Notater Prosjekt del 3

Oppgave 2

Bruker Can0, dette bestemmes i koden. Kan også bruke Can1, må da bytte pinnene på CAN-bus kortet.

Legger inn et CAN filter og kaller det defaultMask

Med FlexCAM biblioteket kan man sende meldinger som er opp til 8 byte når funksjonen «write(message)» brukes. Se readMe filen. To hex tall (FF) er det samme som 1111 1111 som er en byte, og man kan sende opp til 8 slike.

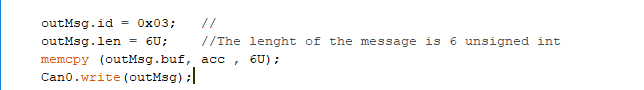
Bruker alternativ can pinner, så setter pinne 28 og 35 som outputs.

Oppgave 3

1. I setupen så settes ledPin som output, og så skrives den som høy. Sjekker om meldinger kommer i fra meldings ID 33, som tilsvarer 21 i hex. Når den meldingen kommer, samme hva den inneholder så vil LED på Teensyen toggles.
2. Leser av meldingen som er sendt og og’er den med 00000001. Hvis dette er lik 00000001 i tillegg til at meldingen kommer fra ID 34 (som er desimaltallet for 22 hex) så vil LED settes høyt. Hvis meldingen fra ID 34 og’es og blir lik 0, så vil Led settes lav.
3. Inkluderer Metro.h, definerer ledMetro = Metro(1000), som vil si at ledMetro er sann en gang hvert sekund, som er det samme som 1 Hz. Bruker deler av del 2 oppgave 1.

Koblet sammen to CAN busser med høy på høy og lav på lav.

void begin (uint32\_t baud = 250000, const CAN\_filter\_t &mask = defaultMask, uint8\_t txAlt = 0, uint8\_t rxAlt = 0);



Kilder:

[https://github.com/collin80/FlexCAN\_Library/blob/master/README.md (8](https://github.com/collin80/FlexCAN_Library/blob/master/README.md%20(8) byte)

<https://www.peak-system.com/produktcd/Pdf/English/PCAN-USB_UserMan_eng.pdf> (PCAN)