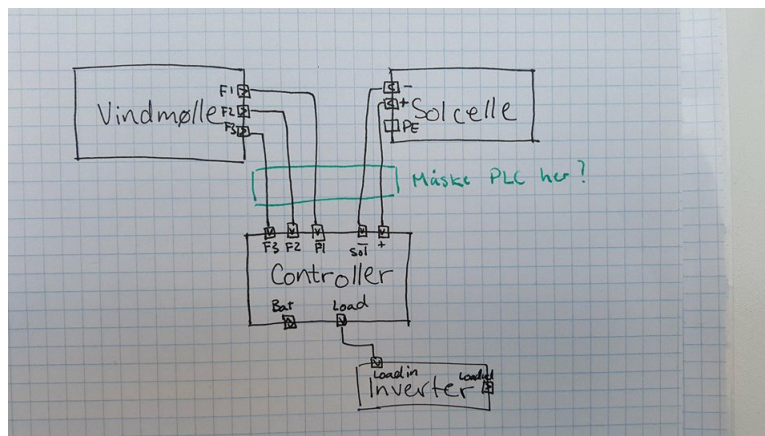


Figur 1: BDD for Nul-energi hus

Her ses det overordnede IBD for Nul-energi huset. Nul-energi huset består af en vindmølle og solcelle, som producerer energien til huset. Derudover består den af et batteri, hvor energien kan lagres, samt en backup, hvis der ikke produceres nok energi. For at kunne overvåge vindmøllen, solcellen og batteriet, er der behov for en styreenhed. Denne styreenhed består af en controller og en PLC, hvor data fra sensorerne bliver behandlet.



Figur 2: Første udkast til IBD.

Dette er det overordnede IBD for Nul-energi huset. (men mangler batteri g PLC) På det overordnede IBD kan man se forbindelser mellem de forskellige elementer, samt om det er input, output eller in/out. Vindmøllen består af 3 faser, som er output, og solcellen har positiv og negativ output, samt en PE. Disse forbindelser går gennem en controller.

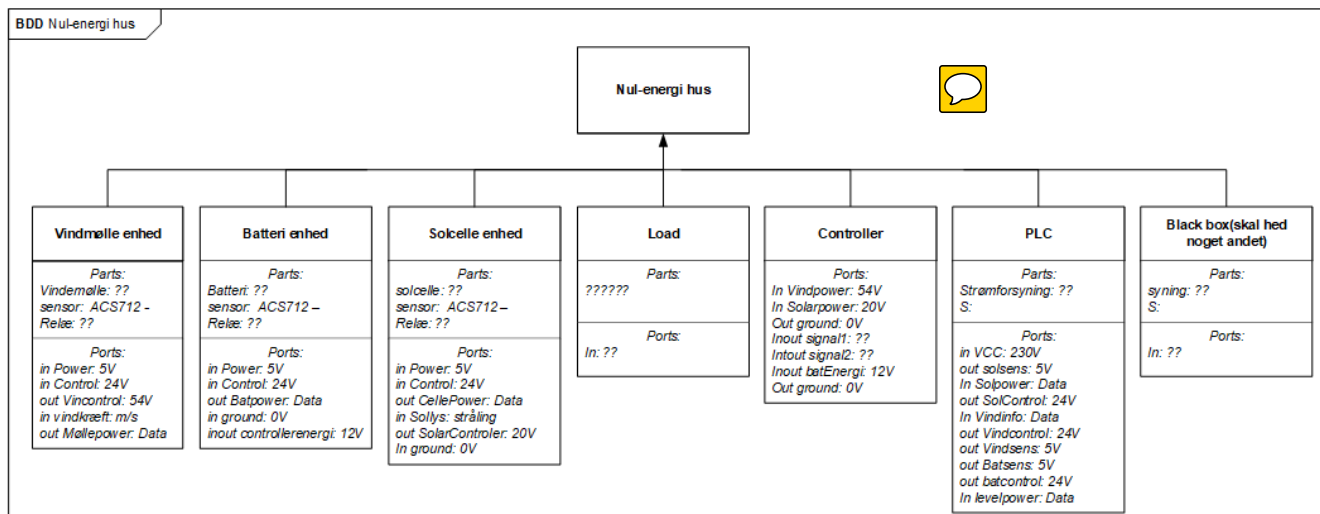


1 Hardware Arkitektur

Først anvendes BDD til at sætte systemet op som blokke, hvor det efterfølgende kan brydes ned i mindre blokke, indtil der opnåes tilstrækkelig overskuelighed. Hvert af disse blokke får en kort blokbeskrivelse. Til at illustrere hvordan blokkene er forbundet til hinanden, samt hvilke signaler der sendes mellem blokkene. Her anvendes IBD samt en signal beskrivelses **tabel**, der fortæller signalerne med ord.

1.1 Block Definition Diagram

Block **definition** gram eller BDD, angiver hvilke blokke et system består af. Blokkene er karakteriseret ved at beside lav kobling og høj intern samhørighed.



Figur 3: BDD overordnet system

Block Beskrivelse Nul-energi hus

Det beskrevet system indeholder forskellige blokke. Disse blokke har forskellige funktioner, der udføres for systemet. BDD'et der er vist på figur 3, og er den første iteration af systemet. Nedenunder kan man se de beskrevet hardwareblokke som systemet indeholder.

Nul-energi hus

Denne blok er hovedblokken, der indeholder inputs og outputs der går til og fra systemet udefra, dette ses fra brugerens side.

Vindmølle

Vindmølle blokkens funktion er at generere strøm vha. de ydre omgivelser, som i dette tilfælde er vind. Der forventes at den kan afgive en spænding på op til 54V. Denne arbejde også tæt sammen med både controlleren og PLC'en. Vindmøllen generere en AC spænding, som skal ændres til DC.




Batteri

Denne blok agere nødgenerator for systemet, og skal forsyne systemet i tilfælde af der skulle mange strøm. Den afgiver en spænding på 12 V DC.


Solcelle

Solcellen er på samme måde som Vindmøllen, en strømkilde, der generer strøm vha. ydre omgivelser, som i dette tilfælde er sol energi. Den generere op til 20V DC, og er også i tæt samarbejde med både controlleren og PLC'en.


Load

Load'en er den blok der skal agere som  nul-energi-husets forbrug. Dvs at den skal fungere  som det der statistisk set  svare til det gennemsnitlige forbrug.


Controller

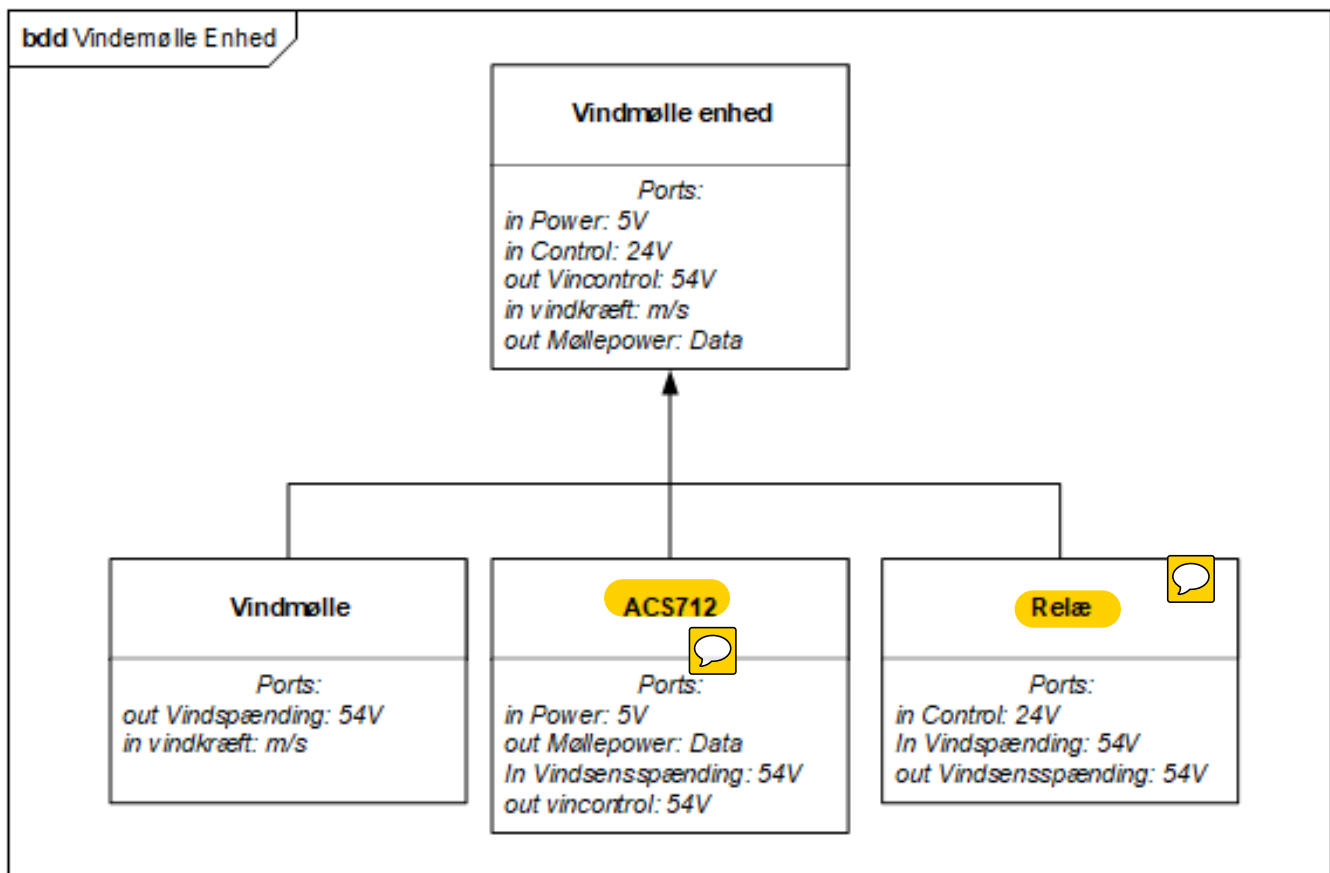
Controller blokken, fungere på den måde at den kopler både vindmøllen, batteriet og solcellen sammen. Den giver derfor et samlet output som kan variere alt efter den energi der genereres. 

PLC

PLC blokken er den enhed der står for at styre de 3 enheder, vindmølle, solcelle og batteri. Den fungere også på samme måde som en GUI, da det er en indbygget funktion der er i den. Desuden skal den kunne åbne og lukke for den genererede spændning og grided alt efter behov. 

Observer

 Observeren har til formål at forbinde load outputtet, grid og controller outputtet med PLC'en, således at den kan åbne og lukke de forskellige kilder efter behov.



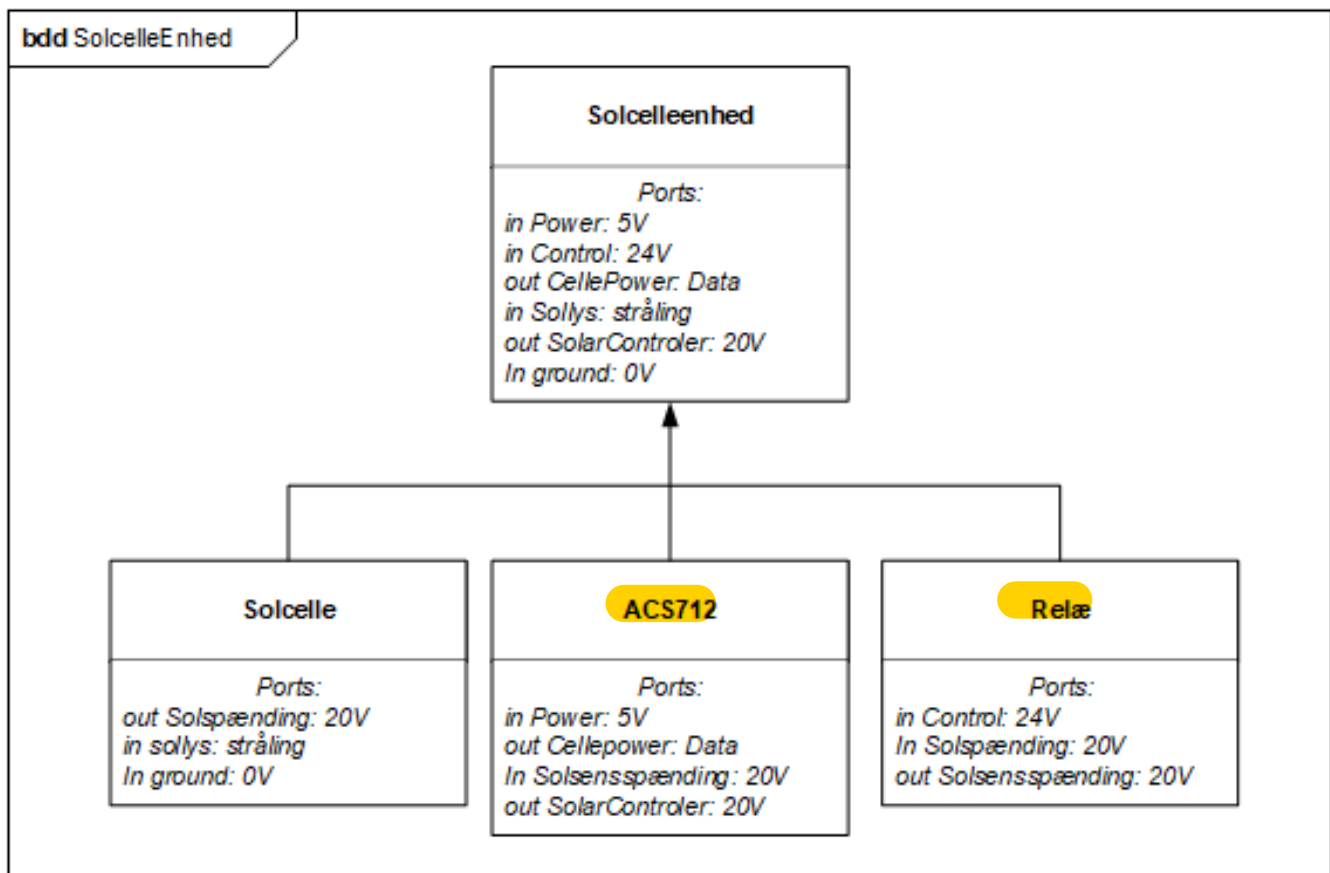
Figur 4: BDD Vindmølle

Block Beskrivelser Vindmølle

Vindmølle

ACS712

Relæ



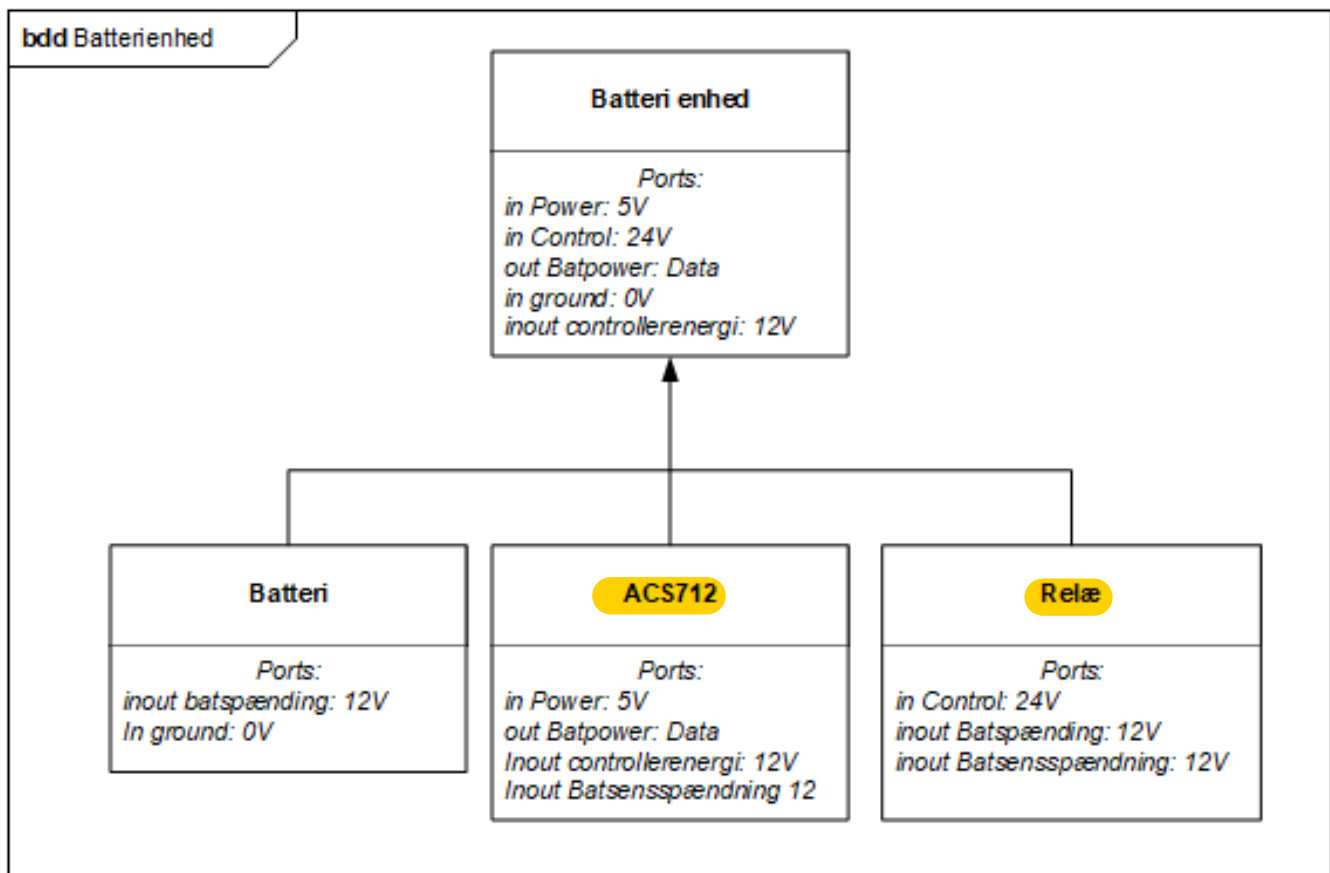
Figur 5: BDD Solcelle

Block Beskrivelser Solcelle

Solcelle

ACs712

Relæ



Figur 6: BDD Batteri

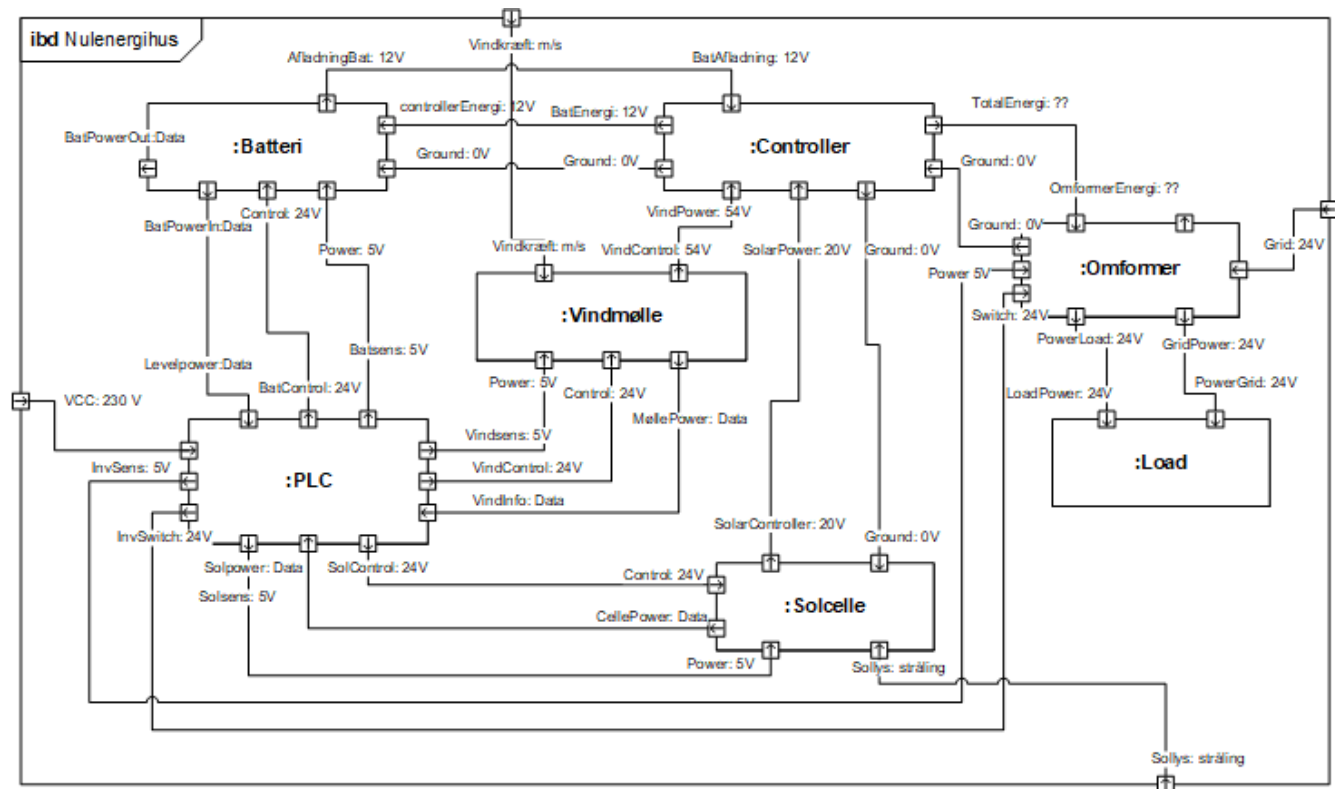
Block Beskrivelser Batteri

Batteri

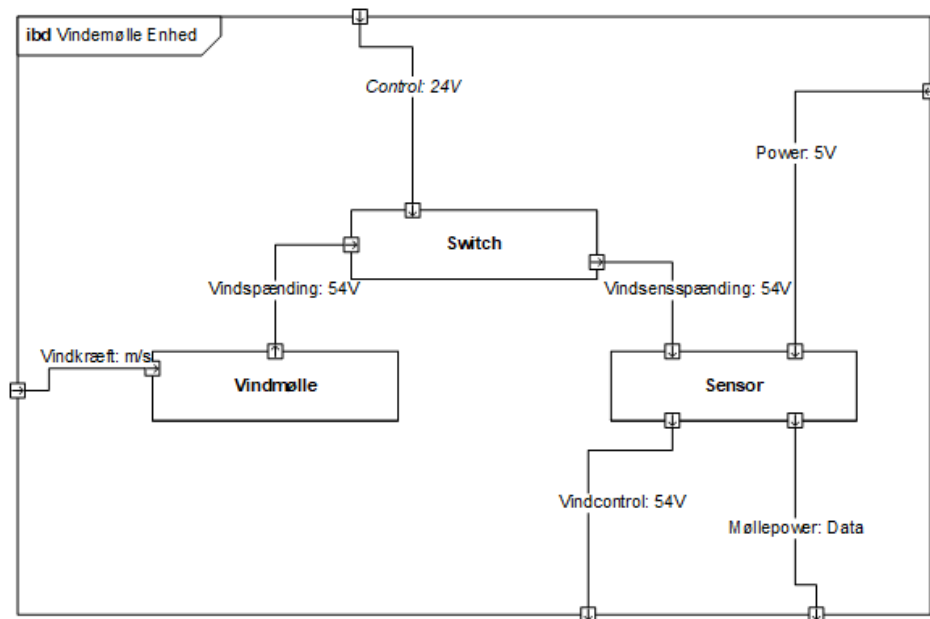
ACS712

Relæ

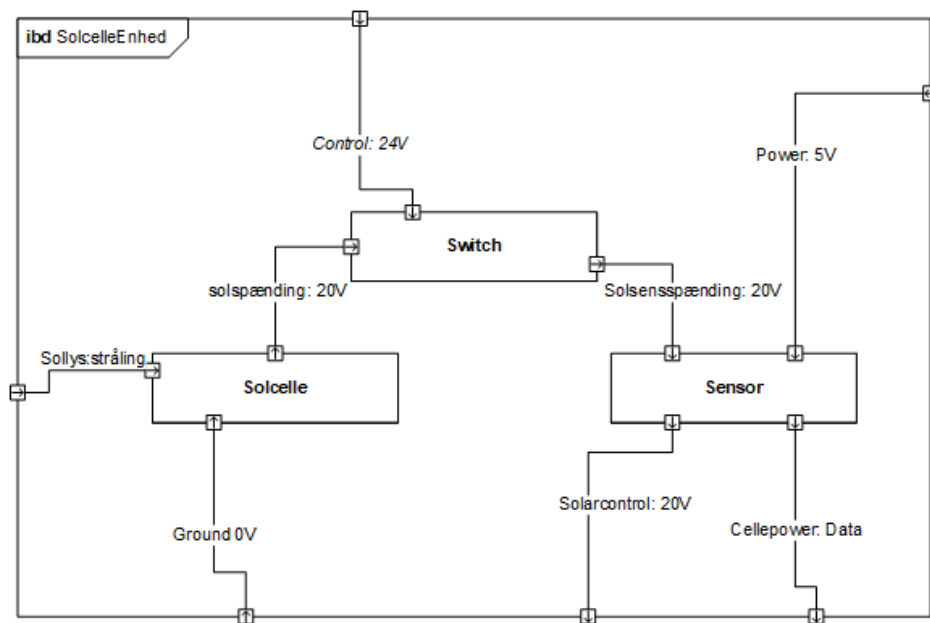
1.2 Internal Block Diagram



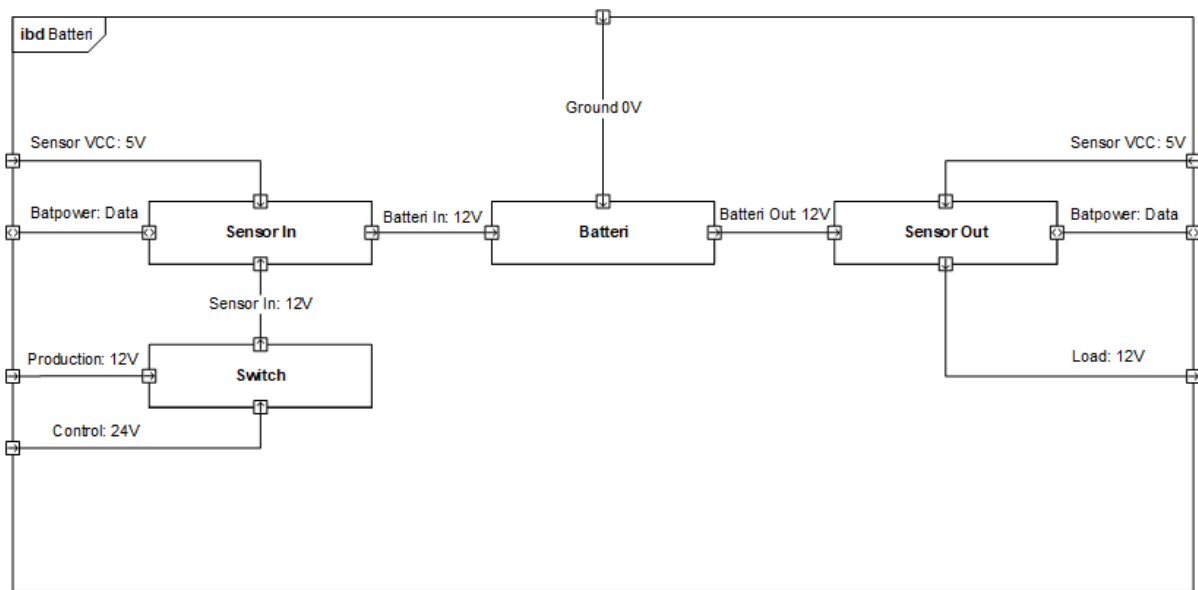
Figur 7: IBD af overordnet system



Figur 8: IBD vindmølle



Figur 9: IBD solcelle



Figur 10: IBD batteri

1.3 Signalbeskrivelse

| Load | | | | |
|----------------------|----------|--------|------------|-------------------|
| Funktionsbeskrivelse | Signaler | IN/OUT | Signalnavn | Signalbeskrivelse |
| | | | | |
| | | | | |
| | | | | |

Tabel 1:

| Grid | | | | |
|----------------------|----------|--------|------------|-------------------|
| Funktionsbeskrivelse | Signaler | IN/OUT | Signalnavn | Signalbeskrivelse |
| | | | | |
| | | | | |
| | | | | |

Tabel 2:

| Observer | | | | |
|----------------------|----------|--------|------------|-------------------|
| Funktionsbeskrivelse | Signaler | IN/OUT | Signalnavn | Signalbeskrivelse |
| | | | | |
| | | | | |
| | | | | |

Tabel 3:

| Controller | | | | |
|----------------------|----------|--------|------------|-------------------|
| Funktionsbeskrivelse | Signaler | IN/OUT | Signalnavn | Signalbeskrivelse |
| | | | | |
| | | | | |
| | | | | |

Tabel 4:

| PLC | | | | |
|----------------------|----------|--------|------------|-------------------|
| Funktionsbeskrivelse | Signaler | IN/OUT | Signalnavn | Signalbeskrivelse |
| | | | | |
| | | | | |
| | | | | |

Tabel 5:

| Solcelle | | | | |
|----------------------|----------|--------|------------|-------------------|
| Funktionsbeskrivelse | Signaler | IN/OUT | Signalnavn | Signalbeskrivelse |
| | | | | |
| | | | | |
| | | | | |

Tabel 6:

| Vindmølle | | | | |
|----------------------|----------|--------|------------|-------------------|
| Funktionsbeskrivelse | Signaler | IN/OUT | Signalnavn | Signalbeskrivelse |
| | | | | |
| | | | | |
| | | | | |

Tabel 7:

| Batteri | | | | |
|----------------------|----------|--------|------------|-------------------|
| Funktionsbeskrivelse | Signaler | IN/OUT | Signalnavn | Signalbeskrivelse |
| | | | | |
| | | | | |
| | | | | |

Tabel 8:

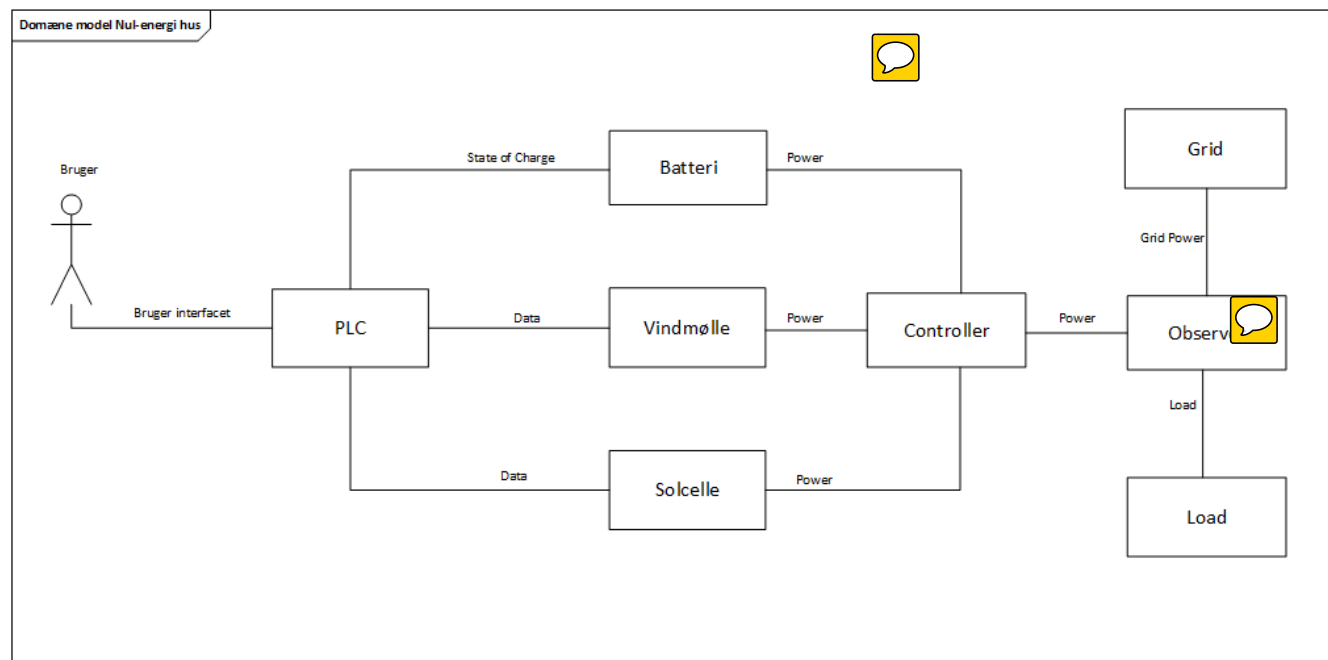
2 Software arkitektur

Ud fra Nul-energi husets **use cases** udledes konceptuelle klasser, der danner grundlagt for domænemodellen. ud fra domænemodellen, udarbejdes sekvensdiagrammer for hver use case.



2.1 Domæne Model

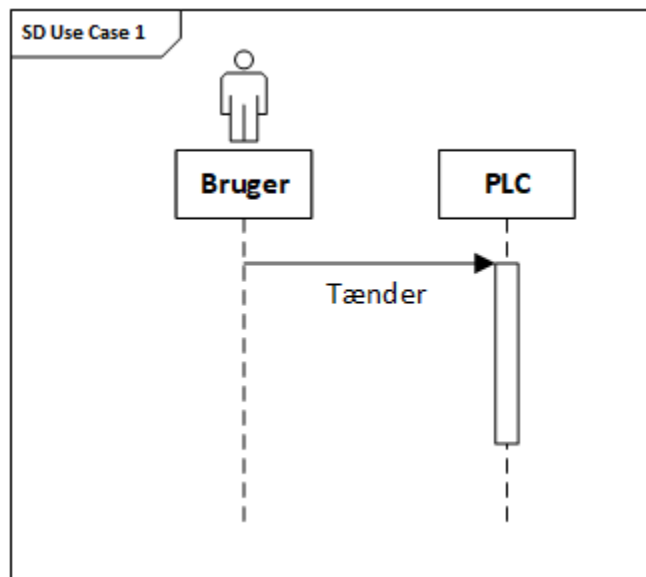
Domænemodellen illustrerer, hvordan de konceptuelle klasser overordnet interagerer med hinanden og bruges til at danne et ide mæssigt overblik over systemet. De konceptuelle klasser udledes ved at identificere navneordene ud fra fully-dressed **use cases**. Det kan være nødvendigt at tilføje ekstra klasser for at varetagesystemets funktioner.



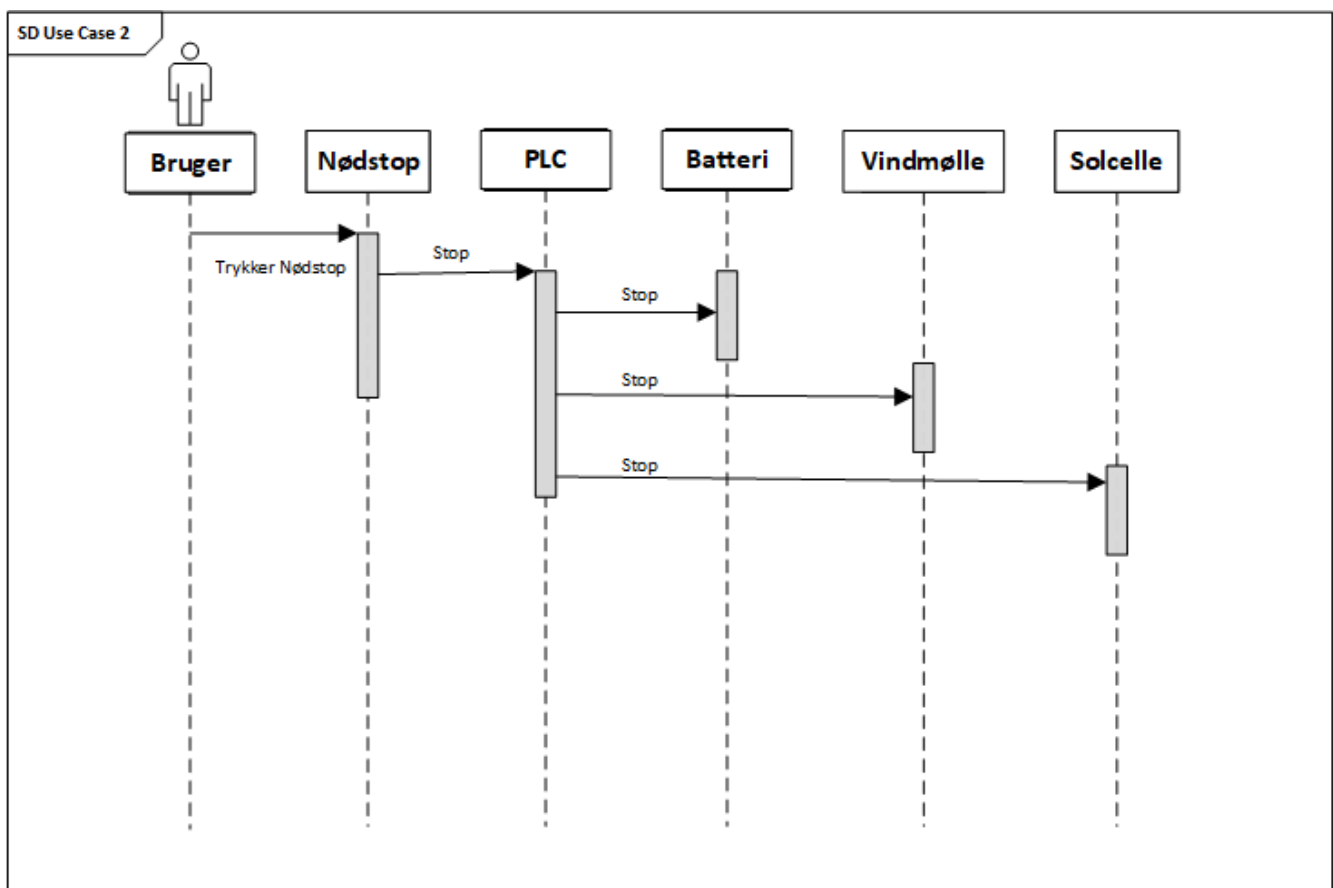
Figur 11: Domæne model

2.2 Sekvens diagrammer

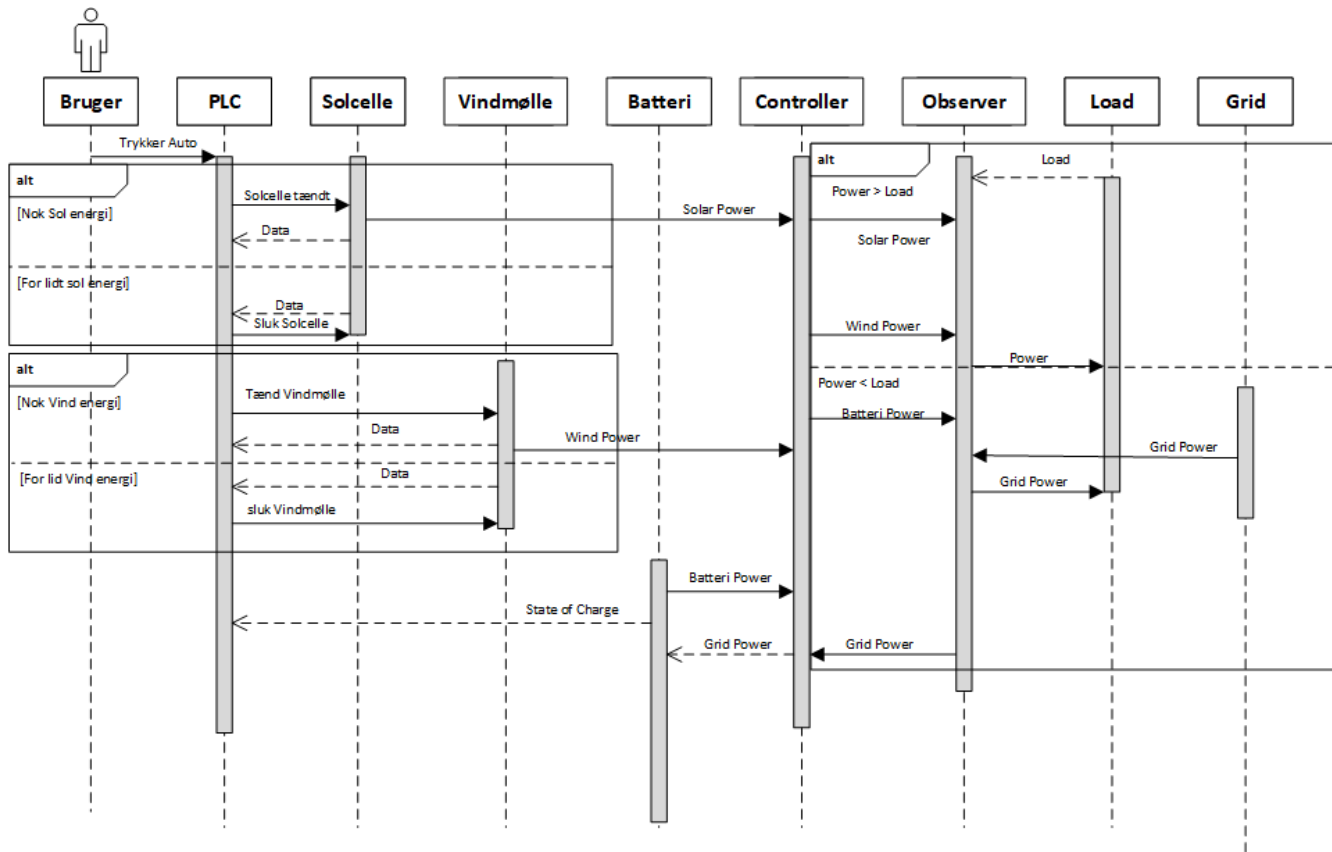
Sekvensdiagram illustrerer handlingsforløbet gennem systemet for hver **use case** mellem hver konceptuel blok fra domænemodellen i en story-line. Handlingsforløbet tager sit udgangspunkt i fully dressed for hver **use case**.



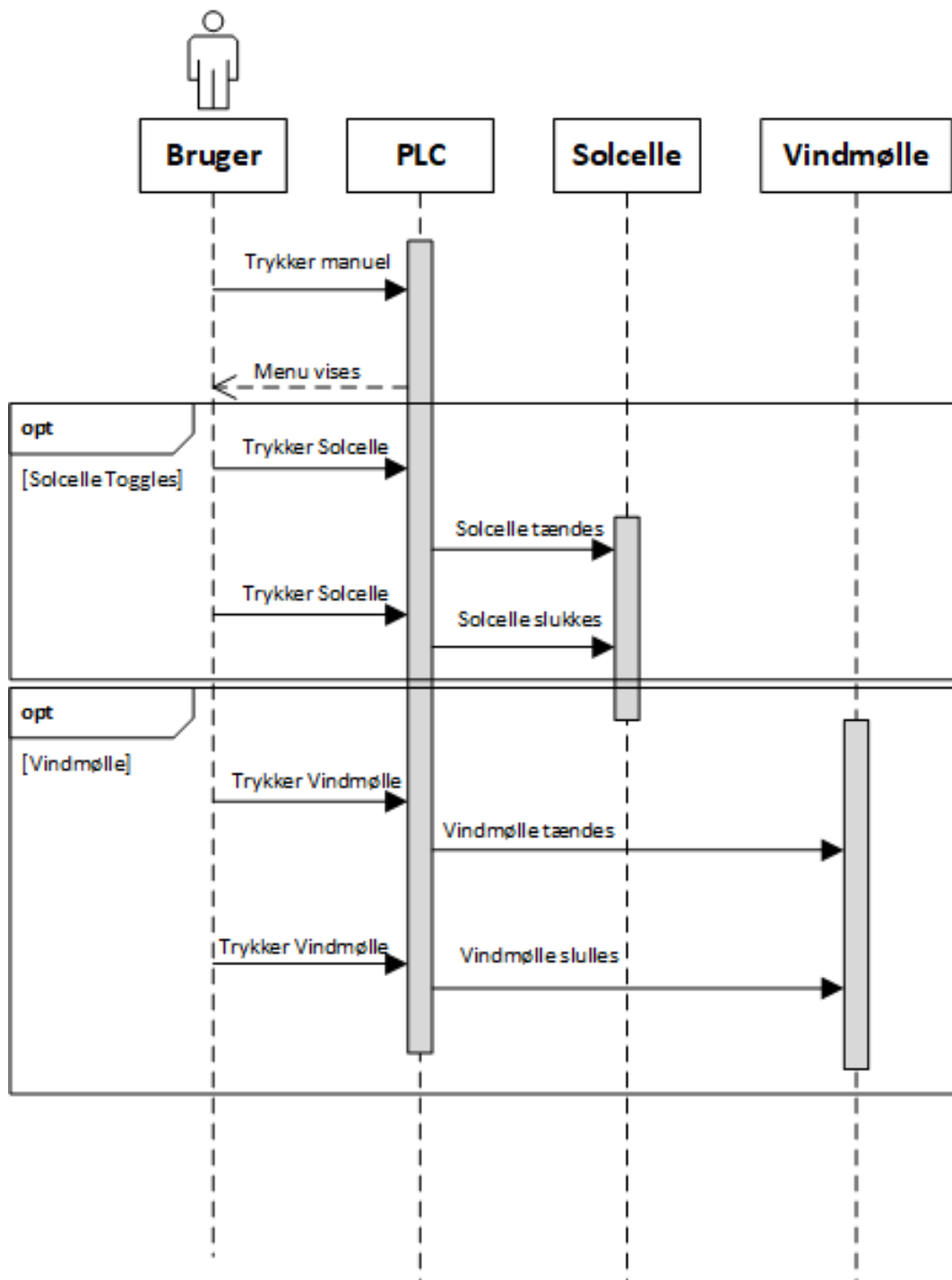
Figur 12: Sekvense digram af use case 1



Figur 13: Sekvense digram af use case 2



Figur 14: Sekvense digram af use case 3



Figur 15: Sekvense digram af use case 4