E5IOT

Internet of Things

Automatisering af hønsehus

5. semester 2020

Science and Technology, Ingeniørhøjskolen, Aarhus Universitet, EDE, Herning

|  |  |  |  |
| --- | --- | --- | --- |
| Deltagere |  | |  |
| Studienummer | Navn | | Initialer |
| 201808044 | Lasse Greve Rasmussen | | LGR |
|  |  | |  |
| Undervisere:  Morten Opprud Jakobsen  Klaus Kolle | |  |  |

# Resume

Denne rapport gennemgår hvordan der er arbejdet med en IoT løsning til automatisering af døren i et hønsehus.

Rapporten beskriver analyse, design, implementering og verifikation af projektet.

Indhold

[Resume 1](#_Toc56431585)

[Forord 3](#_Toc56431586)

[Begreber og forkortelser 4](#_Toc56431587)

[Indledning 4](#_Toc56431588)

[Problemformulering 4](#_Toc56431589)

[Afgrænsning 4](#_Toc56431590)

[Verifikation (LGR) 6](#_Toc56431591)

[Resultat 6](#_Toc56431592)

[Konklusion 7](#_Toc56431593)

[Referenceliste for samlet dokumentation 8](#_Toc56431594)

[Bilag 9](#_Toc56431595)

# Forord

Denne rapport er dokumentation for det selvvalgte projekt der er arbejdet med under faget ’E5IOT’ der omhandler begrebet ’Internet Of Things’.

**Hvem er jeg:**

Lasse Greve Rasmussen

**Uddannelsessted:** Aarhus Universitet, Herning

**Studieretning:** Ingeniør - Elektronik

**Semester:** 5. semester

**Underviser:** Morten Opprud Jakobsen

Afleveringsdato: december 2020

Bedømmelsesdato: december 2020

## Begreber og forkortelser

|  |  |
| --- | --- |
| Begreb | Definition |
| hønsehus | Hønsehuset består af selvehønsehuset samt en tilhørende lille hønsegård. Det er hele denne enhed der lukkes og åbnes for |
| Argon | Udviklingsboard fra Particle |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# Indledning

Formålet med denne opgave er at lave en løsning der automatiserer døren i et hønsehus. Løsningen skal kunne åbne døren ved solopgang, og efter solnedgang skal den lukke døren igen.

Høns er meget pålidelige i at de altid søger ind i deres hus når det begynder at blive mørkt. Det er derfor ikke nødvendigt at kontrollere om hønsene er inde, når blot man sikrer at lågen først lukkes efter at det er blevet mørkt.

Formålet med at holde hønsehuset lukket om natten, er at sikrer at der ikke kommer skadedyr i hønsenes foder.

Systemet skal være baseret på data om dagens solopgang og solnedgang der ligger online, så enheden der styrer systemet skal være forbundet til internettet.

# Analyse

For at klarlægge hvad systemet skal kunne er der lavet krav der definerer dette. kravene

## Krav

**ufravigelige**

* Systemet skal kontrollere en servomotor der kan åbne og lukke døren i et hønsehus.
* Systemet skal have informationer om solopgang og solnedgang via internettet.

**Handling**

* Når solen er stået op, skal lågen åbne.
* Når solen er gået ned, skal lågen lukke.

**Tilstand**

* Mens systemet ikke har nogle funktioner, skal systemet bruge så lidt strøm som muligt.

**Uønsket adfærd**

* Hvis der ikke er wififorbindelse, skal systemet styres ud fra lysmålinger målt med en photoresistor.

**Valgfri**

* Systemet kan indeholde styring af belysning i hønsehuset
* Systemet kan være drevet af et batteri der om dagen oplades af en solcelle

# Design

I dette afsnit vi tankerne

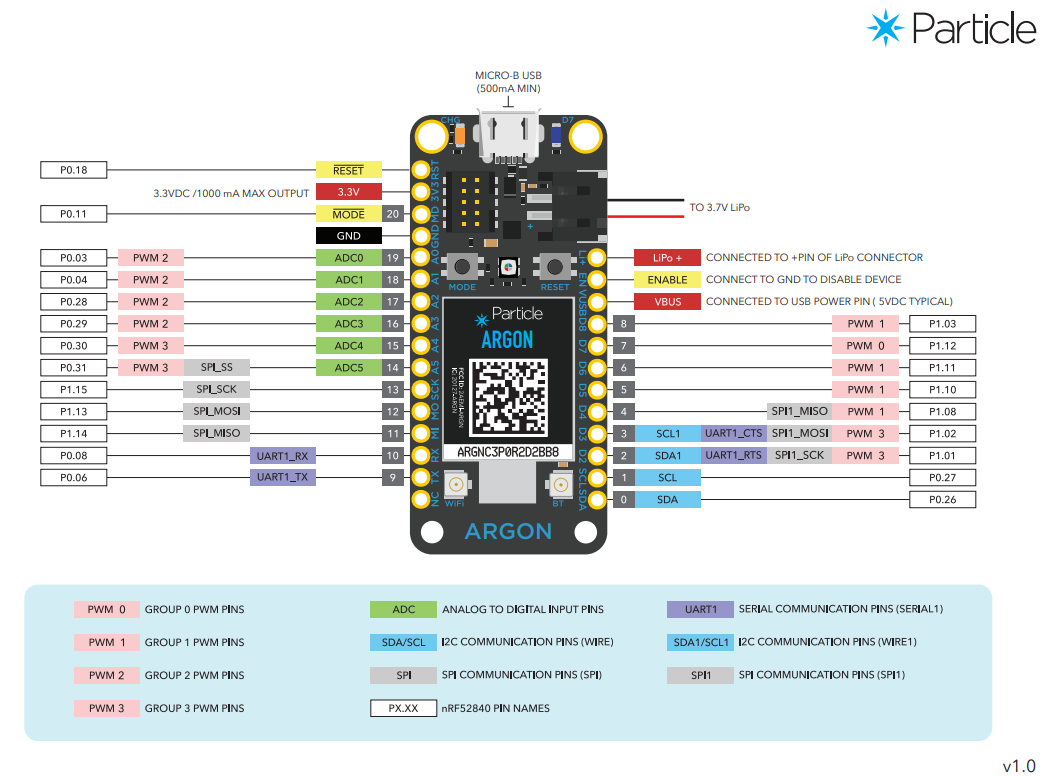
## Platform

Projektets krav til platform

* Minimum én ADC til photoresistor.
* Minimum én PWM udgang til at kontrollere servomotoren.
* Mulighed for internetforbindelse via WiFi.
* Systemet stiller ikke store krav indenfor proces og ressourcer.

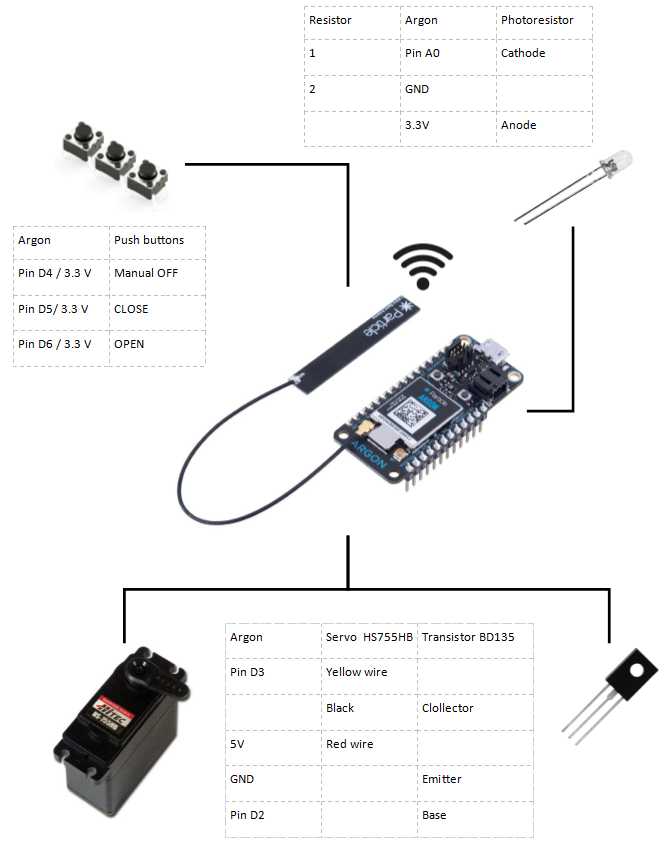
Til faget er der anvendt en particle argon, der er en god platform med mange muligheder til IOT løsninger. Det vurderes at denne platform vil være god til projektet, da den overholder ovenstående krav til en platform.

I Figur 1 ses et overblik over en Particle Argon. Her ses at der er flere muligheder for både PWM og ADC, og boardet kan forbinde til internettet via WiFi. Particle Argon bygger på en 64MHz ARM Cortex-M4F 32-bit processor og har 1 MB flash, og 256 KB RAM [3]. Dette er rigeligt til de relativt simple opgaver der skal løses i dette projekt.

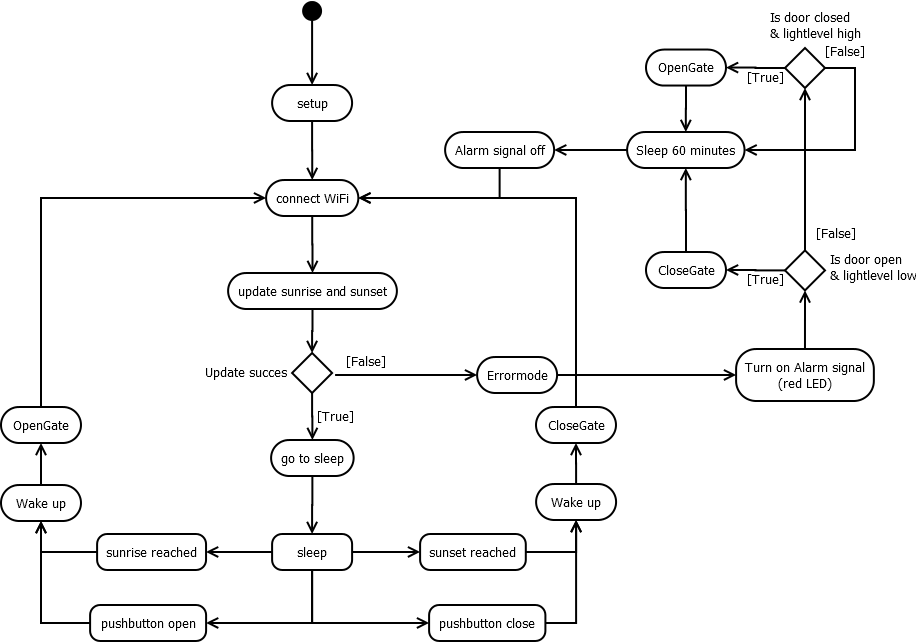


Figur 1 - pinout for Paricle Argon [03]

## Interface analyse



## Aktivitetsdiagram



## Energiforbrug

For at systemet potentielt skal kunne drives af et batteri, er det vigtigt at det har et lavt strømforbrug. Det meste af

## Sleep mode

Der er tre sleep modes der skal vælges mellem. De tre er ’Stop’, Ultra Low Power’ og Hibernate.

Kriterier

* Der er ikke behov for ofte at vågne op, så forbrug ved opvågning har ikke den store betydning.
* Den største del af tiden har enheden ingen funktioner, så der ønskes et sleep mode med meget lavt forbrug.
* Skal kunne styres Wifi
* Skal kunne styres af gpio.
* Der er ikke brug for andre kilder som netværk eller BLE til at vække enheden.
* Der kan muligvis blive behov for at arbejde videre med variabler.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Kan styres af GPIO | Kan styres af RTC | Kan styres af wifi | Forbrug ned til | Gemmer variabler |
| Stop | Ja | Ja | Ja | Ca. 400 µA | Ja |
| Ultra Low Power | Ja | Ja | Ja | Ca. 80 µA | Ja |
| Hibernate | Ja | Nej | Nej | Ca. 65 µA | Nej |

Tabel 1 - kilde: [https://docs.particle.io/reference/device-os/firmware/argon](https://docs.particle.io/reference/device-os/firmware/argon/#sleep-sleep-)

Ud fra denne sammenligning virker det oplagt at vælge Ultra Low Power, der ligger langt under forbruget i ’stop’, men som stadig har de funktioner der skal anvendes.

Langt de største energiforbrug vil dog være under drift af servoen.

## Energiforbrug

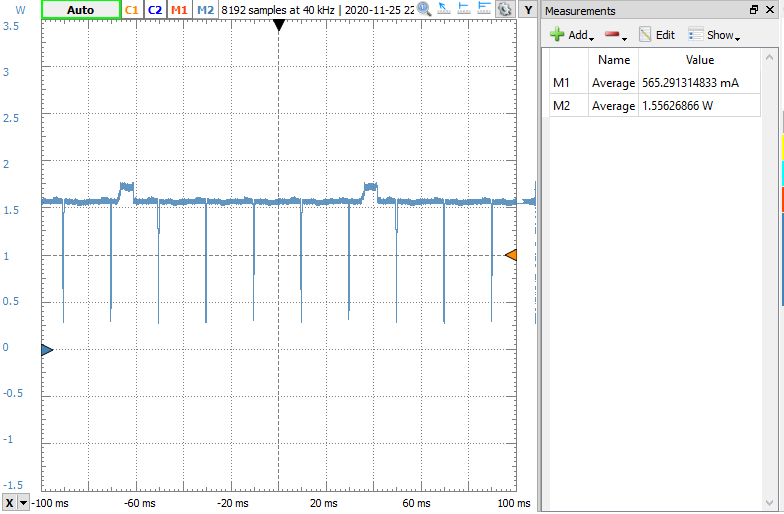
For at vurdere prioriteringen af energi optimering

5 sekunders med aktiv låge to gange om dagen.

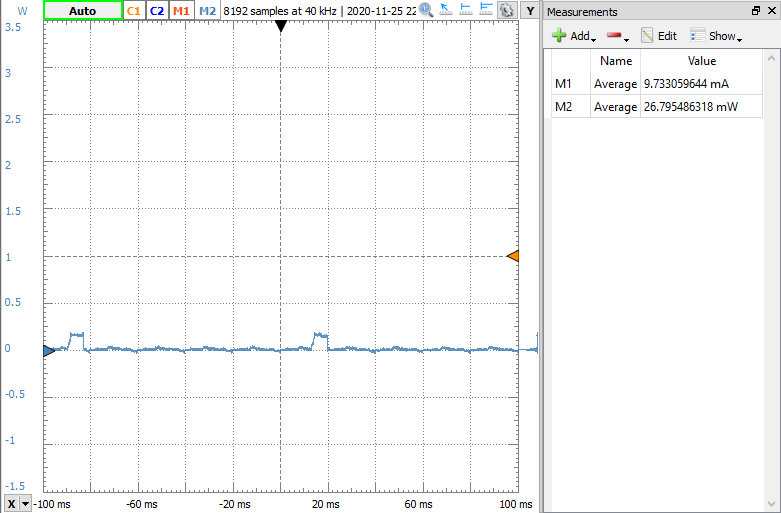
Et døgn med argon i standby

Et døgn med argon i STOP

Det er vigtigt at finde en løsning hvor argon kommer i en grad af sleep, når den ikke skal være aktiv.



Figur 2 - effektforbrug med servo aktiveret



Figur 3 - effektforbrug med servo deaktiveret

# Implementering

# Videre arbejde

# Verifikation (LGR)

## Resultat

# Konklusion

# Referenceliste for samlet dokumentation

|  |  |  |
| --- | --- | --- |
| Id | Beskrivelse | Reference |
| [01] | Github til projektet | <https://github.com/spejderlasse/IOT> |
| [02] | Paticle reference manual | <https://docs.particle.io/reference/device-os/firmware/argon/> |
| [03] | Particle datablad | <https://docs.particle.io/datasheets/wi-fi/argon-datasheet/> |
| [04] |  |  |
| [05] |  |  |
| [06] |  |  |
| [07] |  |  |
| [08] |  |  |
| [09] |  |  |
| [10] |  |  |
| [11] |  |  |
| [12] |  |  |
| [13] |  |  |
| [14] |  |  |
| [15] |  |  |
| [16] |  |  |
| [17] |  |  |

# Bilag