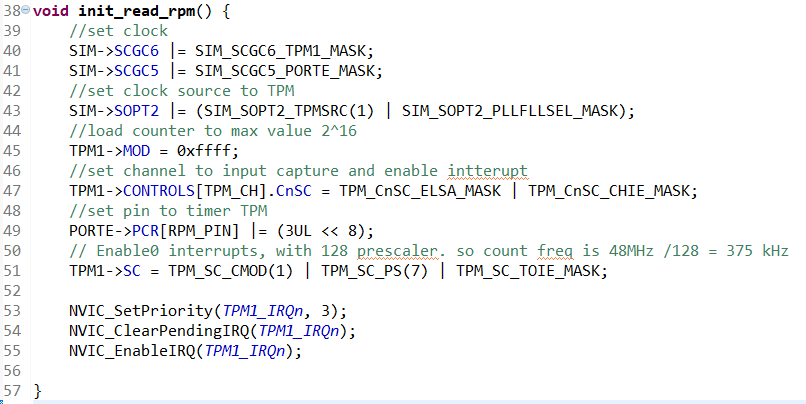
Implementering af kode til at aflæse rpm, der skal bruges til vores PID-regulering. Vi har taget udgangspunkt i aflæse et tacho signal som kommer fra vores tændspole. Dette signal lave en høj spænding ved hver omdr, og går lav igen.

For at afkode dette signal, bruges der input capture, der er implementeret i TPM periferenheden på kl25. Funktionen input capture, fungere ved at den tælle op til hver ’rising edge’ og gemmer denne værdi. Og når vi kender værdien og frekvensen vi tæller op med, kan vi omregne dette til RPM.

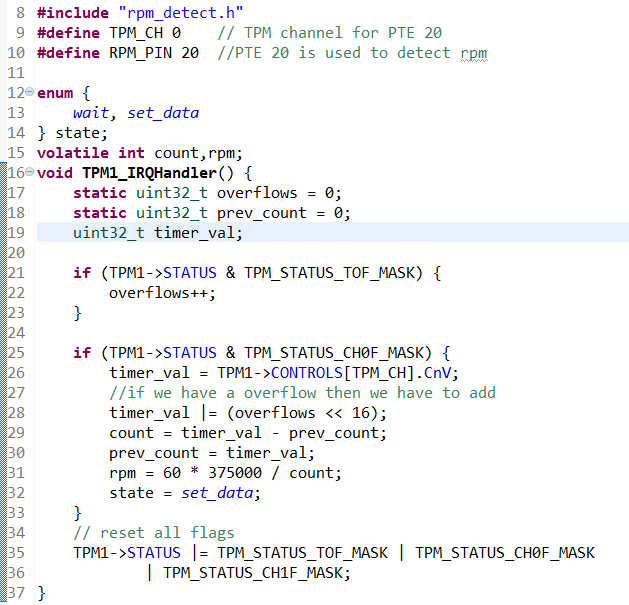
Ved formlen: *60 sek \* frekvens /input capture værdien = RPM*

Selve koden er vist her forneden, hvor vi bruger interrupts, til at opfange overflows og når vi får et rising edge, gemmer den værdi i CNV-registeret.



Her initialiseres vores TPM1 modul. Vores frekvens der bruges til at tælle op med, bliver nedskaleret, til 375 kHz. Dvs. imellem hver overflows går der 5,75 Hz, hvilket passer med vores måle område (17 - 167 Hz) vi arbejder i.

Ved testen, simuleret vi et tastesignal med frekvensen 10 Hz, og kunne decode det til 600 RPM -+ 20. Da der var lidt støj på signalet.



Her ses vores handler, hvori RPM bliver beregnet ud fra værdierne fra TPM1 perifer enhed. Og videre i koden kan man detektere om der er kommet ny RPM-data, ved bruge enum ’state’. Da det tager minimum en omgang på krumtappen, før den har beregnet en ny værdi.