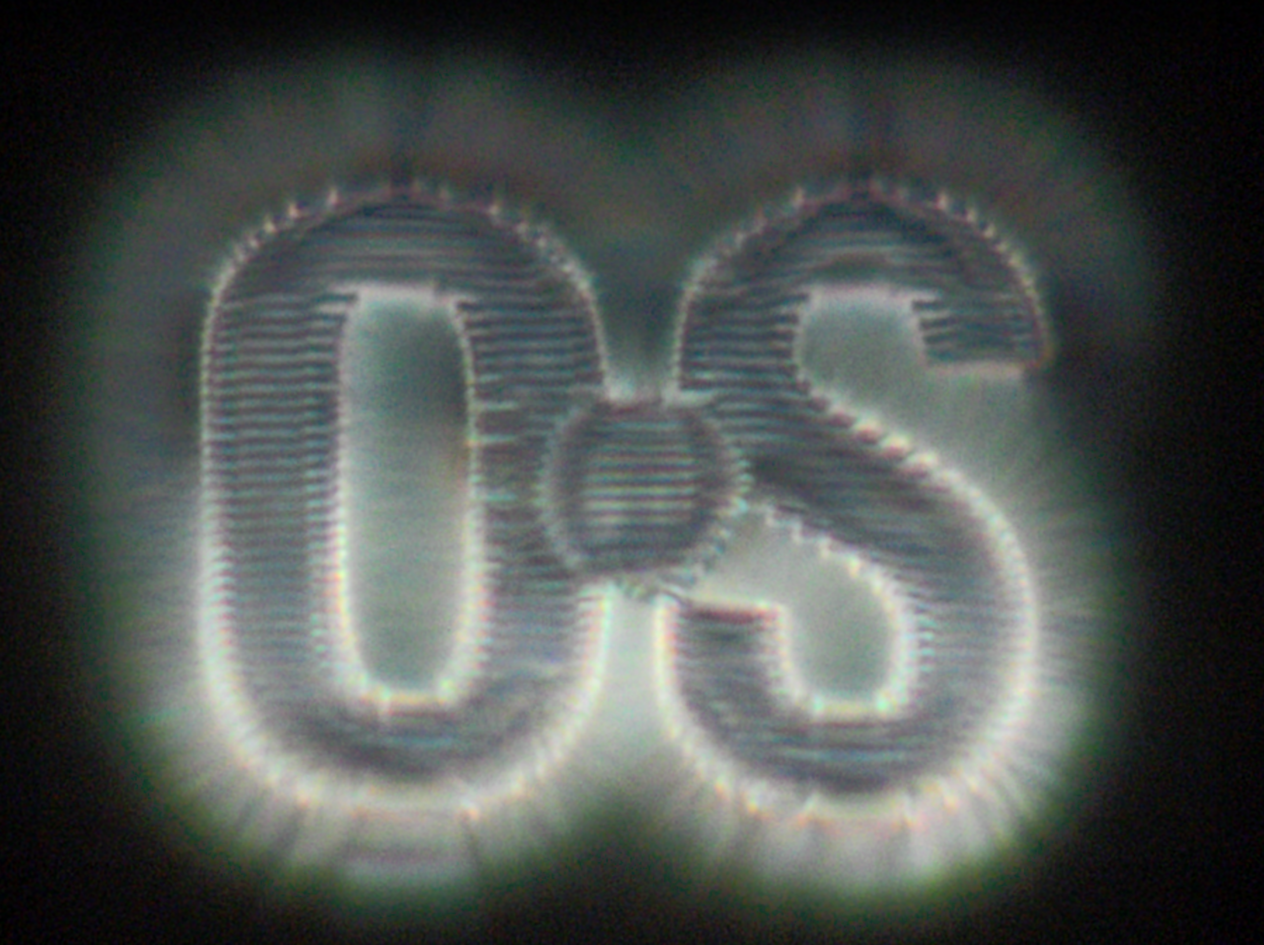
High speed direct laser writing in microchannels



Niels Spannenburg

BSc thesis

December 12, 2013

University of Twente

High speed direct laser writing in microchannels

Niels Spannenburg

December 12, 2013

A bachelors thesis presented to the Faculty of Science and Technology of the University of Twente in partial fulfilment of the requirements for the degree of Bachelor of Science.

Graduation Committee
Dr. Ir. H.L. Offerhaus, Optical Sciences, University of Twente
Dr. Ir. G. Koster, Inorganic Materials Science, University of Twente
J.J.F. van 't Oever MSc, Optical Sciences, University of Twente

Optical Sciences
University of Twente
P.O. Box 217
7500 AE Enschede, the Netherlands

Abstract

This report is the result of the bachelor thesis assignment of Niels Spannenburg. The goal of the assignment was to gain experience with using a newly acquired piece of cleanroom equipment, the Nanoscribe Photonics Professional, and to apply this experience to solve the problem of low writing speeds. Additional goals were to identify and create structures of interest and exploring writing inside microchannels.

Theoretical and practical work has been done to increase the writing speed of the Nanoscribe Photonics Professional. Multiple methods to decrease writing times have been explored. Optimal values for the laserpower, piezo scan speed and settling time when writing structures with the machine were determined. A speedup of over 186x was achieved. Microfilters, woodpiles and multiple kinds of pillars were successfully designed, scripted, fabricated and characterized. For the first time structures were written directly inside microchannels. Coherent anti-Stokes Raman and fluorescence spectroscopy were explored and found to be suitable for characterizing polymerized structures inside microchannels.

Contents

Chapter 1

Introduction

1.1 Motivation

Fresh, clean drinking water is one of the most important resources on the planet. Ubiquitous and cheap in some places, rare and a source of strife in others. Developing technology to control the quality of water is therefore important. One of the possible contaminants in water are bacteria that can cause typhoid, dysentery, gastroenteritis, infectious hepatitis, and cholera, among other illnesses. Methods to check water supplies for harmful bacteria like e. coli, salmonella or cryptosporidium using microbacterial indicators are available but commonly require an incubation period and a laboratory. This prevents these methods from being used in-line. If one was able to detect these pathogens faster it would open up new ways to continuously control waste flows and water supplies.

1.1.1 Early detection of pathogens

The early detection of pathogens is a project involving the Optical Sciences and the Physics of complex fluids groups at the University of Twente, funded by the Wetsus centre of excellence for sustainable water technology. The aim of the project is to be able to detect a single bacterium or cyst per 10ml of water where 10ml is sampled within 1 hour. To achieve this a two step approach is taken. First the concentration of the to be detected particles is to be increased by a factor 100, then the particles will be detected by stimulated emission or stimulated Raman spectroscopy.

The concentration step was first attempted by using a deterministic bumping array[1] as shown in figure 1.1. It was determined that the bumping array works very well for samples containing particles of uniform size but does not perform adequately when multiple sizes of particles are present. Because bacteria and spores are not uniform in size another approach was taken to concentrate samples.

In 2006 it was shown by Laurell *et al* that it is possible to manipulate particles acoustically in microchannels[2]. Based on this work an attempt was made by Jorick van 't Oever to concentrate particles with an ultra-sonic standing wave inside a silicon microchannel. Figure 1.2 shows an experiment by him in which a ultrasonic standing wave is created inside a silicon-glass microfluidic channel by a piezo element. A solution containing 20 µm polystyrene beads is slowly flowing

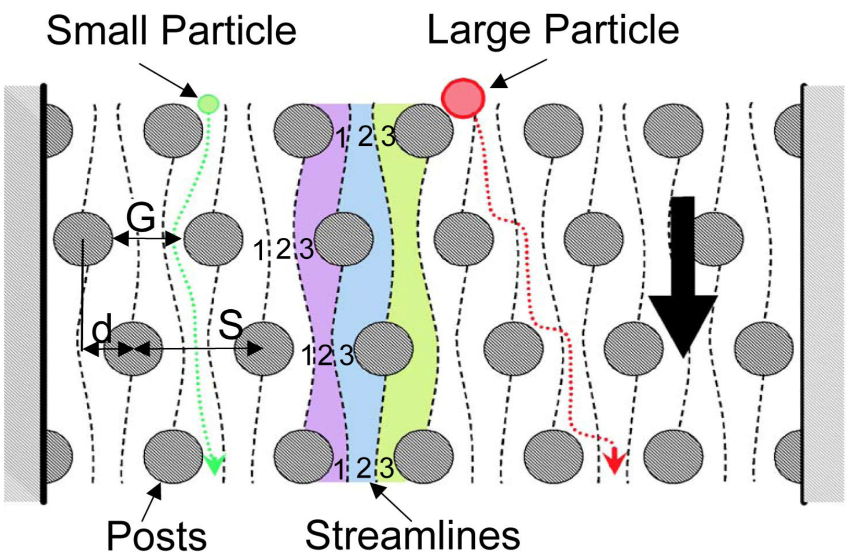


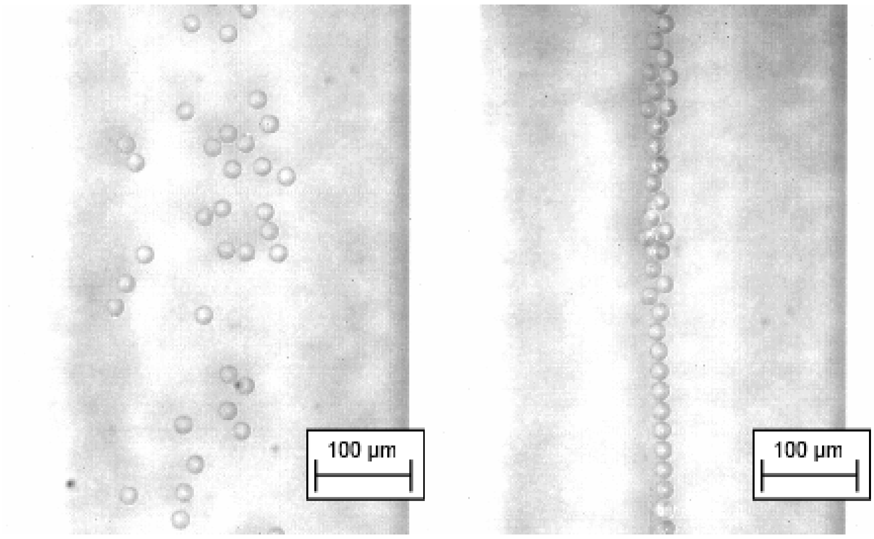
Figure 1.1: Seperation of particles by bumping.

through the channel and when the wave is applied the beads cluster together in the centre of the channel. If the centre part of the flow is then separated from the rest an increase of concentration will have been achieved. This method is not problem-free however as for smaller particles the force applied by the standing wave will become too small compared to the Stokes drag caused by ultra-sonically induced flows, so called streaming[3]. To be able to use this method of concentration for smaller particles it is therefore necessary to reduce this effect or compensate for it.

One way of reducing streaming inside the microchannels is by placing walls or pillars inside the channel that obstruct the undesired flows. Creating these structures with established photolithography and anisotropic wet etching methods would be difficult: these structures would be made of silicon which is not transparent to the ultrasonic wave applied to the channel. This changes the behaviour of the wave inside the channel and thus the movement of particles. It would also be a challenge to create full channel height pillars that properly bond to the glass wafer part of the channel and iteration would be discouraged by high photolithographic mask costs. Finding another way to create these structures inside the microchannels was therefore interesting. Jorick van 't Oever made some attempts to use the recent addition to MESA+'s Nanolab, the Nanoscribe Photonic Professional, to directly write structures inside the channel which has not been done before. He found that the system was promising but not fast enough, additional steps needed to be taken to discern if the method was suitable for the project.

1.2 Objectives and outline

The research presented in this work is a follow up of the work done by Jorick van 't Oever. Its aim is to find the limits of writing relatively big structures with direct laser writing, how these limits can be overcome, what the suitable photoresists are and what interesting structures can be made. This section summarizes the choices that have been made in order to achieve these



(a) Ultrasonic wave off. **(b)** Ultrasonic wave on.

Figure 1.2: Microchannels filled with 20 µm polystyrene beads showing the effect of the ultrasonic standing wave (1.98 Mhz).

goals and will give an outline of the report.

1.2.1 The writing system

Chapter 2 describes the Nanoscribe Photonic Professional system, the two photon polymerization process and the software used to design and script structures. It will also mention the photoresists that can and have been used and the general workflow used. It will also describe the objective lenses used to focus the laser beam, chapter 3 expands on this by showing a derivation of the relation between objective characteristics and the size of the per-shot polymerized volume (from here on referred to as voxel).

1.2.2 Hitting the (speed) limits

To increase the writing speed of the system, which is the limiting factor in the fabrication process, five methods were identified:

- Optimizing structure design
- Increasing piezo scan speed
- Decreasing piezo settling time
- Increasing voxel size
- Optimizing software

Chapter 4 details how these methods were used to improve write times of chosen structures and shows the iterations and choices made during the design, fabrication and characterization of them.

1.2.3 Test structures

Initially rectangular cuboids (bars) were fabricated, they were designed to be 40 µm long, 20 µm wide and 10 µm tall. These relatively small structures were chosen to be able to rapidly iterate and perform parameter sweeps. Later single width pillars were made to test the width of the written voxels and what the achievable aspect ratio is. Woodpile structures were also made to test voxel dimensions, both width and height.

1.2.4 Other structures

Multiple designs for microscopic filters were also fabricated with different photoresists and other parameters. The logo of the research group was also recreated on a micrometre scale. These structures were characterized using optical and scanning electron microscopy (SEM). Chapter 4 also shows the workflow and setup made for writing inside microchannels as opposed to the normal substrates used. Characterizing these structures non-destructively with SEM is however not possible, and multiple alternative characterization methods are proposed and examined.

Chapter 2

Nanoscribe Photonics Professional system

2.1 Introduction

The device used to fabricate the structures is a new addition to the cleanroom, it was installed in March of 2013. It allows for 3D printing on a nanometre scale. The principle on which it operates is two photon polymerization: it uses a focused laser to polymerize small voxels inside a larger volume of photoresist to create structures. This method of fabrication was first proposed in 1997 by Maruo *et al.*[4]

The Nanoscribe company was founded in 2007 and partnered with Zeiss in 2008. When this project started there was not a lot of in-house experience with the setup, many things were learned along the way. This chapter describes the Nanoscribe Photonic Professional system, the two photon polymerization process and the software used to design and script structures. It will mention the photoresists that can and have been used and the general workflow used. It will also describe the objective lenses used to focus the laser beam.

2.2 The setup

As shown in figure 2.2, the Photonic Professional consists of a combination of optical and mechanical parts. The laser beam is generated by a femtosecond pulse fibre laser. The laser power is calibrated by the use of an acousto-optic modulator (AOM). The beam is then focused into the substrate by an objective. For writing structures the focus needs to be moved relative to the substrate, in the Photonic Professional this is done by moving the substrate with a combination of a motorized coarse positioning stage and a piezo stage. There is another version of the machine, the Photonic Professional GT, that adds galvanic mirrors to change the laser focus spot position. The piezo stage has a range of 300 μm by 300 μm with a positioning accuracy of 10 nm and is usually used during the writing process. The motorized stage has a positioning accuracy of 1 μm and is used to change between substrates and substrate positions, but also for writing structures larger than the piezo range allows. The writing process is imaged using a

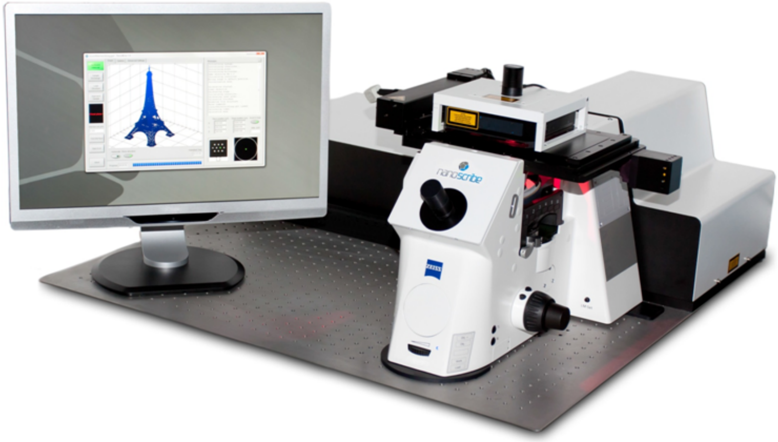


Figure 2.1: The Nanoscribe Photonic Professional.[5]

camera attached to a Zeiss inverted microscope. It is for the most part a black box device, as the only thing that is easily adjustable is the objective.

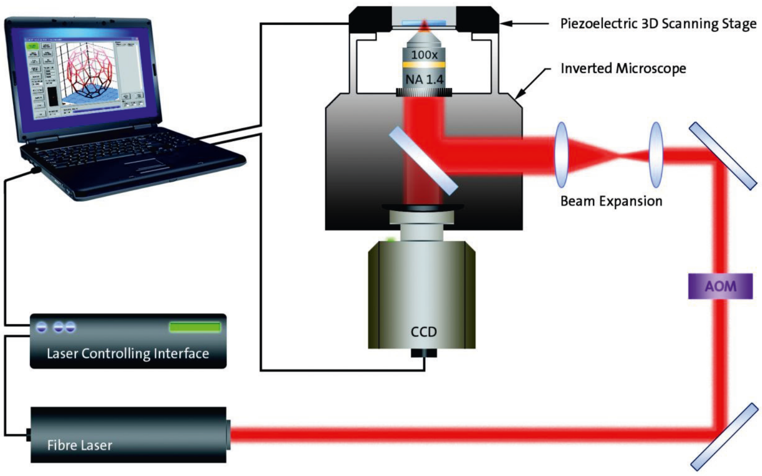


Figure 2.2: The Photonic Professional setup.[5]

2.3 Two photon polymerization

The system writes structures using a two photon polymerization process. In the more common single photon lithography used for fabricating silicon integrated circuits structures are made by applying a layer of photoresist to a substrate, a mask is placed over it and is then exposed to a large amount of ultraviolet light. This polymerizes the entire photoresist layer except where the UV light is blocked by the mask. The polymerized photoresist then protects the underlying silicon from the etching chemicals that remove part of the substrate. When the polymerized

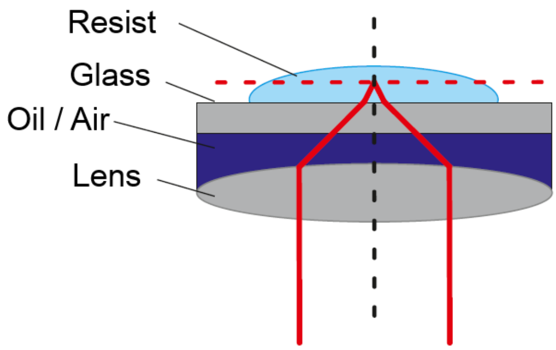


Figure 2.3: Close-up of the setup.

photoresist is removed one ends up with a 2D silicon structure.

With two photon polymerization the polymerized photoresist is the structure. Instead of exposing a large volume of photoresist, a small voxel is exposed by focusing a laser beam into it with a high NA objective, as in figure 2.4. To write a small voxel and not a big cone the wavelength of the laser is changed. The photoresist used is sensitive to UV light, meaning that it polymerizes when it absorbs photons with a wavelength of 390 nm. Instead a laser is used that emits photons with wavelength of 780 nm. Because they have double the wavelength and thus half the energy they can't polymerize the resist, but if the photoresist molecules can absorb two of these photons at the same time polymerization will take place. This is called two photon polymerization. It is a non-linear process and the polymerization rate is proportional to the square of light intensity. Therefore a very high photon intensity is required for the two photon polymerization process to take place. To achieve this high photon density a pulsed erbium doped femtosecond fibre laser is used with an average power of about 100mW with pulse length between 100 and 200 fs. The laser beam is then focused into the photoresist. If the intensity exceeds the necessary threshold the usually ellipsoidal shaped voxel is polymerized. The size of the voxel can be changed by changing the laser power or the focusing optics. The short pulses of the laser also prevent thermal effects from affecting the writing process.

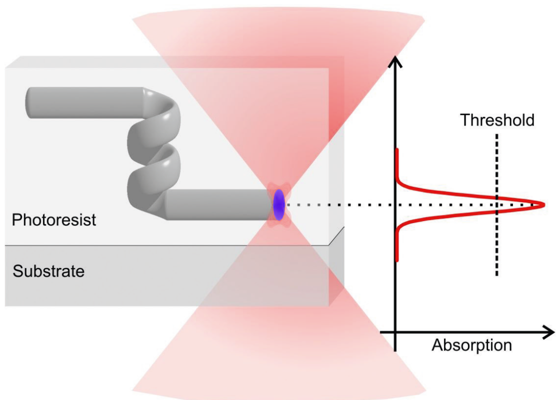


Figure 2.4: The writing process.[5]

2.4 Software

There are two ways to design structures for this system, one way is to use a computer-aided design (CAD) program like Solidworks or AutoCAD to create a 3D model of the desired structure. This model can be exported to the Standard Tessellation Language (STL) format. These files can be imported into a piece of proprietary software from Nanoscribe called Nanoslicer which can convert it into Nanoscribe's General Writing Language (GWL) format. This GWL file can then be fed to the NanoWrite program that controls the system. During the conversion process the 3D model described by the STL file is sliced into 2D layers by representing the 3D model as a series of parallel planes, each of these planes is then transformed into a set of hatched lines. These lines are stored as coordinates for the stage system to move to during the writing process. A few settings can be changed like slicing axis, the distance between the planes and the distance between lines. Nanoslicer is normally used to design structures that are easier to model than script.

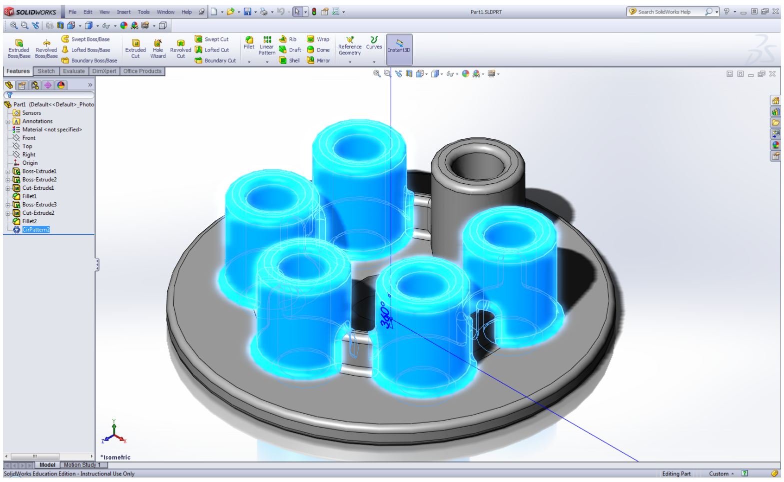


Figure 2.5: An example of a CAD program in action.

The second way of designing structures is to script them directly in GWL using the Describe program. Scripting structures allows for more flexibility with regards to structure size, laser-power, writing speed and so on. In principle any text editor can be used but Describe 2.0 has a few useful features like syntax highlighting, write time estimation, command completion and most importantly a 3D structure preview. GWL is not a very comprehensive language but it does allow one to easily script repeating structures and parameter sweeps with for-loops. Scripting structures by hand is generally done for very simple or repetitive structures or for complex tasks that require the flexibility.

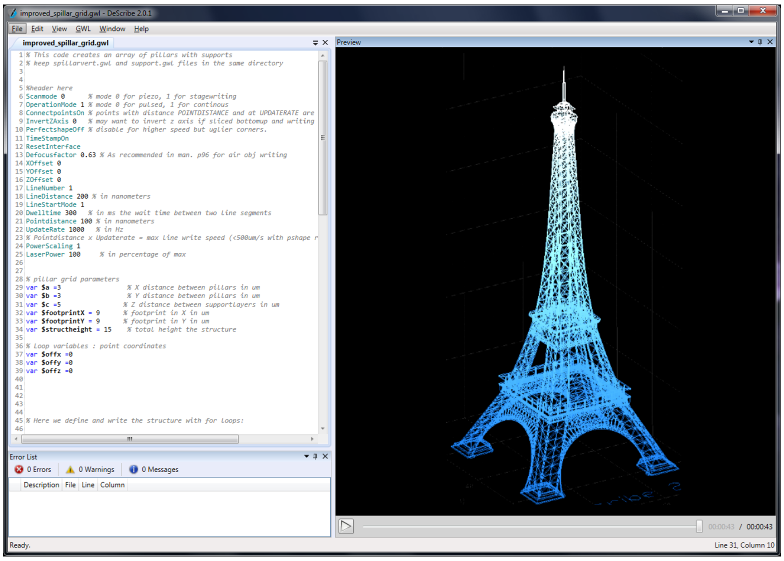


Figure 2.6: The Eiffel tower scripted in Describe.

2.5 Photoresists

Because the laser in the Photonic Professional system emits light at a wavelength of 780 nm any photoresist that is sensitive at 390 nm can be used. There are two groups of photoresists: positive and negative. A positive photoresist becomes soluble to its developer after exposure. This means that to create a structure with a positive photoresist everything that is not the structure is exposed. A negative photoresist is the other way around, what you expose is the structure you end up with, the unexposed photoresist is removed by the developer. In this project only negative photoresists were used to minimize the required writing time. There are two main groups of negative photoresists: acrylate-based and epoxy-based. Acrylate-based photoresists mainly consist of unsaturated monomers with some solvent added to control the viscosity of the resist. During exposure internal bonds inside the monomers can break allowing the monomers to link up with others creating a polymer. This is called radical polymerization. With epoxy-based photoresists no polymerization takes place during the exposure, but instead a photoacid generator is activated releasing an acid. This acid then polymerizes the epoxy groups in the monomers, this is called cationic polymerization. Epoxy-based photoresists need to have very high viscosity to keep the generated acid localized to ensure small feature size capability.[6]

For this project two acrylate-based photoresists were used: IP-L and IP-G. These resists are designed by Nanoscribe for use with the Photonic Professional system and have low shrinkage, good adhesion to glass substrates and are easy to use. They can be applied by dropcasting, don't require a post exposure bake step. Furthermore, IP-L does not require a prebake as opposed to IP-G which needs to be prebaked at 100 degrees Celsius for one hour. Because they are acrylate-based and are thus immediately polymerized during exposure and polymerized resist has different optical properties, the writing process can be observed in real time and in-situ using the Nanowrite software. This allows for rapid prototyping of structures.

2.6 Workflow

A generalized workflow for creating a structure with the Photonic Professional is as follows:

1. Design the structure
2. Prepare substrate and dropcast/spincoat photoresist
3. Prebake
4. Exposure with Photonic Professional
5. Post-exposure bake
6. Development
7. Hardbake

Note that steps 2,3,5 and 7 depend on the photoresist used. IP-L does not require steps 3,5 and 7 and can be dropcast. For IP-G the prebake step is required. An epoxy-based photoresist like SU-8 would require all the steps with the hardbake being optional. To develop the IP resists the samples are first submerged in the solvent RER600 (1-Methoxy-2-propanol acetate) for thirty minutes to remove undeveloped photoresist and then submerged in IPA (isopropyl alcohol) to remove remaining photoresist and RER600. After removing the sample from IPA it is gently blown dry with a stream of nitrogen.

2.7 Objectives

Three objectives were available during the project:

- 100x 1.4NA oil immersion objective
- 63x 0.75NA air objective
- 10x 0.3NA air objective

The oil immersion objective is supplied with the system. The 63x air objective was acquired post-installation. The 10x air objective was on loan from Nanoscribe for this project. During the project the 100x objective was not used because to achieve the goals set a large focus spot was required. Most work was done with the 63x objective. The 10x objective was used to test the influence of low NA on voxel parameters and suitability for high speed writing. The height to width aspect ratio of voxels written with the 100x oil immersion objective is about 3[5].

Chapter 3

Theory

3.1 Introduction

One of the five identified methods to increase writing speed is increasing the polymerized voxel size, this can be achieved by using an objective with a lower numerical aperture. This chapter describes a method to calculate the polymerized voxel height and width as function of the numerical aperture amongst other parameters.

3.2 Calculation of voxel size

For polymerization to take place a high light intensity is required. The polymerization rate is proportional to the square of the light intensity and to the duration that the photoresist is exposed to the laser. The condition for two-photon polymerization is then[7]

$$I^2 \beta \tau v t \geq F_{th} \tag{3.1}$$

With F_{th} the threshold level of the photoresist $[J/m^2]$, β a constant depending on pulse shape $[m^2 s/J]$, τ the pulse duration $[s]$, v the repetition rate [Hz] and t the exposure time $[s]$.

Lets assume our laserbeam has a Gaussian profile.

The intensity profile of the beam I $[W/m^2]$ at the focal plane is defined by[8]

$$I(r) = I_0 e^{-2r^2/w_0^2} \tag{3.2}$$

Where w_0 is the $I = I_0/e^2$ focal spot radius and $I_0 = I(z=0)$.

When we combine formula 3.1 with formula 3.2 and assume we are on threshold of polymerization and the width of the voxel $d = r$.

$$\beta \tau v t I_0^2 e^{-d^2/w_0^2} = F_{th} \tag{3.3}$$

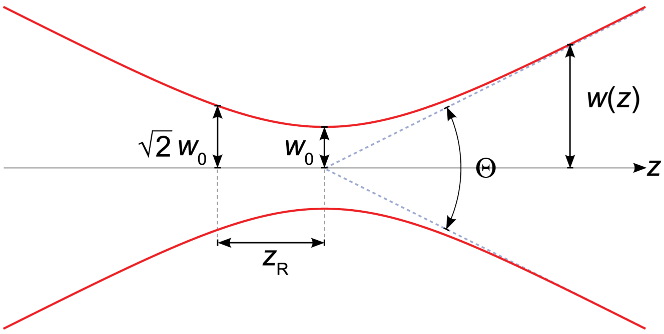


Figure 3.1: Gaussian laserbeam.

$$-\frac{d^2}{w_0^2} = \ln\left(\frac{F_{th}}{\beta\tau vtI_0^2}\right) \tag{3.4}$$

$$d = w_0\sqrt{\ln\left(\frac{\beta\tau vtI_0^2}{F_{th}}\right)} \tag{3.5}$$

Which we can abbreviate into

$$d = w_0\sqrt{\ln(\alpha)} \tag{3.6}$$

The height of the voxel is equal to

$$h = \frac{2z}{n} = \frac{2Z_R}{n}\sqrt{e^{(1/2)(d/w_0)^2} - 1} \tag{3.7}$$

With Z_R the Rayleigh length and n the diffractive index of the photoresist.

The beam width is defined by

$$w(z) = w_0\sqrt{(1 + \left(\frac{\lambda z}{\pi w_0^2}\right)} \tag{3.8}$$

The Rayleigh length (Z_R) is the distance over which the cross-sectional area of the beam doubles, which is at $w(z) = \sqrt{2}w_0$. It follows from the above equation for $w(z)$ that

$$Z_R = \frac{\pi w_0^2}{\lambda} \tag{3.9}$$

If we combine equation 3.7 with equations 3.5 and 3.9 we get (after simplifying which shall be left as an exercise to the reader)

$$h = \frac{2\pi w_0^2}{n\lambda}\left(\sqrt{\alpha} - 1\right)^{1/2} \tag{3.10}$$

At large distances from the focus spot ($z \gg Z_R$) the angular width of the beam Θ approaches $2w(z)/z$ and equation 3.8 can be simplified into

$$w(z) \approx \frac{\lambda z}{\pi w_0} \tag{3.11}$$

Which means that

$$\Theta \approx \frac{2\lambda}{\pi w_0} \tag{3.12}$$

Because the objective used to focus the laserbeam is relatively far away from the focus spot compared to the radius of the objective we can relate the numerical aperture of the objective to the beam characteristics like this

$$NA = n \sin\left(\frac{\Theta}{2}\right) \approx \frac{n\Theta}{2} = \frac{n\lambda}{\pi w_0} \tag{3.13}$$

If we then combine this equation with 3.6 and 3.10 we obtain

$$d = \frac{n\lambda}{\pi NA}\sqrt{\ln(\alpha)} \tag{3.14}$$

$$h = \frac{2n\lambda}{\pi NA^2}\left(\sqrt{\alpha} - 1\right)^{1/2} \tag{3.15}$$

From these equations we can see that the ratio of height to width of the voxel is inversely proportional to the numerical aperture. Meaning that if we increase the voxel size by reducing NA the height of the voxel will increase faster than the width. The aspect ratio is:

$$\mathrm{AR} = \frac{h}{d} = \frac{2}{NA}\left(\frac{\sqrt{\alpha} - 1}{\ln \alpha}\right)^{(1/2)} \tag{3.16}$$

Chapter 4

Fabrication and characterization

4.1 Introduction

This chapter describes the structures that were designed, fabricated and characterized in pursuit of the goals set. To test the influence of structure design, piezo scan speed and piezo settling time bar-shaped structures were made. They were designed to be 40 µm long, 20 µm wide and 10 µm tall. The small size meant a grid of them could be made to test multiple parameters with one writing session. Woodpiles and single width pillars were made to test voxel sizes and aspect ratio. Fabrication of microscopic filters was attempted using different photoresists and designs. After some writing experience was acquired multiple pillar structures were made inside silicon-glass microchannels. The most important parameters that can be modified in the GWL code are:

- Laserpower
- Powerscaling
- Pointdistance
- Updaterate
- Settlingtime

4.1.1 Laserpower and powerscaling

Laserpower sets the power of the laser, it goes from 0 to 100. The system is calibrated so that when laserpower is set to 100 the average power at the aperture of the objective is 20mW. Powerscaling is a multiplier applied to the laserpower. The value of it can be set from 0 to a maximum of about 3.5, the maximum is determined by the calibration process during startup. Powerscaling is usually set to 1. The maximum has shown a small decrease over the course of this project from 3.6 to 3.45, possibly due to the laser going out of alignment or degrading. The actual applied power can be found by multiplying the powerscaling with the laserpower setting times 20mW.

4.1.2 Pointdistance and updaterate

In the writing mode that was used every line that is to be written is transformed into a set of interpolated points for the piezo stage to move to. The pointdistance and updaterate settings control the speed which the piezo moves the substrate, and thus the focus spot of the laser on the substrate. The updaterate is the rate that the points sent to the piezo controller are addressed with in Hz. The pointdistance is the distance between the points and is set in µm. The piezo scan speed in µm/s can be found by multiplying the pointdistance with the updaterate. Note that there are three other writing modes but they are not very useful for our goals.

4.1.3 Settlingtime

The settling time is the time waited between two line segments, this time is necessary for the piezo to travel from the last writing position to the next writing position. It is recommended to use settling times between 100 ms and 500 ms. Lowering the settling time increases overall writing speed, but comes at a cost of decreased piezo positioning accuracy. When using a lower NA objective however the voxel size increases and the piezo accuracy becomes less important.

4.1.4 Standard methods

Unless mentioned otherwise structures were written with the 63x air objective in a bottom-up manner with IP-L photoresist. Bottom-up manner means that the writing process is started at the interface of the substrate, gradually writing up through already polymerized resist. This anchors the structure to the substrate. When writing top-down this is not the case, the writing process starts at the top of the structure and ends at the substrate.

GWL files used to write the structures can be found in appendix A.

4.1.5 Substrates

For the first sections round glass substrates were used with a diameter of roughly one inch and a thickness of 0.17 mm. Later on custom microchannels were used that are etched into a silicon wafer with channel dimensions of 377 µm width, 157 µm height and 10 cm length. Both anisotropic KOH etching and the Bosch process were used to etch the channels, the first creating smooth walls and a somewhat rough bottom, the second creating a flat bottom and walls with a sub-micron roughness. The silicon wafer is then bonded with a glass borofloat wafer with a 500 µm thickness. Holes are then drilled at the endpoints of the microchannel in the silicon wafer to allow access to it.

4.1.6 Characterization

Characterization was done with an Olympus MX61 microscope with 5x, 10x, 20x and 50x objectives with brightfield and dark field modes. SEM pictures were taken with the Zeiss 1550 HR-SEM and the NOVA 600 FIB/SEM.

4.2 Speeding up with bars

The bar GWL files were created by making a 50x50x50 µm cube in Solidworks, which was then loaded into Nanoslicer and transformed into a 40x20x10 µm bar.

4.2.1 Reference bar

For the reference bar the standard settings known at that point were used:

Laserpower (%)	95-100
Powerscaling	1.0
Pointdistance (nm)	100
Updaterate (Hz)	1000
Settling time (ms)	300
Method	bottom-up
Photoresist	IP-L
Line distance XY/Z (nm)	200/200

Table 4.1: Write parameters for the reference bar.

The laserpower varied from 95 at the bottom of the bar to 100 at the top, this is a standard setting in Nanoslicer to compensate for the fact that the photoresists optical properties change slightly when polymerized. This causes the laser focus spot to deteriorate as the light has to move through more polymerized resist as the writing process progresses when structures are written bottom-up. The lines were written along the X axis and were spaced 200 nm apart in Y and Z axis. The camera was on during the writing process.

The SEM picture shown in figure 4.1 is not very clear due to the lack of a conductive coating of paladium/gold that is commonly used when characterizing weak conducting materials with scanning electron microscopy. It shows that the dimensions of the bar are correct, with some unintended rounded corners on top. Total writing time was one hour, 19 minutes and 24 seconds.

4.2.2 Faster bars

To reduce this writing time a number of steps are taken, first the camera is turned off during the writing process. This is an unseemly optimization but it increased structure file loading speeds with a factor 3 and writing speed roughly 20%. The line seperation values used for the reference bar are very small and most likely used when writing with the 100x oil immersion objective. Because we know that the 63x air objective we use has a lower NA and thus higher voxel size we can increase these values. By changing the line seperation values we strongly decrease the number of lines required to write the bar. We also reduced the settling time as we don't need maximum piezo positioning accuracy. We increased the piezo scanspeed as 100 µm/s is not very high. When we increased scanspeed we also increased the laserpower to compensate for reduced dose per exposed volume. We also changed the way the bar is created, instead of a solid block composed of perpendicular lines we only write the outer shell and some "floors" of the bar in the same way that skyscrapers are constructed. Another optimization we made is changing the line hatching process. Normally the lines are perfectly parallel as shown in figure 4.2a. If we exchange the start and end points of every other hatch line we can reduce the amount of time the

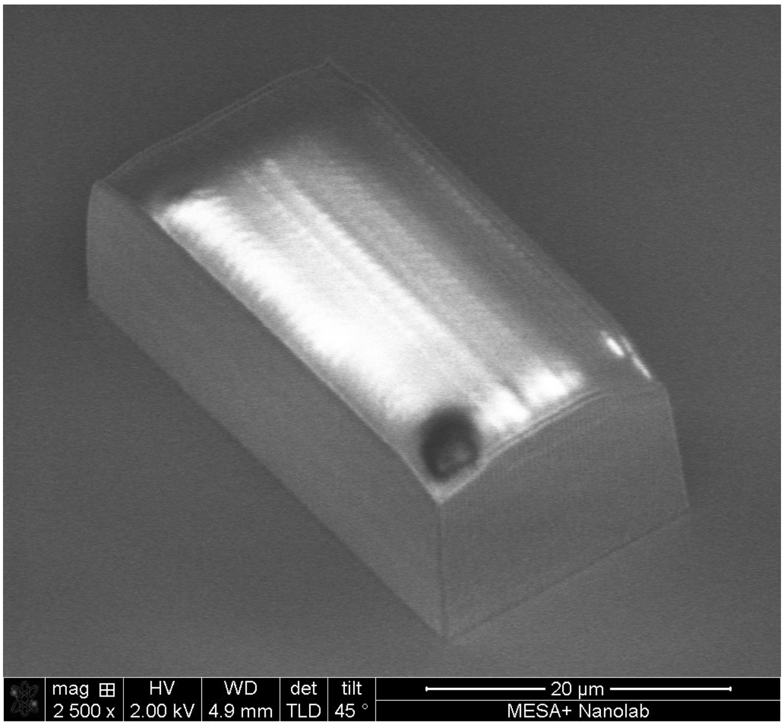


Figure 4.1: SEM picture of the reference bar.

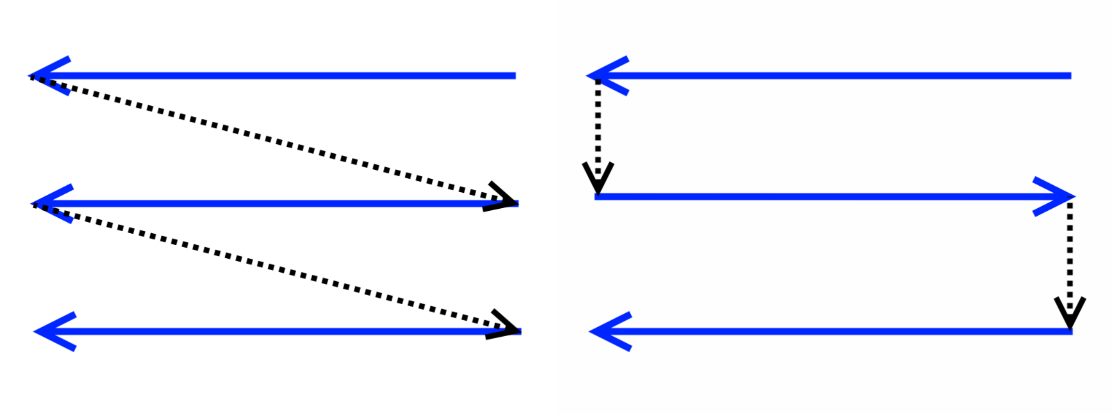
piezo spends moving to a new position, an example of this is shown in figure 4.2b. An example of a bar written with more aggressive settings is shown in figure 4.3. To find optimal values for these parameters large grids of bar structures were created and characterized to get a feel for their limits. By using these settings and changing the line seperation values to 400nm for the Y axis and 2000 nm for the Z axis we can achieve a significant speedup: write time was reduced to 25.6 seconds. A speed increase of 186x compared to the reference bar.

Laserpower (%)	95-100
Powerscaling	2.0
Pointdistance (nm)	1000
Updaterate (Hz)	1000
Settling time (ms)	50
Method	bottom-up
Photoresist	IP-L
Line distance XY/Z (nm)	400/2000

Table 4.2: Write parameters for the bar in figure 4.3.

Interesting to note is that a curled wall appears to be present through the structure. This is caused by the system trying to create the outline of the structure but the piezo not being able to reach the corners due to the high scanspeed. This effect is magnified as piezo scanspeed is increased as shown in figure 4.4.

Testing showed that for the bar structures a piezo scanspeed of 250 µm/s was the maximum



(a) Normal line hatching. **(b)** Back and forth hatching.

Figure 4.2: Two figures showing the difference in piezo movement between the two types of line hatching. Writing lines is in blue. Moving to the starting position of a new line is in black.

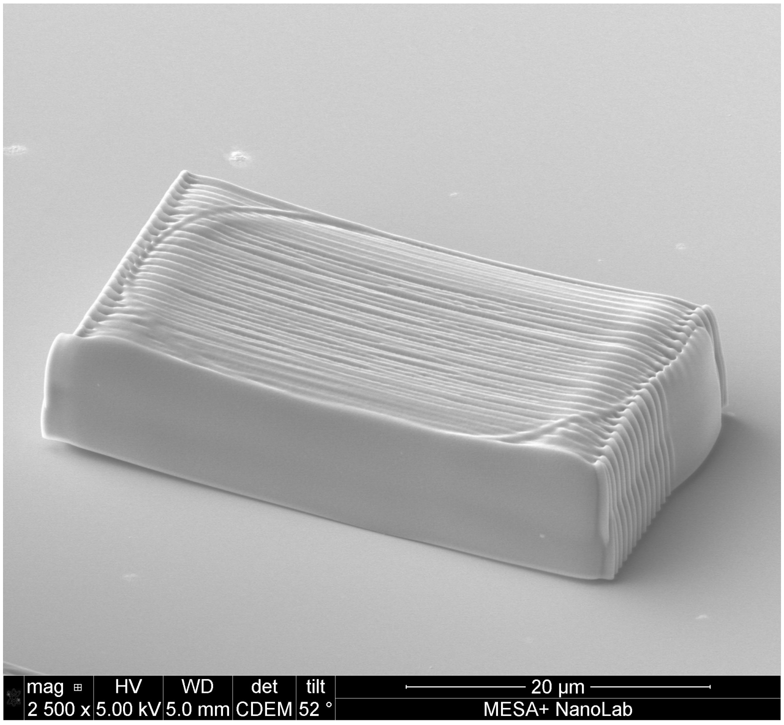


Figure 4.3: SEM picture of a faster written bar.

at which curling does not occur. Note that the outer shell was not written as intended, but the bar shape is still intact despite the intention to create a hollow structure with floors. It appears the voxel is significantly taller than the 2000 nm used for Z axis line distance.

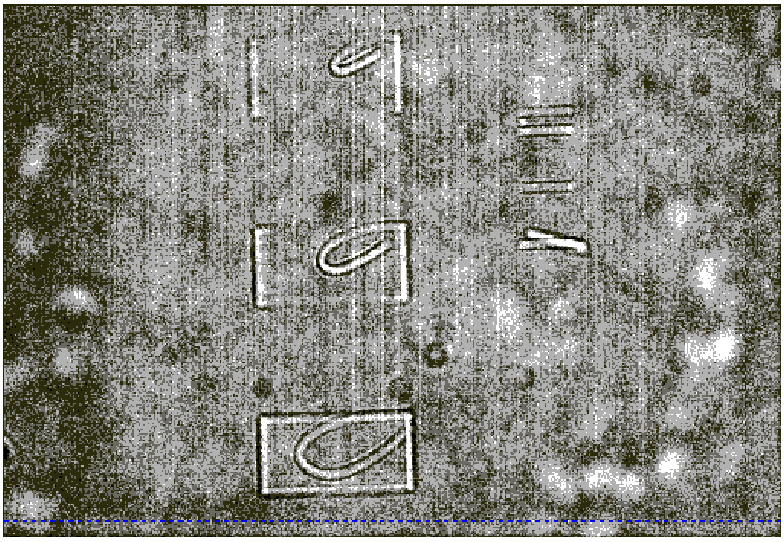


Figure 4.4: Picture taken with built-in camera during writing showing the curling effect at high piezo scanspeeds.

4.2.3 10x Objective bars

Tests were also done with the 10x objective borrowed from Nanoscribe. The comparatively low NA of 0.3 of the objective should create a very high voxel.

Laserpower (%)	100
Powerscaling	2.5-3.6
Pointdistance (nm)	250
Updaterate (Hz)	1000
Settling time (ms)	0-300
Method	bottom-up
Photoresist	IP-L
Line distance XY/Z (nm)	1000/5000

Table 4.3: Write parameters for the structures in figure 4.5.

The written structures can be seen in figure 4.5, they show no resemblance to the intended 40x20x10 µm bars. The per shot polymerized voxel of the 10x objective is simply too high to create them. The writing of the bar design results in tall pyramid like structures. Note the wall around the structures, it is part of a 500x500 µm marker written to more easily find the structures in the SEM. The marker is detached from the substrate at its corners and shows grass like pillars on top. The aspect ratio of voxels written with the 10x objective is very high, these images allow us to estimate it to be between 20 and 40. These tests show the objective can only be used for structures that are very high (over 40 µm).

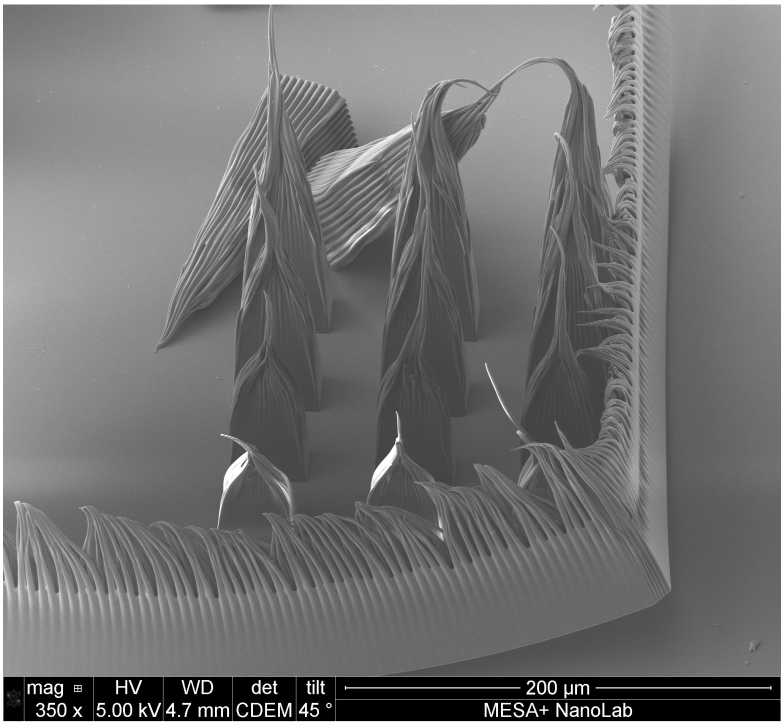


Figure 4.5: SEM picture of structures written with the 10x air objective.

4.3 Single width pillars

A field of single width pillars was made to test the voxel width and the highest achievable aspect ratio. The single width pillars are called that because the instruction given to the Photonic Professional is to write a single line upwards, creating the thinnest pillar it can make with the objective and laserpower settings used. A field of 100 pillars ranging from 1 μm high to 50 μm high was made in a 10x10 grid.

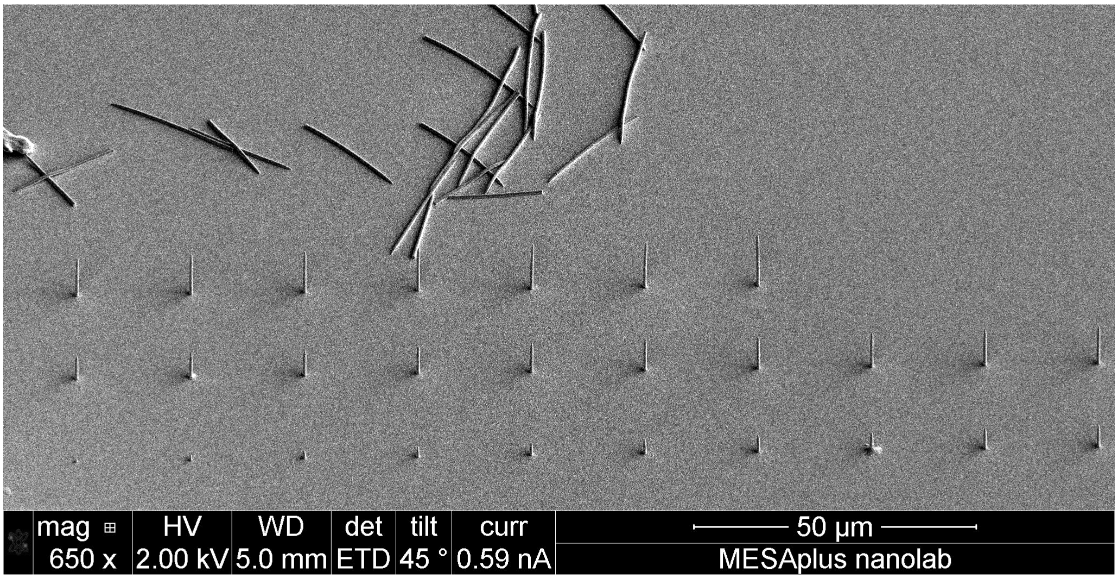


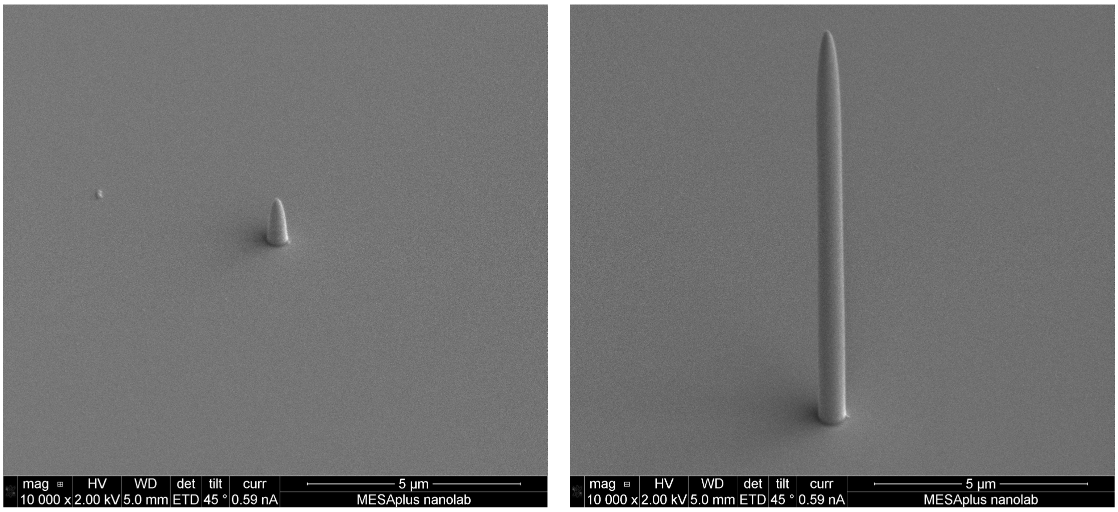
Figure 4.6: SEM picture of the pillarfield.

Laserpower (%)	100
Powerscaling	1.5
Pointdistance (nm)	100
Updaterate (Hz)	1000
Settling time (ms)	300
Method	bottom-up
Photoresist	IP-L
Line distance XY/Z (nm)	N/A

Table 4.4: Write parameters for the pillar field.

As figure 4.6 shows, a large amount of pillars did not survive development upright. The tallest still standing measured 13 μm, the smallest was measured to be 1.6 μm tall.

The pillars that did not maintain an upright position showed a clean disconnect from the substrate at their base. They were most likely toppled during development. The fallen pillars allow us to give get an accurate measurement of the voxel width. It is measured to be about 590 nm as shown in figure 4.8. This means that the highest aspect ratio achieved was 22.



(a) The smallest pillar. **(b)** The tallest pillar still standing.

Figure 4.7: Two figures showing the shortest and tallest pillars in the pillar field.

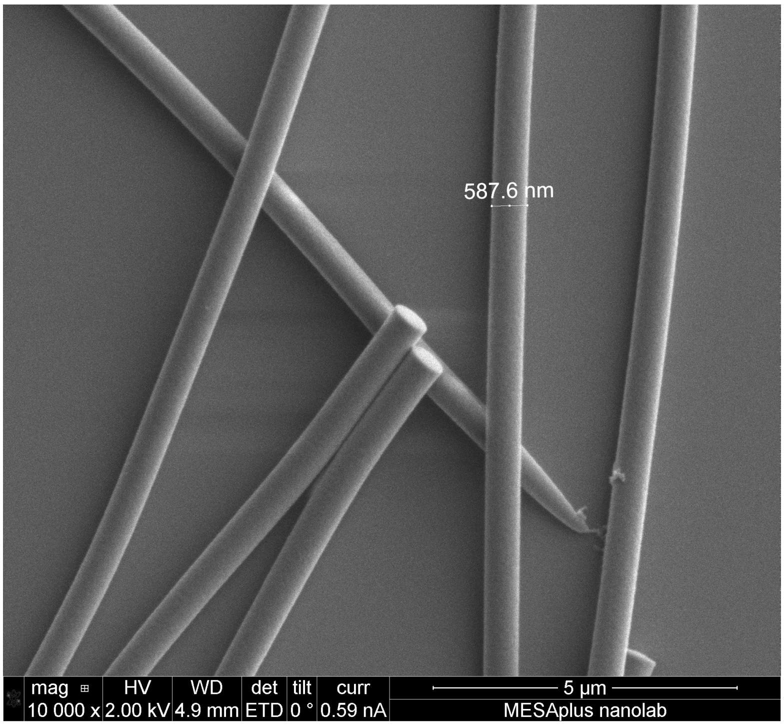
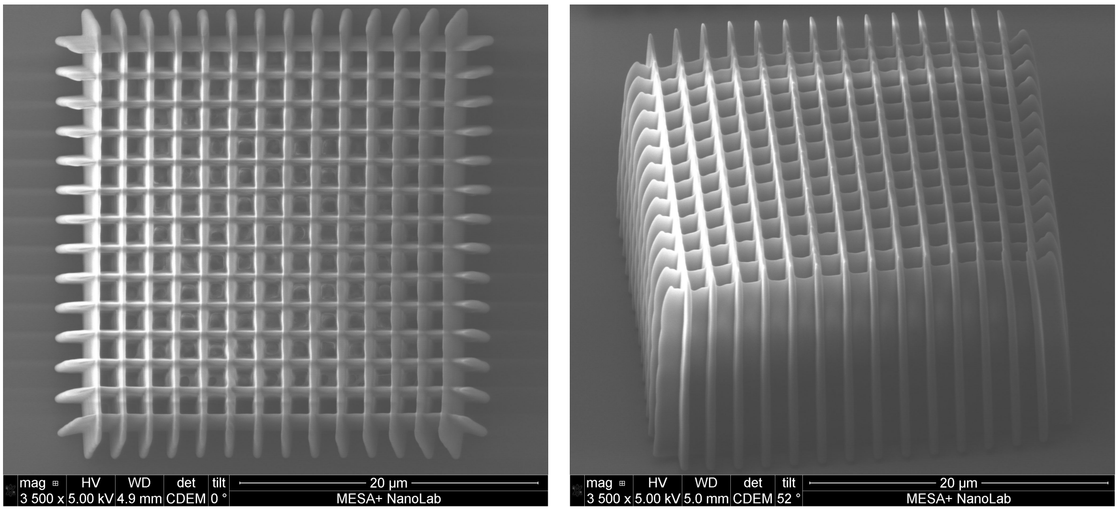


Figure 4.8: SEM picture of the toppled pillars.

4.4 Woodpiles

Now that the width of the voxel when writing with the 63x air objective is known, we strived to find out the height of the voxel. One way to do this is by writing a large amount of single voxel dots above the substrate, hoping enough stick to the surface after developing and characterizing them with scanning electron microscopy. Another option is to create woodpile structures and create tilted SEM images of them. Multiple woodpiles were created with a base of 30 µm by 30 µm and 15 µm tall. Height difference between the pile layers was chosen to be 1.5 µm.



(a) Top down view of early woodpile attempt. **(b)** Tilted view of early woodpile attempt.

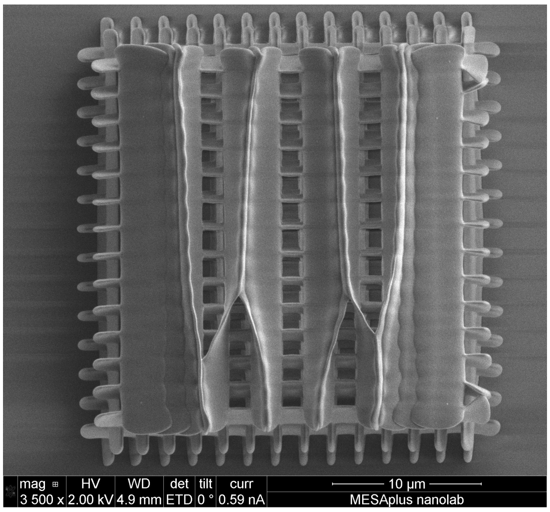
Figure 4.9: Two views of an early woodpile attempt.

Laserpower (%)	100
Powerscaling	1.5
Pointdistance (nm)	100
Updaterate (Hz)	1000
Settling time (ms)	300
Method	bottom-up
Photoresist	IP-L
Line distance XY/Z (nm)	N/A

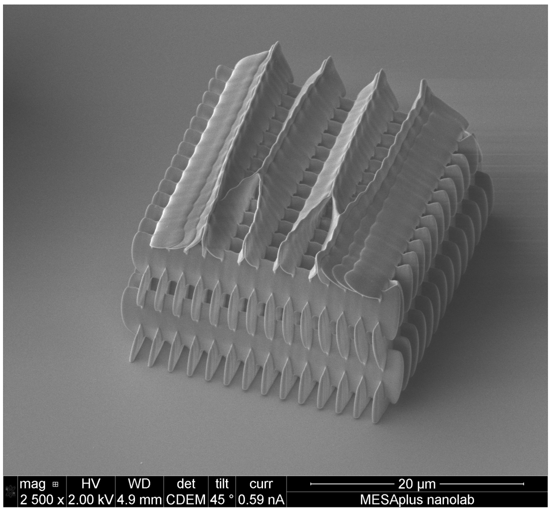
Table 4.5: Write parameters for the woodpile in figure 4.9.

Figure 4.9 shows that these settings don't result in a woodpile but more of a waffle structure: the written voxel is considerably higher then earlier estimated and this causes significant overlap between pile layers. More woodpiles were made with a higher distance between the layers, 4.5 µm was chosen for the woodpile in figure 4.10a. Other writing settings were kept constant.

The tilted view in figure 4.10b shows that the voxel is over 6.3 µm tall for the lower layer (not the first layer that clips into the substrate for stability) and decreases slightly to 5.5 µm for the layer above it. The voxel is much higher than originally estimated. Combined with the width found using the pillars we can say the 63x objective produces voxels with an aspect ratio



(a) Top down view of improved woodpile.



(b) Tilted view of improved woodpile.

Figure 4.10: Two views of an improved woodpile design .

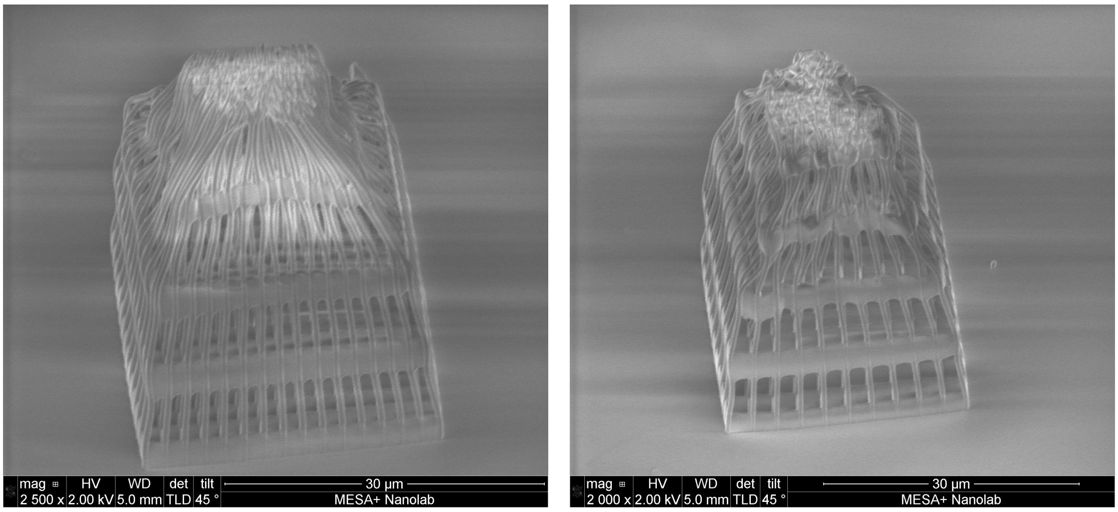
of around 10. From the theory and the aspect ratio we found for the 10x objective we would have expected it to be between 8.5 and 16. Note in figure 4.10a that for the top layer the polymer walls seem to collapse into each other. This is most likely caused by capillary forces during the drying process post-development.

4.5 Microfilters

Being able to write arbitrary 3D structures has some distinct advantages over common 2D lithography. A structure that could not have been made with 2D lithography is a microfilter. We envision a filter directly written inside a microchannel that has gaps in it that allows fluids through but blocks particulates larger than a few microns like bacteria. The first microfilter design was intented to create filters 30 μm by 30 μm at the base and 50 μm high. The code instructed the machine to first write a grid of 50 μm high pillars separated by a few microns and then bind them together with support layers in the XY plane. The support layers are spaced 10 μm apart.

Laserpower (%)	100
Powerscaling	1.0
Pointdistance (nm)	100
Updaterate (Hz)	1000
Settling time (ms)	300
Method	bottom-up
Photoresist	IP-L
Line distance XY/Z (nm)	N/A

Table 4.6: Write parameters for the microfilters in figure 4.11.



(a) Filter with 2 micron pillar separation. **(b)** Filter with 3 micron pillar separation.

Figure 4.11: Two views of an improved woodpile design .

As shown in figure 4.11 these filters look quite acceptable close to the glass substrate. But as filter height increases the structure is malformed, the top support layer is not even present. Possible reasons for this include the fact that first the pillars are written and later connected. The photoresist used, IP-L, is not very viscous and this might allow the pillars to move between the time they are written and connected. An analogy to it is to vertically place a handful of uncooked spaghetti on a table and expect them to stay in place while you get a rubber band to hold them together. Another option is that the structure was written bottom-up, this explains the degree of malformation gradually increasing with structure height as the voxel is distorted by having to pass through more polymerized resist. To test these ideas more microfilters were made with IP-G used as photoresist because it is considerably more viscous. This reduces possible flows in the resist during writing. We also compare top-down and bottom-up written filters. Note that it is generally not recommended to write top-down in low viscosity photoresists because there is a strong possibility of flows deforming the structure in writing when it is not anchored to the substrate. The following chapter shows top-down writing also has drawbacks.

Sadly no close-up for the bottom-up written filter is available, but it can still be seen that the written structure is considerably closer to its design than the previous iterations. The top-down written filter shown in figure 4.13 looks slightly better than the bottom-up variant, showing that the main difference in structure quality is caused by writing with a more viscous photoresist. Also the upper part of the top-down written filter looks significantly less deformed than the bottom-up one, showing the advantage of writing that way.

Another set of microfilters was fabricated using a completely different method. Instead of writing a pillargrid and connecting them, a filter is created by repeating a small rectangular unit cell. This is intended to reduce the influence of possible flows inside less viscous photoresist during writing. Write and design parameters were kept constant, centre to centre distance of the support pillars was varied from 1 µm to 3 µm. Photoresist used was IP-G.

As figure 4.14a shows the unit cell method of writing the microfilters performs worse than the previous method. The structure show significant deformation at the top while looking similar

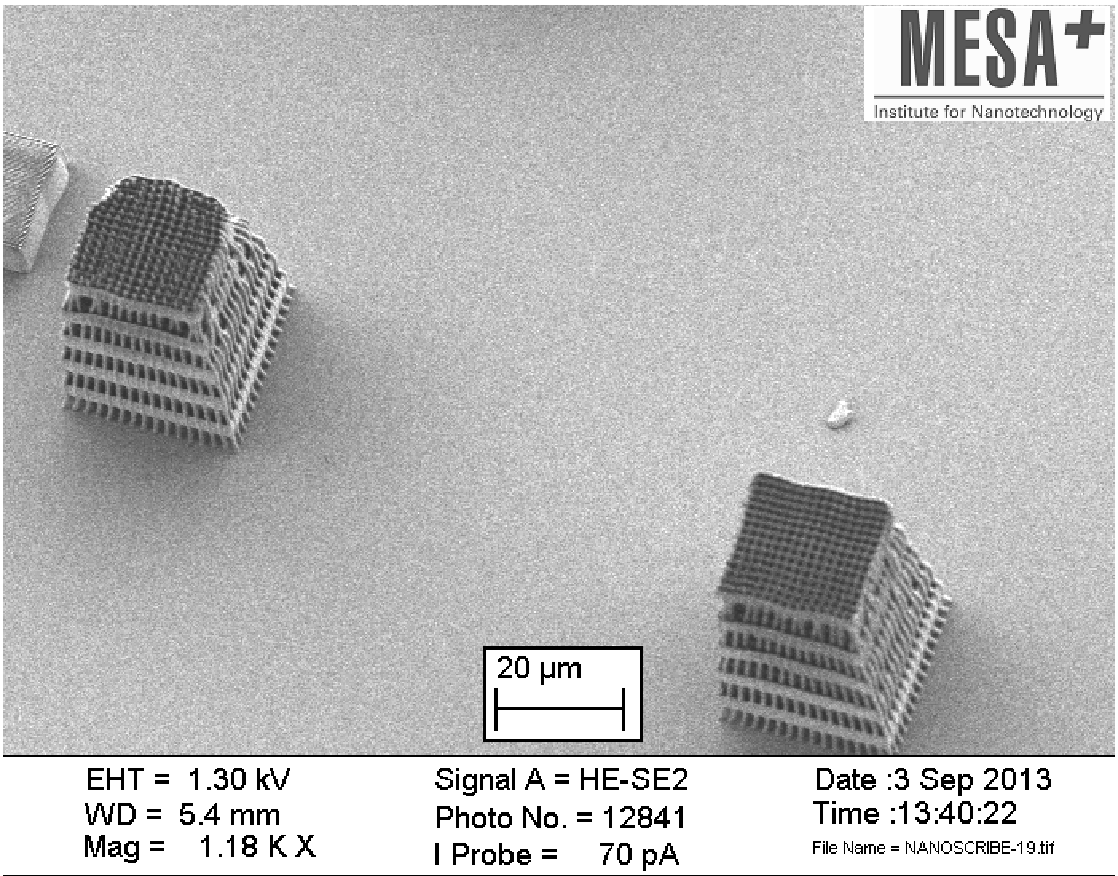


Figure 4.12: Bottom-up filter on the left, top-down filter on the right.

to their predecessors at the base. The write time of these structures also scaled poorly with gap size, the filter with 1 µm centre to centre distance took one hour, 7 minutes and 12 seconds to write compared to the 3 µm version which took 8 minutes and 37 seconds. What we can conclude from these experiments is that for these tall self supporting structures it is strongly beneficial to use viscous photoresist and even then it would be wise to design the structure to be as rigid as possible.

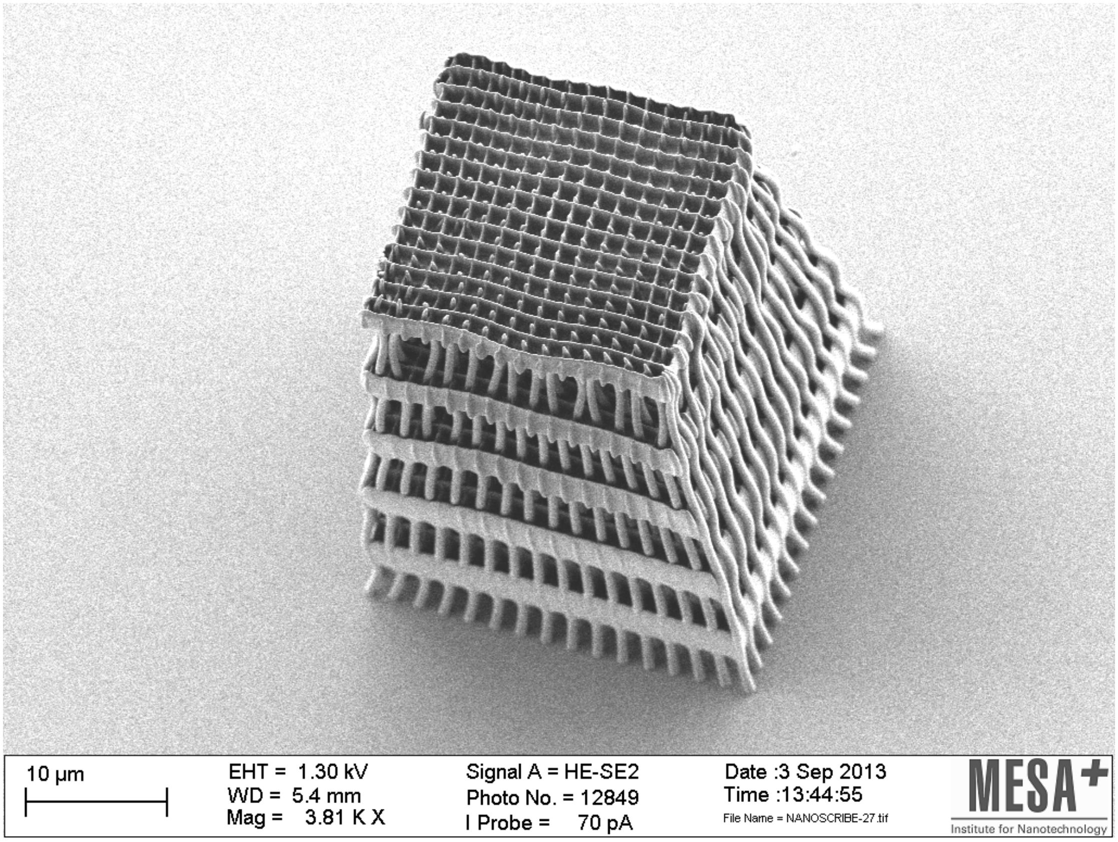
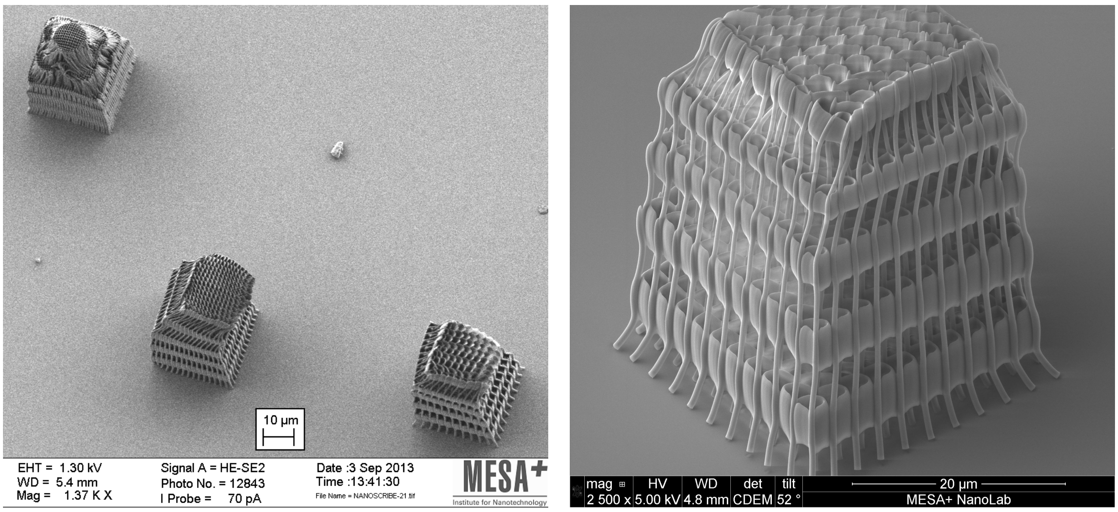


Figure 4.13: Close-up view of the top-down filter.

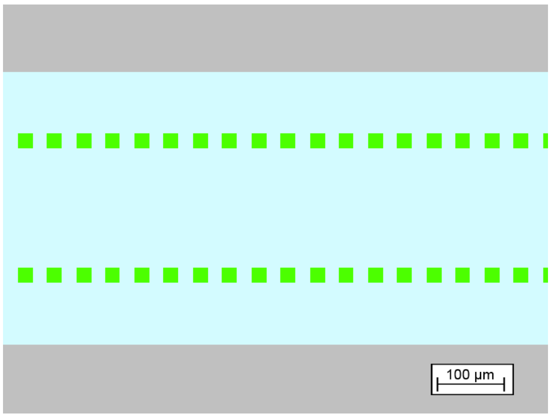


(a) Microfilters written in unit cell manner. **(b)** Close-up of microfilter with 3 µm separation.

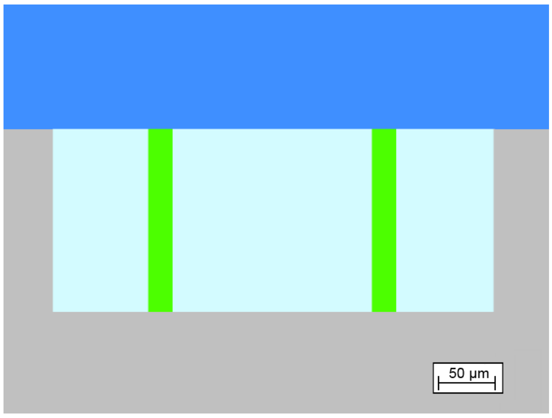
Figure 4.14: SEM pictures of the unitcell microfilters.

4.6 Writing in microchannels

After succesfully increasing the writing speed of the machine with the bars and gaining experience with designing and writing different structures with multiple methods attempts were made to write large structures inside microchannels to reduce the currents preventing small particles from being concentrated with the ultrasonic concentration method. A square pillar structure was chosen, to be placed in two rows inside the microchannel as shown in figure 4.15. The pillars would have to be 150 µm high, which previous experiments have shown to be difficult to make with a bottom-up approach.



(a) Top down view of the microchannel with pillars written inside.



(b) Side view of the microchannel with pillars written inside.

Figure 4.15: Pictures showing the basic pillar in channel design. Grey is the silicon wafer, green the pillars and dark blue the glass top layer.

To test if it would be possible to write the high pillars a grid of 20 µm by 20 µm wide pillars was made with heights varying from 15 µm to 135 µm. They were fabricated bottom-up on a normal glass substrate.

Laserpower (%)	100
Powerscaling	2.0
Pointdistance (nm)	1000
Updaterate (Hz)	1000
Settling time (ms)	0
Method	bottom-up
Photoresist	IP-L
Line distance XY/Z (nm)	400/2000

Table 4.7: Write parameters for the pillar grid made on glass substrate.

As figure 4.16 shows the 15 µm pillar was fabricated successfully but the rest were not. It appears that writing through more than 20-25 µm of photoresist degrades the focused laserbeam enough to cease polymerization. Interesting to note is the fact that the pillars designed to be 45 µm to 135 µm have identical malformed shapes. This indicates that optical diffraction is limiting the maximum height in the case of bottom-up. It is shown that to create the tall pillars

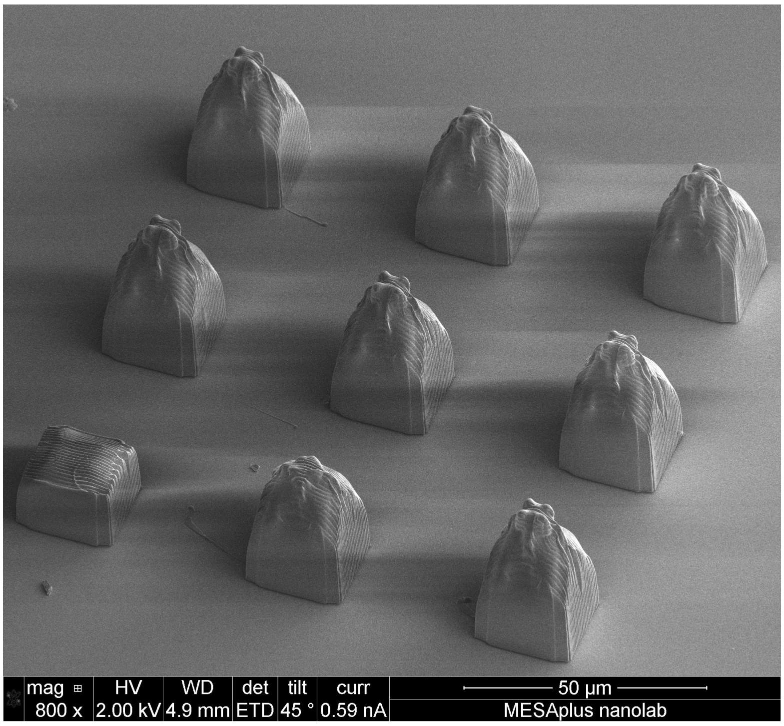


Figure 4.16: Tilted SEM picture of tall pillars on glass substrate.

necessary they will have to be written top-down. Generally this might cause problems because the structures would not be anchored to the substrate, but in the microchannel case we start writing on the silicon surface when using the top-down method, so this is not a concern.

To write in the microchannels a few challenges had to be overcome. The channel itself would have to be filled with photoresist leaving no air bubbles inside. The highly filled channel would then have to be put in the Photonic Professional. After exposure the superfluous photoresist would then need to be flushed out with developer chemicals. This means that viscous resists could most likely not be used and that pre- and post-baking steps are undesirable as solvent gas would bubble out of the resist. This left IP-L as the most likely candidate. For filling the channels with resist and later flushing with developer chemicals a pump setup was made by Jorick van 't Oever. This syringe pump setup as shown in figure 4.17 allows us to slowly pump specific amounts of developer solvent at specific rates.

The first pillars were designed to have a base of 20 µm by 20 µm and be 170 µm high to allow for some overlap with the glass and silicon interfaces it connects to. The line separation distance was set to 400 nm horizontal by 2000 nm vertical. The developing was done with the syringe pump setup by first pushing 20 ml of RER600 through the channel at 20 ml/hr followed by 20 ml of IPA at 20 ml/hr. This developing method was also used for all follow-up experiments inside microchannels.

Note in table 4.8 that the power used was at maximum. This was because small write tests inside the channel indicated no significant polymerization would otherwise occur. The reason for this is that the 63x air objective has a correction ring that was incorrectly set to 0.17 mm. Later

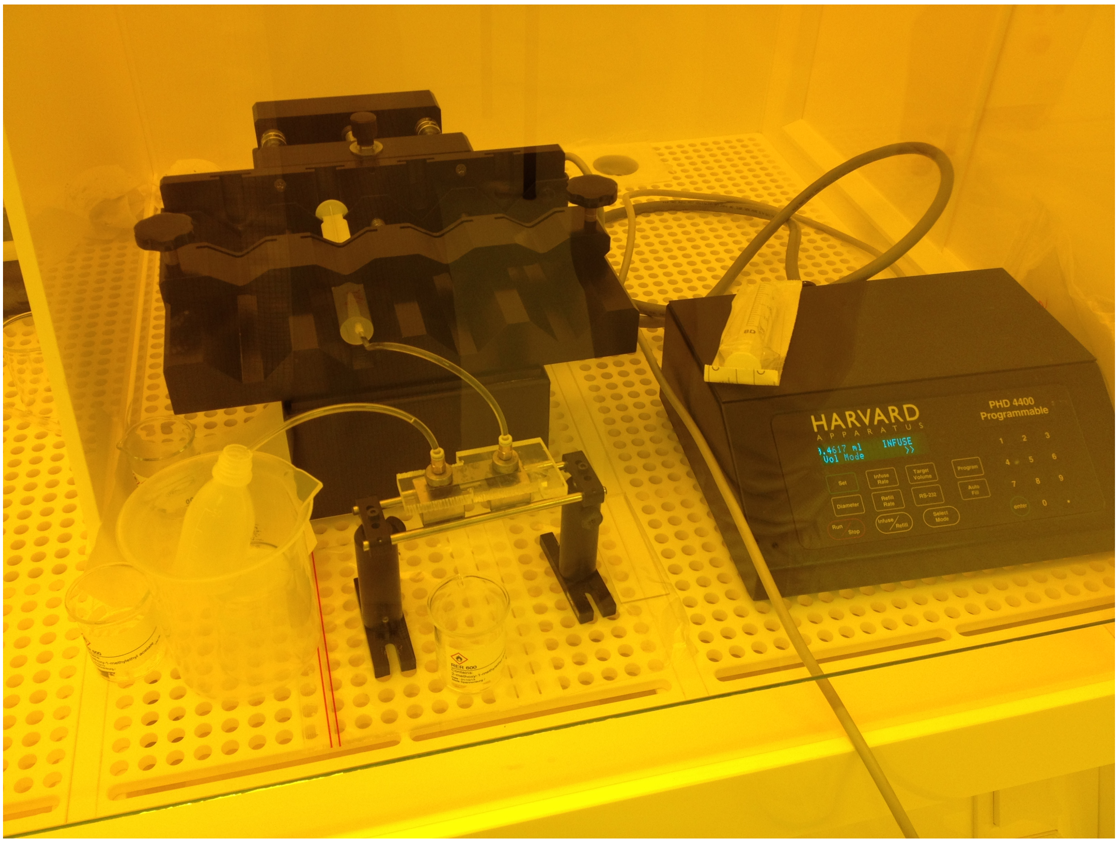


Figure 4.17: Picture of the syringe pump setup in the cleanroom.

Laserpower (%)	100
Powerscaling	3.5
Pointdistance (nm)	1000
Updaterate (Hz)	1000
Settling time (ms)	50
Method	top-down
Photoresist	IP-L
Line distance XY/Z (nm)	400/2000

Table 4.8: Write parameters for the first pillars made inside a microchannel.

it was found that the glass wafer used was 500 µm high, after changing the correction ring to that value polymerization was observable at a powerscaling value of 2.0 with laserpower at 100. Each pillar took 10 minutes and 05 seconds to make. Figure 4.18 shows the pillars surrounded by a bubble of IPA, the developer was not completely removed from the microchannel yet.

To reduce writing time another experiment with pillars inside microchannels was done with significantly smaller pillars. Two 1 cm long rows of pillars were made inside the microchannel. The base of the pillars was changed to 10 µm by 10 µm, they were spaced apart at a centre to centre distance of 20 µm. To increase stability they were connected to each other using crossbars,

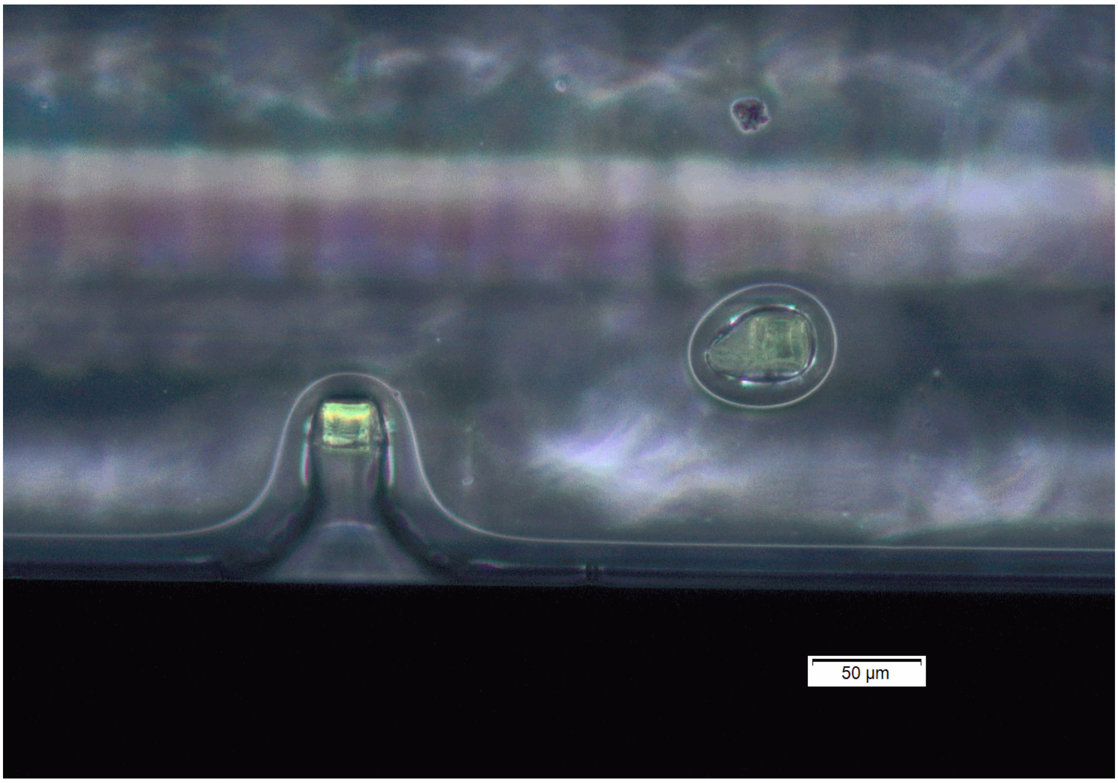
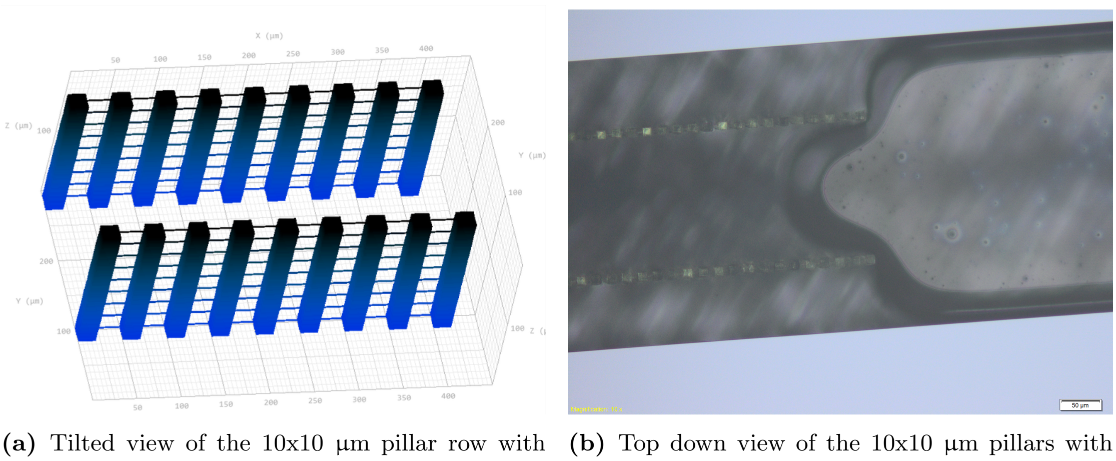


Figure 4.18: Picture of the first pillars created in a microchannel.

making use of the 3D capabilities of the machine.



(a) Tilted view of the 10x10 µm pillar row with crossbars design.

(b) Top down view of the 10x10 µm pillars with crossbars in the channel.

Figure 4.19: Pictures showing the 10x10 pillar row design and written structure.

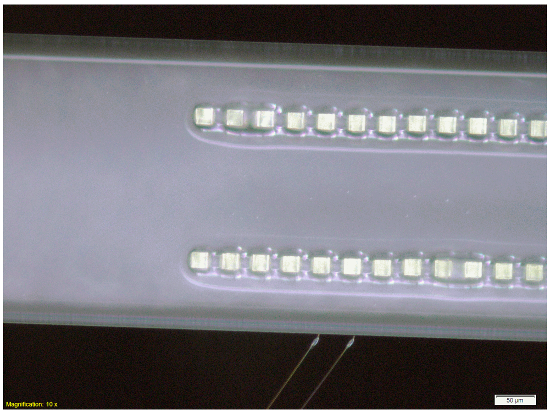
The total write time of this centimetre long design containing 1000 pillars was 47 hours, 24 minutes and 20 seconds. Interesting to note in figure 4.19b is the fact that the pillars are causing

Laserpower (%)	100
Powerscaling	3.4
Pointdistance (nm)	1000
Updaterate (Hz)	1000
Settling time (ms)	0
Method	top-down
Photoresist	IP-L
Line distance XY/Z (nm)	400/2000

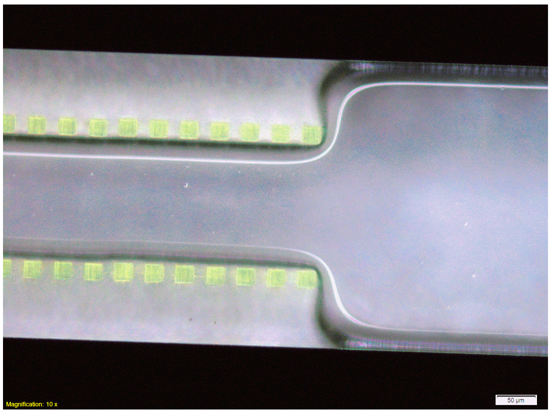
Table 4.9: Write parameters for the 10x10 μm pillar rows with crossbar supports.

the IPA meniscus to change shape.

Further testing of pillar writing inside the channels focused on creating double rows of pillars with 20 μm by 20 μm bases spaced apart with 40 μm centre to centre distance without any crossbars for support. The rows are positioned at 1/4 and 3/4 of the channel. The write parameters mentioned in table 4.9 were used for these, with the exception of powerscaling which could be turned down to 2.0 after changing the correction ring of the objective to its proper setting. Three different writing methods were tested. Because the piezo range is only 300 μm and the desired writing area was in the centimetre range the stage has to be used. There are multiple ways to do the stage extension of a design, first a method was chosen where one entire row was completed first, after which the stage moved back to the beginning of the row, move up to the starting position of the second row which would then be written. This method consistently resulted in deformed pillars at the beginning of the second row. A second method first wrote a section of a pillar row as far as the piezo range allows, then move up with the stage to write the second row partially and then move to the start of a new segment of the first row. This method was later changed to use the piezo for the upwards motion as it was still in piezo range. These latter two methods produced uniform pillars over the entire range of the centimetre long design as can be seen in figure 4.20. Total write time of this design containing 588 pillars was 60 hours, 35 minutes and 24 seconds or 6 minutes and 11 seconds per pillar.



(a) The beginning of dual pillar row with little IPA present.



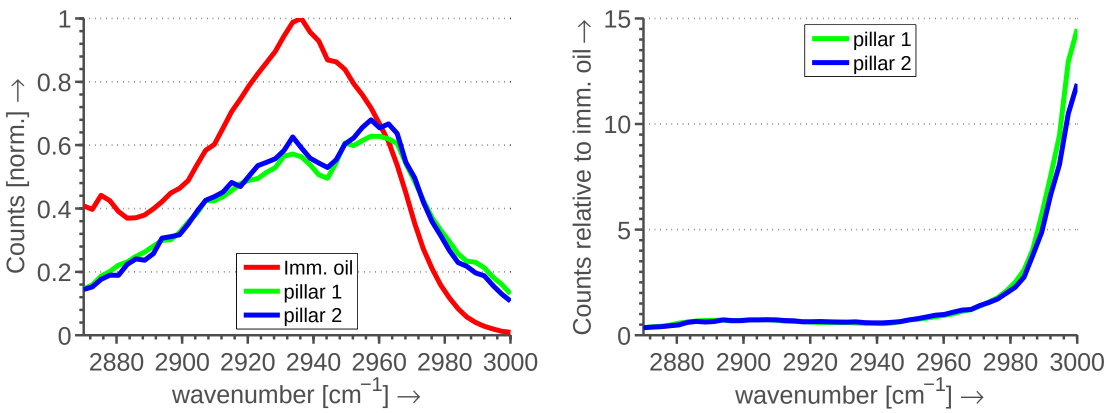
(b) End of dual pillar row with some IPA still present.

Figure 4.20: Dark field images showing a dual row of 20x20 μm pillars in a microchannel.

A side effect of the glass wafer covering the channels is that it is no longer possible to use scanning electron microscopy as we have before to characterize the structures we write. The glass would block the electrons emitted. The top down views of the pillars inside the microchannels leave the possibility open that the pillar does not reach the silicon side of the channel. Tilting the sample under a microscope was attempted but gave inconclusive results. Thus alternative characterization methods were searched for. Two confocal microscopy techniques were explored.

4.6.1 Coherent anti-Stokes Raman microscopy

Coherent anti-Stokes Raman spectroscopy or CARS is a third order non-lineair inherently confocal process involving three pulsed laser beams. Two of the pulsed laser beams are used to coherently excite molecular vibrations in the sample, the third laser beam is used as a probe beam to scan for Raman resonances. The OS group has a significant amount of experience with CARS and a setup was for a brief time available to test with. Since we are looking for polymer inside a glass and silicon channel, we scanned for molecular vibrations between 2870 cm^{-1} and 3000 cm^{-1} where we expect a broad C-H band to be present. The sample chosen to test with was the first channel we made pillars in containing just two 20 µm by 20 µm base pillars. We filled the channel with a synthetic immersion oil with a refractive index similar to the photoresist (∼1.5) to prevent optical effects.



(a) CARS signal from three different spots inside the channel.

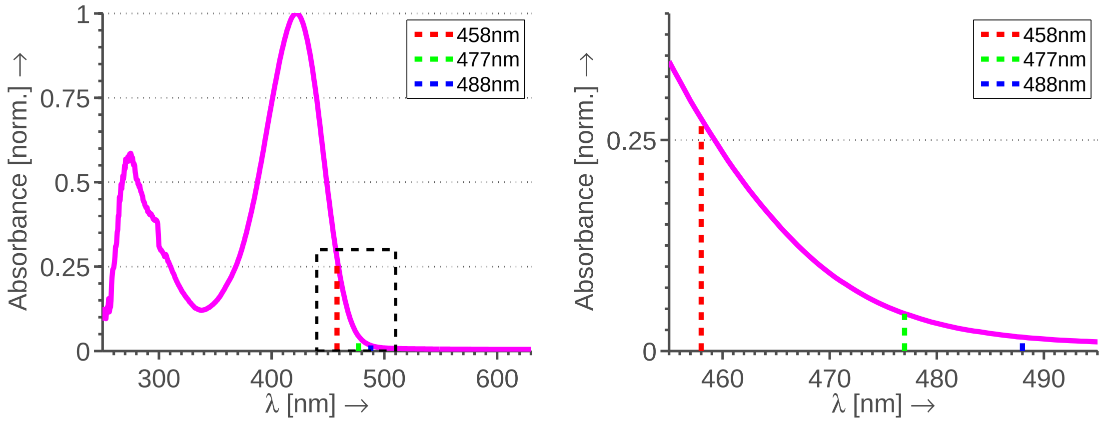
(b) Counts at pillars relative to immersion oil signal.

Figure 4.21: Graphs showing CARS measurements done in the microchannel.

As shown in figure 4.21a the synthetic immersion oil also shows a broad signal in the C-H range, which is not surprising considering the fact that it is usually made of mostly mineral oil. At 3000 cm^{-1} the polymer that the pillar structures are made of give a significantly stronger signal. These results indicate that CARS could be used to characterize polymer structures inside the microchannels.

4.6.2 Fluorescence spectroscopy

The CARS setup has very low availability however so another method was also explored. The MIRA institute at the University of Twente has a Zeiss LSM510 laser-scanning confocal microscope with multiple possible excitation wavelengths which could be used to characterize our samples using confocal fluorescence measurements. To test if the polymerized resist fluoresces first absorbance measurements were done with a Shimadzu UV-1800 spectrophotometer.



(a) The measured absorbance. **(b)** Zoomed in view of 455-495 nm range.

Figure 4.22: Graphs showing the absorbance of polymerized IP-L photoresist with LSM510 excitation wavelengths added.

The graphs in figure 4.22 show that three excitation sources available on the LSM510 (see Appendix B2) are absorbed by the photoresist. An Edinburgh Analytical Instruments spectrophotometer using a xenon lamp as an excitation source was used to test the emission of the photoresist at these wavelengths.

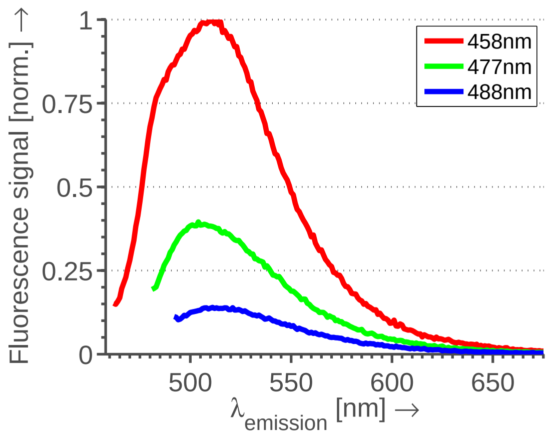
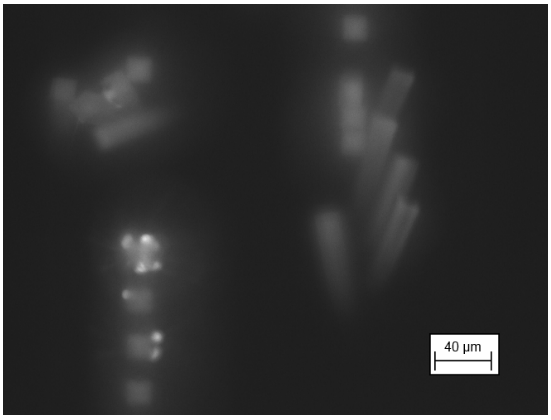


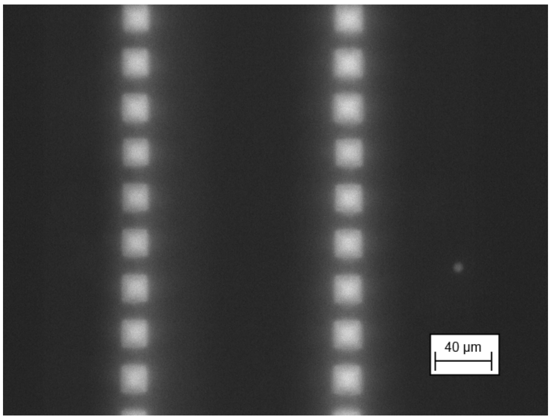
Figure 4.23: Emission spectra of IP-L photoresist at specific excitation wavelengths.

Figure 4.23 shows that the polymer fluoresces. As can be expected from the absorbance graphs using 458 nm as excitation wavelength gives the highest fluorescence signal. These results show that the LSM510 could be used to characterize polymer structures inside the microchannels and that 458 nm would be the best excitation wavelength to use.

After these measurements Jorick van 't Oever made some pictures of a microchannel with pillars written directly inside while testing with ultrasonic waves, two examples can be seen in figure 4.24. They were made with regular fluorescence microscopy. Some of the pillars in that sample failed during the development process and detached from the glass interface as can be seen in figure 4.24a.



(a) Fallen pillars inside a microchannel.

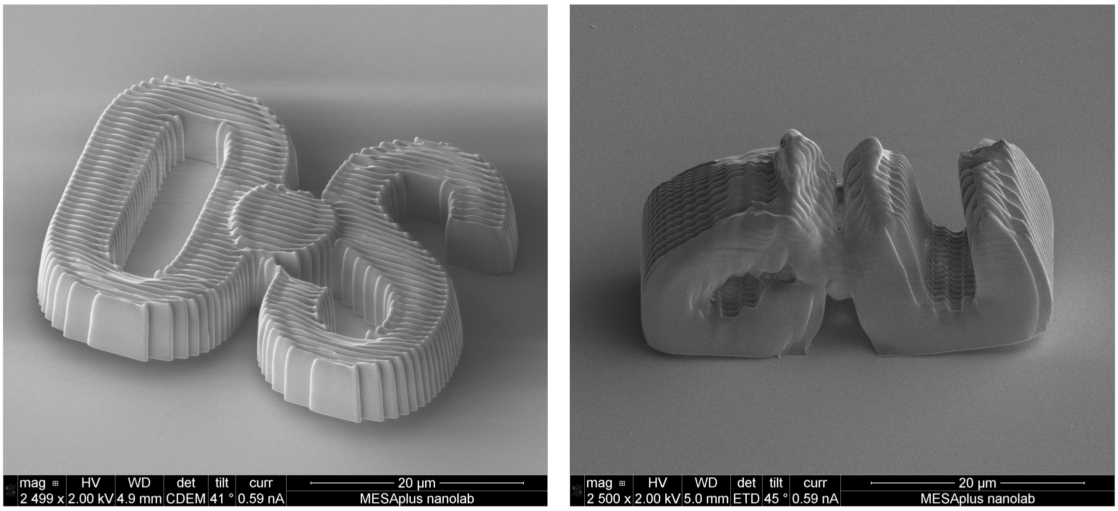


(b) Dual pillar row inside a microchannel.

Figure 4.24: Pictures of pillars inside a microchannel made using regular fluorescence microscopy.

4.7 OS logo

As a small extra one attempt was made to create the logo of the research group on a micrometer scale. The logo was first recreated in Solidworks 2012, exported to Nanoslicer and fabricated in two different orientations. A 50x magnification dark field image of it graces the cover page of this thesis.



(a) Tilted view of the OS logo written lying flat on the substrate.

(b) Tilted view of the OS logo written standing upright.

Figure 4.25: SEM pictures of the written OS logos.

Laserpower (%)	100
Powerscaling	2.0
Pointdistance (nm)	1000
Updaterate (Hz)	1000
Settling time (ms)	50
Method	bottom-up
Photoresist	IP-L
Line distance XY/Z (nm)	400/2000

Table 4.10: Write parameters for the OS logos.

As shown in figures 4.25a and 4.25b there was a significant difference between the two orientations. The version written flat on the substrate looks quite close to the design. It is approximately 41 µm wide 30 µm long and 8 µm high and took 4 minutes and 2 seconds to write. The striped pattern is caused by the high horizontal line separation value, lowering it would create a smoother surface. The upright logo was written in 6 minutes and 39 seconds and looks like it was made of ice and partially melted. The reason it looked as deformed as it does is most likely because it is solid, tall and written bottom-up, the same effect that was noticed with the tall pillar grid on a glass substrate.

Chapter 5

Conclusion and recommendations

In the course of this project a lot of experience has been gathered with using the Photonic Professional. Multiple structures were designed, scripted, fabricated and characterized. For the first time direct laser writing in microchannels was achieved. Useful results were obtained regarding the optimization of writing parameters ultimately allowing the fabrication of centimetre scale designs inside silicon glass-microchannels. Additional characterization methods for 3D polymerized structures were explored.

- The aspect ratio of written voxels increases significantly as numerical aperture of used objectives is decreased. The aspect ratio of voxels written with the 100x oil immersion objective is around 3. The aspect ratio with the 63x air objective is measured to be around 10. The aspect ratio with the 10x objective is measured to be between 20-40.
- Writing speeds can be increased by 2 orders of magnitude through optimizing structure design, increasing piezo scan speed tenfold, removing piezo settling time, increasing the voxel size by using lower NA objectives and turning off the camera in Nanowrite.
- Bars, microfilters, woodpiles, pillar fields and the OS logo were fabricated and characterized using scanning electron microscopy showing the effects of various writing parameters.
- It is possible to write large scale designs inside microchannels by using a low-viscosity photoresist that requires no pre- and post-bake steps with the Photonics Professional.
- Polymerized 3D structures can be characterized using fluorescence microscopy, spectrum measurements show coherent anti-Stokes Raman microscopy could also be used.

5.1 Recommendations

Based on gained experiences with the Nanoscribe Photonics Professional, I make these recommendations.

- One should not assume that the light from a regular light microscope does not polymerize residual photoresist inside a microchannel resulting in blocked microchannels.
- The pillar design can be optimized further by using line separation values closer to the actual voxel size. Values of 4000 nm vertical and 500 nm horizontal might be useable.
- The syringe pump setup used to fill microchannels needs optimizing to reduce leakage of resist and developer chemicals and improve the handling of samples.
- It would be interesting for the fabrication of large scale structures with the Photonic Professional to acquire an objective with a numerical aperture between 0.4 and 0.6. The 10x objective showed that lower than this is impractical unless the design allows for voxels with very high aspect ratios.
- A database of GWL files written by local users of the Photonic Professional would be useful for future users.
- Nanoslicer and Describe software should be available for users on their work desktops.

Chapter 6

Acknowledgements

This research project could not have been completed without the support of my direct supervisor Jorick van 't Oever. Without his enthusiasm and drive this project would not have come into fruition or even exist. Thanks go out to everyone in the Optical Sciences group for providing a friendly and professional environment, may you all find many cake events in your future! I would like to thank Frans Segerink for making beautiful SEM pictures using his talent for finding the most photogenic angles and positions, be it on the metre or nanometre scale. More thanks for SEM pics go out to Mark Smithers and Daniel Wijnperl. Big thanks go out to Andrew Fussell for doing CARS measurements for us, I know that setup is booked solid weeks in advance! Big thanks go to Florian Sterl for helping me do proper fluorescence measurements and for sharing his wisdom regarding Matlab scripts, graphs, Latex, Cup-a-Soup flavours and many other things.

I am thankful for the help of the MESA+ cleanroom staff, especially Huib van Vossen for his guidance regarding the various nuances of writing with the Nanoscribe and optically characterizing samples.

I would like to thank Alexander Legant from Nanoscribe for letting me borrow the 10x objective for a long time without asking any questions.

Finally I would like to thank all the people that were very helpful but I have not mentioned yet: Jennifer Herek, Herman Offerhaus, Gert Jan Koster, Himanshu Chaudhary, Erik Garbacik, Sergio Vzquez-Crdova, Lourens van Emmerik, Joris de Graaf, Carin Krijnen and Karen Munnink. Last but not least I would like to thank my family, friends and flatmates for all the support.

Bibliography

[1] Lotien Richard Huang et al. "Continuous particle separation through deterministic lateral displacement". In: *Science* 304.5673 (2004), pp. 987–990.

[2] Thomas Laurell, Filip Petersson, and Andreas Nilsson. "Chip integrated strategies for acoustic separation and manipulation of cells and particles". In: *Chemical Society Reviews* 36.3 (2007), pp. 492–506.

[3] S Melker Hagsäter et al. "Acoustic resonances in microfluidic chips: full-image micro-PIV experiments and numerical simulations". In: *Lab on a Chip* 7.10 (2007), pp. 1336–1344.

[4] Shoji Maruo, Osamu Nakamura, and Satoshi Kawata. "Three-dimensional microfabrication with two-photon-absorbed photopolymerization". In: *Optics letters* 22.2 (1997), pp. 132–134.

[5] Nanoscribe Gmbh. *Photonic Professional User Manual*. June 19, 2013.

[6] Christopher N LaFratta et al. "Multiphoton fabrication". In: *Angewandte Chemie International Edition* 46.33 (2007), pp. 6238–6258.

[7] Yihong Liu, David D Nolte, and Laura J Pyrak-Nolte. "Large-format fabrication by two-photon polymerization in SU-8". In: *Applied Physics A* 100.1 (2010), pp. 181–191.

[8] Eugene Hecht and Alfred Zajac. *Optics*. Vol. 4. Addison Wesley San Francisco, CA, 2002.

Appendix A

GWL code

A.1 Interface test

This code writes a single 50 μm line, allowing you to easily find the substrate interface by changing the stage height and writing lines.

Interfacetest.gwl:

```
Scanmode 0      % mode 0 for piezo, 1 for stagewriting
OperationMode 1 % mode 0 for pulsed, 1 for continous
ConnectpointsOn % points with distance POINTDISTANCE and at UPDATERATE are given to piezo to go to
InvertZAxis 0
PerfectshapeOff % disable for higher speed but uglier corners.
TimeStampOn
ResetInterface
Defocusfactor 0.63 % As recommended in manual for air objective writing.
XOffset 0
ZOffset 0
Settlingtime 300   % in ms the wait time between two line segments

Updaterate 1000
Pointdistance 100

PowerScaling 2.00
LaserPower 100

YOffset 0 % change this for every new line, add 6 or so um

0 0 0
50 0 0
Write
```

A.2 Marker

This code writes a 500 µm by 500 µm box, which allows you to more easily find the written structures when characterizing.

squaremarker.gwl:

```
InvertZAxis 0
PerfectshapeOff
TimeStampOn
ResetInterface
Defocusfactor 0.63 % As recommended in manual for air obj writing
Settlingtime 300   % in ms the wait time between two line segments

ScanMode 1
XOffset -250
YOffset -250
ZOffset 0 %set this right
LaserPower 100
PowerScaling 1.0

var $r2=0
for $r2 = 1 to 6
0 0 0
500 0 0
500 500 0
0 500 0
0 0 0
write
AddXOffset 1
AddYOffset 1
AddZOffset 0.5
end
write
```

A.3 Bars

This code writes an arbitrarily sized grid of bars with automatically scaling laserpower and/or piezo scan speed.

beam_grid.gwl:

```
% This code creates an array of rectangular cuboids, or bars for short.
% keep beam.gwl file in the same directory

Scanmode 0      % mode 0 for piezo, 1 for stagewriting
OperationMode 1 % mode 0 for pulsed, 1 for continous
ConnectpointsOn % points with distance POINTDISTANCE and at UPDATERATE are given to piezo to go to
InvertZAxis 0
PerfectshapeOff
TimeStampOn
ResetInterface
Defocusfactor 0.63 % As recommended in manual for air objective writing.
XOffset 0
YOffset 0
ZOffset 0
SettlingTime 50   % in ms the wait time between two line segments

% beam grid parameters
var $a = 30                 % distance between beams in um
var $beamsX = 1             % number of beams in X
var $beamsY = 5             % number of beams in Y
var $beamlengthX = 40       % size of beam side on X axis
var $beamwidthY = 20        % size of beam side on Y axis

var $footprintX = 0
var $footprintY = 0

set $footprintX = $beamsX * ($beamlengthX + $a) - $a  % total footprint in X in um
set $footprintY = $beamsY * ($beamwidthY + $a) - $a   % total footprint in Y in um

% change parameters for beams, paramater spread is applied over the set of beams written
var $powerscalbegin = 1.0     % powerscale in percentage/100 of max
var $powerscalend = 2.0       % powerscale in percentage/100 of max
var $pointdistbegin = 250     % pointdistance in nanometers
var $pointdistend = 250       % pointdistance in nanometers
var $updateratebegin = 1000   % piezo updaterate in Hz
var $updaterateend = 1000     % piezo updaterate in Hz

var $numbeams = 0
var $powerscaleach = 0
var $pointdisteach = 0
var $updaterateeach = 0

set $numbeams = $beamsX * $beamsY                                       % calculate number of beams
set $powerscaleach = ($powerscalend - $powerscalbegin) / ($numbeams - 1)    % powerscale step added per beam
set $pointdisteach = ($pointdistend - $pointdistbegin) / ($numbeams - 1)    % pointdistance step added per beam
set $updaterateeach = ($updaterateend - $updateratebegin) / ($numbeams - 1) % piezo updaterate step added per beam

% Loop variables : point coordinates
var $offx =0
var $offy =0

% Position of first beam
var $offsetx = 0
var $offsety = 0

% we set the initial pointdistance, updaterate and powerscale settings as defined above. We have to do some fiddling because we also add the per struct parameters to the first structure.
var $powerscalfirst = 0
var $pointdistfirst = 0
```

```
var $updateratefirst = 0
set $powerscalfirst =  $powerscalbegin - $powerscaleach
set $pointdistfirst = $pointdistbegin - $pointdisteach
set $updateratefirst = $updateratebegin - $updaterateeach

Pointdistance $pointdistfirst
UpdateRate $updateratefirst
PowerScaling $powerscalfirst

%These are for reporting the time and current parameter values in the log
var $beamnumber = 0
var $powerscalecurrent = 0
var $pointdistcurrent = 0
var $updateratecurrent = 0
set $powerscalecurrent = $powerscalfirst
set $pointdistcurrent = $pointdistfirst
set $updateratecurrent = $updateratefirst

% beam placement via $offx and $offy
for $offy = $offsety to $footprintY + $offsety step $a + $beamwidthY
for $offx = $offsetx to $footprintX + $offsetx step $a + $beamlengthX
include beam.gwl % the file describing a 40x20x10 um beam
end % offy
end % offx
```

Note that the beam.gwl file needs the output from Nanoslicer pasted into it before it will work.

beam.gwl:

```
% in this file we write a rectangular cuboid, or bar for short.

% we apply the parameter changes per beam with these lines:
AddPowerscaling $powerscaleach
AddPointdistance $pointdisteach
Addupdaterate $updaterateeach


%This is for reporting the time and parameter values in the log
set $beamnumber = $beamnumber + 1
set $powerscalecurrent = $powerscalecurrent + $powerscaleach
set $pointdistcurrent = $pointdistcurrent + $pointdisteach
set $updateratecurrent = $updateratecurrent + $updaterateeach

MessageOut "Starting writing beam number %d" #($beamnumber)
MessageOut "Current Powerscale: %.1f" #($powerscalecurrent)
MessageOut "Current Pointdistance: %d" #($pointdistcurrent)
MessageOut "Current Updaterate %d" #($updateratecurrent)


%we take the x and y offsets from the two for-loops.
XOffset $offx
YOffset $offy

%paste contents of sliced beam GWL file below:
```

A.4 Single width pillars

This code writes an arbitrarily sized grid of single width pillars with automatically scaling height, laserpower and/or piezo scan speed.

pillar_grid_parametersweep.gwl:

```
% This code creates a rectangular array of pillars
% keep pillar.gwl file in the same directory

Scanmode 0      % mode 0 for piezo, 1 for stagewriting
OperationMode 1 % mode 0 for pulsed, 1 for continous
ConnectpointsOn % points with distance POINTDISTANCE and at UPDATERATE are given to piezo to go to
InvertZAxis 0
PerfectshapeOff
TimeStampOn
ResetInterface
Defocusfactor 0.63 % As recommended in manual for air objective writing.
XOffset 0
YOffset 0
ZOffset 0
Settlingtime 300   % in ms the wait time between two line segments

Laserpower 100

% pillar grid parameters
var $a =20                 % distance between pillars in um
var $numX = 10             % number of pillars on X axis
var $numY = 10             % number of pillars on Y axis

%footprint calc
var $footprintX = 0
var $footprintY = 0
set $footprintX = $a*($numX-1)
set $footprintY = $a*($numY-1)

% change parameters for pillars, paramater spread is applied over the set of pillars written
var $pillarheightbegin = 1    % height of each pillar in um
var $pillarheightend = 50    % height of each pillar in um
var $powerscalbegin = 1.5     % powerscale in percentage/100 of max
var $powerscalend = 1.5       % powerscale in percentage/100 of max
var $pointdistbegin = 100     % pointdistance in nanometers
var $pointdistend = 100       % pointdistance in nanometers
var $updateratebegin = 1000   % piezo updaterate in Hz
var $updaterateend = 1000     % piezo updaterate in Hz

var $numpillars = 0
var $powerscaleach = 0
var $pointdisteach = 0
var $updaterateeach = 0
var $pillarheighteach = 0

set $numpillars = $numX*$numY               % calculate number of pillars
set $powerscaleach = ($powerscalend - $powerscalbegin) / $numpillars  % powerscale step added per pillar
set $pointdisteach = ($pointdistend - $pointdistbegin) / $numpillars  % pointdistance step added per pillar
set $updaterateeach = ($updaterateend - $updateratebegin) / $numpillars % piezo updaterate step added per pillar
set $pillarheighteach = ($pillarheightend - $pillarheightbegin) / $numpillars % height step added per pillar

% Loop variables : point coordinates
var $offx =0
var $offy =0

% Position of first pillar
var $offsetx = 0
var $offsety = 0

```

```
% we set the initial pointdistance, updaterate and powerscale settings as defined above
Pointdistance $pointdistbegin
UpdateRate $updateratebegin
PowerScaling $powerscalbegin
var $pillarheight = $pillarheightbegin

% pillar placement via $offx and $offy
for $offy = $offsety to $footprintY + $offsety step $a
for $offx = $offsetx to $footprintX + $offsetx step $a
include pillar.gwl
end % offy
end % offx
```

pillar.gwl:

```
% in this file we write a pillar.

% we apply the parameter changes per pillar with these lines:
AddPowerscaling $powerscaleach
AddPointdistance $pointdisteach
Addupdaterate $updaterateeach
set $pillarheight = $pillarheight + $pillarheighteach


%we take the x and y offsets from the two for-loops.
XOffset $offx
YOffset $offy

%we write the pillar with predesignated height, upwards.
0       0       0
0       0       $pillarheight
write
```

A.5 Woodpiles

This code writes a woodpile structure with arbitrary size, layer and line distances.

woodpile.gwl:

```
% This code creates a woodpile structure
% keep xlines and ylines files in the same directory


Scanmode 0      % mode 0 for piezo, 1 for stagewriting
OperationMode 1 % mode 0 for pulsed, 1 for continous
ConnectpointsOn % points with distance POINTDISTANCE and at UPDATERATE are given to piezo to go to
InvertZAxis 0
PerfectshapeOff
TimeStampOn
ResetInterface
Defocusfactor 0.63 % As recommended in manual for air objective writing.
XOffset 0
YOffset 0
ZOffset 0
Settlingtime 300   % in ms the wait time between two line segments
Pointdistance 100 % in nanometers
UpdateRate 1000   % in Hz
PowerScaling 1.5
LaserPower 100     % in percentage of max

% woodpile parameters
var $Xdist =2           % X distance between lines in um
var $Ydist =2           % Y distance between lines in um
var $Zdist = 4.5          % Z distance between woodpile layers in um
var $Xlength = 30       % footprint in X in um
var $Ylength = 30       % footprint in Y in um
var $pileheight = 18    % height of each pillar in um

% defining some vars
var $Zdisttwo = 0
set $Zdisttwo = 2 * $Zdist

% Loop variables : point coordinates
var $offx =0
var $offy =0
var $offz =0

% woodpile placement
for $offz = 0 to $pileheight step $Zdisttwo % loop below over Z

for $offy = $Ydist to $Ylength - $Ydist step $Ydist % create layer of lines in X direction
include Xlines.gwl
end % offy

set $offy=0
for $offx = $Xdist to $Xlength - $Xdist step $Xdist % create layer of lines in Y direction
include Ylines.gwl
end % offx
set $offx=0

end %$offz

set $offx=0
set $offy=0
set $offz=0
```

xlines.gwl:

```
% We use this file to create our X lines


%we take the offsets from the two for-loops.
XOffset $offx
YOffset $offy
ZOffset $offz

%we write the lines
0       0       0
$Xlength       0       0
write
```

ylines.gwl:

```
% We use this file to create our Y lines


%we take the offsets from the two for-loops.
XOffset $offx
YOffset $offy
ZOffset $offz

%we write the lines
0       0       $Zdist
0       $Ylength       $Zdist
write
```

A.6 Microfilters

Below are two methods for making microfilter structures with arbitrary structure and pore size.

Supported pillar method:

woodpile.gwl:

```
% This code creates an array of pillars with supports
% keep spillarvert.gwl and support.gwl files in the same directory

Scanmode 0      % mode 0 for piezo, 1 for stagewriting
OperationMode 1 % mode 0 for pulsed, 1 for continous
ConnectpointsOn % points with distance POINTDISTANCE and at UPDATERATE are given to piezo to go to
InvertZAxis 0
PerfectshapeOff
TimeStampOn
ResetInterface
Defocusfactor 0.63 % As recommended in manual for air objective writing.
XOffset 0
YOffset 0
ZOffset 0
Settlingtime 300   % in ms the wait time between two line segments
Pointdistance 100 % in nanometers
UpdateRate 1000   % in Hz
PowerScaling 1.0
LaserPower 100     % in percentage of max

% pillar grid parameters
var $a =2                 % X distance between pillars in um
var $b =2                 % Y distance between pillars in um
var $c =10                % Z distance between supportlayers in um
var $footprintX = 30      % footprint in X in um
var $footprintY = 30      % footprint in Y in um
var $pillarheight = 50    % height of each pillar in um

% Loop variables : point coordinates
var $offx =0
var $offy =0
var $offz =0

% Defining some vars
var $supportX = 0
var $supportY = 0
var $pillarZ = 0

% Position of first pillar
var $offsetx = 0
var $offsety = 0

%height first support
var $offsetz = 0

% pillar placement via $offx and $offy
set $pillarZ = $Pillarheight
for $offy = $offsety to $footprintY + $offsety step $b
for $offx = $offsetx to $footprintX + $offsetx step $a
include lines.gwl
end % offy
end % offx

% support placement X direction

set $offx =0
set $offy =0
set $offz =0
set $supportX = $footprintX
set $supportY = 0
set $pillarZ = 0

```

```
for $offz = $offsetz to $pillarheight + $offsetz step $c
for $offy = $offsety to $footprinty + $offsety step $b
include lines.gwl
end % offz
end % offy

% support placement Y direction

set $offx =0
set $offy =0
set $offz =0
set $supportX = 0
set $supportY = $footprintY
set $pillarZ = 0

for $offz = $offsetz to $pillarheight + $offsetz step $c
for $offx = $offsetx to $footprintx + $offsetx step $a
include lines.gwl
end % offz
end % offx
```

lines.gwl:

```
% We use this file to create our lines, from the pillars to the supports.

%we take the offsets from the two for-loops.
XOffset $offx
YOffset $offy
ZOffset $offz

%we write the lines, $pillarZ is in the bottom row to write bottoms-up
0       0       0
$supportX       $supportY       $pillarZ
write
```

repeating unit cell method:

improved_spillar_grid.gwl:

```
% This code creates an array of pillars with supports
% keep repeatable_struct, groundgrix and first_row files in the same directory

Scanmode 0      % mode 0 for piezo, 1 for stagewriting
OperationMode 1 % mode 0 for pulsed, 1 for continous
ConnectpointsOn % points with distance POINTDISTANCE and at UPDATERATE are given to piezo to go to
InvertZAxis 0
PerfectshapeOff
TimeStampOn
ResetInterface
Defocusfactor 0.63 % As recommended in manual for air objective writing.
XOffset 0
YOffset 0
ZOffset 0
Settlingtime 300   % in ms the wait time between two line segments
Pointdistance 100 % in nanometers
UpdateRate 1000   % in Hz
PowerScaling 1.0
LaserPower 100     % in percentage of max

% pillar grid parameters
var $a =1                  % X distance between pillars in um
var $b =1                  % Y distance between pillars in um
var $c =10                 % Z distance between supportlayers in um
var $footprintX = 30       % footprint in X in um
var $footprintY = 30       % footprint in Y in um
var $structheight = 50     % total height the structure

% Loop variables : point coordinates
var $offx =0
var $offy =0
var $offz =0

% Here we define and write the structure with for loops:

%first we make the bottom grid attached to the interface
for $offy = 0 to $footprintY step $b
include groundgridX.gwl
end % offy

set $offy=0
for $offx = 0 to $footprintX step $a
include groundgridY.gwl
end % offx

%Here we loop the entire base layer structure over Z untill we reach desired height:
for $offz = 0 to $structheight-$c step $c

%here we start the support structure in X
XOffset 0
YOffset 0
ZOffset $offz

%we write the first pillar
0       0       0
0       0       $c
write
%we write the second pillar
0       $b       0
0       $b       $c
write
%we write the support bar
0       0       $c
0       $b       $c
write

%we fill the X support row
set $offx =0
```

```
set $offy =0

for $offx = 0 to $footprintX-$a step $a
include first_rowX.gwl
end % offx

%here we make the Y support row
set $offx =0
set $offy =0

for $offy = $b to $footprintY-$b step $b
include first_rowY.gwl
end % offy


%here we fill the remaining layer with the repeating structure:
set $offx =0
set $offy =0

for $offy = $b to $footprintY-$b step $b
for $offx = $a to $footprintX-$a step $a
include repeatable_struct.gwl
end % offy
end % offx

end % offz: this is the end of the Z loop
```

first_rowX.gwl:

```
% We use this file to create our first row of supported pillars

%we take the offsets from the two for-loops.
XOffset $offx
YOffset $offy
ZOffset $offz

%we write the first pillar
$a       0       0
$a       0       $c
write
%we write the second pillar
$a       $b       0
$a       $b       $c
write
%we write the support bars
0       $b       $c
$a       $b       $c
$a       0       $c
0       0       $c
write
```

first_rowY.gwl:

```
% We use this file to create our first row of supported pillars

%we take the offsets from the two for-loops.
XOffset $offx
YOffset $offy
ZOffset $offz

%we write the first pillar
0        $b        0
0        $b        $c
write
%we write the second pillar
$a        $b        0
$a        $b        $c
write
%we write the support bars
$a        0        $c
$a        $b        $c
0        $b        $c
0        0        $c
write
```

groundgridX.gwl:

```
%We use this file to create the X part of our bottom support grid in XY plane.

%we take the offsets from the two for-loops.
XOffset $offx
YOffset $offy
ZOffset $offz

%we write a line
0        0        0
$footprintX        0        0
write
```

groundgridY.gwl:

```
%We use this file to create the Y part of our bottom support grid in XY plane.

%we take the offsets from the two for-loops.
XOffset $offx
YOffset $offy
ZOffset $offz

%we write a line
0        0        0
0        $footprintY        0
write
```

repeatable_struct.gwl:

```
%We use this file to create our repeatable row of supported pillars

%we take the offsets from the two for-loops.
XOffset $offx
YOffset $offy
ZOffset $offz

%we write the pillar
$a        $b        0
$a        $b        $c
write
%we write the X and Y support bars
0        $b        $c
$a        $b        $c
$a        0        $c
write
```

A.7 Microchannel pillar rows

Below are two methods for making dual pillar rows in microchannels with arbitrary spacing and length.

Method without crossbars:

bigpillar_grid_stage_om_en_om_MOREPIEZO.gwl:

```
% This code creates a double row of big pillars
% keep bigpillar.gwl and or biggerpillar.gwl in the same directory

Scanmode 0      % mode 0 for piezo, 1 for stagewriting
OperationMode 1 % mode 0 for pulsed, 1 for continous
ConnectpointsOn % points with distance POINTDISTANCE and at UPDATERATE are given to piezo to
    go to
InvertZAxis 0   % may want to invert z axis if sliced bottomup and writing high struct (p96)
PerfectshapeOff % disable for higher speed but uglier corners.
TimeStampOn
ResetInterface
Defocusfactor 0.63 % As recommended in manual for air objective writing.
XOffset 0
YOffset 0
ZOffset 0
Settlingtime 0   % in ms the wait time between two line segments
Pointdistance 1000 % in nanometers
UpdateRate 1000   % in Hz
PowerScaling 2.0


% pillar grid parameters

var $a = 20               % distance between pillars in um wall to wall
var $footprintX = 280      % max footprint in X in um
var $footprintY = 20      % max footprint in Y in um
var $pillarlengthX = 20     % size of pillar side on X axis
var $pillarwidthY = 20      % size of pillar side on Y axis
var $channelwidth = 377     % Y width of microchannel in um
var $walldist = 0 % creating the var of distance from microchannel wall to center of pillar
set $walldist = $channelwidth / 4 % distance from microchannel wall to center of pillar
var $walloffset = -6 % change the initial distance moved from the wall with this. SHOULD be
    0... could be -6

% added stage stuff

var $stagemultX = 2  % total grid X length will be $footprintX * $stagemult
var $stage = 1

% channel width calcs and so on

var $moveuprowone = 0
set $moveuprowone = $walldist-($pillarwidthY/2)+$walloffset

var $rowtwoyoffset = 0
set $rowtwoyoffset = $channelwidth - 2 * $walldist

% Loop variables : point coordinates
var $offx =0
var $offy =0
var $offz =0


% Position of first pillar
var $offsetx = 0
var $offsety = 0
var $offsetz = 0


%FIRST ROW

% use stage to move up from wall edge to distance you want pillar row
```

```
MoveStageY $moveuprowone

%loop voor stage multiplier
for $stage = 1 to $stagemultX step 1

% pillar placement via $offx and $offy
for $offy = $offsety to $footprintY + $offsety - $pillarwidthY step $a + $pillarwidthY
for $offx = $offsetx to $footprintX + $offsetx - $pillarlengthX step $a + $pillarlengthX
include bigpillar.gwl
end % offy
end % offx

%SECOND ROW

set $offsetx = 0
set $offsety = $rowtwoyoffset
set $offsetz = 0


% pillar placement via $offx and $offy
for $offy = $offsety to $footprintY + $offsety - $pillarwidthY step $a + $pillarwidthY
for $offx = $offsetx to $footprintX + $offsetx - $pillarlengthX step $a + $pillarlengthX
include bigpillar.gwl
end % offy
end % offx

set $offsetx = 0
set $offsety = 0
set $offsetz = 0


MoveStageX $footprintX

end %stage loop

%return back to origin
var $moveback = 0
set $moveback = -($stagemultX * $footprintX)
MoveStageX $moveback
PiezoGotoX 0
PiezoGotoY 0

%finished
```

bigpillar.gwl (biggerpillar.gwl is the same but with different sliced pillar):

```
% in this file we write a grid of big pillars

%we take the x and y offsets from the two for-loops.
XOffset $offx
YOffset $offy
ZOffset $offz

%paste contents of sliced pillar GWL file below:
```

Method with crossbars:

Dual_row_smallpillars_with_crossbars.gwl:

```
% This code creates two rows of 10x10 pillars with crossbar support
% keep smallpillar.gwl and crossbar.gwl in the same directory

Scanmode 0      % mode 0 for piezo, 1 for stagewriting
OperationMode 1 % mode 0 for pulsed, 1 for continous
ConnectpointsOn % points with distance POINTDISTANCE and at UPDATERATE are given to piezo to go to
InvertZAxis 0
PerfectshapeOff
TimeStampOn
ResetInterface
Defocusfactor 0.63 % As recommended in manual for air objective writing.
XOffset 0
YOffset 0
ZOffset 0
Settlingtime 0   % in ms the wait time between two line segments
Pointdistance 1000 % in nanometers
UpdateRate 1000   % in Hz
PowerScaling 2.0


% pillar grid parameters

var $a = 10                % distance between pillars in um
var $footprintX = 200      % footprint of repeating struct (note that there is always a pillar at the beginning)
var $footprintY = 10      % max footprint in Y in um
var $pillarlengthX = 10     % size of pillar side on X axis
var $pillarwidthY = 10      % size of pillar side on Y axis
var $pillarheight = 170    % size of pillar on Z axis
var $channelwidth = 377    % Y width of microchannel in um
var $walldist = 0 % distance from microchannel wall to center of pillar
set $walldist = $channelwidth / 4 % distance from microchannel wall to center of pillar

% added stage stuff

var $stagemultX = 5  % total grid X length will be (($footprintX * $stagemult) + $a + $pillarlengthX
var $stage = 1

% channel position calcs and variable setting
var $movestageeach = 0
set $movestageeach = $footprintX

var $moveuprowone = 0
set $moveuprowone = $walldist-($pillarwidthY/2)

var $moveuprowtwo = 0
set $moveuprowtwo = $channelwidth - (2 * $walldist)

% here we define our crossbars positions and a var
var $crossbardist = 20 % distance between crossbars
var $crossbarstart = 10 % height at which first crossbar is made

var $crossbarXoffset = 0

var $crossbarX = 0
set $crossbarX = $a+$pillarlengthX
var $crossbarY1 = 0
set $crossbarY1 = $pillarwidthY/2 - 0.800
var $crossbarY2 = 0
set $crossbarY2 = $pillarwidthY/2 - 0.400
var $crossbarY3 = 0
set $crossbarY3 = $pillarwidthY/2
var $crossbarY4 = 0
set $crossbarY4 = $pillarwidthY/2 + 0.400
var $crossbarY5 = 0
set $crossbarY5 = $pillarwidthY/2 + 0.800

```

```
% Loop variables : point coordinates
var $offx =0
var $offy =0
var $offz =0


% Position of first pillar
var $offsetx = 0
var $offsety = 0
var $offsetz = 0

%FIRST ROW

% use stage to move up from wall edge to distance you want pillar row
MoveStageY $moveuprowone


% make an initial pillar
include smallpillar.gwl
var $movefirstX = 0
set $movefirstX = $pillarlengthX / 2
MoveStageX $movefirstX

%loop voor stage multiplier
for $stage = 1 to $stagemultX step 1

% pillar placement via $offx and $offy loops
for $offy = $offsety to $footprintY + $offsety - $pillarwidthY step $a + $pillarwidthY
for $offx = $offsetx+$a+$movefirstX to $footprintX + $offsetx +$a step $a + $pillarlengthX
include smallpillar.gwl


%Z loop voor crossbars
for $offz = $crossbarstart to $pillarheight step $crossbardist
include crossbar.gwl
end % offz
set $offz = 0

end % offy
end % offx

set $offx =0
set $offy =0
set $offz =0

set $offsetx = 0
set $offsety = 0
set $offsetz = 0

MoveStageX $movestageeach

end %stage loop

%return back to origin
var $moveback = 0
set $moveback = -$movefirstX - ($movestageeach * $stagemultX)
MoveStageX $moveback
PiezoGotoX 0
PiezoGotoY 0

%SECOND ROW

% use stage to move up from wall edge to distance you want pillar row
MoveStageY $moveuprowtwo

% make an initial pillar
include smallpillar.gwl

set $movefirstX = $pillarlengthX / 2
MoveStageX $movefirstX

%loop voor stage multiplier
for $stage = 1 to $stagemultX step 1
```

```

% pillar placement via $offx and $offy loops
for $offy = $offsety to $footprintY + $offsety - $pillarwidthY step $a + $pillarwidthY
for $offx = $offsetx+$a+$movefirstX to $footprintX + $offsetx +$a step $a + $pillarlengthX
include smallpillar.gwl


%Z loop voor crossbars
for $offz = $crossbarstart to $pillarheight step $crossbardist
include crossbar.gwl
end % offz
set $offz = 0

end % offy
end % offx

set $offx =0
set $offy =0
set $offz =0

set $offsetx = 0
set $offsety = 0
set $offsetz = 0

MoveStageX $movestageeach

end %stage loop

%return back to origin
MoveStageX $moveback
PiezoGotoX 0
PiezoGotoY 0

%finished
```

crossbar.gwl:

```
% in this file we write a crossbar

%adjusting the Xoffset for the crossbars
set $crossbarXoffset = $offx - $a - $pillarlengthX / 2

%we take the x and y offsets from the two for-loops.
XOffset $crossbarXoffset
YOffset $offy
ZOffset $offz

Laserpower 100

0 $crossbarY1 0
$crossbarX $crossbarY1 0
write

$crossbarX $crossbarY2 0
0 $crossbarY2 0
write

0 $crossbarY3 0
$crossbarX $crossbarY3 0
write

$crossbarX $crossbarY4 0
0 $crossbarY4 0
write

0 $crossbarY5 0
$crossbarX $crossbarY5 0
write
```

smallpillar.gwl:

```
% in this file we write a grid of small pillars

%we take the x and y offsets from the two for-loops.
XOffset $offx
YOffset $offy
ZOffset $offsetz

%paste contents of pillar GWL file below:
```

Appendix B

Fluorescence measurement notes

The fluorescence measurements taken with the Edinburgh Analytical Instruments spectrophotometer have been corrected for the monochromator efficiency. An excerpt of the manual showing this efficiency can be found in figure B.1.

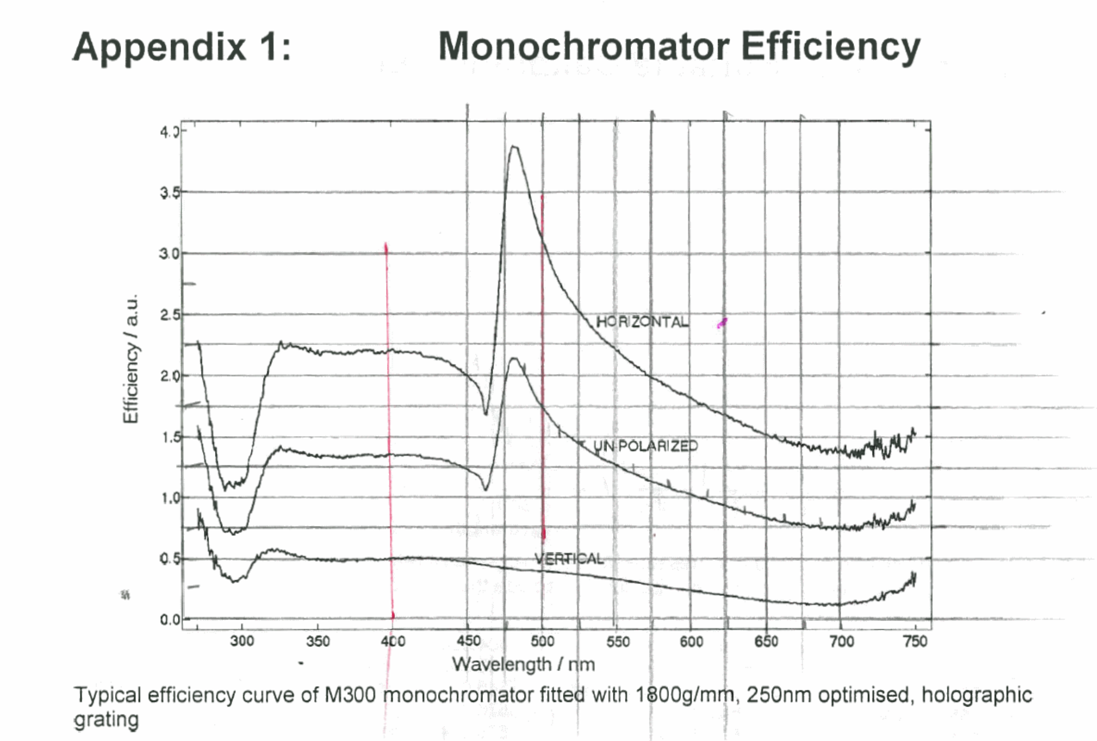


Figure B.1: Graph showing the monochromator efficiency.

The specifications of the light path hardware of the Zeiss LSM510 mentioned in chapter 4.6.2 can be found on the next page.

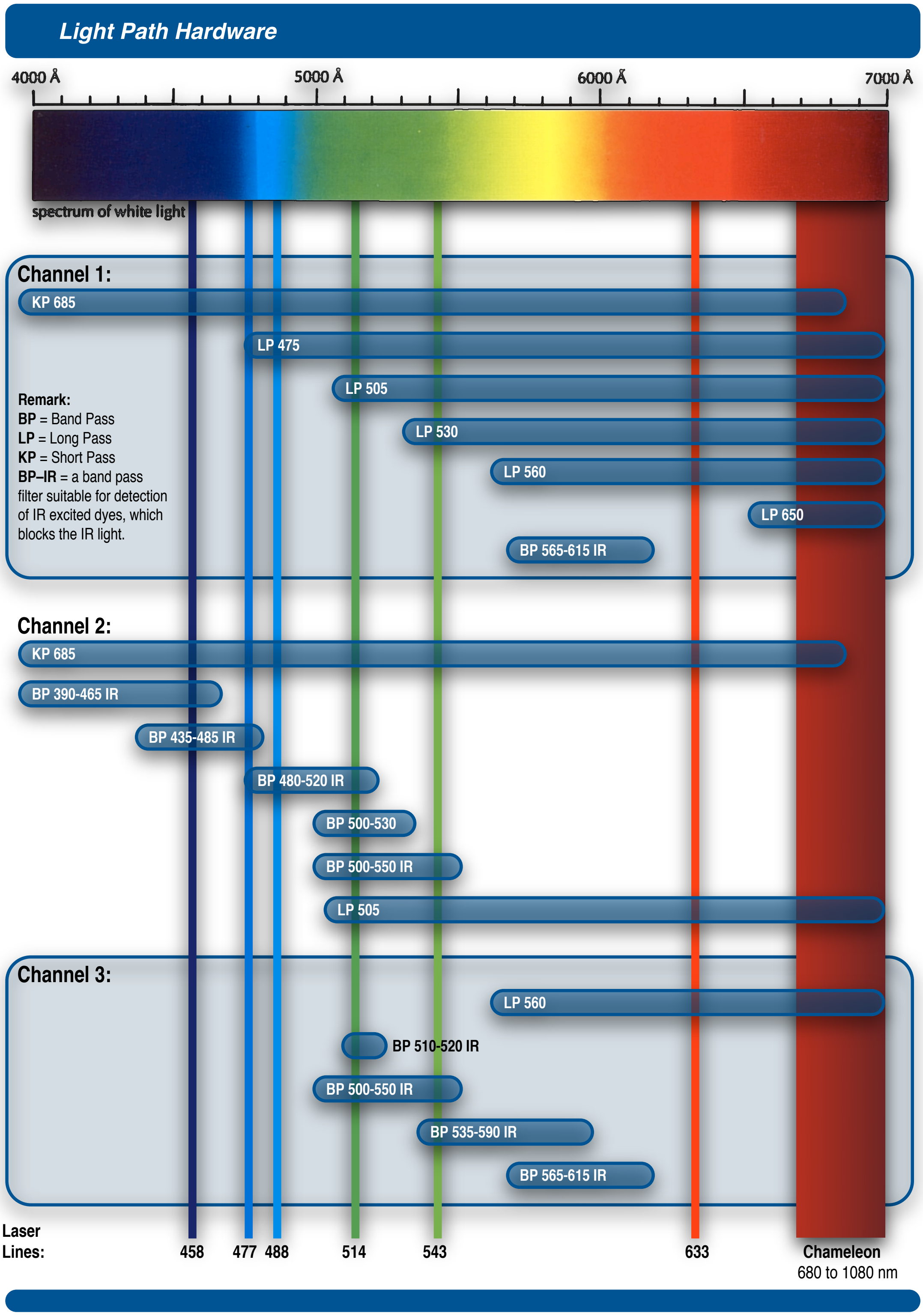


Figure B.2: Picture showing the light path hardware of the Zeiss LSM510.

Appendix C

Dual pillar row in microchannel recipe

Below is a recipe for writing dual pillar rows inside microchannels, written to ease future attempts at making them by experienced users.

Use the "bigpillar_grid_stage_om_en_om_MOREPIEZO.gwl", "bigpillar.gwl" and "biggerpillar.gwl" files and place them in the same folder. The "biggerpillar.gwl" file contains a version of the pillar design with larger line separation values. Determine how long you want the pillar rows to be, divide that by 280 μm and use the resulting number for the $stagemultX variable. Note that when using "bigpillar.gwl" as the pillar source file, each pillar takes about 6 minutes to write. 600 pillars is doable in a weekend. With the wall to wall pillar distance of 20 μm set as variable $a, the number of pillars written is the $stagemultX variable multiplied by 14.

Use a syringe, nipple and a tube to put a small amount of IP-L photoresist inside the microchannel, take care not to introduce air bubbles. After cleaning residual resist glue the channel onto the center of the rectangular part of the DILL substrate holder. Make sure the correction ring of the 63x objective is set to the correct value (500 μm). Wait 5 minutes for the glue to dry and put the holder in the Photonics Professional.

Approach the sample and note the height. Apply tilt correction, normal values for the tilt are $X \leq 1$ degrees, $Y \leq 0.1$ degrees. After tilt correction do alignment rotation. Move the stage ∼350 μm up, find the channel and move to the far left side of it. Centre on the horizontal edge, set this coordinate as Marker 1: 0 0 0. Move to the far right of the channel and centre on the same horizontal edge, set this coordinate as Marker 2: 1 0 0. Change the "Apply to:" setting to Stage only, because unless you are using a fixed 1.7.3 build of Nanowrite, the piezo rotation is broken. Enable rotation. Move to the centre of the channel, then to the bottom horizontal edge, manually move the stage 6 μm down. Load the "bigpillar_grid_stage_om_en_om_MOREPIEZO.gwl" file and abort after a few seconds, otherwise the machine will take a long time to calculate write time which you have estimated already. Press write.

After writing remove the channel from the Photonics Professional. Place the microchannel in the syringe pump setup and flush the channel with 20 ml of RER600 at a rate of 20 ml/hr. Afterwards flush with 20 ml of IPA at a rate of 20 ml/hr. Dry with a gentle flow of nitrogen.