

# Eksamen Embedded Systems

## Treat-box

I min løsning ville jeg lage en form for gadget. Jeg ville også lage noe som hadde med hunden min å gjøre, så løsningen ble til en app-styrt godbit-dispenser.

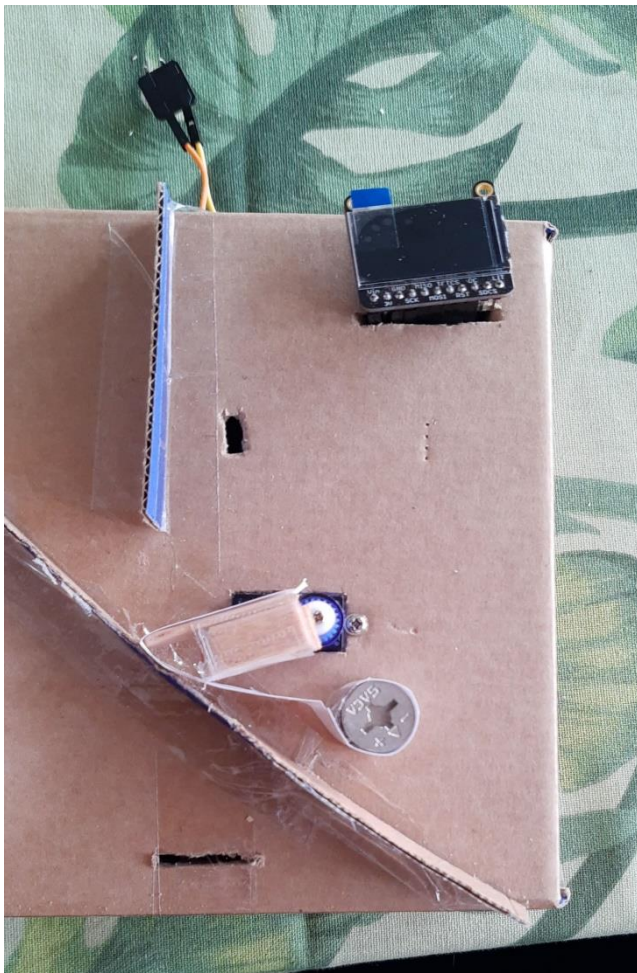
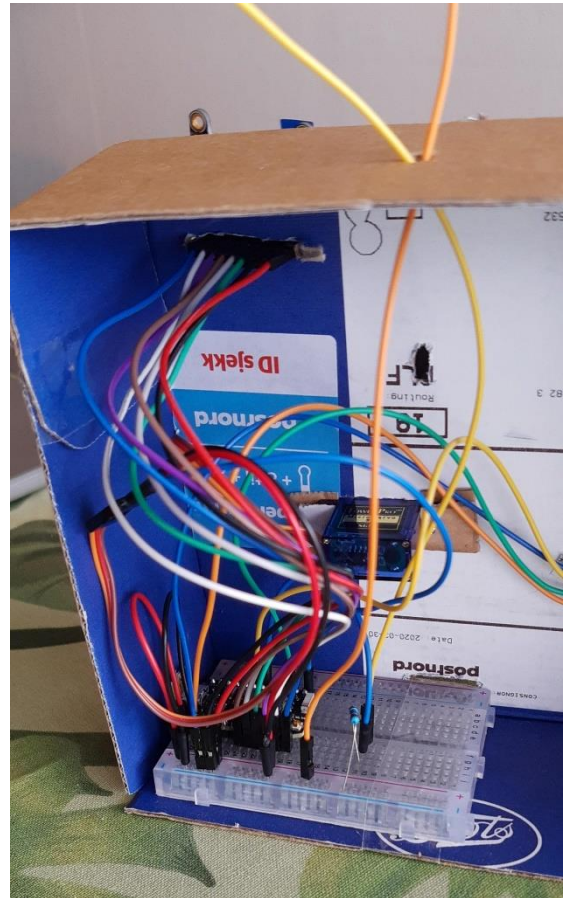
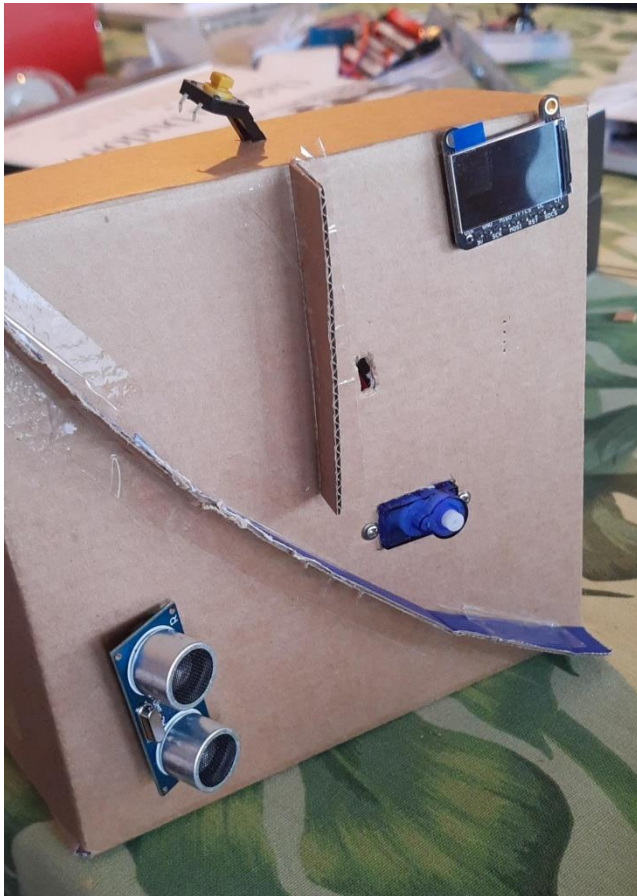
Jeg bruker en servo-arm til å dytte en godbit ut av en boks. På boksen har jeg montert en TFT-skjerm som viser hvor mange godbiter som er igjen i boksen. Antall godbiter som er i boksen kan økes med en knapp montert på toppen av boksen, og vil automatisk gå ned hver gang man trykker på appen for å gi ut godbit. Det er også montert på en Ultrasonisk avstandsmåler på siden av boksen som sender et varsel hvis den oppdager bevegelse i nærheten av boksen, slik at man vet når kjæledyret vil ha en godbit.

Jeg satte sammen hele boksen med kun teip og papp. Den består av en boks som huser alle komponentene, slik at den er enkel å ta ut og inn av en større boks for å gjøre modifikasjoner. Jeg så for meg mange måter å løse oppgaven på. Med mitt oppsett er det enkelt å gi ut godbiten på mange forskjellige måter enn kun å putte det i en boks. Noen tanker jeg hadde var for eksempel å lage en katapult-lignende sak, eller ha et opplegg der servo-armen slår godbiten som en golfkølle. Jeg valgte en løftemekanisme med en sklie kun fordi det virket som den mest utfordrende, men likevel gøy i løsningen for min del, men muligheten for å ta de komponentene jeg brukte til å lage andre løsninger, er mange. Jeg var inne på tanken på å bruke en steppermotor, men en servo ville gjøre jobben like greit; samt at den er lettere og tar mindre plass. Med en steppermotor har jeg nok ville valgt en vannmølle-løsning, men da ser jeg for meg at litt 3D-printing ville vært best å bruke. [Jeg har valgt et enkelt oppsett for TFT-skjermen, da dette først og fremst er ment som en tilleggsfunksjon for å vise at alt er oppe og går, og ikke som en større funksjon siden boksen er ment å opereres når man ikke er hjemme.

Utfordringene jeg hadde under dette prosjektet var i hovedsak appen som kobler seg opp mot particle photoen. Denne var ikke spesielt intuitiv, og hadde dårlige guides på hvordan man skulle få de forskjellige varslingene til å fungere. Denne appen er også tidvis treg, slik at varsler som er tiltenkt å komme fra den ultrasoniske avstandsmåleren ikke alltid kommer igjennom.

En utvidelse av prosjektet som jeg kunne tenkt meg å legge til, hadde jeg hatt komponentene tilgjengelige, ville vært et kamera for å holde øye med kjæledyret og en høyttaler for å kunne kommunisere med det. Jeg ville også laget boksen ut av noe mer hardført en papp, siden min hund i alle fall ville revet i stykker pappen for å komme til godbitene. Dette er også grunnen at boksen står i en høyde når den demonstreres i vedlagte videofil. Han ville nok fort revet i stykker hele prosjektet.

Jeg legger også ved noen bilder for å vise innmaten av prosjektet da dette ikke kommer frem i videoen.



Festet en liten vekt sammen med litt papir for å passe på at ingenting skulle komme inn under vippe-armen.

Oppsettet til appen:

The screenshots show the configuration of three applets in the Particle IDE:

- Applet 1:** Condition: **inFront on "Jerry"** (Greater, 0). Action: **Send a notification from the IFTTT app** (Message: "Give the dog a treat?").
- Applet 2:** Condition: **counter on "Jerry"** (Less, 0). Action: **Send a notification from the IFTTT app** (Message: "out of treats").
- Applet 3:** Action: **Call a function** (Then call: "feed on "Jerry"", with input: "give").

Første bilde viser oppsettet for å sjekke om hunden er i nærheten med ultrasonisk måling.

Det andre bilde viser oppsettet for varsling på at det er tomt for godbiter.

Det tredje bilde er for å kalle på funksjonen som styrer servoen.

### Referanser:

TFT-skjerm: <https://github.com/jenschr/Arduino-ST7789-Library/>

Servo: <https://docs.particle.io/tutorials/hardware-projects/maker-kit/>

App: <https://ifttt.com/>

Ultrasoniskavstandsmåler: <https://community.particle.io/t/ultrasonic-range-finder-simple-code/19326>