

unlock (6, sense);

IN1060 - unlock(6, sense);

Teknisk rapport

Eksamensinnlevering for prosjektgruppe,
våren 2023 sluttrapport for
IN1060-prosjekt

UNIVERSITETET I OSLO

våren 2023

Innholdsfortegnelse

1 Introduksjon	3
1.1 Målgruppe, tema og konsept	3
1.2 Mål	3
1.3 Krav	4
2 Utforskning	5
2.1 Stempelsystem	5
2.2 DC-motor	6
2.3 Timing	7
3 Endelig løsning	7
3.1 Videopresentasjon av løsningen	8
3.2 Komponenter brukt i løsningen	8
3.3 Bilder, kretstegning og diagrammer	10
3.4 Arduinokode	12
3.4.1 Instansvariabler:	12
3.4.2 Metoder:	13
4 Referanser	14

1 Introduksjon

Formålet med denne tekniske rapporten er å gi en detaljert presentasjon av de tekniske aspektene ved løsningen vår. Rapporten tar for seg grunnlaget for løsningen, våre krav til løsningen, utforskninger vi har gjort gjennom implementeringen av den tekniske løsningen, og en detaljert beskrivelse av hvordan den endelige løsningen fungerer.

1.1 Målgruppe, tema og konsept

Hentet fra hovedrapport:

Målgruppen vår er småbarnsforeldre med småbarn som er 4 år eller yngre.

Vi fant småbarnsforeldre interessante som målgruppe, ettersom ingen av oss hadde betydelig kunnskap om deres utfordringer, som gjorde at vi kunne lære mye fra dem.

Vårt tema er søvn, og vår problemstilling er hvordan hjelpe foreldre med å få deres barn til å sove.

Konseptet vårt er bevegelse, med formkonseptet boks.

Visjonen vår er at vår løsning skal hjelpe både barn og foreldre med å sove bedre.

1.2 Mål

Gjennom flere iterasjoner med datainnsamlinger og analyser, oppdaget vi at rollen som småbarnsforelder påvirker og går på bekostning av søvn og søvnkvalitet. En bruker opplever gjentatte oppvåkninger under natten grunnet at barnet deres blir urolig. Brukeren må opp for å berolige/trøste barnet, før brukeren selv forsøker å legge seg igjen. Dette kan skje flere ganger i løpet av en natt. En annen bruker beskriver innsovingen av barnet som tidkrevende og til tider utfordrende. Etter å ha analysert funnene fra selvrapporteringen, ble det tydelig at selv om brukerne hadde ulikheter i opplevelsen av å være småbarnsforeldre, stod de alle overfor en felles utfordring når det gjaldt å legge barnet til sengs. Denne utfordringen førte igjen til søvnkvalitetsproblemer på grunn av hyppige oppvåkninger hos barnet i løpet av kvelden og natten.

Med hjelp av brukerne valgte vi å løse utfordringen ved å lage en “*selv-vuggende*” vuggeseng som hjelper med å berolige barnet gjennom barnets sovetid. Vuggesengen lages først og fremst med tanke på barnets behov, dermed blir barnet primærbrukere av vuggesengen. Småbarnsforeldrene blir sekundærbrukere ettersom de påvirkes av vuggesengens funksjonalitet.

Vi ønsker også å gi brukeren funksjonalitet til å bytte mellom den automatiserte og manuelle vuggemodusen selv. En bruker ønsket også at det skal være en visuell indikator som formidler statusinformasjon om hvilken modus vuggesengen står på.

En omfattende beskrivelse av idémyldringen, datainnsamlingen og analysene som førte oss til denne løsningen kan leses mer om i hovedrapporten.

1.3 Krav

Gjennom analysen av workshop med bruker (under tredje iterasjon) kom det frem flere funksjonelle og ikke-funksjonelle krav til artefakten (den automatiserte vuggesengen).

Funksjonelle krav	Forklaring
Artefakten skal ha en automatisert vuggemodus som trekker sengen frem og tilbake.	Artefakten skal vugge av seg selv, så motoren må være forhåndsprogrammert for å gi en beroligende vuggebevegelse.
Artefakten skal ha et eller flere lys som gir feedback.	Lysene skal indikere start og når motoren er i gang.
Artefakten skal kunne startes og stoppes manuelt.	Det skal være mulig for bruker å stoppe motoren når som helst etter at den har startet.

Tabell 1: Oversikt over funksjonelle krav for artefaktet.

Ikke-funksjonelle krav	Forklaring
Motoren må kjøre med en frekvens som virker beroligende for barnet.	Vuggebevegelsen drives av at motoren alternerer kjøreretning, denne alterneringen må skje slik at det har en beroligende effekt.
Artefakten må ikke være støyende, og motvirke innsovning.*	Kontrollboksen bør være tilstrekkelig lydisolert for å dempe støy fra motoren.

Tabell 2: Oversikt over ikke-funksjonelle krav for artefaktet.

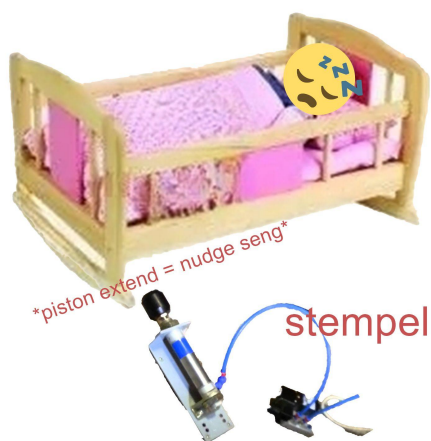
* Vi klarte ikke helt å oppfylle dette kravet, da DC-motoren avga mer lyd enn vi hadde forventet oss. Vi hadde på forhånd sett for oss at kontrollboksen skulle virke lydisolerende nok, men motoren kunne fortsatt høres under bruk. En mulig forbedring her ville vært å ha kuttet materialene til boksen med en

CNC-maskin slik at man unngår små unøyaktigheter og sprekker som lekker lyd. Eventuelt også kle innsiden av boksen med et tynt lag gummi eller annet lydisolerende materiale.

2 Utforskning

2.1 Stempelsystem

Vi vurderte opprinnelig å bruke et stempelsystem for å gi vuggesengen et forsiktig dytt. Tanken bak bruken av et stempel var basert på ønsket om å oppnå pålitelighet og jevnhet i vuggemekanismen. Vi så for oss at et stempelsystem kunne gi en presis og konsekvent vuggebevegelse for å berolige barnet.



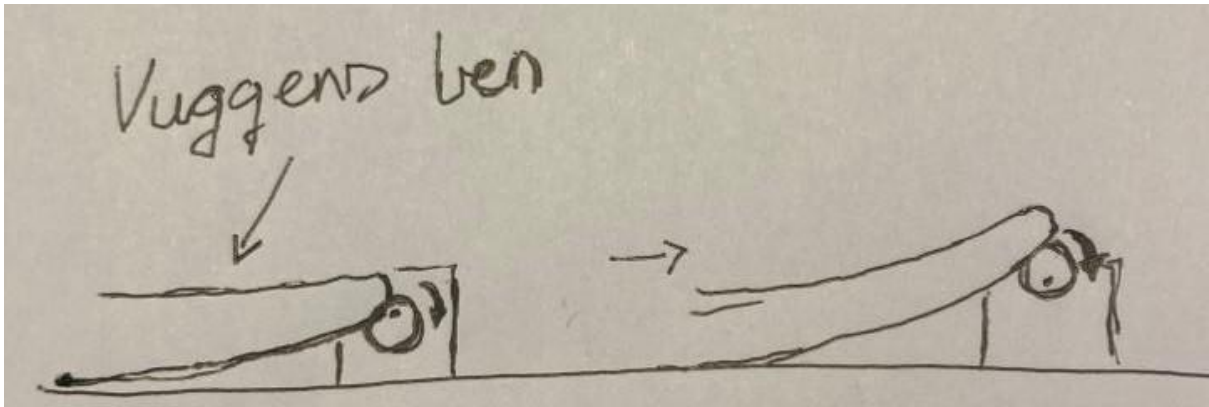
Figur 1: En illustrasjon som viser hvordan vi hadde forestilt oss at stempelsystemet forsiktig dytter vuggesengen.

Etter idémyldringer og diskusjoner, konkluderte vi med at implementering av et stempelsystem ikke var en optimal løsning for den automatiserte vuggesengen. Stempelsystemet ville ha vært for kostbart og stempelets støynivå kunne ha resultert i oppvåkning av barnet. Vi valgte derfor å utforske andre alternativer som kunne oppfylle de samme kravene for en pålitelig og behagelig vuggemekanisme for vuggesengen.

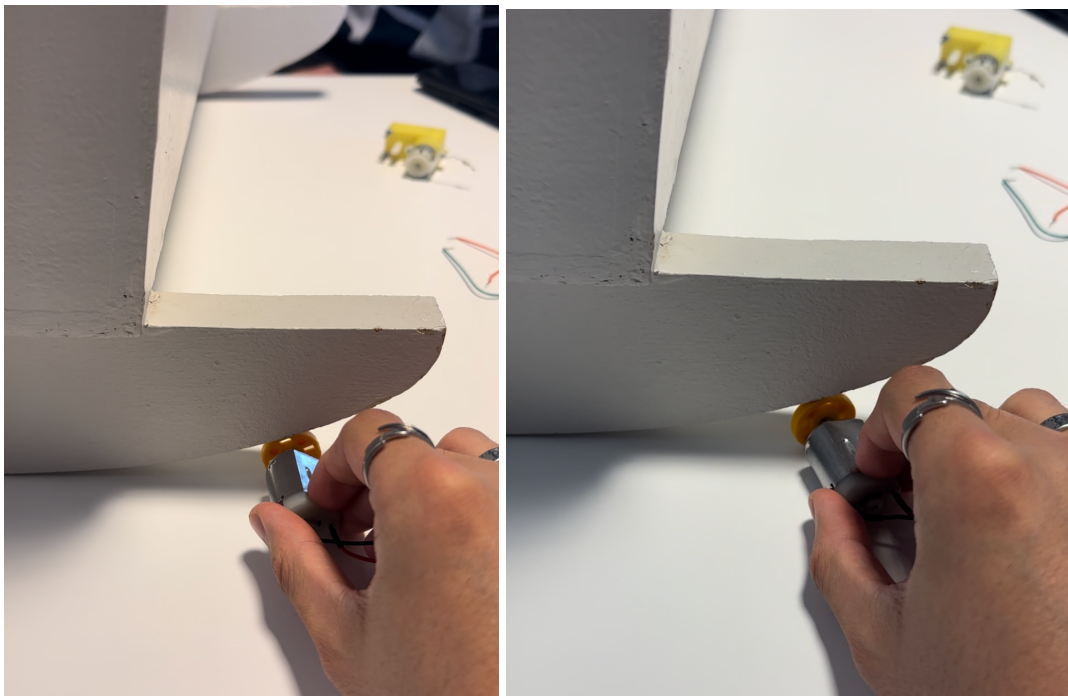
2.2 DC-motor

Vi bestemte oss for å begrense løsningen vår til motorer som allerede har dokumentasjon på at de er compatible med arduino, og ikke er altfor dyre. En DC-motor ble et godt alternativ, da det allerede er

instruksjoner på hvordan det settes opp i Arduino sin medfølgende prosjektbok. Vi prøvde først å få vugget sengen ved at vi festet et hjul "off-center" til tuppen av motoren. Se tegning under.

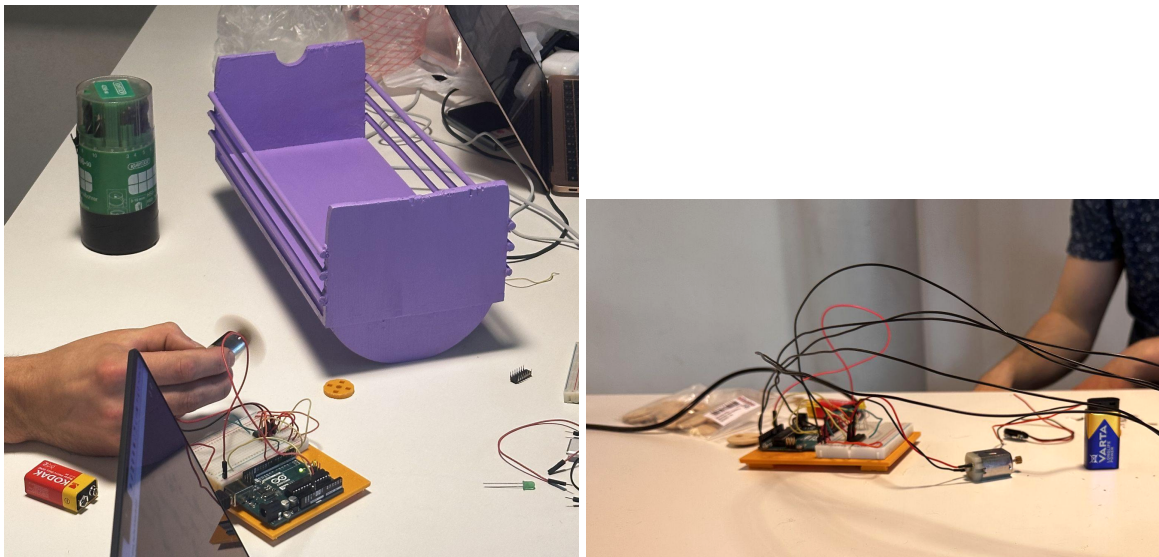


Figur 2: Det viste seg at motoren ikke var sterk nok til å løfte sengen, og vi måtte finne en annen måte for å få til vuggebevegelsen.



Figur 3: Etter å teste DC-motoren, fant vi raskt ut av at motoren ikke er sterk nok til vuggesengen vi opprinnelig hadde tenkt til å teste med.

Løsningen vi endte opp med var å bore et ekstra hull i motsatt ende av hjulet, og deretter trekke en tråd fra dette hullet til sengen. Ved at vi alternerte kjøreretningen på motoren, ville tråden først stramme seg og trekke vuggen mot seg, og så slakke opp tråden og tillate vugging i motsatt retning. Denne løsningen var vi ikke helt fornøyd med, siden det forutsetter litt lengre setup-tid, altså må tråden festes til sengen av bruker før den kan startes. Dette kunne vært unngått om motoren var litt sterkere, men det var den eneste muligheten vi fant med komponentene vi hadde til rådighet.



Figur 4: Utforskning av 9V batteri, H-bridge og DC-motor.

2.3 Timing

For at løsningen vår skulle fungere på tiltenkt måte, var det viktig at vuggebevegelsen var beroligende. Dette skulle vise seg å være svært utfordrende å få til, da tråden ville surre seg fast til akselet på motoren dersom den kjørte med for stor fart, eller at motoren endret kjøreretning midtveis i en vuggebevegelse. Etter mye prøving og feiling fant vi et passende tidsintervall til aktivering av motoren. Den endelige løsningens motorhastighet var i sjiktet hvor den ikke ble for treg, altså ikke rask nok til å generere noen vugging, og ikke så rask at tråden surret seg fast.

3 Endelig løsning

Løsningen vår består av to deler:

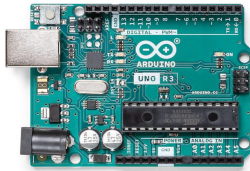
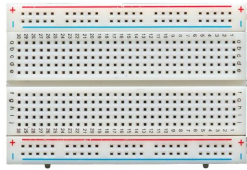

- Gyroskop som festes under sengen. Her er vi interessert i posisjonsdataen fra aksene som forteller oss hvor mye sengen vugger eller ikke.
- Kontrollboks plassert ved siden av sengen. Den har tre LED-lys som lyser fra lokket, og gir feedback om artefaktet er slått på, når motoren kjører, og oppstartssekvens til motor. Motoren er festet til et hjul, som igjen er festet med en tråd til sengen.

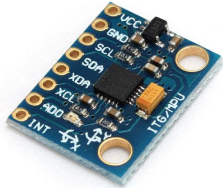





3.1 Videopresentasjon av løsningen



Videoen viser et bruksscenario over en *reklamefilm* der en småbarnsforelder forsøker å legge barnet deres, men barnet er urolig og klarer ikke å sove. Småbarnsforelder er utmattet og ønsker også å legge seg snart. Han fortsetter å vugge sengen og håper på at barnet snart kommer til ro. På TV-en spilles en reklamefilm som presenterer en løsning. Løsningen er en vuggeseng som vugger av seg selv uten hjelp fra småbarnsforeldrene. Løsningens funksjoner beskrives i reklamefilmen og til slutt bruker småbarnsforelder løsningen og er fornøyd! ☺

Lenke til videopresentasjon av løsningen: <https://www.youtube.com/watch?v=jsznKhsXDcQ>

3.2 Komponenter brukt i løsningen

Komponent	Bilde	Bruk	Antall
Arduino Uno		Mikrokontrollerbrett hvor funksjonaliteten blir programmert inn.	1
Half-size breadboard		Koblingsbrett hvor kretsen kobles sammen.	2
Gult LED-lys		Brukes til å indikere når artefakten starter opp, og når motoren kjører.	3

Aksellerometer/gyrosk op (MPU6050)		Brukes for å lese posisjonen av sengen, hvor mye den er rotert til en av sidene.	1
Jumper wire		Brukt til å koble alle komponentene til Arduinoen.	24
Knapp		Knapp som lar bruker slå av artefakten.	1
Motstand (220 ohm)		Brukt for å unngå overbelastning på LED-lysene.	3
DC-motor		Brukt for å drive vuggebevelsen.	1
H-bridge (L293DNE)		Brukt for å styre hastighet og rotasjonsretning på motoren.	1

Batteritilkobling		Brukt for å koble batteriet til motoren.	1
9V batteri		Strømkilden til motoren.	1

Tabell 3: Tabellen viser oversikt over alle komponentene brukt i den endelige løsningen og forklaring på hva de brukes til.

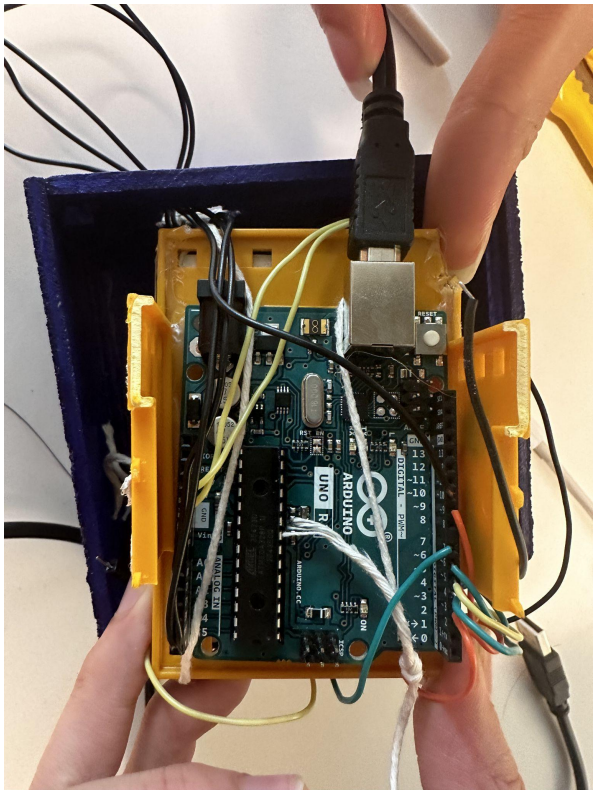
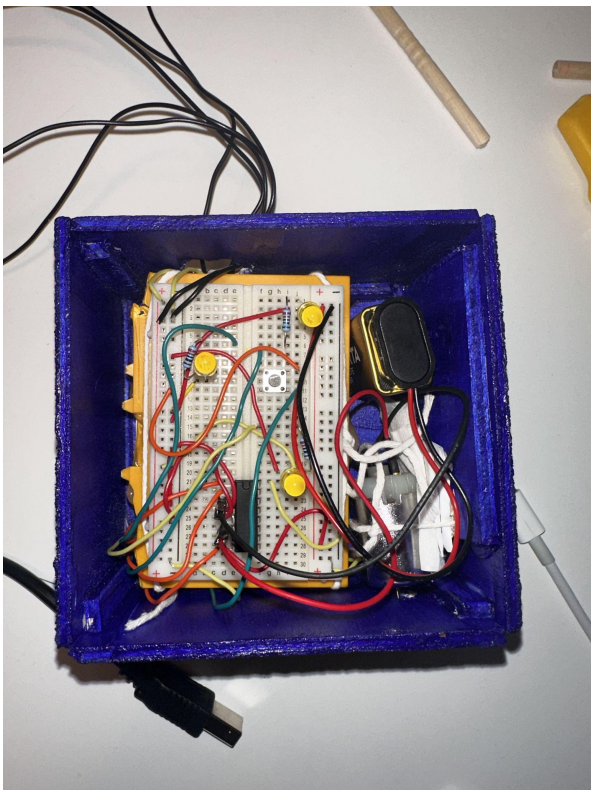
3.3 Bilder, kretstegning og diagrammer



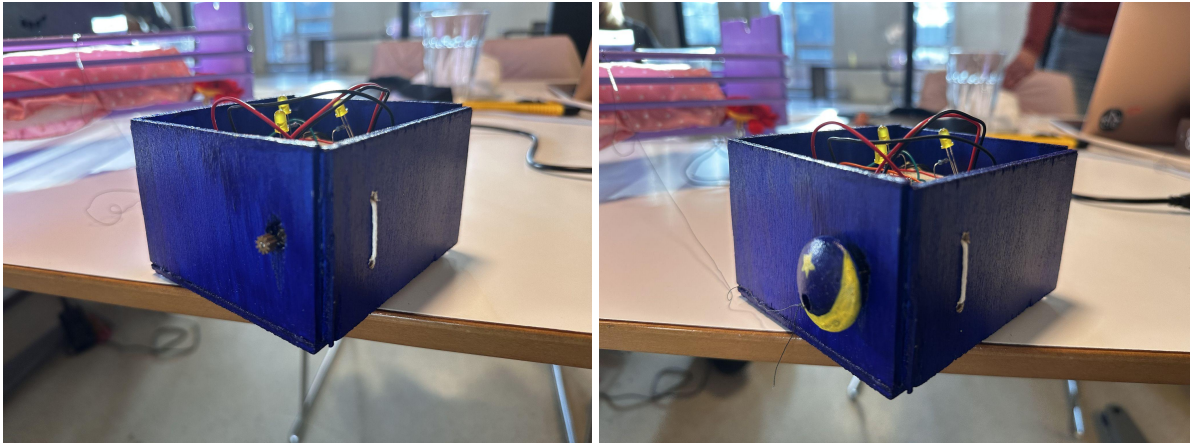
Figur 5: Kontrollboksen koblet til strøm, LED-lys skrur på i et sekund for å indikere auto-modus.



Figur 6: Sensoren sitter på et breadboard limt på undersiden av vuggesengen. Sensoren kobles til Arduino gjennom wire. For å skjule wire brukte vi en type "sleeve".



Figur 7: Slik ser innsiden av boksen ut. Arduino Uno er satt under breadboard.



Figur 8: DC-motor som stikker ut av boksen. Med og uten hjulet som trekker sengen.

3.4 Arduinokode

Vedlagt kodefil i innleveringen.

Kodefil: vugge.ino

3.4.1 Instansvariabler:

Variabelnavn	Forklaring
MPU6050 mpu	Representerer gyroskopsensoren (MPU6050). Brukes til å kommunisere med sensoren.
int16_t ax, ay, az	Variablene brukes til å lagre akselerasjonsdata. Brukes til å hente gyroskopdata, gjennom mpu.getMotion6().
int16_t gx, gy, gz	Variablene brukes til å lagre gyroskopdataen. Relevant for å hente ut verdien til de ulike aksene.
int vuggThresholdMin	Heltall som lagrer den minste terskelverdien for vuggingen. Minimums threshold defineres som: (når vuggesengen står rolig-10).
int vuggThresholdMax	Heltall som lagrer den største verdien for vuggingen. Minimums threshold defineres som: (når vuggesengen står rolig+10).
const unsigned long vuggTimeout	Konstant variabel som angir timeout-verdien mellom vuggene i millisekunder.
int vuggCounter	Heltallsverdi som teller antall vugg for vuggesengen.
unsigned long lastVuggTime	Lagrer tidspunktet for forrige vugg.
const int motorControlPin1	Konstant som angir pin-nummeret til den <i>første</i> DC-motorstyring pinnen.

	Brukes til å styre hvilken retning motoren beveger seg i, sammen med motorControlPin2.
const int motorControlPin2	Konstant som angir pin-nummeret til den <i>andre</i> DC-motorstyring pinnen. Brukes til å styre hvilken retning motoren beveger seg i, sammen med motorControlPin1.
const int motorEnablePin	Konstant som angir pin-nummeret til pinnen som aktiverer motoren. Brukes til å skru motoren av og på.
bool motorRunning	Boolsk verdi, som holder indikerer om motoren er i gang eller ikke.
const unsigned long motorRunTime	Konstant som angir kjøretiden til motoren i millisekunder. Brukes til å angi hvor lenge den automatiserte vuggingen skal kjøres.
unsigned long motorStartTime	Variabel som holder på motorens starttidspunkt. Brukes for å sjekke når den automatiserte vuggingen skal stoppes (ut fra motorRunTime).
const int toggleButton	Konstant som angir pin-nummeret til knappen. Brukes til å slå av/på vuggemekanismen.

Tabell 4: Tabellen beskriver alle instansvariablene i koden og hva de brukes til.

3.4.2 Metoder:

Metode	Forklaring
bool start()	Starter vuggingen av sengen basert på bevegelsesdataene fra MPU6050-sensoren. Den sjekker om bevegelsen er innenfor et definert terskelområde, og teller antall bevegelser (vugg i dette tilfellet).

	Når det oppdages at sengen vugges 4 ganger, starter motoren, og metoden returnerer true.
void startMotor()	<p>Styrer styrke og retningen på vuggemotoren ved å aktivere og deaktivere de forskjellige pinnene (motorControlPin1, motorControlPin2, motorEnablePin).</p> <p>Motoren kjører i en bestemt tid (5 minutter). Hvis knappen holdes inne i mens motoren kjører, stoppes motoren og den automatiske modusen deaktiveres.</p> <p>Metoden er inspirert av oppgaven i <i>The arduino projects book</i> (Fitzgerald & Shilo, 2012) på side 102.</p>
void startVuggCounterLys()	<p>Lyser opp LED-lysene ut ifra verdien til vuggCounter.</p> <p>Avhengig av vuggCounter verdi, skrur bestemte LED-lys på for å indikere antall vugginger som har vært, samtidig som hvor mange flere vugg som må til før motoren startes.</p>
bool buttonWatcher()	Overvåker tilstanden til toggleButton. Når knappen trykkes ned, returnerer metoden true, ellers returneres false.
void blinkLys(int antGanger)	<p>Blinker LED-lysene ut fra parameteren, antGanger.</p> <p>Lysene slås av og på med en bestemt forsinkelse, slik at lysene kan <i>blinke</i>.</p>
void setup()	<p>Setup kjører bare en gang under oppstart, og setter opp de nødvendige pinnene og innstillingene for LED-lysene, motoren og sensoren.</p> <p>Her initialiseres også kalibrering av vuggemekanismens terskelverdier. Vuggesengens stasjonære posisjon -10 for vuggThresholdMin, og +10 for vuggThresholdMax.</p>
void loop()	Loop kjører kontinuerlig i en løkke.

	<p>Her overvåkes knappen for å aktivere/deaktivere auto-modus.</p> <p>Hvis auto-modus er enabled, vil koden se etter bevegelse for å starte vuggingen ved hjelp av start()-metoden.</p>
--	---

Tabell 5: Tabellen beskriver alle metodene i koden og hva de brukes til.

4 Referanser

Fitzgerald, S. & Shiloh, M. (2012). *The arduino projects book*. Torino: Arduino LLC