

MEMS-peilin hyödyntäminen  
kaksoisfotoniviritteisessä fluoresenssiteknologiassa

Pro Gradu-tutkielma  
Turun yliopisto  
Fysiikan ja tähtitieteen laitos  
Fysiikka  
2020  
LuK Matilda Sipilä  
Tarkastajat:  
Prof. Jarno Salonen  
FM Jori Soukka

Turun yliopiston laatujärjestelmän mukaisesti tämän julkaisun alkuperäisyys on tarkastettu Turnitin OriginalityCheck-järjestelmällä

TURUN YLIOPISTO  
Fysiikan ja tähtitieteen laitos

Sipilä Matilda MEMS-peilin hyödyntäminen kaksoisfotoniviritteisessä fluoresenssi-  
teknologiassa

Pro Gradu-tutkielma, 55 sivua, 20 sivua liitteitä  
Fysiikka  
Syyskuu 2020

---

Kaksoisfotoniviritys on ilmiö, jossa kaksi lähes samanaikaisesti molekyyliin saapuvaa fotonia virittää elektronin ylemmälle energiatilalle. Elektronin palatessa perustilaansa se emittoi fotonin, jonka aallonpituus eroaa virittävien fotonien aallonpituudesta. Kaksoisfotoniviritystä voidaan hyödyntää esimerkiksi diagnostiikassa, jossa sen avulla kyetään määrittämään näyteliuoksessa olevien antigeenien määrä ja näin selvittämään onko näytteessä jotakin tiettyä taudinaiheuttajaa.

ArcDia International Oy Ltd:n kaksoisfotoniviritykseen perustuvassa diagnostiikkalaitteessa käytetään viritykseen lasersädetä. Tällä hetkellä tuotannossa olevissa laitteissa lasersäteen poikkeuttamiseen näyteliuoksessa käytetään kahta pietsosähköiseen ilmiöön perustuvaa skanneripeiliä. Tässä opinnäytetyössä selvitettiin, olisiko mahdollista käyttää näiden peilien sijasta yhtä MEMS-peiliä (engl. micro-electro-mechanical-mirror).

MEMS-peiliä varten tehtiin optiselle pöydälle koejärjestely, jossa kahden linssin avulla suurennettiin MEMS-peililtä objektiiville saapuva lasersäde. Sopivien polttovälien linssit päätettiin kokeellisesti. MEMS-peiliä kontrolloimiseen käytetty ohjelmisto oli alussa Arduino IDE ja myöhemmässä vaiheessa Vivado Design Suite. MEMS-peiliä ja koejärjestelyä testattiin mittaamalla mikropartikkeleja sisältävästä liuoksesta sekä partikkeleista siroavaa laservaloa että fluoroforeista emittoitunutta valoa.

Peilin tarkkuuden ja liikkeen tutkimiseksi mitattiin sirontaa myös skannaamalla mikrokokoluokan hilaa ja mitta-asteikkoa. Samalla kun peiliä liikuteltiin, tallennettiin eri kohdista saatu sirontasignaali datavektoriin. Sirontasignaalit, joiden spatiaalinen järjestys datavektorissa tiedettiin, koottiin mustavalkokuviksi ja näitä kuvia tarkastelemalla päätettiin peilin liikkeen olevan tarpeeksi tarkka tarkasteltavana olevaan laitteistoon.

Lopputuloksena päätettiin MEMS-peilin olevan teoriassa sopiva käytettäväksi pietsoskannereiden tilalla, mutta tämän opinnäytetyön puitteissa koejärjestelyä ei saatu toimimaan niin hyvin kuin nykyisin tuotannossa olevia laitteita. Ongelmia aiheutti erityisesti koejärjestelyn herkkyys esimerkiksi rakennuksen liikkeille. Erityisesti sirontasignaalien laatu ei ollut niin hyvä kuin olisi toivottu, mutta mahdollinen jatkotutkimus tai vakaampi koejärjestely voisivat ratkaista ongelman.

# Sisältö

<b>Johdanto</b>	<b>2</b>
<b>1 Teoreettinen osuus</b>	<b>2</b>
1.1 Kaksoisfotoniviritteinen fluoressenssi . . . . .	2
1.1.1 Matemaattinen malli . . . . .	4
1.2 Konfokaalimikroskopia . . . . .	5
1.2.1 Konfokaalimikroskoopin optiikka . . . . .	6
1.3 TPX-teknologia . . . . .	9
1.4 MEMS-peili . . . . .	12
1.4.1 Mirrorclen MEMS-peili . . . . .	13
1.4.2 DA-muunnin . . . . .	16
<b>2 Kokeellinen osuus</b>	<b>18</b>
2.1 Motivaatio ja lähtötilanne . . . . .	18
2.2 Optiikka . . . . .	19
2.2.1 Koejärjestely . . . . .	22
2.2.2 Linssien kokeellinen tarkastelu . . . . .	24
2.3 Elektroniikka ja sulautetut järjestelmät . . . . .	27
2.3.1 MEMS-peilin ajaminen A-Star 32U4 Prime SV:n avulla . . . . .	27
2.3.2 MEMS-peilin ajaminen ZedBoardin avulla . . . . .	31
2.4 Sironta- ja fluoressenssisignaalin mittaaminen ja analysointi . . . . .	34
2.4.1 Sirontasignaalin mittaaminen mikropartikkeliliuoksesta . . . . .	34
2.4.2 Mittakaavan ja hilan sironnan mittaaminen . . . . .	40
2.4.3 Sironnan ja fluoressenssin mittaaminen liuoksesta . . . . .	45
<b>3 Yhteenveto</b>	<b>53</b>
<b>Liitteet</b>	<b>56</b>

<b>A</b>	<b>Koodi MEMS-peilin ohjaamiseen Arduino IDE:llä</b>	<b>56</b>
<b>B</b>	<b>Kytkimen lisääminen MEMS-peilin ohjaamiseen</b>	<b>58</b>
<b>C</b>	<b>MEMS-peilin ohjaaminen Xilinx SDK:n avulla</b>	<b>64</b>

## Johdanto

Akuutti hengitystieinfektio on ihmisten yleisin sairaus kaikkialla maailmassa riippumatta iästä tai sukupuolesta [1]. On arvioitu, että akuutit hengitystieinfektiot, joihin luetaan kaikki tavallisesta nuhakuumeesta influenssaan, aiheuttavat vuosittain noin 4 miljoonaa kuolintapausta koko maailmassa. Ne ovat myös alle viisivuotiaiden lasten yleisin kuolinsyy [2], ja voivat aiheuttaa suuria epidemioita. Epidemioista viimeaikaisin on Covid-19 koronaviruksen aiheuttama pandemia. Hengitystieinfektioiden lisäksi suolistoinfektiot ovat yleisimpiä infektioita. Ne ovat yksi suurimmista kuolinsyistä etenkin kehittyvissä maissa, ja alle viisivuotiaiden lasten kuolemista koko maailmassa ne selittävät 15 prosenttia [3].

Hengitystie- ja suolistoinfektioihin otetuista antibiooteista suuri osa otetaan turhaan, sillä usein infektion aiheuttaja on virus tai antibiooteille resistentti bakteeri. Lääkärillä on paine määrätä antibiootteja heti potilaalle, vaikka infektion aiheuttaja ei olisikaan tiedossa, sillä usein keskuslaboratorioon lähetettyjen näytteiden käsittelemisessä ja tutkimisessa menee kauan, ja taudinaiheuttaja saadaan selville vasta pitkän ajan kuluttua [4]. Tämä sekä altistaa potilaat turhaan antibioottien sivuvaikutuksille että edistää antibioottiresistenttien bakteerien lisääntymistä.

ArcDia International Oy Ltd:n päätuote on kaksoisfotoniviritteiseen fluoresenssiin perustuva nopea potilaslähtöinen diagnostiikkalaitteisto, jonka avulla voidaan selvittää taudinaiheuttaja yhden vastaanottokäynnin aikana. Testin avulla lääkäri voi perustaa päätöksensä potilaan hoidosta spesifiseen diagnoosiin. Samalla testi-paneelilla, jossa eri taudinaiheuttajat mitataan eri kuopista, voidaan nykyisellään määrittää yli kymmenen eri virusta tai bakteeria hengitysteistä otetusta näytteestä, ja suolistoinfektioiden tapauksessa viisi eri virusta ja bakteeria samasta ulostenäytteestä [5]. Antibioottien turhan käytön lisäksi oikean taudinaiheuttajan tietäminen vähentää potilaan pelkoa sairaudesta, auttaa oireita lievittävän ja kroonistumista estävän hoidon määrämistä ja helpottaa myös esimerkiksi epidemiologisia ennustuksia.

# 1 Teoreettinen osuus

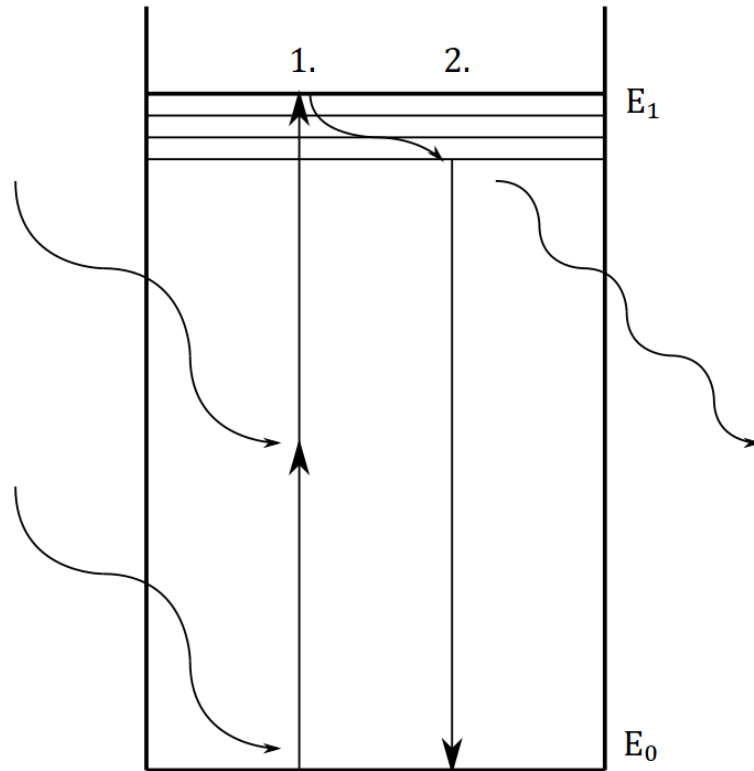
## 1.1 Kaksoisfotoniviriteinen fluoressenssi

Kun tarpeeksi suurienerginen fotoni törmää atomiin, pystyy se virittämään elektronin ylemmälle energiatasolle. Tämä viritys voi tapahtua, kun fotonin luovuttama energia on sama tai suurempi kuin energia-aukko elektronin perustilan ja virittyneen tilan välillä. Kaksoisfotonivirityksessä tähän viritykseen osallistuu kaksi lähes samanaikaisesti atomiin törmäävää fotonia, joiden yhteenlaskettu energia riittää virittämään elektronin ylemmälle tasolle. Tällöin yksittäisen fotonin energia voi olla alhaisempi kuin energia-aukon koko. Kvanttimekaanisesti ensimmäinen fotoni virittää elektronin tämän perustilaa korkeammalla olevalle virtuaalitalalle, ja toinen fotoni nostaa elektronin tältä väliaikaiselta virtuaalitalalta lopulliselle uudelle energiatasolle [6]. Virtuaalitala on molekyylin energiatiilojen superpositiotila, eikä ominaistila, kuten perustila ja lopullinen virittynyt tila [7].

Kun elektroni palaa virittyneeltä tilalta takaisin perustilalleen, se emittoi fotonin, jonka aallonpituus riippuu virittyneen tilan ja perustilan energiaerosta. Jos tämä viritystilän purkautuminen tapahtuu lähestulkoon heti elektronin virittymisen jälkeen, kutsutaan tapahtumaa fluoressenssiksi, ja jos taas elektronin virittymisen ja viritystilän purkautumisen välillä on hetki aikaa, on kyseessä fosforessenssi. Fluoressenssissa spinin multiplisiteetti ei muutu, toisin kuin fosforessenssissa, jossa spinin multiplisiteetti muuttuu usein tripletistä singlettitalaksi tai toisinpäin. Spinin multiplisiteetin muuttuminen aiheuttaa emittoinnin viiveen [8].

Kuvassa 1 on kaksoisfotoniviriteistä fluoressenssia kuvaava skemaattinen Jablonskin energiadiagrammi. Kohdassa 1. ensimmäinen fotoni virittää elektronin perustilalta  $E_0$  virtuaalitalalle  $V$  ja toinen fotoni lopulliselle viritystilalle  $E_1$ . Kohdassa

2. taas kuvataan viritystilan purkautuminen, jolloin elektroni palaa takaisin perustilalleen ja emittoi fotonin, jonka aallonpituus on virittäneisiin fotoneihin verrattuna pienempi.



Kuva 1: Kaksoisfotonivirittaisen fluoresenssin skemaattinen energiadiagrammi.

Kaksoisfotonivirityksen ratkaiseva ero yksittäisen fotonin aiheuttamaan viritykseen on se, että absorboidun yksittäisen fotonin energia voi olla paljon alhaisempi kuin vain yhdellä fotonilla tapahtuvassa virittymisessä. Kaksi erillistä absorboitavaa fotonia kuitenkin aiheuttavat sen, että lopulta emittoidun fotonin energia on samaa suuruusluokkaa kuin yhdellä energettisemmällä fotonilla tapahtuneessa virityksessä. Tällöin virittävien fotonien erottelminen emittoiduista elektroneista optisilla suotimilla on helppoa ja emittoituneiden fotonien määrä pystytään mittaamaan tarkasti [6]. Tätä ilmiötä, missä emittoituneen fotonin aallonpituus eroaa virittäneen fotonin aallonpituudesta, kutsutaan Stokesin tai anti-Stokesin siirtymäksi [9].

Toinen etu, mikä kaksoisfotonivirityksellä on yhdellä fotonilla tapahtuvaan vi-

rittymiseen nähden on se, että virittämiseen tarvittava aallonpituus voi olla niin pitkä ettei se aiheuta ylimääräistä fluoresenssia esimerkiksi näytettä ympäröivästä liuksesta. Pidempi aallonpituus ei myös aiheuta vahinkoa biologiselle näytteelle, jos kaksoisfotoniviritystä käytetään biologisen aineen kuvantamiseen. Kaksoisfotonivirityksessä käytetään yleensä punaisen valon tai lähi-infrapuna-alueen eli NIR-säteilyn (engl. near infrared) aallonpituuksia virittämään sellaisia fluoroforeja, jotka yhdellä fotonilla viritettäessä viritetään lyhyen aallonpituuden ultraviolettisäteilyllä (engl. deep ultraviolet), ultraviolettisäteilyllä tai lyhyiden aallonpituuksien näkyvä-  
lä valolla. Lyhyillä ultraviolettiaallonpituuksilla, jotka käsittävät alle 300 nanometrin aallonpituusalueen säteilyn, virittäminen aiheuttaa valovaurioita eläviä soluja kuvannettaessa [10][11].

### 1.1.1 Matemaattinen malli

Teoreettinen malli kaksoisfotoniviritykselle esitettiin jo vuonna 1931, kun Maria Göppert-Mayer käsitteli aihetta väitöskirjassaan. Viritys tapahtuu virittyvän atomin tai molekyylin ja virittävän sähkömagneettisen kentän vuorovaikutuksen johdosta, ja tämä prosessi voidaan kuvailla käyttäen aikariippuvaista Schrödingerin yhtälöä. Tämän yhtälön Hamiltonianissa, eli energiaoperaattorissa, on elektronin dipolin vuorovaikutustermi  $\vec{E}_\gamma \cdot \vec{r}$ , jossa  $\vec{E}_\gamma$  on fotonien sähkökenttävektori ja  $\vec{r}$  paikkaoperaattori. Yhtälö voidaan ratkaista perturbaatioteorian avulla, jolloin kaksoisfotoniviriteisessä tapauksessa transition todennäköisyys perustilan  $|i\rangle$  ja virittyneen tilan  $|f\rangle$  välillä

$$P \approx \left| \sum_m \frac{\langle f | \vec{E}_\gamma \cdot \vec{r} | m \rangle \langle m | \vec{E}_\gamma \cdot \vec{r} | i \rangle}{\varepsilon_\gamma - \varepsilon_m} \right|^2, \quad (1)$$

missä  $\varepsilon_\gamma$  on fotonin sähkömagneettiseen kenttävektoriin  $\vec{E}_\gamma$  liittyvä energia ja summa lasketaan kaikkien perustilan ja virittyneen tilan välillä olevien tilojen  $m$  yli. Muuttuja  $\varepsilon_m$  on perustilan ja tilan  $m$  välinen energiaero [6].

Absorptiovaikutusala kuvaa absorptioon todennäköisyyttä hiukkassysteemissä. Absorptiovaikutusalan yksikkö on  $\text{cm}^4\text{s}$ , ja  $1 \cdot 10^{-50} \text{ cm}^4\text{s}$  tunnetaan nimellä yksi Göppert-Mayer eli GM [12]. Jos molekyylin perustilan  $i$  ja virtuaalitalan  $j$  välinen absorptiovaikutusala on  $\sigma_{ij}$ , virtuaalitalan ja lopullisen virittyneen tilan  $f$  absorptiovaikutusala  $\sigma_{jf}$  ja  $\tau_j$  virtuaalitalan kestoaika, kaksoisfotonivirityksen absorptiovaikutusala

$$\sigma_2 = \sigma_{ij}\sigma_{jf}\tau_j. \quad (2)$$

Virtuaalitalan kestoaikaa voidaan arvioida Heisenbergin epätarkkuusperiaatteella, sillä  $\tau_j$ :n pitää olla tarpeeksi lyhyt, jotta se ei riko energian säilymislakia. Tällöin

$$\tau_j \approx \frac{1}{\Delta \omega} = \frac{1}{|\omega_{ij} - \omega|}, \quad (3)$$

missä  $\omega_{ij}$  on transitiotaajuus ja  $\omega$  virittävän fotonin taajuus. Jos  $\omega_{ij}$  on elektronin transitiotaajuus ja arvioidaan, että perustilan ja virittyneen tilan energiaero ei ole suuri, virtuaalitalan  $\tau_j$  kestoaika on  $10^{-15}$ - $10^{-16}$  s. Absorptiovaikutusalan laajuus taas kaksoisfotonivirityksessä on noin  $10^{-49} \text{ cm}^4 \text{ s/fotoni}$  [7].

## 1.2 Konfokaalimikroskopia

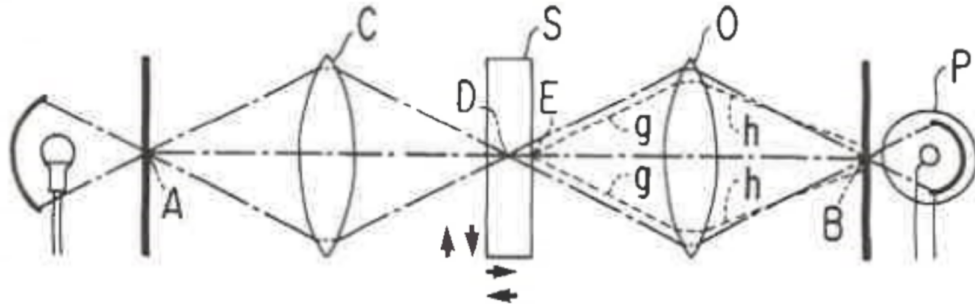
Konfokaalimikroskopia on erityisesti biologiassa käytetty kuvantamismenetelmä, jossa pystytään kuvaamaan ohuita leikkeitä näytteestä ja estämään muualta kuin fokus-tasolta tulevan säteilyn havaitseminen. Konfokaalimikroskoopissa on tavallisesta valomikroskoopista poiketen mikroskoopin kokoojalinssin tilalla objektiivilinssi, ja sen valaistuskenttää (engl. field of illumination) säädellään erillisen neulanreiän (engl. pinhole) avulla. Alkuperäisen konfokaalimikroskoopin idean keksi Marvin Minsky vuonna 1957 Harvardin yliopistossa, mutta tämän jälkeen konfokaalimikroskooppeja on kehitetty lukuisia erilaisia niin, että niille kaikille yhteistä on lähinnä valonsäteen rajoittaminen neulanreiällä tai ohuella raolla. Laserviritteisessä konfokaalimikroskoopissa valonlähteenä käytetään laseria, joka tuottaa koherentin ja yhdensuun-

taisen valonsäteen ja pystyy virittämään kuvantamisen kohteena olevan näytteen molekyylejä saaden aikaan fluoresenssi-ilmiön.

Vaikka konfokaalimikroskoopilla saavutettava resoluutio onkin tavallisen valomikroskoopin luokkaa, sillä on kuitenkin huomattavasti suurempi kyky tuottaa kontrastiltaan parempia kuvia. Konfokaalimikroskoopilla saadaan myös näytteestä otettua useita syvyysuuntaisia leikkeitä, jotka voidaan signaalinkäsittelyohjelmalla muuntaa yhdeksi kolmiulotteiseksi kuvaksi. Tämän vuoksi konfokaalimikroskoopin suurimmat sovelluskohteet ovat biologisten kudosten ja mikrobien kuvantamisessa, joissa usein tärkeää on saada kolmiulotteinen kuva näytteestä. Konfokaalimikroskopiolla pystytään myös ottamaan esimerkiksi kuvia elävistä soluista ilman, että näyte kärsii. Tämän vuoksi se on tärkeässä roolissa biologisten prosessien kuvantamisessa [13].

### 1.2.1 Konfokaalimikroskoopin optiikka

Kuvassa 2 on yksinkertaistettu konfokaalimikroskoopin periaate.



Kuva 2: Yksinkertainen konfokaalimikroskoopin optinen polku [13].

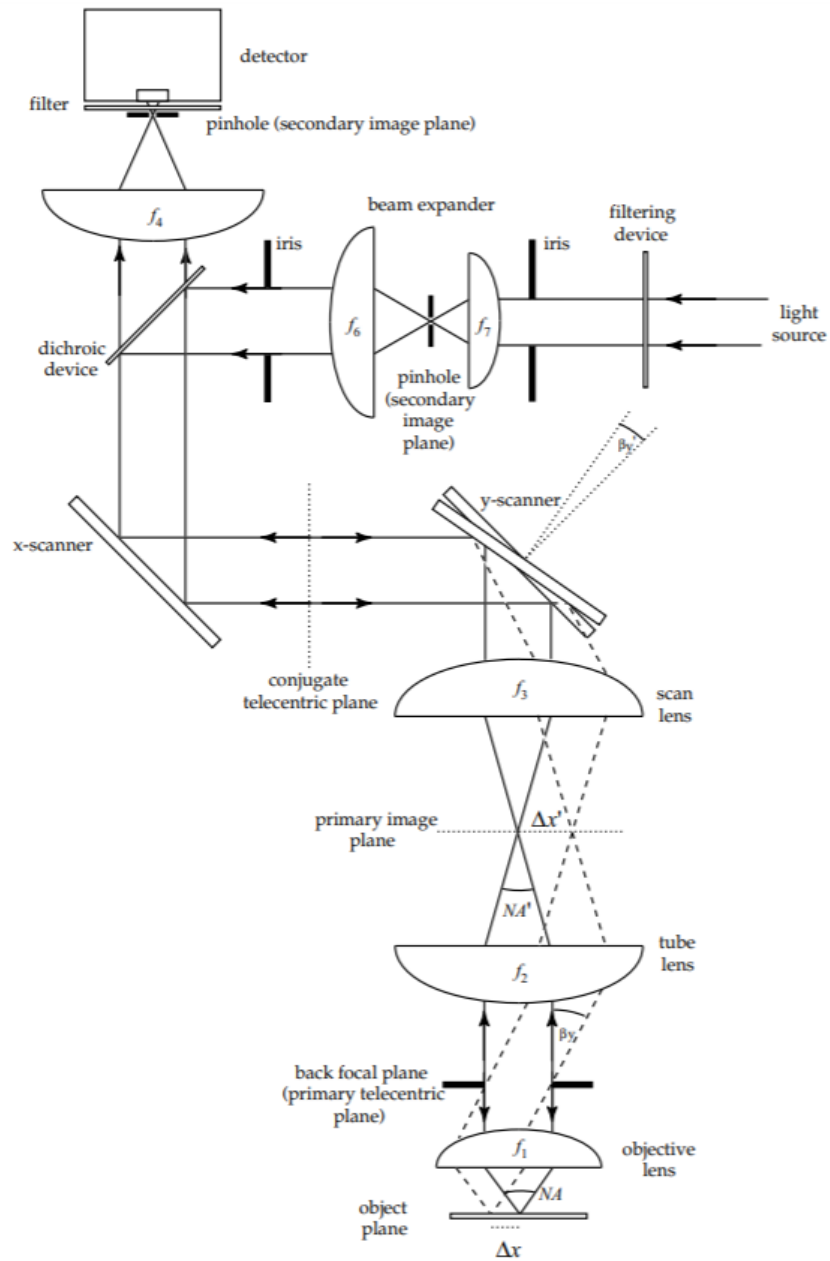
Kuvassa 2 valo tulee vasemmalta puolelta ja kulkee sen jälkeen ensimmäisen neulanreiän A läpi. Kokoojalinssi C muodostaa kuvan neulanreistä näytteen S keskelle konfokaalipisteeseen D. Objektiivilinssi O muodostaa pisteestä D kuvan toiseen neulanreikään B. Tämä neulanreikä on neulanreiän A ja pisteen D kanssa konfokaalinen eli neulanreiät A ja B kuvautuvat samaan pisteeseen D näytteessä. Kuvassa oleva

piste E, joka ei ole ensimmäisen neulanreiän vuoksi fokuksessa saa huomattavasti vähemmän valaistusta kuin piste D. Tällöin vain pisteestä D saatu kuva päättyy neulanreiän B kautta valonmonistinputkelle tai muulle valonilmaisimelle P. Lisäksi suurin osa näytteen pisteestä D ja sen ympäristöstä sironneesta säteilystä ei läpäise toista neulanreikää B, joten valomonistinputkelle päättyy vain tarkalleen kohdasta D saatu kuva. Jos neulanreikä B on lisäksi vielä tarpeeksi pieni, vähentää se kohdasta D saatuja diffraktiorenkaita ja näin mikroskoopin resoluutio paranee. Kun näytettä S skannataan liikuttamalla sitä, saadaan valomonistinputkelle eri näytteen kohdista signaali. Nämä skannatut pisteet  $D_1$ ,  $D_2$ ,  $D_3$  jne. voivat olla optista polkua vasten lateraalissa tasossa tai aksiaalisessa tasossa valonsäteen suuntaisesti. Kun lopuksi näytteen eri pisteistä saadut signaalit yhdistetään tietokoneella ja tiedetään mitä spatiaalista pistettä minkäkin vahvuinen signaali vastaa, saadaan lopulta näytteestä koostettua kuva [13].

Kuvassa 2 mikroskoopin toiminta perustuu tavalliseen näytteen skannaamiseen samalla tavalla kuin valomikroskoopissa, mutta samaa periaatetta voidaan käyttää myös fluoresenssiin perustuvassa mikroskoopissa. Tällaista mikroskooppia käytettäessä kuvattavassa näytteessä tulee olla sellaisia molekyylejä, jotka voivat virittyä tulevasta säteilystä. Kun viritystilat purkautuvat, näytteessä olevat molekyylit emittoivat fotoneja, jotka voidaan taas havaita valonilmaisimella ja koostaa signaaleista samalla tavalla kuva kuin edellä.

Kuvassa 3 on skemaattinen kuva tällaisen fluoresenssiin perustuvan konfokaalimikroskoopin yhdestä mahdollisesta rakenteesta. Tässä kuvassa oikealta tuleva viritysvaio kulkee ensin kahden linssin läpi, jotka suurentavat sädettä ja heijastuu sitten dikroistisesta peilistä. Dikroistinen peili poikkeaa tavallisesta peilistä siten, että se heijastaa tietyn aallonpituusalueen valoa ja päästää muun aallonpituusalueen valon lävitseen. Dikroistisen peilin jälkeen säde johdetaan kahteen peiliin, joista toista voidaan liikuttaa X- ja toista Y-suunnassa. Näiden skanneripeilien jälkeen säde kulkee

vielä kahden linssin läpi joiden tarkoitus on saada valonsäde tulemaan viimeiselle objektiivilinsille samaan paikkaan, mutta eri kulmissa, jolloin objektiivilinssi kokoaa säteen näytteeseen. Polttopisteen paikka näytteen tasossa (kuvassa engl. object plane) riippuu tästä kulmasta [13].



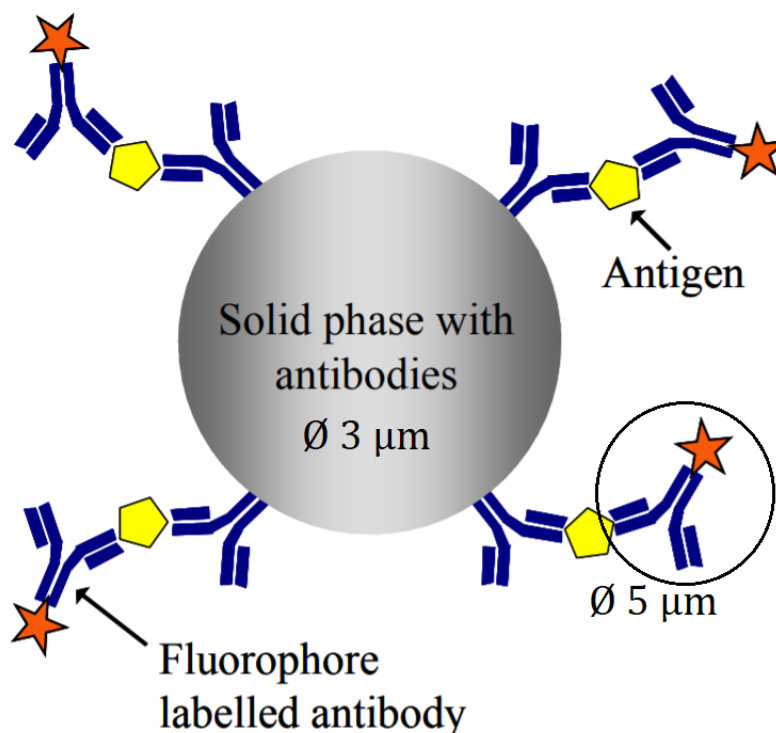
Kuva 3: Yksinkertaistettu fluoresenssiin perustuvan konfokaalimikroskoopin optinen polku [13].

### 1.3 TPX-teknologia

TPX-teknologia on Turun yliopiston Biofysiikan laboratoriossa kehitetty kaksoisfotoniviritteseen fluoresenssiin perustuva tapa määrittää mikropartikkeliin kiinnittyneiden bioaffiniteettikompleksien määrä. Tätä määrittystä käyttämällä voidaan liuoksesta, joka sisältää esimerkiksi hengitystieinfektiosta kärsivän ihmisen limaa, selvittää mitä taudinaiheuttajaa eritteessä on.

Mittaukset tehdään liuoksesta, joka sisältää kiinteänä faasina mikropartikkeleita, eli vasta-aineilla päällystettyjä pallomaisia polystyreenipartikkeleita, joiden halkaisija on kolme mikrometriä [14]. Vasta-aineet, joita kutsutaan myös immunoglobuliineiksi, ovat immuunijärjestelmän tuottamia proteiineja. Nämä proteiinit tunnistavat ja kiinnittyvät immuunijärjestelmän ulkopuolisiin aineisiin, joita kutsutaan antigeneiksi [15]. Kuvassa 4 keskellä oleva ympyrä kuvaa pallomaista polystyreenipartikkelia (engl. solid phase with antibodies). Kun samaan liokseen lisätään näytettä, joka mahdollisesti sisältää mitattavaa kohdemolekyyliä, näytteen antigeenit kiinnittyvät tässä mikropartikkelissa oleviin vasta-aineisiin. Kuvassa 4 mikropartikkeliin kiinnittyneitä antigenejä kuvaavat viisikulmiot. Näihin taas kiinnittyy fluoresoivilla komponenteilla, eli fluoroforeilla, leimattuja vasta-aineita, joita on myös ollut alkuperäisessä mikropartikkeliliuoksessa. Fluoroforeja kuvataan tähdillä kuvassa 4. Lopuksi mikropartikkeleihin kiinnittyneiden fluoresoivien aineiden määrä on siis suoraan verrannollinen partikkeleihin kiinnittyneiden taudinaiheuttajien määrään [16] [17].

TPX-teknologiassa tätä liuosta skannataan fokusoidulla lasersäteellä, joka on aallonpituudeltaan ja teholtaan sopiva virittämään liuoksen fluoroforit. Tällä hetkellä tuotannossa olevissa laitteissa skannaus tehdään kahdella pietsosähköiseen ilmiöön perustuvalla skanneripeilillä, joista toinen poikkeuttaa sädettä X- ja toinen Y-suuntaan. Näillä saadaan lasersäde liikkumaan liuoksessa nopeasti säännötöntä Lissajous-kuviota, joka peittää koko skannausalueen. Kun mikropartikkeli osuu la-

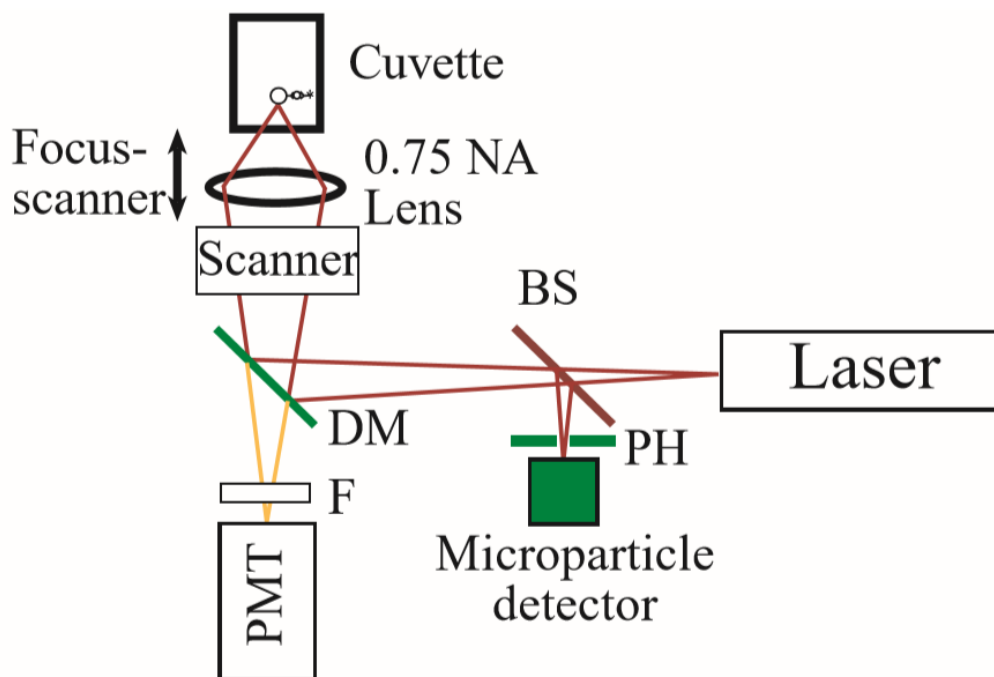


Kuva 4: Skemaattinen kuva analyyttien kiinnittymisestä kiinteään faasiin [17].

säteeseen, havaitaan valonilmaisimella partikkelista sironneet virittävän valon fotonit. Valonilmaisimen edessä on konfokaalisesti kohdistettu neulanreikä, joka estää muun kuin fokuksesta tulevan valon pääsyn ilmaisimelle. Kun partikkeli on havaittu, skannerit pysäytetään ja optisten voimien annetaan vetää mikropartikkeli fokuksen läpi [16]. Tämä optinen kaappaus perustuu siihen, että kun pallomainen partikkeli, jonka taitekerroin on suurempi kuin sen ulkopuolella olevan liuoksen taitekerroin, heijastaa säteilyä, heijastuneen fotonin liikemäärä muuttuu. Liikemäärän säilymislain mukaan partikkelin liikemäärä muuttuu saman verran, mutta vastakkaiseen suuntaan. Tämä voima työntää partikkelia lasersäteiden polttopisteen läpi [13].

Kun mikropartikkelista havaitaan sirontaa, aletaan rekisteröidä partikkelista tulevaa fluoresenssisignaalia. Jos mikropartikkeliin on kiinnittyneenä fluoroforeja, virittää lasersäde nämä ylemmille viritystiloille, jotka purkautuvat heti. Viritystilan purkautuessa fluorofori emittoi fotonin, jonka aallonpituus on kaksoisfotonivirityksen mukaisesti pienempi kuin saapuneen lasersäteiden aallonpituus. Tällä tavalla emittoi-

tuneet energettisemmät fotonit havaitaan muun aallonpituusalueen fotonit suodat-  
tavan valomonistinputken avulla ja havaituista signaaleista voidaan päätellä mikro-  
partikkeleihin sitoutuneiden fluoroforien määrä ja näin liuoksessa olevien antigeen-  
ien määrä. Kuvassa 5 on kaavakuva mittauslaitteistosta [14].



Kuva 5: TPX-teknologiaan perustuva mittauslaitteisto [14].

Kuvassa 5 lasersäde, jonka aallonpituus on 1,064 mikrometriä, kulkee ensin puolilä-  
päisevän peilin BS läpi (engl. beamsplitter) toiselle, dikroistiselle, peilille DM (engl.  
dichroistic mirror), joka heijastaa vain tietyn aallonpituusalueen säteilyä. Lasersäde  
heijastuu tästä ja kulkee skannerille, jona voi toimia yksi tai kaksi poikkeutettavaa  
peiliä. Ennen kyvetissä olevaa liuosta lasersäde kulkee objektiivin läpi, joka ko-  
koaa säteen polttopisteeseensä. Jos säde siroaa mikropartikkelista fokuksessa, kulkee  
sironnut valo takaisin optista polkua pitkin peilille BS, joka heijastaa tältä puolel-  
ta tulevan säteilyn neulanreiän kautta havaitsijalle. Jos taas polttopisteessä olevassa  
mikropartikkelissa tapahtuu virittyminen ja fotonin emittoiminen, kulkee emittoitu-  
nut foton peilin DM ja emissiosuotimen F läpi valomonistinputkelle PMT (PMT =

photomultiplier tube). Emissiosuodin estää laserin aallonpituuksien pääsyn valomonistinputkelle, mutta päästää kaksoisfotonivirityksessä emittoidut fotonit lävitseen. Hajavalon havaitsemista vähentää myös valomonistinputken IR-säteilyä lähellä oleva herkkyysalue.

Menetelmän hyvä puoli on se, että mikropartikkeleiden käyttö kiinteänä faasina on käytännöllisempi ja kustannustehokkaampi kuin tekniikat, joissa kiinteänä faasina on koeputken seiniä päällystävät vasta-aineet [14]. Kaksoisfotonivirityksen etu yhdellä fotonilla tapahtuvaan viritykseen on se, että kaksoisfotonivirityksellä päästään syvemmälle fluoroforiin ja emittoituneiden fotonien aallonpituusjakauman ero virittävien fotoneiden aallonpituuksiin mahdollistaa tarkan fotonilaskennan [18]. TPX-mittauksessa olevissa mikropartikkeleissa ja koettimena voi olla myös erilaisia vasta-aineita, jolloin näytteestä voidaan etsiä useampaa kohdetta. Laserin aallonpituus on myös sellainen, että se ei absorboidu hyvin biologiseen aineeseen ja ei näin tuhoa sitä, joten teknologian käyttö esimerkiksi kokonaisten solujen mittaamiseen on mahdollista [14]. Teknologian suorituskyky ei myös riipu tilavuudesta, jolloin jo pieni määrä tutkittavaa ainetta riittää. Liuosta mitattaessa on myös mahdollista erottaa liuostaustan aiheuttamat signaalit ja erottaa nämä mikropartikkeleista tulevista signaaleista, jolloin liuostaustan vaikutus lopputulokseen pystytään kompensoimaan [17].

## 1.4 MEMS-peili

MEMS-peilit (engl. micro-electro-mechanical-system, MEMS) ovat mikromittaluokan peilejä, joiden kulmaa peilin normaaliin nähden pystytään muuttamaan ulkoisen jännitteen avulla. Peilejä on vain Y- tai X- akselin suuntaan kääntyviä, sekä molempiin suuntiin kääntyviä, mutta yhteistä niille kaikille on pieni koko ja nopea liikuteltavuus. MEMS-peileissä heijastava pinta on kiinnitetty tai höyrystetty suoraan substraattiin, jota liikutellaan. Nimensä mukaisesti komponentin koko voidaan

kuvata mikrometreissä, vaikka peilien halkaisija vaihteleeekin käyttötarkoituksen mukaan nykyään mikrometreistä joihinkin millimetreihin. Yhteen suuntaan kääntyvissä peileissä keskellä on sarana, joka kääntää peiliä kun saranan päissä oleviin elektrodeihin johdetaan jännitettä [19]. On myös MEMS-peilejä, joiden toiminta perustuu peilin ympärillä johtimissa kulkevien sähkövirtojen aiheuttaman magneettikenttään. Tämä magneettikenttä aiheuttaa peilin kääntymisen Lorentzin voimien vuoksi [20].

MEMS-peileillä on useita käyttötarkoituksia, joista yleisimpiä ovat esimerkiksi lidar-sovellukset (engl. light detection and ranging), jotka käyttävät optisia tutkia[21]. MEMS-peiliä on käytetty myös esimerkiksi optisella endoskoopilla biologisten kudosten kuvantamisen apuna [18], interferometrisissä järjestelmissä [22] ja aberratioiden korjaamisessa [23]. Ne ovat myös yleistymässä juuri konfokaalisissa mikroskoopeissa korvaten aiempia esimerkiksi galvanometrisiä tai pietsosähköiseen ilmiöön perustuvia skanneripeilejä [24][25].

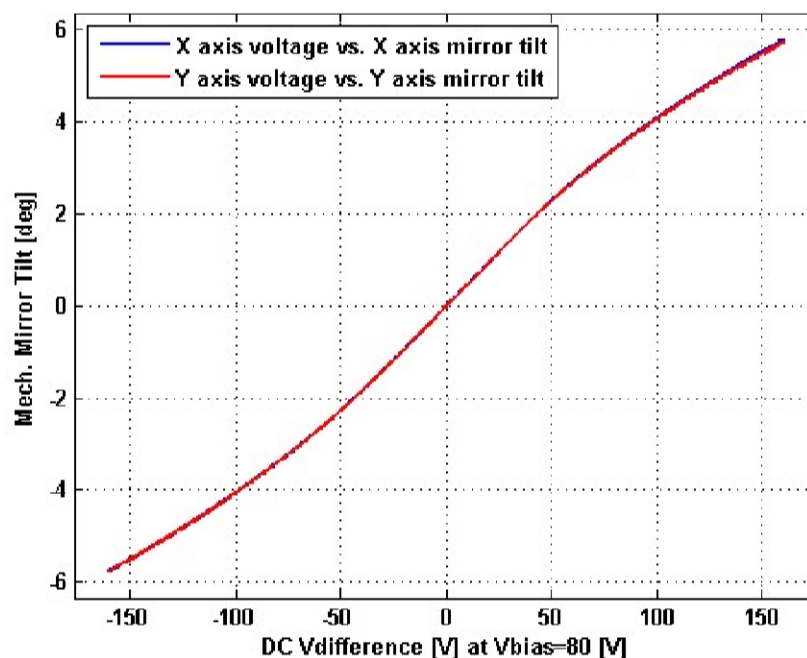
#### 1.4.1 Mirrorclen MEMS-peili

Tässä opinnäytetyössä tarkastellaan sekä X- että Y-suuntaan poikkeutettavaa Mirrorcle Technologies Inc.:in MEMS-peiliskanneria. Työssä käytetty MEMS-peili oli Integrated MEMS Mirror A5M24.2-2400AL, jossa heijastavan pinnan halkaisija oli 2,4 mm [26].

Mirrorclen MEMS-peili saa sisääntuloina neljä eri analogista signaalia, jotka AD-muunnin on muuntanut tietokoneelta SPI-väylän (engl. serial peripheral interface) kautta saaduista digitaalisista signaaleista. SPI-väylä on sulautetun laitteiston yksikö, joka on vastuussa ulkoisten oheislaitteiden ja mikrokontrollereiden kanssa kommunikoinnista. Sen toiminta perustuu siirtorekistereihin, joita käytettäessä tavujen lähettämisen ja vastaanottamisen tiedonsiirtonopeus on korkea. SPI-väylä käyttää synkronoitua viestintää, sillä databitit lähetetään ja vastaanotetaan synkronoivan kellon mukaisesti. Sellaisille mikrokontrollereille, joiden nastaluku ei ole suuri, SPI

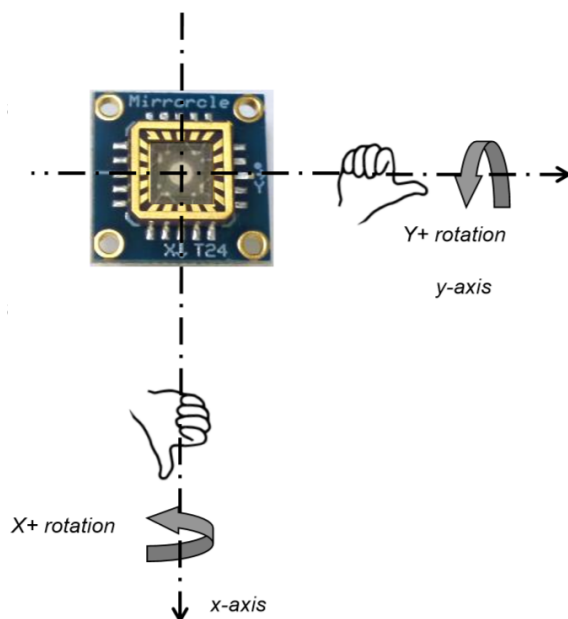
on suhteellisen yksinkertainen ja helppokäyttöinen [27].

Neljä analogista peiliin saapuvaa kanavaa vastaavat peilin negatiivista Y-suuntaa, positiivista Y-suuntaa, negatiivista X-suuntaa ja positiivista X-suuntaa. Kun esimerkiksi negatiiviseen Y-kanavaan ajetaan signaalia, kääntyy peili negatiiviseen Y-suuntaan akselinsa ympäri. Y- ja X-kanavat ovat peilissä täysin identtisiä. MEMS-peili on nollassa asennossa, eli ei kallistuneena mihinkään suuntaan, kun jokaiseen neljään eri analogiseen kanavaan ajetaan 80 voltin jännitettä. Peilin liike riippuu siitä, millaisia tämän 80 voltin molemmiin puolin olevia jännitteitä kanaviin ajetaan. Jännitteet ovat myös kytköksissä toisiinsa, sillä esimerkiksi kun negatiiviseen Y-kanavaan ajetaan 90 voltin jännitettä, täytyy positiiviseen Y-kanavaan ajaa 70 voltin jännitettä, jotta jännitteet ovat yhtä kaukana 80 voltista. Näin peili kääntyy oikein, eikä siihen synny ylimääräisiä jännitteitä. Kuvassa 6 on peilin X- ja Y-suuntien kulmat jännite-eron funktiona. Tämä jännite-ero on ero 80 voltin korkeaajännitteestä [28].



Kuva 6: Peilin kulmat jännite-eron funktiona [29].

Kuten kuvasta 6 huomataan, peilin mekaaniset kulmat kasvavat melko lineaarisesti jännite-eron funktiona ja peilin maksimikulmat ovat noin 5,8 asteen suuruiset. Kuvassa 7 on esitetty peilin kiertosuunnat eri kanaville oikean käden sääntöä käyttäen. Peili kääntyy X-suuntaan oikean käden peukalon osoittamaan suuntaan,



Kuva 7: Peilin kulmien kiertosuunnat [28]

kun positiiviseen X-kanavaan ajetaan nollapositiojännitettä korkeampaa jännitettä. Y-suunnassa peili taas kääntyy oikean käden peukalon osoittamaan suuntaan, kun positiiviseen Y-kanavaan ajetaan nollapositiojännitettä matalampaa jännitettä.

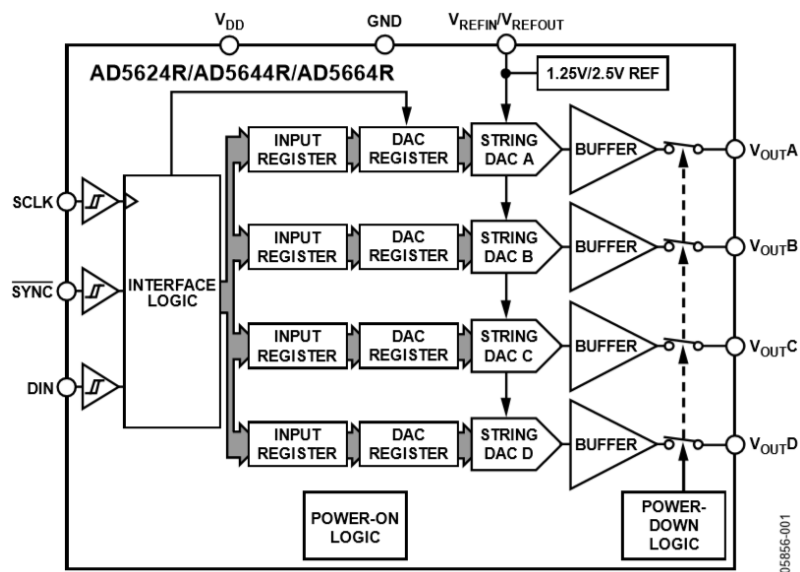
Itse peilin lisäksi sen käyttämistä varten tarvitaan myös erillinen ajuri, joka kytketään lattakaapelilla kiinni MEMS-peiliin. Kyseessä olevan peilin ajuri on tyypiltään digitaalisen sisääntulon ajuri, jossa sisääntuleva signaali ajetaan SPI-väylää pitkin yhden piirilevyn tietokoneesta (engl. single-board computer, SBC).

Ajuri vaatii viiden voltin käyttöjännitteen, synkronointisignaalin, suodatin- ja sarjakellosignaalit, datasignaalin sekä signaalin, josta luetaan voidaanko ajuriin kytkeä korkeajännitteet. MEMS-peili tarvitsee liikkuaakseen yli 80 voltin jännitteen ja tätä varten ajurissa on kytkentä jolla 5 voltin käyttöjännitteestä muodostetaan 200

voltin korkeajännite. Tämä korkeajännite syötetään MEMS-peilin ajuripiiriin nelikanavaiselle korkeajännitevahvistimelle Microchip HV264:lle. Vahvistin vahvistaa DA-muuntimelta tulevat matalat ohjausjännitteet, niin että peilejä saadaan liikutettua.

### 1.4.2 DA-muunnin

Peilin ajurin merkittävin komponentti on DA-muunnin (engl. digital to analog converter), joka muuntaa tietokoneelta tulevan digitaalisen lukuarvon analogiseksi jännitteeksi. Muunnin on 16-bittinen (lukualue 0 - 65535) ja 4-kanavainen, sen tyyppi on AD5664R ja valmistaja Analog Devices. Kuvassa 8 on DA-muuntimen lohkokaavio, jossa näkyy neljä eri ulostulokanavaa A-D, joista tulevat jännitteet vastaavat peilin kanavia X+, X-, Y+ ja Y-.



Kuva 8: DA-muuntimen lohkokaavio [30].

DA-muuntimeen lähetetyt signaalit peilin ajamiseen ovat välin 0-65535 integerdatatyyppin numeroita, jotka vastaavat lopullisia peiliin ajettavia -70-230 voltin jännitteitä. Ajurin datalehdessä on kaava peilin ajamiseksi nollassa, jossa ajurin

kaikkiin kanaviin ajettava signaali

$$\begin{aligned} DV_{80\text{ V}} &= \frac{80\text{ V}}{200\text{ V}} \cdot 65535, \text{ eli} \\ &= 26214, \end{aligned}$$

jotta jokaiseen kanavaan ajetaan 80 voltin jännitettä. Jos peilin taas halutaan kääntyvän jonkin kulman verran, tulee ensin kuvasta 6 lukea haluttua kulmaa vastaava jännite-ero  $\Delta V$  sekä selvittää kuvan 7 avulla mihin suuntaan peiliä käännetään. Lukemat, jotka asetetaan kanaviin X+, X-, Y+ ja Y-, saadaan seuraavista kaavoista, joissa

$$\begin{aligned} X+ &= 26214 + \frac{\Delta V}{200\text{ V}} \cdot 32767, \\ X- &= 26214 - \frac{\Delta V}{200\text{ V}} \cdot 32767, \\ Y+ &= 26214 - \frac{\Delta V}{200\text{ V}} \cdot 32767 \text{ ja} \\ Y- &= 26214 + \frac{\Delta V}{200\text{ V}} \cdot 32767. \end{aligned}$$

Näin esimerkiksi kun peili halutaan ajaa positiivisessa X-suunnassa kahden asteen kulmaan ja Y-suunnassa pitää paikallaan, on X-suunnassa kuvasta 6 luettu jännite-ero 50 voltia ja Y-suunnassa taas 0 voltia. Nyt ajettava signaali

$$\begin{aligned} X+ &= 26214 + \frac{50\text{ V}}{200\text{ V}} \cdot 32767 \approx 34406, \\ X- &= 26214 - \frac{50\text{ V}}{200\text{ V}} \cdot 32767 \approx 18022, \\ Y+ &= 26214 - \frac{0\text{ V}}{200\text{ V}} \cdot 32767 = 26214 \text{ ja} \\ Y- &= 26214 + \frac{0\text{ V}}{200\text{ V}} \cdot 32767 = 26214. \end{aligned}$$

## 2 Kokeellinen osuus

Opinnäytetyön kokeellinen osuus toteutettiin ArcDia International Oy:ssä. Opinnäytetyön tarkoituksena oli tutkia ArcDia International Oy:n diagnosointilaitteiston pietsosähköiseen ilmiöön perustuvien skanneripeilien korvaamista MEMS-peilillä.

### 2.1 Motivaatio ja lähtötilanne

Yhtiön päätuote on kaksoisfotoniviritteiseen fluoresenssimäärittämiseen perustuva monianalyttinen diagnosointilaitte, jolla pystytään määrittämään keskuslaboratoriotason tarkkuudella muun muassa hengitystieinfektioiden ja suolistoinfektioiden taudinaiheuttajia nopeasti lähellä potilasta.

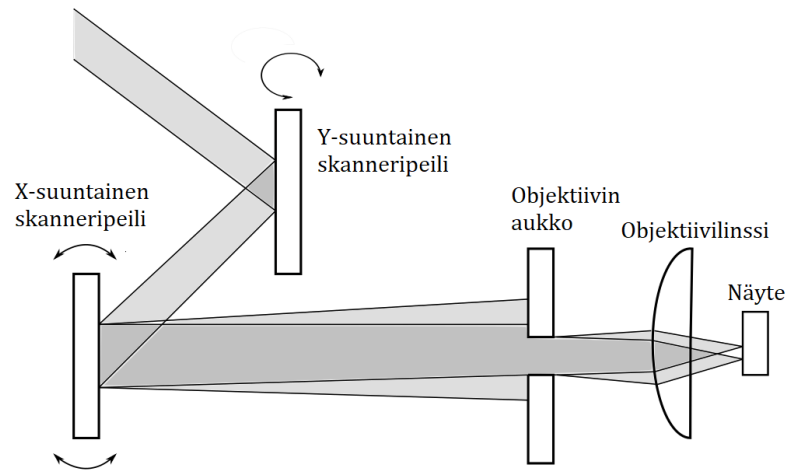
Laitteistossa taudinaiheuttajat tunnistetaan kappaleessa 1.3 kuvatun teknologian mukaisesti sekä fluoresenssisignaalin että sironneen laservalon avulla liuoksesta, jossa on fluoresoivalla aineella merkittyjä taudinaiheuttajia. Laitteistossa lasersädetä poikkeutetaan kahden pietsosähköiseen ilmiöön perustuvan skanneripeilin avulla, ja näin saadaan aikaan lasersäteen nopea pyyhkäisymäinen liike liuoksessa. Kun liuoksessa oleva partikkeli osuu lasersäteen tielle, säde siroaa partikkelista ja tämä sironnut valo havaitaan valodiodin avulla. Kun sironnut valo havaitaan, skannerit pysäytetään ja niiden viiveen kompensoimiseksi niitä otetaan hieman takaisinpäin, jolloin lasersäteen pitäisi osua suoraan partikkeliin. Partikkeli pysyy hetken lasersäteen fokuksessa optisten voimien vuoksi. Jos fokuksessa olevassa partikkelissa on fluoresoivaa ainetta, saadaan tästä myös havaittua fluoresenssisignaali valomonistinputken avulla. Skannerit ovat paikoillaan vähintään 40 millisekuntia ja jos tänä aikana ei havaita fluoresenssisignaalia, skanneripeilien pyyhkäisymäinen liike jatkuu. Jos taas fluoresenssisignaalia saadaan, peilit jatkavat liikettään heti fluoresenssisignaalin pudottua eli partikkelin kuljettua fokuksen läpi. Peilejä ei kuitenkaan pidetä paikoillaan yli sataa millisekuntia, sillä tänä aikana oletettavasti on jo saatu tarpeeksi fluoresenssisignaalia. Sirontasignaali on siis tärkeä partikkeleiden havaitsemisen

vuoksi, sillä vasta sirontasignaalia havaittaessa rekisteröidään fluoresenssisignaalia .

Optiikkaa pyrittiin uudistamaan, jotta sirontasignaali tunnistettaisiin helpommin ja jotta liuostaustan vaikutus fluoresenssisignaaliin olisi mahdollisimman vähäinen. Liuostaustaa aiheuttavat esimerkiksi mikropartikkeleihin sitoutumattomat leimatut vasta-aineet, näytteen mukana tullut biologinen materiaali ja esimerkiksi pölyhiukkaset. Muita haluttuja asioita uudistetulta optiikalta olivat tuotannollistettavuus, helppo säädettävyys ja alhainen tarve uudelleensäätämislle. Toistettavuuden piti myös olla hyvä, sillä samalla näytteellä tehtyjen peräkkäisten testien tulee antaa mahdollisimman sama tulos.

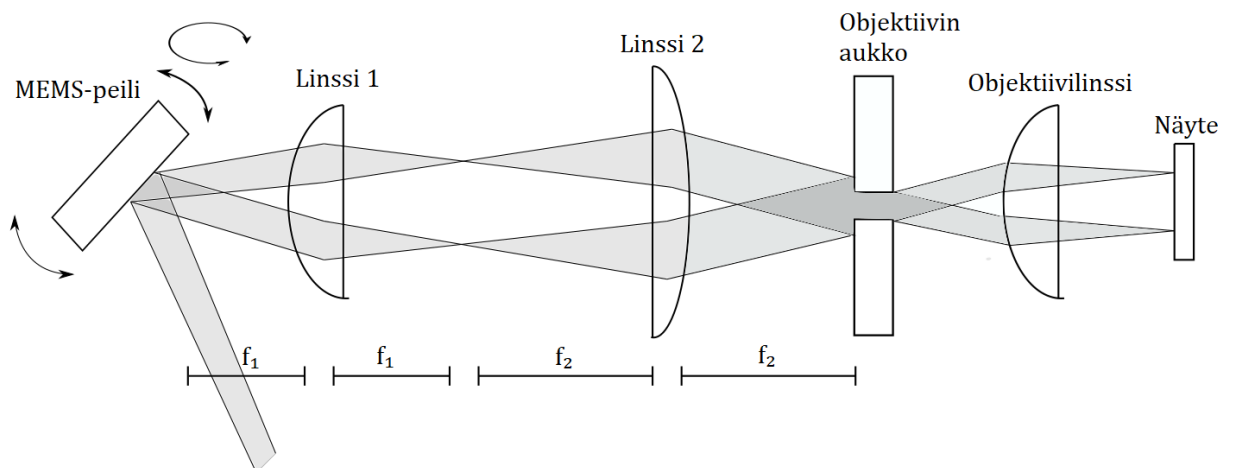
## 2.2 Optiikka

Kokeellisessa työssä rakennettiin ensin optinen polku sirontasignaalille. Alun perin laitteistossa oli kuvan 9 mukainen polku laservalolle. Divergoiva lasersäde, jonka profiili on Gaussinen, saapuu skanneripeileille ja peilien kulmista riippuen jatkaa matkaansa kohti mahdollisimman lähellä skannereita sijaitsevaa objektiivilinssiä tietyssä kulmassa. Linssi fokusoi säteen pieneen tilavuuteen näytteessä, joka on esimerkiksi tarkasteltavana olevaa liuosta sisältävä kyvetti tai kuoppalevyn kuoppa. Tilavuuden pienentämiseksi objektiivin sisääntuloaukko ylitäytetään, sillä muuten objektiivin numeerisesta aukosta käytettäisiin vain osa, ja fokuksen koko kasvaisi erityisesti syvyysuunnassa. Tästä ylitäytöstä seuraa, että mitä enemmän skannuskulma poikkeaa keskiasennosta, sitä enemmän teho vaimenee ja kuvausvirheet kasvavat säteen leikkautuessa epäsymmetrisesti.



Kuva 9: Tuotannossa olevan sirontasignaalin optinen polku. Kuva ei ole mittakaavassa.

Laservalon polku on ollut laitteistossa siis huomattavan yksinkertainen. Tässä työssä sirontasignaalin optinen polku muutettiin mukailemaan kappaleessa 1.2.1 olevan kuvan 3 optista polkua, jossa skannereiden ja objektiivilinsin välissä on kaksi linssiä. Kuvassa 10 on tämä polku.



Kuva 10: Skemaattinen kuva uudesta optisesta polusta.

Kuvassa 10 kahden skanneripeilin tilalla on kahteen suuntaan poikkeutettava

MEMS-peili. Peilin jälkeen kuvassa lasersäteen polulle on asetettu vielä kaksi tasokuperaa linssiä, joiden tasopinnat ovat vastakkain. Nämä linssit sekä suurentavat lasersäteen sopivaksi mikroskoopin objektiivin sisääntuloaukkoa varten, että ohjaavat säteen oikeaan suuntaan skanneripeilin liikuttamisen jälkeenkin. Kuvassa  $f_1$  tarkoittaa ensimmäisen linssin polttoväliä ja  $f_2$  toisen. Peilin ja ensimmäisen linssin etäisyyden tuli siis olla linssin polttovälin verran, jotta linssi kokoaa tulleen säteilyn toiselle puolelle polttovälin etäisyydellä olevalle tasolle. Tästä tasosta taas etäisyyden toiselle linssille tuli olla toisen linssin polttovälin verran, jotta linssi 2 kokoaa säteet taas objektiivin sisääntuloaukolle eli objektiivilinssille.

Uuden optisen polun etu vanhaan, yksinkertaiseen optiseen polkuun on kontrolloitu suurennos, sillä kahden linssin polttovälit määräävät miten paljon lasersäteen halkaisija on suurentunut objektiiville tullessa. Suurennos

$$M = \frac{f_2}{f_1},$$

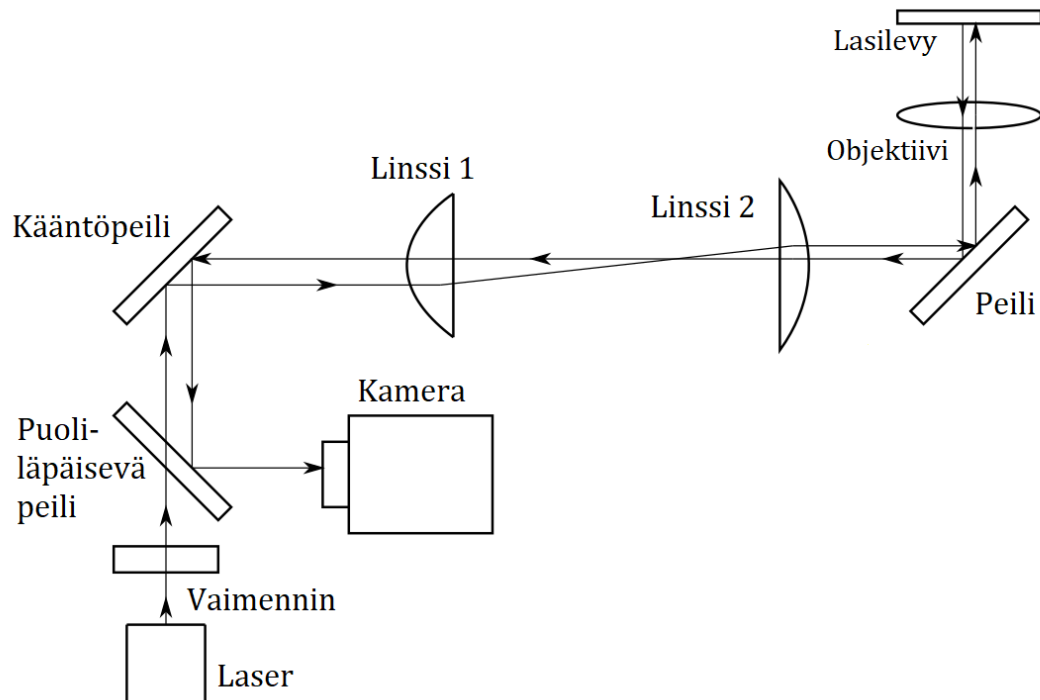
eli esimerkiksi jos ensimmäisen linssin polttoväli on 20 mm ja toisen 60 mm, on suurennos siis kolminkertainen. Suurennoksen kontrollointi on tärkeää, sillä laserin teho on kiinteä ja valonsäteen teho objektiivilinssin polttopisteessä vaikuttaa optisiin voimiin, jotka vetävät mikropartikkeleita fokuksen läpi. Suurennoksella saadaan säädettyä tehon suuruutta objektiivin sisääntuloaukolla ja näin myös objektiivin polttopisteessä.

Toinen uuden optisen järjestelmän ero vanhaan on se, että nyt lasersäteen saaminen kontrolloidusti tulemaan objektiivin sisääntuloaukolle ilman, että säteen leikkaantuminen riippuu kulmasta. Aikaisemmin pelkkään säteen luonnolliseen laajenemiseen perustunut optinen polku on ollut myös hyvin herkkä etäisyyksien muutoksille, mutta uudessa polussa eri linssiyhdistelmillä olisi mahdollista joko kasvattaa tai lyhentää lasersäteen kulkemaa matkaa. Aiemmin lasersäteen on täytynyt myös laajentua ennen kuin se saapuu skanneripeileille, mutta jos systeemin pietsosähköiseen ilmiöön perustuvat skanneripeilit halutaan vaihtaa yhteen pieneen MEMS-peiliin,

voi lasersäde olla edelleen huomattavan pieni peilille saapuessaan peilin pienen halkaisijan vuoksi.

### 2.2.1 Koejärjestely

Käytetty laser oli Nd:YAG-kidelaser, joka tuottaa 20 mikrosekunnin välein noin nanosekunnin pituisen laserpulssein. Laserin aallonpituus on 1064 nm, joka asettuu sähkömagneettisessa spektrissä NIR-säteilyn aallonpituusalueelle.



Kuva 11: Koejärjestely sirontasignaalin havaitsemiseksi.

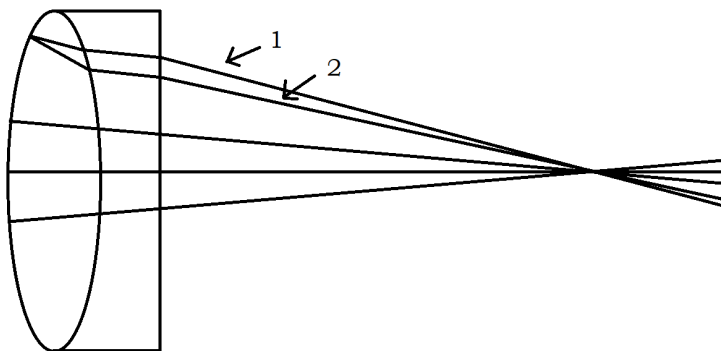
Tutkimuksessa kokeiltiin optisella pöydällä eri linssipareja kuvan 10 mukaisesti ja niiden soveltuvuutta tähän laitteistoon. Optiselle pöydälle tehty koejärjestely oli kuvan 11 mukainen.

Valo kulkee kuvassa 11 laserilta vaimentimen läpi pienen osan valosta heijastavan lasilevyn läpi kääntöpeilille. Vaimentimen tehtävä oli tutkimusvaiheessa vaimentaa lasersäteen tehoa, jotta säde ei polta reikää objektiivin päällä olevaan heijastavaan pintaan. Skanneripeiliä koejärjestelyssä korvaavana kääntöpeilinä käytettiin tässä tutkimusvaiheessa vain tavallista hyvin heijastavaa peiliä, jota pystyttiin kääntämään pöytäpinnan suuntaisesti. Kääntöpeilin jälkeen optiselle polulle asetettiin ensin pienemmällä polttovälillä varustettu linssi ja tämän jälkeen suuremman polttovälin linssi, jolloin nämä aikaansaivat lasersäteen poikkileikkauksen suurentamisen. Linssien jälkeen säde ohjattiin uudelle peilille, joka käänsi säteen pöytäpinnasta ylöspäin ja ohjasi sen ylöspäin asetetun mikroskoopin objektiivin sisääntuloaukkoon. Objektiivina käytettiin Leican 40-kertaisen suurennoksen ja 0,65:n numeerisen apertuurin linssiä, jollaista käytetään myös nyt käytössä olevissa laitteissa. Objektiivin päällä oli lasilevy jonka korkeutta kyettiin säätämään, jolloin se toimi sirottavan partikkelin vastineena heijastaen lasersäteen objektiivin polttopisteessä. Lasilevystä heijastunut lasersäde kulki optista polkua takaisin puoliläpäisevälle peilille asti, jossa se heijastui 90 astetta ilmaisimena toimivalle CMOS-kameralle. Kameran tyyppi oli Point Grey:n BFLY-PGE-2356M-C-kamera ja se oli kytketty tietokoneeseen, jossa kameran kuvaa tarkasteltiin Point Grey:n FlyCapture-ohjelman avulla.

Linssien sopivuutta tutkittiin niin, että optiselle polulle asetettiin tietty linssipari ja objektiivin päältä mitattiin teho Ophir-merkkisellä tehomittarilla. Tehon tuli olla objektiivin polttopisteessä tietyllä välillä, jotta optiset voimat kykenevät vetämään partikkelin polttopisteeseen, mutta eivät kuljeta mikropartikkelia liian nopeasti polttopisteen läpi. Kohdistuksen tarkkuutta tarkkailtiin myös kameran avulla, sillä jos optinen polku oli oikein rakennettu ja kohdistettu näkyi kameran kuvassa

selkeä symmetrinen heijastuskuva objektiivin fokusoimasta lasersäteestä.

Optisella pöydällä kokeiltavia linssejä oli neljä: 30 mm polttovälin linssi, 50 mm polttovälin linssi, 100 mm polttovälin linssi ja 150 mm polttovälin linssi. Linssit olivat tyypiltään akromaattisia duplettilinssejä, mikä tarkoittaa niiden rakenteen vähentävän pallo- ja väriaberraatiota. Kuvassa 12 on tällainen duplettilinssi, jossa ensimmäisen, kuperan, linssin taitekerroin on matalampi kuin toisen, puolikoveran, linssin [31]. Kuvaan merkityt valonsäteet 1 ja 2 kuvaavat eri aallonpituuksien säteitä, jotka kuitenkin lopulta fokusoituvat samaan pisteeseen paraksiaalisten eli linssille kohtisuorasti tulevien säteiden kanssa.

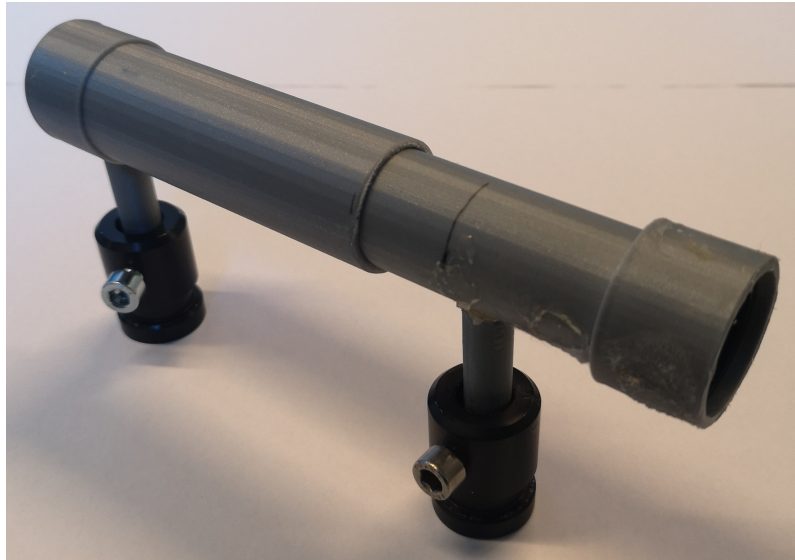


Kuva 12: Akromaattisen duplettilinssin rakenne ja periaate.

### 2.2.2 Linssien kokeellinen tarkastelu

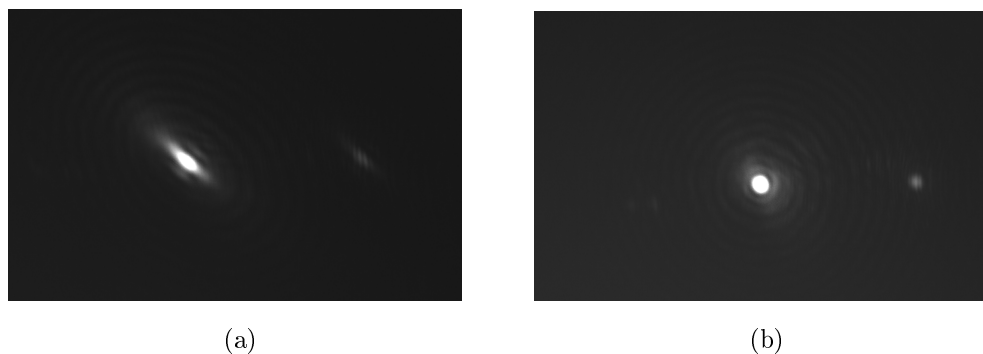
Kehyksettömille linssille suunniteltiin putkimaiset kehykset FreeCAD-suunnitteluohjelmalla ja nämä tulostettiin Ultimaker Cura 3D-tulostimella. Linssit asetettiin putkiin ja putket sisäkkäin, jolloin tiedettiin linssien olevan toistensa suhteen samassa asennossa. Kuvassa 13 on linssien sisäkkäin asetetut kehykset ja jalustat.

Kokeiltavia linssipareja oli yhteensä kuusi, sillä ensimmäisen linssin polttovälin piti olla lyhyempi kuin toisen. Systemin kohdistamisessa apuna käytettävän kameran kuvasta katsottiin minkä näköinen lasersäteen poikkileikkaus oli siinä kohdassa, mihin myöhemmin tulotaisiin asettamaan neulanreikä ja valodiodi. Kuvan 14 a-



Kuva 13: Linssien kehysputket ja näiden jalustat.

kohdassa on kameran kuva, kun piste oli pahasti aberroitunut. Tällaisessa tilanteessa voitiin arvioida, että esimerkiksi toinen linssi oli vinossa optisen polun suhteen tai, että etäisyys linssien välillä ei ollut optimaalinen, jolloin polku voitiin kohdistaa uudelleen. Kuvan 14 b-kohdassa taas lasersäteen poikkileikkaus näyttää lähestulkoon ympyrältä, joten voidaan päätellä että linssit on säädetty oikein. Kuvassa havaitaan myös himmeänä Airyn kuviot kirkkaan pisteen ympärillä olevina renkaina, mikä johtuu valon diffraktoitumisesta sen kulkiessa neulanreiän läpi. Molemmissa kuvissa havaitaan myös himmeät fokuksen kuvat kuvien oikeassa reunassa, mikä johtuu laservalon heijastumisesta myös puoliläpäisevän peilin laserin puoleiselta pinnalta, jolla on heijastuksenestopinnoite.



Kuva 14: Aberroitunut ja aberroitumaton laserpiste neulanreiän kohdalla, kun systeemin linssit olivat polttoväleiltään 50 mm ja 150 mm.

Optisella pöydällä kokeiltiin eri linssikonfiguraatioita ja tehon sopivuus objektiivin päältä mitattuna koottiin taulukkoon 1. Tehoa verrattiin olemassaolevan laitteen objektiivin päältä mitattuun tehoon, sillä tämän tehon tiedettiin olevan tarpeeksi suuri saamaan mikropartikkelit fokukseen.

Taulukko 1: Linssiparien tehon sopivuus koejärjestelyyn.

Linssi 1:sen polttoväli (mm)	Linssi 2:sen polttoväli (mm)	Teho	Suurennos
30	50	Liian suuri	1,67
30	100	Sopiva	3,33
30	150	Liian pieni	5
50	100	Liian suuri	2
50	150	Sopiva	3
100	150	Liian suuri	1,5

Kuten taulukosta nähdään, linssipareja joiden aikaansaaman suurennos aiheutti sopivalle alueelle osuvan tehon objektiivin päällä oli kaksi. Ensimmäisessä linssiparissa ensimmäisen linssin polttoväli oli 30 mm ja toisen 100 mm ja toisessa linssiparissa ensimmäisen linssin polttoväli oli 50 mm ja toisen 150 mm. Näistä päädyttiin käyttämään linssiparia 30 mm ja 100 mm, sillä lyhyemmät polttovälit ja näin lyhyempi

optinen polku olisivat eduksi tuotannollisessa laitteistossa.

## 2.3 Elektroniikka ja sulautetut järjestelmät

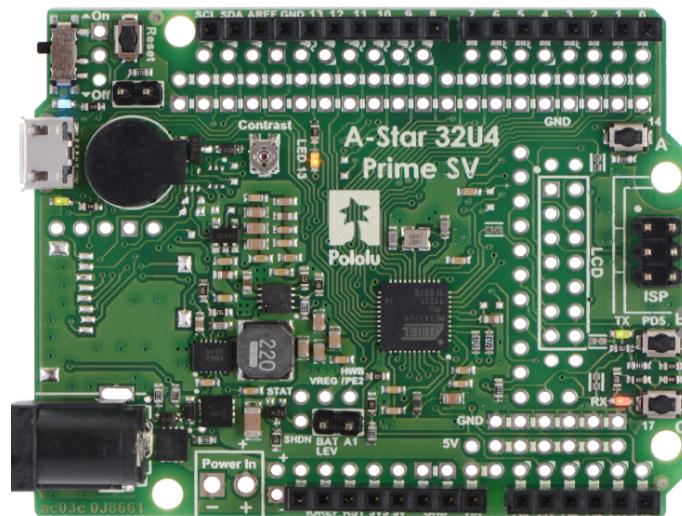
Laitteistossa on tähän asti käytetty pietsosähköiseen ilmiöön perustuvia skannereita, jotka liikkuvat sähkövirran aiheuttaman mekaanisen jännityksen vuoksi. Tähän asti joka laitteessa on ollut kaksi skanneria, joista toinen on poikkeuttanut lasersädetä Y- ja toinen X-suuntaan. Pietsoskannereiden tilalle kokeiltavaksi MEMS-peiliksi valikoitui Mirrorcle Technologies Inc.:in Integrated MEMS Mirror A5M24.2-2400AL siksi, että sitä oli mahdollista ajaa sekä X- että Y- suuntaan ja sen halkaisija oli niin suuri (2,4 mm), että koko lasersäde mahtui siihen.

### 2.3.1 MEMS-peilin ajaminen A-Star 32U4 Prime SV:n avulla

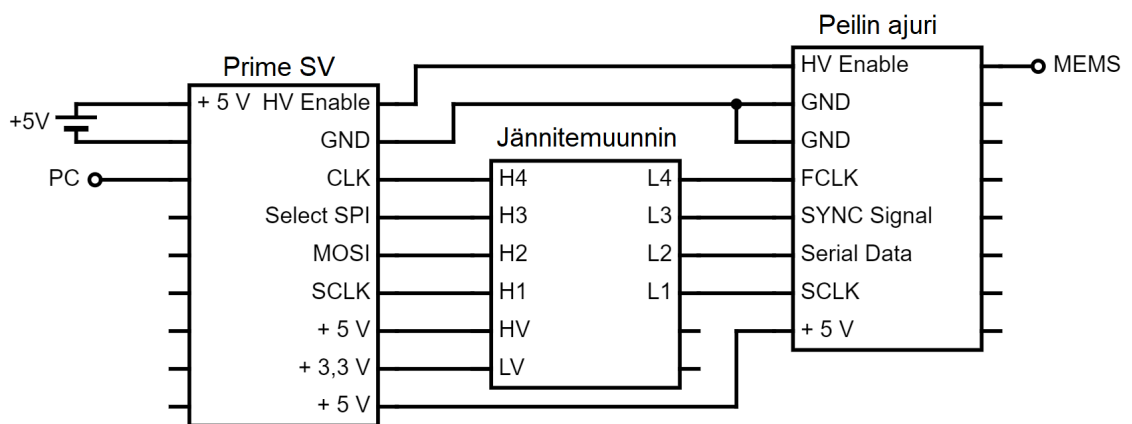
MEMS-peili, johon noin pankkikortin kokoinen ajuripiiri oli kytketty, kiinnitettiin optiseen pöytään aiemmin käytetyn kääntöpeilin tilalle. Ensimmäinen MEMS-peilin ajamiseen kokeiltava prototyyppi tehtiin Pololu Corporationin A-Star 32U4 Prime SV -ohjelmoitavan piirin avulla. Pololun A-Star 32U4 Prime SV on Arduino Leonardo -tyyppinen mikrokontrolleripiiri, joka on suunniteltu pienelektronikan prototyyppien rakentamista ja kokeilemistä varten. Kuvassa 15 on Pololun A-Star 32U4 Prime SV.

A-Star 32U4 Prime SV:tä ajettiin ohjelmalla Arduino IDE, joka on avoimen lähdekoodin alusta sulautettujen järjestelmien testaamiseen. Tietokoneeseen mikrokontrolleripiiri kytkettiin USB-kaapelilla. Mikrokontrolleripiiri kytkettiin kiinni jännitemuuntajan kautta peilin ajuriin, joka oli yhdistetty peiliin. KytKentä oli kuvan 16 mukainen.

Peilin ajurin periaate oli se, että kuvan 16 mukaiseen Serial Data-nastaan ajettiin signaalia, jonka peilin ajurissa oleva DA-muunnin muutti analogiseksi signaaliksi. Tästä signaalista se poimi erikseen peilin X- ja Y-kanaviin tarvittavat jännitteet



Kuva 15: A-Star Prime SV [32].

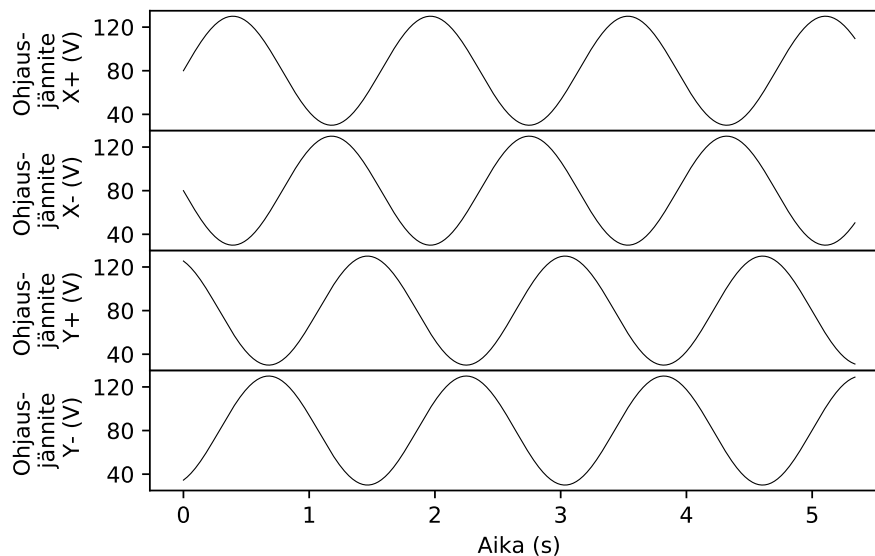


Kuva 16: A-Star Prime SV:n kytkentä MEMS-peilin ajuriin.

ja ohjasi nämä eri kanaviin. Kun myös HV Enable-nastaan lähetettiin signaalia, signaalit vahvistettiin ajurissa korkeajännitepuolella 80 V molemmin puolin oleviin jännitteisiin. Nämä X- ja Y-kanavien korkeajännitteet ajettiin lopulta peiliin, joka liikkui niiden mukaan. A-Star Prime SV:n ja peilin ajurin väliin kytketyn jännitemuuntimen tehtävä oli muuntaa mikrokontrolleripiiristä tulevat viiden voltin logiikkatasoiset signaalit käyttämään 3,3 voltin logiikkaa, jota peilin ajurin data- ja kellosignaalinastat käyttivät.

Peilin ajamiseen Arduino IDE:ssä käytettiin Githubista haettua ohjelmaa [LIITE

A] ja ajurin ohjelmointiin käytettävää kirjastoa AD56X4.h [33]. Ohjelman periaatteenä oli ajaa sinimuotoista jännitettä jokaiseen MEMS-peilin kanavaan niin, että peili tämän johdosta tekee ympyräliikettä. Kuvassa 17 on esitetty neljään kanavaan ajetus ohjausjännitteet. Kuvasta huomataan, miten saman ulottuvuuden positiiviseen ja negatiiviseen kanavaan ajetus jännitteet ovat vastakkaiset.

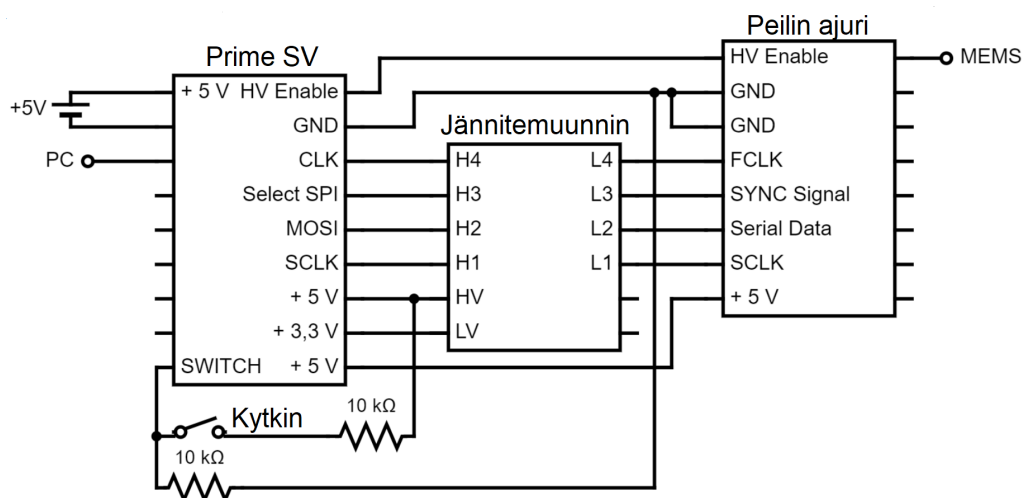


Kuva 17: MEMS-peilin kanaviin ajetus ohjausjännitteet, jotta peili tekee ympyräliikettä.

Tämä koodi ajettiin suoraan tietokoneeseen kiinnitettyyn A-Star Prime SV:hen, joka oli kiinnitetty peilin ajuriin ja tämä taas peiliin. Ohjelma toimi ja peili saatiin tekemään ympyräliikettä, joka havaittiin osoittamalla peiliä lasersoittimella ja seuraamalla heijastuneen lasersäteen liikettä. Peiliä ajettaessa tuli olla varovainen, jotta peilin kulmia ei vaihdeta liian nopeasti tai että kahden peräkkäisen asennon välinen ero ei ole liian suuri.

Kytkeä muuttettiin seuraavaksi hieman alkuperäisestä, jotta korkeajännitteen puolta pystyttiin kontrolloimaan paremmin. Korkeajännite oli tarpeellista kytkeä pois päältä aina kun tehtiin muutoksia kytkentään, sillä jokin sähköinen häiriö korkeajännitteessä olisi voinut hajottaa peilin. Tällä tavoin sähköisille häiriöille herkkä

MEMS-peili saatiin aina esimerkiksi laitettua turvallisesti systeemiin kiinni. Kytkentään lisättiin kytkin, jonka ollessa kiinni Prime SV sai signaalin SWITCH-nastaan ja ajoi HV Enable-nastaan signaalia, jolloin MEMS-peilin ajurin korkeajännitepuoli alkoi toimia. Jos kytkin avattiin, Prime SV ei enää saanut signaalia SWITCH-nastaan ja HV Enable ei enää vienyt signaalia peilin ajuriin ja korkeajännitepuoli ei toiminut. A-Star Prime SV:n ja MEMS-peilin ajurin kytkentä on kuvassa 18.

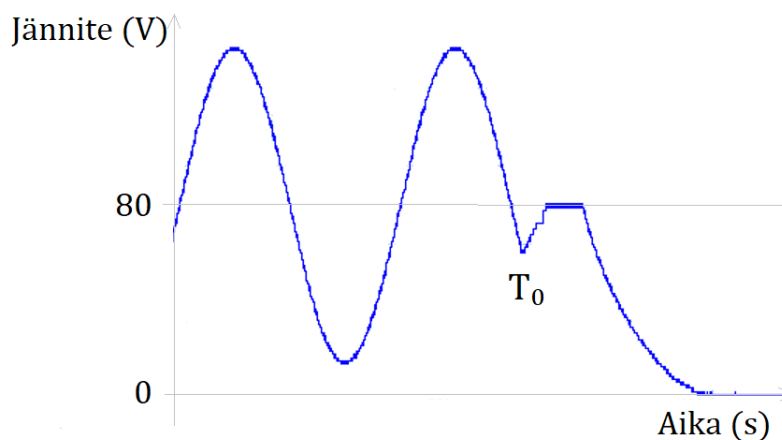


Kuva 18: A-Star Prime SV:n ja MEMS-peilin ajurin kytkentä, jossa on mukana erillinen HV-puolta kontrolloiva kytkin.

Kytkimen käyttöön tarvittavat muutokset tehtiin myös Arduino IDE:n ohjelmaan [LIITE B]. Tärkeää oli kytkimen avaamisen jälkeen ajaa MEMS-peili portaittain taas nollassa, sillä muuten olisi ollut mahdollista esimerkiksi peilin liian nopea liike ääriasennosta nollassa, mikä olisi voinut rikkoa peilin. Kun peili on ajettu nollassa, HV Enable-nastan signaali katkeaa ja korkeajännitepuoli lakkaa toimimasta.

Kytkimen toimintaa tutkittiin oskilloskoopin avulla ja kuvassa 19 on kuva oskilloskoopin signaalista peilin positiivisesta X-kanavasta sinä hetkenä, kun kytkin avataan ja peili lopettaa ympyräliikkeen. Kuvasta huomataan miten jännite siirtyy portaittain 80 volttiin ja peili nollassa ja tämän jälkeen, kun korkeajännite

puretaan, laskee jännite nolnaan.



Kuva 19: Jännitekäyrä kytkimen sulkuhetkellä.

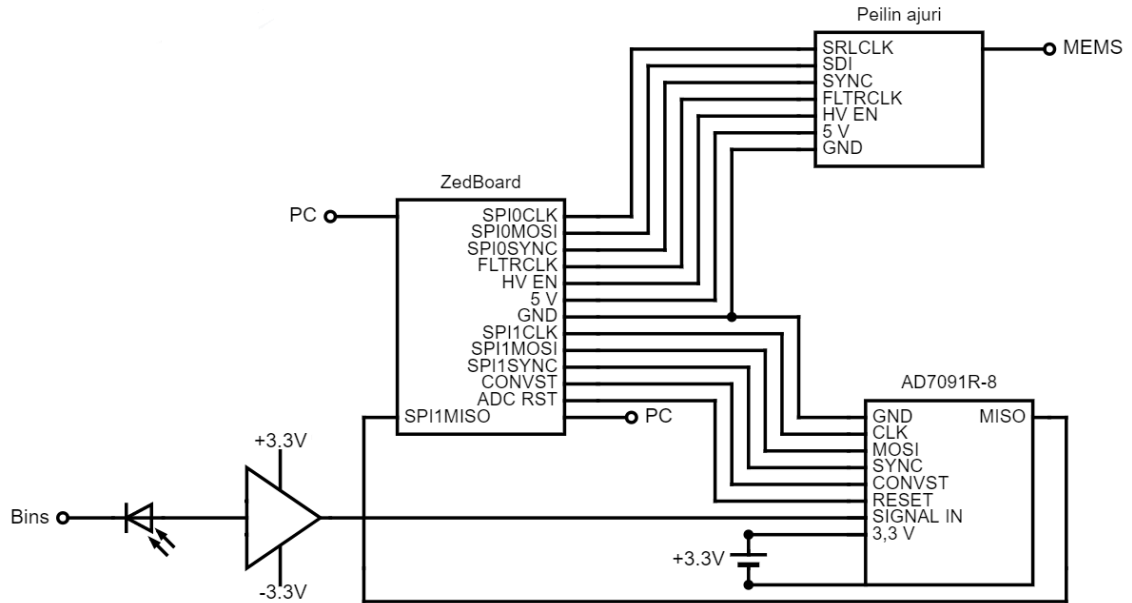
### 2.3.2 MEMS-peilin ajaminen ZedBoardin avulla

Kun edellisessä kappaleessa oli todettu peilin toimivan halutulla tavalla, tehtiin seuraavaksi MEMS-peilille kytkentä Digilentin ZedBoard-tyypin ohjelmoitavalle systeemilevyille. ZedBoard Zynq-7000 ARM/FPGA Soc Development Board on uudelleenohjelmoitava kehityslevy, jonka avulla voidaan ohjelmoida FPGA-piirejä (engl. field programmable gate array), eli uudelleenohjelmoitavia mikropiirejä.

KytKentä tehtiin tälle levyille, sillä sen avulla kyettiin lukemaan myös sirontasignaali. Kun fokuksessa olevasta mikropartikkelista sirosi osa lasersäteestä, kulki tämä sironnut valo konfokaalisesti sijoitetun neulanreiän läpi valodiodille. Valodiodi havaitsi saapuneet fotonit ja lähetti analogisen signaalin AD-muuntimelle (engl. analog to digital converter), mikä muunsi signaalin digitaaliseksi. Digitaalinen signaali kulki ZedBoardille, josta se luettiin tietokoneella. Käytössä ollut neulanreikä oli halkaisijaltaan  $78 \mu\text{m}$  ja se asetettiin kuvassa 11 olevan kameran paikalle. Neulanreiän taakse asetettiin valodiodi, joka kykeni havaitsemaan laserin aallonpituuden valoa. Neulanreiän etäisyyden puoliläpäisevältä peililtä piti olla saman kuin puoliläpäisevän peilin etäisyys laserilta, jotta palaava sirontasignaali kohdistuisi oikein

neulanreiän kohdalle.

Kuvassa 20 on AD-muuntimen sekä MEMS-peilin ajurin kytkentä ZedBoardiin.



Kuva 20: ZedBoardin kytkentä MEMS-peilin ajuriin ja AD-muuntimeen.

Kuvassa peilin ajurin kytkentä ZedBoardiin on hyvin samanlainen kuin mitä sen kytkentä ensimmäisessä prototyypissä A-Star Prime SV:hen oli. Signaalinvahvistuskortin vahvistama valodiodilta saatu sirontasignaali ajettiin AD-muuntimen signal in -nastaan. MISO-nastasta (engl. master in slave out) digitaaliseksi muutettu signaali ajettiin ZedBoardille. Synkronisaatiosignaali saatiin AD-muuntimelle ZedBoardin nastasta SPI1SYNC ja CONVST-nastasta taas signaali, jonka mukaan datasiinaalinvaihto kahden laitteen välillä aloitettiin.

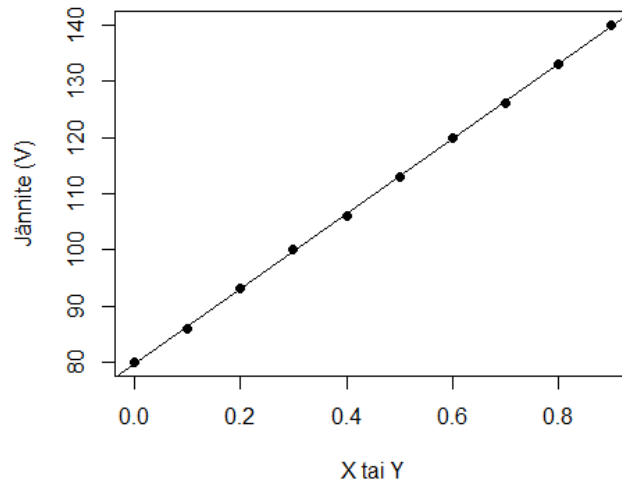
ZedBoardin ohjelmointiin käytettiin Xilinx Inc.:n Vivado Design Suitea, joka on järjestelmäpiirien ohjelmoimiseen tarkoitettu kehitysympäristö. Vivado Design Suiteella hallittiin järjestelmäpiirin arkkitehtuuria, ja piirin ohjelmointi tapahtui Xilinx SDK:n avulla C-ohjelmointikielellä. Käytetyssä C-ohjelmassa määriteltiin spi-väylän toiminta tiedostossa spi.c, mittauksen toiminta tiedostossa tpx\_measurement.c ja koko ohjelman suoritus tiedostossa measurement.c [LIITE C].

Ohjelmassa olevien koordinaattien vastaavuudet peiliin ajettaviin jännitteisiin kerättiin taulukkoon 2. Nämä float-datatyypin arvot asetettiin `moveMirrorTo()`-funktion argumenteiksi.

Taulukko 2: Koordinaattien vastaavuus peiliin ajettaviin jännitteisiin.

X tai Y	Jännite
0,0	80
0,1	86
0,2	93
0,3	100
0,4	106
0,5	113
0,6	120
0,7	126
0,8	133
0,9	140

Kuvassa 21 on näistä vastaavuuksista piirretty kuvaaja, johon sovitetun suoran yhtälöksi saatiin  $X = Y \cdot 66,63 \text{ V} + 79,70 \text{ V}$ . Tässä X on jännite ja Y `moveMirrorTo()`-funktion argumentti.



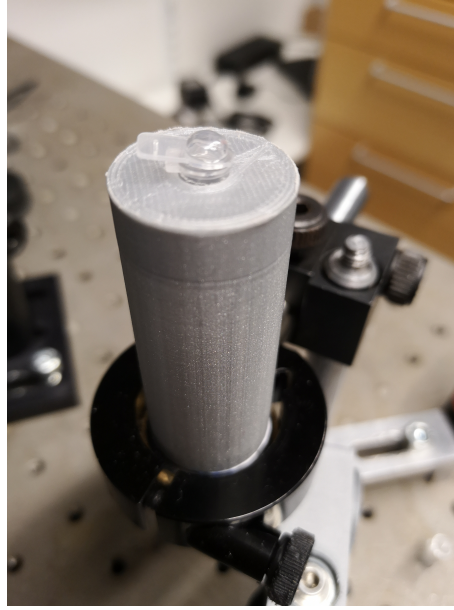
Kuva 21: Ajettavan jännitteen riippuvuus koordinaatista.

## 2.4 Sironta- ja fluoresenssisignaalin mittaaminen ja analysointi

### 2.4.1 Sirontasignaalin mittaaminen mikropartikkeliliuoksesta

Sirontaa mitattiin liuoksesta, joka sisälsi pallomaisia, halkaisijaltaan  $3 \mu\text{m}$ , mikropartikkeleita  $2,0 \cdot 10^6$  kpl/ml. Koejärjestely oli muuten kuvan 11 mukainen, mutta kääntöpeilin tilalla oli nyt MEMS-peili, lasilevyn tilalla mikropartikkeliliuosta sisältävä  $200 \mu\text{l}$  yksittäiskyvetti ja kameran paikalla neulanreikä ja tämän takana ZedBoardiin kytketty valodiodi. Kyvettiä varten objektiivin päälle suunniteltiin ja 3D-tulostettiin pidike (kuva 22).

Xilinx SDK:lle tehdyn ohjelman avulla [LIITE C] MEMS-peiliä liikutettiin niin että se skannasi neliönmuotoista aluetta jaksottomasti pienin askelvälein. Tämä tehtiin Lissajous-käyrien avulla, sillä näin oli mahdollista saada aikaan kuvio, jossa koko



Kuva 22: 3D-tulostettu pidike kyvetille.

skannausalue katettiin ilman toistuvia skannauskuvioita. Esimerkiksi kun

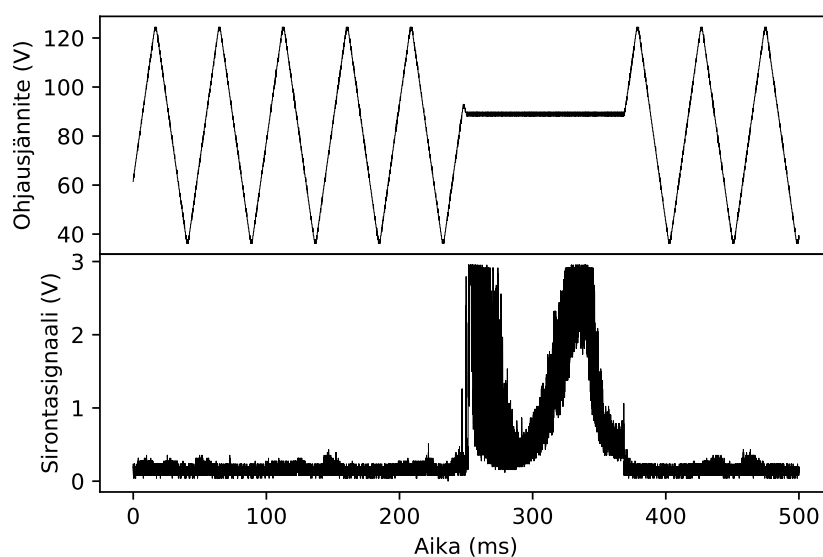
$$x = A_x \sin(\omega_x t) \text{ ja} \quad (4)$$

$$y = A_y \sin(\omega_y t), \quad (5)$$

jossa  $x$  on peilin x-suuntainen paikka ajanhetkellä  $t$  ja  $y$  y-suuntainen paikka samalla ajanhetkellä,  $A_x$  ja  $A_y$  amplitudit ja  $\omega_x$  ja  $\omega_y$  kulmataajuudet, saadaan koko alueen  $2A_x \times 2A_y$  kattava epäjatkuva skannaus, kun kulmataajuuksien suhde on irrationaalinen [34]. Tässä koejärjestelyssä skannattiin yleensä Lissajous-kuvioilla skannatessa käytettyjen siniaaltoliikeratojen sijaan kolmioaaltojen muotoisilla liikeradoilla, sillä näitä oli käytetty myös aiemmin pietsosähköisillä skanneripeileillä sädettyä liikuttellessa. Samoin MEMS-peilin nopeus pyrittiin pitämään koejärjestelyssä samanlaisena, kuin alkuperäisissä pietsosähköisissä skannereissa.

Xilinx SDK:lle tehdyssä ohjelmassa skannattiin tällä tavalla niin, että MEMS-peilin kulma sekä X- että Y-suuntaan oli maksimissaan  $\pm 2^\circ$ . Kun valodiodista havaittiin ZedBoardin avulla tarpeeksi korkea sirontasignaali, peili pysäytettiin ja sitä ajettiin hieman taaksepäin. Tämä tehtiin sen vuoksi, että kun mikropartikkeli

skannatessa osuu fokukseen, sinä ajanhetkenä mikä kuluu, kun signaali luetaan ja skanneri pysäytetään, on fokus ehtinyt jo liikkua mikropartikkelin ohi. Kun peili oli liikkunut hieman takaisinpäin ja pysähtynyt, jatkettiin signaalin mittaamista. Takaisinventoaskeleksi haarukoitiin 20 kertaa askelvälin pituus. Kuvassa 23 on kuva sekä sirontasignaalista että peilin positiiviseen X-kanavaan ajetusta ohjausjännitteestä. Sirontasignaalin korkein arvo on noin 2,91 voltia sillä ADC:n maksimi-arvo oli 4095.

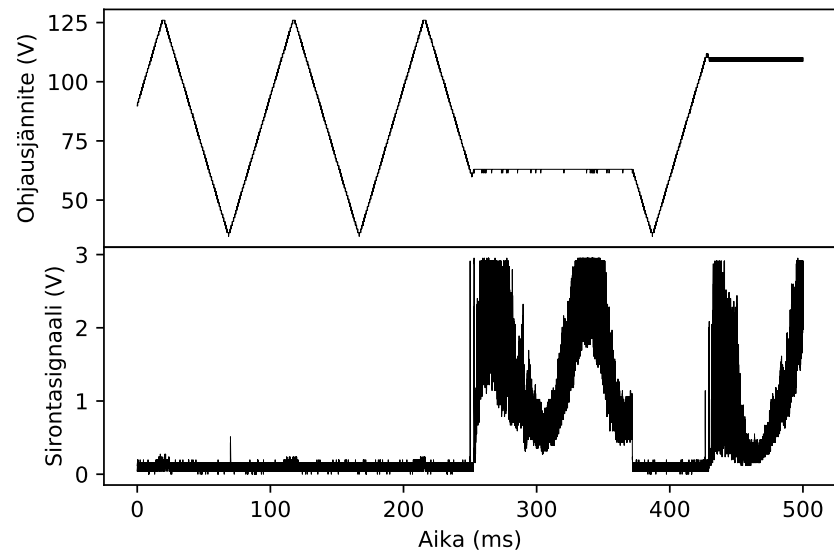


Kuva 23: Positiivisen X-kanavan ohjausjännite ja sirontasignaali.

Kuvasta 23 huomataan miten peilin positiiviseen X-kanavaan ajetaan kolmioaalton muotoista jännitettä, kunnes sirontaa havaitaan, jolloin peili palaa hieman takaisin ja pysähtyy. Kun sirontasignaalia on havaittu tietty aika, tässä noin 100 ms, ja havaitaan signaalin laskevan, jatketaan taas peilin liikettä.

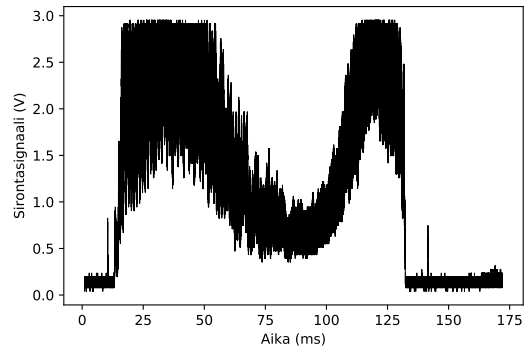
Kuvassa 24 on sekä peilin positiivisen Y-kanavan jännite että sirontasignaali ajan funktiona. Sirontasignaali ei ole sama kuin kuvassa 23, mutta kuvien 23 ja 24 peilin ohjausjännitekäyriä vertailemalla nähdään että kolmioaaltojen taajuus on erilainen X- ja Y-suunnissa, jotta Lissajous-kuvio kattaa koko alueen.

Saatujen sirontasignaalien muotoa ja intensiteettiä vertailtiin nykyisin tuotan-

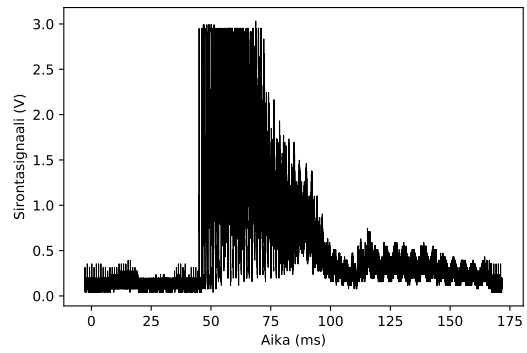


Kuva 24: Positiivisen Y-kanavan ohjaujännite ja sirontasignaali.

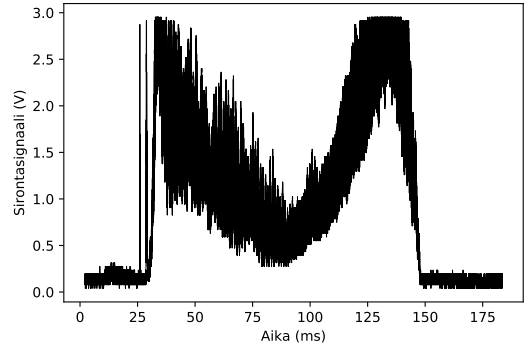
nossa olevien laitteiden sirontasignaaleihin. Signaalin koko ja muoto nähtiin oskilloskoopilla, josta kerätystä sirontadatasta piirrettiin kuvan 25 signaaliprofiilit.



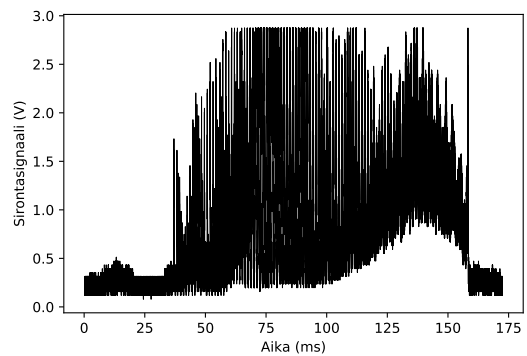
(a)



(b)



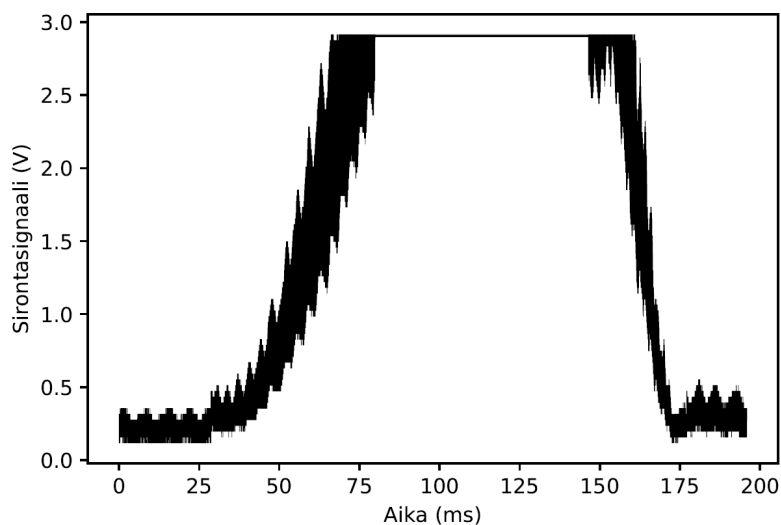
(c)



(d)

Kuva 25: Sironsignaaleja.

Kuvan 25 signaaleista huomataan, että yhdestä partikkelista saadaan signaalia yli 100 millisekuntia kaikissa paitsi kuvan 25 b-kohdassa. Tämä noin 100 millisekuntia on samaa suuruusluokkaa kuin aiempien laitteiden partikkeli-fokuksessaoloaika. Sirontasignaalien muoto sen sijaan ei ole sellainen kuin sen pitäisi olla. Kuvan 25 a- ja c-kohdassa näkyy se miten partikkeli ilmeisesti ensin vedetään fokukseen, sillä sironta nousee nopeasti 2,91:een volttiin partikkelin havaitsemisen jälkeen. Tämän jälkeen kuitenkin partikkeli ei ole kokonaan fokuksessa, sillä signaali laskee huomattavasti kunnes taas nousee ennen kuin partikkeli kokonaan häviää fokuksesta. Kuvan 25 b-kohdassa partikkeli ei taas pysy tarpeeksi pitkää aikaa fokuksessa ja kohdassa d sirontasignaali on liian matala. Partikkelit eivät siis pysy fokuksessa niin stabiilisti kuin niiden pitäisi, sillä sirontasignaalista saatujen kuvien pitäisi näyttää enemmän kuvan 26 signaalilta.

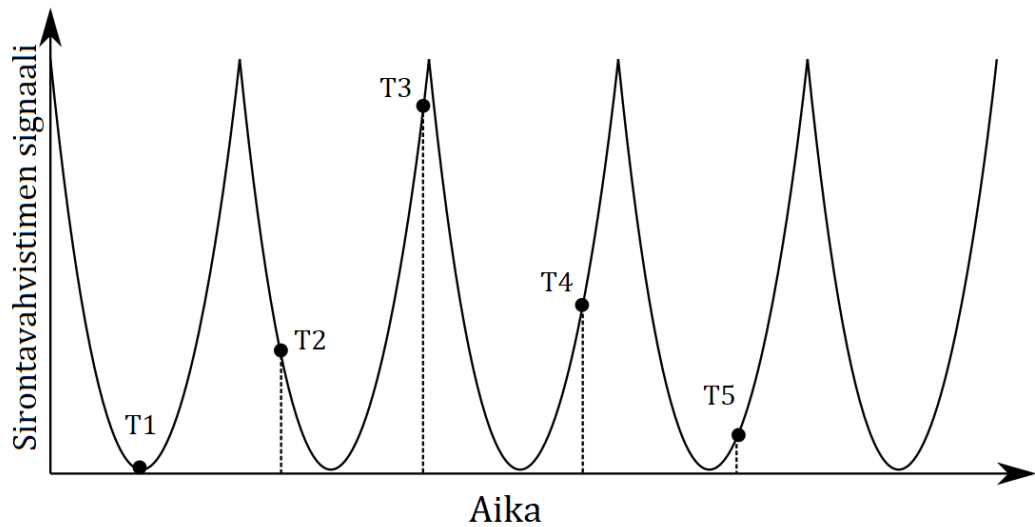


Kuva 26: Tavoiteltava sirontasignaali.

Kuvassa 26 signaali pysyy huipussaan noin 100 ms ja laskee vasta kun partikkeli on kulkenut fokuksen läpi. Koska sirontasignaalit eivät koejärjestelyssä olleet tarpeeksi voimakkaita ja stabiileja, päätettiin tutkia sirontasignaalia skannaamalla objektiivin päälle asetettavaa hilaa ja mittakaava-asteikkolevyä.

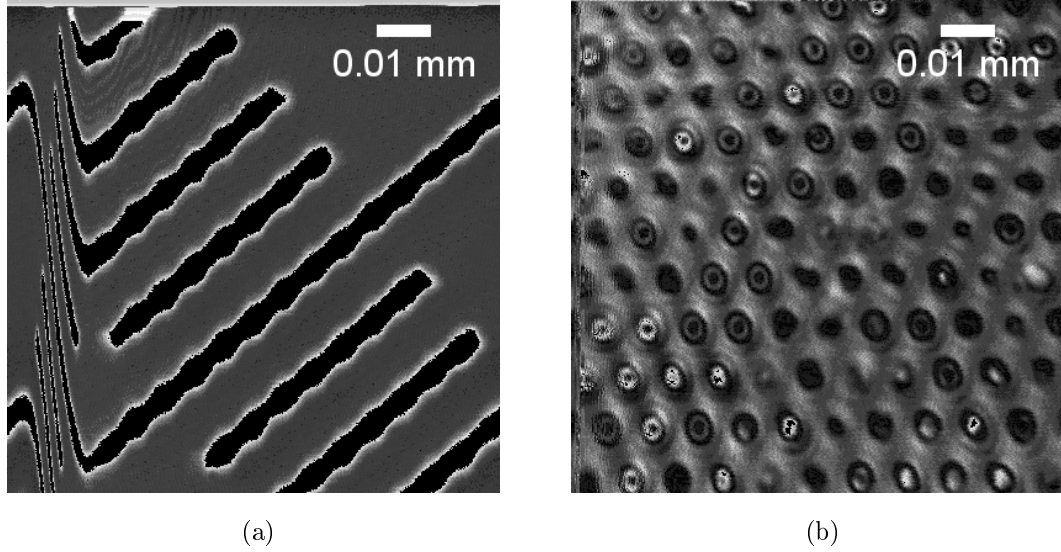
### 2.4.2 Mittakaavan ja hilan sironnan mittaaminen

Tehtiin koejärjestely, jossa mitattiin objektiivin polttopisteeseen asetetusta hilalevystä sekä mittakaavalevystä saatua sirontaa. Kun MEMS-peilillä tehtiin rasteris-kannausta ja joka kerta kun peiliä liikutettiin tallennettiin koordinaatti sekä koordinaatista saatu sirontasignaali datavektoriin, saatiin lopulta datavektorista luettua sirontakuva. Sirontasignaaleja luettiin viisi kappaletta joka kerta kun peiliä liikutettiin ja näistä datavektoriin tallennettiin mediaaniarvo. Tämä tehtiin sen vuoksi, että AD-muuntimen näytteenottoa ei ollut tahdistettu laserpulsseihin. Jokaisen pulsin jälkeen sirontavahvistimen signaali nousi nopeasti huippuarvoonsa ja laski taas ennen seuraavaa pulssia. Kuvassa 27 on skemaattinen kuva tästä sirontasignaalien mediaaniarvoperiaatteesta.



Kuva 27: Skemaattinen kuva laserpulsseista ja sironnan mitta-ajoista.

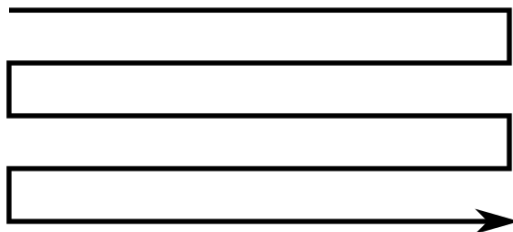
Kuvan 27 käyrä kuvastaa sirontavahvistimen signaalin intensiteettiä ajan funktiona ja tähän merkityt mittapisteet T1-T5 ajanhetkiä, jolloin sirontasignaali luettiin. Näistä sirontasignaaleista tallennettiin se arvo, joka oli intensiteetiltään näiden viiden luetun arvon mediaaniarvo. Esimerkiksi kuvan 27 ajanhetkistä ajanhetkenä T2 mitattu sirontasignaali olisi lopulta tallennettu.



Kuva 28: Kohdassa a on mittakaavasta saaduista sirontasignaaleista ja kohdassa b hilasta saaduista sirontasignaaleista koostettu kuva.

Kuvassa 28 on tarkkuudella 400 x 400 pikseliä ja 0,0005:n askeleella otetut sirontasignaalikuvat hilasta ja mittakaavasta. Kuviin pystyttiin asettamaan mittaasteikko, sillä mittakaavassa olevien viivojen tiedettiin olevan 10  $\mu\text{m}$ :n etäisyydellä toisistaan. Näissä kuvissa peiliä käännettiin aina toisesta maksimiasennosta toiseen, niin että kun peili oli X-suunnassa maksimikulmassaa, palautettiin se nopeasti X-suunnan minimikulmaan ja portaittainen skannaaminen aloitettiin uudelleen.

Kuten kuvasta 28 nähdään, sirontasignaalin tarkkuus vaikuttaa hyvältä. Kuvan 28 a -kohdassa nähdään vasemmassa reunassa kuitenkin häiriötä, mikä ilmeisesti johtuu osittain peilin nopeasta kääntämisestä toisesta reunasta toiseen, mutta on myös mahdollista että lasilevy on heilahtanut tai ollut vinossa objektiivin päällä mittauksen aikana. Häiriötä on myös havaittavissa kuvan 28 b -kohdan kuvan vasemmassa reunassa, samoin kuin lähellä tämän kuvan yläreunaa. Koska peilin nopea liike ääriasennosta toiseen on voinut aiheuttaa sähköisen häiriön, tämän eliminomiseksi mittakaavan ja hilan skannauskuvio muutettiin kuvan 29 mukaiseksi, jossa ääriasennot vältetään skannaamalla joka toinen rivi eri suuntaan. Skannaus tehtiin suunnanvaihtoja lukuun ottamatta vakionopeudella.

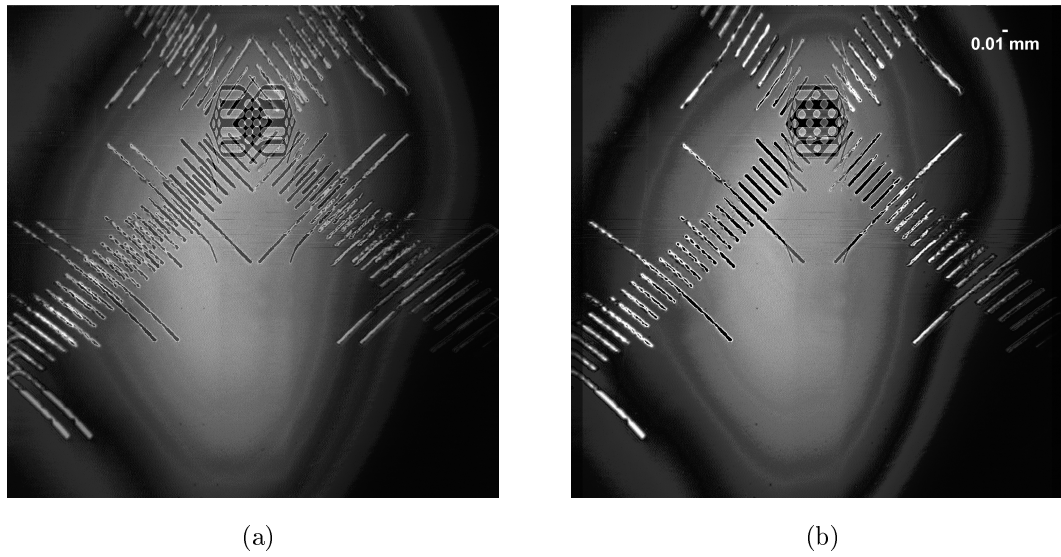


Kuva 29: Hilan ja mittakaavan skannauskuvio.

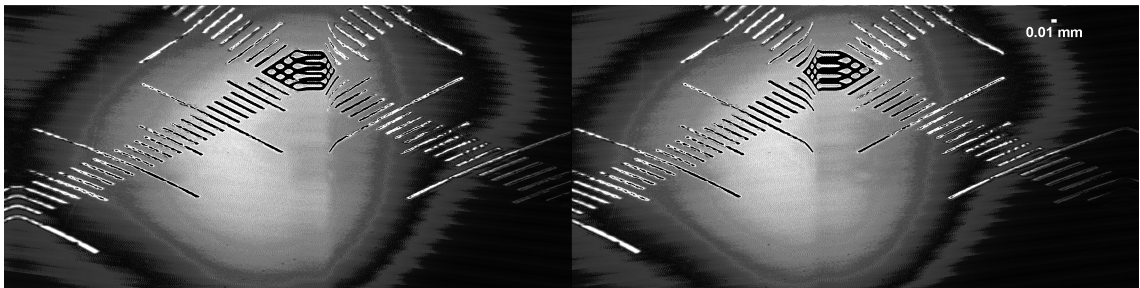
Kuvassa 30 a -kohdassa on tällä tavalla mittakaavasta skannattu kuva, jonka tarkkuus on 2000 x 2000 pikseliä. Askelen pituus tässä on ollut 0,000707 ja skanneri on pysäytetty kuvan keskellä Y-akselin mukaisesti 100:n askelen verran. Kun skanneri pysäytettiin aina keskellä skannatessa ja arvojen tallentamista datavektoriin jatkettiin, pystyttiin vertailemaan pysähtykö peili heti kun ohjelmallisesti se pysäytetään vai jatkaako se vielä hetken liikettään. Kuvan 30 a -kohdasta kuitenkin huomataan miten X-akselin suuntaiset rivit eivät ole kohdakkain. Tämä johtuu sähköisestä hystereesistä, jonka vuoksi rivi on kohdistunut eri tavalla riippuen kumpaan suuntaan skanneripeili on liikkunut rivillä. Tätä varten rivit kohdistettiin ImageJ-ohjelman avulla ja tämä kohdistettu kuva on kuvan 30 b -kohdassa. Kohdistus tehtiin siirtämällä joka toista riviä 93 pikseliä lähemmäs näiden väliin jääviä rivejä. Tämä arvo haarukoitiin.

Kuvan 30 b-kohdasta nähdään miten keskellä olevia 100 pikseliä, joiden aikana skanneri on ollut pysäytettynä on edelleen hankala tarkastella, sillä nyt pysäytyskoh- ta riippuu myös rivin skannaussuunnasta. Kuva 31 on samasta kohtaa mittakaavaa otettu kuva, mutta eri suuntiin skannatut rivit on eroteltu, joten kuvaa on helpompi tarkastella kun se on jaettu kahtia.

Kuten kuvasta nähdään, peili ei pysähdy heti, kun se pysäytetään, vaan se jatkaa matkaansa noin 40 pikselin verran. Tämä nähdään helposti kuvasta 32, jossa on suurennos kuvan 31 vasemman reunan kuvion ristikosta. Ristikokuvion pitäisi näyttää siltä, että sen keskellä kuva on venytetty 100 pikselin verran, mutta ku-



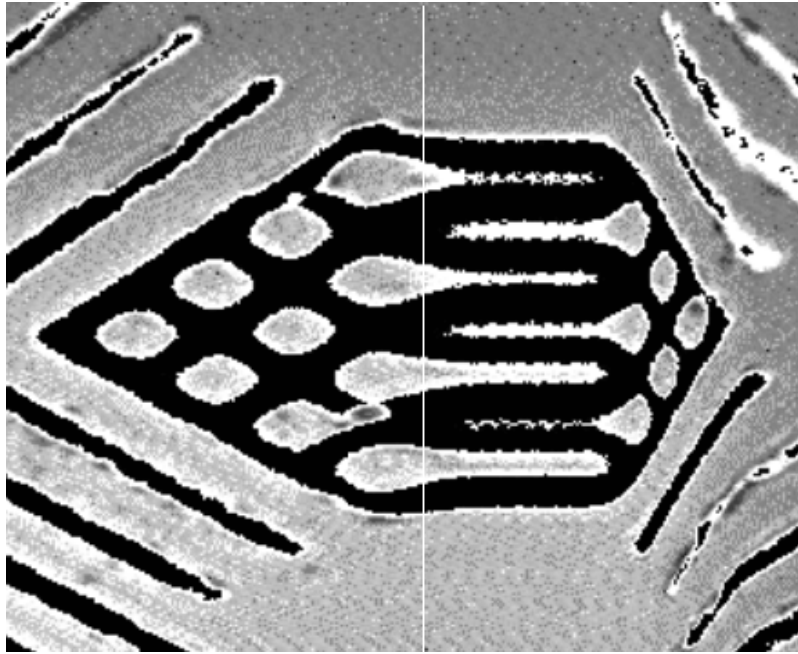
Kuva 30: Kohdistamaton ja kohdistettu kuva mittakaavasta, jossa keskellä MEMS-peili on ollut pysäytettynä 100 pikselin verran.



Kuva 31: Mittakaavasta skannattu kuva, joka on jaettu kahtia skannaussuunnan mukaan.

vaan piirretyn valkoisen pystysuoran viivan, joka kuvaa skannerin pysäytyskohtaa, oikealla puolella havaitaan vielä kuvion muuttuvan, ennen kuin se stabiloituu.

Tämä tulee ottaa huomioon MEMS-peilillä skannattaessa, sillä esimerkiksi nyt takaisinventoaskeleen tulisi olla suurempi, kuin mitä sen tarvitsisi olla, jos peili pysähtyisi heti, kun sen pysäyttää. Viive ei kuitenkaan vaikuttanut niin suurelta, etteikö MEMS-peilin käyttöä skanneripeilinä voisi edelleen kokeilla koejärjestelyssä. Muuten kuvien hyvästä tarkkuudesta pystyttiin päätelemään peilin skannaavan tasaisesti ilman ylimääräistä värähtelyä ja optisen polun olleen muutenkin oikein kohdistettu.

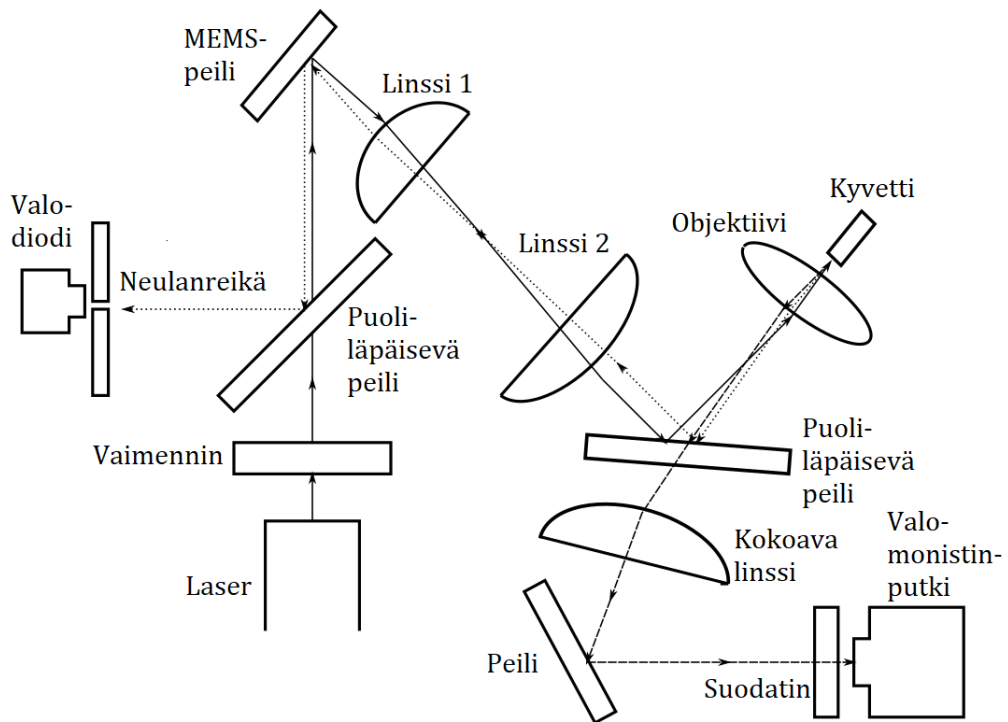


Kuva 32: Mittakaavasta skannattu kuva, joka on jaettu kahtia skannaussuunnan mukaan.

Päätettiin jatkaa sirontasignaalin mittaamista liuoksesta ja lisätä fluoresenssin mittaamiseen tarvittavat komponentit.

### 2.4.3 Sironnan ja fluoresenssin mittaaminen liuksesta

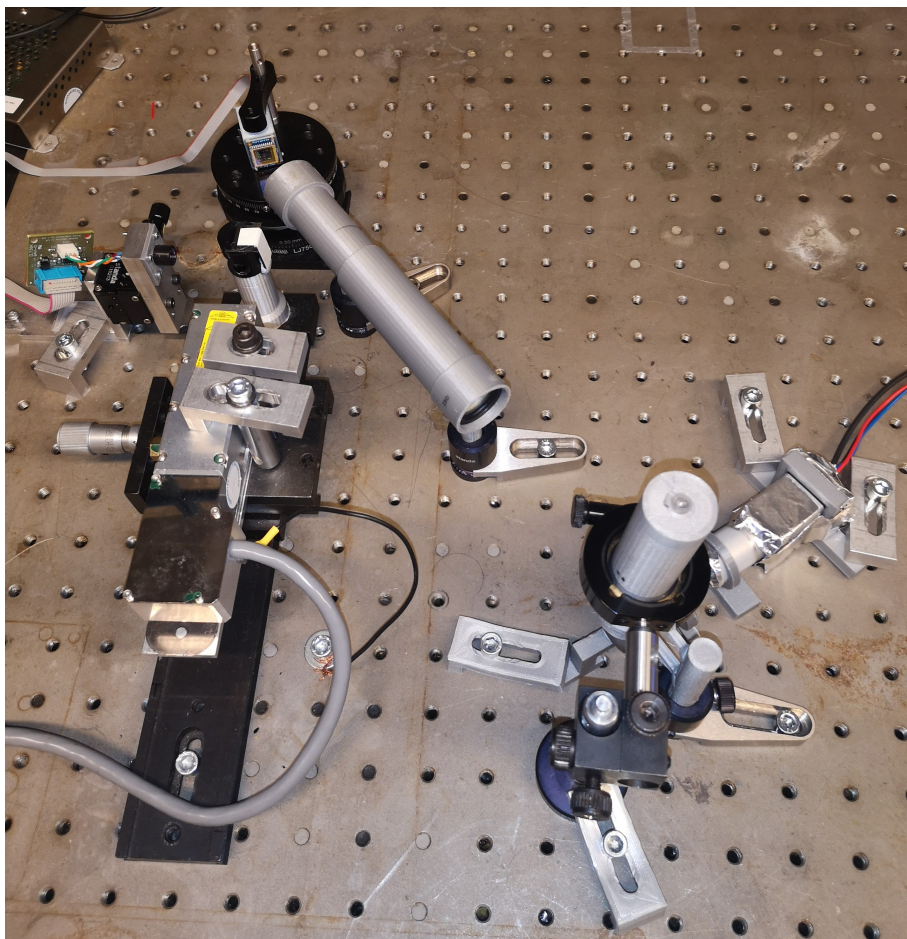
Optiselle pöydälle lisättiin kappaleen 2.4.1 sirontasignaalin havaitsemispolun lisäksi myös komponentit fluoresenssin havaitsemista varten. Koejärjestely oli kuvan 33 mukainen. Lasersäteen tulokulma MEMS-peilille yritettiin pitää koejärjestelyssä pienenä, sillä jos prototyyppi joskus tuotannollistettaisiin, olisi tiiviimmästä järjestelystä hyötyä.



Kuva 33: Koejärjestely sironta- ja fluoresenssisignaalin havaitsemista varten.

Kuvassa 33 objektiivin päällä polttopisteessä olevassa  $200 \mu\text{l}$  kyvetissä oleva liuos sisälsi  $3 \mu\text{m}$  halkaisijaltaan olevia mikropartikkeleita  $2,0 \cdot 10^6$  kpl/ml. Partikkelit olivat kappaleen 2.4.1 liuksesta poiketen suhteellisen korkean fluoresenssin standardeitua partikkeleita, jotta niistä saataisiin varmasti fluoresenssisignaalia. Kuvassa katkoviiva merkitsee fluoresenssissa emittoitunutta valoa ja pisteviiva partikkelista sironnutta laservaloa. Kun lasersäde virittää fluoroforin, emittoituneet fotonit kulkevat objektiivin ja puoliläpäisevän peilin läpi kokoavalle linssille, jonka polttopiste on

valomonistinputken sisääntuloaukolla. Kokoavan linssin jälkeen oleva peili kääntää fotonivuon pöytäpinnan suuntaiseksi ja ennen valomonistinputkea oleva suodatin estää ylimääräisen säteilyn pääsyn valomonistinputkelle. Valomonistinputki on hyvin herkkä hajavalolle, minkä vuoksi mittaukset suoritettiin pimeässä, optisen pöydän päälle oli asetettu mustalla huovalla vuorattu pahvilaatikko ja suodatin oli kiinnitetty valomonistinputken suulle alumiinifolioteipillä. Suodatin päästi lävitseen  $550 \pm 25$  nm aallonpituusalueen säteilyn. Kuvassa 34 on valokuva koejärjestelystä, jossa laser on vasemmalla ja liuosta sisältävä kyvette oikealla puolella.



Kuva 34: Valokuva optisen pöydän koejärjestelystä sironta- ja fluoresenssisignaalin havaitsemista varten.

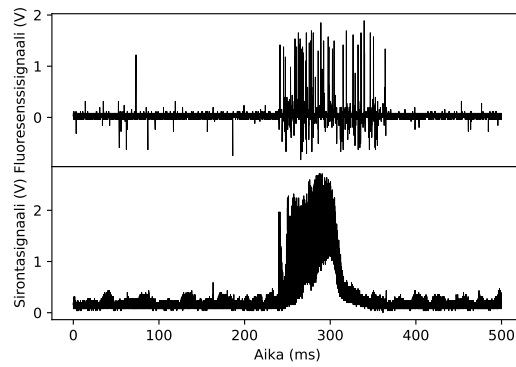
Fluoresenssin mittaamista varten valomonistinputken signaali johdettiin kytkentäkortin kautta lattakaapeliin, josta signaali poimittiin ZedBoardille. Sironta- ja

fluoresenssisignaalia tutkittiin myös oskilloskoopilla. Kuvassa 35 on oskilloskoopista kerätystä datasta piirrettyjä sironta- ja fluoresenssisignaaleja.

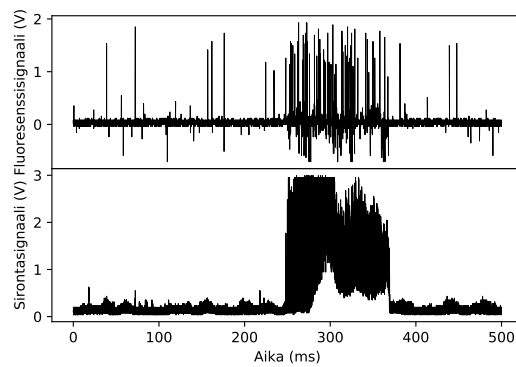
Kuvasta 35 nähdään että fluoresenssisignaali ja sirontasignaali selkeästi korreloivat keskenään, sillä kun sirontasignaali on korkealla havaitaan myös enemmän fluoresenssiipukkeja. Valomonistinputki on kuitenkin niin herkkä, että se havaitsee signaaleja myös silloin, kun partikkeli ei ole fokuksessa, kuten hajapiikeistä huomataan, mutta näin vähäinen hajapiikkien havaitseminen ei ole koejärjestelyn kannalta haitallista. Nämä signaalit ovat todennäköisin peräisin koejärjestelyssä olevien elektronisten piirien kytkentävaloista tai koejärjestelyn ulkopuolelta tulevasta näkyvästä valosta.

Fluoresenssia ja sirontaa mitattiin myös niin, että ohjelmalla kerättiin talteen jokaisen havaitun partikkelin kohdalla se aika, jonka partikkeli oli polttopisteessä sekä valomonistinputken havaitsemien fluoresenssisignaalien määrä jaettuna polttopisteessäoloajalla. Tämä fluoresenssisignaalien määrän suhdetta polttopisteessäoloaikaan kutsutaan intensiteetiksi ja kuvassa 36 on mittauksessa havaittuja intensiteettejä polttopisteessäoloajan funktiona.

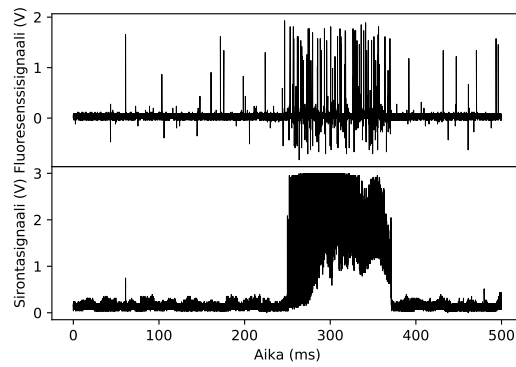
Kuvasta 36 huomataan, että partikkelien ajalla polttopisteessä on ollut suurta hajontaa, sillä lyhimmillään partikkeli on ollut polttopisteessä alle 60 millisekuntia ja pisimmillään taas yli 160 millisekuntia. Intensiteetissä variaatiota on vähemmän, etenkin, jos todetaan nollan lähellä olevat mittaukset epäonnistuneiksi. Voidaan arvioida, että jos intensiteetti on ollut nolla, fluoresenssi ei ole poikennut taustan fluoresenssista. Tällöin on todennäköisesti havaittu jotain muuta kuin fluoresoiva partikkeli, kuten esimerkiksi liuksessa oleva pölyhiukkanen. Datapisteisiin sovitetun suoran kulmakerroin on  $0.026608 \text{ kpl/ms}^2$ , mistä voidaan päätellä intensiteettien ja polttopisteessäoloajan korreloivan positiivisesti. Kulmakertoimeen vaikuttavat kuitenkin myös hyvin suuresti nollan lähellä olevat mittaukset, joten voidaan olettaa että ilman näitä kulmakerroin olisi lähempänä tavoiteltua nollaa. Intensiteetistä ja



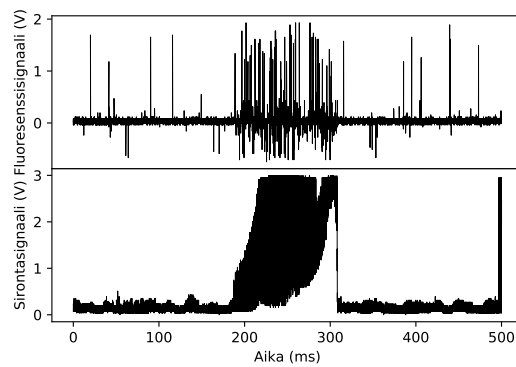
(a)



(b)

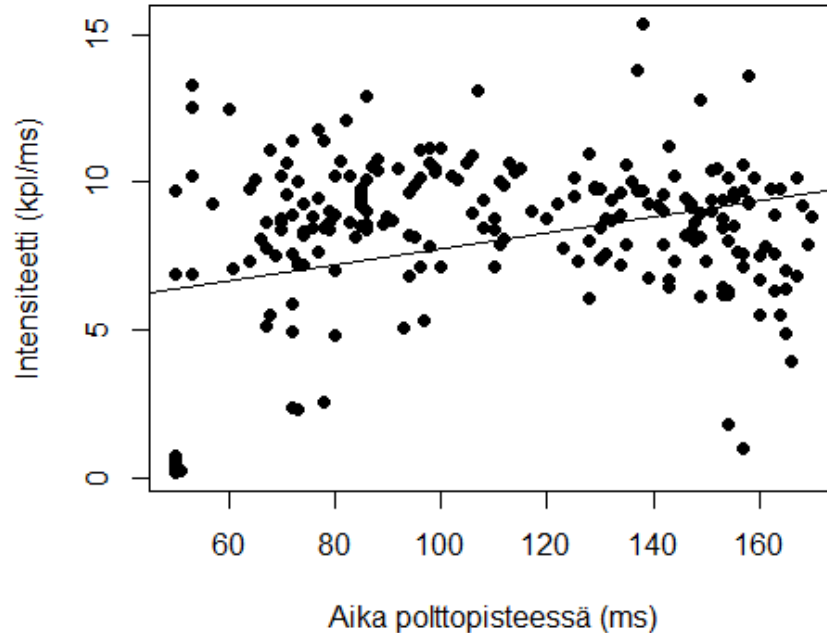


(c)



(d)

Kuva 35: Mitattuja sironta- ja fluoresenssisignaaleja. Ylempi signaali on fluoresenssi- ja alempi sirontasignaali.



Kuva 36: Fluoresenssimittausten intensiteetti polttopisteessäoloajan funktiona ja tähän sovitettu suora.

polttopisteessäoloajasta määritettiin myös taulukossa kolme olevat arvot.

Taulukosta huomataan, että polttopisteessäoloajan keskiarvo ja mediaani eivät eroa kuin vajaat 7 ms toisistaan ja kun näitä arvoja verrattiin aiemmissa laitteissa mitattujen partikkelien aikoihin polttopisteessä, todettiin niiden olevan riittävän samanlaiset, jotta mittaus voisi onnistua. Taulukon variaatiokertoimia verrattiin myös nyt käytössä olevien laitteiden fluoresenssisignaalien variaatiokertoimiin. Taulukkoon neljä laskettiin nyt tuotannossa olevien laitteiden fluoresenssimittausten intensiteettien variaatiokertoimia.

Variaatiokertoimista huomataan, että nyt uudella koejärjestelyllä mitattu variaatiokerroin on jonkin verran suurempi kuin käytössä olevien laitteiden variaatiokerroin. Todennäköisesti suurin vaikutus tähän suurempaan variaatiokertoimeen oli

Taulukko 3: Fluoresenssimittauksessa määritettyjä suureita.

	Intensiteetti (kpl/ms)	Aika polttopisteessä (ms)
Keskiarvo	7,94	106,88
Mediaani	8,67	100,00
Keskihajonta	3,14	38,05
Variaatiokerroin	39,53 %	35,59 %

Taulukko 4: Tuotannossa olevien laitteiden fluoresenssimittausten intensiteettien variaatiokertoimia. Variaatiokertoimet ovat mikropartikkelien välisiä samoin kuin

taulukon 3 variaatiokertoimet.

Laite	Variaatiokerroin
1	20,71 %
2	16,60 %
3	32,44 %
4	22,02 %
5	20,55 %

kuvassa 35 näkyvien intensiteetiltään lähes nollan datapisteillä. Jos datasta poistettiin kaikki alle yhden intensiteetiltään olleet datapisteet, saatiin variaatiokertoimeksi 23,60 %. Tämä variaatiokerroin on jo samaa suuruusluokkaa kuin tuotannossa olevilla laitteilla.

Todettiin siis, että periaatteessa sironta- ja fluoresenssimittaus toimi muuten hyvin koejärjestelyssä, mutta sirontasignaali oli liian matala, ja signaalien muoto oli käytössä olevien laitteiden sirontasignaaleista poikkeava.

Mittauksissa suurin haaste oli optisen polun kohdistaminen, mikä on voinut myös aiheuttaa sirontasignaalin jäämisen matalammaksi kuin käytössä olevissa laitteissa. Etenkin neulanreiän kohdistaminen manuaalisesti osoittautui hankalaksi, sillä usein kohdistuksessa jäätii esimerkiksi lasilevystä mitatun sironnan lokaaliin maksimiin

ja globaali maksimi jäi löytämättä. Kohdistettaessa myös laserin korkea teho voi häiritä maksimin löytämistä, sillä se saattoi polttaa kuopan kohdistettaessa käytettävään lasilevyyn tai mittauksissa käytetyn kyvetin pohjaan, jolloin sironta jäi matalaksi. Neulanreiän paikkaa haarukoitaessa jouduttiin myös käyttämään useampaa vaimenninta, jotka olivat vain silmämääräisesti kohtisuorassa suhteessa saapuvaan valonsäteeseen. Jos nämä vaimentimet ovat olleet hieman vinossa, ovat ne aiheuttaneet laservalon sivuttaissiirtymän neulanreiän kohdalla, ja neulanreiän paikka siis on ollut ilman vaimentimia liuosta skannattaessa väärä.

Myös mittauksissa käytetty yksittäiskyvetti vaikutti sirontasignaaliin, sillä mittauksissa huomattiin, että kyvetin pohja toimii polarisaattorina, minkä vuoksi kyvetin asento tulevaan säteeseen nähden piti määrittää tarkasti. Kyvetin etäisyyttä objektiivista ei myös pystytty säätämään, jolloin sirontasignaali on voinut vaimentua jo siksi, että polttopiste on ollut liian syvällä kyvetin sisällä. Tällöin liian suuri määrä liuoksen läpi palaavia fotoneja absorboituu liuokseen, jolloin niitä havaitaan vähemmän kuin jos polttopiste olisi ollut lähempänä kyvetin pohjaa.

Heikkoon sirontasignaaliin saattoi myös vaikuttaa mahdollisesti vanha liuos, sillä mikropartikkelit voivat muodostaa agglomeraatteja pitkän ajan kuluessa sekä laskeutua kyvetin pohjaan painovoiman vaikutuksesta. Liuoksen vaihtamisesta säännöllisin väliajoin pidettiin kyllä huolta, mutta oli epäselvää miten nopeasti agglomeraatteja muodostuu.

On myös mahdollista, että neulanreiän kohdalle saapuva sironnut valo on ollut halkaisijaltaan leveämpi kuin neulanreikä, jolloin se ei pääse koko intensiteetiltään valodiodille. Neulanreiän suurentaminen ei kuitenkaan ratkaisisi ongelmaa, sillä silloin taustakohina nousisi liian suureksi.

Tulevaisuudessa ongelmia saataisiin ratkaistua sillä, että koejärjestelyyn rakennettaisiin kuoppalevyä kannatteleva systeemi, jossa kuoppalevyn korkeutta objektiivista pystyttäisiin säätämään tarkasti. Tämä eliminoisi sekä yksittäiskyvetin pola-

risaattoritaipumukset että näin voitaisiin saada tarkemmin säädettyä polttopisteen etäisyys kuopan pohjasta kuopan sisällä. Paremmiin valoa estäviä koppia olisi myös tulevaisuuden mittauksia ajatellen tarpeellinen, sillä nyt käytetty mustalla huoval- la päällystetty pahvilaatikko ei toiminut aukottomasti. Jos huone, missä mittauk- set suoritettiin, ei ollut täysin pimeä, pääsi esimerkiksi kopan alareunan ja optisen pöydän välistä valoa koejärjestelyyn. Vaimentimien käyttöä tulisi polkua kohdistet- taessa välttää tai sitten käyttää niin ohuita vaimentimia, ettei sivuttaissiirtymä olisi suuri.

Nyt yksi ongelma oli myös se, että mittauksissa virhettä aiheutti myös raken- nuksen liikkeet kun esimerkiksi ulkona tuuli voimakkaasti. Tähän auttaisi yksittäis- kyvetin tai kuoppalevyn kiinnittäminen tukevammin objektiivin päälle ja se, että niiden säätömekanismi olisi tarkempi. Nyt rakennuksen liikkuessa objektiivin polt- topiste osui välillä esimerkiksi yksittäiskyvetin pohjaan mitattaessa tai kohdist- misessa käytetty lasilevy ei pysynyt stabiilisti paikoillaan, jolloin kohdistus ei ollut tarkka.

### 3 Yhteenveto

Tässä työssä tutkittiin MEMS-peilin sopivuutta hengitystieinfektioiden diagnostiikkalaitteistoon. Mittaukset onnistuivat työssä muutoin, mutta sirontasignaalin vahvuudessa ei päästy käytössä olevien laitteiden tasolle. Sirontasignaalit sekä jäivät toistuvasti alle toivotun 100 ms:n että olivat intensiteetiltään heikompia kuin olisi toivottu. Tähän suurimpana syynä todennäköisesti oli neulanreiän paikan epätarkkuus. Optisen polun kohdistaminen ja etenkin neulanreiän oikean paikan löytäminen osoittautuivat kokeellisessa osuudessa huomattavasti ajateltua haasteellisemmiksi.

Polun kohdistamista olisi mahdollista helpottaa kehittämällä tähän apuvälineitä. Kohdistettaessa tulisi myös käyttää mahdollisimman vähän vaimentimia tai vaihtoehtoisesti varmistua siitä, että vaimentimet eivät aiheuta sivuttaissiirtymää lasersäteeseen neulanreiän kohdalla. Komponentit voisi myös kiinnittää tukevammin pöytään kiinni ja varmistaa optisen pöydän pysyvän paikallaan.

MEMS-peili saatiin toimimaan hyvin laitteistossa ja kokeellisessa osuudessa ei havaittu mitään, mikä estäisi pietsoskannereiden korvaamisen MEMS-peilillä. MEMS-peilin pienen halkaisijan vuoksi lasersäteen tulee kuitenkin olla vielä melko kapea sen osuessa peilille, mikä vaatii suurentavaa optiikkaa myöhemmin optisella polulla. Jos suurentavat linssit saadaan konfiguroitua oikein ja neulanreikä oikeaan kohtaa, pitäisi MEMS-peilin toimia yhtä hyvin laitteistossa, kuin pietsosähköiseen ilmiöön perustuvien skanneripeilien. Tarkempaa tutkimusta vaativat MEMS-peilin muut ominaisuudet, kuten miten se fyysisesti on mahdollista implementoida laitteistoon tai mikä on sen keskimääräinen elinikä ja voiko sitä käyttää pitkiä aikoja yhtäjaksoisesti ilman pelkoa esimerkiksi kuumentumisesta.

## Viitteet

- [1] A. S. Monto, *The American Journal of Medicine* **112**, 4 (2002).
- [2] M. J. Loeffelholz, *Clinics in Laboratory Medicine* **34**, xi (2014).
- [3] R. A. Britton ja J. Versalovic, *Interdisciplinary perspectives on infectious diseases* **2008**, (2008).
- [4] A. Machowska ja C. Stålsby-Lundborg, *International Journal of Environmental Research and Public Health* **16**, (2018).
- [5] [arcdia.com/fi/maripoc/](http://arcdia.com/fi/maripoc/), saatavilla [www-muodossa](http://www-muodossa.com) <[arcdia.com /fi/maripoc/](http://arcdia.com/fi/maripoc/)> (viitattu 16.02.2020) .
- [6] P. T. So, C. Y. Dong, B. R. Masters ja K. M. Berland, *Annual reviews of biomedical engineering* **2**, 399–429 (2000).
- [7] C. Xu ja W. W. Webb, 471 (2002).
- [8] B. Valeur ja M. N. Berberan-Santos, *Journal of Chemical Education* **88**, 731 (2011).
- [9] J. R. Albani, *Structure and dynamics of macromolecules : absorption and fluorescence studies*, 1st ed. ed. (ElsevierAmsterdam, 2004).
- [10] J. F. Da Silveira Petrucci, M. G. Liebetanz, A. A. Cardoso ja P. C. Hauser, *Journal of Chromatography A* **1512**, 143 (2017).
- [11] C. Xu *et al.*, *Proceedings of the National Academy of Sciences of the United States of America* **93**, 10763 (1996).
- [12] W. R. Zipfel, R. M. Williams ja W. W. Webb, *Nature Biotechnology* **21**, 1369 (2003).
- [13] J. B. Pawley, *Handbook of Biological Confocal Microscopy*, 3 ed. (Springer, 2006), pp. 5, 16, 89, 208–209, 381.
- [14] P. Hänninen *et al.*, *Nature Biotechnology* **18**, 548 (2000).
- [15] *Encyclopedia Britannica* (viitattu 28.03.2020). Saatavilla [www-muodossa](http://www-muodossa.com) <[www.https://www.britannica.com/science/antibody](https://www.britannica.com/science/antibody)> 143 (2020).
- [16] J. T. Soini, Väitöskirja, Turun yliopisto, Lääketieteellinen tiedekunta; Biolääketieteen laitos, 2002.
- [17] J. O. Koskinen, Väitöskirja, Turun yliopisto, Lääketieteellinen tiedekunta; Biolääketieteen laitos, 2008.
- [18] L. Fu *et al.*, *Optics Express* **14**, 1027 (2006).
- [19] J. Bowers *et al.*, *United States Patent Application Publication* (2002).

- [20] Hamamatsu.com, (viitattu 28.03.2020). Saatavilla [www.muodossa.com/hamamatsu.com/eu/en/product/optical-components/mems-mirror/Structure](http://www.muodossa.com/hamamatsu.com/eu/en/product/optical-components/mems-mirror/Structure).
- [21] A. Kasturi, V. Milanovic, B. H. Atwood ja J. Yang, in *Laser Radar Technology and Applications XXI*, International Society for Optics and Photonics (SPIE Proceedings, 2016), Vol. 9832, p. 98320M.
- [22] F. A. Chollet, G. M. Hegde, A. K. Asundi ja A. Q. Liu, in *Advanced Photonic Sensors and Applications II*, International Society for Optics and Photonics (SPIE Proceedings, 2001), Vol. 4596, pp. 272–279.
- [23] P. Himiner ja D. Dickensheets, in *IEEE/LEOS International Conference on Optical MEMs* (IEEE, 2002), pp. 193–194.
- [24] H. Ra *et al.*, *Journal of Microelectromechanical systems* **16**, 969 (2007).
- [25] H. Miyajima *et al.*, *Journal of microelectromechanical systems* **12**, 243 (2003).
- [26] mirrorcletech.com, saatavilla [www.muodossa.com/mirrorcletech.com/wp/products/mems-mirrors/dual/integrated/](http://www.muodossa.com/mirrorcletech.com/wp/products/mems-mirrors/dual/integrated/) (viitattu 16.01.2020) .
- [27] N. K. Haddad, *Microcontroller system design using PIC18F processors* (IGI Global, 2017), Chap. 13, pp. 240–241.
- [28] Mirrorcle Technologies Inc, MEMS Driver 5x User Guide (2019).
- [29] Mirrorcle Technologies Inc, Device parameters summary (2019).
- [30] Analog Devices, Data Sheet AD5624R/AD5644R/AD5664R (2013).
- [31] edmundoptics.com, saatavilla [www.muodossa.com/edmundoptics.com/knowledge-center/application-notes/optics/](http://www.muodossa.com/edmundoptics.com/knowledge-center/application-notes/optics/) (viitattu 15.01.2020) .
- [32] pololu.com, saatavilla [www.muodossa.com/https://www.pololu.com/product/3113](http://www.muodossa.com/https://www.pololu.com/product/3113) (viitattu 13.03.2020) .
- [33] F. Nordsiek, saatavilla [www.muodossa.com/github.com/frejanordsiek](http://www.muodossa.com/github.com/frejanordsiek), (viitattu 30.01.2020) (2013).
- [34] A. Kovács, in *Millimeter and Submillimeter Detectors and Instrumentation for Astronomy IV*, International Society for Optics and Photonics (SPIE Proceedings, 2008), Vol. 7020, p. 702007.

# Liitteet

## A Koodi MEMS-peilin ohjaamiseen Arduino IDE:llä

```

#include <AD56X4.h>
#include "Arduino.h"
#include "math.h"
#include <SPI.h>

/*Asetetaan nasta 10 AD-muuntimen select-SPI-nastaksi.*/
int AD56X4_SS_pin = 10;
/*Asetetaan nasta 8 nastaksi, joka kontrolloi
korkeajännitteen aktivoimista.*/
int HV_out_EN = 8;
/*Asetetaan nasta 9 kellosignaalin ulostulonastaksi.*/
const byte CLOCKOUT = 9;

/*Määritellään parametrit, joiden mukaan jokaiseen eri
kanavaan ajetaan sinijännitettä ja peili saadaan
tekemään ympyräliikettä.*/
float frequencies [] = {20, 20, 20, 20};
float phases [] = {180, 0, 180, 0};
float offsets [] = {32768, 32768, 32768, 32768};
float amplitudes [] = {20000, 20000, 20000, 20000};

void setup ()
{
    /*Alustetaan ajastin kellonastaan CLOCKOUT.*/
    pinMode (CLOCKOUT, OUTPUT);
    TCCR1A = bit (COM1A0);
    TCCR1B = bit (WGM12) | bit (CS10);
    OCR1A = 128;

    /*Alustetaan SPI-väylä. Arduino on master- ja peilin
ajuri (käytännössä AD-muunnin) orjalaite.*/

    /*Asetetaan AD56X4_SS_pin ulostulonastaksi.*/
    pinMode (AD56X4_SS_pin, OUTPUT);
    /*Asetetaan SPI:n kellotaajuus.*/
    SPI.setClockDivider (SPI_CLOCK_DIV2);
    SPI.begin ();

```

```

/*Resetoidaan AD56X4_SS_pin ja käytetään sisäistä
referenssijännitettä.*/
AD56X4.reset(AD56X4_SS_pin, true);
AD56X4.useInternalReference (AD56X4_SS_pin, true);

/*Asetetaan korkeajännitteen kontrollointinasta
ulostulonastaksi.*/
pinMode(HV_out_EN, OUTPUT);
/*Asetetaan korkeajännitepuoli toimivaksi kirjoittamalla
korkeajännitepuolen kontrollointinastaan arvo HIGH.*/
digitalWrite(HV_out_EN, HIGH);
}

void loop ()
{
/*Otetaan tämänhetkinen aika.*/
float t = 1e - 6 * float(micros());
/*Alustetaan taulukko peilin neljälle eri kanavalle.*/
byte channels [] = {AD56X4_CHANNEL_A, AD56X4_CHANNEL_B,
AD56X4_CHANNEL_C, AD56X4_CHANNEL_D};

/*Lasketaan siniaallon arvot jokaiselle kanavalle
erikseen ja asetetaan nämä jokaisen kanavan
sisääntulorekistereihin.*/
for (int i = 0; i < 4; i++)
{
/*Lasketaan siniaallon arvo ajanhetkellä t.*/
float y = offsets[i] + amplitudes[i]
* sin(2.0*M_PI*(t*frequencies[i]
+ phases[i]/360.0));

/*Muunnetaan siniaallon arvo word-tyypiksi
(unsigned int), jotta se pysyy alueella [0, 0xFFFF].*/
word output = word(y);

/*Tarkistetaan vielä ettei y:n arvo mene rajojen yli.*/
if (y > 0xFFFF)
output = 0xFFFF;
else if (y < 0)
output = 0;

/*Kirjoitetaan arvo kanavaan.*/
AD56X4.setChannel(AD56X4_SS_pin, AD56X4_SETMODE_INPUT,

```

```

                                channels[i], output);
    }

/*Päivitetään kaikkien DAC-rekisterien arvot samanaikaisesti,
jotta nyt ne ovat uudet siniaaltojen arvot.*/
AD56X4.updateChannel(AD56X4_SS_pin, AD56X4_CHANNEL_ALL);
}

/*
  Arduino library handling SPI communication with 56x4:
  Copyright (c) 2013, Freja Nordsiek
  All rights reserved.

  Redistribution and use in source and binary forms, with or
  without modification, are permitted provided that the
  following conditions are met:

  1. Redistributions of source code must retain the above
  copyright notice, this list of conditions and the following
  disclaimer.

  2. Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following
  disclaimer in the documentation and/or other materials
  provided with the distribution.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
  CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
  INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
  MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
  DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR
  CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
  SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
  NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
  LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
  HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
  CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
  OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
  SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

```

## B Kytkimen lisääminen MEMS-peilin ohjaamiseen

```

/*Koska tämä koodi on tehty liitteen A koodin päälle, vain

```

```

ne rivit jotka ovat muuttuneet on kommentoitu.*/
#include <AD56X4.h>
#include "Arduino.h"
#include "math.h"
#include <SPI.h>

int AD56X4_SS_pin = 10;  the AD56X4.
int HV_out_EN = 8;
const byte CLOCKOUT = 9;

/*Kytkimen sisääntulonastaksi asetetaan nasta 11.*/
int input_pin = 11;
/*Tähän muuttujaan tallennetaan arvo nastasta 11 (onko
kytkin kiinni vai auki)*/
int value = 0;

float frequencies [] = {20, 20, 20, 20};
float phases [] = {0, 180, 0, 180};
float offsets [] = {32768, 32768, 32768, 32768};
float amplitudes [] = {26000, 26000, 26000, 26000};

word output0 = 0xFFFF;
word output1 = 0xFFFF;
word output2 = 0xFFFF;
word output3 = 0xFFFF;

void setup ()
{
/*Asetetaan nasta 11 sisääntulonastaksi*/
pinMode (input_pin, INPUT);

pinMode (CLOCKOUT, OUTPUT);
TCCR1A = bit (COM1A0);
TCCR1B = bit (WGM12) | bit (CS10);
OCR1A = 128;

pinMode (AD56X4_SS_pin, OUTPUT);
SPI.setClockDivider (SPI_CLOCK_DIV2);
SPI.begin ();

AD56X4.reset (AD56X4_SS_pin, true);
AD56X4.useInternalReference (AD56X4_SS_pin, true);

```

```

pinMode(HV_out_EN,OUTPUT);
digitalWrite(HV_out_EN, LOW);
}

void loop ()
{
    float t = 1e-6 * float(micros ());

    /*Muuttujaan value tallennetaan arvo input_pin-nastasta
    (nasta 11), joka riippuu kytkimen asennosta. */
    value = digitalRead(input_pin);

    /*Jos kytkin on auki, peili ajetaan ensin 0-positioon ja
    sitten sammutetaan korkeajännitepuoli.*/
    if (value == 0) {
        mirrorToZero(output0 , output1 , output2 , output3);
        digitalWrite(HV_out_EN, LOW);
    }else {
        digitalWrite(HV_out_EN, HIGH);

        byte channels [] = {AD56X4_CHANNEL_A, AD56X4_CHANNEL_B,
                            AD56X4_CHANNEL_C, AD56X4_CHANNEL_D};

        for (int i = 0; i < 4; i++){
            float y = offsets [i] + amplitudes [i]
                    * sin (2.0*M_PI*(t*frequencies [i]
                    + phases [i]/360.0));

            word output = word(y);

            if (y > 0xFFFF){
                output = 0xFFFF;
            }
            else if (y < 0){
                output = 0;
            }
            }

        AD56X4.setChannel(AD56X4_SS_pin, AD56X4_SETMODE_INPUT,
                        channels [i] , output);

        /*Tallennetaan äsken kirjoitetut outputit muistiin.*/
        if (i==0){
            output0 = output;
        }
        else if (i==1){

```

```

        output1 = output;
    }
    else if (i==2){
        output2 = output;
    }
    else if (i==3){
        output3 = output;
    }
}

AD56X4.updateChannel(AD56X4_SS_pin, AD56X4_CHANNEL_ALL);
}
}

/*Määritellään funktio , joka ajaa peilin 0-positioon.
Funktio argumentteina ovat peilin neljään eri kanavaan
ajettavat signaalit.*/
void mirrorToZero(word outputA, word outputB, word outputC,
word outputD)
{
    /*Tallennetaan nämä muuttujiin opA-opD.*/
    word opA = outputA;
    word opB = outputB;
    word opC = outputC;
    word opD = outputD;

    /*Tallennetaan nämä myös muuttujiin newoutputA-newoutputD.*/
    word newoutputA = outputA;
    word newoutputB = outputB;
    word newoutputC = outputC;
    word newoutputD = outputD;

    /*Seuraavaksi iteroidaan tätä silmukkaa kunnes peili
    saavuttaa 0-positioon.*/
    while (opA!=32768 && opB!=32768 && opC!=32768 && opD!=32768){

        /*Askelen pituus.*/
        word stp = 1000;

        /*AD-muuntimeen kirjoitettavien arvojen ylä- ja alaraja
        sekä peilin 0-positioimiseksi kirjoitettava arvo center. */
        word upperlimit = 33768;
        word lowerlimit = 31768;
        word center = 32768;

```

```

/*Muutetaan seuraavaksi AD-muuntimen A-kanavaan
kirjoitettavaa arvoa askelen verran. */
if (opA > upperlimit){
    newoutputA = opA - stp;
}
else if (opA < lowerlimit){
    newoutputA = opA + stp;
}
else {
    newoutputA = center;
}

/*Asetetaan arvo AD-muuntimeen.*/
AD56X4.setChannel(AD56X4_SS_pin, AD56X4_SETMODE_INPUT,
                  AD56X4_CHANNEL_A, newoutputA);
    /*Tehdään sama B-kanavan, ja sen jälkeen C- ja D-kanavan
arvoille.*/
if (opB > upperlimit){
    newoutputB = opB - stp;
}
else if (opB < lowerlimit){
    newoutputB = opB + stp;
}
else {
    newoutputB = center;
}
AD56X4.setChannel(AD56X4_SS_pin, AD56X4_SETMODE_INPUT,
                  AD56X4_CHANNEL_B, newoutputB);

if (opC > upperlimit){
    newoutputC = opC - stp;
}
else if (opC < lowerlimit){
    newoutputA = opC + stp;
}
else {
    newoutputC = center;
}
AD56X4.setChannel(AD56X4_SS_pin, AD56X4_SETMODE_INPUT,
                  AD56X4_CHANNEL_C, newoutputC);

if (opD > upperlimit){
    newoutputD = opD - stp;
}

```

```

else if (opD < lowerlimit){
    newoutputD = opD + stp;
}
else {
    newoutputD = center;
}
AD56X4.setChannel(AD56X4_SS_pin, AD56X4_SETMODE_INPUT,
                  AD56X4_CHANNEL_D, newoutputD);

/*Päivitetään uudet arvot kanaviin.*/
AD56X4.updateChannel(AD56X4_SS_pin, AD56X4_CHANNEL_ALL);

/*Sitten vaihdetaan uudet arvot vanhojen arvojen tilalle ja
jatketaan iterointia.*/
opA = newoutputA;
opB = newoutputB;
opC = newoutputC;
opD = newoutputD;

}
}

```

```

/*
  Arduino library handling SPI communication with 56x4:
  Copyright (c) 2013, Freja Nordsiek
  All rights reserved.

```

*Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:*

- 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.*
- 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.*

*THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR*

*CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.*

*\*/*

## C MEMS-peilin ohjaaminen Xilinx SDK:n avulla

```
/*measurement.c
Ohjelma Mirrorclen MEMS-peilin ajamista varten.
ArcDia / Jori Soukka
2019-05-24
*/

#include <stdio.h>
#include <sleep.h>
#include <xil_printf.h>
#include "platform.h"
#include "xgpio.h"
#include "xscugic.h"
#include "xparameters.h"
#include "spi.h"
#include "tpx_measurement.h"

/*Määritellään tarvittavia nastoja.*/
#define SWS_DEVICE_ID XPAR_AXI_GPIO_0_DEVICE_ID
#define LEDS_DEVICE_ID XPAR_AXI_GPIO_1_DEVICE_ID
#define LEDADDR XPAR_AXI4_LITE_TST_IP_0_S00_AXI_BASEADDR

/*Otetaan käyttöön ulkoinen TMirror-luokan muuttuja mirror.*/
extern TMirror mirror;
/*Alustetaan datavektori, johon voi tallentaa u32-tyyppin arvoja.*/
u32 data[1000];

XGpio swInst, ledInst;

/*Alustetaan AD-muunnin käyttäen kanavaa 0 ja käynnistetään se.*/
u16 testADC() {
```

```

initADC (0);
startADC ();
return readWordFromADC (0);
}

```

```

int main () {

```

```

    int status;
    init_platform ();

```

```

    /*Alustetaan kaikki ledit.*/
    status = XGpio_Initialize(&ledInst , LEDS_DEVICE_ID);
    if (status != XST_SUCCESS)
        return XST_FAILURE;
    status = XGpio_Initialize(&swInst , SWS_DEVICE_ID);
    if (status != XST_SUCCESS)
        return XST_FAILURE;

```

```

    /*Alustetaan ulostulot.*/
    XGpio_SetDataDirection(&ledInst , 1, 0x00);
    /*Alustetaan sisääntulot.*/
    XGpio_SetDataDirection(&swInst , 1, 0xFF);

```

```

    /*Alustetaan SPI0-kanava sekä DA-muunnin.*/
    initSPI0 ();
    initDAC ();
    /*Asetetaan ajurin tilaksi 0 ja alustetaan AD-muunnin.*/
    setDriverState (0);
    initADC (0);

```

```

    /*Nyt vielä ajurin tila on 0 ja measurementDone myös, sillä
    mittausta ei ole vielä suoritettu.*/
    int driverState = 0;
    int measurementDone = 0;

```

```

    /*Aloitetaan mittaus.*/
    while (1) {
        usleep (200);
        /*Luetaan vaihtotilat.*/
        int swval = XGpio_DiscreteRead(&swInst , 1);
        /*Sytytetään ledit.*/
        XGpio_DiscreteWrite(&ledInst , 1, swval);

```

```

        /*Tarkistetaan mikä on ajurin tila.*/

```

```

if ((swval & 0x1) != driverState) {

    /*Jos driverState muuttunut, asetetaan muutettu arvo ajurin
       tilaksi.*/
    driverState = swval & 0x1;
    setDriverState(driverState);
        continue;
}
/*Jos driverState on 0 ja mittaus on loppu
(measurementDone=1), lopetetaan iteroiminen.*/
else if (driverState == 0) {
    if (measurementDone) {
        break;
    } else continue;
}

/*Jos driverState on 1, määritellään peilin X- ja Y-
suuntaisten askelten pituudet ja aloitetaan mittaus
kutsumalla measureSample-funktiota. Funktio on määritelty
c-tiedostossa tpx_measurement.c.*/
/*Askelten pituudet on valittu niin, että peilin nopeus ja
kolmioaaltojen taajuudet olisivat mahdollisimman samanlaiset
kuin aiemmissa pietsosähköisissä skanneripeileissä.*/
if (driverState) {

    const float STEP_Y = (0.89 / (21000.0 * 0.05));
    const float STEP_X = (STEP_Y * 137.0 / 67.0);

    print("Starting_TPX_measurement_simulation\n\r");

    /*Aloitetaan mittaus.*/
    measureSample(26, 500, 120, 50, STEP_X, STEP_Y, 20,
400, 1800, 500);

    print("Ending_TPX_measurement_simulation\n\r");

    measurementDone = 1;

}
}
cleanup_platform();
print("All_done\n\r");
return 0;
}

```

```

/*
  spi.c
  C-tiedosto, jossa määritellään SPI-väylän käyttämiseen
  ja mittauksen suorittamiseen tarvittavia funktioita.
  ArcDia / Jori Soukka
  2019-12-20
  */

#include "../src/spi.h"

#include "xparameters.h"
#include "xil_io.h"
#include "sleep.h"
#include "xspips.h"

static XSpiPs spiInstance_EMIO;
XSpiPs_Config *spiConfig_EMIO;

/*Määritellään valintanasta.*/
u8 ctrlVal = ADC_CONVST_N | ADC_RESET_N;

volatile TMirror mirror;

/*Alustetaan SPI-väylä.*/

int initSPI0() {
  spiConfig_EMIO = XSpiPs_LookupConfig((u16) SPI_EMIO);
  int status = XSpiPs_CfgInitialize(&spiInstance_EMIO,
  spiConfig_EMIO, spiConfig_EMIO->BaseAddress);

  if (status != XST_SUCCESS) {
    print("SPI0_init_failed!\r\n");
    return XST_FAILURE;
  }

  XSpiPs_SetOptions(&spiInstance_EMIO, XSPIPS_MASTER_OPTION
  | XSPIPS_FORCE_SSELECT_OPTION
  | XSPIPS_CLK_ACTIVE_LOW_OPTION);

  XSpiPs_SetClkPrescaler(&spiInstance_EMIO,
  XSPIPS_CLK_PRESCALE_8);

  return status;
}

```

```

/* Alustetaan DA-muunnin.*/
int initDAC() {
    setDriverState(0);
    writeWordToDAC(MIR_RESET, 1);
    writeWordToDAC(MIR_EN_IREF, 1);
    writeWordToDAC(MIR_EN_DAC_CH, 0xf);
    writeWordToDAC(MIR_EN_SW_LDAC, 0);

    /* Asetetaan peili nollassuunnitukseen.*/
    mirror.fromX = 0.0;
    mirror.fromY = 0.0;
    mirror.toX = 0.0;
    mirror.toY = 0.0;
    mirror.dirX = 1;
    mirror.dirY = 1;
    return 0;
}

/* Määritellään funktio, jossa asetetaan ajurin tila.*/
void setDriverState(int enabled) {
    moveMirrorTo(0.0, 0.0);
    ctrlVal = (enabled > 0)
                ? (ctrlVal | MIR_EN_DRIVER_HV)
                : (ctrlVal & (~MIR_EN_DRIVER_HV));
    Xil_Out32(XPAR_AXI4_LITE_TST_IP_0_S00_AXI_BASEADDR,
              ctrlVal);
    usleep(200);
}

/* Määritellään funktio, jossa asetetaan ajurin eri kanaviin
arvot.*/
int setDriverOutputs(float x, float y) {
    if (x * x + y * y > 1.0) return 1;

    u16 x1 = MIR_BIAS + MIR_DIFF_MAX * x;
    u16 x2 = MIR_BIAS - MIR_DIFF_MAX * x;
    u16 y1 = MIR_BIAS + MIR_DIFF_MAX * y;
    u16 y2 = MIR_BIAS - MIR_DIFF_MAX * y;

    /* Kanava A eli positiivinen X-kanava.*/
    writeWordToDAC(MIR_WRITE_REG | 0x0, x1);

    /* Kanava B eli negatiivinen X-kanava.*/

```

```

writeWordToDAC(MIR_WRITE_REG | 0x1, x2);

/*Kanava C eli negatiivinen Y-kanava.*/
writeWordToDAC(MIR_WRITE_REG | 0x2, y2);

/*Kanava B eli positiivinen Y-kanava.*/
writeWordToDAC(MIR_WRITE_REG_UPDATE_ALL | 0x3, y1);
return 0;
}

/*Määritellään funktio, jonka avulla DA-muuntimeen voidaan
kirjoittaa arvo.*/
void writeWordToDAC(u8 cmd, u16 data) {
    u8 writeBuffer[3] = {
        cmd,
        data >> 8,
        data & 0xff
    };
    XSpiPs_SetSlaveSelect(&spiInstance_EMIO, SPI0_SLAVE_DAC);
    XSpiPs_PolledTransfer(&spiInstance_EMIO, writeBuffer, NULL, 3);
}

/*Määritellään funktio, jossa liikutetaan peiliä seuraavaan
pisteeseen.*/
int moveMirrorTo(float x, float y) {
    mirror.toX = x;
    mirror.toY = y;

    mirror.dirX = (mirror.fromX < mirror.toX) ? 1 : -1;
    mirror.dirY = (mirror.fromY < mirror.toY) ? 1 : -1;

    const float MAX_STEP = 2.0/30.0;
    int status = 0;
    while (1) {
        float dx = x - mirror.fromX;
        float dy = y - mirror.fromY;

        float nextX = (dx > MAX_STEP)?(mirror.fromX + MAX_STEP) :
            (dx < -MAX_STEP)?(mirror.fromX - MAX_STEP) : x;
        float nextY = (dy > MAX_STEP)?(mirror.fromY + MAX_STEP) :
            (dy < -MAX_STEP)?(mirror.fromY - MAX_STEP) : y;
        status = setDriverOutputs(nextX, nextY);
        mirror.fromX = nextX;
        mirror.fromY = nextY;
        if ((nextX != x) || (nextY != y)) {

```

```

    usleep(40);
} else break;
}
return status;
}

```

*/\*Määritellään funktio, jossa lasketaan peilin seuraava positio.\*/*

```

int updateScanner(float stepSizeX, float stepSizeY) {
    float x = mirror.fromX + mirror.dirX * stepSizeX;
    float y = mirror.fromY + mirror.dirY * stepSizeY;
    if (x * x > 0.2) {
        mirror.dirX *= -1;
        x += 2 * mirror.dirX * stepSizeX;
    }
    if (y * y > 0.2) {
        mirror.dirY *= -1;
        y += 2 * mirror.dirY * stepSizeY;
    }
    return moveMirrorTo(x, y);
}

```

*/\*Määritellään funktio, jonka avulla peili tekee takaisinventoaskeleen.\*/*

```

int stepBack(float stepSizeX, float stepSizeY) {
    mirror.dirX *= -1;
    mirror.dirY *= -1;
    int status = updateScanner(stepSizeX, stepSizeY);
    mirror.dirX *= -1;
    mirror.dirY *= -1;
    return status;
}

```

*/\*Määritellään AD-muuntimen alustusfunktio.\*/*

```

int initADC(int channel) {
    ctrlVal &= ~ADC_RESET_N;
    Xil_Out32(XPAR_AXI4_LITE_TST_IP_0_S00_AXI_BASEADDR, ctrlVal);
    usleep(1);
    ctrlVal |= ADC_RESET_N;
    Xil_Out32(XPAR_AXI4_LITE_TST_IP_0_S00_AXI_BASEADDR, ctrlVal);
    usleep(1);
    u8 channel1 = channel & 0x3;
    writeWordToADC(0x01, 1 << channel1);
/*Sisäinen referenssijännite.*/
    writeWordToADC(0x02, 1);
}

```

```

    return 0;
}

/*Määritellään funktio, jossa AD-muuntimeen kirjoitetaan
arvo.*/
u16 writeWordToADC(u8 addr, u16 data) {
    /*5 bittiä.*/
    u8 addr1 = addr & 0x1f;
    /*10 bittiä.*/
    u16 data1 = data & 0x3ff;
    u8 writeBuffer[2] = {
        (addr1 << 3) + (0x4 + (data1 >> 8)),
        data1 & 0xff
    };
    u8 readBuffer[2] = { 0, 0 };
    XSpiPs_SetSlaveSelect(&spiInstance_EMIO, SPI0_SLAVE_ADC);
    XSpiPs_PolledTransfer(&spiInstance_EMIO, writeBuffer,
        readBuffer, 2);
    return (readBuffer[0] << 8) + readBuffer[1];
}

/*Määritellään funktio, jossa AD-muuntimesta luetaan arvo.*/
u16 readWordFromADC(u8 addr) {
    u8 addr1 = addr & 0x1f;                                // 5 bits
    u8 writeBuffer[2] = {
        addr1 << 11,
        0
    };
    u8 readBuffer[2] = { 0, 0 };
    XSpiPs_SetSlaveSelect(&spiInstance_EMIO, SPI0_SLAVE_ADC);
    XSpiPs_PolledTransfer(&spiInstance_EMIO, writeBuffer,
        readBuffer, 2);
    return (readBuffer[0] << 8) + readBuffer[1];
}

/*Määritellään funktio, jonka avulla AD-muunnin
käynnistetään.*/
int startADC() {
    volatile int i;
    ctrlVal &= ~ADC_CONVST_N;
    Xil_Out32(XPAR_AXI4_LITE_TST_IP_0_S00_AXI_BASEADDR, ctrlVal);
    for (i = 0; i < 10; i++) {
        /*Odotetaan.*/
    }
    ctrlVal |= ADC_CONVST_N;
}

```

```

Xil_Out32(XPAR_AXI4_LITE_TST_IP_0_S00_AXI_BASEADDR, ctrlVal);
for (i = 0; i < 20; i++) {
    /*Odotetaan.*/
}
return 0;
}

```

```

/*
    tpx_measurement.c
    C-tiedosto, jossa määritellään mittauksen kulku.
    ArcDia/Jori Soukka
    2020-01-10
    */

```

```

#include "../src/tpx_measurement.h"
#include <sleep.h>
#include <stdio.h>
#include <xtime_l.h>
#include "../src/spi.h"

```

```

volatile TMirror mirror;
extern u8 ctrlVal;

```

```

/*Määritellään funktio, jossa mittaus tapahtuu.*/
int measureSample(int measTime, int minBgT, int maxPartT,
int maxLostT, float stepX, float stepY, int stepBackFactor,
int bgThr, int stopThr, int intThr) {

```

```

    TRunState runState = (stopThr > 0) ? RUN_BG : RUN_PART;
    XTime currT;
    XTime_GetTime(&currT);
    XTime prevT=currT;
    XTime endT = currT + (((XTime) (1000000 * measTime)) *
COUNTS_PER_USECOND);
    XTime bgT = 0;
    XTime foundT = 0;
    XTime intT = 0;
    XTime lostT = 0;
    int userInterrupt = 0;
    int timer = 0;
    u16 scat1 = 0, scat2 = 0;
    const int BASEADDR = TPX_REG32;
    int counter = 1;

```

```

/*Vektoriin data lisätään sekä intensiteetti että
fokuksessaoloaika.*/
extern u32 data [];

/*Fluoresenssisignaalien määrät luetaan signaalinastoista.*/
u32 startphcount = Xil_In32(BASEADDR + 13*4);
u32 finishphcount = Xil_In32(BASEADDR + 13*4);
u32 lscount = Xil_In32(BASEADDR + 14*4);
u32 starttime = 0;

setDriverState(1);

do {
  XTime_GetTime(&currT);
  if (currT < prevT + MEAS_TICK) {
    continue;
  }
  startADC();
  /*Luetaan sironnesignaali AD-muuntimelta.*/
  scat1 = readWordFromADC(0);

  /*Riippuen runStaten tilasta, mennään joko lohkoon RUN_BG,
  RUN_PART tai RUN_STORE.*/
  switch (runState) {

    /*Skannataan liuosta.*/
    case RUN_BG:

      if ((scat1 > stopThr) &&  //( -scat2) (bgT > minBgT *
COUNTS_PER_MSECOND) ) {
        /*Jos havaitaan sironnata partikkelista, liikutetaan peiliä
hieman takaisinpäin ja pysäytetään se ja tallennetaan
fluoresenssisignaalit.*/

        Xil_Out32(XPAR_AXI4_LITE_TST_IP_0_S00_AXI_BASEADDR,
ctrlVal);
        stepBack(stepBackFactor * stepX, stepBackFactor *
stepY);
        startphcount = Xil_In32(BASEADDR + 13*4);
        XTime_GetTime(&currT);
        starttime = currT/COUNTS_PER_MSECOND;
        runState = RUN_PART;
        foundT = currT;
      }

```

```

else {
    /*Liikutetaan peiliä seuraavaan asentoon.*/
    updateScanner(stepX, stepY);

    if (timer%200 == 0){
        XTime_GetTime(&currT);
        starttime = currT/COUNTS_PER_MSECOND;
        u32 startphcount = Xil_In32(BASEADDR + 13*4);

        u32 finishphcount = Xil_In32(BASEADDR + 13*4);
        XTime_GetTime(&currT);
        u32 endtime = currT/COUNTS_PER_MSECOND;

        float phocount = finishphcount - startphcount;
        float phtime = endtime - starttime;
        float intensity = phocount/phtime;
        printf("Photon_count_was_%.6f\n", phocount);
        printf("Photon_time_was_%.6f\n", phtime);
        printf("Intensity_was_%.6f\n", intensity);
    }

    if (scat1 < bgThr) {
        bgT += currT - prevT;
        if (bgT > (XTime) 10 * COUNTS_PER_SECOND) {
            // Long bg measurement --> store
            runState = RUN_STORE;
        }
    }
}
break;

case RUN_PART:

    /*Kun on havaittu partikkeli, tallennetaan aika jonka se on
    fokuksessa ja sen intensiteetti vektoriin data.*/

    if (scat1 > intThr + bgThr) {
        lostT = 0;
        intT += currT - prevT;
        if (currT - foundT > maxPartT * COUNTS_PER_MSECOND) {
            runState = RUN_STORE;
        }
    } else {
        lostT += currT - prevT;

```

```

if (lostT > maxLostT * COUNTS_PER_MSECOND) {
    runState = RUN_STORE;
    XTime_GetTime(&currT);
    finishphcount = Xil_In32(BASEADDR + 13*4);
    float phocount = finishphcount -
    startphcount;

    u32 endtime = currT/COUNTS_PER_MSECOND;
    float phtime = endtime - starttime;
    float intensity = phocount/phtime;

    data[counter] = phtime;
    data[counter+1] = intensity;

    counter = counter + 2;
}
}
break;
case RUN_STORE:
    bgT = 0;
    foundT = 0;
    intT = 0;
    lostT = 0;

    Xil_Out32(XPAR_AXI4_LITE_TST_IP_0_S00_AXI_BASEADDR,
    ctrlVal);
    runState = RUN_BG;
    break;
}
prevT=currT;
}
while ((currT < endT) && (!userInterrupt));
    setDriverState(0);
    return 0;
}

```