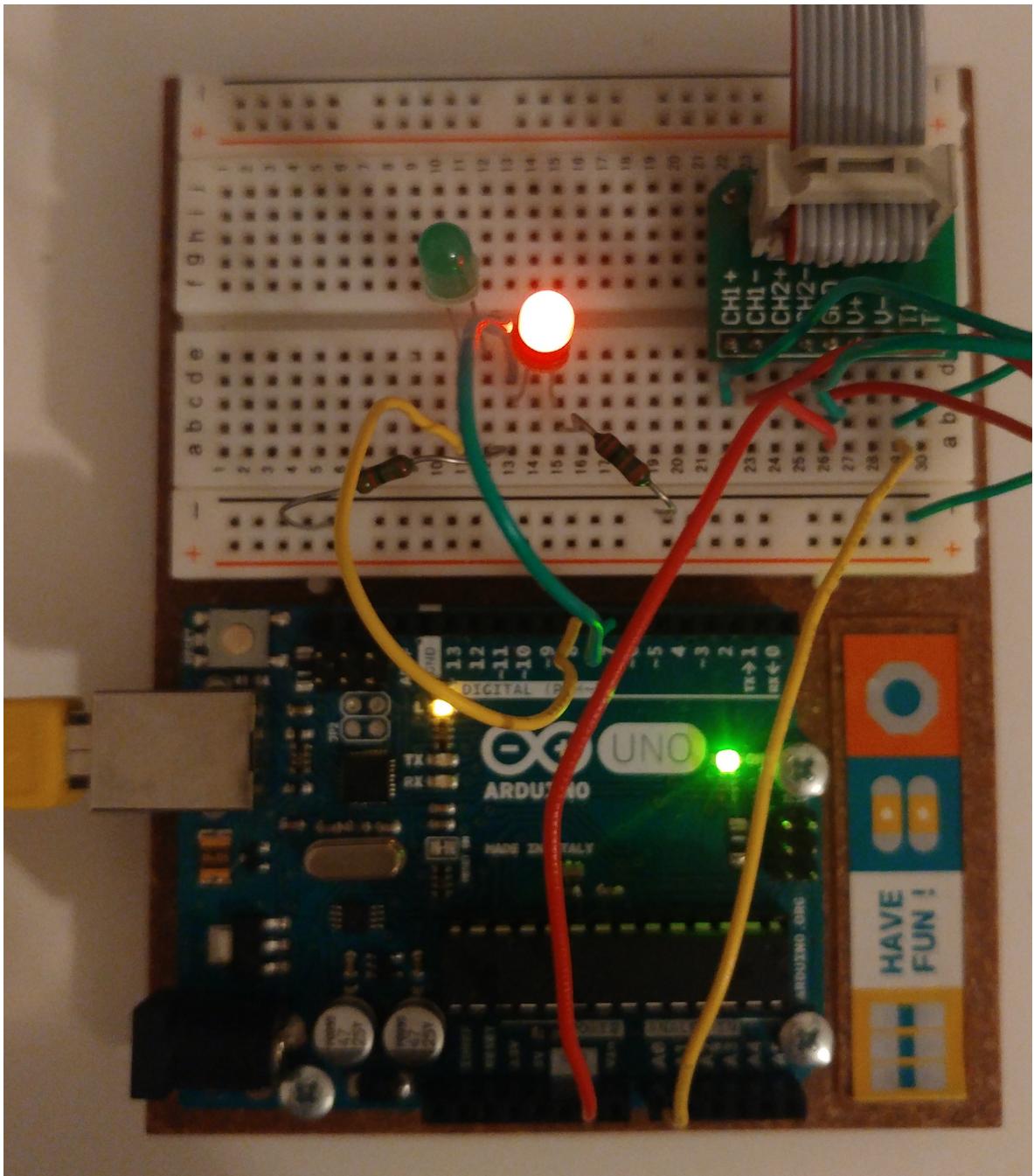
3.5 Test

Signalet $r(t)$ sendes gjennom FSK-demodulatoren. Figur 7. 8. og 9. viser de ulike tilstandene i kretsen.



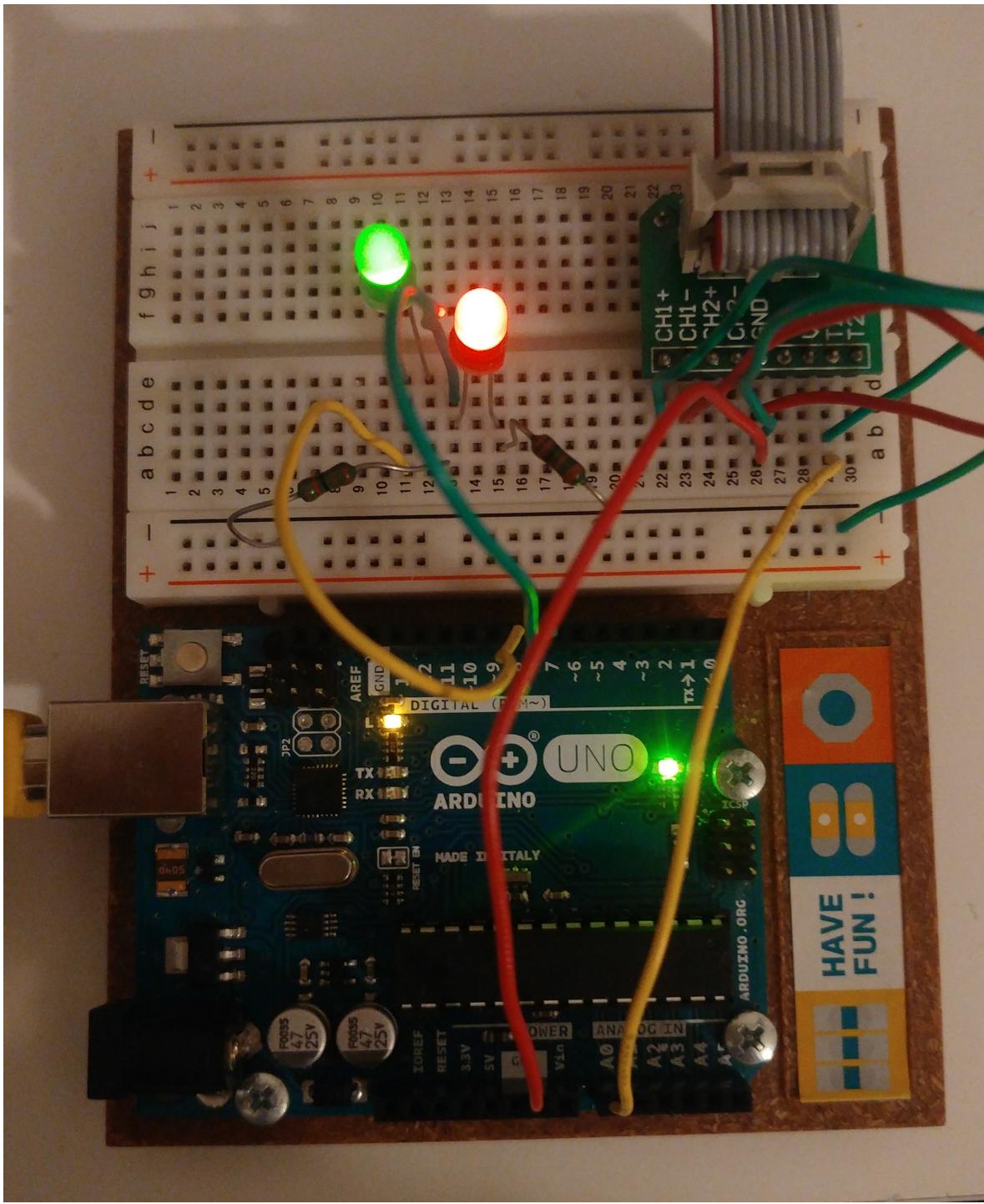
Figur 7: Bilde av FSK-demodulatoren: rød LED for utgangssignalet $u(t)$ og grønn LED for utgangssignalet $b(t)$.

Figur 7. viser at både $u(t)$ og $b(t)$ er LAV, det vil si at verken f_0 eller f_1 er tilstede i inngangssignalet $r(t)$.



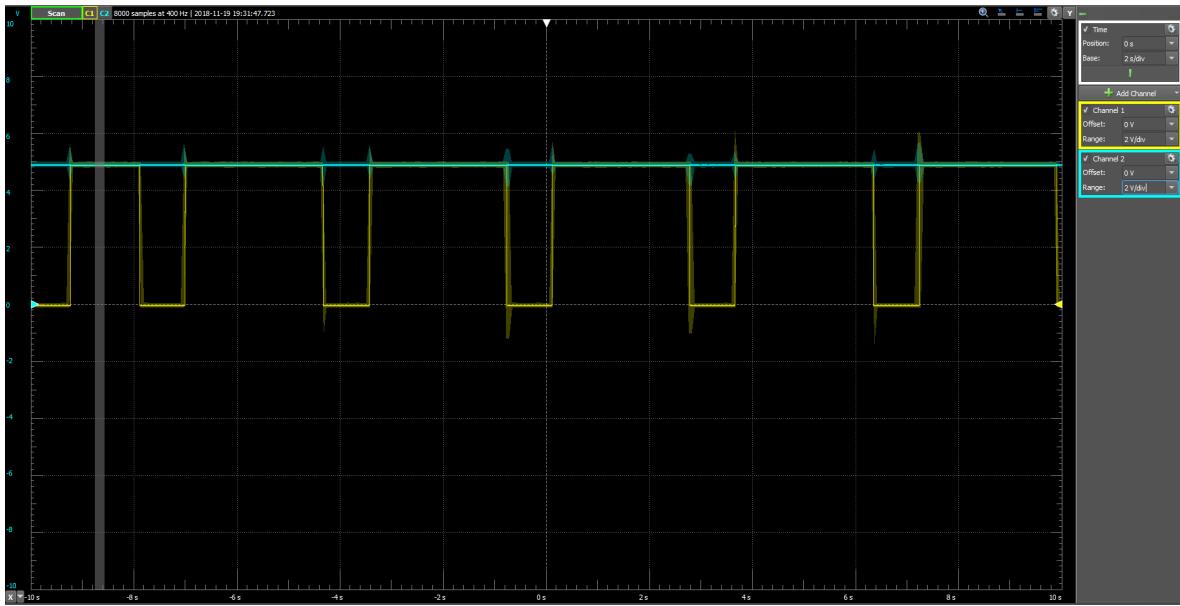
Figur 8: Bilde av FSK-demodulatoren: rød LED for utgangssignalet $u(t)$ og grønn LED for utgangssignalet $b(t)$.

Figur 8. viser at både $u(t)$ er høy og $b(t)$ er lav, det vil si at f_0 er tilstede i inngangssignalet $r(t)$.



Figur 9: Bilde av FSK-demodulatoren: rød LED for utgangssignalet $u(t)$ og grønn LED for utgangssignalet $b(t)$.

Figur 9. viser at både $u(t)$ og $b(t)$ er HØY, det vil si at f_1 er tilstede i inngangssignalet $r(t)$. Figur 10. viser utgangssignalene $u(t)$ og $b(t)$ målt i et oscilloskop.



Figur 10: Skjerm bilde av utgangssignalene i et oscilloskop: Gul graf = $b(t)$. Blå graf = $u(t)$. X-akse - tid [s]. Y-akse - spenning [V]

Figur 10. viser utgangssignalene $u(t)$ og $b(t)$ som endrer seg etter inngangssignalet $r(t)$ målt i et oscilloskop.

4 Konklusjon

Notatet tar for seg et design av en FSK-demodulator. Demodulatoren fungerte tilnærmet perfekt i henhold til kravene. Kravet om et areal på mindre enn 4cm^2 ble oppfylt ved å bruke en mikrokontroller. Kravet om indikasjon på utgangssignalene $u(t)$ og $b(t)$ ble oppfylt ved hjelp av programmeringslogikk i utviklingsmiljøet til mikrokontrolleren. En mulig feilkilde er dersom inngangssignalet $r(t)$ inneholder en tredje frekvens f_2 , vil det kunne gi utslag på utgangssignalet dersom middelverdien gjennom et av filterene er høyere en terskelen som er satt opp i logikken for indikasjon.

Referanser

- [1] Designprosjekt 8 - Problembeskrivelse - Elektronisk systemdesign og innovasjon
- [2] Digital kommunikasjon med FSK - Lars Lundheim referanse: *Elsys-2017-LL-1.2*
- [3] ATmega328P - Atmel <https://www.sparkfun.com/datasheets/Components/SMD/ATMega328.pdf>
- [4] Arduino IDE - Arduino <https://www.arduino.cc/>