

GrowPro Växthusautomation



Isak Monrad-Aas
Albin Wilson

Division of Industrial Electrical Engineering and Automation
Faculty of Engineering, Lund University

GrowPro
Växthusautomation
2017/06/22 - 2018/04/10



LUND UNIVERSITY

Isak Monrad-Aas 920830-1474
920830-1474
Tel: 0760170023
E-mail: isak.monradaas@gmail.com
Albin Wilson 911029-3959
911029-3959
Tel: 0730431868
E-mail: albin.t.wilson@gmail.com

Abstract

With food shortage becoming a growing problem at the same time as the world population increases and cities become more densely populated than ever before, resource efficiency has become a more dire issue than ever. One major problem in food production today is that production efficiency is very weather dependent, transportation is also a big factor in environmental impact and almost half the world's landmass is used for farming, meaning that cultivation of new land is under more pressure than ever.¹

Furthermore pesticides have become commonly used to increase productivity and minimize waste while in recent years it has been proven that these methods can cause health issues.²

This is where indoor farming has become a growing trend in densely populated areas and it's proving to be a financially feasible way to meet growing food production demand. Some vertical farms proving to be 100 times more efficient per square meter compared to traditional methods, with 40% less power consumption, 80% less food waste and 99% less water usage compared to outdoor fields³. Furthermore the indoor environment can be kept sanitary and therefore the usage of pesticides can be kept at a minimum.

With this as a background this project aims to design and build an automated indoor growing environment. An iterative approach will be taken and several prototypes will be constructed, one after another, in order to end up with a small-scale indoor farming setup, also known as a GrowPro.

¹ Owen, National Geographic. 2005. Farming Claims Almost Half Earth's Land, New Maps Show. [http://news.nationalgeographic.com/news/2005/12/1209_051209_crops_map.html]. 19/9 2017.

² US National Library of Medicine. 2007. Cancer health effects of pesticides. [<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2231435/>]. 21/9 2017.

³ Benke and Tomkins. 2017. Future food-production systems: vertical farming and controlled-environment agriculture. [<https://www.tandfonline.com/doi/full/10.1080/15487733.2017.1394054>]. 21/9 2017.

Innehållsförteckning

INNEHÅLLSFÖRTECKNING	3
INTRODUKTION	4
TEORI	6
METOD	8
AVGRÄNSNINGAR	9
MODELL	10
ITERATION 1	10
<i>Konstruktion av Iteration 1</i>	10
TEST AV LUFTFUKTIGHETSREGLERING	13
ITERATION 2	16
RESULTAT	22
DISKUSSION	27
ANALYS AV ITERATION 1	27
ANALYS AV ITERATION 2	27
MÄTVÄRDEN ITERATION 2	28
FORTSATT ARBETE	30
ETT HELAUTOMATISKT SYSTEM - UNDERHÅLLSFRITT PÅ LÅNG SIKT	30
<i>Vattentank</i>	30
<i>Lokal Bevattning</i>	30
<i>Näring</i>	30
<i>Elförsörjning</i>	30
<i>Co2 mätare och gastub</i>	30
<i>pH mätare</i>	31
ÖVERGE JORD SOM ODLINGSMEDIUM	31
SKALA SYSTEMET TILL INDUSTRIELLT BRUK	31
OPTIMERA REGLERPARAMETRAR FÖR EN VISS PLANTA	32
<i>Redan i det konstruerade systemet:</i>	32
<i>Ytterligare parametrar:</i>	32
ANPASSA SYSTEMET TILL UTOMHUSODLING/JORDBRUK	32
ANPASSA SYSTEMET TILL EN KOMMERSIELL PRODUKT	33
SLUTKOMMENTAR	34
REFERENSER	35
APPENDIX	37

1. Introduktion

Syftet med denna kandidatuppsats är att undersöka hur en specifik gröda kan odlas inomhus utan tillgång till externt solljus eller vatten. Vidare skall odlingen ske automatiskt och regleras av ett system med möjlighet att styra en rad faktorer relevanta för odlingen. Reglersystemet skall fungera enligt de optimala parametrarna för en utvald gröda. Efter att systemet sätts igång ska detta sedan underhålla grödorna helt automatiskt utan yttre påverkan. Data skall mätas under projektets gång och användas för att bygga kunskap om optimeringsmöjligheter.

Projektet tar avstamp i omvärldsläget: en dramatisk befolkningsökning i kombination med klimatförändringar samt urbanisering skapar en situation där framställandet av födoämnen måste optimeras.

I takt med att jordens befolkning växer ökar också urbaniseringen. Detta leder till att städer blir allt mer tätbebyggda och därmed saknas det tillgång till åkermark som traditionellt funnits nära dessa områden. Samtidigt ökar efterfrågan på närodlade produkter i och med de negativa effekterna av långa transporter.

De ovan nämnda tendenserna bidrar till en uppfattning att inomhusodlingar i städer kommer att öka. Detta projekt ämnar undersöka huruvida det finns möjlighet att optimera processen så att mindre energi konsumeras och mindre arbetskraft krävs.

Då använd yta är en av de faktorer som spelar störst roll för att kunna vara effektivare än traditionellt jordbruk behöver grödorna växa tätt på bredden och likaså på höjden. Därmed kan inte enbart solljus användas utan en annan ljuskälla för grödorna måste substituera solljuset, detta gäller också bevattning samt näringstillsetser. I och med att alla förutsättningar för fotosyntes nu måste skapas artificiellt, och därmed också styrs av inputvärden som användaren anger, finns även möjlighet för reglering. Naturliga vädercykler så som dag/natt samt regn/uppehåll kan göras ännu mer optimala än en naturlig process som kan drabbas av torka eller översvämning.

Det övergripande målet är därmed att i slutet av arbetet ha tagit fram ett småskaligt automatiserat odlingssystem. Detta system skall kunna reglera åtminstone vatten och ljus genom att mäta fukt och gå efter ett förprogrammerat ljusschema för en specifik gröda. Systemet skall kunna hålla en växt vid liv i en inomhusmiljö utan tillgång till solljus. Därtill önskas utvecklas fler parametrar att regleras/mätas så som temperatur och/eller luftfuktighet. Dessa parametrar kan antingen ha ett förprogrammerat schema för att kunna distribueras automatiskt eller sensorer med ett återkopplat system. Följande givare kan vara relevanta för ovanstående reglering: termometer, effektmätare och fuktighetsgivare för både luft och jord. Det är också tänkbart att vissa av givarna ovan enbart förser systemet med data som kan loggas, men inte regleras, för att se en möjlig korrelation mellan de olika parametrarna och för

att i efterhand kunna studera datan. Dessutom är det intressant att undersöka hur mycket energi systemet kräver samt hur mycket vatten som används och finns kvar i eventuell vattentank.

2. Teori

Ett automatiskt bevattningssystem kan vara uppbyggt på en rad olika sätt. I sin mest grundläggande form styrs enbart vatten enligt en timer, förutsatt att plantan har tillgång till solljus. Detta fungerar bra utomhus när plantan inte är lika känslig för övervattning eftersom allt överskott av vatten då kan dräneras ner i jorden under odlingen utan att plantan blir övervattnad. Om plantan däremot befinner sig i en inomhusmiljö, där överskott av vatten inte kan rinna av, eller om en vill ta hänsyn till vattenåtgång, så är detta system inte tillräckligt bra utan en fuktgivare krävs.

Genom att lägga till en fuktgivare och ett regelsystem som styr vattnet kan bättre resultat uppnås. Fuktgivaren mäter då regelbundet hur fuktig jorden är och ger återkoppling i form av en elektronisk signal till en regulator som i sin tur styr pumpen. Detta system sparar därmed stora mängder vatten eftersom ingen överbevattning sker, se figur 1.

Innan systemet tas i bruk måste regulatören kalibreras enligt önskad fuktighet.



Figur 1: Teoretisk modell av enklast möjliga system.

Eftersom systemet som byggs i detta projekt skall kunna fungera inomhus, även utan tillgång till naturligt ljus, måste också ljus kunna styras elektroniskt så att växternas fotosyntes kan fortgå utan ljus från solen. Genom att låta regulatören styra en växtlampa, vilket är en lampa med våglängder optimerade för fotosyntes, kan solljus imiteras. Ljuscyklarna kan då styras av en realtidsklocka så att en regelbunden "natt" och "dag" kan uppnås, dvs lampan är tänd och släckt för att efterlikna dygnets gång. I länder som Sverige, där de långa nätterna under vinterhalvåret inte ger så mycket ljus, kan detta med fördel även användas då naturligt ljus finns att tillgå men inte är tillräckligt för fotosyntes.

För att ytterligare förbättra systemet kan styrning av luftfuktighet adderas till systemet. För att inte behöva höja luftfuktigheten i hela rummet som plantan står i krävs det en inbyggnad av odlingen, gärna i genomskinligt material. En luftfuktighetsgivare (hygrometer) och en fukt-skapare (förångare) läggs nu också till i regelsystemet för att artificiellt skapa en mer gynnsam miljö, se figur 2.



Figur 2: Teoretisk modell av det mer utvecklade systemet som önskas byggas i detta projekt.

För att kunna optimera kalibreringen av de parametrar som regleras är det också önskvärt att kunna mäta och spara in- och utsignalerna från regulatorn, dvs de data som givarna ger, samt hur ljus, pump och förångare slås på samt av.

Ett av skälen till att bedriva inomhusodling är reducerat CO2 utsläpp eftersom livsmedel såsom kryddor och sallad då kan produceras nära slutkonsumenten i en stad och skördas efter behov. Det går dock åt energi för att driva systemet, så för att kunna få en indikator på energiåtgången är energimätare också tillagd i systemet.

3. Metod

Arbetsättet för kandidatarbetet kommer att använda den litteratur som är tillgänglig kring de viktigaste förutsättningarna för traditionell odling. De mest basala förutsättningar för att växten ska klara sig kommer att analyseras för att kunna ha en kunskapsbas innan projektets bygghfas börjar. Dessa aspekter kommer att vägas in i hur noggrant de måste kontrolleras med ett reglersystem. Sedan kommer de viktigaste reglerparametrarna att implementeras med hjälp av utvärdering av olika komponenter för att konstruera ett så lättstyrt samt justerbart system som möjligt. Allt skall skapas med komponenter som är tillgängliga för allmänheten och relativt kostnadseffektiva. Detta för att göra datainsamlingen och byggprocessen så effektiv och billig som möjligt och därmed möjliggöra flera iterationer. Framst görs detta för att se hur robusta de olika lösningarna är i prototypform och för att kunna testa delar av funktionaliteten istället för att börja med ett alltför komplext system.

4. Avgränsningar

I förberedelsearbetet för detta projekt identifierades alla olika möjliga tillvägagångssätt och systemparametrar upp för att få överblick och insikt i vad som skulle ingå. Därefter gjordes också en rad avgränsningar för att begränsa omfattningen och öka sannolikheten att målen faktiskt skulle kunna uppnås. Nedan följer en rad tekniker, system och mätvärden som inte ingår i detta projekt:

- Styrning och mätning av pH-värde i jorden.
- Styrning och mätning av växtnäring.
- Styrning och mätning av koldioxidhalt/syrehalt.
- Odling i vatten (hydroponics) eller aerosol/luft (aeroponics).
- Odling utomhus.
- Odling i symbios med djur (tex. fisk) eller mikroorganismer.

Vidare bör nämnas att detta projekt är en mindre prototyp som avser undersöka hur ett automatiserat system kan byggas. Därefter kan systemet skalas upp till de industriella nivåer som omnämnts.

5. Modell

5.1 Iteration 1

Det första och mest grundläggande systemet bestod av en fuktsensor⁴, ett pumpsystem, en växt-ljuskälla, en realtidsklocka och en microprocessor, se figur 3. Detta system var öppet, dvs inte i ett separat rum utan delade atmosfär med rummet systemet stod i.



Figur 3: Implementation av det första systemet.

5.2 Konstruktion av Iteration 1

I den första iterationen implementerades ett system bestående av en pump och en vattentank för att styra vattentillförsel. Visserligen gör användandet av en tank systemet mindre underhållsfritt i jämförelse med ett system som direkt kopplas till vattennätet men riskerna med att ge systemet obegränsad vattentillgång i ett prototypstadium ansågs för stora då systemet kan pumpa för mycket vatten och därmed svämma över. För att mäta fuktigheten i jorden användes en hygrometer, som avgör vattenkoncentrationen i jorden genom att skicka en svag ström mellan två pinnar och eftersom spänningen hålls konstant kan resistansen i jorden avgöras och således vattenkoncentrationen, se figur 4. När sensorn är aktiverad eroderas dock piggarna och slits därför ut snabbt om sensorn är på konstant. Eftersom systemet är trögt så räcker det att mäta fuktigheten i relativt stora intervall, till en början en gång i timman. Hygrometern har en drivkrets som skickar en analog signal mellan 0-5 V till en mikrokontroller, i detta fall en Arduino Uno, som är systemets regulator. Arduinons A/D omvandlare har en upplösning på 1024 steg och det blir också systemets upplösning på fuktighet. När jorden är för torr så aktiverar Arduinon en pump via en MOSFET som då pumpar vatten ur en tank och vattnar plantorna genom en slang med ett flertal hål i för att få bra spridning. I detta system, med två krukor, så mäts fukten enbart i den ena av krukorna och eftersom de är nästan lika stora så antas de torka ut lika snabbt och vattnas därför lika mycket

⁴ Aosong Electronics Co. Temperature and humidity module. DHT11 Product Manual, http://akizukidenshi.com/download/ds/aosong/DHT11_20171226.pdf

genom de blåa cirkulära rören som syns i figur 8 & 10. Det lila-röda ljuset som syns i figur 5 är från växtljuset som i detta fall är LED-listor med den optimala ljus-frekvensen för fotosyntes⁵, dessa monterades ovanför krukorna för att ge bra ljusspridning. De styrs enligt ett schema med hjälp av en realtidsklocka med eget batteri för att garantera att tiden hålls även vid tillfällig urkoppling av systemet eller vid strömavbrott. Lamporna ställs till att lysa under 12 timmar och vara avstängda under 12 timmar för att det antas att växter mår bra i ekvatoriala miljöer där dagen varar 12 timmar. Lamporna hålls dessutom på ett jämnt avstånd från växterna för att se till att de får maximal mängd ljus under varje steg i växtprocessen. Efterhand som växterna blir högre kan därmed lamphöjden justeras med hjälp av skenorna.

5.3 Påbyggnader iteration 1

Efter ett par veckors drift så började pumpen vattna för mycket och det rann över lite vatten ur faten som plantorna stod på, detta var en indikator på att något var fel och att det var rätt beslut att inte koppla systemet till vattennätet. För att felsöka anslöts Arduinon till en dator och där kunde sensorvärden och andra relevanta data läsas av via serie-monitorn. Det visade sig då att mätningarna gav utslag för torr jord även när systemet var igång och jorden var ordentligt fuktig. Orsaken till detta skulle kunna vara att proben har börjat oxidera eller att jord har sjunkit undan och det därför är mindre kontaktyta och således flyter en lägre ström igenom mätkretsen än vid kalibreringstillfället. Detta uppfattades därmed som torrare jord än det faktiskt var.

Det var relativt enkelt att ändra om referensvärdet för jordfuktigheten i koden men för att slippa behöva ladda upp ny kod med en dator om detta hände igen så anslöts en potentiometer så att värdet kunde justeras manuellt, när som helst, och därmed ändrade referensvärdet för fuktigheten. Detta gjorde även omkalibrering enklare.

Under försöket med iteration 1 stod hela systemet i ett sovrum. Så för att undvika att väckas av den ganska högljudda pumpen lades det också till ett sidokriterium i koden som enbart lät pumpen pumpa om det var dagtid. Pumpen fick efter detta enbart vattna samtidigt som lamporna var påslagna. Dock klarade inte AC/DC-omvandlaren att driva både pump och lampor samtidigt, så under de sekunder som pumpen kördes så inaktiverades lamporna tillfälligt.

I denna första prototype valdes två chiliplantor som testobjekt för att se om systemet fungerade och hölls igång under tiden av nästa iterations byggprocess.

⁵ Chalmers Tekniska Högskola. 2013. Växter berättar vilket ljus de vill ha. [<https://www.chalmers.se/sv/nyheter/Sidor/Vaxterna-berattar-vilket-ljus-de-vill-ha.aspx>]. 10 oktober 201



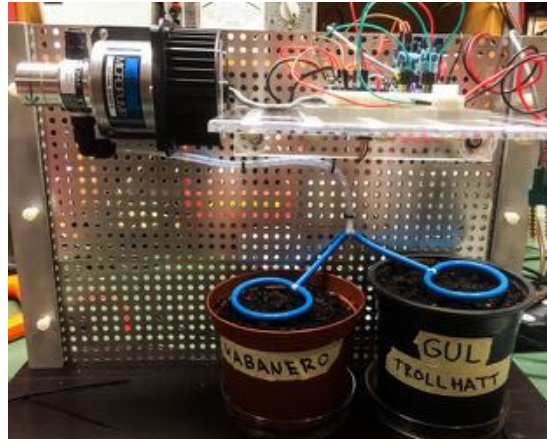
Figur 4: Växtpassade LEDs.



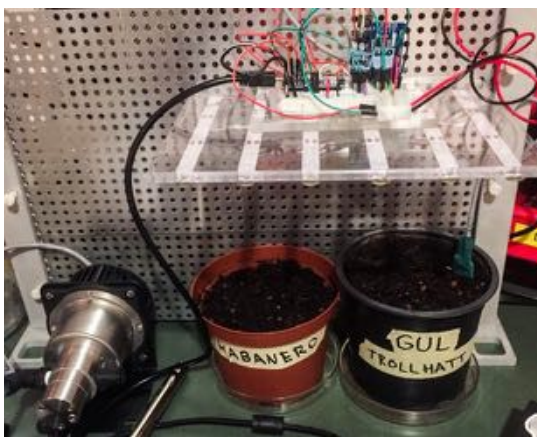
Figur 5: Realtidsklocka.



Figur 8: Iteration 1 under konstruktion, LEDs påslagna på plats, Fröna börjar gro.



Figur 6: Styrelektronik på plats.



Figur 7: Överblick av iteration 1.



Figur 9: Bevattningsslangar på plats, Fröna börjar gro.



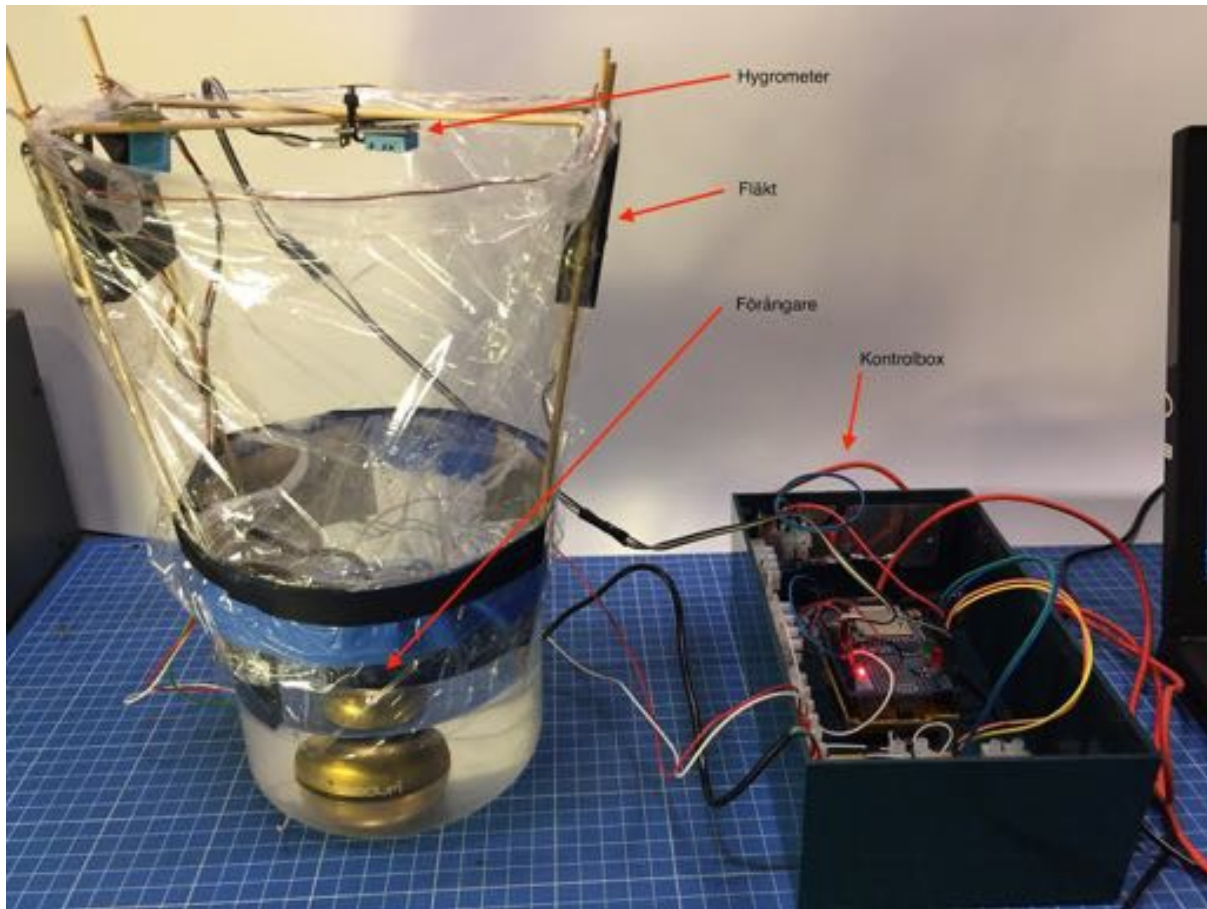
Figur 10: Iteration 1 efter ett par veckor.

5.4 Test av Luftfuktighetsreglering

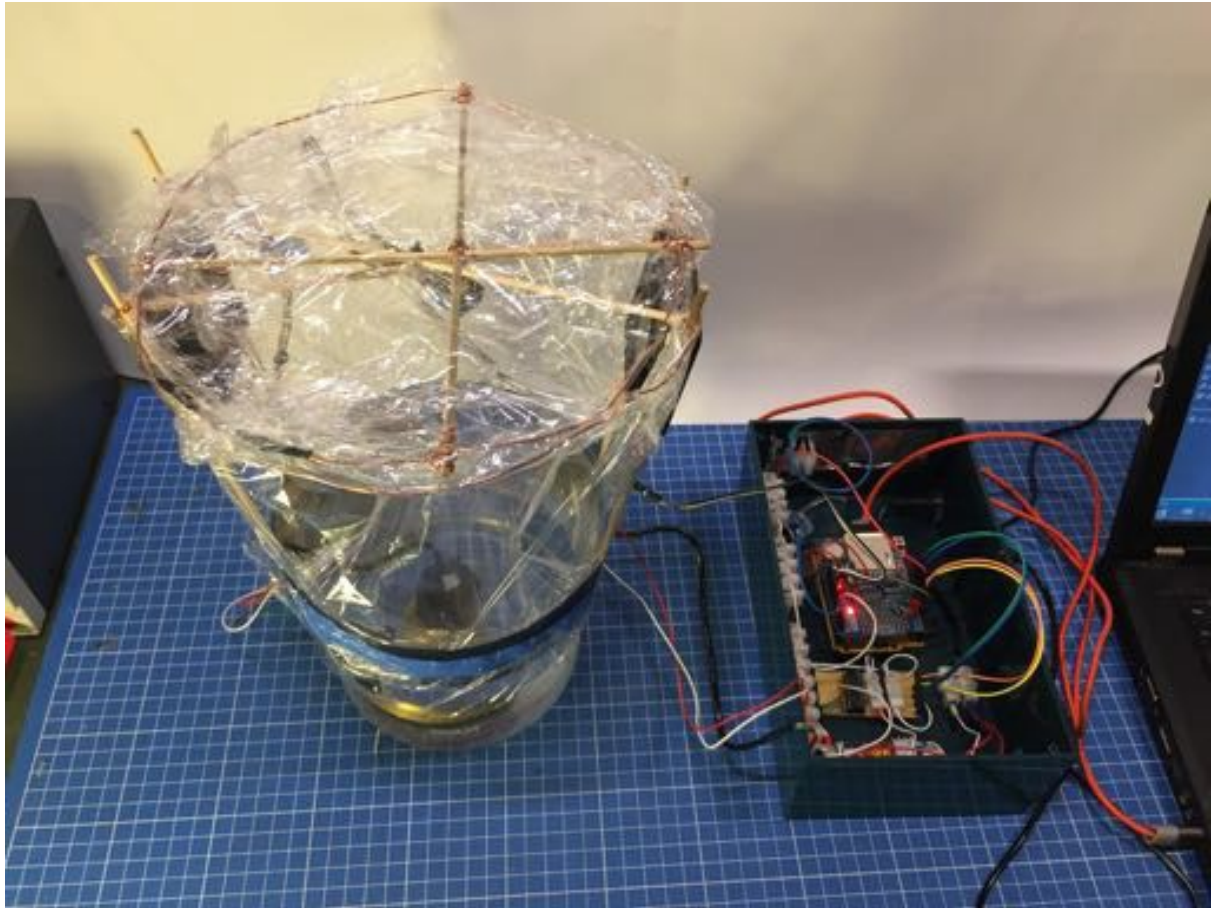
Som synes i figur 12 gav försöket med iteration 1 bra resultat. Systemet fungerade i stort som tänkt och projektet kunde gå vidare till nästa fas och ett mer avancerat system. Målet med projektet var att ha en miljö där inte bara jordfuktigheten kunde styras utan också luftfuktigheten. För att undersöka om detta var rimligt att utföra konstruerades en prototyp av ett växthus med hjälp av plastfolie och en behållare. Botten av denna behållare fylldes med vatten och i vattnet placerades en ultraljudsförångare⁶ för att kunna producera vattenånga och på så sätt höja luftfuktigheten i en sluten miljö. Denna förångare består av ett piezoelektriskt element som vibrerar i ultraljudsfrekvenser och på så sätt "piskar" upp vattenånga i luften.

I toppen av växthuset placerades en luftfuktighetsgivare med inbyggd termometer för att ge feedback till regleringssystemet samt två fläktar för att vädra ut luft och på så sätt kunna sänka luftfuktigheten igen.

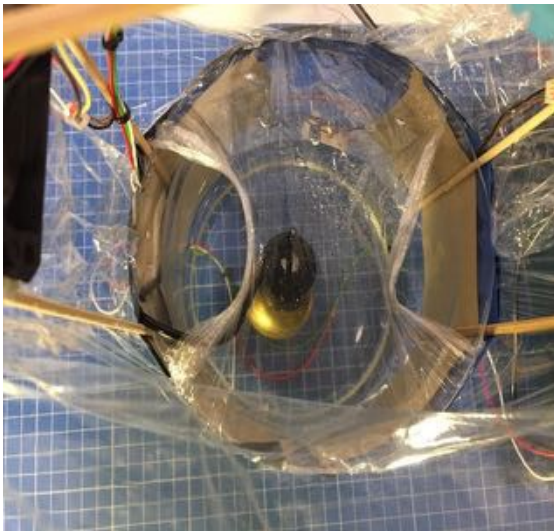
⁶ Lucky Reptile. Mini Nebler Fogging Device Mini Brumisateur, www.hoch-rep.com/html/seiten/support_download.php



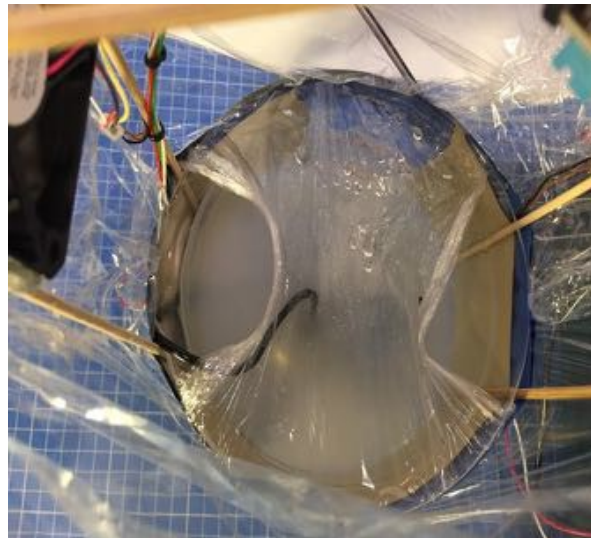
Figur 11: Test av luftfuktighetsreglering.



Figur 12: Test av det slutna systemet.



Figur 15: Förångare av.



Figur 16: Förångare på.

I denna prototyp byggdes även en datalagringsenhet med SD-kortskrivare⁷ in för att enkelt kunna spara all data från alla sensorer samt när pump, lampa, fläktar och förångare aktiverades. Nedan i tabell 1 syns resultaten från det första försöket att öka luftfuktigheten från den naturliga i rummet till ett riktvärde på 90%.

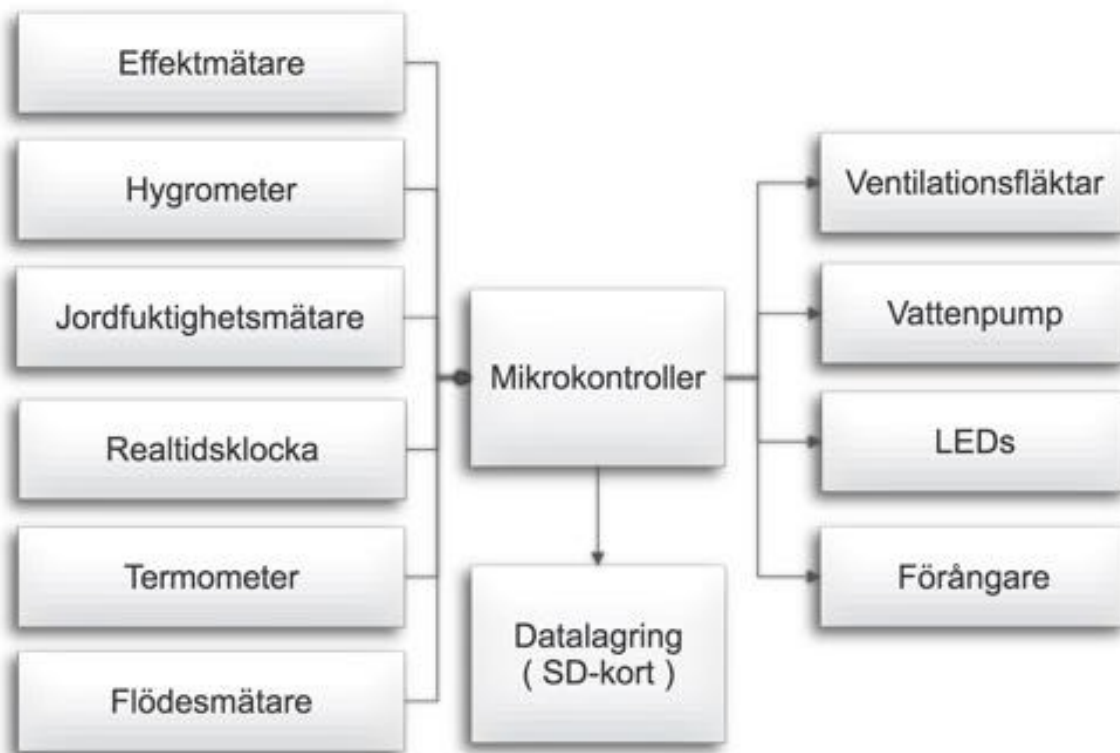


Tabell 1: Första testet av luftfuktighetsstyrning.

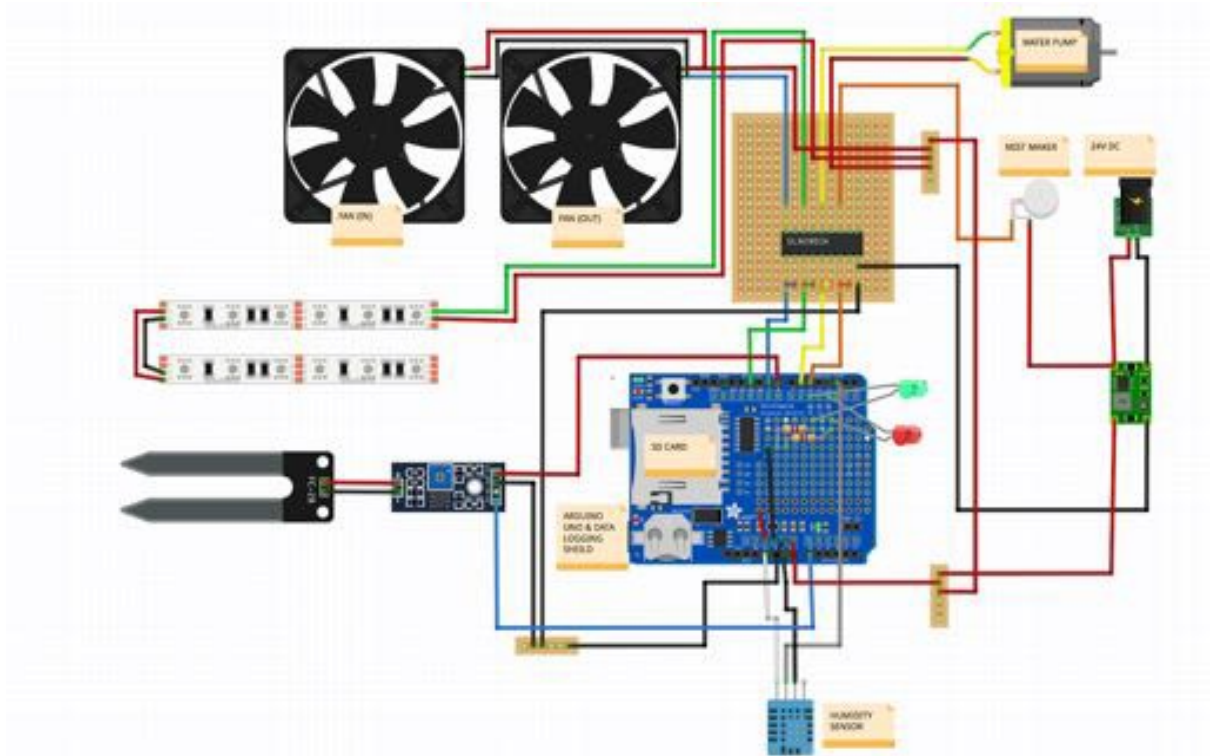
5.5 Iteration 2

Då de parametrar som testades i luftfuktighetstestet fungerade väl och gav mätbara värden påbörjades konstruktionen av iteration 2. Nu kombinerades alla de styrmedel och mätvärden som byggts tidigare till ett system i figur 17.

⁷ Adafruit. 2017. Data Logger Shield. <https://cdn-learn.adafruit.com/downloads/pdf/adafruit-data-logger-shield.pdf>

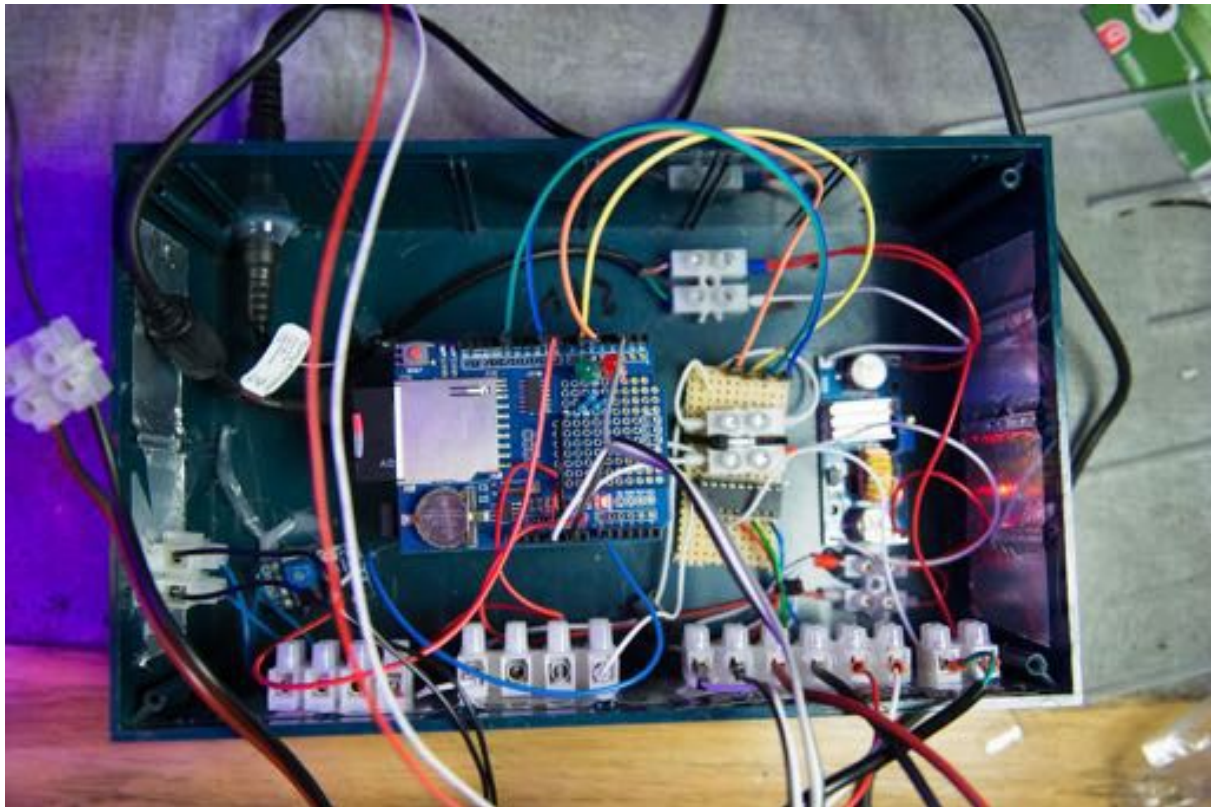


Figur 17: Systemschema över det kompletta systemet i iteration 2.



fritzing

Figur 18: Kopplingschema över det kompletta systemet i iteration 2.



Figur 19: Bild på det kompletta systemet i iteration 2.

I och med att iteration 2 behövde kunna styra luftfuktigheten så byggdes växtkammaren in i ett slutet utrymme. Genom att laserskära plexiglasskivor efter en metallram kunde ett miniatyrväxthus skapas som både håller relativt lufttätt och dessutom ser bra ut i ett hem.



Figur 20: Iteration 2 i planteringsfasen med växtbelysning på.

I den inneslutna miljön reglerades luftfuktigheten med en ultraljudsförångare (figur 21) som skapar en fuktig miljö inuti växthuset genom att skjuta upp vattenånga med ett membran som vibrerar i frekvenser över det hörbara spektrat. Denna fukt kan också vädras ut med hjälp av två fläktar (figur 24) som skapar en luftström genom växthuset. Detta regleras genom att en fuktensensor (figur 22) som mäter hur fuktigt det är i växthuset kopplas till mikrokontrollern. Denna reglerar efter ett utsatt börvärde satt sedan tidigare. Fläkt, förångare, LEDs och pump styrdes via en Darlington Array⁸.



Figur 21: Förångare.



Figur 22: Luftfuktighetssensor.



Figur 23: Temperatursensor.



Figur 24: Ventilationsfläkt.



Figur 25: SD-adaptör.



Figur 26: Energilogger.

Vidare önskades energin som systemet använder mätas för att kunna föra en relevant diskussion om hur CO₂-utsläpp kan påverkas av systemet. För att minimera riskerna med

⁸Texas Instruments. 1997. ULN2803A Darlington Transistor Arrays.
<http://www.ti.com/lit/ds/symlink/uln2803a.pdf>

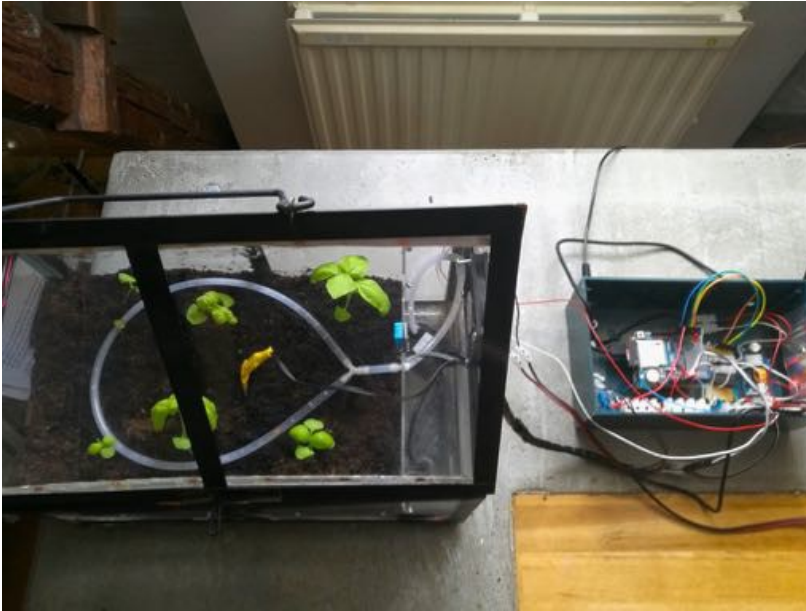
projektet användes en sluten, kommersiell, energiloggare⁹ fristående från Arduinon (se figur 26) för att inte koppla in sig på elnätet som kräver officiell elektrikerbehörighet. Dessvärre loggas den datan inte på SD-kortet men den har en egen loggningsfunktion som kan läsas av manuellt på en display. När hela systemet var färdigställt så planterades basilika i växthuset för att ha en planta som normalt sett behöver en väldigt annorlunda miljö än den vi har i Sverige. Växthusets ljuscykel samt fuktighet bestämdes efter de miljöer som basilika normalt lever i för att få bra skörd.¹⁰

⁹ Luxorparts. 2016. Energimätare. <https://www.kjell.com/se/sortiment/el-verktyg/el-produkter/starkstrom/energimatare/luxorparts-energimatare-p50003>

¹⁰ New World Encyclopedia. 2008. Basil. [<http://www.newworldencyclopedia.org/entry/Basil>]. 25 November 2017.

6 Resultat

I figur 27 & 28 synes sex basilikaplantor jämnt planterade efter fyra veckor.

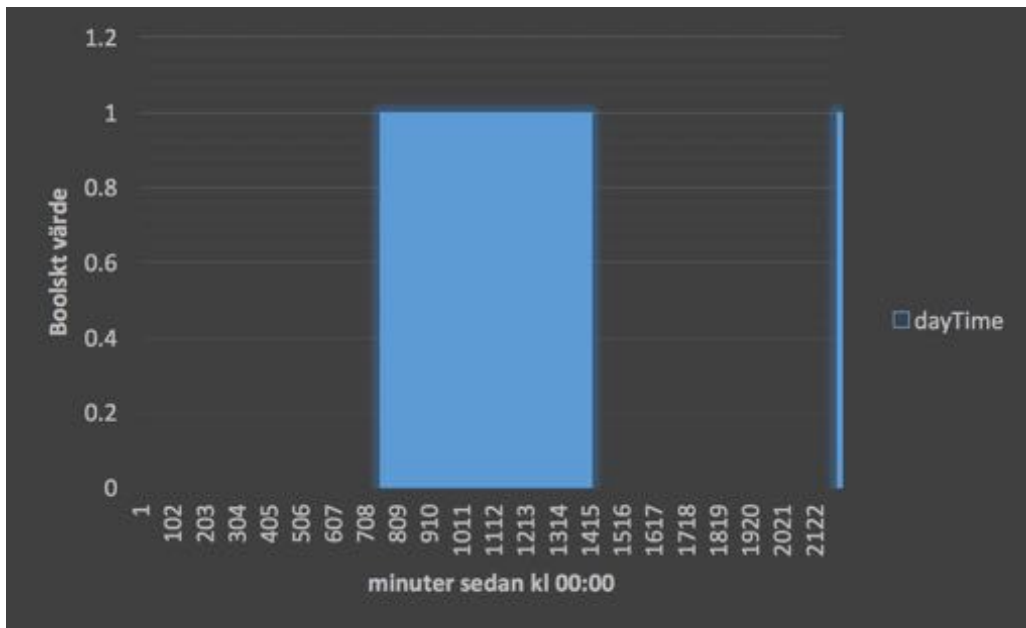


Figur 27: Basilika.



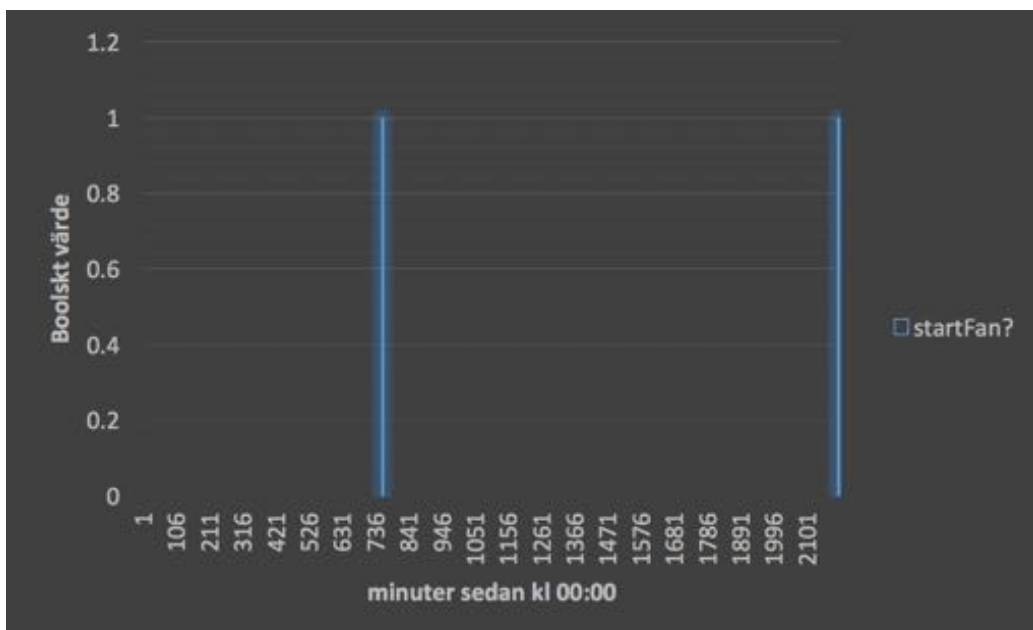
Figur 28: Basilika i växtljus.

Nedan presenteras mätvärden och reglerdata i sju tabeller under en 35h-period i intervaller av en minut per mätning detta visas i samtliga diagram på x-axeln i minuter. Under varje tabell förklaras vad som visas och i diskussionsdelen analyseras mätdata.



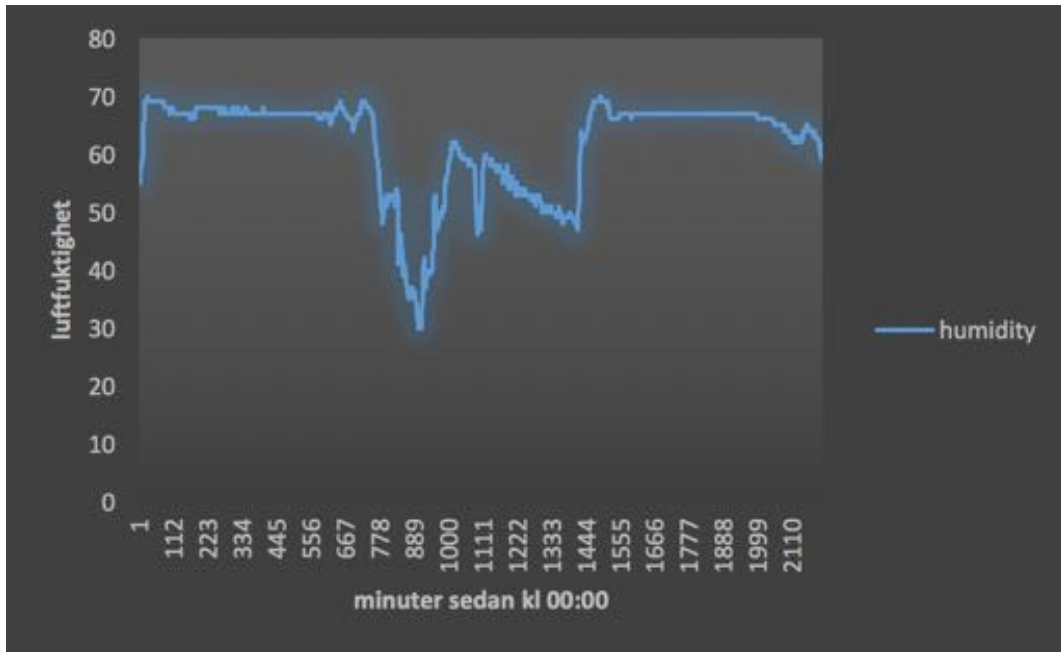
Figur 29: Parametern *dayTime?*.

Figur 29 ovan visar en boolsk variabel från koden som kallas *dayTime?*, den är sann om det är dagtid enligt ett schema som manuellt matas in i kod vid programmeringstillfället. Under denna mätperiod var *dayTime?* sann mellan kl 09:00 och 21:00.



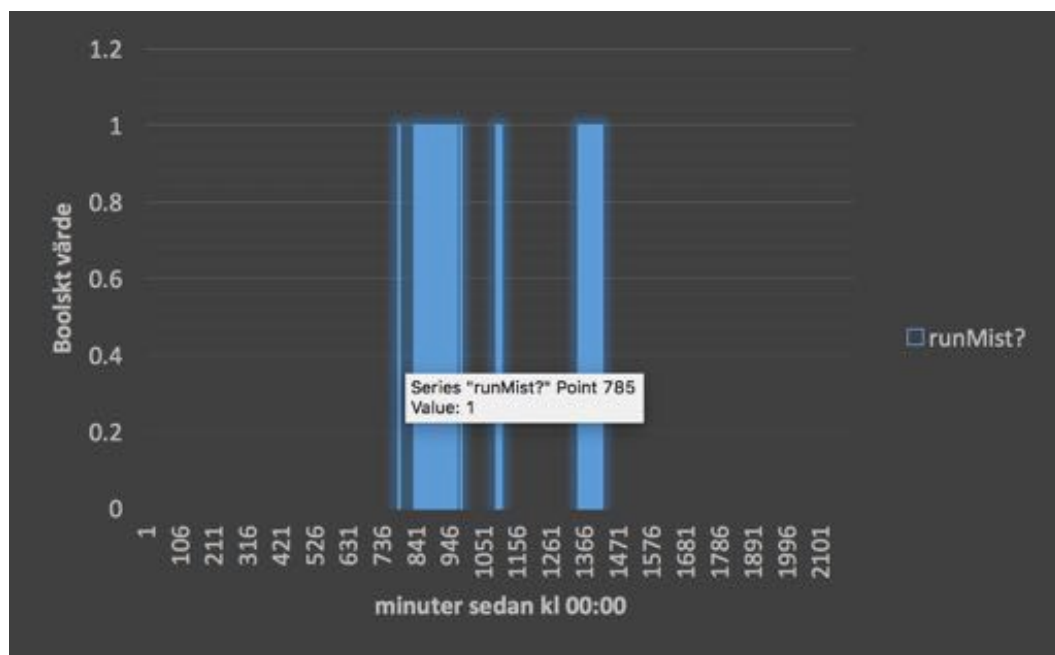
Figur 30: Parametern *startFan?*

Figur 30 ovan visar en boolsk variabel från koden som kallas *startFan?* och är sann då ventilationsfläktarna skall vara igång.



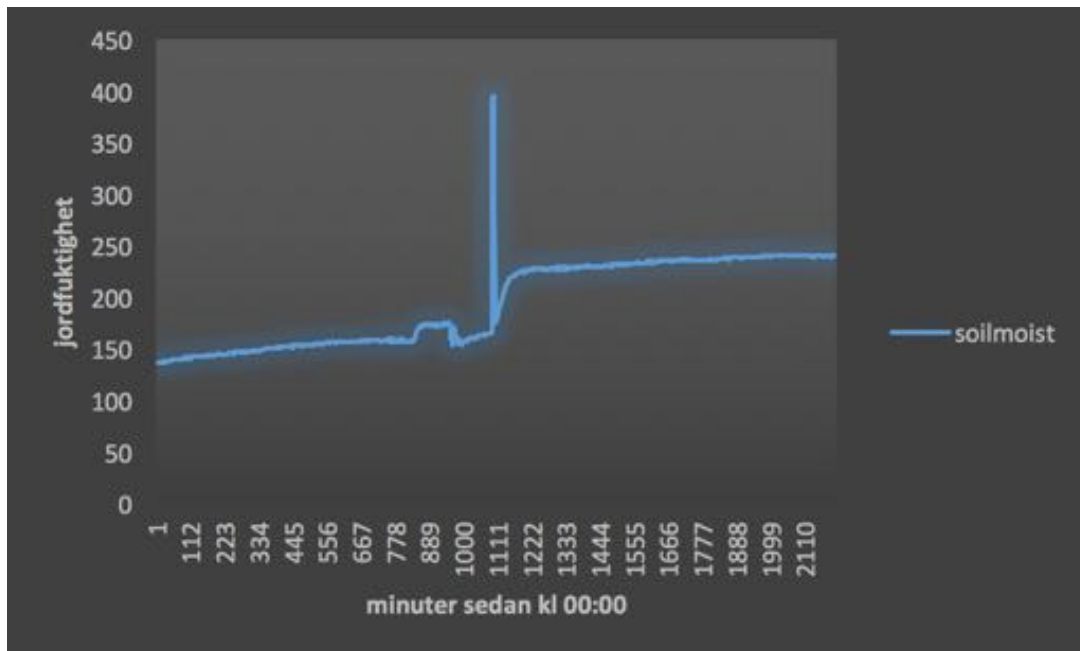
Figur 31: Parametern humidity.

Figur 31 ovan visar ett analogt värde på luftfuktighet, angivet i %.



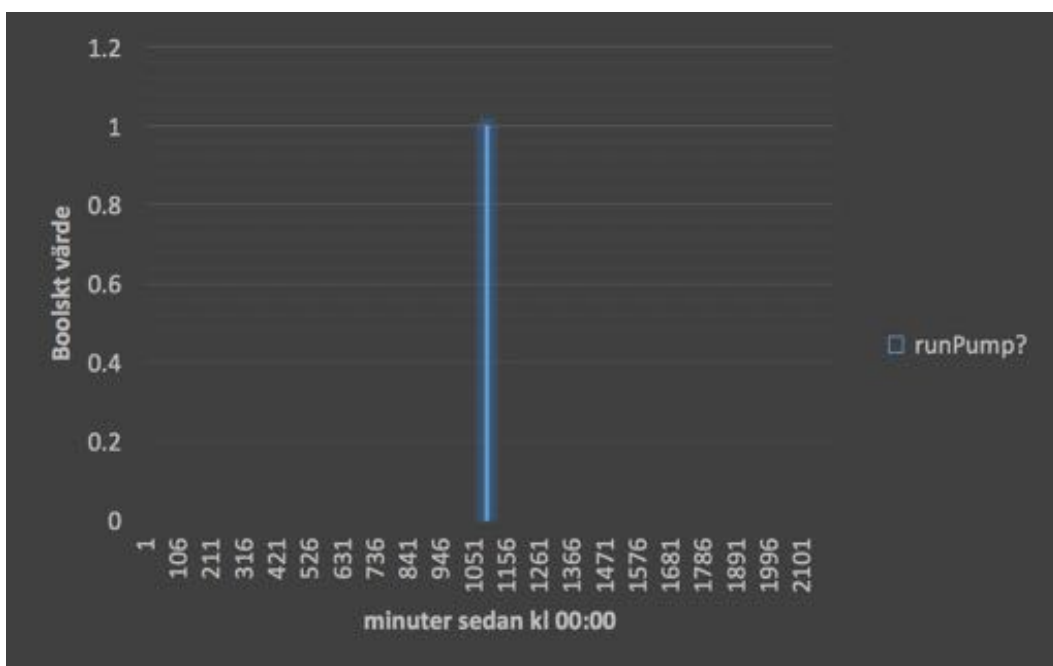
Figur 32: Parametern runMist?

Figur 32 ovan visar en boolsk variabel från koden som kallas *runMist?* och är sann då förångaren skall vara igång.



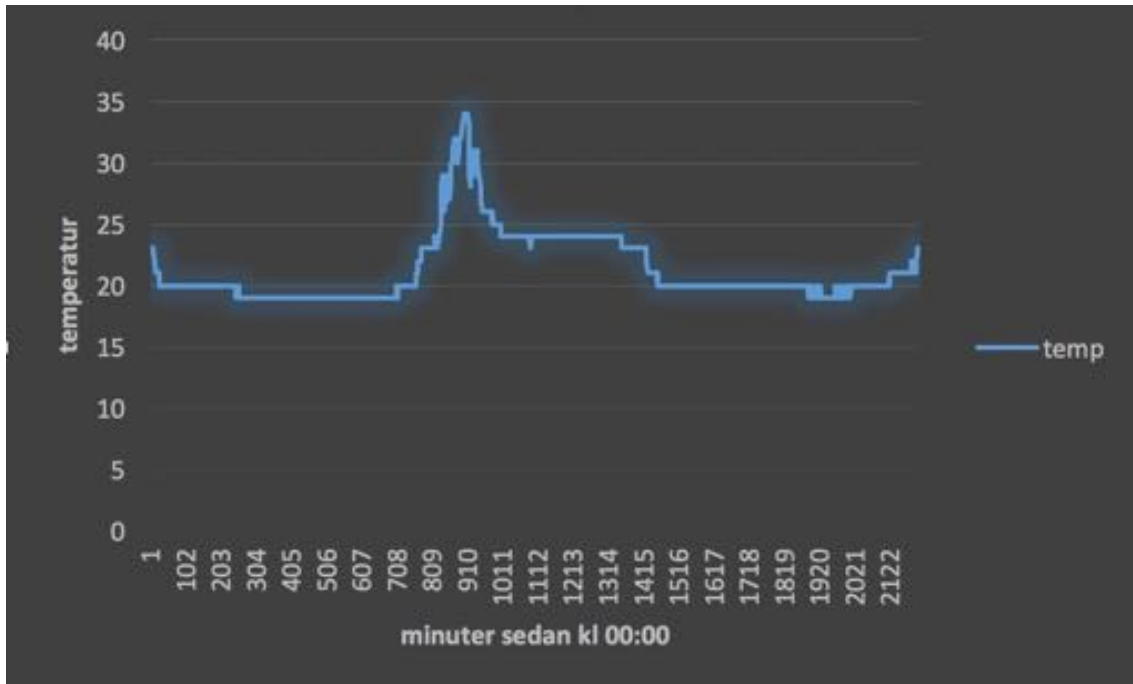
Figur 33: Parametern soilmoist.

Figur 33 ovan visar ett analogt värde på jordfuktighet, angivet i det värde som tas emot av mikroprocessorn från 0 - 1023, där ett högre värde innebär mer fukt i jorden. Den höga spiken i tabellen motsvarar när pumpen precis har vattnats. Detta ses också i figur 34 nedan då runPump? är sann vid precis samma tidpunkt.



Figur 34: parameter runPump?

Figur 34 ovan visar en boolsk från koden som kallas *runPump?* och är sann då vattenpumpen skall vara igång.



Figur 35: Parametern temp.

I figur 35 ovan visas temperaturen i grader celsius. Under fyra veckor fyllides vattentanken på med 2 liter vatten. Slutligen presenteras i figur 36 den åtgångna energin efter 4 veckors konstant drift.



Figur 36: Energiförbrukningen under 4 veckor.

7 Diskussion

7.1 Analys av iteration 1

Det första systemet fungerade på ett tillfredsställande vis och klarade av att lämnas längre tider utan underhåll. En förbättring av iteration 1 hade varit ett tillräckligt bra slutresultat för enklare inomhusodling även utan förbättringarna som gjordes i iteration 2. Dock hade en del mindre justeringar behövts för att till exempel kunna göra en kommersiell produkt. Som nämnt tidigare fanns ett problem med jordfuktighetsgivaren vars data drev med tiden. Detta berodde förmodligen på eroderade mätprobar vilket löstes tillfälligt med en potentiometer. Detta har så klart den stora nackdelen att systemet kräver regelbunden (ett par gånger i månaden) tillsyn. En bättre lösning hade kunnat varit att använda sig av kapacitiva fuktgivare istället, dessa ger fuktighetsdata utan att behöva skicka lika hög ström och således minimeras erosionen. Dessa är dock är komponenten aningens dyrare.

Ytterligare en insikt var att i alla fall chiliplantor snabbt blir väldigt höga. Detta var i viss mån planerat för då lampskärmen var reglerbar i höjdlid på skivan med hål. Efter ett tag behövdes dock längre stavar monteras och efter ytterligare ett par veckor var även dessa för korta. För framtida iterationer hade det varit intressant att belysa växterna från sidan eller hitta sätt att kunna höja lamporna ytterligare utan att göra riggen för stor. En lösning hade varit fästa växtlampan i taket på en justerbar upphängning i lina. Alternativt att användaren i förväg väljer växter som inte blir så höga eller använder systemet enbart som plantskola.

7.2 Analys av iteration 2

Den andra iterationen fungerade väldigt väl och det olika parametrarna som adderades med tillhörande reglersystem som fuktighet samt fläktsystemet verkade ha positiv inverkan på plantorna. Det som var tanken med den andra iterationen var att ha ett klimat inuti växthuset som var mer gynnsamt än det rum som iteration 1 stod i när det kom till fuktighet och temperatur. För att förbättra detta kunde växthuset varit tätare i fogarna samt haft skydd över fläktarna så den varma/fuktiga luften inte sipprade ut allt för mycket. Detta skulle också i sin tur göra reglersystemet mer stabilt och därmed energieffektivare samt mer lättstyrt.

Ytterligare en förbättring hade varit att koppla systemet till vattenkranen så momentet med att fylla på vattentanken försvann, dock skulle systemet verkligen behöva varit beprövat och tätat. Sedan skulle själva växthuset varit större då plantorna slog i taket relativt fort som nämnt tidigare dock var detta inte lika stort problem eftersom basilika är en växt som kan plockas eftersom.

Dock skulle detta kunna vara den perfekta produkten att ha i hemmet för småskalig odling då det tar relativt liten plats, är enkel att koppla in och relativt underhållsfri.

7.3 Mätvärden iteration 2

Som vi ser i tabell 2 så börjar vår mätperiod med ca 12h natt innan systemet sätts i aktivt läge och *dayTime?* Blir sann.

I den iteration av kod som körts då dessa mätvärden tagits så är ventilationen styrd enbart av en timer. Fläktarna körs i 10 sekunder precis då *dayTime?* kl 09:00 på morgonen. Detta syns i tabell 3.

Luftfuktighetskurvan i tabell 4 har ett par intressanta egenskaper. Vi ser att under natten ligger luftfuktigheten relativt stabilt kring 70% utan reglering. Sedan under morgonen sjunker luftfuktigheten avsevärt tills den når ner till 50% (vilket är gränsvärdet för regulatoren). Vi ser i tabell 5 att efter det att luftfuktigheten når under 50% så blir *runMist?* sann och förångaren aktiveras. Detta har, som synes i tabell 4, effekt och kurvan planar ut i en terrassnivå. Dock ser vi att luftfuktigheten fortsätter att sjunka långt under riktvärdet på 50% dryga 30 minuter senare. Inte ens då förångaren aktiverats lyckas luftfuktigheten stabiliseras direkt. Detta har troligtvis sin förklaring i temperaturkurvan i tabell 8. Vi ser att temperaturen kraftigt ökat under samma tid som relativa luftfuktigheten minskat, detta bekräftar den inversa relationen mellan temperatur och relativ luftfuktighet.¹¹

Vidare ser vi att luftfuktigheten fortsätter att stiga efter den djupaste dalen trots att vi ser i tabell 5 att *runMist=false*. Detta beror på en tröghet i systemet som sågs redan i testerna av luftfuktigheten. Även om förångaren fyller en behållare med vattenånga så tar det lång tid innan vattenpartiklarna kan absorberas av luften och uppfattas som högre relativ luftfuktighet av sensorn. Detta leder till ett visst överslag som vi ser i tabell 4, där den relativa luftfuktigheten når ända upp över 60% innan effekterna av förångaren avtar och kurvan igen sjunker pga den höga temperaturen.

Vi ser sedan åter en minskning av relativ luftfuktighet efter det lokala max värdet över 60%. Nu sjunker dock kurvan inte lika snabbt och detta kan förklaras av temperaturen, som ses i tabell 8, som nu inte är så hög som tidigare utan nu ligger runt 25° C.

Vi ser sedan en underlig nedåtgående spik i tabell 4 som inte går att härleda med en korresponderande uppåt-spik i tabell 8. Vad som kan ha hänt här är osäkert. Efter spiken fortsätter i alla fall kurvan som innan och den relativa luftfuktigheten avtar långsamt tills den når 50% och förångaren slås på igen. Trögheten i systemet leder till en översveng varpå luftfuktigheten åter stabiliseras runt 70% när temperaturen når 20°C och systemet går in i nattlig vila då *dayTime?* falsk.

¹¹ Utah State University. 2008. Temperature-Moisture Relationship. [http://ocw.usu.edu/Forest_Range_and_Wildlife_Sciences/Wildland_Fire_Management_and_Planning/Unit_4_Temperature-Moisture_Relationship_4.html]. 25 November 2017.

Det som orsakat de kraftiga skillnaderna i temperatur är antagligen växthusets placering. Under mätningarna har växthuset befunnit sig på en vindsvåning med ett takfönster, den stora spiken upp till över 30° uppnås troligtvis när solen sken direkt in i växthuset. Den senare stabiliseringen runt 25° C svarar mot hela rummets varmare luft och beror inte direkt av solljus in i växthuset.

Vad gäller energiförbrukningen på 7680 Wh se figur 29, som motsvarar ungefär en 60W glödlampa i lite mer än 5 dygn, så är den största förbrukaren utan tvekan LED-lamporna. Eftersom LED-lamporna drar 1.7A vid 12V blir deras effekt 20.4W. De har varit på hälften av tiden (12 h per dygn) under fyra veckor, dvs 336h. Detta motsvarar 6854 Wh. Detta skulle kunna förbättras genom att använda färre LEDs och istället fler reflektorer som då kunde ta till vara på allt ljus, alternativt optimera kombinationen av LED belysning och solljus genom fönster.

8 Fortsatt arbete

Det finns många vägar ett fortsatt arbete hade kunnat gå. Nedan beskrivs ett par relevanta spår som hade kunnat undersökas vidare.

8.1 Ett helautomatiskt system - Underhållsfritt på lång sikt.

Det finns ett par skäl till varför de två systemen i detta projekt bara är någorlunda automatiserade på lång sikt.

8.2 Vattentank

För att minimera riskerna för eventuella systemfel som skulle kunna leda till översvämning användes en intern vattentank. Detta har den uppenbara följden att denna tank måste fyllas på. Det hade varit önskvärt att på samma sätt som diskmaskiner och tvättmaskiner gör, kunna ha fri tillgång till en byggnads vattenförsörjning. Dagens vitavaror använder sig av solenoider men säkerhetsavstängning och en liknande lösning hade varit önskvärd även i detta system.

8.3 Lokal Bevattning

Då fler olika plantor med olika behov skulle dela på samma växthus, skulle ett lokalt bevattningssystem kunna implementeras, detta genom att ha en mätprob för jordfuktigheten vid varje enskild planta och sedan ha t.ex. solenoider som ledde pumpens vatten till de olika delarna av växthuset.

8.4 Näring

I den blomsterjord som köps i affär finns redan näring. Denna näring räcker dock inte hur länge som helst utan efter en tid behöver näringen fyllas på. Att ha en reservoar med flytande växtnäring hade förmodligen förlängt livslängden avsevärt.

8.5 Elförsörjning

Genom en solcell eller annan lokal, förnyelsebar, energikälla hade systemet kunnat bli självförsörjande och inte ens behövs energi ifrån elnätet.

8.6 Co2 mätare och gastub

Att ha extra tillgång till Co2 hade kunnat göra förutsättningarna i miljön mer gynnsamma för plantorna dock är frågan om detta är ett kostnadseffektivt tillägg.

8.7 pH mätare

pH halten i jorden har en inverkan på hur plantor mår och skulle därmed kunna adderas som mätparameter dock är frågan om hur stor inverkan det har på växterna men detta hade varit intressant att utforska.

8.8 Överge jord som odlingsmedium

Det hade varit intressant att undersöka odling helt utan jord som näringsmedium, då det finns många tekniker utvecklade för att göra just detta, nedan listas ett par stycken:

- Hydroponics (odling där växternas rötter doppas i näringsrikt cirkulerande vatten medan själva växten växer genom ett hål).
- Aquaponics (vattendränkta växter med fisk i vattnet för att skapa cirkulation och näringsämnen från avföring).
- Aeroponics (bevattning genom en fuktig gasmiljö med vattenspridare som enbart förgasar vattnet och näringsämnena på rötterna av plantorna).

Detta kan ha många fördelar. En av dem hade varit just näringstillförsel. Det finns exempel på system där vatten och växtnäring cirkuleras mellan växter och tankar. Ett bra avvägt system kan därmed använda sig av fiskarnas avföring eller näringsämnena för att helt växa utan jord. Det är också så att vid användning av aeroponics sker näringsabsorption mer effektivt och därmed behövs det mindre näring samt vattenförbrukning minskar. Intressant är att NASA utvecklade dessa tekniker för att öka effektiviteten vid odling i rymden.

Ytterligare fördelar med aeroponics är att rötterna får bättre tillgång till syre och därför växer snabbare. Det finns också många fördelar i hur snabbt grödorna växer då de får ett bättre näringsupptag eftersom de inte söker efter näring på samma sätt, men den helt klart största fördelen att växterna kan växa tio gånger tätare än vid användning av jord.¹²

8.9 Skala systemet till industriellt bruk

Genom att överge jord som odlingsmedium skulle också systemet vara bättre anpassat för storskaligt, kommersiellt bruk i exempelvis vertikala odlingsystem i större lokaler. Efter skapandet av projektet har vi insett att det är lätt att göra ett större system och hade i det stora inte inneburit någon ökad komplexitet. Då skulle systemet förslagsvis byggas i lagerlokaler för större odlingar, och därmed utnyttja möjligheten till vertikala odlingar samt verkligen gynnas av de sänkta personalkostnaderna.

¹² NASA. 2006. Spinoff: Progressive Plant Growing Has Business Blooming. [https://www.nasa.gov/pdf/164449main_spinoff_06.pdf]. 10 januari 2018.

8.10 Optimera reglerparametrar för en viss planta

I synnerhet för storskaligt bruk så hade en grundläggande undersökning av en viss plantas optimala levnadsförhållanden varit intressant. Ett testscenario med ett stort antal skott av samma art där alla har olika levnadsförhållanden kunde utföras. Således för att se efter ett par iterationer, där de mest framgångsrika kombinationer med den största plantan blir utvalda till nästa steg. Följande skulle kunna undersökas:

Redan i det konstruerade systemet:

- Luftfuktighet. Kanske är det önskvärt att ändra luftfuktigheten under växtens gång, antingen i en dygnsrytm eller i olika faser så som frö, skott, större planta, växt i blom etc.
- Vatten. Även här kan finnas fördel för vissa grödor med simulerade torrperioder med efterföljande blöt period.
- Ljus. Kanske finns det en bättre cykel än 12 timmar mörker sen 12 timmar ljus.

Ytterligare parametrar:

- Koldioxid. Med en ansluten gastank kan CO₂ tillföras i avvägd mängd för att optimera fotosyntesen.
- Näring. Detta har redan behandlats ovan och hade i fallet med en näringsämnes reservoar kunnat regleras optimalt.
- Temperatur. Med kyl och/eller värmeaggregat hade en optimal temperaturcykel kunnat åstadkommas, inklusive dag/natt och sommar/vinter.

Eftersom ett inomhussystem med ovan nämnda styrmöjligheter hade helt kunnat bortse från den naturliga miljön är det tänkbart att det för vissa plantor går att hitta helt nya förhållanden som faktiskt passar bättre. Kanske mår en basilikaplant bättre av en dygnsrytm med 5h dag och 3h natt?

8.11 Anpassa systemet till utomhusodling/jordbruk

Både för en bonde och för en privatperson med trädgårdsland hade det funnits möjligheter att använda ett system byggt på det som konstruerats ovan.

I ett sådant system hade ljus inte längre styrts utan fokus hade legat på bevattning. Eftersom regn fortfarande hade förekommit (förutsatt att odlingen inte befinner sig i en öken, vilket är fullt tänkbart) så skulle bevattning endast ske vid långvarig torka.

En stor fördel gentemot dagens storskaliga, ofta timerstyrda, bevattning är att mycket vatten hade kunnat sparas genom att bara vattna där det behövs och när det behövs.

8.12 Anpassa systemet till en kommersiell produkt

Tidigt i planeringen togs beslutet att använda delar och komponenter som är lättillgängliga och kostnadseffektiva. Detta banar också väg för en kommersialisering av systemet som en säljbar produkt. Medvetna konsumenter, konsumenters önskan om färska och närproducerade grönsaker, transportkostnader och en ökande trend vad gäller självförsörjning är alla faktorer som tyder på att produkten hade haft en marknad. Troligtvis då som ett komplett system till privatpersoners hem och kök för odling av kryddväxter och liknande. Det finns också möjlighet att utveckla en kontinuerlig stabil affärsmodell med försäljning av specialanpassade tillbehör såsom extra sensorer, molnbaserade informations- och stödtjänster eller påfyllnad av växtnäring.

Vidare så hade en kommersiell produkt behövt anpassas till ett mer optimerat system genom att inte använda moduler utan istället designa ett enda system med precis de komponenter som produkten kräver. Komponenterna i modulerna som använts i detta projekt hade dock varit en bra utgångspunkt då modulerna som separata system sannolikt redan är optimerade för kostnad och tillgänglighet.

Något som också hade behövs lösas är livstiden på sensorerna. Detta kunde göras dels genom att låta systemet anpassa sig efter hur sensorerna åldras, dels genom att använda mer robusta sensorer och dels genom att sälja reservdelar. Som nämns ovan är detta också en bra möjlighet för merförsäljning.

9. Slutkommentar

Som avslut anses projektet vara väldigt lyckat då alla test har fungerat enligt plan, växterna har kommit upp och kunnat växa i den "halvartificiella" miljön vi skapat. Mätvärdena har också gett inblick i vad som faktiskt händer och har kunnats appliceras i ett regelsystem. Som nämnts tidigare finns det också ett antal olika riktningar projektet kan utvecklas via fortsatt arbete.

10. Referenser

Sida 2: Owen, National Geographics. 2005. Farming Claims Almost Half Earth's Land, New Maps Show. [http://news.nationalgeographic.com/news/2005/12/1209_051209_crops_map.html]. 19/9 2017.

Sida 2: US National Library of Medicine. 2007. Cancer health effects of pesticides. [<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2231435/>]. 21/9 2017.

Sida 2: Benke and Tomkins. 2017. Future food-production systems: vertical farming and controlled-environment agriculture. [<https://www.tandfonline.com/doi/full/10.1080/15487733.2017.1394054>]. 21/9 2017.

Sida 11: Chalmers Tekniska Högskola. 2013. Växter berättar vilket ljus de vill ha. [<https://www.chalmers.se/sv/nyheter/Sidor/Vaxterna-berattar-vilket-ljus-de-vill-ha.aspx>]. 10/10 2017.

Sida 21: New World Encyclopedia. 2008. Basil. [<http://www.newworldencyclopedia.org/entry/Basil>]. 25/11 2017.

Sida 28: Utah State University. 2008. Temperature-Moisture Relationship. [http://ocw.usu.edu/Forest_Range_and_Wildlife_Sciences/Wildland_Fire_Management_and_Planning/Unit_4_Temperature-Moisture_Relationship_4.html]. 25/11 2017.

Sida 31: NASA. 2006. Spinoff: Progressive Plant Growing Has Business Blooming. [https://www.nasa.gov/pdf/164449main_spinoff_06.pdf]. 10/1 2018.

Datablad

Sida 10: Aosong Electronics Co. Temperature and humidity module. DHT11 Product Manual, http://akizukidenshi.com/download/ds/aosong/DHT11_20171226.pdf

Sida 13: Lucky Reptile. Mini Nebler Fogging Device Mini Brumisateur, www.hochrep.com/html/seiten/support_download.php

Sida 16: Adafruit. 2017. Data Logger Shield. <https://cdn-learn.adafruit.com/downloads/pdf/adafruit-data-logger-shield.pdf>

Sida 20: Texas Instruments. 1997. ULN2803A Darlington Transistor Arrays. <http://www.ti.com/lit/ds/symlink/uln2803a.pdf>

Sida 21: Luxorparts. 2016. Energimätare. <https://www.kjell.com/se/sortiment/el-verktyg/el-produkter/starkstrom/energimatare/luxorparts-energimatare-p50003>

Appendix

Arduino kod:

```
// libraries definition

#include <SPI.h> #include <SD.h> #include <Wire.h> #include "RTClib.h" #include "DHT.h"

// how many milliseconds between grabbing data and logging it. 1000 ms is once a second #define LOG_INTERVAL 1000 // mills between entries (reduce to take
more/faster data)

// how many milliseconds before writing the logged data permanently to disk// set it to the LOG_INTERVAL to write each time (safest)// set it to
10*LOG_INTERVAL to write all data every 10 datareads, you could lose up to // the last 10 reads if power is lost but it uses less power and is much faster!
#define SYNC_INTERVAL 1000 // mills between calls to flush() - to write data to the card uint32_t syncTime = 0; // time of last sync()

#define ECHO_TO_SERIAL 1 // echo data to serial port

#define DHTPIN 2 #define DHTTYPE DHT11 //pins definition

// the digital pins that connect to the LEDs #define greenLEDPin 3#define redLEDPin 4

// Darlington Array;#define mistPin 5#define pumpPin 6#define soilSensorEnablePin 8 #define fanPin 9

#define lightPin 11

// for the data logging shield, we use digital pin 10 for the SD cs line #define chipSelect 10

//Sensors#define soilSensorPin A0

//defines what hours the pump and LED is allowed to run #define morning 9#define evening 21

#define wateringInterval 3000 //time pump stays on at a time #define wateringWaitingInterval 3000 //time waiting for water to sink in

//variablesint soilMoistureSensorValue; int soilMoistureLimit = 700;

float humiditySensorValue; int humidityLimit = 50;

int temperatureSensorValue;

bool startPump = false; int pumpTime = 0;

// stores the moisture sensor values

bool startFan = false; int fanTime = 10000; bool startLight = false; bool startMist = false; int mistTime = 0;

bool dayTime = false;

RTC_DS1307 RTC; // define the Real Time Clock object DHT dht(DHTPIN, DHTTYPE); // define the sensor object

DateTime now;

// the logging file File logfile;

//DateTime now = RTC.now();

void readSensors() { digitalWrite(soilSensorEnablePin, HIGH);delay(30);soilMoistureSensorValue = analogRead(soilSensorPin);
digitalWrite(soilSensorEnablePin, LOW);

humiditySensorValue = dht.readHumidity(); temperatureSensorValue = dht.readTemperature();

// Serial.println("humiditySensorValue: "); // Serial.println(humiditySensorValue);
```

```

}

void calculate() {if (humiditySensorValue < humidityLimit) {

startMist = HIGH;

mistTime = 6000; }else{

startMist= LOW; }

if (soilMoistureSensorValue > soilMoistureLimit){ startPump = HIGH;pumpTime = 2000;

} else { startPump = LOW;

}

int H = now.hour(); int M = now.minute();

//(H == 10 && M == if(H == 10 && M == startFan = HIGH;

}

// (H > morning if(H > morning && H

dayTime = true; }

0) 0){

&& H < evening) < evening){

}void logData() {

#if ECHO_TO_SERIAL Serial.print(m); Serial.print(", ");

#endif

// fetch the timenow = RTC.now();// log timelogfile.print(now.unixtime()); // seconds since 1/1/1970 logfile.print(", ");

logfile.print(""); logfile.print(now.year(), DEC); logfile.print("/"); logfile.print(now.month(), DEC); logfile.print("/"); logfile.print(now.day(), DEC);

logfile.print(" "); logfile.print(now.hour(), DEC); logfile.print(":"); logfile.print(now.minute(), DEC); logfile.print(":"); logfile.print(now.second(),

DEC); logfile.print("");

#if ECHO_TO_SERIALSerial.print(now.unixtime()); // seconds since 1/1/1970 Serial.print(", ");Serial.print("");Serial.print(now.year(),

DEC);Serial.print("/");Serial.print(now.month(), DEC);Serial.print("/");Serial.print(now.day(), DEC);Serial.print(" ");Serial.print(now.hour(),

DEC);Serial.print(":");Serial.print(now.minute(), DEC);Serial.print(":");Serial.print(now.second(), DEC);Serial.print("");

#endif //ECHO_TO_SERIAL

logfile.print(", "); logfile.print(dayTime);

logfile.print(", "); logfile.print(startPump);

logfile.print(", "); logfile.print(soilMoistureSensorValue);

digitalWrite(greenLEDPin, HIGH); // log milliseconds since starting uint32_t m = millis(); logfile.print(m); logfile.print(", ");

// milliseconds since start // milliseconds since start

logfile.print(", "); logfile.print(startMist);

logfile.print(", "); logfile.print(humiditySensorValue);

logfile.print(", "); logfile.print(startFan);

logfile.print(", "); logfile.println(temperatureSensorValue);

#if ECHO_TO_SERIAL

Serial.print(", "); Serial.print(dayTime);

```

```

Serial.print(", "); Serial.print(startPump);

Serial.print(", "); Serial.println(soilMoistureSensorValue);

Serial.print(", "); Serial.print(startMist);

Serial.print(", "); Serial.print(humiditySensorValue);

Serial.print(", "); Serial.print(startFan);

Serial.print(", "); Serial.println(temperatureSensorValue);

#endif //ECHO_TO_SERIAL

digitalWrite(greenLEDPin, LOW);

// Now we write data to disk! Don't sync too often - requires 2048 bytes of I/O to SD card // which uses a bunch of power and takes timeif ((millis() -
syncTime) < SYNC_INTERVAL) return;syncTime = millis();

// blink LED to show we are syncing data to the card & updating FAT! digitalWrite(redLEDPin, HIGH);logfile.flush();digitalWrite(redLEDPin, LOW);

}

void runLights(bool runLights){ digitalWrite(lightPin, runLights);

}

void runMist() { if (startMist) {

analogWrite(mistPin, 255); delay(mistTime); analogWrite(mistPin, 0);

}

startMist = false; //

// analogWrite(fanPin, 250); // delay(300);// analogWrite(fanPin, 0);

}

void runPump() { if(startPump){

analogWrite(pumpPin, 255); delay(pumpTime); analogWrite(pumpPin, 0);

}

startPump = false; }

void runFan() {

if(startFan){ analogWrite(fanPin, 255); delay(fanTime); analogWrite(fanPin, 0);

}

startFan = false; }

void execute() {

if(dayTime){ runMist(); runPump(); runFan(); runLights(true);

}else{startPump = false; startMist = false; startFan = false; runLights(false);

}

}

void error(char *str) {

Serial.print("error: "); Serial.println(str);

```



```

// red LED indicates error while (1);
}

void setup() {
// serial initialization Serial.begin(9600); Serial.println();

// Arduino pins initialization pinMode(mistPin, OUTPUT); pinMode(pumpPin, OUTPUT); pinMode(fanPin, OUTPUT); pinMode(lightPin, OUTPUT);
pinMode(soilSensorEnablePin, OUTPUT);

// // use debugging LEDs pinMode(redLEDpin, OUTPUT); pinMode(greenLEDpin, OUTPUT);

//start humidity sensor dht.begin();

// initialize the SD cardSerial.print("Initializing SD card...");// make sure that the default chip select pin is set to // output, even if you don't use
it: pinMode(chipSelect, OUTPUT);

// see if the card is present and can be initialized: if (!SD.begin(chipSelect)) {
error("Card failed, or not present"); }

Serial.println("card initialized.");

// create a new filechar filename[] = "LOGGER00.CSV"; for (uint8_t i = 0; i < 100; i++) {
filename[6] = i / 10 + '0'; filename[7] = i % 10 + '0'; if (! SD.exists(filename)) {

// only open a new file if it doesn't exist logfile = SD.open(filename, FILE_WRITE); break; // leave the loop!
} }

if (! logfile) { error("couldnt create file");
}

Serial.print("Logging to: "); Serial.println(filename);

// connect to RTC Wire.begin();if (!RTC.begin()) {
logfile.println("RTC failed"); #if ECHO_TO_SERIAL
Serial.println("RTC failed"); #endif //ECHO_TO_SERIAL
}RTC.adjust(DateTime(F(__DATE__), F(__TIME__)));

//First written to SD card

logfile.println("millis,stamp,datetime,dayTime, runPump?,soilmoist,runMist?,humidity,startFan?,temp"); #if ECHO_TO_SERIAL
Serial.println("millis,stamp,datetime,dayTime, runPump?,soilmoist,runMist?,humidity,startFan?,temp"); #endif //ECHO_TO_SERIAL
}void loop() {

// RTC parameters definition

//// int H = now.hour();

// int M = now.minute(); // int S = now.second();

startMist = true; mistTime = 5000;

if (true) { readSensors(); calculate(); logData(); execute();
}

delay(1000); }

```

