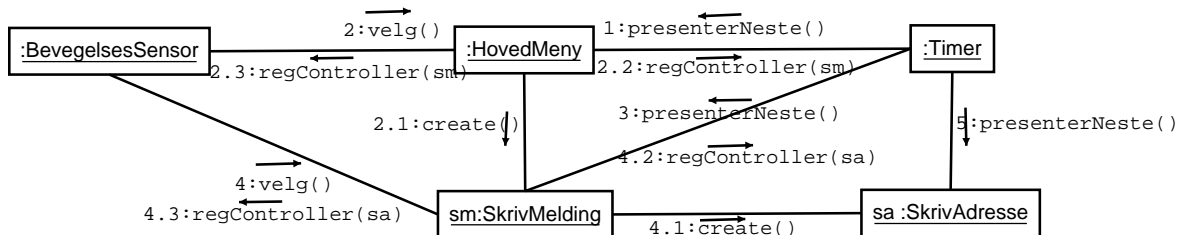


Figur 1: Generelt kommunikasjonsdiagram for en SnakkeController



Figur 2: Kommunikasjonsdiagram for start av SkrivMelding/SkrivAdresse

## Skisse til løsning av SnakkForMeg

Problemet er ganske komplekst med mange klasser. For kontrollen har vi laget figur 1. Det vil alltid være én aktiv controller. En controller erstatter en annen. Derfor vil de to objektene som genererer events, `BevegelsesSensor` og `Timer` alltid vite hvem som er gjeldende controller (metoden `regController`). Idéen her er altså at `Timer` med regulære intervaller presenterer neste bokstav, kommando, ord, osv. Hva som presenteres neste gang er avhengig av hvilken controller som kjører, og av tilstanden til denne controlleren. Når brukeren gjør et valg, vil `BevegelsesSensor` kalle `velg()` i controlleren.

Diagrammet i figur 2 er ment å illustrere oppstarten av et skriv melding-usecase. Det vil være mange `presenterNeste()`-meldinger her, mange, mange flere enn illustrert i diagrammet. Controlleren `HovedMeny` blir erstattet av controlleren `SkrivMelding` når brukeren har valgt dette. I slutten av vil controlleren `SkrivAdresse` bli skapt og startet opp, slik at en adresse kan bli skrevet og meldingen kan bli sendt.

Figur 3 viser en skisse til et klassediagram for systemet.

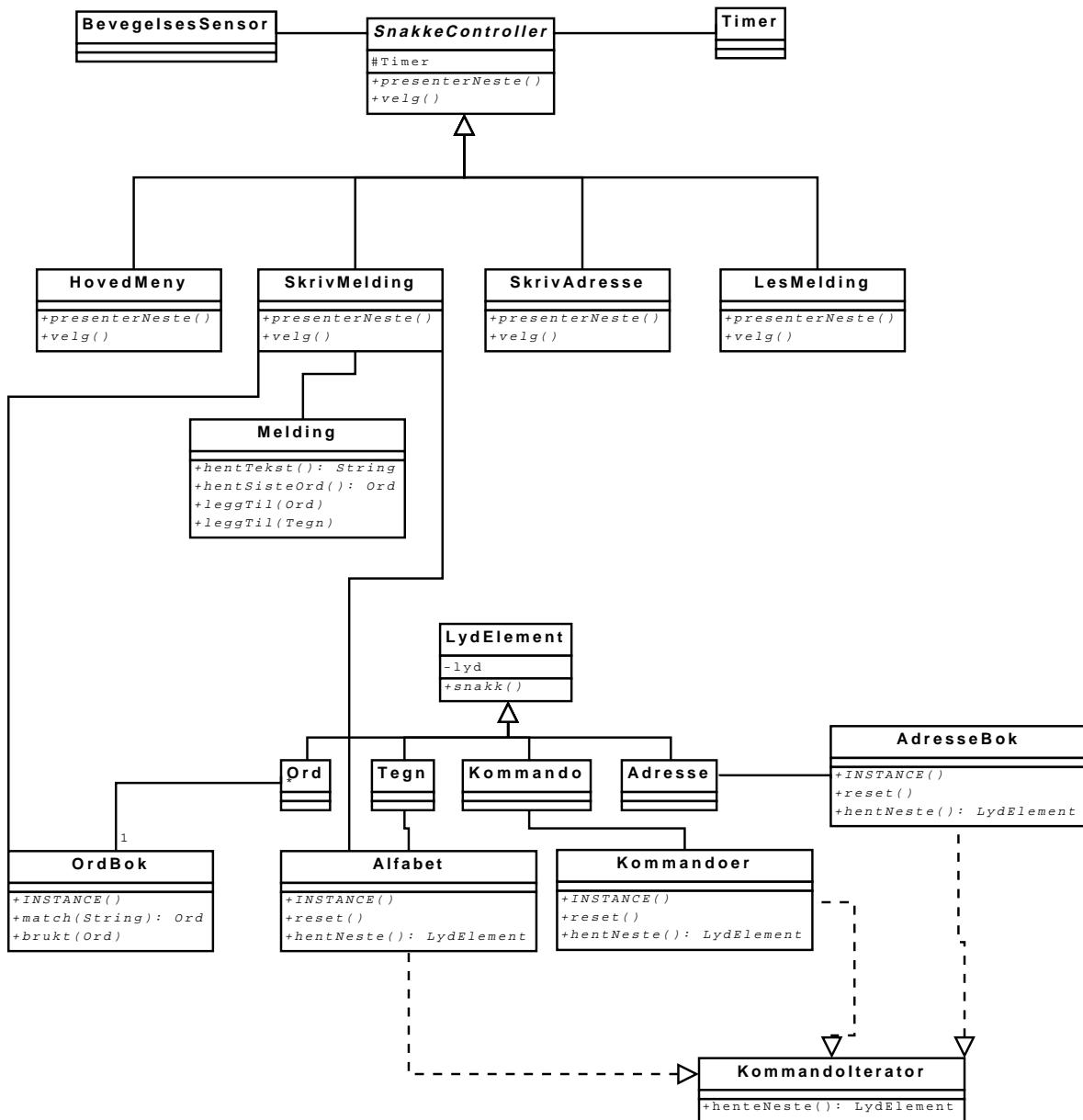
Under har vi vist en skisse til noe av innholdet i `SkrivMeldings` to viktigste metoder. De andre controllerne vil ha noe tilsvarende.

```

SkrivMelding.presenterNeste() {
    if (antallTegnSkrevet >= 2 && !foreslåttOrd) {
        sistPresentert = OrdBok.INSTANCE().match(melding.hentTekst());
        foreslåttOrd = true;
    } else {
        sistPresentert = Alfabet.INSTANCE().hentNeste();
    }
    sistPresentert.snakk();
}
  
```

```

SkrivMelding.velg() {
  
```



Figur 3: Klassediagram for SnakkeControllere

```
if (sistPresentert == ' ' || sistPresentert=='.') {
    OrdBok.INSTANCE().brukt(melding.hentSisteOrd());
    antallTegnSkrevet = 0;
} else {
    antallTegnSkrevet += sistPresentert.antallTegn();
}
melding.leggTil(sistPresentert);
foreslåttOrd = false;
Alfabet.INSTANCE().reset();
}
```