Arduino Rapport

Indholdsfortegnelse

[Design 3](#_Toc90241843)

[Flowchart 3](#_Toc90241844)

[Pseudokode 3](#_Toc90241845)

[Scrum (tidsplan, trello & evt. vandfald?) 5](#_Toc90241846)

[Dokumentation 6](#_Toc90241847)

[Code kommentar 6](#_Toc90241848)

[code style (code beautifier) 6](#_Toc90241849)

[funktion beskrivelser 6](#_Toc90241850)

[class beskrivelse og diagram 6](#_Toc90241851)

[deling 7](#_Toc90241852)

[github 7](#_Toc90241853)

# Design

## Flowchart

Figur 1. Flowchart af projekt.

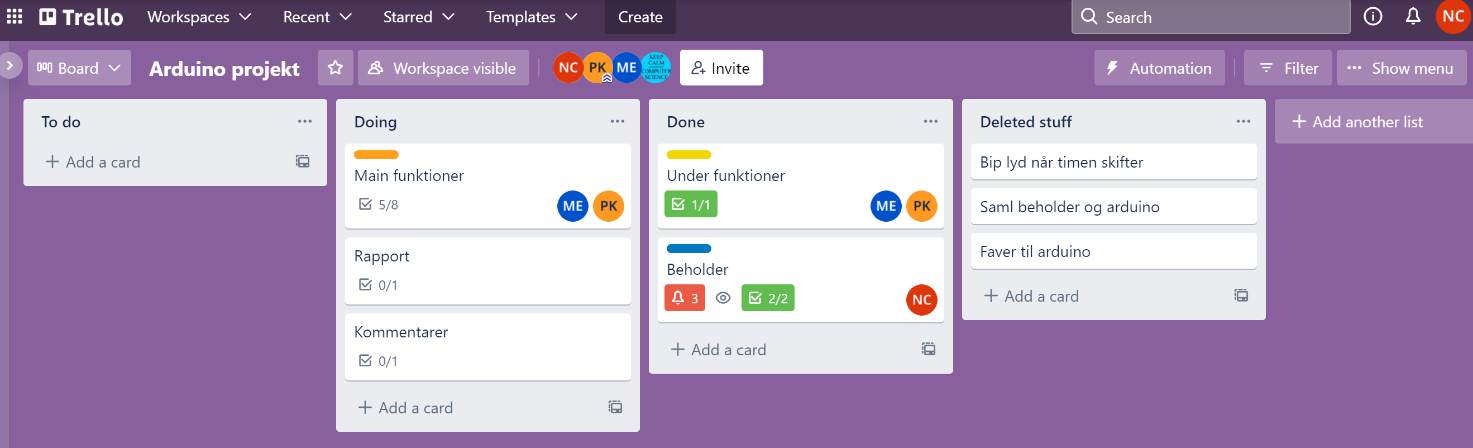
## Pseudokode

* Definer variabler
* Definer skærm
* Henter værdier fra vores peripherals
* Kalder funktioner
* Funktion for stadieX  
   funktion tager mod værdi, som er lig antal ønsket mulige stadier  
   tager værdier fra 3-Axis Digital  
   if (x bliver vendt til den ene side)  
   læg 1 til stadiet  
   if (x bliver vendt til den anden side)  
   træk 1 fra stadiet  
   returner stadiet for x-aksen
* Funktion for stadieY  
   funktion tager mod værdi, som er lig antal ønsket mulige stadier  
   tager værdier fra 3-Axis Digital  
   if (y bliver vendt til den ene side)  
   læg 1 til stadiet  
   if (y bliver vendt til den anden side)  
   træk 1 fra stadiet  
   returner stadiet for y-aksen
* Funktion for Ur  
   funktion beregner tid med Clock  
   funktion beregner ugedagen  
   funktion skriver fødselsdage  
   kalder funktion for at ringe til time  
   kalder funktion for juledag
* Funktion for Julekalender  
   funktion tager mod værdi, som er lig dage til juleaften  
    
  if (måned er lig december)  
   if (datoen ikke er over d. 23)  
   skriv dage tilbage til juleaften på skærm  
  if (måned er lig december)  
   if (datoen er d. 24)  
   skriv ’god jul’ på skærmen  
  if (måned er ikke lig december)  
   skriv ’jul kommer igen’
* Funktion for temperatur  
   funktion tager mod temperatur fra sensor  
   omregner temperatur til celsius  
   skriver temperaturen på skærmen
* Funktion for spillet  
   lav random variable  
   if (sensorValue fra analogRotation er inden i et interval rundt om random-værdien)  
   skift farve på skærm  
   kald funktion for knap  
    
   if (sensorValue fra analogRotation er inden i et mindre interval rundt om random-værdien)  
   if (knap er nede)  
   skærm lyser grøn  
    
   if (sensorValue fra analogRotation ikke er inden i et mindre interval rundt om random-værdien)  
   if (knap er nede)  
   skærm lyser rød  
     
   if (sensorValue fra analogRotation er uden for det store interval rundt om random-værdien)  
   skift farve på skærm tilbage til start  
  if (sensorValue fra analogRotation lig 0)  
   lav ny random-værdi
* Funktion for random elev  
   lav random variable der går til antal elever  
   skriv elevnavnet der kolorerer til random-værdien

## Scrum (tidsplan, trello & evt. vandfald?)

Her er vores tidsplan over Arduino-projektet lavet med Trello.

<https://trello.com/invite/b/FYMIeLYg/770eca2ba8c8c2266ab65841ed233e32/arduino-projekt>

Figur 2. Billede af vores tidplan fra Trello.

# Dokumentation

## Code kommentar

Et billede, der indeholder tekst

Automatisk genereret beskrivelseTil venstre er et eksempel på, hvordan vi har brugt kommentarer i dette projekt til at informerer om, hvad de forskellige kommandoer gør. Når man samarbejder på et programmeringsprojekt, er det ofte forvirrende og måske endda kaotisk at se på et stykke kode for første gang som en anden har person har skrevet. Det er derfor vi bruger kommentarer.

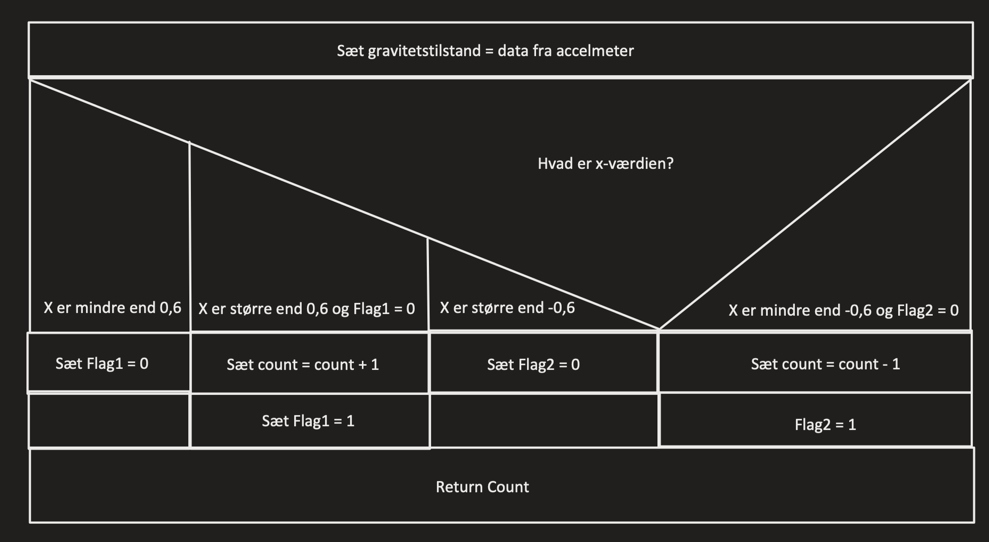
I den første linje på billedet ser man ude til højre en kommentar med bogstaverne MNE det er initialerne på den, der har skrevet det stykke kode, så man altid ved, hvem man skal henvende sig til hvis man har nogen spørgsmål.

Figur 3. Funktion for temperatur fra vores program.

Derudover er der gennem vores kode flere steder, hvor vi har brugt kommentarerne til at forklarer hvad bestemte linjer eller funktioner gør, og hvilken rolle de spiller i vores projekt. De forklarer nogle gange konkret hvad en linje bestemmer, men andre gange giver de bare et overblik over et interval eller forskellige variable som har betydning for udkommet.

Til sidst har vi også nogle enkelte steder udkommenteret flere linjer kode, fordi vi ikke skulle bruge dem i det øjeblik, men de kunne være gode at have senere, hvis man for eksempel ville ændre noget.

## funktion beskrivelser

Vi har lavet flere funktioner under dette forløb. En af dem var vores gravityFunktion, som skulle holde styr på hvilket stadie vores kasse er i.

Figur 4. Nassi Schneider funktion diagram over gravityFunktion.

# deling

## github

Som værktøj til at dele vores arbejde, har vi brugt Github. VI lavede et repository og delte det med hele gruppen, og vores lærer, Mirsad. Med Github er det nemt at lave version control, hvilket også er rigtig godt til vores software udviklingsproces. Det tillader os at sammarbejde på ét projekt samtidig og

Et billede, der indeholder tekst, monitor, skærmbillede, skærm

Automatisk genereret beskrivelse