

Emergency call button

Stabilt og simpelt

Agenda

- Områder af speciel interesse
- Gennemgang
- Hvad har jeg lært?
- Spørgsmål

Områder af speciel interesse

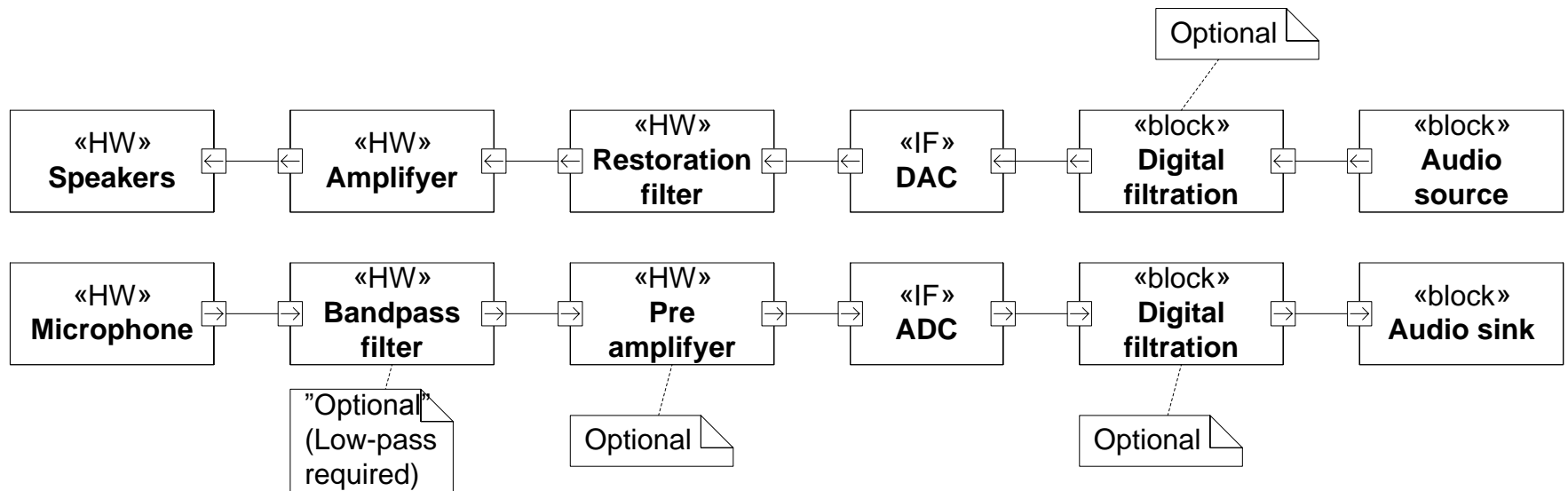
- Domæne, Krav, Use Cases, Kvalitetsattributter
- Arkitektur (SysML)
- SystemC
 - TLM
 - TTLM
- Mapping af arkitektur
 - Design space exploration
 - LBA/LPT

Kvalitetsattributter

- Undervurderet i INCOSE (fokus på format)
- Anvendes både til krav, arkitektur design og test.
- Meget ofte modstridende
- Risiko, kompleksitet og fleksibilitet som kvalitetsattributter.

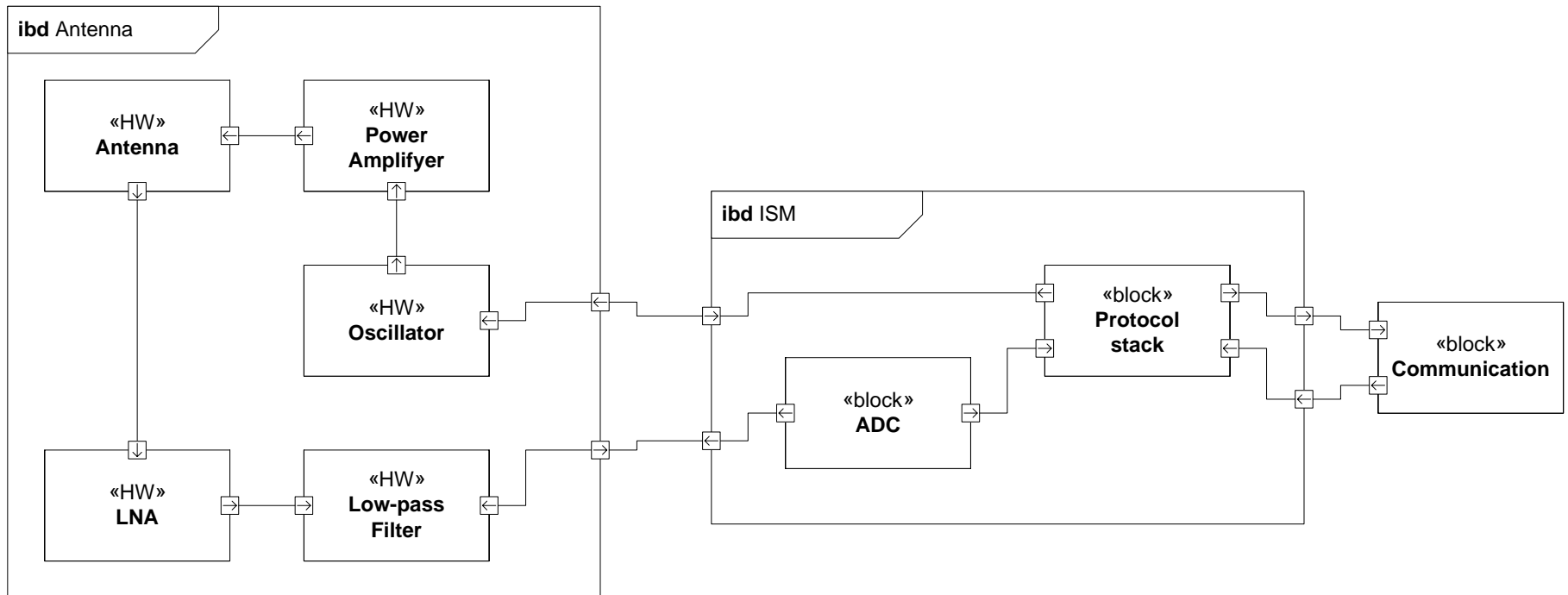
Arkitektur (struktur)

- IBD/BDD allerede vist
- SysML til andet end IBD/BDD (fælles sprog)



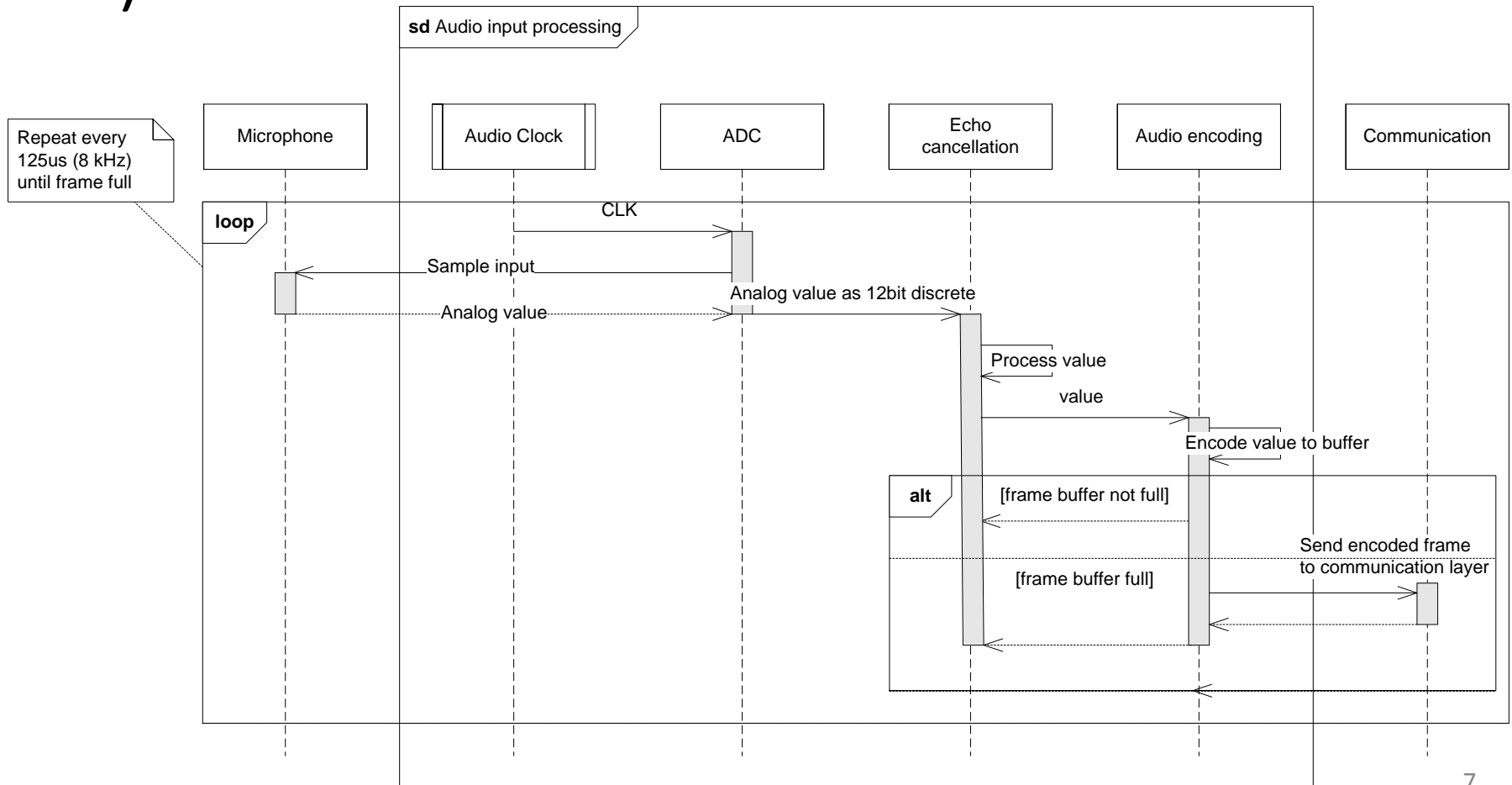
IBD ISM/Antenna

- Deling af ansvar



Arkitektur (timing)

- Andre SysML diagrammer (Sekvens, Aktivitet, ...)



SysML mangler

- Krav diagrammer svære at overskue uden tool-support (EA)
- Timing. Noget timing er lavet i sekvens diagram, men MARTE har stereotyper til det.

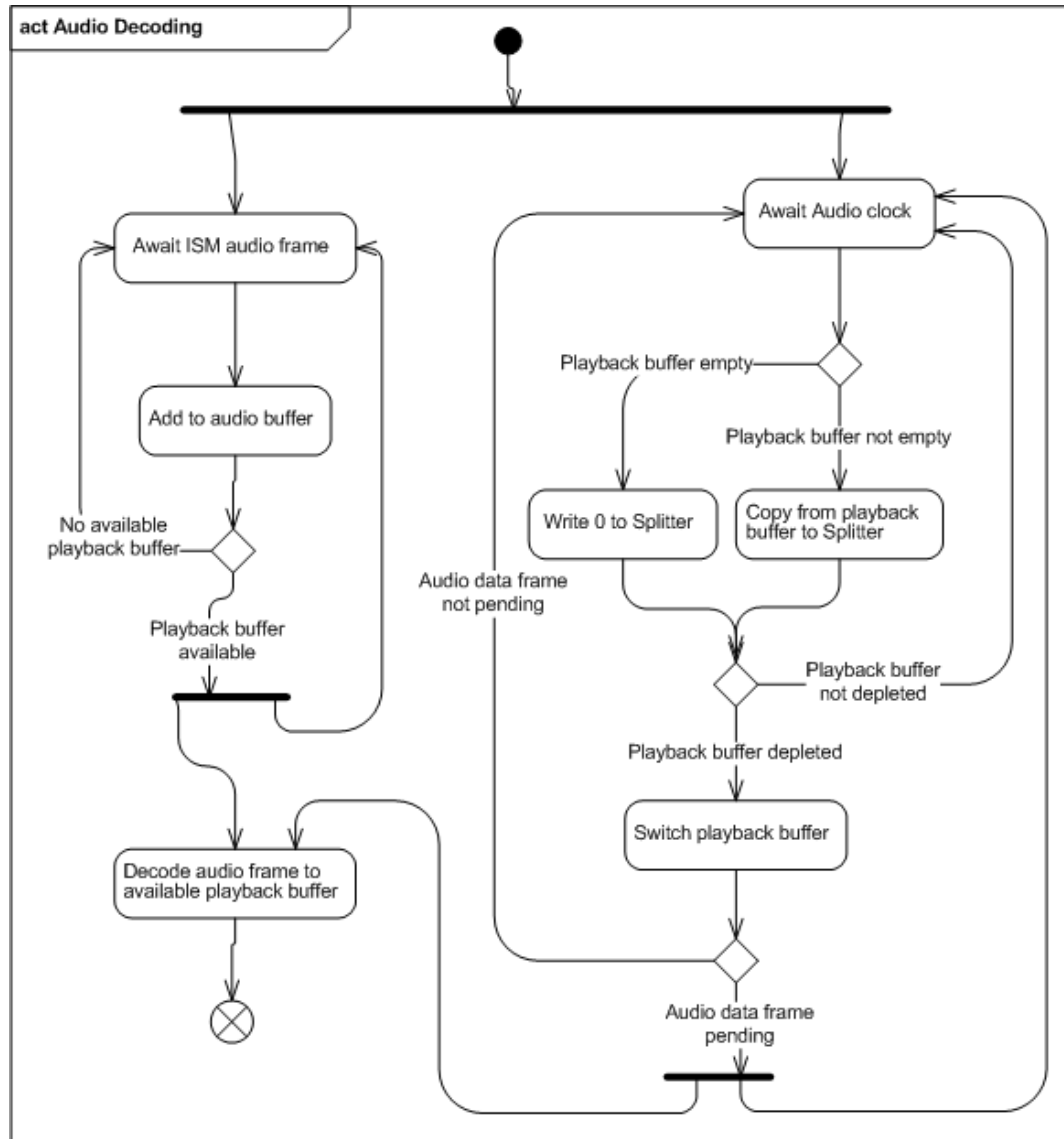
SystemC (1)

- TLM ved stor usikkerhed om algoritmer, funktionalitet eller arkitektur
- TTLM ved verificering af timing, evt. med ISS
- BCAM/CCAM/CAM/RTL ved simulering af HW
- Eclipse er en kendt IDE, men autogenerering skal bruges hvis tiden skal kunne forsvares for andet end BCAM/CCAM/CAM/RTL
- Når man har opbygget en modul DB bliver det nemmere

SystemC (2)

- I projektet kunne der have været lavet aktivitetsdiagrammer for at gøre SysML -> SystemC nemmere (aktivitet -> en eller flere Threads). – se næste side.

Aktivitetsdiagram



Mapning af arkitektur

- Design space exploration er mindst struktureret, men nemmest at tweake til et givent sæt kvalitetsattributter.
- LPT/LBA er ikke modent til at forlade det akademiske domæne, med mindre platformen er lagt fast.

Hvad har jeg lært? (1)

- Praktisk erfaring med SysML
 - Det vil jeg bruge (minus requirements)
- Praktisk erfaring med SystemC
 - Det vil jeg ikke bruge med mindre har behov for at simulere Custom HW som ikke er til rådighed.
- Praktisk erfaring med mapning
 - Det bruger jeg allerede, men nu har jeg bedre forståelse for teorien bag.

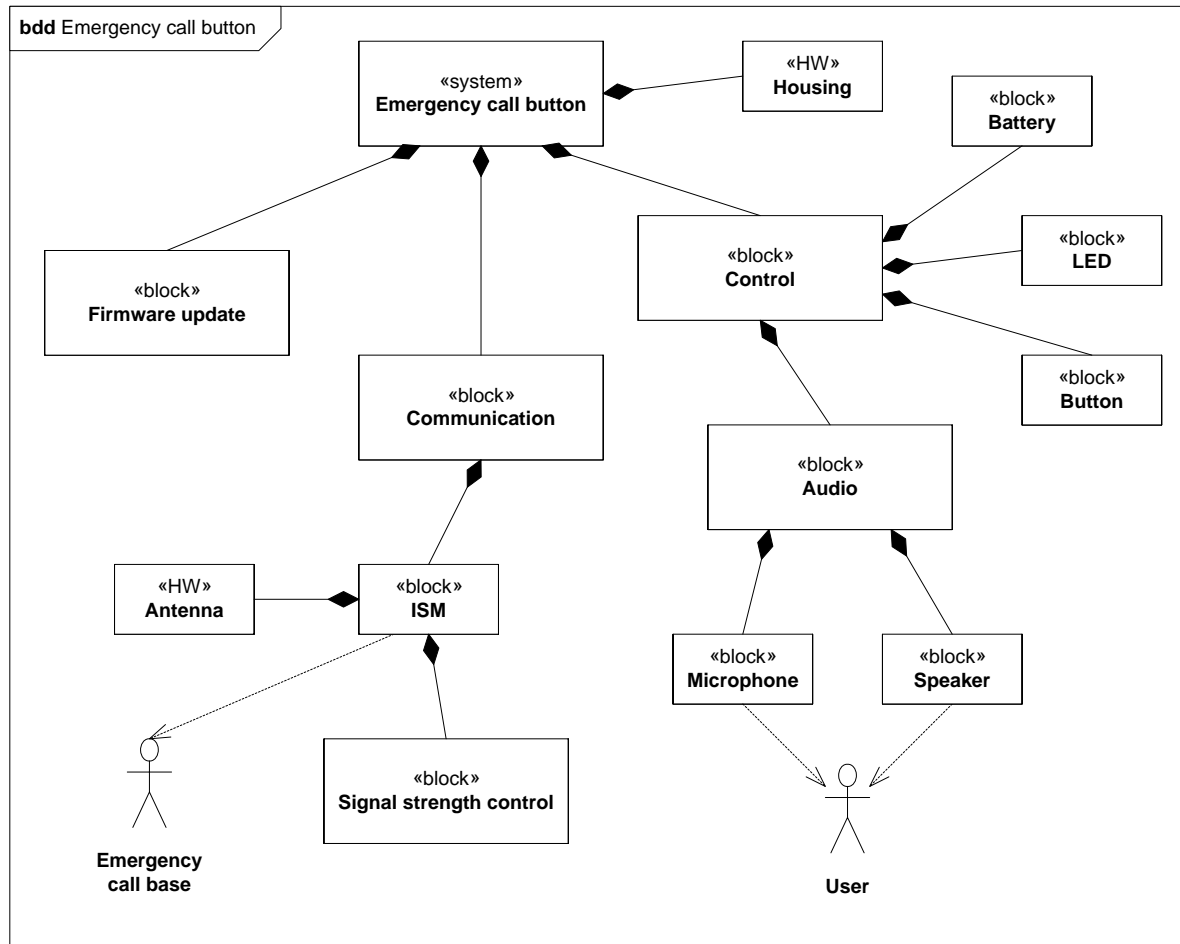
Hvad har jeg lært? (2)

- Jeg har prøvet at bruge top-down (HW/SW Co-design) til flere projekter, men kun hvis en eksisterende platform ikke findes. At kunne genbruge en eksisterende platform er både nemmere, billigere og meget sikrere (risiko).
- En mindre slavisk tilgang til INCOSE (modsat modul 1) gør det mere brugbart.

Spørgsmål

- ?

BDD



IBD

