



CHALMERS



Kalibrering av ljussensor för Parans solpanel

Examensarbete i högskoleingenjörsprogrammet för datateknik

PÄR SVEDBERG
OSKAR ÅKERGREN

EXAMENSARBETE

Kalibrering av ljussensor för Parans solpanel

PÄR SVEDBERG
OSKAR ÅKERGREN

Institutionen för data- och informationsteknik
CHALMERS TEKNISKA HÖGSKOLA

Göteborg, Sverige 2015

**Kalibrering av ljussensor för
Parans solpanel**
PÄR SVEDBERG
OSKAR ÅKERGREN

© PÄR SVEDBERG, OSKAR ÅKERGREN, 2015

Institutionen för data- och informationsteknik
Chalmers tekniska högskola
SE-412 96 Göteborg
Sverige
Telefon: +46 (0)31-772 1000

Omslag:
Parans solpaneler modell SP3 installerade på taket till Centrum för vuxnas lärande i Växjö
Fotograf: Oskar Åkergren

Abstract

The aim of this project is to automatically calibrate a photo sensor on a sun panel that is following the sun in its path, to maximize the light intake that is to be used as a light source in doors, a calibration that was originally made by hand. The project was performed in iterations according to a design science research method and has two main focus tasks. The first is to develop a calibration algorithm and the second is suggest a communication solution between the room and the panel. The result of the project is a fully functioning calibration application, and two suggestions for the communication where the fibre optic cables is used to transport the data.

Keywords: Calibration, RS-232, serial communication, optical commication, fibreoptics, microcontroller, indoor lighting, Parans, light sensor

Sammandrag

Detta projekt syftar till att möjliggöra en automatisk kalibrering av fotosensorn på en solpanel som aktivt följer solen för att maximera dess solintag som nyttjas till belysning, en kalibrering som tidigare utförts manuellt. Projektet har genomförts i iterationer med stöd av en 'design science research'-metod och är fokuserat på två huvudsakliga områden, dels en effektiv kalibreringsalgoritm och dels en kommunikationslösning mellan solpanelen och det rum som panelen lyser upp. Projektet har resulterat i en färdig mjukvarulösning som kalibrerar panelen och projektet har lämnat två förslag på lösningar gällande kommunikationen, där vi rekommenderar att använda panelens egna fiberkablar som datamedium.

Nyckelord: Kalibrering, RS-232, seriell kommunikation, optisk kommunikation, fiberoptik, mikrokonroller, belysning, Parans, ljussensor

Förord

Detta examensarbete skrivs inom ramen för kursen LMTX38, Examensarbete vid Data- och informationsteknik vårterminen 2015 på Chalmers tekniska högskola.

Författarna vill rikta ett stort tack till Parans Solar Lighting AB för tillhandahållande av materiel som projektet nyttjat sig av, lokal att arbeta i och stöd vid arbetet. Ett särskilt tack riktas till handledare Karl Nilsson vid Parans för förslaget till examensarbetets problemområde och avsatt tid för handledning av projektet. Vi riktar även ett tack till Simon Larsson vid Parans för bollande av idéer och stöttande vid utvecklingsarbetet. Vidare vill vi tacka handledare Lennart Hansson vid Chalmers för hans stöd vid skrivande av denna rapport och förslag för att driva arbetet framåt. Ett avslutande tack skickas till Sakib Sisteck vid Chalmers för hans förmedling av kontakten till Parans och hans hjälp vid förberedningen till detta examensarbete.

Beteckningar

C	Programmeringsspråk
I/O	Input/Output
I ² C	Standard för synkron seriell datakommunikation
lm	lumen, SI-enhet för ljusflöde
lux	SI-enhet för belysning, $1 \text{ lux} = 1 \text{ lm/m}^2$
RS-232	Standard för asynkron seriell datakommunikation
UART	Universal Asynchronous Receiver/Transmitter, gränssnitt för seriell kommunikation

Innehåll

1	Introduktion	1
1.1	Bakgrund	1
1.2	Syfte	1
1.3	Mål	2
1.4	Frågeställning	2
1.5	Avgränsningar	2
1.5.1	Hårdvara	2
1.5.2	Mjukvara	2
2	Metod	3
2.1	Vetenskaplig metod	3
2.2	Arbetsmetodik	4
3	Teknisk bakgrund	5
3.1	Parans SP3	5
3.1.1	Mikrokontrollerkortet	5
3.1.2	Fiberoptik	5
3.1.3	Ljussensor	5
3.2	Luxmätare	6
3.2.1	Adafruit TSL2591 Digital Light Sensor	6
3.2.2	Yoctopuce Yocto-Light-V3	6
3.3	Arduino	7
4	Genomförande	8
4.1	Fas 1	8
4.2	Fas 2	8
4.3	Fas 3	9
4.3.1	Utveckling av kommunikationslösning	9
4.3.2	Utveckling av algoritm	10
4.3.3	Avläsning av luxmätare	13
4.3.4	Utveckling av applikation	13
4.4	Fas 4	16
5	Resultat	17
5.1	Algoritm	17
5.2	Optisk kommunikation	17
5.2.1	Förutsättningar	17
5.2.2	Tillförlitlighet	18
5.3	Applikation	19

6 Slutsats	21
6.1 Sammanfattning	21
6.2 Diskussion	21
6.3 Vidareutveckling	23

Referenser	24
-------------------	-----------

Bilagor	I
----------------	----------

A Simuleringsresultat av kalibreringsalgoritm	I
--	----------

B Flödesschema av kalibreringsalgoritmen	II
---	-----------

C UML Diagram	III
----------------------	------------

D Specifikationer	IV
--------------------------	-----------

D.1 Arduino	IV
-----------------------	----

D.2 Sändare	IV
-----------------------	----

D.3 Mottagare	IV
-------------------------	----

E Uppmätt fokuspunkt från SP3	V
--------------------------------------	----------

Figurer

1 Utformning av metod	3
---------------------------------	---

2 Exempel på förväntade fokuspunkter	8
--	---

3 Schema över sändare/mottagare	10
---	----

4 Jämförelse algoritmernas antal söksteg	11
--	----

5 Översikt av uppmätt fokuspunkt	12
--	----

6 Skiss av det grafiska gränssnittet	15
--	----

Tabeller

1 Minskning av antal söksteg mellan algoritmversioner	12
---	----

1 Introduktion

1.1 Bakgrund

Parans är utvecklare av en produkt som via optiska fibrer levererar naturligt solljus in i byggnader, som ett alternativ till dagens traditionella ljuskällor. Bolaget är baserat i Göteborg men levererar systemen globalt och har flera installationer runt om i världen.

Produkten fokuserar in solljus i optiska fibrer och styrs med hjälp av två stegmotorer för att följa solens bana. Styrningen sker dels med en algoritm som, baserat på geografisk position och tid, ger en solposition i grader och dels från en solsensor med fotocell som ger data för en finstyrning av panelens positionering då solen är framme. Detta för att alltid maximera solljuset som levereras in i fibern.

Styrkortet och motorerna till panelen drivs av en spänning om tolv (12) volt och kortet är en egen design kring mikrokontrollen PIC32. Källkoden till panelen är skriven i C och kommunikation till enheten sker via seriell förbindelse över USB, där en USB till RS-232 omvandlare är integrerad på styrkortet. För att skicka instruktioner till panelen används en terminalemulator.

Fotosensorn som används i solpanelen kan representeras som ett koordinatsystem, där sensorn förväntar sig att ljuset fokuseras till en punkt som träffar origo som standard. Problemet som Parans har är tvådelat, det första problemet att i tillverkning av panelen kan linsen fokusera ner ljuset något vid sidan av origo på sensorn, vilket leder till sämre ljusintag i de optiska fibrerna. Det andra problemet är att solen inte går att fokusera ner till en punkt, utan kommer alltid att representeras av en disk, vilket kan förvirra sensorn något och då även detta leda till sämre ljusintag i de optiska fibrerna.

Idag använder Parans en manuell metod för att kalibrera sensorn, flytta den punkt på koordinatsystemet som ljuset fokuserar ner till, genom att vrida solpanelen med hjälp av en terminalemulator och sedan kontrollera värdet på en separat luxmätare.

1.2 Syfte

Syftet med projektet är att ta fram en helt automatisk process som kan kalibrera fotosensorn i Parans solpaneler, så att det maximala ljusflödet från panelen kan uppnås och detta med en lägre tidsåtgång och högre precision än dagens manuella metod. Vidare syftar projektet till att möjliggöra kommunikation mellan panelen och en luxmätare inne i byggnaden, så att kalibreringstekniken kan nyttjas till systemen generellt, oavsett om de är i produktion eller i fabrik.

1.3 Mål

Målet med det här projektet är att ta fram ett automatiskt system som justerar fokuspunkten på ljussensorn, vilket då vrider på solpanelen för att lokalisera det X- och Y-värde där intaget av solljus är som störst. Ljusstyrkan mäts med hjälp av en luxmätare som levererar ljusintaget till en dator eller till en annan programmerbar enhet. När det maximala ljusintaget är uppmätt, registreras X- och Y-värdena som den nya fokuspunkten för ljussensorn, istället för det förinställda värdet på origo. Vidare är målet att ta fram en fram av kommunikation mellan en luxmätare inne i byggnaden och en panel som befinner sig på taket, så att även enheter som redan är satta i bruk kan kalibreras.

1.4 Frågeställning

- Vilken algoritm kan anses vara lämplig för kalibreringen?
- Vilka förutsättningar för kommunikation finns det mellan solpanelen och det upplysta rummet?
- Hur tillförlitligt är det valda kommunikationssättet?

1.5 Avgränsningar

1.5.1 Hårdvara

Redan existerande hårdvara kommer att användas, det vill säga sådan avsedd att användas för de ändamål nödvändiga för projektet. Den primära hårdvaran, solpanel och luxmätare, kommer att tillhandahållas av uppdragsgivaren och inga alternativ till dessa kommer att undersökas. Eventuell övrig hårdvara kan antingen vara helhetslösningar eller sådana som löser delproblem och kombineras. Gällande lösningen som tas fram är den begränsad till att stödja företagets nu gällande panel SP3.

1.5.2 Mjukvara

Mjukvara kommer att utvecklas för att nå projektets uppsatta mål. Denna kan komma att inkludera användning av båda medföljande och externa ramverk och bibliotek för att lösa olika delproblem, exempelvis grafisk framställning och kommunikation mellan olika enheter.

2 Metod

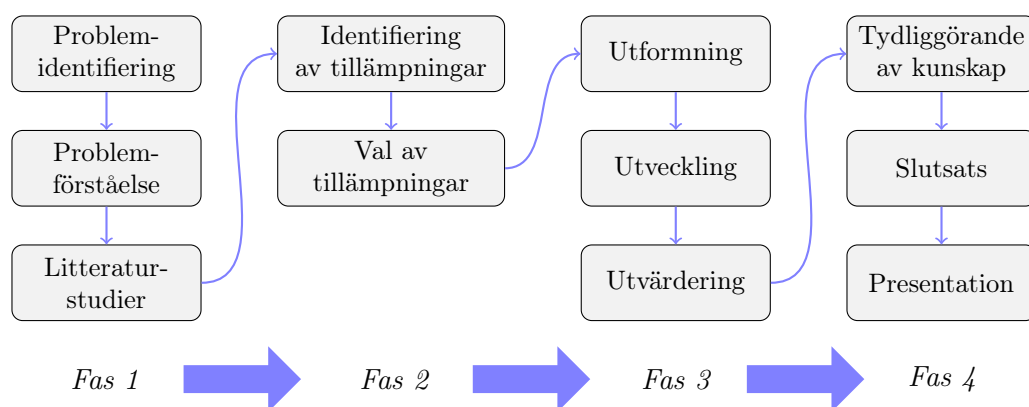
2.1 Vetenskaplig metod

Detta projekt har tillämpat en variant av den vetenskapliga metoden Design Science Research (DSR). Metoden anses lämplig till problemlösande forskning där redan existerande produkter ska vidareutvecklas [1]. Målet med DSR är att skapa artefakter, exempelvis en praktisk lösning, metod eller lösningsförslag, som löser de problem som identifierats inom projektet.

Design Science valdes då dess mål stämmer bra överens med det som projektet syftar till att göra. Detta kan sättas i kontrast med mer traditionella vetenskaper som snarare syftar till att utforska, förklara eller förutse fenomen [1, s. 13]. Att DSR valdes som metod över fallstudier eller 'action research' är återigen att målen överensstämmer med projektet, men även att typen av kunskap som anskaffas stämmer bättre överens än de andra två alternativen [1, s. 95].

Dresch et al. rekommenderar, baserat på studier av flera metoder för DSR, en metod i 12 steg [1, s. 118–126]. De tre inledande stegen är en analys av de problem som ska lösas, problemidentifiering, problemförståelse och litteraturstudier. Denna inledande fas mynnar ut i att hitta eventuella befintliga lösningar som kan vara lämpliga och att sedan föreslå en vidareutveckling och tillämpning av denna eller att föreslå en ny lösning. Steg sex till åtta är sedan att utforma, utveckla och utvärdera lösningen. Därefter ska den kunskap som givits av tidigare steg tydliggöras och slutsatser dras. Tidigare nämnda steg itereras vid behov för att uppnå önskat resultat. Slutligen ska generalisering av lösningen utformas och resultatet presenteras.

Ovan nämnda metodik har för detta projekt förenklats något för att anpassas till projektets storlek och omfattning. Metoden indelas i fyra faser, enligt figur 1, där fas 1–3 itereras efter behov.



Figur 1: Utformning av metod

2.2 Arbetsmetodik

Projektet arbetsmetodik utgick ifrån versionshanteringsverktyget 'git' för den mjukvara som projektet använde sig av. För att få tillgång till en central hantering av dokumenten använde sig projektet av 'GitHub.com' vilket även bistod med ett grafiskt gränssnitt till git, då git i sig själv endast har ett textbaserat gränssnitt.

Vidare var arbetsmetodiken inspirerad av 'Scrum' där större mål sattes upp och bröts ner till mindre så kallade 'issues' [2, kap. 8]. Dessa issues sattes upp på en virtuell tavla med hjälp av verktyget 'Waffle.io' för att få en bättre överblick kring hur projektet utvecklades och vad som behövde göras.

Anledningen till att inte hela Scrum-metodiken anammades var att projektet utfördes av få personer så den rollfördelning som hör till i Scrum gick ej att utföra på något meningsfullt vis [2, kap. 6], samt att ovanan vid denna typ av utveckling gjorde att kostnaderna för varje issue var svårt att bestämma. Vidare var projektets omfång väl avgränsat av uppdragsgivaren så de avgränsningarna användes som milstenar (inbyggd funktion i GitHub) istället för de föreslagna användarberättelserna [2, kap. 9].

3 Teknisk bakgrund

3.1 Parans SP3

SP3 är tredje generationens solpanel utvecklad av Parans [3]. Panelen monteras på utsidan av en byggnad, ofta på taket, och fokuserar solljus genom linser in i optisk fiber för att sedan genom armatur lysa upp inomhus. Varje panel har sex utgående kablar med fiberoptik, vardera ansluten till en armatur, vilkas räckvidd är upp till 20 meter. Panelen är utformad så att ultraviolett och infrarött ljus avskärmas från det ljus som leds in i fibern. Två stegmotorer används för att justera panelens riktning horisontellt och vertikalt, styrda av ett mikrokontrollerkort, så att linserna alltid är vända mot solen. Motorernas rörelser bestäms av en algoritm i mjukvaran som räknar ut solens nuvarande position med hjälp av att kombinera mätvärden från en ljussensor och solens förväntade position på himlen baserat på tid, datum och installationsplatsens geografiska position angivet i longitud och latitud. Mjukvaran som körs på mikrokontrollern är skriven i C.

3.1.1 Mikrokontrollerkortet

Mikrokontrollerkortet som används i panelen är formgivet av Parans och är baserat på en PIC32 mikrokontroller. PIC32 är en kategori mikrokontroller tillverkade av Microchip Technology för användning i inbyggda system och ger tillgång till bland annat flera I/O-anslutningar och UART för seriell kommunikation [4, s. 42–44]. För att kommunicera med mikrokontrollerkortet med en dator finns en USB-port som ger en seriell anslutning som hanteras av en UART-krets från Silicon Laboratories, CP2102. Detta kräver att den anslutna datorn har en drivrutin för CP2102 installerad och möjliggör anslutning via en terminalemulator för installation, diagnostik och underhåll.

3.1.2 Fiberoptik

Den fiberoptiska kabel som i dagsläget används av Parans är en plastfiber som har en ljusöverföring om 96 % per meter och saluförs i fyra olika längder, 5, 10, 15 och 20 meter [5]. Var kabel består av sex stycken fibrer och ger ett ljusflöde om 730 lm till 430 lm från solpanelen i fullt soljus, beroende på längd av kablagen. Vid fibrernas paneländar finns IR-speglar monterade som reflekterar den infraröda delen av solljuset och de används för att fibern annars riskerar att smälta.

3.1.3 Ljussensor

För att optimera ljusintaget i panelens fibrer, finjusteras vinkeln till solen med hjälp av en ljuskänslig krets monterad på panelens front. Kretsen skyddas av ett gråfilter med 10 % ljusgenomsläpp för att dämpa solljusets intensitet. Via en lins fokuseras genomträngande ljus till en fokuspunkt på sensorn som ger ett utslag och kretsen omvandlar denna data till fyra strömmar vilkas värden representerar avståndet från

sensors mittpunkt, sett ur en programmerares synvinkel hanteras x- och y-värden på ett koordinatsystem. Själva ljussensorn är tillverkad av Hamamatsu, med beteckningen S5901, och sitter integrerad på en egendesignat mönsterkort. Det exakta databladet för den använda sensorn har ej kunnat lokaliseras men enligt den information som finns inom Parans kan den antas arbeta efter samma princip som sensorerna S5990-01 och S5991-01 [6].

3.2 Luxmätare

För att mäta upp ljusstyrkan från panelen används en lux-mätare kopplat till en av de fiberkablar som leder ner till det upplysta rummet. De luxmätare som stöds av projektet är mätare som är till för privatbruk och är inte att anses som professionella i det avseendet att ljusmiljön i ett rum kan bestämmas med hjälp av dem. Syftet med mätarna är istället att registrera skillnaden i ljusstyrkan från panelen, när panelen är rätt kalibrerad kommer luxmätaren leverera ett högre värde än vid en felkalibrerad panel. Det exakta värdet är inte av intresse i detta fall, utan endast möjligheten att hitta den inställning på panelen som ger det maximala luxvärdet hos mätaren.

3.2.1 Adafruit TSL2591 Digital Light Sensor

Sensorkortet från Adafruit innehåller ljussensorn TSL2591 från ams och används till att uppmäta ljusintensitet upp till 88 000 lux och kan anslutas till en mikrokontroller via I²C. På sensorn finns två fotodioder, där den ena reagerar på IR-ljus och den andra reagerar på fullspektrumljus [7]. Fotodiodernas avläsning kan ske oberoende av varandra eller kombineras vilket möjliggör avläsning av endast synligt ljus. Sensors värden avläses digitalt.

3.2.2 Yoctopuce Yocto-Light-V3

Yocto-Light-V3 är en luxmätare i form av ett kretskort, baserad på ljussensorn BH1751FVI från ROHM, som är avsedd att mäta synligt ljus upp till 100 000 lux [8]. Kretskortet har en USB-port för anslutning till dator och kräver ingen extra drivrutin mer än de som medföljer vanliga operativsystem för att användas. Tillverkaren Yoctopuce tillhandahåller kodbibliotek till flera vanligt förekommande programmeringsspråk som möjliggör avläsning av sensorvärden.

3.3 Arduino

Arduino Uno är ett mikrokontrollerkort som innehåller mikrokontrollern ATmega328, USB-anslutning och 20 pins, 6 analoga och 14 digitala, för att ansluta externa enheter och kringutrustning [9, kap. 1]. Mikrokontrollern är en del av AVR-serien från Atmel och den har en RISC-baserad processor, 32 KB flashminne för lagring av programkod och 2 KB internminne. ATmega328 har också en UART som möjliggör seriell kommunikation enligt RS-232. Plattformen för Arduino, där Uno är en implementering, är öppen och det är fritt för var och en att bygga sina egna mikrokontrollerkort enligt tillgängliga scheman. Till Arduino tillhandahålls även en tillhörande utvecklingsmiljö som möjliggör programmering via en dator och ger tillgång till exempelkod och färdigskrivna kodbibliotek. Kod till Arduinoenheter skrivs i Arduinos egen implementation av språken C och C++ [10, kap. 2, 11]. För fullständig specifikation, se bilaga D.1.

4 Genomförande

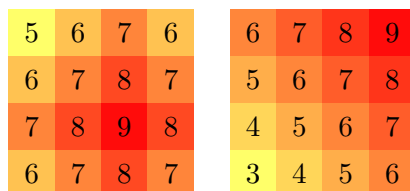
4.1 Fas 1

Projektet genomfördes med stöd av den valda metoden. I problemidentifikations-fasen, fas ett, hölls möten med uppdragsgivaren i syfte att få en enhällig uppfattning om vad företaget efterfrågade och formaliserade det praktiska problem som företaget sökte en lösning till. Vidare är det även i den här fasen som introduktionen till projektets rapport utvecklades för att fastslå vad projektet avser att utföra, som ett led i problemidentifieringen.

Den initiala problemanalysen resulterade i att projektet i stort kommer vara uppdelat i två mindre delar, dels den algoritm som klarar av att kalibrera panelen och dels en kommunikationslösning mellan panelen och rummet som den levererar ljuset till.

Förutsättningen vid litteraturstudien, gällande kommunikationen mellan taket och byggnadens innandöme, var att den trådlösa kommunikationen skall ske med standardiserade protokoll. Detta för att underlätta mottagandet av den trådlösa sändningen, i syfte att undvika tidssänken i felsökning då projektet har en relativt snäv tidsram.

För kalibreringsalgoritmens del bestod problemförståelsesteget av att undersöka vilka typer av datastrukturer som skulle komma att beröras. När problemet analyserades insågs att värdena som samlas in kan representeras som en tvådimensionell matris (eng. 'array'), där det finns ett unikt maxvärde och kring detta minskande värden som blir lägre ju längre från maxvärdet de befinner sig, se figur2.



Figur 2: Exempel på förväntade fokuspunkter

4.2 Fas 2

Litteraturstudien resulterade i en förståelse att trådlösa standarder för datakommunikation så som 802.11 standarderna har problem att sända när betongkonstruktioner hindrar utspridningen av radiovågorna och kräver speciell apparatur för att klara av att skicka data igenom sådana förhållanden [12]. Detta medför att trådlös kommunikation inte är lämplig för företaget, då de på förhand inte kan veta ifall deras kommunikation kommer att fungera på plats hos deras kunder. Inköp av nämnda apparatur är inte aktuellt.

Ett lämpligare medium att kommunicera via är istället de fiberoptiska kablar som redan är dragna, då rummet lysas upp av just dessa kablar. Enligt företaget kommer det finnas mer än en fiberkabel dragen till varje rum, vilket öppnar upp för möjligheten att koppla in apparatur för kommunikation i en fiberkabel, medan den eller de andra kablarna kan fortsätta hämta in ljus till rummet. Med de svårigheter som den trådlösa kommunikationen medförde i kombination med att ett fungerande alternativt medium redan finns draget, valde projektet att fokusera på det senare. När ”val av tillämpningar” genomfördes för kommunikationen, konstaterades det att panelens fiberändar har ett reflekterande skydd mot infrarött ljus, enligt 3.1.2, och att ultraviolett ljus inte leds genom panelens yttre glasskiva, vilket leder till att kommunikationen över fiberkabeln måste ske med ljus i det synliga spektrumet.

Gällande kalibreringsalgoritmen framstod det vid datorkörningar att en algoritm som itererar över hela matrisen för att leta det högsta värdet kommer att vara väldigt ineffektiv. Ett beslut fattades om att utveckla en algoritm som kräver så få steg som möjligt, detta då den praktiska implementationen kommer att innebära fördröjningssteg vid två punkter i körningen, dels när panelen flyttar på sig, och dels när ljuset ska hämtas in från luxmätaren. Algoritmen ska kontinuerligt söka efter ett högre värde tills det maximala värdet är funnet, likt figur 2.

4.3 Fas 3

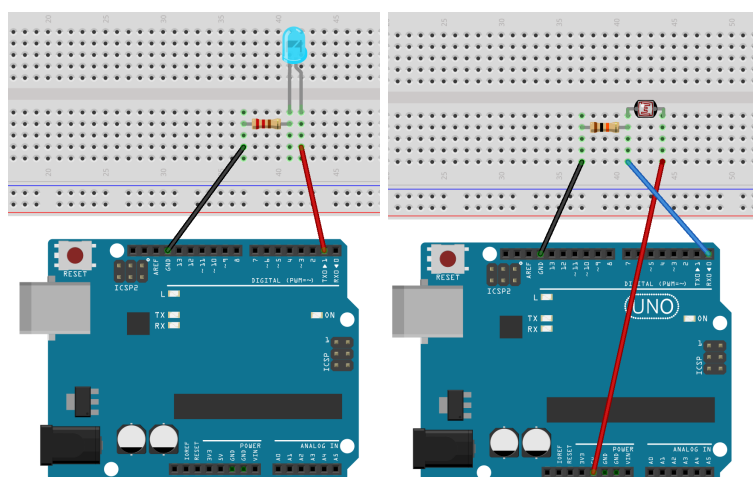
Den tredje fasen i metoden syftar till själva utvecklingen artefakter, det vill säga att faktiskt framställa någonting som går att använda. Projektet spenderade största delen av den projekttiden i denna fas, då varje artefakt utvecklades genom iterationer av de tre delstegen utformning, utveckling och utvärdering. Utveckla en del i artefakten och sedan utvärdera delen för att undersöka hur den beter sig och vad som kan göras bättre.

4.3.1 Utveckling av kommunikationslösning

Kommunikationsdelen i projektet har genomgått flera iterationer av fas 3 och två huvudlösningar utformades.

Den första lösningen bestod av två uppsättningar av mikrokontrollerkortet Arduino Uno, för en fullständig specifikation se bilaga D.1 [13]. Till sändaren kopplades en lysdiod till det gränssnitt som skickar data via den seriella standarden, vilket då omvandlade från RS-232 standardens höga och låga läge till ljus på och ljus av. Till mottagaren kopplades en fotoresistor som ändrar motståndet när den träffas av ljus. När den kopplades in till det seriella gränssnittet skapade resistorn spänningsförändringar som registrerades som högt eller lågt värde av standarden. För en enkel översikt av kopplingen se figur 3. Fullständig specifikation finns i bilaga D.

För att möjliggöra överföringen användes den i förhållandevis låga överföringshastigheten om 300 baud, vilket gjorde att komponenterna (fotoresistorn och lysdioden)



Figur 3: Schema över sändare/mottagare.
Figuren skapad med hjälp av verktyget 'Fritzing'

hann ändra sina värden under den tid de förväntades göra det, då dessa komponenter inte var särskilt utformade för denna uppgift och innehöll vissa fördröjningar vid skifte av tillstånd.

Den data som avsågs skickas i denna kommunikation bestod av ett 16 bitars värde, det vill säga två byte. Med en överföringshastighet om 300 baud ges en teoretisk möjlighet att skicka $\frac{300 \text{ baud}}{8 \text{ bit} \cdot 2} = 18,75$ värden per sekund. Den faktiska hastigheten blir något lägre då extra bitar skickas enligt RS-232 protokollet, men överföringshastigheten var inte att anse som ett hinder i denna implementation.

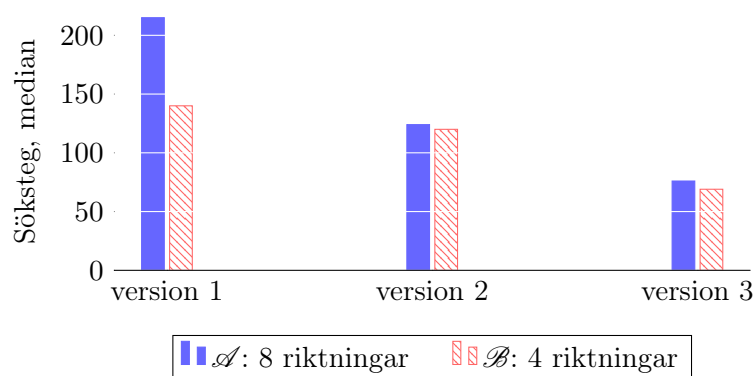
Den andra lösningen använde sig endast av en luxmätare uppe på taket för att inhämta den data som efterfrågades. Genom att koppla samman två fibrer nere i rummet kan ljuset skickas upp till taket igen och en rundgång i systemet kan skapas. För att nyttja den här metoden behöver samtliga linser som tillhör en fiberkabel täckas över på panelen för att förhindra att dubbelt intag av solljus transporteras i de båda fiberkablarna. Kablarna är anpassade för att hantera den värme som produceras av det ljus som normalt transporteras vid fullt dagsljus men om den dubbla mängden skulle transporteras riskerar fibrerna att smälta på grund av värmebildningen. Lösningen använder sig inte av fiber som bärare av genererad information utan endast det solljus som panelen tar in skickas, vilket minskar både komplexiteten i systemet och risken för störningar.

4.3.2 Utveckling av algoritm

Utformningen av algoritmen skedde genom flera iterationer av steg 3 i den valda metoden. I den första iterationen utvecklades en sökalgoritm, algoritm \mathcal{A} , som inledningsvis genomsöker en 3x3-matris i syfte att finna en inledande sökriktning.

Därefter genomförs sökning medurs i åtta riktningar med utgångsriktning motsvarande österut. När ett större värde påträffas uppdateras nuvarande position och sökningen återupprepas. För jämförelse utvecklades även en variant med fyra sökriktningar, algoritm \mathcal{B} . Efterkommande iterationer var samtliga en vidareutveckling av den förra. Den andra iteration vidareutvecklade algoritmen så att den lagrar information om tidigare besökta koordinater, så att dessa ej undersöks vid upprepade tillfällen. Ytterligare förbättringar implementerades i den tredje iterationen, då riktningen på den senaste förflyttningen, då ett nytt större värde påträffats, registreras. Med hjälp av denna information undersöks först samma riktning som den senast lyckade förflyttningen innan medurs sökning. Här slopades också den inledande sökningen i 3x3 matris, då den inte kunde påvisas ha några märkbara fördelar. Samtliga iterationer innebar märkbara förbättringar vid simulering. Den slutgiltiga algoritmen arbetar enligt det flödesschema som finns presenterad i bilaga B.

För att jämföra sökalgoritmer genomfördes simuleringar för att mäta det antal steg som behövs för att ta sig från utgångspositionen till den position där det maximala värdet påträffades. Simuleringarna använde sig av 10 000 100x100-matriser där varje matris hade en slumpvis genererad utgångsposition och maximalt värde. Algoritmerna har alltså alla genomsökt samma matriser med samma utgångspositioner. Samtliga positioners värden var strängt avtagande i hänseende till avståndet från det maximala värdet. För varje iteration av algoritmen undersöktes sökning med både fyra och åtta sökriktningar. Sökning i fyra riktningar visade sig mer effektivt i samtliga fall, enligt figur 4 och bilaga A. Skillnaderna i antal steg mellan samma algoritm med olika antal sökriktningar vara störst i den första iterationen, med 54 % fler steg, men de visade sig även i övriga iterationer. I den tredje och slutgiltiga iterationen var skillnaden 10 % fler steg. Varje iteration av algoritmen minskade det antal steg som behövdes för att finna det största värdet. Största förändringen mellan iterationer skedde mellan iteration två och tre, där antalet steg minskades med 43 %, se tabell 1. Minskningen mellan iteration ett och tre var 51 %.



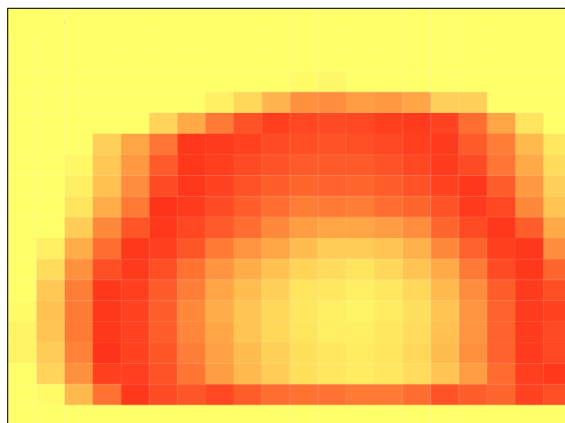
Figur 4: Jämförelse algoritmernas antal söksteg

Från	Till version 2	Till version 3
Version 1	14 %	51 %
Version 2	-	43 %

Baserat på medianvärden, se bilaga A

Tabell 1: Minskning av antal söksteg mellan algoritmversioner

För att testa algoritmen på solpanelen genomfördes först en simpel genomsökning av panelens solintag till en tvådimensionell matris, i syfte att undersöka ifall utformningen av panelens fokuspunkt överensstämmer med projektets antagande i figur 2. Resultatet av denna sökning gav en annorlunda bild av hur fokuspunkten ser ut från panelen, vilket visas i figur 5. För fullständig data av denna sökning, se bilaga E. Trots att fokuspunkten inte är utformad på samma sätt som förutspått, fungerar sökalgoritmen så pass att den riktar in sig till det högsta funna värdet, dock på grund av panelens utformning av fokuspunkten finns det inget exakt högsta värde, utan algoritmen stannar i en punkt i den cirkel som panelen genererar. Algoritmen är inte utformad för att söka igenom hela cirkeln och sätta det högsta värdet, utan lokaliserar ett lokalt maximum. Detta ger ett högt belysningsvärde ut från panelen även om det kan finnas ett något högre värde existerar på motsatt sida av cirkeln. Skillnaden inom denna cirkel är dock så pass små att det rör sig om så små värdeskillnader att dessa kan anses ligga inom felmarginalen. Att utveckla en algoritm som arbetar sig igenom hela cirkeln är fullt möjlig, men det handlar om en avvägning om hur lång tid kalibreringen kommer ta, den nu gällande algoritmen är utvecklad för att hitta ett lokalt maximum på så få steg som möjligt.



Figur 5: Översikt av uppmätt fokuspunkt

4.3.3 Avläsning av luxmätare

De båda luxmätarna skiljer sig åt gällande det gränssnitt som de rapporterar data över, där mätaren från Adafruit kräver en koppling till mikrokontroller för tolkning av rapporterad data, medan mätaren från Yoctopuce kopplas in direkt till en enhet via USB, vilket då underlättar för inkoppling till en persondator. Även om luxmätarnas gränssnitt skiljer sig åt, beter sig de båda luxmätarna väldigt lika vid användning. De båda genererar ett positivt heltal som representerar det luxvärde som uppmätts och skickar ut denna data när en metod i respektives bibliotek anropas. Från en programmerares perspektiv är resultatet från- och användningen av de båda mätarna i praktiken identisk.

För avläsning av luxvärdet för kalibrering av panelen är det av intresse att samla in så mycket ljus som möjligt från en panel, men undvika bakgrundsljus från andra eventuella ljuskällor. Detta för att skapa mätbara skillnader mellan de olika kalibreringsstegen hos panelen. För att säkerställa maximalt ljusintag från den panel som ska kalibreras, placeras luxmätaren så nära fiberändan som möjligt, vilket i praktiken i detta projekt innebär tre till fyra centimeter från fibern. För att hålla mätaren stabil används olika metoder beroende på mätare, där den ena har monterats in i ett metallhölje som skruvas fast på fiberändan, medan den andra vilar i en armatur. Att metoderna skiljer sig åt beror på de olika utformningarna av mönsterkortet för luxmätarna och leder till att de absoluta värden som läses in av mätarna inte direkt kan jämföras.

Vid testning av luxmätarnas inrapporterade värden upptäcktes det att resultaten från solpanelen fluktuerar, detta kan bero på flera orsaker så som att panelen ständigt justerar för att följa solen, eller att något färdas förbi panelen som skuggar den momentant. Denna skiftning i värde är inget som uppfattas det mänskliga ögat, men syns på mätningarna. För att kompensera för detta när luxvärdet hämtas så registrerar applikationen fyra värden och medianvärdet rapporteras vidare, för att få en bättre uppskattning av värdet vid den nuvarande kalibreringen.

4.3.4 Utveckling av applikation

Efter att ha implementerat en algoritm som i simuleringar uppfyllde önskvärd funktion påbörjades utveckling av en applikation där den kan tillämpas. Förutom själva kalibreringsalgoritmen skulle applikationen ha egenskaper som kommunikation med panelen, ett användarvänligt grafiskt gränssnitt och inhämtning av värden från luxmätare. Programmeringsspråket som valdes för utveckling av applikationen var Python och valet hade stöd av flera motiveringar. Den första anledningen till att språket valdes var att företaget har kompetens och erfarenhet att utveckla i detta språk, då den nya panelen SP4 kommer att drivas av en källkod skriven i just Python. Att kompetens och kunskap finns inom företaget medför att applikationen kan underhållas av företaget själv och modifieras vid framtida behov. Vidare har företaget sedan tidigare utvecklat i detta programmeringsspråk, så att

utveckla applikationen i samma språk underlättar för en eventuell implementering av mjukvaran i de existerande produkterna. Mer generella fördelar med att utveckla i `Python` är att språket är plattformsoberoende och är enkelt att utveckla grafiska gränssnitt i. Nackdelar med språket är att det är långsamt i förhållande till andra språk så som `C` eller `Java` men då applikationen kommer att kommunicera med mekanik, vilket leder till flertalet inbyggda fördröjningar för rotation av panelen och inhämtning och sändning av information, ser projektet att språkets åverkan på hur snabb applikationen upplevs som försumbar [14].

För att kommunicera med panelen och den Arduino som mottar värden från luxmätaren Adafruit TSL2591 behövdes stöd för seriell kommunikation. Ett bibliotek som möjliggjorde seriell kommunikation med enheter anslutna till en dator var `pySerial`. Eftersom `pySerial` har stöd för operativsystem baserade på Windows, Linux och BSD är biblioteket i praktiken plattformsoberoende [15].

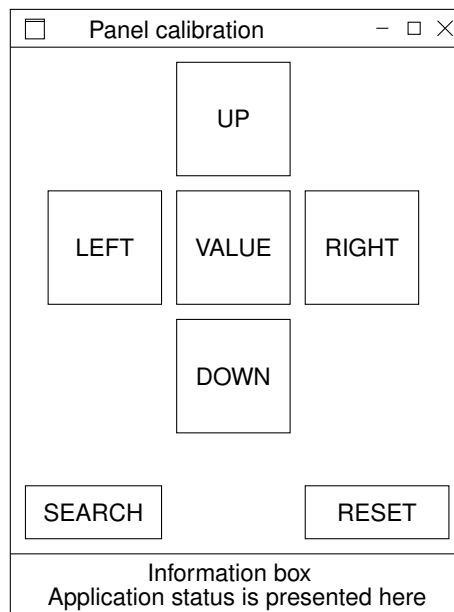
Ett beslut behövdes tas om ramverk för grafiskt gränssnitt. Till ändamålet fanns flera ramverk utvecklade och anpassade till `Python`. Det som ansågs mest lämpligt för applikation var `TkInter`, där en anledning är att Parans sedan tidigare har kringutrustning som använder `TkInter` [16]. Hänsyn togs till att underlätta både framtida hantering av koden för bolaget och en eventuell integrering av tidigare produkter med den av projektet utvecklade applikationen. En annan anledning är att ramverket distribueras med standardinstallationer av `Python` till Windows och Mac OS X [17]. På Linuxsystem behövs `TkInter` installeras separat och det finns tillgängligt i flera stora Linuxdistributioners pakethanterare. Behovet av extra installationer för ett ramverk kunde minimeras och det grafiska gränssnittet kunde göras plattformsoberoende.

Till luxmätaren Yocto-Light-V3, som anslöts direkt till den dator som kör applikationen, fanns kodbibliotek för `Python` att tillgå från tillverkaren Yoctopuce.

För att underlätta framtida hantering av den producerade koden eftersträvades en modulär uppbyggnad av applikationen. Inledningsvis isolerades den framtagna algoritmen i en egen klass, `Search`. Anslutningarna till panelen och de olika luxmätarna implementerades i separata klasser med specifik kod för att hantera varje enskild enhet. För att hantera de olika anslutningarna skapades klassen `SerialHandler` som ett mellanliggande gränssnitt till applikationens övriga komponenter. Det grafiska gränssnittet hanteras av klassen `GUI` och där skapades tryckknappar för att aktivera applikationens funktioner och ett fält för återkoppling och presentation av information. De tre klasserna `Search`, `SerialHandler` och `GUI` fick tillsammans utgöra grundstommen i applikationen och utformades för att vara plattformsoberoende. Genom att sökalgoritmen och det grafiska gränssnittet endast använder sig av det mellanliggande gränssnittet i `SerialHandler` och ej de underliggande anslutningarna uppnåddes en mer modulär design och ett oberoende av de implementationsspecifika delarna av applikationen. En översikt av klasserna finns i bilaga C.

Då applikationen är beroende av anslutning till externa enheter lades funktionalitet till för att underlätta anslutningen. Applikationen kan användas på olika plattformar och olika förfaranden framställdes för olika operativsystem. Stöd för automatisk identifiering av ansluten panel och Arduino implementerades till Mac OS X och Linux. Förutsättningen för detta var att inga fler externa enheter av samma typ anslutna är anslutna till den aktuella datorn. För Windowssystem utvecklades en dialogruta som vid applikationsstart frågar användaren efter de aktuella COM-portarna för respektive enhet. Om ingen Arduino är tillkopplad försöker applikationen använda sig av Yocto-Lux-V3.

Det grafiska gränssnittet gavs en enkel design och baserades på det utseende som används i företagets kringutrustning [16]. Nämnda kringutrustning var utformad för att kunna användas med pekskärm och således beaktades detta även här. Fyra knappar i form av ett styrkors lades till för att manuellt ändra panelens justeringsvärden för ljussensorn i x- och y-led. I mitten av styrkorset placerades en knapp för manuell avläsning av den anslutna luxmätaren. Nedanför styrkorset placerades en knapp som startar automatisk kalibrering och en knapp som återställer panelens justeringsvärden till de som var aktuella vid applikationens starttillfälle. Ett fält som kan förmedla aktuell information placerades i applikationens underkant. Det grafiska gränssnittet är avbildat i figur 6.



Figur 6: Skiss av det grafiska gränssnittet

4.4 Fas 4

Den fjärde fasen i den valda metoden avhandlar inte arbetsgången som sådan, utan visar på att resultatet från den tredje fasen ska analyseras och delges i syfte att sprida kunskapen vidare i vad som har uppnåtts. För detta projekt innebär fas fyra att skriva denna rapport vilket förtydligar och sammanfattar det resultat som har uppnåtts genom de iterationer som genomförts. Vidare hålls en presentation av resultatet inom ramen för den kurs som genomförs, vilket även är en del av metoden.

5 Resultat

5.1 Algoritm

För att möjliggöra automatisk kalibrering av panelens ljussensor har projektet utvecklat en sökalgoritm med positionsregistrering och stegreducering. Den utvecklade algoritmen gav svar på den första av projektets tre frågeställningar, "Vilken algoritm kan anses vara lämplig för kalibreringen?". Algoritmen uppsöker ett lokalt maximum i ljusstyrka, beskrivet i avsnitt 4.3.2, och fanns lämplig för ändamålet.

Algoritmen söker kontinuerligt stegvis efter det maximala inlästa värdet tills inga kringliggande större värden påträffas och justerar i varje söksteg panelens korrigeringsvärde för ljussensorn, vilket får panelen att vrida sig till den position som ger solljusets fokus i ljussensors korrigerade mittpunkt. Sökning sker i fyra riktningar, representerade av väderstrecksuttryck motsvarande den koordinatsystemsrepresentation panelen har för korrigeringsvärden där positiva x och y är öst respektive nord, och sker medurs med utgångsriktning österut. Om ett lika stort eller större värde avläses efter en vridning av panelen så kommer nästkommande undersökta position vara i samma riktning som den senast utförda, då avlästa värden kan representeras enligt figur 2. Algoritmen registrerar besökta positioner så att samma position ej undersöks upprepade gånger. Flödesschema för algoritmen finnes i bilaga B.

Tillgången till kontinuerligt solljus är en förutsättning för kalibrering av enheter tagna i bruk då variationer i molnighet markant påverkar ljusintensiteten och således det avlästa värdet. I händelse av längre tids molnighet deaktiveras ljussensorn av panelens mjukvara och kalibrering går då ej att genomföra. Om så sker under pågående kalibrering återställs panelens korrigeringsvärden till de värden som var aktuella innan den automatiska kalibreringen startade. För att ytterligare motverka oförutsedda problem vid kalibreringstillfället har kontroller för timeout och korrigeringsvärdernas rimlighet implementerats, där båda kontrollerna vid utslag avbryter sökningen och korrigeringsvärdena återställs.

5.2 Optisk kommunikation

För att svara upp mot målet att ta fram en lösning för "kommunikation mellan en luxmätare inne i byggnaden och en panel som befinner sig på taket" besvaras i efterföljande frågeställningar för förutsättningar för kommunikation och hur tillförlitliga dessa lösningar kan anses vara.

5.2.1 Förutsättningar

Frågeställningen "[v]ilka förutsättningar för kommunikation finns det mellan solpanelen och det upplysta rummet" har resulterat i en undersökning som visade att trådlös kommunikation inte är att anses som lämplig, utan nyttjande av den redan dragna optiska fiberkabeln är det kommunikationsmedia som bör nyttjas. Projektet föreslår

två olika metoder för att nyttja fibern som databärare, dels en metod där asynkron seriell data skickas och en annan metod där två optiska fibrer kopplas samman, för att på sådant sätt skicka ut ljusintaget genom panelens linser och där mäta upp ljusstyrkan.

Den första lösningen som projektet föreslog var en lösning där en mikrokontroller inne i det upplysta rummet omvandlar utdata från en luxmätare till en optisk signal som sedan sänds seriellt, enligt standarden RS-232, upp till panelen för att där avkodas av en andra mottagande mikrokontroller. När signalerna skickas via fibrerna strålas ljuset ut ur solpanelens linser, vilket mottagaren då kan analysera. Mottagaren är monterad på panelen och har en ljuskänslig sensor som omvandlar de optiska signalerna till digitala. De digitala signalerna skickas från mottagaren vidare till den enhet som utför den algoritm som är avsedd att kalibrera ljussensorn.

Den andra lösningen var att, istället för att mäta upp ljusstyrkan i rummet, koppla ihop två stycken optiska fibrer från samma panel i det rum de är avsedda att upplysa, vilket då skickar ljusintaget tillbaka upp till panelen. Genom att täcka över de linser som förser den ena fiberkabeln med ljus kommer den andra fiberkabeln att skicka ut sitt ljusintag mot de nu täckta linserna. I detta förslag kan en luxmätare placeras i övertäckningsanordningen och där, via omvägen till det upplysta rummet och tillbaka, mäta upp hur mycket ljus panelen tar emot. Då luxmätaren nu befinner sig på panelen kan den direkt skicka sin data till den enhet som förväntas utföra algoritmen.

5.2.2 Tillförlitlighet

För att komma fram till ett svar på frågan ”[h]ur tillförlitligt är det valda kommunikationssättet?” utreddes flera alternativ för att sända data. Genom att använda de fördragna fiberoptiska kablarna säkerställs att det ljus som skickas från rummet upp till panelen alltid kommer att levereras, förutsatt att ljuskällan är tillräckligt stark. Detta är i skarp kontrast mot en trådlös lösning, där flera lager betong mellan rummet och panelen inte är en osannolik företeelse vilket då skulle resultera att datan aldrig når till panelen under normala förutsättningar.

Huruvida de två föreslagna lösningarna för ljussändning är tillförlitliga är beroende på hur ljuset läses av uppe vid panelen. I förslaget med två mikrokontroller, där data sänds seriellt via RS-232, är tillförlitligheten lägre, då den lösningen är störningskänslig mot bakgrundsljus så en tillförlitlig fästningsanordning för panelen behöver tillverkas. Vid försök i labbmiljö har data kunnat sändas med hjälp en vanlig lysdiod driven av 5 volt, 20 mA via en fiberkabel om 16 meter och där tolkats från en enskild fiber. Varje fiberkabel består av sex stycken fibrer och varje fiber är kopplade till en egen lins för att fokuserar fokusera in soljuset. Nu när data skickas ”bakvägen” kommer linsen att agera omvänt genom att omvandla fiberns fokuserade ljus till parallella strålar. Då ljuset nu är parallellt istället för fokuserat finns det risk att ljusintensiteten mellan hög och låg inte är tillräckligt stor för att fotosensorn ska registrera skillnaden, eller

ändra sitt motstånd tillräckligt mycket. För att fotoresistorn ska ha möjlighet att registrera ljusändringarna hade det ideala varit en fokuseringslins till fotoresistorn och att resistorn hade varit innesluten i någon form av behållare att fästa över de sex linserna.

Det andra förslaget, där två fiberkablar kopplas samman för att skapa rundgång, är tillförlitligheten högre eftersom ingen behandlad data skickas. Ingen kommunikation behöver läsas av utan endast rådata i form av ljusstyrka skickas ut genom linserna. När ljusstyrkan då ska mätas är luxmätaren inte beroende av snabba ändringar i ljuset, utan ljuset är konstant vilket leder till en högre tillförlitlighet. Ljusstyrkan kommer vara lägre när den kommer upp till panelen, jämfört med om den skulle stråla ut i rummet, då den behöver färdas dubbelt så långt i fiberkabeln, men mätningen av ljusstyrkan är inte beroende av ett korrekt absolut värde. När panelen kalibreras är det istället av intresse att finna det högsta relativa värdet, det värde när panelen tar in mest ljus, för det är ljuset där som panelen ska vara inställd på, oavsett vilket värde som projektets luxmätaren nu rapporterar. Detta medför också att denna metod är mindre känslig för bakgrundsljus, så länge bakgrundsljuset är konstant eftersom skillnaden i det utstrålade ljuset ändå kan registreras.

För att kontrollera vilket faktiskt luxvärde som panelen levererar till rummet behöver mer kvalificerad materiel användas, materiel som är kalibrerad och granskad för att mäta upp luxvärden i rum. Det är apparatur som företaget har tillgängligt, men som ligger utanför detta projekt.

5.3 Applikation

Den framtagna applikationen gav en förutsättning att uppfylla målet ”att ta fram ett automatiskt system som justerar fokuspunkten på ljussensorn”. En automatisk kalibrering av panelens korrigeringsvärde för ljussensorn kan utföras genom att applikationen har kopplat samman sökalgoritmen med solpanelen och en luxmätare.

Sökalgoritmen implementerades i form av en Pythonapplikation med grafiskt gränssnitt och applikationen stödjer inhämtning av ljusvärden från antingen en till samma dator direkt ansluten luxmätare eller seriell mottagning från en annan enhet. Den luxmätare som användes vid implementationen av direkt anslutning var Yocto-Light-V3 medan Adafruit TSL2591 användes för avläsning som sedan överfördes seriellt via en Arduino Uno. Luxmätarna är beskrivna i avsnitt 3.2.2 respektive 3.2.1. För seriell kommunikation använder applikationen sig av pySerial, ett bibliotek som kan hantera seriell kommunikation på de flesta vanligt förekommande operativsystem [15].

Hos Parans fanns sedan tidigare kringutrustning som använder en Pythonapplikation med grafiskt gränssnitt anpassat till en pekskärm på en Raspberry Pi, där det grafiska gränssnittet är implementerat med ramverket TkInter [16]. Samma ramverk och grafiska formgivning har använts till den applikation som utvecklats av projektet. Detta

upplägg är tänkt att underlätta framtida hantering och utveckling av applikationerna och möjliggör en eventuell framtida integrering av de båda.

Applikationen är utvecklad enligt en objektorienterad utvecklingsmodell och nya avläsningsmetoder kan implementeras utan större ingrepp i befintlig kod. För en översikt av källkodens struktur se UML diagrammen i bilaga C. Mjukvaran till Parans kommande solpanel, SP4, var ej färdigställd under projektets gång och således är applikationen riktad till SP3. Implementeringen av sökalgoritmen kan återanvändas till kommande versioner av solpaneler men vissa anpassningar kan behövas.

6 Slutsats

6.1 Sammanfattning

Detta projekt har kommit fram till att en automatiserad metod för att kalibrera Parans solpanel SP3 är möjlig och levererar en mjukvara som utför just denna uppgift. Krav för att kalibreringsmetoden ska fungera är att rätt hårdvara finns att tillgå, där kalibreringen i sig kräver en luxmätare för intag av belysningsgrad och en kommunikation till SP3an.

För att kommunicera mellan panelen och det upplysta rummet har projektet lämnat två huvudsakliga förslag, dessa är dock inte lika klara till leverans så som algoritmen. Det behöver arbetas mer på den fysiska delen av kommunikationen, där fram för allt en fästordning på panelen för att fästa en ljuskänslig mottagare behöver utvecklas. Även sändarna och mottagarna behöver designas, då dessa är i prototypstadiet med kopplingar på kopplingsbrädor. Teorin bakom kommunikationen är dock utredd och i laborationsmiljö har kommunikation upprättats via företagets eget kablage, så vidare utveckling av denna kommunikationsprincip är möjlig.

6.2 Diskussion

Projektet kan anses ha två huvudsakliga syften, där det första är att ”ta fram en helt automatisk process som kan kalibrera fotosensorn i Parans solpaneler [. . .] med en lägre tidsåtgång och högre precision än dagens manuella metod”. Den metod som bolaget tidigare använde sig av var dels baserad på manuell inmatning av värden, vilket tar tid och kan leda till fel på grund av den mänskliga faktorn, och dels på en manuell uppskattning av ljusstyrkan vilket också kan leda till en felaktig kalibrering. Med hjälp av den algoritm som projektet har utvecklat och redovisat, i samverkan med den tekniska implementationen, anser författarna att detta syfte är uppnått. Processen kan skötas helt automatiskt så till vida att ljusflödet ut från panelen kan uppmätas. Denna automatiserade kalibrering är att anse som tidsbesparande då inga värden behöver anges manuellt, särskilt då skillnaden mellan sensorns ursprungliga värde och det optimala inställningsvärdet och är stort så att många kalibreringssteg behöver göras. Den framtagna algoritmen är generell i det avseende att den inte är begränsad till användning för SP3-panelen utan bör även kunna användas för andra liknande produkter, såsom bolagets efterföljande modell SP4, eller tillämpningar där datastrukturer likt de illustrerade i figur 2 kan påträffas.

Gällande bestämningen av ljusintensiteten finns det både för- och nackdelar med att göra en uppmätning av ljusstyrkan och en mänsklig uppskattning. Fördelarna med en automatiserad inläsning är att kalibreringen blir standardiserad och inte behöver bero på den person som utför kalibreringen. När författarna deltog i en manuell kalibrering av installerade paneler ute i produktion upplevde vi att ljusintensiteten varierar väldigt mycket, från bländande till i princip helt släckt och med tanke på att det mänskliga ögats anpassning till olika ljusintensiteter varierar beroende på om

intensiteten ökar eller minskar kan kalibreringen tappa i precision vid en manuell bedömning [18, s. 273]. Det är svårt att jämföra hur två inställningar förhåller sig till varandra, vilken som är starkare eller svagare, om ljuskällan blivit väldigt mörk mellan de båda tillfällena.

Ett problem som kan uppstå är ifall projektets luxmätare är mer känslig för andra frekvenser än de frekvenser som det mänskliga ögat är känsligt för och om panelen vid felkalibrering tar in ett högre antal av de för luxmätarens känsliga frekvenser på grund av brytning vid linsen, så skulle mätaren registrera ett högre belysningsvärde än vad en människa skulle anse. Detta är en risk som projektet har accepterat då med stöd av litteratur som visar på att ljusmiljö är komplext att bedöma och kräver personer med erfarenhet för att bedömas korrekt, personer som företaget inte har att tillgå vid paneler redan satta i drift [18, s. 278]. De kalibreringar som har genomförts i testmiljön har inte kunnat visa på att fel frekvenser skulle leda till en lägre belysningsgrad, dock finner vi att den uppmätta fokuspunkten, figur 5, av panelen som märklig och luxmätarens frekvensspektra kan vara en orsak. Oavsett den mån teknikern skulle kunna bedöma belysningsgraden så finns vanligtvis inte möjlighet för denna att befinna sig i rummet dit ljuset leder då teknikern befinner sig vid panelen för att sköta kalibreringen. Detta leder då antingen till att teknikern behöver gå emellan panelen och det upplysta rummet, något som är väldigt tidsödande, eller att det krävs två personer för att utföra kalibreringen, en som sköter inmatningen till panelen och en som rapporterar ljusstyrkan. Sammantaget är vår bedömning att en uppmätning av ljusstyrkan är en lämplig metod då det sparar tid vid kalibreringen och resultatet blir oberoende av operatörers erfarenhet gällande bedömning av ljusintensitet.

Det bör dock påpekas att den ljusuppmätningen som detta projekt utför endast är till för att hitta det maximala ljuset ut från panelen och ska inte tas för en ljusmätning som kan representera belysningen i rummet. För att mäta upp belysningen av rummet behövs mer avancerade mätanordningar som kan ge korrekta värden.

Projektet syftar vidare till att ”möjliggöra kommunikation mellan panelen och en luxmätare inne i byggnaden” där vi har undersökt två huvudsakliga metoder, trådlöst eller via de fiberoptiska kablarna som redan är dragna. Då företaget önskade att kommunicera med standardiserade trådlösa metoder för persondatorer uteslöts den möjligheten snart, i och med att hinder i form av betongkonstruktioner så som golv, väggar och tak befinner sig emellan panelen och luxmätaren, vilket hindrar spridningen av radiovågor och överföringen av data.

För optisk kommunikation lämnas dels ett förslag för att lösa detta delproblem och ett annat förslag som möjliggör att flytta luxmätaren från rummet upp till panelen, vilket då omintetgör behovet av att skicka datan från luxmätaren, eftersom den finns tillgänglig för den beräkningsenhet som utför algoritmen. Detta syfte är till del uppnått, då information kring hur dessa två förslag kan realiserats har redovisats, dock saknas det viss materiel för att kunna testa förslagen. Fotoresistorn som nyttjas

vid mottagande av data behöver en infästning till panelen som är ljusskyddad samt eventuellt en lins för att fokuserar det utsända ljuset till resistorn. Denna infästning behöver tillverkas, men med avseende på projektets avgränsningar så anser vi att denna tillverkning ligger utanför projektets ramar. Är det möjligt att skicka data optiskt i det kablage som företaget använder i produktion? Ja det har bevisats möjligt, men har inte kunnat testas på deras produkt utan endast via kablaget.

6.3 Vidareutveckling

Det område som behöver vidareutvecklas från detta projekt är framförallt kommunikationsmöjligheten över optisk fiber. Författarna rekommenderar en undersökning i hur en fästnanordning till solpanelen kan utformas på det lämpligaste sättet för att minska bakgrundsljus, samt hur mottagaren och sändaren kan utformas för att lämna prototypstadiet. Dessa förslag är att anses som utanför data- och informationstekniks institutionen utan är mer lämplig för ett designinriktat examensarbete.

Vidare är en undersökning av en modulerad sändning av ljus i det synliga spektrumet av intresse. Detta projekt har kännedom om grunderna men har inte undersökt det närmare på grund av avsaknad av materiel och tidsutrymme.

Referenser

- [1] A. Dresch, D. P. Lacerda och J. A. V. Antunes. *Design science research: a method for science and technology advancement*. Springer Verlag, 2015. ISBN: 3319073737; 9783319073736.
- [2] C. Sims och H. L. Johnson. *The elements of Scrum*. English. Foster City, Calif.: Dymaxicon, 2011. ISBN: 0982866917; 9780982866917.
- [3] *Parans Installation manual SP3*. 2013/06. Parans Solar Lighting AB. 2013. URL: <http://parans.com/eng/customerservice/documents/ParansInstallationManual.pdf> (hämtad 2015-03-27).
- [4] D. Ibrahim. "Chapter 2 - PIC32 Microcontroller Series". I: *Designing Embedded Systems with 32-bit PIC Microcontrollers and MikroC*. Utg. av D. Ibrahim. Oxford: Newnes, 2014, s. 41–90. ISBN: 978-0-08-097786-7. DOI: 10.1016/B978-0-08-097786-7.00002-6. URL: <http://www.sciencedirect.com/science/article/pii/B9780080977867000026>.
- [5] *Product Specifications*. Parans Solar Lighting AB. 2015. URL: http://www.parans.com/eng/customerservice/documents/ParansProductSpecifications_web.pdf (hämtad 2015-03-30).
- [6] *Two-dimensional PSD*. Hamamatsu Photonics K.K. 2013. URL: http://www.hamamatsu.com/resources/pdf/ssd/s5990-01_etc_kpsd1010e.pdf (hämtad 2015-03-31).
- [7] *TSL2591 Datasheet Apr. 2013 ams163.5*. ams AG. 2015. URL: http://www.adafruit.com/datasheets/TSL25911_Datasheet_EN_v1.pdf (hämtad 2015-05-12).
- [8] *Yocto-Light-V3, User's guide*. Yoctopuce. 2015. URL: <http://www.yoctopuce.com/projects/yoctolightV3/LIGHTMK3.usermanual-EN.pdf> (hämtad 2015-05-12).
- [9] D. Wheat. *Arduino Internals*. Apress, 2011. ISBN: 9781430238829; 1430238828.
- [10] J. Purdum. *Beginning C for Arduino: learn C programming for the Arduino and compatible microcontrollers*. Apress, 2012. ISBN: 9781430247777; 9781430247760.
- [11] *Arduino Build Process*. Arduino. 2015. URL: <http://www.arduino.cc/en/Hacking/BuildProcess> (hämtad 2015-05-11).

- [12] S. Mizushina och A. Adachi. "A technique for wireless LAN connection through building concrete wall at 2.4GHz". I: *Microwave Symposium Digest (MTT), 2010 IEEE MTT-S International*. Maj 2010, s. 260–263. DOI: 10.1109/MWSYM.2010.5514751.
- [13] *Arduino Uno*. Arduino. 2015. URL: <http://www.arduino.cc/en/Main/ArduinoBoardUno> (hämtad 2015-05-05).
- [14] R. Power och A. Rubinsteyn. "How fast can we make interpreted Python?" I: *CoRR* abs/1306.6047 (2013). URL: <http://arxiv.org/abs/1306.6047v2> (hämtad 2015-05-12).
- [15] C. Liechti. *pySerial*. 2015. URL: <http://pyserial.sourceforge.net/pyserial.html> (hämtad 2015-05-13).
- [16] J. Strand, P. Svedberg och O. Åkergren. *Styrdosa till Parans solpanel*. 2015. URL: <https://github.com/MrAkergren/SolarRemote/blob/master/doc/thesis.pdf> (hämtad 2015-05-13).
- [17] *How to install Tkinter*. Tkinter Wiki. 2015. URL: http://tkinter.unpythonic.net/wiki/How_to_install_Tkinter (hämtad 2015-05-19).
- [18] M. Bohgard. *Arbete och teknik på människans villkor*. Swedish. Upplaga 2. Stockholm: Prevent, 2011. ISBN: 9173651109; 9789173651103.

Bilagor

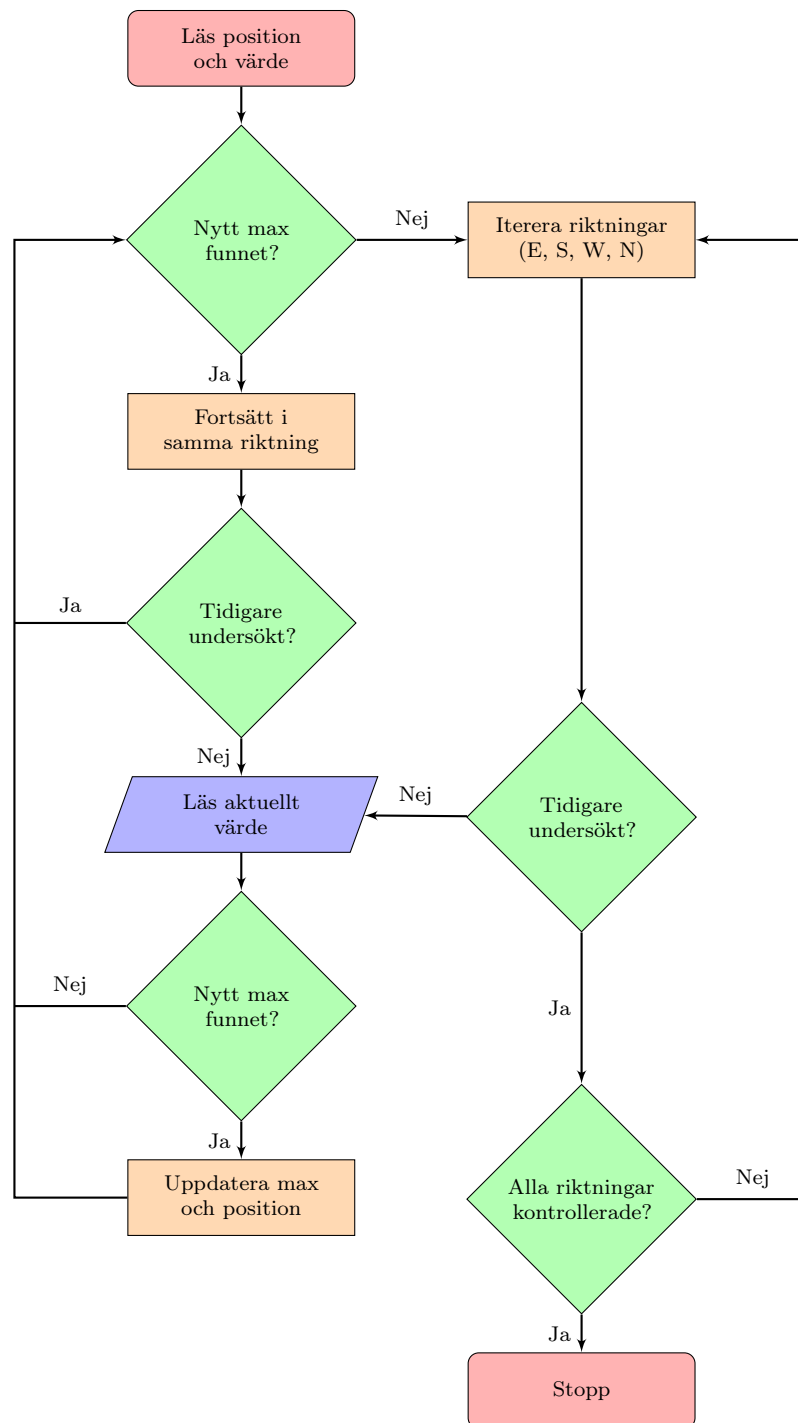
A Simuleringsresultat av kalibreringsalgoritm

Steg	\mathcal{A} : 8 riktningar			\mathcal{B} : 4 riktningar		
	v.1	v.2	v3	v.1	v.2	v3
Medel	255,7	134,2	78,8	160,2	130,5	72,0
Median	215	124	76	140	120	69
Max	1065	450	195	621	448	186
Min	1	7	9	1	4	5
σ	176,3	73,5	33,1	101,1	73,4	32,9

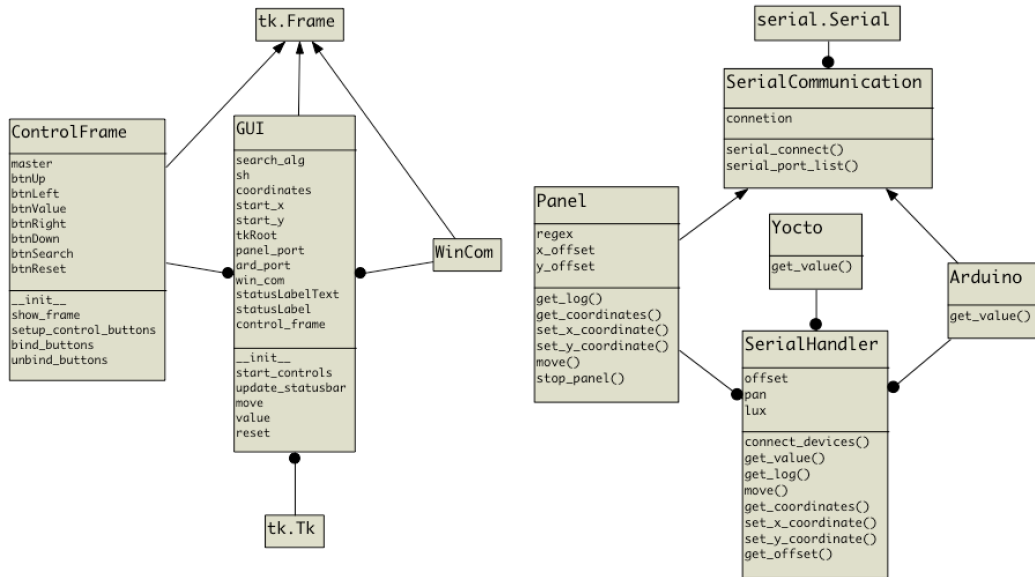
Varje version implementerar ny funktionalitet:

1. Söker medurs i åtta alternativt fyra riktningar.
2. Testa ej redan kontrollerade koordinater.
3. Testa först den riktning som senast lyckad förflyttning.

B Flödesschema av kalibreringsalgoritmen

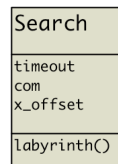


C UML Diagram



(a) Översikt av det grafiska gränssnittet

(b) Översikt av den seriella kommunikationen



(c) Klassen som håller kalibreringsalgoritmen. Ärver inte av andra

```

# Main.py
from search import Search
from serial_handler import SerialHandler
from gui.gui import GUI
import sys

def is_windows():
    return sys.platform.startswith('win')

sh = SerialHandler(0.01, 0.01)
search = Search(sh)
GUI(sh, search, windows=is_windows())
  
```

(d) Skriptet som initierar applikationen.

Fullständig källkod finns att tillgå på: github.com/MrAkergrén/calibration

D Specifikationer

D.1 Arduino

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz
Length	68.6 mm
Width	53.4 mm
Weight	25 g

D.2 Sändare

Mikrokontroller	Arduino Uno
Lysdiod	Vit, 21 000 mcd, 3,4 V, 15°
Resistor	100 Ω

D.3 Mottagare

Mikrokontroller	Arduino Uno
Fotoresistor	18 - 50 k Ω
Resistor	10 k Ω

E Uppmätt fokuspunkt från SP3

cccc	6	7	9	9	10	12	13	14	15	16	16	15	15	14	13	11	9	8	7
7	8	10	11	13	14	17	18	18	19	20	21	21	21	20	18	16	13	12	11
9	11	13	14	17	20	21	24	25	28	30	32	31	30	28	24	23	20	16	15
11	13	16	20	22	26	31	35	38	42	49	50	45	44	40	36	39	30	24	19
14	17	22	26	32	39	44	52	58	67	76	77	73	74	71	60	60	47	38	27
17	24	31	38	47	59	70	82	92	97	95	94	95	97	98	96	85	71	55	42
20	34	45	60	71	83	99	98	96	94	93	92	93	95	97	98	90	81	66	54
25	39	50	62	74	90	99	97	94	92	90	90	90	92	95	98	94	83	70	57
31	40	52	64	77	94	99	96	92	89	87	88	87	89	92	96	99	89	74	60
40	42	55	69	82	100	98	94	89	85	81	82	81	85	87	92	98	95	78	63
42	46	61	78	91	99	95	90	85	78	73	71	71	73	77	87	94	99	89	70
44	52	69	85	99	97	91	82	75	71	65	61	61	63	68	78	88	97	99	77
45	57	76	93	98	93	84	75	69	64	59	57	55	58	63	70	82	94	99	87
48	62	81	99	97	90	81	71	65	60	55	54	52	54	58	66	77	89	98	93
49	64	82	99	96	89	78	69	63	58	54	52	51	53	56	63	75	88	98	96
51	63	82	99	96	89	79	71	63	58	55	52	52	54	57	64	75	88	98	94
51	61	80	100	96	91	81	72	65	60	56	54	53	55	58	65	76	88	98	95
48	59	76	97	97	92	83	73	66	61	56	54	53	54	57	63	73	86	97	99
46	49	66	84	99	94	91	95	89	85	84	84	82	82	85	92	89	87	96	93
43	9	10	12	14	15	17	20	21	22	22	24	25	23	22	20	18	16	13	11

Tabell över normerade värden uppmätta från Parans solpanel SP3. Värdena utgör procent från det maximalt uppmätta värdet, avrundat till närmsta heltal. Anledningen till normeringen är att visa på storleksändring mellan talen, då de faktiska värdena är intetsägande då de inte direkt kan representera lux, på grund av felmarginal av luxmätaren och uppmätningssmetoden.

