Distortion in Conformable Masks for Evanescent Near Field Optical Lithography

---

A thesis submitted in partial fulfilment of the

requirements for the Degree of

Master of Engineering

in Electrical and Computer Engineering

by A. J. Wright B.E. (Hons)

University of Canterbury,

Christchurch, New Zealand.

March 2007

Abstract

In this thesis the in-plane pattern distortion resulting from the use of Evanescent Near Field Optical Lithography (ENFOL) masks was investigated. ENFOL is a high resolution low-cost technique of lithography that is able to pattern features beyond the diffraction limit of light. Due to its use of the evanescent near field, ENFOL requires the use of conformable masks for intimate contact. Such masks can stretch and skew as they come into contact with silicon substrates and therefore distort the high resolution features patterned on them. It was desired to measure this distortion to ascertain the patterning performance of ENFOL masks and possibly correct for any uniform distortion found.

To this end a sophisticated measuring process was successfully demonstrated. This involved the use of a Raith 150 Electron Beam Lithography (EBL) system with precision laser interferometer stage and metrology software module for automated measurements. Custom software was written for the Raith to enable it to take additional measurements to compensate for electron beam drift. Processing algorithms were then employed to using the measurements to compensate for beam drift and correcting for shift and rotation systematic errors. The performance of the in-plane distortion measuring process was found to have a precision of 60nm.

With the ability to measure distortion, ENFOL masks were used to pattern substrates and distortion was found to be large, on the order of 1μm. This is much larger than desired for sub 100nm patterning as is expected of ENFOL. The distortions were non-uniform patterns of localised displacements. This, the observation of Newton's rings beneath a test mask and the observation of a single particle distortion across measurements of the same mask across different loadings in the EBL pointed to particulate contamination causing the distortion.

In order to prove beyond doubt that particulate contamination was the cause of the spurious distortions, mechanical modelling using the Finite Element Method (FEM) of analysis was employed. The results from this matched the distortions observed experimentally, with particles 20-40μm modelling the observed distortion.

Acknowledgements

I would like to express my sincere gratitude toward my supervisor, Professor Richard Blaikie, for his unfailing optimism, intelligent guidance, always being encouraging and for taking me on as one of your masters students. I would like to thank Helen Devereux for training me so many machines and processes, for helping me source necessary materials from photoresist to the crucial glass membranes for my masks. I sought your advice on many an occasion and enjoyed going out and taking photos with you. Thank you Gary Turner for training me upon the use of the electron beam lithography system, and for many questions I asked of you during my stay in the laboratory. I would also like to thank David Melville for teaching me processes in the lab, uses of equipment and for helpful discussion. A big thank you to my good friend Chun Hong Yoon, for your much appreciated help with ideas, Matlab expertise and time.

For technical or training support I would like to thank the following staff or students: James Oxnam of the information technology department for providing training courses, one on EndNote software, a tool which proved very helpful; Erwin Berthier for taking the time to show me how to make, bake and use PDMS silicone rubber and for also allowing me the use of your own supply; Robert McGregor at the glassblowing workshop in the Chemistry department for preparing my mask holder plates; Brandon Hutchison and Alvaro Gonzalez of the civil engineering department, for respectively setting up the Abaqus license server and for lending me the Abaqus installation disks; Florin Predan, Mike Shurety, Pieter Kikstra and Dave van Leeuwen for computer support; and Mike Cusdin for the lease on my exquisite cubicle.

Thank you so much to my parents, Dad, Mum and di Tam for your unwilting support, optimism, guidance and love. I would not have been able to do this without you. I would like to thank my brother David Wright for inspiring me with his achievements and for his support and love. I would like to thank my good and loyal friend William Lee for help at the very beginning of this thesis on how to find Journal articles – to the end with invaluable support. Thank you to Li (Judy) Zhou for being my good friend,

and for always being so cheerful and uplifting to be around. Thank you to Chuck for your caring support running into (very) late evenings keeping me company as I worked. Thank you to my good friend Caspar Harmer for your kind support and for being such a positive influence on me. Thank you to my good friend and confidant Edward Sickmann, you are an inspiration to me. I would like thank Mai Abdelhalim for your love, support and for believing in me.

I'd like to acknowledge the welcome support I received from my friends Seung Ho Choi, Sanitta Thongpang, Ping Heng (Wallace) Kuo, Samara Alzaidi, Allan Chin, Ping Yu (Patrick) Kuo, Han Hao (Elliot) Cheng, Mark Noonchester, Hugo Vincent, Shehan Perera, Julian Maclaren, Simon and Margaret Tupuhi.

Table of Contents

Acknowledgments iii
List of Figures ix

Chapter 1: Introduction 1
1.1 The Development of the Modern Computer 1
1.2 Photolithography: the main tool for the manufacturing of integrated circuits 5
1.3 Possible Next Generation Lithographies 7
1.3.1 Extreme Ultraviolet Lithography (EUVL) 7
1.3.2 Electron Projection Lithography 9
1.3.3 Nanoimprint Lithography 10
1.3.4 Evanescent near field optical lithography 12
1.4 Aim of this thesis 12
1.5 Thesis Arrangement 13

Chapter 2: Background and Literature Review 15
2.1 Evanescent near field optical lithography 15
2.2 Conformable Contact Masks for ENFOL 17
2.2.1 Protruding Absorber Masks 18
2.2.2 Embedded Amplitude Masks 22
2.2.3 Light Coupling Masks 24
2.3 Extending ENFOL 25
2.3.1 Surface plasmon illumination 25
2.3.2 Planar Lens Lithography 26

Chapter 3: Experimental Techniques: Apparatus, Processes and Manufacturing 29
3.1 Essential Apparatus 29
3.1.1 Raith 150 Electron Beam Lithography System 30
3.1.2 The Mask Aligner for Photolithography 36
3.1.3 The RIE 37

3.1.4 The Sputterer 39
3.2 Processes 40
3.2.1 Manufacture of ENFOL Conformable Masks 40
3.2.2 Silicon Substrate Processing and Patterning 47

Chapter 4: Data Capture and Analysis 53
4.1 Data Capture 53
4.1.1 The Measuring Process, Scan Parameters and Command File Generation 54
4.1.2 The Threshold Algorithm 58
4.2 Data Processing 64
4.2.1 Raw Data Processing and Marker Feature Centres 65
4.2.2 Compensating for Electron Beam Drift 66
4.2.3 Removal of Defective Markers 68
4.2.4 Correcting for Scan Shift and Rotation Error 70
4.2.5 Results: Statistical Information and Maps of Distortion 74

Chapter 5: Experimental Results 77
5.1 Measuring Accuracy 77
5.1.1 Effect of Particulate Contamination 79
5.2 Initial ENFOL Mask Patterning Distortion 81
5.3 PDMS Mask Mounts 83
5.3.1 The First PDMS Mask Mounts 84
5.3.2 The Embedded PDMS Mesa Mask 86
5.4 Original Mask Mount with 50μm Separation 91
5.5 Summary 95

Chapter 6: Mechanical Analysis 97
6.1 Finite Element Method Analysis Using Abaqus/CAE 97
6.1.1 Abaqus FEM Analysis Software 98
6.1.2 Assumptions 98
6.1.3 Creating a System of Units within Abaqus/CAE 100
6.1.4 3-Space, 2-Space and Axisymmetric 103
6.1.5 Model Creation Using Abaqus/CAE 105
6.2 Simulation Results 109

6.2.1 Distortion effects from singular dust particles..109
6.2.2 Residual distortion in 'x' of membrane due to friction............................112
6.2.3 Usable Mask Surface Area...113

Chapter 7: Conclusions and Future Work... 117

References.. 97

Appendix A... 129

Appendix B... 141

Appendix C... 149

Appendix D... 155

List of Figures

Figure 1.1 The central air data computer for the F14 jet fighter aircraft, the first computer built using large scale integration. The modules are: ROM read only memory, RAS random access storage, SLU special logic unit, PMU programmable multiplier unit, PDU parallel divider unit and SLF special logic function. The final processor used: 1-PMU, 1-PDU, 1-SLF, 3RAS's, 2SLU's and 19-ROM's [4]................ 3

Figure 1.2 The Intel 4004 microprocessor, the worlds first computer constructed as a single IC using large scale integration, and the basis for all future processors [6]... 4

Figure 1.3 Projection optical lithography [7]. UV light illuminates a photoresist covered silicon substrate through a mask and lenses. 5

Figure 1.4 Pattern transfer processes [7] 6

Figure 1.5 Diffraction: Red light passing through a gap half its wavelength is significantly diffracted (left) whereas blue light with a shorter wavelength through the same size gap is not (right) [8]................................. 6

Figure 1.6 An extreme ultraviolet lithography system [9]. The mirror optics are shown in blue with the mirror mask shown in grey at the top................... 8

Figure 1.7 Results from using EUVL [11] 9

Figure 1.8 Nikon electron beam projection lithography system NSR-EB1A. The wide beams can be seen which are projected through the reticle/mask as it is moved beneath it exposing the whole mask pattern onto the wafer beneath [12]. 10

Figure 1.9 Highest resolution NIL results to date, 5nm half pitch [14]........................... 11

Figure 1.10 NIL process using a 3D mold (a) substrate coated in thermo plastic polymer. (b) imprinting of mold into substrate under heat and pressure . (c) mold removed after cooling and imprint remains in substrate [15].. 11

Figure 2.1 Ordinary propagating light and evanescent near field light [16]. Ordinary light propagates, while near field light decays rapidly................... 15

Figure 2.2 Schematic diagram depicting the ENFOL process[7]. A conformable mask is held in intimate contact with a thin photoresist coated substrate, and illuminated with UV light. 16

Figure 2.3 Conformable masks[26]: (a) protruding absorber mask (b) embedded amplitude mask (c) light coupling mask. 17

Figure 2.4 (a) Silicon nitride mask with amorphous silicon absorbers. (b) 64nm period features produced using ENFOL. The features were patterned using a chemically amplified resist [30]. 18

Figure 2.5 Exposure depth (half contrast) as a function of grating period [17]................ 19

Figure 2.6 80nm half pitch features patterned in chemically amplified resist a) and AZ DNQ-novolac (standart) i-line resist b) [16]..................................... 20

Figure 2.7 Molecular weight distribution of the CAP resist (a) and the standard i-line resist (b) [16]. 21

Figure 2.8 Near-field lithography stepper prototype [16].. 21

Figure 2.9 200nm pitch EAM mask showing embedded chrome lines in between the SiO2 substrate [24]. 22

Figure 2.10 Simulation results for EAM masks[24] – using 220nm UV light................. 23

Figure 2.11 500nm pitch light coupling mask[25], with 5nm gold in recesses (not visible). 24

Figure 2.12 Surface plasmon illumination[33] : a) normal LCM mask operation b) SPP's generated under a metal film c) LCM mask with metal film above and generated SPP's stimulating near field radiation to the exposure of resist beneath the mask. 25

Figure 2.13 Electric field intensity 10nm into resist a) 630nm wavelength b) 248nm wavelength [33]. 26

Figure 2.14 Planar lens lithography patterning schematic [34]. 27

Figure 2.15 silver lens amplification of the evanescent near field radiation. 27

Figure 2.16 Atomic Force Microscope scans of 250nm features (a,b) and 200nm features (c,d) for ENFOL (a,c) and Planar Lens Lithography (c,d). The height scale is 50nm [36]... 28

Figure 3.1 The ECE department's Raith 150 electron beam lithography system. From the left: a rack contains the control electronics and laser interferometer source; the Raith computer; the vacuum chamber of which the beam column can be seen extending through the top; the work station of which the joystick stage control can be seen in front of two screens: one connected to the Raith computer and the other connected to the Leo computer controlling the Leo electron microscope. 30

Figure 3.2 Electron source [38]. 31

Figure 3.3 Electron optics in a scanning electron microscope, the grey boxes indicate the electromagnetic lenses which direct and shape the electron beam (blue). Purple boxes indicate the user selectable objective aperture [8]. 32

Figure 3.4 An electron beam lithography (EBL) system. A pattern generator, blanking plates and a precision stage added to a computer controlled SEM enable it to be used as an EBL. Image modified from ref [8]. 33

Figure 3.5 High resolution patterning (metal on silicon) produced by electron beam lithography in the Raith 150 system [41]. 34

Figure 3.6 Scans of line features (showing intensity) superimposed upon where they were taken. 35

Figure 3.7 2D scan of a line with significant LER. The dimension along 1 is the main dimension of the scan, whereas the resolution and length set along the dimension 2 determines the amount of LER averaging. 35

Figure 3.8 Karl Süss MA6 mask aligner with operator in the yellow room of the ECE micro-fabrication laboratory. 36

Figure 3.9 Operation detail of Karl Süss MA6 mask aligner [42]. 37

Figure 3.10 (a) Oxford Plasmalab 80 Plus reactive ion etching chamber alongside liquid nitrogen tank (b) The reactive ion etching process. (image in (b) from [42]). 38

Figure 3.11 (a) Edwards Auto-500 magnetron sputter (b) the sputtering process [42]. 39

Figure 3.12 ENFOL manufacturing and processing steps. 41

Figure 3.13 Laurel Technologies spinner in fume cupboard within yellow room. 43

Figure 3.14 (a) Marker feature array consisting of 1μm thick 20μm high markers spaces 0.3mm apart. (b) Layout of marker feature arrays. The overlapping squares represent thick and thin cross markers respectively. The thin crosses – 100nm are patterned to decrease charging effects on scans, with the large crosses (1μm thick) patterned in case the thin crosses do not come out. The red lines are thick finder lines used to locate the centre pattern. 44

Figure 3.15 Major processing steps for silicon substrates. 48

Figure 3.16 Wafer diamond scriber. 49

Figure 4.1 The data capture process. 54

Figure 4.2 Parameters used in the CreatePls program to generate Raith command files 'position lists'. Parameters not shown are the frequency of the beam drift measurements and a number for pre scan beam drift measurements. 55

Figure 4.3 Early beam drift measurement marker. Contamination from scans can be seen as lines across the marker arms. 57

Figure 4.4 A marker cross with executed scans overlaid. Coordinate numbers represent order of scan execution. 58

Figure 4.5 Intensity line scan of a cross marker arm showing threshold algorithm position markers. Green lines indicate the threshold interval in which line edges are weighted. The two red lines are where the algorithm has determined the edges of a marker arm and are stored as position 1 and 2. The yellow line is the calculated mid point of a marker arm stored as position 3. 59

Figure 4.6 Operation of threshold algorithm using relative threshold values. The scan above is of a relatively uniform intensity because the cross marker arm in this case was missing. The threshold algorithm

finds intensity changes within a threshold set to relative intensities of an intensity graph and these are recorded as line edges giving erroneous data. 60

Figure 4.7 Absolute threshold values used to avoid false positive identification of line edges in a uniform intensity line scan (due to a missing cross arm). 61

Figure 4.8 False positive identification of feature edges in a line scan. In this case, one side of a line feature has a lower intensity than that of average, it then falls beneath the absolute threshold values, the algorithm searches further along from right to left finding an intensity change within threshold. The result is that the midpoint is determined at the edge of the line. This can be avoided using the minimum width parameter. 62

Figure 4.9 Thicker PMMA has resulted in an intensity line scan graph that is below the normal threshold window defined by absolute values yielding scan error. As a feature is clearly documented and this result is a false negative. 63

Figure 4.10 The data analysis process. 64

Figure 4.11 Cross with superimposed measured coordinates. Lines represent line equations found from the scanned coordinates. The cross centre is found from the line intersection. 65

Figure 4.12 Electron beam drift over time in ‘x’ and ‘y’ for the Raith 150 EBL. 67

Figure 4.13 Beam drift in ‘x’ and ‘y’ over the course of scanning a marker pattern array on a sample. 68

Figure 4.14 Magnification of Figure 4.13 to show interpolation. Arrows indicate points where actual beam drift measurements were taken, with interpolated beam drift points in between. The time delay to drive between beam drift measurements and cross measurements can be seen. The likely curve of beam drift in ‘x’ is overlaid in red to show the interpolation error. 69

Figure 4.15 The marker array pattern used for measurement experiments. It is symmetrical about its coordinate axes. 70

Figure 4.16 Scan shift and rotation error; due to imperfect alignment of scanning coordinate system (red) to that of the marker pattern features (black)... 71

Figure 4.17 Centroid determination, shift and rotation correction demonstrated using a test data set. Figures 'a-c' and 'd-f' show the determination of the centroid for the mask and patterned silicon data sets respectively. Lines of least squares are shown in red. Figure 'g' depicts the translation in the data points as they are shifted such that the centroid coincides with the origin achieving shift correction. Figure 'h' shows the patterned silicon data points mapped exactly onto their mask counter parts upon achieving full correction. 73

Figure 4.18 Displacement histograms.. 75

Figure 4.19 Distortion map showing the displacement between markers in the silicon substrate and that of the mask. The arrows indicate the 'movement' of the substrate markers... 76

Figure 5.1 Measurement Error: (a) a distortion map showing the apparent movement of markers between two consecutive scans of the same mask. (b) Displacement distribution in 'x' and 'y' and (c) 'x' and 'y' distortion distribution statistics. 78

Figure 5.2 Effect of a dust particle. (a) A large localised distortion can be seen at the top of the distortion map. (b) Displacement distribution in 'x' and 'y' and (c) 'x' and 'y' distortion distribution statistics. 80

Figure 5.3 (a) Distortion Map of an early result showing twist distortion in lower portion of pattern, from this it was thought that a PDMS mask could prevent such twisting. (b) Displacement distribution in 'x' and 'y' and (c) 'x' and 'y' distortion distribution statistics...................... 82

Figure 5.4 The first PDMS mesa mask mount.. 83

Figure 5.5 First PDMS mount mask holder. PDMS mesa was too high for mask aligner sample holder balloon to make contact with

surrounding glass to establish a hermetic seal. Note: later PDMS mounts were adequately degassed and as opposed to the above with plenty of bubbles. ... 84

Figure 5.6 Schematic of PDMS mount mask with pull down due to a difference of 0.25mm between the height of the plastic surround and the of the central mesa. ... 85

Figure 5.7 PDMS mask pull down mount. The surrounding plastic ensures that the mask aligner sample holder balloon can reach a flat plane to seal against, and the PDMS mesa is slightly shorter resulting in a 'pull' down of the mask under vacuum to the silicon substrate beneath. ... 85

Figure 5.8 Schematic of embedded PDMS mask mount. ... 86

Figure 5.9 PDMS mount mask with 'pull' down. (a) A radially symmetric systematic distortion can be seen. This uniform distortion result gave hope that in-plane distortion could be eliminated. (b) Displacement distribution in 'x' and 'y' and (c) 'x' and 'y' distortion distribution statistics. ... 87

Figure 5.10 Glass mask mount for embedded PDMS mesa. ... 88

Figure 5.11 Embedded PDMS mount mask. (a) Distortion is non uniform as opposed to the previous PDMS mask mount design. (b) Displacement distribution in 'x' and 'y' and (c) 'x' and 'y' distortion distribution statistics. ... 90

Figure 5.12 (a) 50μm separation original mask mount distortion map. Overall distortion is non-uniform with multiple localised distortions. (b) Displacement distribution in 'x' and 'y' and (c) 'x' and 'y' distortion distribution statistics. ... 92

Figure 5.13 (a) 25μm separation original mask mount distortion map. Overall distortion is non-uniform with multiple localised distortions. (b) Displacement distribution in 'x' and 'y' and (c) 'x' and 'y' distortion distribution statistics. ... 93

Figure 5.14 (a) 0μm separation original mask mount distortion map. Overall distortion is non-uniform with multiple localised distortions. (b)

Displacement distribution in ‘x’ and ‘y’ and (c) ‘x’ and ‘y’ distortion distribution statistics. 94

Figure 6.1 (a) Abaqus/CAE startup dialog (b) Abaqus/CAE integrated environment. 99

Figure 6.2 0.1metre length beam under reverse pressure load of 1 atmosphere. The legend shows the displacement in the ‘z’ component. The maximum displacement in ‘z’ is -1.6x10-7m or 0.16μm. 102

Figure 6.3 The same beam as in Figure 6.2, but in terms of the rescaled units in terms of microns (the legend is in microns). 100,000μm length beam under the same pressure load has a displacement of -0.16μm, matching the result in Figure 6.2 and thus passing the test for unit consistency. 103

Figure 6.4 (a) Mask membrane and particle upon silicon substrate. (b) Axially symmetric distortion of mask membrane resulting from interaction of centrally located particle. 104

Figure 6.5 Pressure load (purple arrows) and boundary conditions (orange and blue arrows) applied to the glass membrane and rigid bodies in the load module. 107

Figure 6.6 Example of seeding: (a) Seeds for the placement of elements are placed on the model. (b) The mesh is generated with the elements being placed as closely as possible to the pattern dictated by the seeds. 108

Figure 6.7 Partial view of meshed glass mask membrane. Elements are 10x10μm. 108

Figure 6.8 The axisymmetric simulation of a 40μm high particle underneath an ENFOL mask. A distortion of -1.05μm is evident from the particles presence. Also the distortion from the raised edge of the mask can be seen. 109

Figure 6.9 Axisymmetric simulation of a 30μm high particle underneath an ENFOL mask. A distortion of -0.85μm is evident from the particles presence. 110

Figure 6.10 Axisymmetric simulation of a 20µm high particle underneath an ENFOL mask. A distortion of -0.625µm is evident from the particles presence. 111

Figure 6.11 Displacement of mask membrane in ‘x’ and ‘z’ due to varying initial separations of mask to substrate of 20,30 and 40µm in ‘z’. Residual displacements of mask in ‘x’ (solid lines) can be seen where mask is in contact with substrate, due to initial separation height in ‘z’ stretching mask membrane. Displacement in ‘z’ is depicted (dotted lines) to indicate contact of mask with substrate. 112

Figure 6.12 Simulation of a 30µm high particle and displacement in ‘z’. It can be seen that from 30µm topological difference, either due to the dust particle or the edge of the mask that the membrane takes thousands of microns to come back into intimate contact with the substrate. 114

Figure 6.13 Area of mask where contact is within 10nm in ‘z’. This represents only one third of the masks radius. 115

Figure 6.14 Area of mask membrane with distortion within +/-40nm in ‘x’ for 30µm topological interference. 115

Figure 6.15 Usable radius of mask for varying offsets in ‘z’. 116

This thesis is dedicated to: Robert Wright, Nuong Wright and Khanh Huynh

Chapter 1: Introduction

1.1 The Development of the Modern Computer

Society today is inextricably linked by computers; digital information in the form of money, government databases, internet pages, email and telephone calls traverse digital links worldwide. From 1712 the industrial revolution spanned two centuries and changed the lives of people around the world through steam power, textiles, chemicals and electricity. However, the digital information age today in which computers form part of our everyday lives has had as profound change on our livelihood as the industrial revolution did, and spanning only a few decades. This modern age is due to the invention of the transistor and in turn the integrated circuit which brought about the modern computer.

In the time of 1945, two devices existed to perform logic; these were relays and vacuum tubes and seemed to be making all manner of things possible. In a relay an electromagnet moves a metal contact (held back by a spring) to another metal contact to perform a switching operation. Relays were strong and reliable, however because of their mechanical nature relays were limited to mechanical switching speeds of approximately 1 millisecond (far to slow for powerful calculation). Vacuum tubes were one million times faster because they relied upon the flight of electrons in a vacuum. A vacuum tube uses a glass bulb in which electrons are emitted from a hot wire filament and are collected at a positively charged plate. A grid in between these electrodes controls the flow of electrons from the filament for switching and amplification. Alas, vacuum tubes were short lived as the filament would burn out or the bulb would leak and lose vacuum. Also the heat of the filament was so intense that they could only practically be used in low numbers, nowhere near the thousands needed to produce computers.

At the time the head of research at Bell Telephone Laboratories (Bell labs), M. Kelly, knew that these issues would limit further advancement. In the summer of 1945 he set up a semiconductor research group which included William Shockley, John Bardeen and Walter Brattain with the aim to create a solid state device that would eventually

replace the tube and the relay [1]. A solid state device promised to overcome the problems of relays and vacuum tubes, as electrons would only have to travel short distances in a solid, and therefore would not require mechanical movement, or hot filaments and vacuums. A solid state device would also allow miniaturization, which had not been possible before. Such a device, the bipolar junction transistor (BJT) was invented in December 1947 by John Bardeen and Walter Brattain [1].

Miniaturization was now possible and in the late 1950's Jack Kilby of Texas Instruments, and Robert Noyce of Fairchild Semiconductor independently developed a process to integrate several BJT transistors on to one 'chip' – developing the first integrated circuits (ICs). The availability of electronics for the masses was changed for good, and BJT technology poured money into the semiconductor industry. However it was not all plain sailing from here; although BJT's enjoyed nearly three decades as the dominant technology they were not suitable for the development of the microprocessor. A different kind of transistor stepped into the limelight, namely the field effect transistor (FET). FETs offer orders of magnitude lower power consumption[1]. The FET had in fact had been designed before that of the bipolar junction transistor. In 1925 Julius Edgar Lilienfeld designed and patented three types of field effect transistor which included the now ubiquitous metal oxide semiconductor field effect transistor [2] (MOSFET). However, for the most likely reason that high quality semiconductor materials were not yet available for some decades and therefore hindering the successful production of FETs, Lilienfelds patents drifted into obscurity. They were however to see the light of day again in December 1947 when William Shockley sought and initially failed to obtain a patent for his idea for a FET. Eventually on the 5th of July 1951 Shockley produced world's first known FET and managed to patent it. FETs then struggled for years with issues of speed, silicon area needed, reliability and a complicated and costly manufacturing process [3]. They were waiting for an advancement which came years later in 1968. This was the invention of the silicon gate MOSFET by Federico Faggin at Fairchild semiconductor and became the tipping point where FETs came into their own. FETs now took half the area needed previously, were four times faster, had less leakage current and within ten years supplanted BJTs as the technology of choice.

[1] When used in complementary metal oxide pairs, CMOS.

With the silicon gate MOSFET large scale integration was now possible. With the technology only a year old Ray Holt and Steven Geller of Garret AiResearch developed six ICs which performed the inner functions of a central processing unit (CPU) producing the first computer using thousands of transistors on only a few pieces of silicon in 1970 [4, 5].

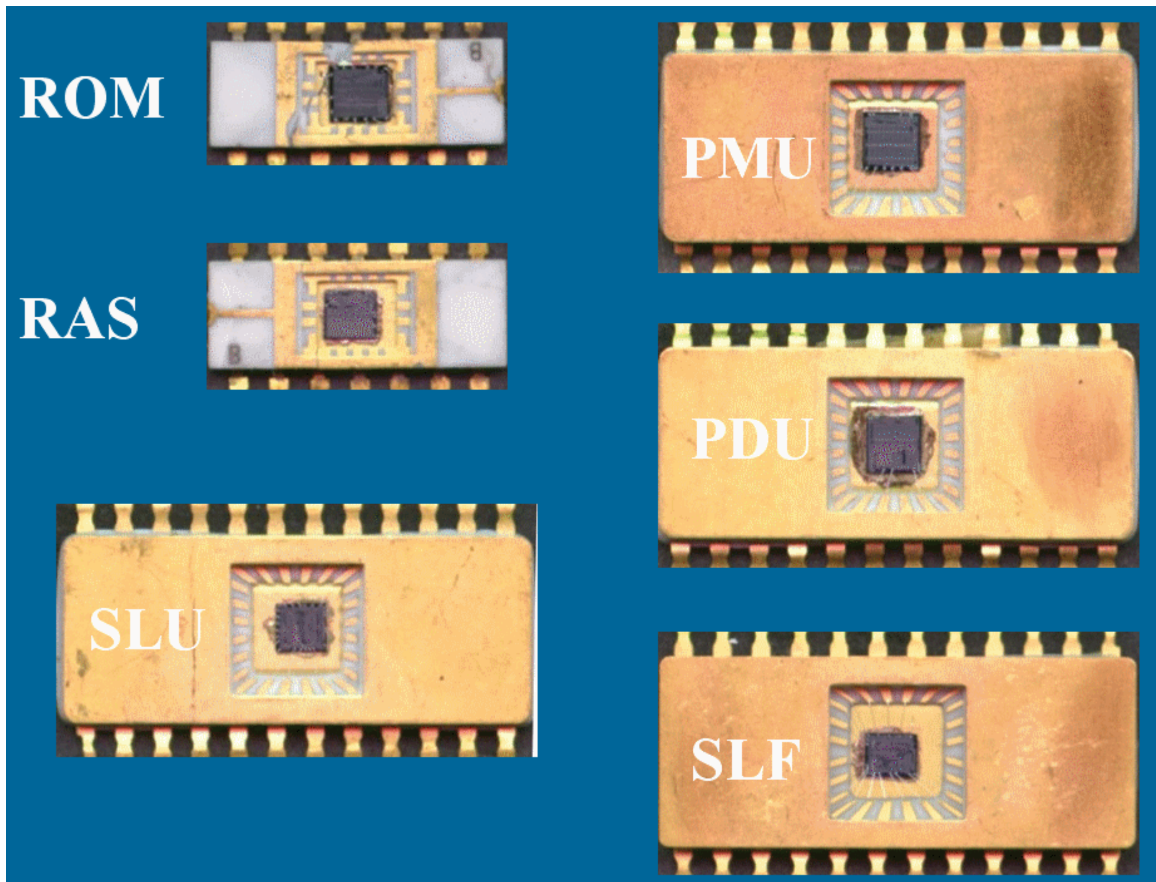


Figure 1.1 The central air data computer for the F14 jet fighter aircraft, the first computer built using large scale integration. The modules are: ROM read only memory, RAS random access storage, SLU special logic unit, PMU programmable multiplier unit, PDU parallel divider unit and SLF special logic function. The final processor used: 1-PMU, 1-PDU, 1-SLF, 3RAS's, 2SLU's and 19-ROM's [4].

The computer was used in the Grumman Aircraft F14 'Tomcat' jet engine fighter aircraft, being delivered to Grumman in early 1970. The complete processor used 1-PMU, 1-PDU, 1-SLF, 3-RAS's, 2-SLU's, and 19-ROM's (Figure 1.1). It was very advanced for its time; the processor was 20 bit with pipelining, abilities which were not to come for years in general computing. It was considered so advanced that the

U.S. Navy kept it secret until 1998 [4]. Hence when the first commercial microprocessor was produced by Faggin and Marcian Hoff at Intel, the 4 bit 4004 in 1971 (Figure 1.2) which wasn't as powerful, it was what further microprocessors were based on.

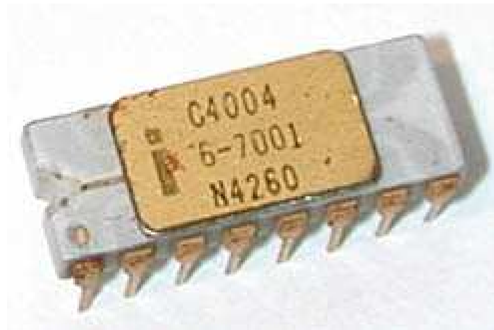


Figure 1.2 The Intel 4004 microprocessor, the worlds first computer constructed as a single IC using large scale integration, and the basis for all future processors [6].

The 4004 was designed for the Japanese company Busicom for their range of calculators and consisted of 2,300 MOS transistors [6]. With more complex circuits produced this processor was then superceded by 8 bit processors, followed by 16 and eventually 32 bit processors. Continually tougher problems could now be tackled by machine, and applications for computers broadened and soared to the extent that they eventually became ubiquitous.

There are two main advantages to miniaturisation that made IC's and hence modern computers possible; the first is that as feature sizes become smaller, charge carriers do not have to travel as far, and therefore device speeds increase – overall performance increases. The second advantage and probably the more important driving factor for this miniaturisation, was the dramatic decrease in cost per device:

"What we didn't realize then was that the integrated circuit would reduce the cost of electronic functions by a factor of a million to one, nothing had ever done that for anything before" - Jack Kilby

1.2 Photolithography: the main tool for the manufacturing of integrated circuits

Photolithography is the technology that produces the patterns that form transistors, metal interconnects, vias, and contacts in ICs. Because it defines all features produced, it is the main tool in the manufacture of ICs. Shown below in Figure 1.3, photolithography works by illuminating light (UV) through fine lenses and a high purity glass mask which has metal patterns on it. The light passes through the uncovered parts of the mask and exposes a light sensitive material called a photoresist below, covering a silicon substrate.

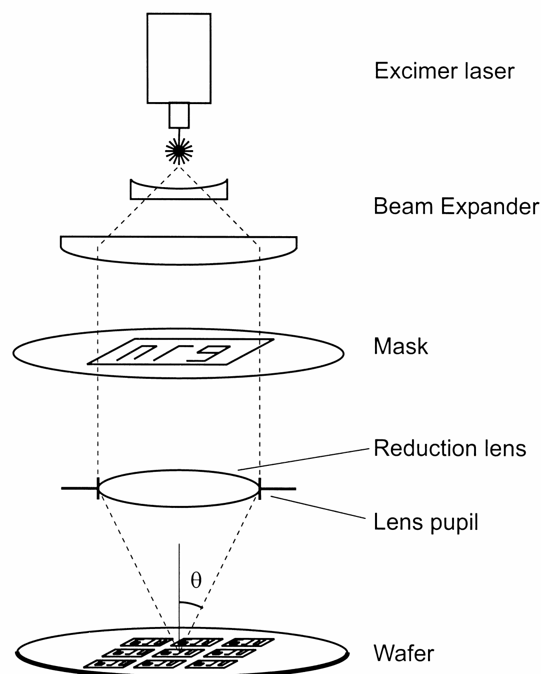


Figure 1.3 Projection optical lithography [7]. UV light illuminates a photoresist covered silicon substrate through a mask and lenses.

The areas exposed in the photoresist are altered chemically so that they are more easily dissolved[2] in a developing chemical. After developing a photoresist pattern is left on the silicon substrate. This pattern may be transferred on the silicon substrate to produce devices using processing steps such as etching, growth, doping and lift-off as depicted in Figure 1.4.

[2] This is the case for a positive photo resist the opposite is true for a negative photo resist.

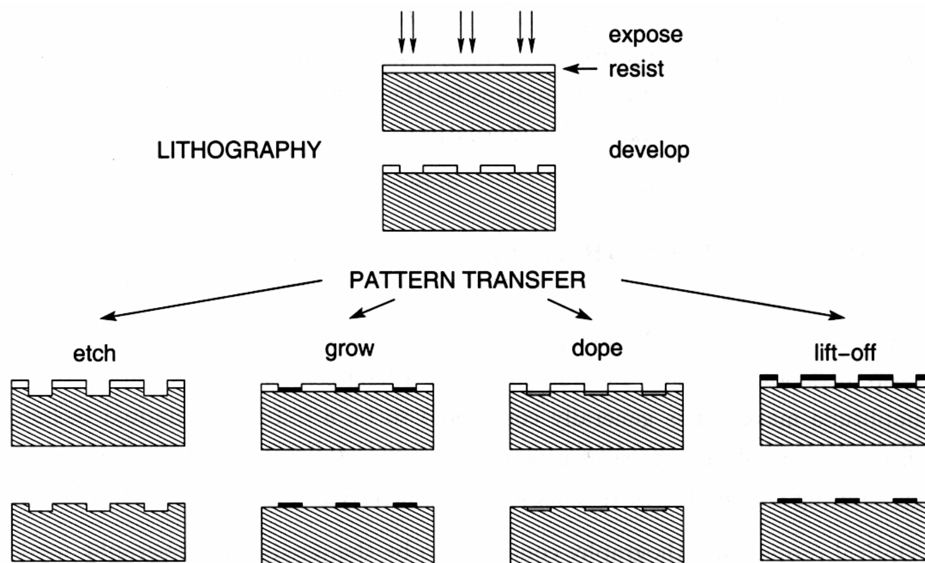


Figure 1.4 Pattern transfer processes [7] .

In the drive for miniaturisation, photolithography for smaller features is getting exponentially more expensive; this is because smaller features require higher precision equipment, finer optics, and smaller wavelength light sources for exposures. Conventional photolithography is also nearing its limit for the smallest features it can produce. Diffraction is the effect of light bending around objects –spreading out as it passes by them. This effect is negligible for features larger that the wavelength of light, however when the gap in a mask is on the order of the illuminating wavelength diffraction is significant as shown in Figure 1.5.

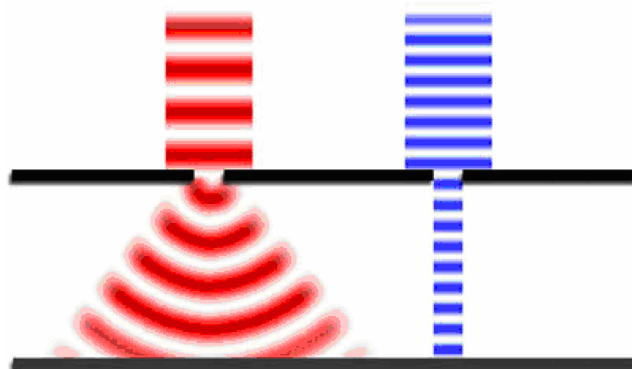


Figure 1.5 Diffraction: Red light passing through a gap half its wavelength is significantly diffracted (left) whereas blue light with a shorter wavelength through the same size gap is not (right) [8].

As semiconductor features are scaled down past half the illuminating wavelength, features that are in close proximity become so blurred due to diffraction, that they cannot be resolved from one another. This limit is called the diffraction limit,

$$p_{\min} = \frac{\lambda}{2NA} \quad , \tag{1}$$

where λ represents the wavelength of the light, *NA* represents the numerical aperture which, in air, can be as high as 1 and $p_{\min}$ represents the minimum period of grating that can be resolved. Therefore ordinary propagating light cannot resolve features smaller than one half the exposing wavelength. Currently an Argon-Fluoride laser is used as the light source with a wavelength of 193nm. The feature sizes required for IC's in a few years time are so small that photolithography on its own, without some form of enhancement, will no longer be able to cope. 130nm period features are planned by industry for 2007 and 90nm period features by 2009. The next section outlines candidate next generation lithography methods to provide high resolution patterning to continue the trend of ever decreasing transistor feature sizes.

1.3 Possible Next Generation Lithographies

1.3.1 Extreme Ultraviolet Lithography (EUVL)

EUVL operates in a similar fashion to that of normal photolithography, and could be considered a natural progression from standard optical lithography. Light travels from a source and is focussed to expose features on a substrate surface, except that the light source is of a much shorter wavelength at 10-14nm [7] and the system users mirrors instead of lenses for optics including the mask as a mirror (Figure 1.6). The latter is because at such short wavelengths, which are almost x-rays, the light rays are no longer significantly refracted by glass elements and therefore lenses are rendered useless. Materials are also very lossy at these short wavelengths, which also hinders the design of refractive optics.

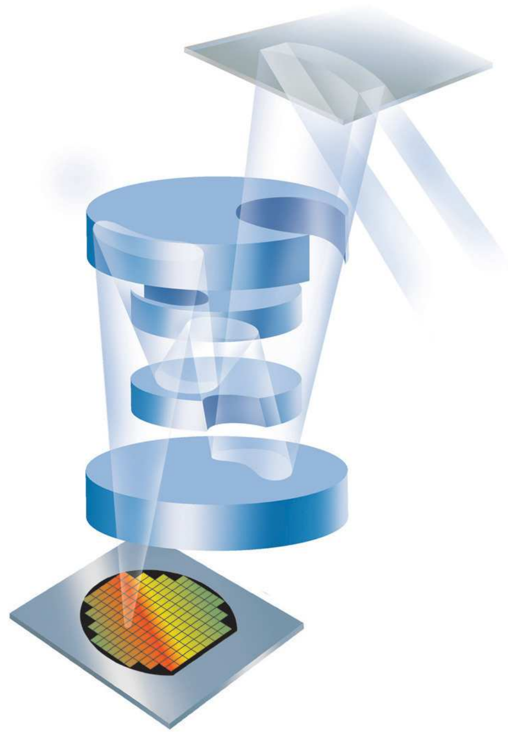


Figure 1.6 An extreme ultraviolet lithography system [9]. The mirror optics are shown in blue with the mirror mask shown in grey at the top.

There are two types of sources that have been used to produce EUV: one is a laser induced plasma, the most common as it would be much cheaper than the alternative, that of synchrotron radiation produced by a type of particle accelerator which generally are exorbitantly expensive. For the case of a laser induced plasma, a laser is fired at a supersonic jet of Xenon gas with only a few percent of energy of the original beam producing EUV[10]. The beam is then reflected off nine surfaces on its way to the target; the issue is however, that with the reflective surfaces only reflecting 70% of incident radiation, only 4% of the generated radiation from the source arrives at the target.

This, coupled with the low efficiency of the source means that exposure times will be long and a more powerful source would be needed to provide adequate exposure times. Another issue is that the plasma sources used cause contamination of the nearby mirror surfaces through sputtering[3], gradually degrading the mirrors such that

[3] A process by which plasma ions impinge upon surfaces, ejecting atoms from them which then roam and settle on surrounding objects

they will require continual replacement. Finally, high performance EUV resists have yet to be developed, but the enabling breakthrough would be the discovery of a practical source. The latest results to date gave resist limited resolving capabilities of 32nm periods[11] (Figure 1.7).

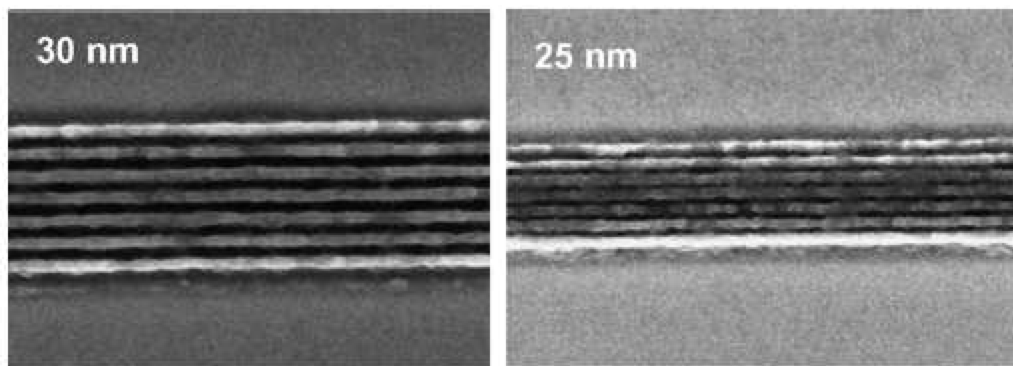


Figure 1.7 Results from using EUVL [11] .

1.3.2 Electron Projection Lithography

Whereas electron beam lithography (EBL) transfers patterns by scanning a tiny beam all of a sample, literally drawing it upon the substrate one dot, line at a time, electron projection lithography (EPL) attempts to project a large swathe of electrons through a mask onto the substrate. EPL not only has the advantage of enabling a parallel exposure but that enabling technology such as electron resists are a mature discipline. The most advanced prototype EPL machine being developed is depicted in Figure 1.8 and has produced 70nm period structures [12]. This machine approximates a flood exposure by using a wider beam under which a mask is scanned underneath.

The system at the time of reporting is limited to 2 wafers per hour. In order to obtain higher throughput higher beam currents must be used. However as the beam current is increased the repulsion of the electrons in the beam (Coulomb interaction) introduces blur, defocusing the pattern. Current efforts are focussed on increasing the width of the beam and the use of higher acceleration voltages to increase the current. The authors expect increased throughput to 20 wafers per hour.

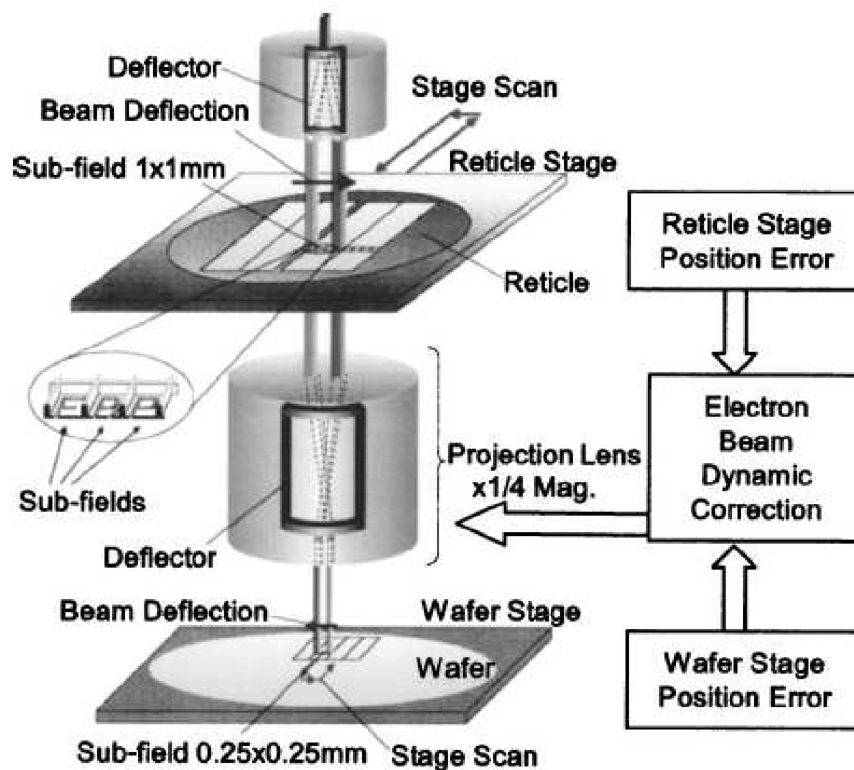


Figure 1.8 Nikon electron beam projection lithography system NSR-EB1A. The wide beams can be seen which are projected through the reticle/mask as it is moved beneath it exposing the whole mask pattern onto the wafer beneath [12].

1.3.3 Nanoimprint Lithography

Nanoimprint lithography (NIL) is a simple method to produce nanometre resolution features at low cost with high throughput. It is a method of producing patterns through mechanical pressure of a mold into a polymer coated substrate to leaving an impression. It does not suffer from any diffraction effects, beam drift or scattering. It is only limited by the production of the mold which is usually by EBL.

Figure 1.9 shows the highest resolution results to date in which 5nm half pitch features have successfully been transferred. This demonstrates NIL as having patterned the highest resolution to date for a high throughput process. There are a few variations on NIL the most widespread are that of thermoplastic NIL and step and flash NIL. The simplest form is Thermoplastic NIL, the original process developed by Professor Stephen Y. Chou and his group at Princeton university in 1994[13].

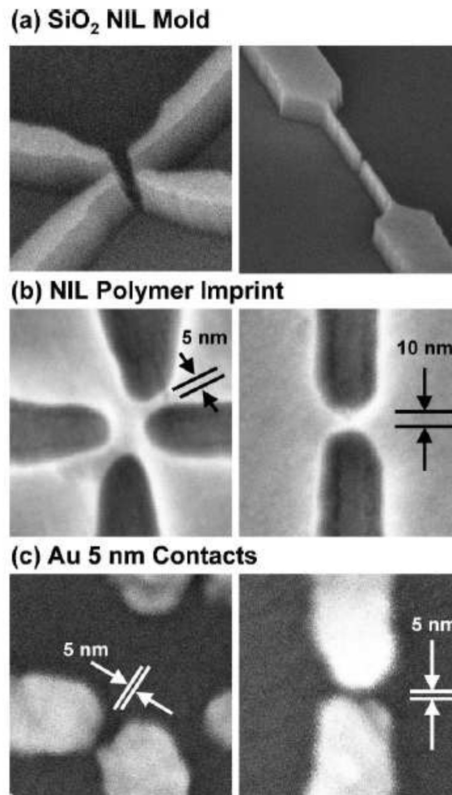


Figure 1.9 Highest resolution NIL results to date, 5nm half pitch [14].

Once a mold is fabricated by EBL, RIE and coated with a surfactant to make it non-stick; the imprint pattern transfer process begins by spin coating an imprint resist (a thermo plastic polymer) onto a substrate. The mold with its predefined topology of features is then brought into contact with the substrate (Figure 1.10) at a given pressure and heated to melt the resist into the depressions of the mold. Once the mold and substrate have cooled the mold is then removed leaving the depressions in the substrate.

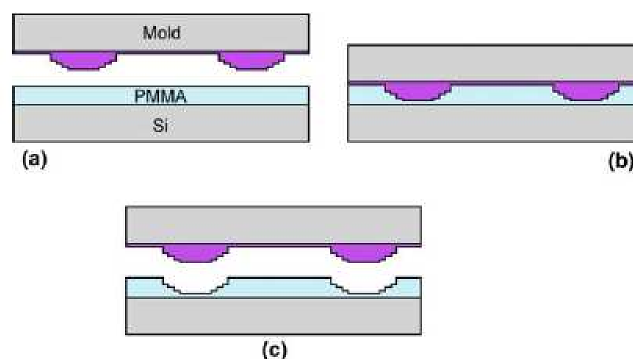


Figure 1.10 NIL process using a 3D mold (a) substrate coated in thermo plastic polymer. (b) imprinting of mold into substrate under heat and pressure . (c) mold removed after cooling and imprint remains in substrate [15].

NIL doesn't have any significant issues holding it back as a next generation lithography process. In fact commercial machines are available[4] and the International Technology Roadmap for Semiconductors added it for the 32nm node as an 'official' next generation lithography technology.

1.3.4 Evanescent near field optical lithography

Evanescent Near Field Optical Lithography (ENFOL) takes advantage of near field light, and is therefore capable of patterning very high resolution without any special radiation sources or high cost equipment[5]. In particular it is not limited by the diffraction limit; rather it is sub-diffraction limited, which means it can pattern features that are smaller than the wavelength of the incident light. The finest experimental results to date[6] are 100nm periods that have been patterned using 365nm illumination[16]. Sufficient intensity and contrast down to $\lambda/20$[17] has been predicted through simulation for protruding amplitude masks, and for plasmon illumination (described in a later section) resolutions down to $\lambda/31.5$ have been predicted. This form of lithography is what this thesis is based on and is described further in the next chapter.

1.4 Aim of this thesis

The successful implementation of ENFOL technology promises a low cost alternative for high-throughput, high resolution patterning for research laboratories and perhaps commercial production. The success of a lithography methodology is its production of features predictably within a known tolerance. This enables the reliable fabrication of devices at increasing resolutions. A requirement of ENFOL is intimate contact between the mask and substrate so that the substrate is exposed by the near field radiation which decays rapidly over a few tens of nanometres. A conformable contact mask is needed to provide such intimate contact. Such a mask is inherently flexible

[4] EV Group http://www.evgroup.com
[5] Relatively high cost. A cheap electron beam lithography system is needed and these are on the order of $3million. However industrial equipment can easily run into the tens or hundreds of millions
[6] without the use of phase shifting lithography

and as it flexes into contact its features can become distorted. In order to assess the viability of processes using conformable masks and hence the success of such a process, the extent of this distortion must be known. The aim of this thesis is to characterise such in-plane distortion, root out the causes of which and proffer a solution to the problem.

1.5 Thesis Arrangement

The following chapters are outlined as follows: the next chapter delves into further detail on the theory and practical aspects of ENFOL and a potential enhancing technology both of which are active research areas. Background information to put ENFOL in the context of other conformable mask lithographic methods is also provided. This chapter is followed by the experimental techniques chapter which gives detail on the essential apparatus involved in the research and the processes by which they are used to manufacture ENFOL masks and pattern silicon substrates. The fourth chapter then describes the in-plane distortion measuring process in detail, outlining the method of data acquisition and the processes involved in the subsequent processing and analysis of the data determining the in-plane distortion. This is followed by Chapter 5 in which the distortion results of ENFOL mask patterning are presented. In order to fully explain the distortion found Chapter 6 presents the results of mechanical modelling of the membrane-like mask and substrate system.

Chapter 2: Background and Literature Review

2.1 Evanescent near field optical lithography

ENFOL is able to overcome the diffraction limit due to its utilisation of near field radiation. For example, underneath a mask grating with a pitch that is not below the wavelength of the UV light illuminating it, two types of light are emitted (Figure 2.1): one type is far-field and it is one we are familiar with, it is the type our eyes use for vision; the other type however, is evanescent, which means that it decays rapidly and does not reach very far – on the order of 10-100nm [16].

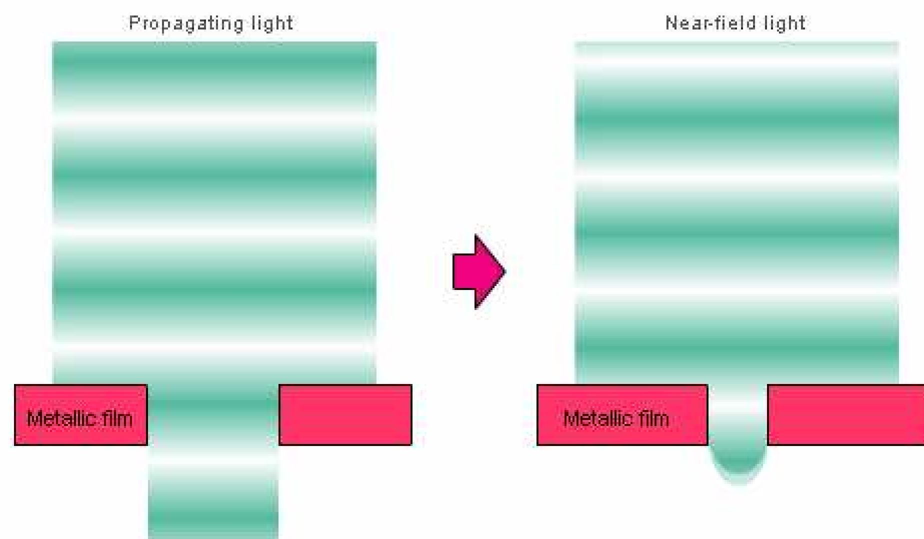


Figure 2.1 Ordinary propagating light and evanescent near field light [16]. Ordinary light propagates, while near field light decays rapidly.

This type of light is called near field radiation which occurs as a result of the interaction of light with metallic surfaces; on the surface of a metal, electrons wander around freely forming a two-dimensional ocean that ripples with waves called surface plasmons [18, 19]. When incident light (photons) strike the metal surface they set the electron sea vibrating – coupling with the surface plasmons already there, to produce what are then called surface plasmon polaritons (SPP's) [19]. These SPP's accumulate the electromagnetic energy of the light, and when they can only move over a short distance - such as that of high resolution semiconductor features; they compress this electromagnetic energy into a small volume producing an intense electric field.

It is then this electric field – resonating at the frequency of the incident light – that causes the re-radiation of photons [19, 20] which expose a thin layer of resist below the mask, producing features that are beyond the diffraction limit of light.

Because near field radiation decays rapidly – in particular, evanescent near field radiation only penetrates a few tens of nanometres – it is essential that conformable masks be used for ENFOL. This is so that the mask conductors are in intimate contact with the photoresist beneath, otherwise the evanescent radiation would decay before being able to expose the photoresist. This is shown in Figure 2.2, where a mask is brought in contact with a photoresist coated substrate by vacuum, and illuminated with UV light.

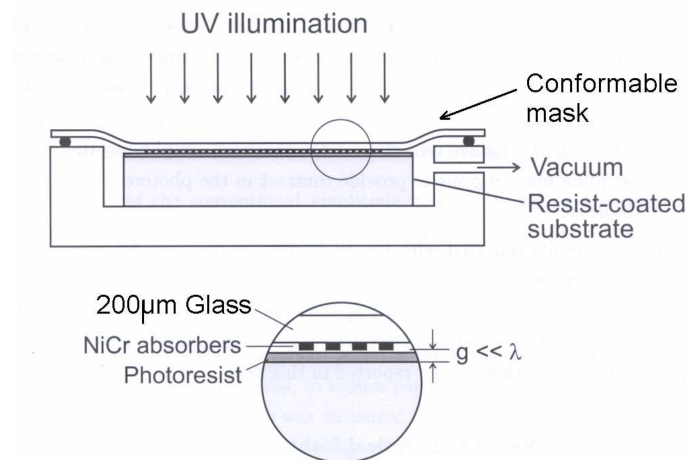


Figure 2.2 Schematic diagram depicting the ENFOL process[7]. A conformable mask is held in intimate contact with a thin photoresist coated substrate, and illuminated with UV light.

As shown in the enlargement in Figure 2.2 the photo-resist is very thin, because the evanescent fields decay with a few tens of nanometres. Using this process the two main processes in the lithography tool box, – additive pattern transfer and subtractive pattern transfer, have been successfully demonstrated for ENFOL [7].

To satisfy the requirement of intimate contact either the mask or the substrate must be conformable. Conformable substrates have been investigated [21, 22], however the use of these limits the range of devices that can be fabricated. The use of a conformable mask is much more prominent, and types of conformable masks are outlined in the next section.

2.2 Conformable Contact Masks for ENFOL

Contact lithography was mainstay of the semiconductor industry into the mid 1970's, upon when it was abandoned because of damage to wafers and masks. However for special applications contact lithography survived and conformable masks were realised [23]. Conformable masks avoid the damage that dogged previous hard masks for contact lithography because they do not build up high stresses when brought into contact with wafers – which led to mechanical breakdown; rather they 'give' – bend so that patterns already on the mask, or the wafer substrate are not harmed. Moreover, this conforming nature of the masks means that intimate contact is achieved over large areas – enabling the use of the evanescent near field.

There are three types of mask that have been used for ENFOL: the first is the protruding absorber mask (PAM) in Figure 2.3(a) which has been used at the UOC [7]; the second is the embedded absorber mask (EAM) (Figure 2.3 b) created at MIT [24]; and the third is the light coupling mask (LCM) (Figure 2.3 c), developed by IBM in Zurich [25].

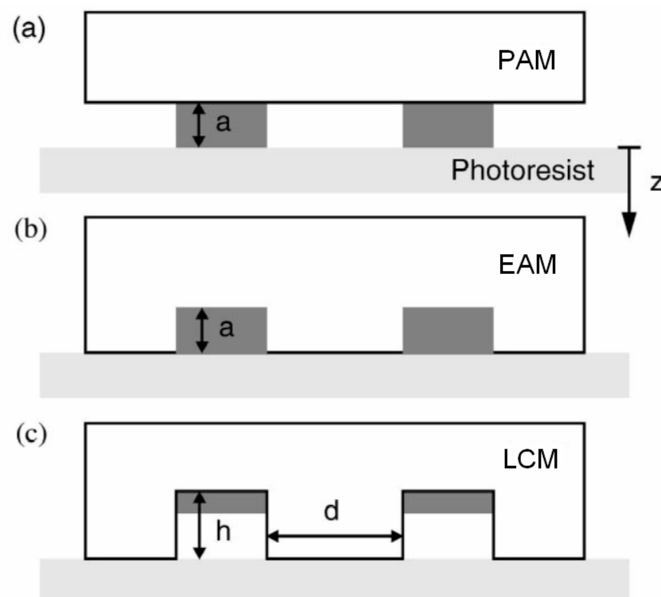


Figure 2.3 Conformable masks[26]: (a) protruding absorber mask (b) embedded amplitude mask (c) light coupling mask.

The only mask to demonstrate sub-diffraction limited resolution – for arbitrary patterns – so far is the PAM [27], however the EAM in theory should achieve the same. Although the LCM in its current form cannot utilise the evanescent field [25, 28], it is included here to describe the potential of the use of a soft polymer (PDMS) and its makeup which may lead to its augmentation to be used for surface plasmon illumination described in section 2.3.1.

2.2.1 Protruding Absorber Masks

The protruding absorber mask for the use of ENFOL was pioneered at Canterbury University[29]. This mask (see Figure 2.2) has been made from low stress silicon nitride in a series of steps to produce a thin membrane which is ~1μm thick. Silicon nitride is used as it is a proven loss-stress membrane mask as it was developed for x-ray lithography. As mentioned earlier, it is the only mask to demonstrate sub-diffraction limited resolution without the use of interference. Figure 2.4b shows the finest experimental result to date for ENFOL, in which 64nm pitch features [30] were patterned using 365nm light – well beyond the diffraction limit. These were produced using a silicon nitride mask with amorphous silicon absorbers (Figure 2.4a) as they can be dry etched – thus producing high aspect ratio features through anisotropic etching. This is opposed to Cr absorbers that Ito et al had previously used which needed to be etched isotropically (using Cl_2 and O_2 plasmas), which made it difficult to produce etched trenches narrower than that of the absorber thickness [30].

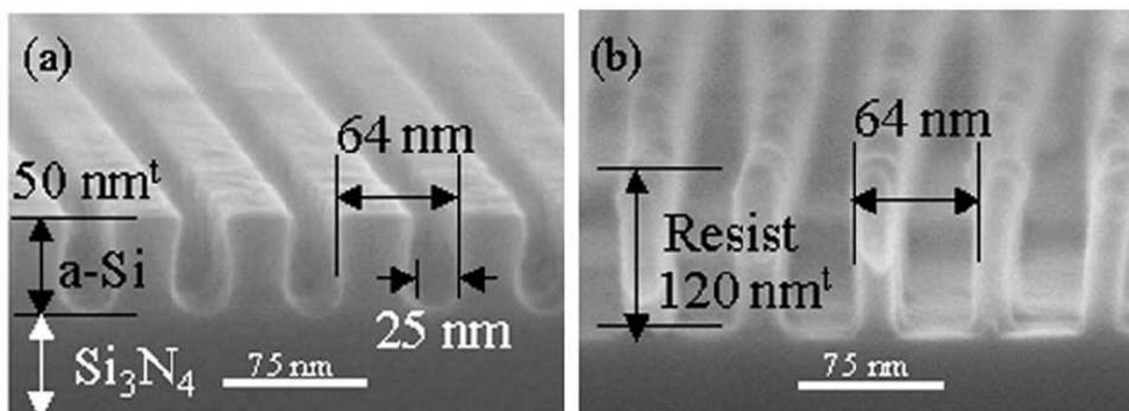


Figure 2.4 (a) Silicon nitride mask with amorphous silicon absorbers. (b) 64nm period features produced using ENFOL. The features were patterned using a chemically amplified resist [30].

The grating pattern produced in Figure 2.4b has wider spaces than peaks, and is a consequence of the patterning of the spaces – apertures on the mask. Although the apertures are 25nm (Figure 2.4a) this results in wider spaces on the substrate due to the isotropic nature of the near field light produced below an ENFOL mask[30]. Narrower apertures are required to produce narrower spaces.

Simulations carried out [17] show the potential for even higher resolution (Figure 2.5). Results indicate the possibility of 20nm pitch features using resists 3nm thick, a very high resolution which presents an exciting prospect for ENFOL researchers. However as resolution increases, the evanescent near field radiation decays more rapidly and resists need to be even thinner. Despite a thin resist high aspect ratio patterning is still possible through the use of a tri-layer resist process [30].

A significant challenge for the ongoing study of ENFOL was the lack of a photoresist that could produce high resolution features using i-line 365nm light. This was the case because current commercially available photoresists are designed for diffraction limited resolution; therefore an i-line resist is only good for periods down to 180nm. Imaging into such resists resulted in results with high LER e.g. Figure 2.6b) such that features were very hard to discern.

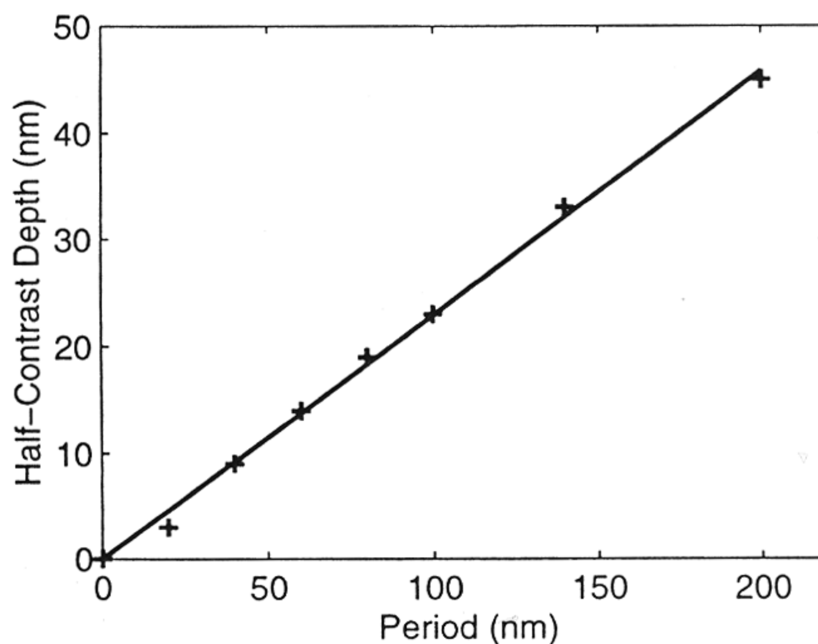


Figure 2.5 Exposure depth (half contrast) as a function of grating period [17].

Research undertaken by Toskiki Ito et al. at Canon has produced a special chemically amplified (CAP) resist [16]. The difference in resist performance is clear (Figure 2.6). The CAP resist was made by blending a high resolution Krypton Fluoride resist, one that was developed for KrF lasers which have a wavelength of 193nm and therefore higher resolution; with that of an i-line photo acid generator (PAG). The PAG makes the KrF resist i-line compatible as, once exposed to i-line radiation, the PAG forms acid attacking the KrF resist 'exposing' it.

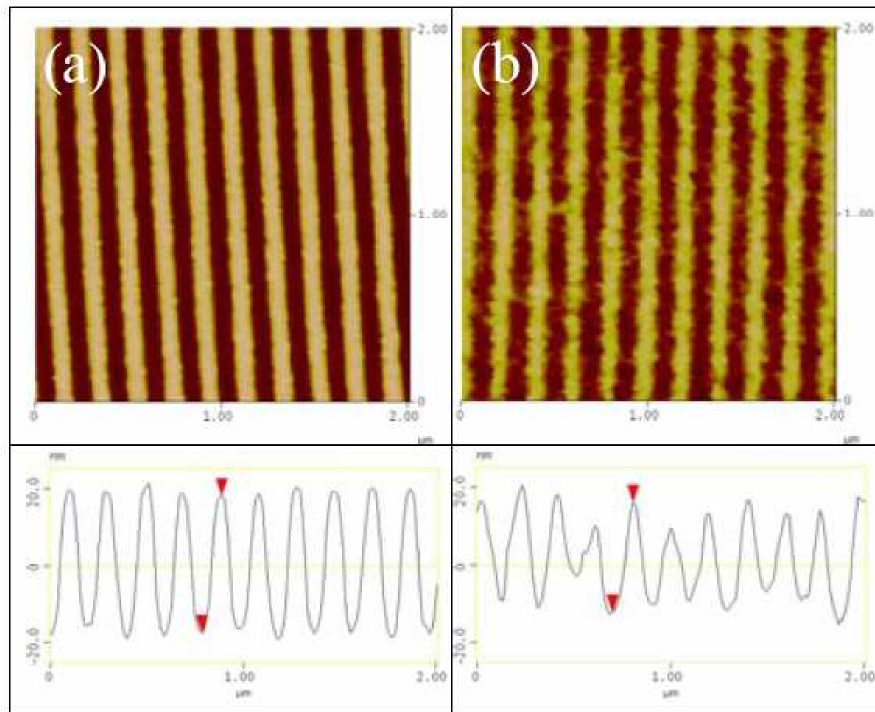


Figure 2.6 80nm half pitch features patterned in chemically amplified resist a) and AZ DNQ-novolac (standart) i-line resist b) [16].

When UV light exposes a positive resist it breaks its bonds; for the ideal resist with polymer chains all of equal length, each polymer chain would require the same energy to break respective bonds and therefore the exposing process would be even producing no LER from an ideal mask. However if all the chains are significantly uneven the distribution of bond breaking energies would increase and an exposure would produce areas of broken chains randomly spaced apart. This gives rise to LER. Figure 2.7 shows the molecular weight distribution of the CAP resist (a) and the DNQ-Novolac standard i-line resist (b).

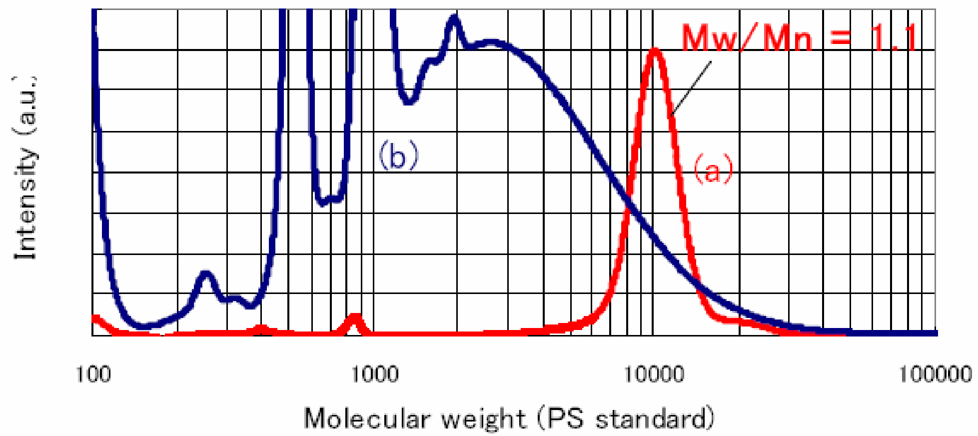


Figure 2.7 Molecular weight distribution of the CAP resist (a) and the standard i-line resist (b) [16].

It can be seen how much better the new CAP resist is by its molecular weight indicating that its polymer chains are much the same length. This is not the case for the standard i-line photoresist of which the molecular weight signature ranges across the band. In addition to the CAP resist and the result in Figure 2.4 the Canon team have also developed a near-field lithography prototype stepper machine for patterning multiple IC's per wafer (Figure 2.8) [16].

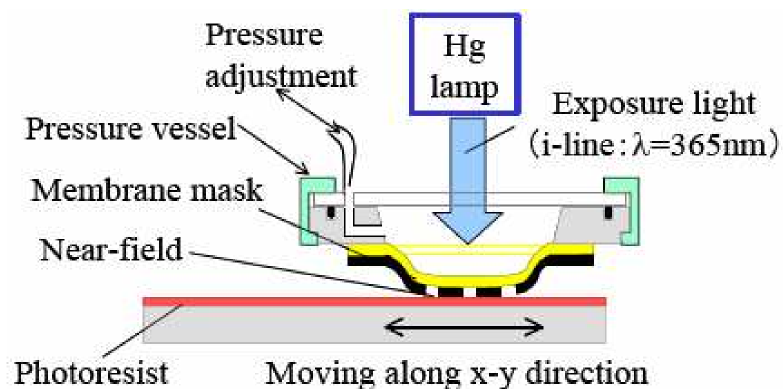


Figure 2.8 Near-field lithography stepper prototype [16].

2.2.2 Embedded Amplitude Masks

EAMs were devised by James G. Goodberlet and Bryan L. Dunn at the Massachusetts Institute of Technology (MIT) [31]. These consist of a 150µm thick fused silica membrane with absorbers and transparent regions separately patterned on top of the flat surface. The effect of embedding the absorbers offers a number of advantages. The first is that the absence of air between the absorbers means that the light does not reflect off an SiO_2 – air boundary as it does with a PAM, and also the transparent regions act as wave guides for the light to reach the surface – giving greater transmission and contrast. This latter effect is backed with simulations by Paulus et al. [26] comparing PAMs with EAMs. Second, the planarity of the mask would protect against contamination – because there are no large grooves for particles to trap themselves in. Additionally the planarity facilitates better cleaning through the use of piranha baths (sulphuric acid and hydrogen peroxide) and ultrasonic bathing. The third advantage is that by embedding the absorbers they are protected from wear [32] and finally the mask boasts the lowest in-plane pattern distortion reported of 58nm over $2cm^2$ [31]. However this is for two consecutive exposures of a mask on a single substrate, not of measuring the distortion of mask features with respect to that patterned on the substrate.

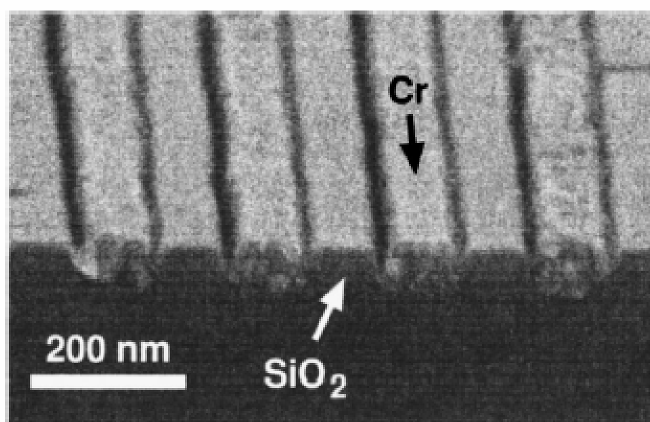


Figure 2.9 200nm pitch EAM mask showing embedded chrome lines in between the SiO_2 substrate [24].

EAMs have been used with deep UV wavelengths – 220nm as fused silica allows wavelengths down to 170nm, unlike silicon nitride that absorbs wavelengths shorter than 365nm. Using 220nm wavelength deep UV illumination, 200nm pitch lines have been produced [24]; finer experimental results have been obtained, however these utilised near field interference – which cannot produce truly arbitrary patterns at the nanoscale.

The EAMs have been used to expose PMMA resist. PMMA is normally used for as a resist for electron beam lithography, however it is sensitive at deep UV wavelengths. It has been used because of its high resolution and low out-gassing. Out-gassing – in which gases are emitted from a resist during exposure – sufficiently high would destroy intimate contact between a mask and resist. The drawback with using PMMA is that exposure times are long – 12 minutes because PMMA is a low sensitivity resist, compared with ordinary photoresists that are on the order of 10 seconds. The authors in [24] have said that from the results shown in Figure 2.10 that patterning of 60-90nm features should be possible with transverse magnetic (TM) polarised light. However considering that EAMs are just PAMs with glass stuck between the absorbers, the simulations conducted for PAMs should hold true. Therefore comparable resolution to PAMs should be possible with EAMs.

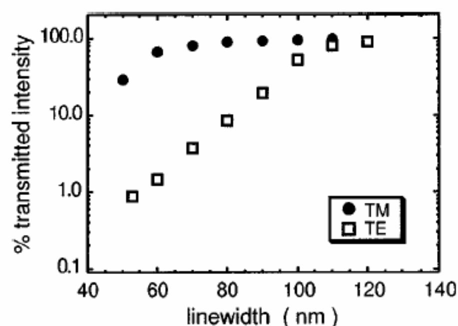


Figure 2.10 Simulation results for EAM masks[24] – using 220nm UV light.

2.2.3 Light Coupling Masks

These masks are made of an elastomeric material – poly di-methyl siloxane (PDMS) (Figure 2.11). This material is made up of long polymer chains and by its very nature it self-adheres to photoresist so that conformal contact is enabled without the use of vacuum. The conformal contact also ensures good light coupling, as the index of refraction of PDMS is matched to that of the resist – ensuring optimal light transmission. However because PDMS is like silicone rubber, it is highly elastic/conformable in nature which means that in-plane distortion may be large [31]; this would need to be characterised. Additionally, features have been known to sag, and high resolution features with high aspect ratio can stick together. In its current form modelling has predicted a limit for LCMs of 60nm[28] due to cross talk of light beams passing through the light coupling structures. Such issues limit the use of LCMs for high resolution features in the near field. However an augmentation to the LCM may solve the latter issue – surface plasmon illumination, outlined in the next section.

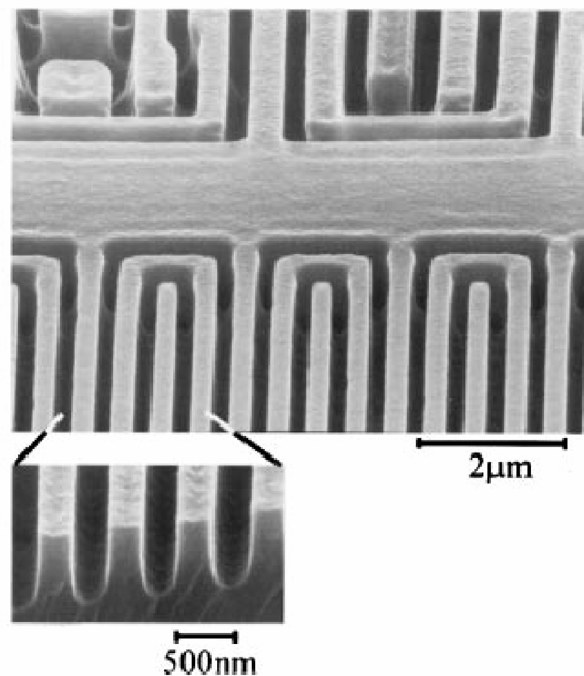


Figure 2.11 500nm pitch light coupling mask[25], with 5nm gold in recesses (not visible).

2.3 Extending ENFOL

This section briefly describes two promising methodologies that extend evanescent near field lithography for higher resolution or enabling proximity exposures. The first is surface plasmon illumination which has been simulated to give high resolution results from oblique illumination of a gold coated LCM which generates an SPP which propagates through the LCM to the resist below [33]. The second is planar lens lithography (PLL) in which a slab of silver is patterned onto the under side of a PAM; this amplifies the evanescent near field radiation through negative refraction such that proximity printing *relaxing the intimate contact requirement* is possible. The operation of has been successfully demonstrated through experimentation and simulation.

2.3.1 Surface plasmon illumination

The surface plasmon illumination methodology simulated by O.J.F Martin is based on the use of an LCM mask with a patterned metal film on top and oblique illumination (Figure 2.12). Oblique incident light impinges on a 40nm thick gold layer, this couples with surface plasmons to produce SPP's which then produce near field radiation that illuminates the resist beneath the mask. Simulations based on Green's tensor techniques show the replication of 20nm features using 630nm wavelength light [33] – this implies a resolution of λ/31.5.

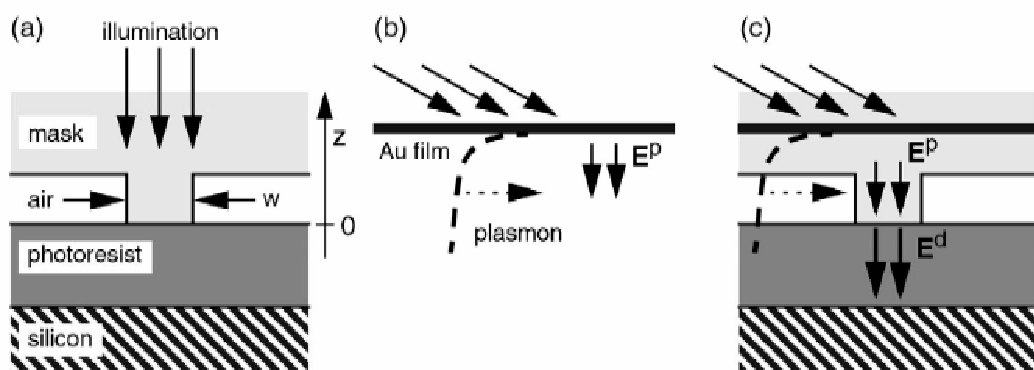


Figure 2.12 Surface plasmon illumination[33] : a) normal LCM mask operation b) SPP's generated under a metal film c) LCM mask with metal film above and generated SPP's stimulating near field radiation to the exposure of resist beneath the mask.

However, drawbacks to the surface plasmon illumination are that it is not able to produce uniform features greater than ~ 1/2 the illuminating wavelength as far field components are not present to correct a dip in intensity for larger features (Figure 2.13). It can be seen that for feature sizes of 100nm in Figure 2.13 a) and 80nm in Figure 2.13 b) that dips in intensity are present. Therefore in order to pattern large features and small features, two masks would be needed, where the mask for the large features need not be an LCM mask.

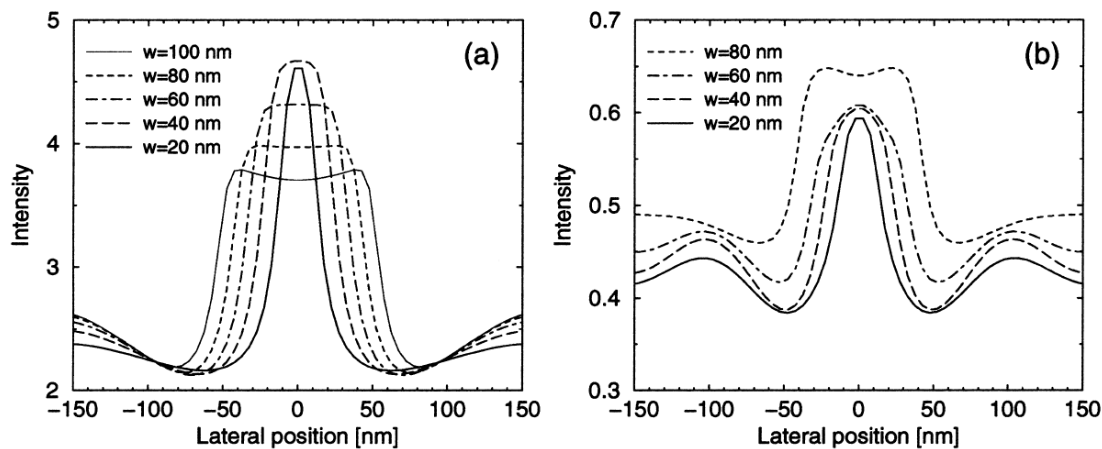


Figure 2.13 Electric field intensity 10nm into resist a) 630nm wavelength b) 248nm wavelength [33].

2.3.2 Planar Lens Lithography

Planar lens lithography (PLL) utilises the discovery of negatively refracting materials, which were first proposed by Russian physicist Vessalago and later further developed and discovered by British physicist John Pendry. Negative refraction works because some materials swap the orientations of the electric and magnetic fields of light. Materials such as specially aligned gratings and metal structures negatively refract microwave frequencies, and silver negatively refracts optical wavelengths at λ = 334nm. PLL utilises a planar silver lens placed beneath an ENFOL mask as shown in Figure 2.14.

Simulations show that when near field radiation is emitted from the bottom of the ENFOL mask this stimulates the resonance of surface plasmons on the silver lens which then re-radiate the near field radiation, but at a higher intensity, this is depicted in Figure 2.15.

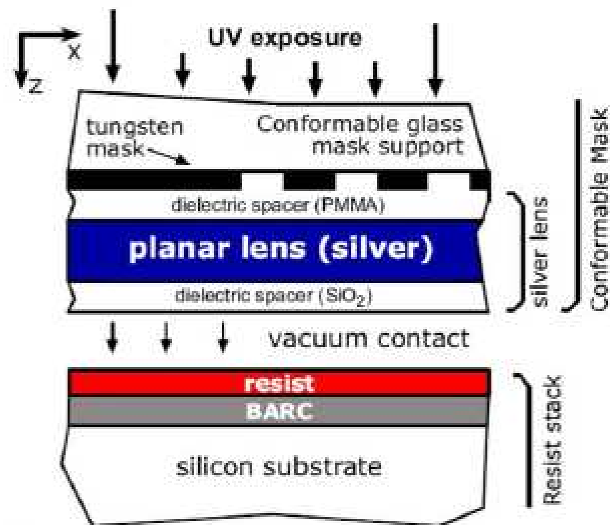


Figure 2.14 Planar lens lithography patterning schematic [34].

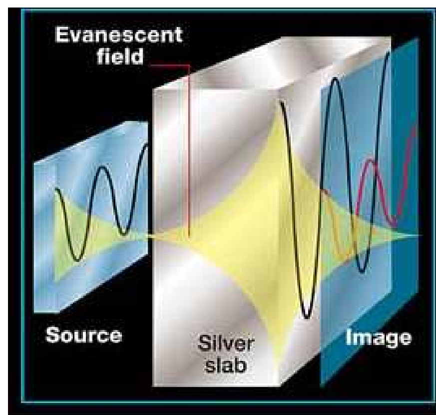


Figure 2.15 silver lens amplification of the evanescent near field radiation.

The imaging of the evanescent near field through a silver lens has successfully been demonstrated as reported in [35]. The performance of PLL compared to that of ENFOL is shown in Figure 2.16. Planar lens lithography has successfully demonstrated negative refraction for ultraviolet light. It has also demonstrated a potential process for relaxing the intimate contact constraint in ENFOL. As can be seen in Figure 2.16 the images of PLL look somewhat defocused compared to that of ENFOL. This is due to losses in the silver which reduce the quality of PLL's results. This could be improved through better planarisation of the silver lens [36].

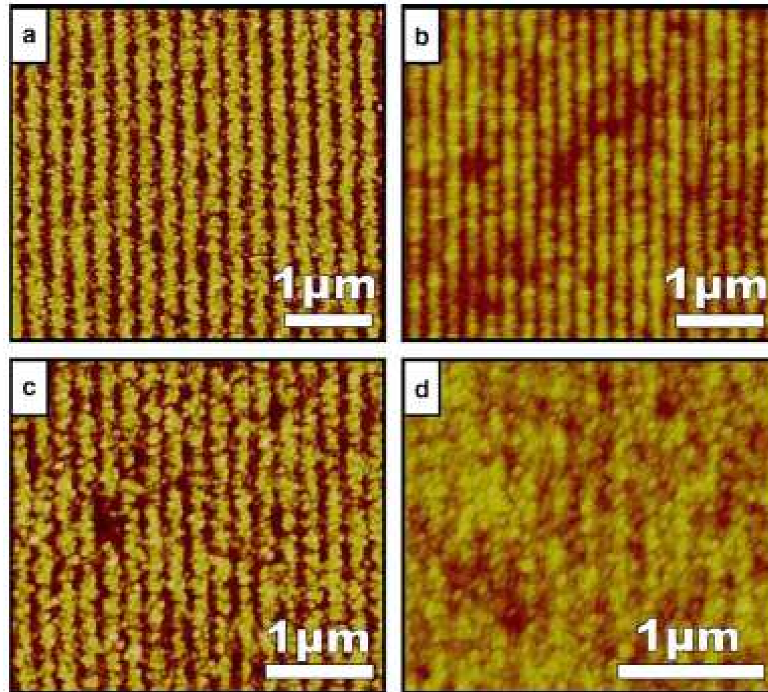


Figure 2.16 Atomic Force Microscope scans of 250nm features (a,b) and 200nm features (c,d) for ENFOL (a,c) and Planar Lens Lithography (c,d). The height scale is 50nm [36].

With the exception of PLL, in which it is hoped that intimate contact constraint will be relaxed, lithography using the near field requires the use of a conformable mask that must make intimate contact with the substrate. The in-plane distortion that results must be characterised and this requires the use of a well equipped and maintained laboratory. The use of certain equipment and their associated processes are crucial to this aim, the details of which are given in the next chapter.

Chapter 3: Experimental Techniques: Apparatus, Processes and Manufacturing

It is desired to produce an ENFOL mask with patterned features that can be used to measure the in-plane distortion resulting from patterning such conformable mask features to silicon substrates. In order to manufacture such masks, measure them, use them to pattern silicon substrates which are then themselves processed and measured – many processing steps must be taken and the use of high technological apparatus is essential. The first section of this chapter is devoted to the description of such apparatus: the electron beam lithography system (with metrology software), photolithographic mask aligner, reactive ion etcher and sputterer. This is followed by second section providing details of all the experimental processes including the use of the equipment and remaining laboratory apparatus involved.

3.1 Essential Apparatus

The process of producing an ENFOL mask is a series of steps that are built up on each other – such that if any step goes wrong, the entire process must be restarted from the beginning with new materials (as occurred for a many masks). Each of the processes and involved apparatus for all manufacturing and processing steps are outlined in this chapter, however a few pieces of equipment warrant additional description due to their complexity, high technology within or crucial enabling factors for this work. These machines include: electron beam lithography (EBL) system, the mask aligner for photolithography, the reactive ion etcher for subtractive patterning, and the sputterer for the deposition of thin films. The EBL is the key enabling apparatus for this thesis, as it is the machine that enables the metrological study of the ENFOL patterning of substrates, through use of its electron microscope, laser-interferometer precision-positioning stage and metrology software module.

3.1.1 Raith 150 Electron Beam Lithography System

An EBL system is at the beginning of any mass produced integrated circuit manufacturing process. Photolithography may be the mainstay of the semiconductor industry due to its high throughput, but it is an EBL machine that writes the masks necessary for use by photolithographic machines. It is for this reason and metrological use of the EBL that places this apparatus at the initial creation of any features (masks) and at the final measurement of all resultant patterned features (masks, silicon substrates) in this work

The Raith 150 EBL available in the Electrical and Computer Engineering (ECE) Department's micro fabrication laboratory is a fully integrated system that consists of a Leo 1500 series electron microscope with added laser-interferometer stage, robotic chamber loading, pattern generator hardware to control the beam and the Raith software package (Figure 3.1).

Figure 3.1 The ECE department's Raith 150 electron beam lithography system. From the left: a rack contains the control electronics and laser interferometer source; the Raith computer; the vacuum chamber of which the beam column can be seen extending through the top; the work station of which the joystick stage control can be seen in front of two screens: one connected to the Raith computer and the other connected to the Leo computer controlling the Leo electron microscope.

Electron Microscopy

The heart of this system (and any EBL system) is the electron microscope. An electron beam is produced in the electron microscope by a combination of two effects: the first is that of thermal excitation of the tungsten filament by heating it to between 2600-3000K resulting in electron emission [37] (Figure 3.2).

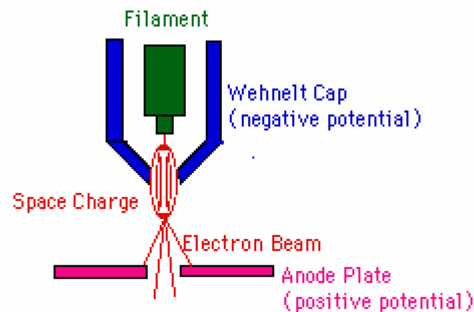


Figure 3.2 Electron source [38].

The second is that of field emission upon which a high voltage (200-100kV) is applied between that of the filament (cathode) and a nearby plate (anode). The filament electrons are highly attracted to the positive plate, and gather at the tip of the filament closest to the positively charged plate. The sharper the filament tip is, the more intense the electric field (the 'lightning rod' effect) – resulting in greater ease of ripping electrons from the filament via electro static force. This is such that much more expensive filaments with very sharp tips require no heating – utilizing field emission alone and deliver the highest current.

Once emission begins to occur the resultant stream of electrons is constrained by the Wehnelt cap beneath the filament charged to ~ -500V which repels the electrons to a small 1mm hole in its base (Figure 3.2). Only electrons of certain energy pass through the hole and are accelerated toward the positively charged anode plate resulting in a monochromatic electron beam. With the beam created the electron microscope then operates on the principle that a beam of electrons can be manipulated by magnetic fields in the same way that visible light is manipulated by use of lenses. Figure 3.3 illustrates this for a scanning electron microscope (SEM), in which grey boxes indicate the electromagnetic coils that manipulate the electron beam (blue) emitted from the electron source.

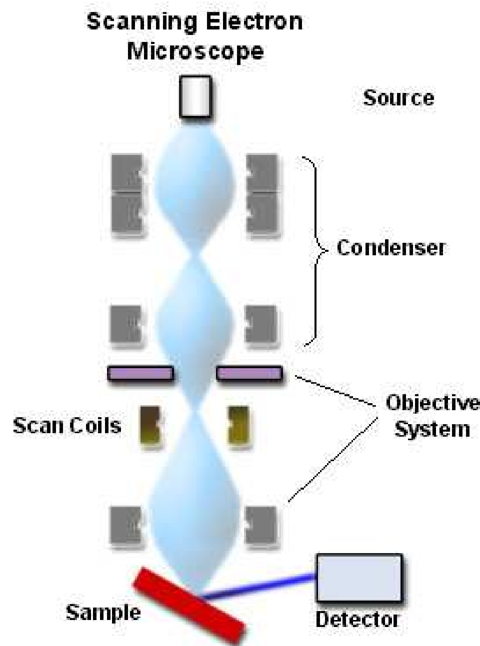


Figure 3.3 Electron optics in a scanning electron microscope, the grey boxes indicate the electromagnetic lenses which direct and shape the electron beam (blue). Purple boxes indicate the user selectable objective aperture [8].

The electromagnetic coils are known as electron optics and can be thought of as lenses. The first set of lenses, condenser lenses, form the beam and limit its current, these are followed by a user selectable aperture which eliminates high energy electrons from the beam and further limits the current; next a set of scan coils sweep the beam in a grid fashion and finally the objective lens focuses the scanning beam onto the sample.

Imaging then proceeds through the detection secondary electrons resulting from interaction of the beam with atoms on and close to the surface of the sample. The secondary electrons are due to the incident beam ionising atoms within a depth of 10nm of the sample surface, resulting in the emission of electrons. The yield of these electrons is affected by changes in the surface topology of 10nm, such that higher areas are brighter than lower lying areas of the sample and thus providing topological contrast [39]. SEM then provides a continuous image by constantly raster scanning the sample surface.

Electron Beam Lithography

With a working SEM three important additions must be made to create an EBL system. The first is that of electronic control of the scan coils directing the beam and the second is the ability to blank the beam, by use of electrostatic blanking plates (Figure 3.4). With the addition of such control, the SEM is able to write any desired series of shapes, with the ability to switch off the beam between forms and thus is able to produce truly arbitrary patterns.

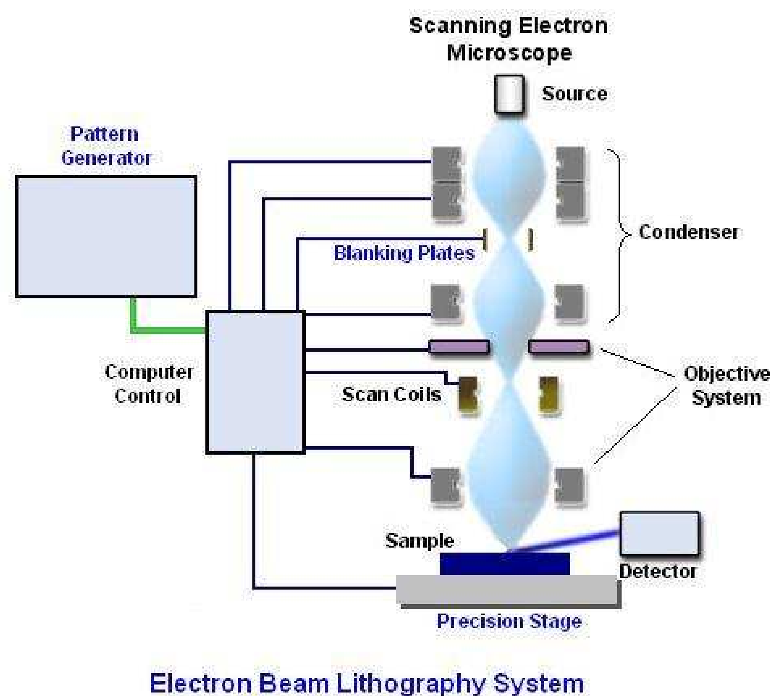


Figure 3.4 An electron beam lithography (EBL) system. A pattern generator, blanking plates and a precision stage added to a computer controlled SEM enable it to be used as an EBL. Image modified from ref [8].

The electronic control of the scan coils and that of the beam blanker is in the form of specialised hardware comprising the pattern generator. This is employed for high speed control to ensure that the time that the beam is not exposing the substrate is kept to an absolute minimum. This in turn ensures that the throughput of the system is limited principally by the size and current of the electron beam rather than delays in its positioning. The last important addition is that of a laser-interferometer controlled precision stage (Figure 3.4). The stage in the Raith 150 is capable of 2nm movement precision [40] by use of the position information provided by the laser-interferometer and fine movement provided by piezo electric actuation.

With arbitrary control of the beam, and precise stage control, the EBL is used to expose electron sensitive resists made of polymers such as polymethylmethacrylate (PMMA) to produce patterns such as photolithographic mask features. The process is similar to that of photolithography with the electron beam breaking the polymer chains into easier-to-dissolve smaller lengths, which are developed away revealing patterned features. Further detail on the exposure process is provided later in this chapter. Finally, the Raith 150 EBL is capable of high resolution patterning with pattern features of <20nm produced (Figure 3.5). In the next subsection and last but certainly not least on the Raith 150, the metrological use of the system is outlined.

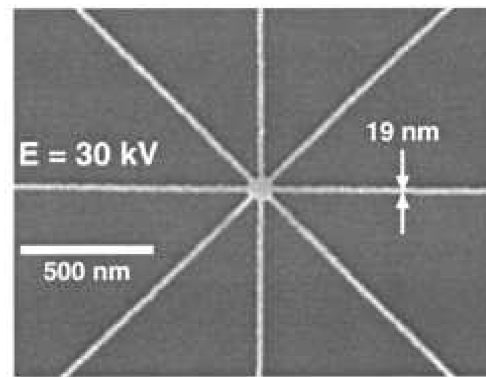


Figure 3.5 High resolution patterning (metal on silicon) produced by electron beam lithography in the Raith 150 system [41].

Metrological Use of the Raith 150 EBL system

The key features endowing the Raith 150 with the ability to be used for metrology are: first and foremost the laser-interferometer stage, the second is that of the high resolution of the electron beam lending to precision measurements through a sub-20nm spot beam, and lastly the metrological software module allowing the automatic measurement of features by ascertaining contrast fringes and thus enabling thousands of measurements to be able to be taken. The metrology module is able to do this by reading intensity from individual line sweeps of the electron beam such that changes in intensity within a certain threshold window (which constitute feature edges) enable the threshold algorithm to decide that there has been a feature edge and record its position (Figure 3.6).

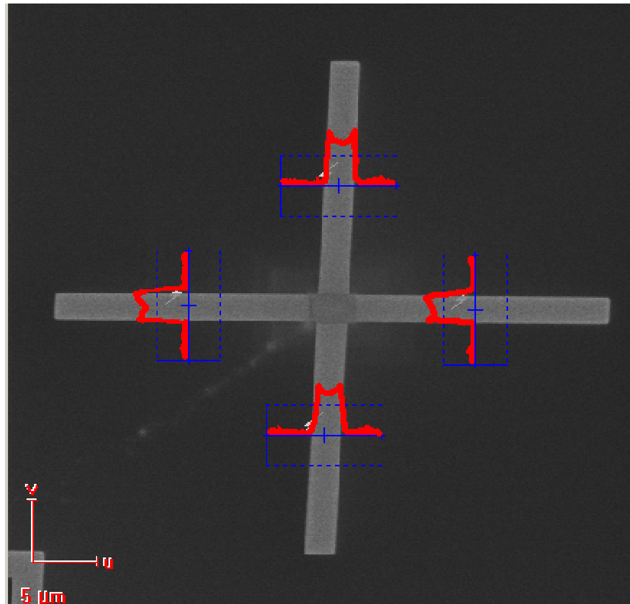


Figure 3.6 Scans of line features (showing intensity) superimposed upon where they were taken.

The module is also able to take the average of several line scans of the same feature and thus patterned features suffering from line edge roughness (LER) may be successfully measured. Figure 3.7 shows this such that measuring a feature has two dimensions, the main dimension across the feature edges and the second dimension determining the breadth across which multiple line scans will be taken. The metrology module is discussed in detail in the next chapter.

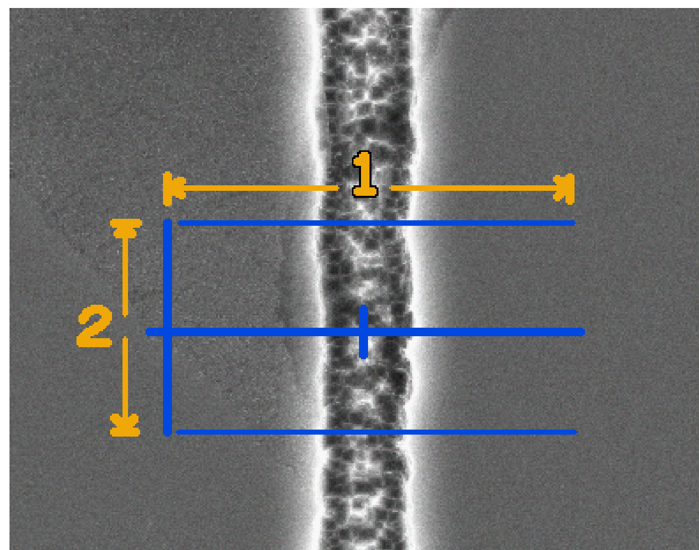


Figure 3.7 2D scan of a line with significant LER. The dimension along 1 is the main dimension of the scan, whereas the resolution and length set along the dimension 2 determines the amount of LER averaging.

3.1.2 The Mask Aligner for Photolithography

The ECE department micro-fabrication laboratory employs a Karl Süss MA6 mask aligner for performing photolithography[7] (Figure 3.8). The MA6 houses a UV light source than can either be broadband or deep UV, optics to collimate the UV light (so that it travels in parallel and arrives normally incident upon the mask), a vacuum mask holder and a substrate stage with interchangeable chucks (Figure 3.9). The MA6 also features dual Olympus optical microscopes for mask substrate alignment and is capable of one micron overlay accuracy. Parameters adjustable on the MA6 include: exposure time, strength of vacuum, separation height of the mask to substrate (through the wedge error correction offset), pre vacuum and post vacuum settling times.

Figure 3.8 Karl Süss MA6 mask aligner with operator in the yellow room of the ECE micro-fabrication laboratory.

In operation the sample is placed on the stage, is held by vacuum and the mask is brought down in contact. During this stage the MA6 performs wedge error correction (WEC) pressing the mask against the substrate in order to align the two planes of the mask and substrate in ‘z’. Following this the MA6 lowers the stage an adjustable height to facilitate alignment of the mask and substrate.

[7] The principle of photolithography is outlined in Chapter 2 this thesis

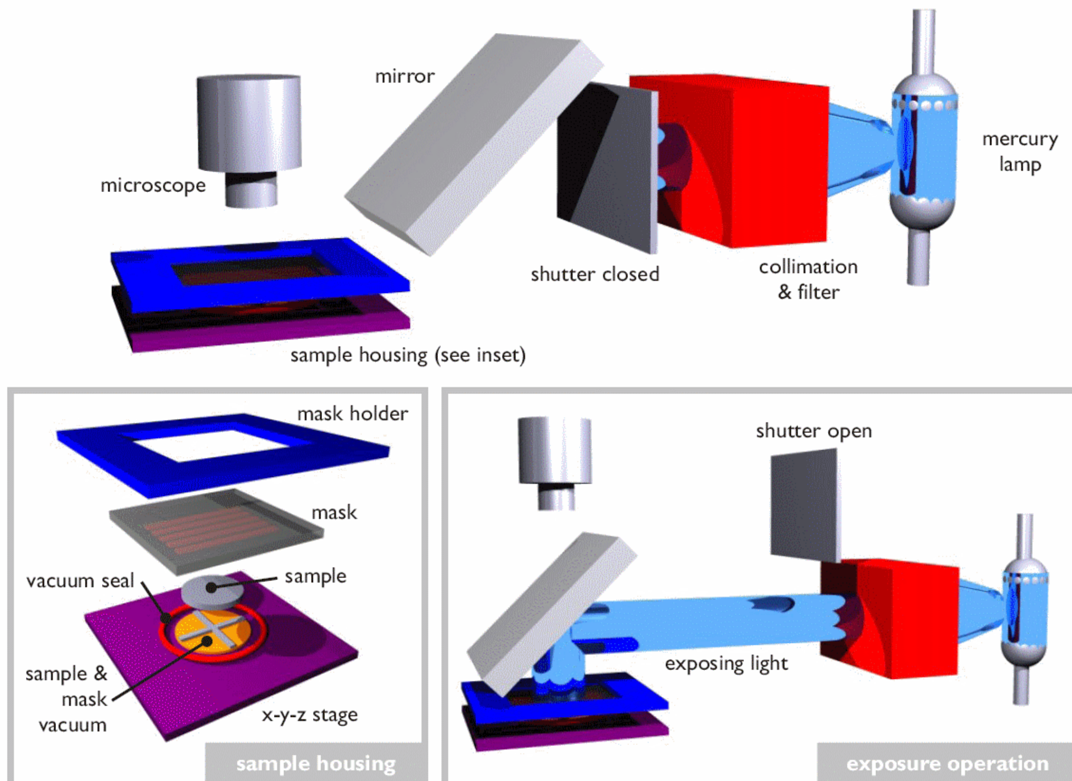


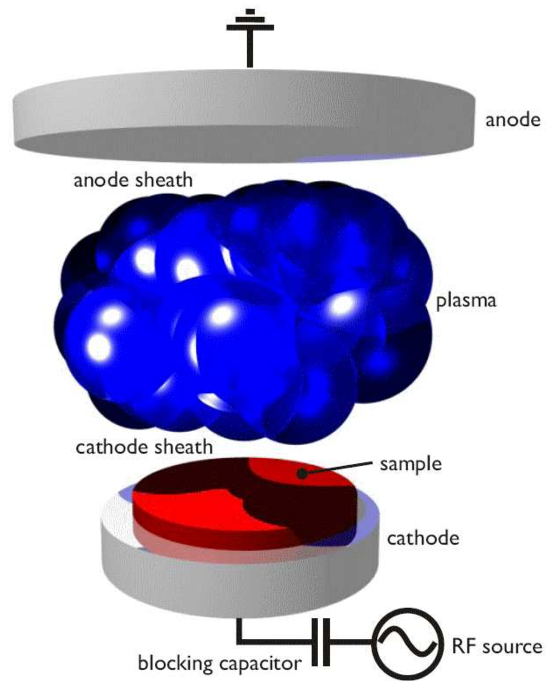
Figure 3.9 Operation detail of Karl Süss MA6 mask aligner [42].

Prior to vacuum the sample stage chuck inflates a circular rubber ring which presses up against the mask (or a mask holder plate with ENFOL mask centrally located for this work) creating a seal and a temporary vacuum chamber (Figure 3.9). When an exposure is ordered the microscope heads hinge up and out of the way, a mirror is brought forth to direct the UV light through the mask and the shutter is opened initiating exposure (Figure 3.9).

3.1.3 The RIE

The laboratory possesses an Oxford Instruments Plasmalab 80 (Figure 3.10a) reactive ion etcher (RIE) for the subtractive patterning of materials. Reactive ion etching is a process by which material is removed from a material substrate chemically and/or physically by ionic bombardment. The path of the ions is normal to that of the substrate and thus with a masked substrate for selective etching (such as a patterned photoresist-coated silicon wafer), the sidewalls of the resultant etched features are straight and smooth.

(a)



(b)

Figure 3.10 (a) Oxford Plasmalab 80 Plus reactive ion etching chamber alongside liquid nitrogen tank (b) The reactive ion etching process. (image in (b) from [42]).

This is opposed to that of wet etching in which the material is agitated in a reactive solution, which produces curved (isotropic) side walls as the solution undercuts the photoresist masking.

Reactive ion etching takes place in a vacuum chamber in which the sample to be etched is placed on an electrically isolated wafer platter. The rest of the chamber is electrically grounded. Gasses are emitted into the chamber e.g. sulphur hexafluoride (SF_6) and radio frequency (RF) energy typically 13.56MHz [43] is applied to the gas stripping electrons from the gas molecules producing a plasma (Figure 3.10b). These free electrons are electrically accelerated up and down the chamber by the RF field and hit the chamber walls or the wafer platter. Electrons striking the chamber walls lose their energy to ground and do not affect the electronic state of the system, however electrons striking the isolated wafer platter build up a net negative charge on the isolated wafer. This negative charge develops a negative bias to the wafer platter attracting the positive ions of the reactive plasma. These ions drift toward the samples to be etched, and react chemically with surface of the sample but can also physically etch due to the high kinetic energy of the incoming ions.

3.1.4 The Sputterer

Positioned near the RIE in the laboratory is the Edwards Auto-500 magnetron sputterer (Figure 3.11a). The sputterer facilitates the coating of samples by liberating atoms from target materials through the use of a magnetically confined plasma. Sputtering is a precision technique for the deposition of thin films and has the advantage over traditional evaporation that the target needs not to be heated, facilitating the use of many more metallic/non-metallic compounds as coatings that would otherwise be very difficult to evaporate.

The principle of operation the sputterer is similar to that of the RIE in that a plasma is induced, and atoms are liberated from the target via ionic bombardment. In the sputterer however the process is physical through the use of Argon ions, and the plasma is magnetically confined above the target (Figure 3.11b). The incoming ions bombard the target causing some of its surface atoms to gain enough energy (through momentum transfer) to evaporate off the target and travel towards the rotating sample holder above, and hence coating the samples.

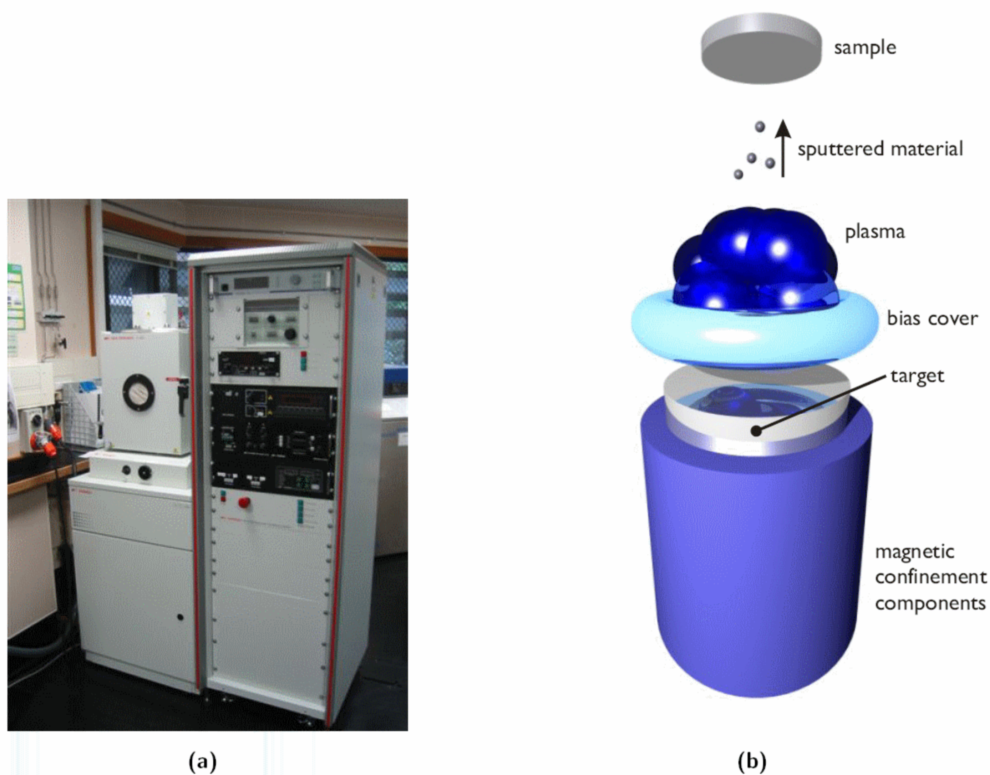


Figure 3.11 (a) Edwards Auto-500 magnetron sputter (b) the sputtering process [42].

This section has outlined the principles of operation of the most complex apparatus used for this work in the ECE micro-fabrication laboratory. With this basis in place the next section gives detail on all the processes involved in the manufacture of ENFOL masks, the processing of the silicon substrates and the use of apparatus at each step.

3.2 Processes

3.2.1 Manufacture of ENFOL Conformable Masks

As discussed previously, it is necessary to produce masks for ENFOL that are made of a conformable material to facilitate intimate contact of mask and substrate, enabling the exposure of the substrate with the evanescent near field. Borosilicate glass cover slips were chosen for the conformable mask as they are available in 100μm thick 22x22mm polished slices. Upon the glass a coating of metal is applied to be used as the shadow (absorber) areas of the mask. The material chosen was tungsten (W) as it easily dry etched via RIE as well as being a good absorber. An alternative absorbing material would be nickel-chromium alloy, however this is not readily etched requiring a lift-off process and therefore would expose the mask to a potential source of particulate contamination. Once the mask is coated in tungsten it is spin coated with PMMA electron resist. The pattern is then written onto the mask in the EBL and developed. Following this the pattern is transferred into the tungsten via RIE. A mask holder is then made ready which consists of a well polished mask plate providing a highly smooth surface for mounting. The final step is the mounting of the mask onto the mask holder using UV curable glue.

There are a lot of steps and processes involved in carrying out the manufacture of ENFOL masks as described above. The following text is divided into sections describing each process step. The sections also serve as an account of the experimental work performed in the laboratory. Figure 3.12 illustrates the main processing steps in the manufacture of 100μm thick 40nm tungsten coated conformable masks. The details of the process steps are given in the following subsections.

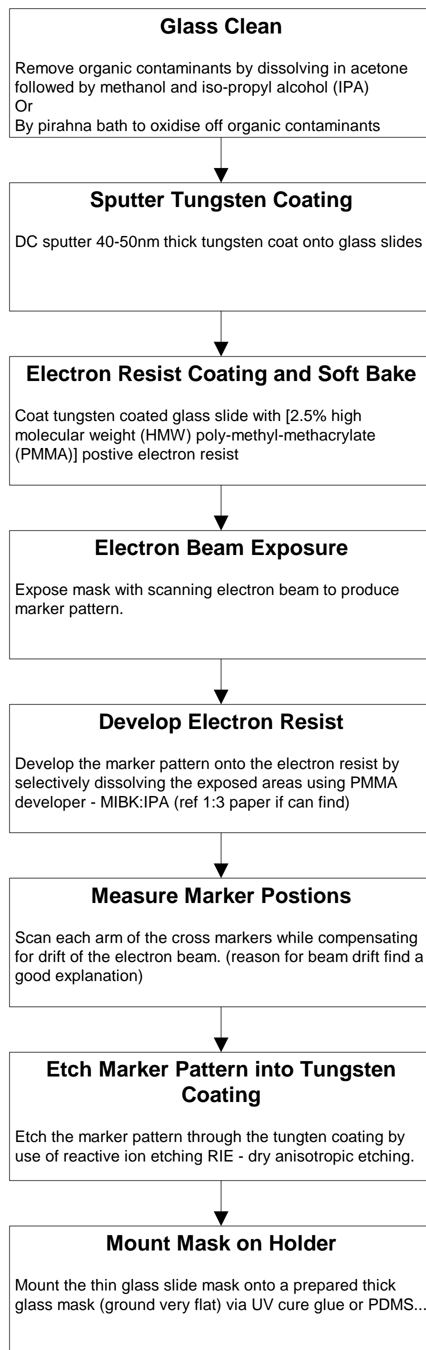


Figure 3.12 ENFOL manufacturing and processing steps.

Glass Coverslip Cleaning

For experiments with conformable masks, it is necessary to have all samples as clean and dust free as possible. This is because any dust particles will have an adverse effect on the measurement of mask pattern distortion by producing local distortions. The 22x22mm 100μm thick borosilicate glass cover slips are cleaned in organic solvents

in an ultrasonic bath. The slips are first cleaned in acetone for two minutes followed by methanol and isopropyl alcohol consecutively for one minute. Contaminants are held by strong electrochemical or Van der Waals forces, the ultrasonic vibration at ~900kHz is effective in shaking the contaminants off the surfaces into the non-polar solvents. Following this the cover slip is then carefully blown dry with nitrogen gas, and baked for 15 minutes at 95°C (to evaporate any remaining solvent residue). An alternative method of cleaning is that of a piranha bath in which 1 part of hydrogen peroxide has 5 parts of concentrated nitric acid added to it, the resultant reaction oxidises organic contaminants such that they are then dissolved into the polar solvent and are subsequently washed away when rinsed with deionised water.

Tungsten Deposition via Sputtering

To coat glass cover slips in tungsten, cover slips are attached to a rotating sample holder which is attached to the top of the sputterer chamber. The slips are taped on using foil tape in such a manner that a pattern with handedness is produced on the edges. This is necessary as the slips once coated in W are not distinguishable W side up or down – necessitating scratching, which is not desired.

Once the slips are loaded, the sputtering chamber is pumped down to a vacuum of at least $1x10^{-5}$ Torr. Following this Argon gas is released into the chamber and the flow allowed to stabilise for 5-10 minutes. The thickness of film desired is entered and shutter control set to the DC target shutter. Once the chamber is ready, the DC sputtering power supply is engaged, this induces a plasma from the argon gas in the chamber by applying a large voltage bias to a tungsten target in the chamber. The power to the DC plasma generator is set at 300W and after two minutes to stabilise the plasma, sputtering is enabled on the control panel – which translates away the shutter allowing the plasma ions attracted towards the negative bias of the target to bombard it.

A thickness of 40nm is chosen as this yields high contrast between shaded and unshaded areas of photoresist while still providing adequate intensity [26]. Once the crystal monitor reads the thickness to have been met, the sputterer closes the shutter to

block the plasma ions from hitting the target. The sputtering power is then quickly switched off manually to prevent further film build up as the shutter does not prevent all ions hitting the target.

Electron Resist Spin Coating and Soft Bake

In order to pattern features onto the mask via electron beam, the mask must be coated in electron sensitive resist. Such a resist consists of a polymer whose long chains can be broken (or cross linked in the case of a negative resist) upon exposure to incident electrons. The resist used is poly-methyl-methacrylate (PMMA) a positive electron resist. Specifically 2.5% high molecular weight (HMW) PMMA is used. To coat a slip in PMMA a spinner is used (Figure 3.13).

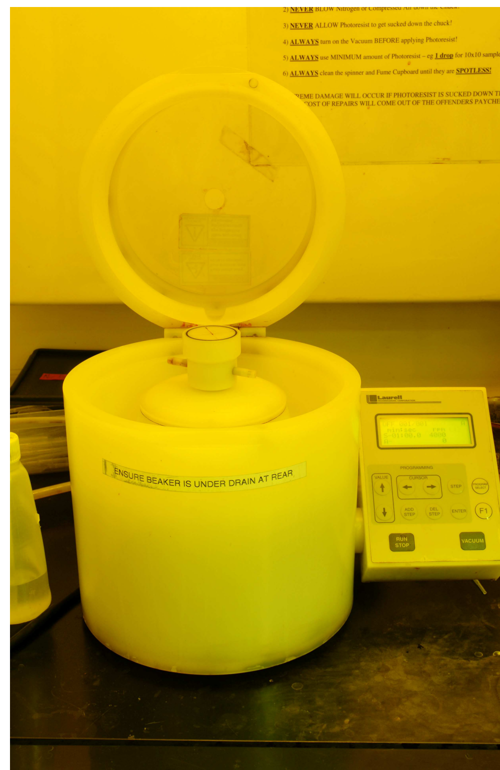


Figure 3.13 Laurel Technologies spinner in fume cupboard within yellow room.

This apparatus is basically a vacuum chuck that spins at high speed such that liquid placed on a sample held in the chuck is spread across the sample evenly via centrifugal reactive force. The procedure is fast, and varying the spin speed governs the thickness of the resulting film. The PMMA is spun on using the spinner at

4000rpm for 60 seconds. This results in a PMMA coating that is ~100nm thick. Upon coating the slip is then placed in an oven heated to 185°C for 20-30 minutes which allows the film to harden to a degree and allows moisture to escape.

Electron Beam Lithographic Exposure

The mask feature pattern (Figure 3.14b) is laid out using mask layout software L-Edit from Tanner Tools. The black features on the mask layout are arrays of marker features (Figure 3.14a), the green circle indicates the 18mm diameter hole cut in the mask plate holder and therefore the usable surface area of the mask, finally the red lines are 10µm thick long features that are patterned onto the mask to facilitate location of the fine features from low magnification in the SEM prior to measurement. Upon completion of the drafting of the mask pattern the L-Edit software is used to output a file format that is acceptable for the Raith 150, GDSII [44].

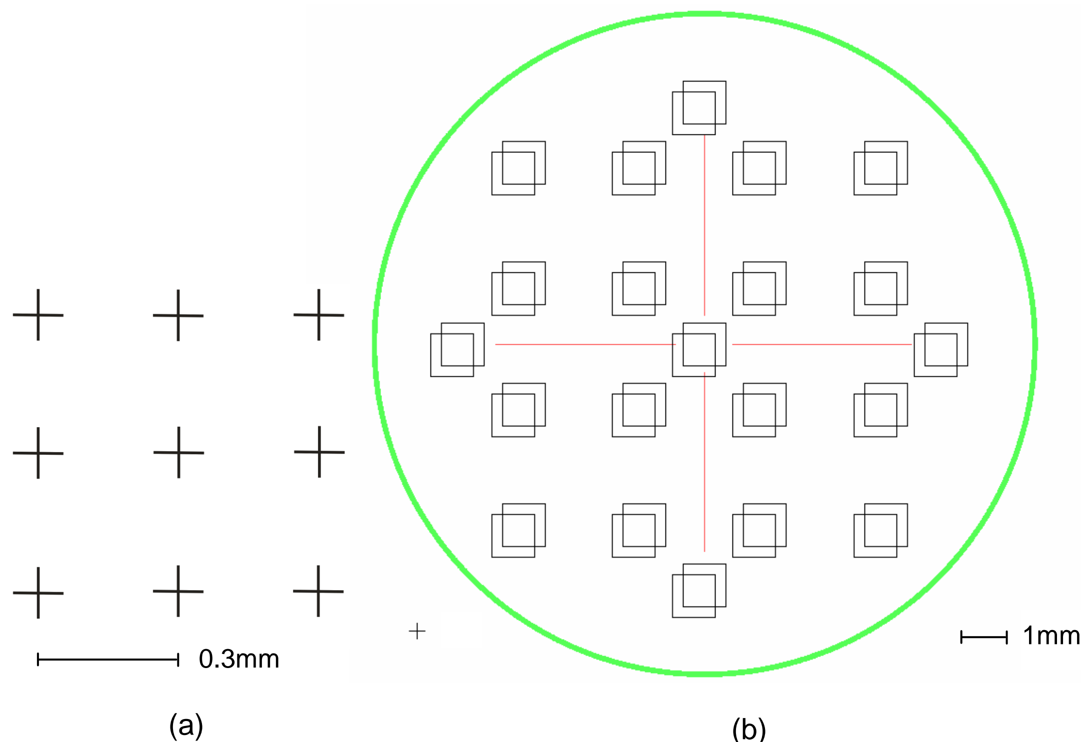


Figure 3.14 (a) Marker feature array consisting of 1µm thick 20µm high markers spaces 0.3mm apart. (b) Layout of marker feature arrays. The overlapping squares represent thick and thin cross markers respectively. The thin crosses – 100nm are patterned to decrease charging effects on scans, with the large crosses (1µm thick) patterned in case the thin crosses do not come out. The red lines are thick finder lines used to locate the centre pattern.

With the mask pattern drafted and the PMMA on tungsten coated glass mask ready both are loaded into the Raith 150 system, the former through the robotic loading system and the latter via portable data storage due to the isolation from the network of the Raith computer. This isolation measure was taken to ensure that control of the multimillion dollar system is not compromised by malicious software that may be present on the university intranet.

With the mask loaded and the chamber brought down to vacuum, the electron column is excited at 10kV and an aperture of 30µm chosen. A working distance of 6.5mm is then set lowering the microscope head. This provides optimal patterning through minimisation of intra-field distortion [41]. The stage is then moved so that the faraday cup (a drilled hole) on the stage is beneath the beam to measure its current. Upon measuring the current, aperture alignment proceeds where the beam column aperture is aligned about the electron beam by wobbling the focus so that the image remains stationary. A misaligned aperture results in the beam not being normally incident upon the surface. Following this the beam is corrected for stigmation, and then the stage is levelled using piezo electric actuators. A coordinate system is then created that is aligned to the orientation of the mask with components 'u' and 'v'. This is then used to transform the movements of the stage in 'x' and 'y' so that the movement of the stage corresponds to movements in 'u' and 'v'. Thus aligning the motion of the stage to the orientation of the sample.

An important step for both EBL and metrology is the alignment of the write field. This is because the field of the electron beam (due to fluctuations and imperfections in the electron optics) can become misaligned with the stage such that the position, orientation and scale of the beam write field is incorrect. The write field is aligned with that of the stage via a series of image scans made about a discernable feature on the sample surface. The stage is first moved so that the feature is directly beneath the beam, the stage is then moved known amounts and images taken, by which apparent movement of the feature is determined automatically by the Raith software or manually by the user selecting the feature on the image. Repeating this three times for different movements of the stage the Raith software determines the orientation, position, scale and rotation of the write field and adjusts it accordingly.

With the write field aligned the dosage parameters are set. For 100nm thick 4% HMW PMMA the dosages shown in Table 3.1 were used. Upon setting the exposure parameters the exposure of the PMMA coated tungsten ENFOL mask may begin.

Table 3.1 Exposure parameters.

Draw type	Dose (Coulombs)
Area	100μC/cm^2
Line	220pC/cm
Dot	200fC/dot

Electron Resist Development

The duration of the exposure for the pattern is one hour, and with an additional ten minutes for unloading the mask is ready for developing. The PMMA exposed to the electron beam has had its polymer chains broken making it more readily soluble in the electron resist developer methyl-iso-butyl-ketone (MIBK) than unexposed areas. These exposed areas in the shape of marker features are selectively dissolved away revealing naked tungsten beneath. Development takes place in 1:3 MIBK:IPA at 23°C for 30 seconds following rinsing in IPA for 15 seconds. Note: to ensure the temperature is the same for each development, the beaker containing the MIBK:IPA is handheld to heat it to 23°C, with a duration of approximately two minutes.

Following this measurement of the developed marker features proceeds after reloading the mask into the EBL and repeating the steps for levelling, write field alignment etc. The detail of this measuring process is presented in the next chapter.

Pattern Transfer by Reactive Ion Etching

Once the marker features have been measured, the next step is to transfer the developed PMMA marker pattern into that of the tungsten layer beneath. The developed PMMA has left holes in the shape of markers revealing naked tungsten beneath. This tungsten will be removed by bombardment of chemically reactive ions. The unexposed, undeveloped PMMA markers will mask the tungsten that surrounds

these markers from the reactive ion attack. With the PMMA providing masking, the naked areas of tungsten are selectively removed in the shape of the marker pattern. This reveals the glass areas beneath the tungsten, creating a shadow mask. The recipe used for the dry etching of the tungsten is shown in Table 3.2 [45]. This recipe uses a combination of reactive etching with sulphur hexafluoride (SF_6) with physical bombardment from the argon ions.

Table 3.2 Tungsten etching recipe [45].

Material	Etch Mask	Etch Rate	Gases	Flow Rates	RF Power	Temperature	Process Pressure
W	PMMA	50nm/min	SF_6/Ar	10/35 sccm	65	313K	0.19 Torr

Mounting of ENFOL Mask to Holder

Once the mask has been etched it is then ready for mounting upon the mask holder plate with 18mm hole drilled in the centre. The mask is then glued to the mask plate holder using a UV curable polymer (Dymax 621 adhesive) and placed beneath a UV for one hour. Upon completion the mask is ready for use in the Suss MA6 mask aligner for photolithography.

3.2.2 Silicon Substrate Processing and Patterning

Silicon substrates must be specially prepared for use with the ENFOL mask in order to avoid particle contamination and ensure uniform photoresist coverage. The first step of this process is the selection of prime silicon wafer with a surface roughness on the order of ~1nm. Such a high degree of flatness is provided through the use of chemical/mechanical polishing the wafer factory and is essential in this study as it provides a flat foundation to measure distortion upon. Prior to any processing the prime wafer is cleaved into small samples, however this process exposes the samples to particulates and thus the samples must be meticulously cleaned. Following this the samples are spin coated in high resolution i-line photoresist and baked. Photolithographic pattern transfer then proceeds with the samples exposed to UV radiation while in contact with the ENFOL mask.

Once a photoresist-coated silicon substrate has been exposed with an ENFOL mask, it is developed revealing patterned features where no photoresist coats the silicon. However this pattern is not able to be measured using the EBL as the photoresist does not image well under electron beam. The pattern must be transferred into the silicon substrate itself. This is done by use of cryogenic RIE which etches the markers 1.5µm deep into the silicon substrate. The silicon sample is then ready to be measured in the EBL.

The processing and patterning of the silicon substrates as just described involves several detailed processing steps, these are shown in Figure 3.15 and are outlined in the following subsections.

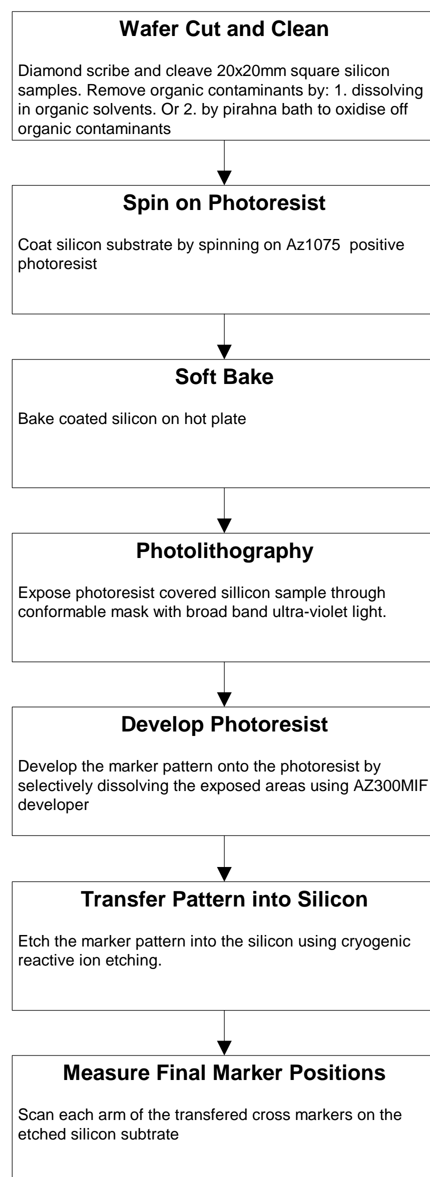


Figure 3.15 Major processing steps for silicon substrates.

Wafer Cut and Clean

The need for cleanliness of the silicon substrates is equally important to that of the mask. 4” prime p or n type prime grade wafers are used, where prime designates that the wafer is unused and is of high quality. Other grades may have undergone topography changing processes and as such are unsuitable for distortion investigation. The wafers are cut up into 20x20mm squares using a diamond scribing station in which the wafer is held by vacuum and a sliding platform moves below a stationary diamond scriber (Figure 3.16).

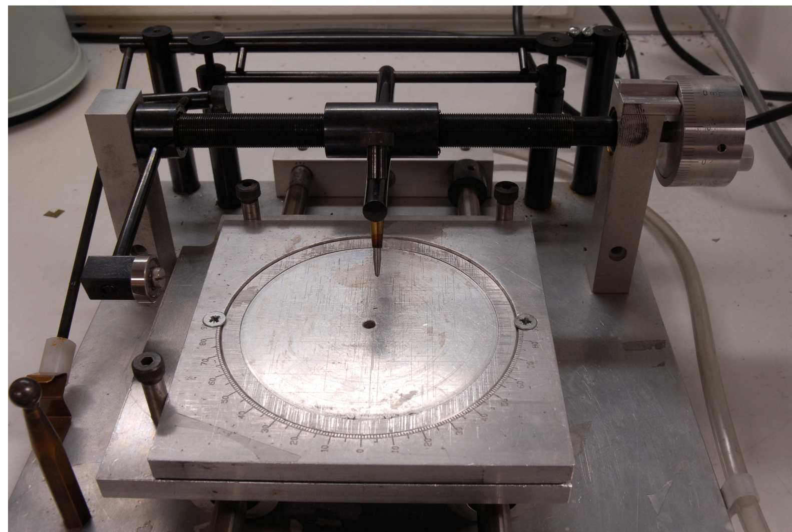


Figure 3.16 Wafer diamond scriber.

The wafers are (100) crystal orientation and therefore cleaves along ‘x’ and ‘y’ with respect to the straight edges made into the side of the wafer. With the wafer aligned correctly in the scribing station, this property is useful as only near the edges need the wafer be scribed to cleave vertical strips of silicon, minimising the amount of silicon dust on the wafer. Additionally the wafer is pre coated in photoresist to protect it from dust, but minimising the amount of dust is useful. The cleaning of the wafers then proceeds in the same manner as that for the cleaning of the glass described in the mask manufacture in the previous section, either through organic solvents or use of piranha bath.

Photoresist Processing

Clariant AZ1075 hi resolution i-line photoresist is spun onto the samples at 4000rpm for 60 seconds providing a coating that is ~600nm thick. Once the photoresist is spun on the sample must be baked at 90°C for 60 seconds on a hotplate to allow the film to harden somewhat, becoming more planarised and to remove moisture from the silicon substrate at the same time.

Photolithography and Subsequent Resist Processing

The ENFOL mask is loaded in the mask holder of the MA6 mask aligner and a silicon sample loaded in the chuck. The W.E.C. offset was set to 50μm for the final experiments providing a vacuum pull-down effect of the mask membrane to substrate ensuring the removal of air between the mask and substrate prior to exposure as the vacuum is engaged. The exposure is set to 8.5 seconds (determined empirically for this resist) and the vacuum is set to hard. Additional parameters are setting the vacuum ramp on and off to 2 minutes.

Following exposure the samples are then subject to a post exposure bake, this step gives the remainder of the energy required to partially complete reactions – bond breaking or cross linking [46]. To facilitate this, the resist is baked on a hotplate at 110˚C for 90 seconds. The development of the photoresist then proceeds through use of a water based developer (Clariant AZ300MIF) which is metal ion free – and hence does not add impurity dopants to the silicon as old developers at one time did. Development takes place for 60 seconds at 23˚C, which is followed by 15 seconds of rinsing in deionised water.

Pattern Transfer by Cryogenic Reactive Ion Etching

This process is the similar to that described in the previous section, except that etching now takes place at -100˚C with SF_6 and O_2 gases with an additional mechanism that produces straighter side walls. Because of the cold temperature once ions react with the silicon surface they freeze and remain there inhibiting further

etching as they are unreactive to the incoming ions, however in the case of surfaces perpendicular to the vertically inbound ions – etching continues as the inhibitor species are physically knocked out of the way by the kinetic energy of the ions. The result of this is that on the side walls of the etch – the inhibitor species remain, reducing the amount of side wall erosion and hence straighter side walls are produced.

Table 3.3 Silicon etching recipe.

Material	Etch Mask	Etch Rate	Gases	Flow Rates	RF Power	Temperature	Process Pressure
Si	AZ1075 photoresist	500nm/min	SF_6/O_2	90/7.5 sccm	150	173K	0.07 Torr

Once the markers have been etched in the silicon they are ready for loading in the Raith where the scanning procedure is exactly the same as the mask marker scanning, except the scan order is laterally inverted. This is necessary as to compensate for the sample pattern inherently being a mirror image of the mask as photolithography is performed.

The detail of the measuring process, from acquiring the measurements involving the setup of automated consecutive measurements of all of the arms of the marker features – to the processing the resulting data is presented in the next chapter of this thesis.

Chapter 4: Data Capture and Analysis

In this chapter, the process by which pattern distortion is measured between an ENFOL conformable mask and substrates is revealed. There are two major stages to this process: data capture and data processing. The first stage involves commanding the Raith EBL to perform thousands of measurements – a scan of each of the cross marker arms to determine the centre of each marker. This involves the use of a program to generate the commands and set the position and resolution of the measuring scans. The metrological software module of the Raith must then be calibrated to detect line feature edges from the measuring scans and avoid false positives. The line edge roughness of patterned features can also be compensated for by use of multiple line scans. Once the scans are complete the second stage is to remove systematic errors from the data. The causes of such range from the non-perfect alignment of the sample to that of the coordinate system assigned to it in the Raith EBL chamber, to missing cross marker features and the unpredictable drift of the electron beam due, to uneven heating and the practical manufacture and operation of the electron ‘optics’. In the following sections the processes and methods to mitigate such errors and how data is captured and processed are presented.

4.1 Data Capture

The measuring process starts with the generation of command files to enable the Raith lithography system to take ordered measurements of the patterned markers on a mask or substrate. This is enabled through use of the *CreatePls* program (see Appendix A) being passed the positions of the markers along with repeated measurements of reference markers to eliminate beam drift. The *CreatePls* program also sets the parameters for the precision and dimensions of measuring scans. Once the command file is ready, the data capture process begins once the algorithm for determining data from scanned features is calibrated to the conditions of the sample. Figure 4.1 shows the data capture process from start to finish.

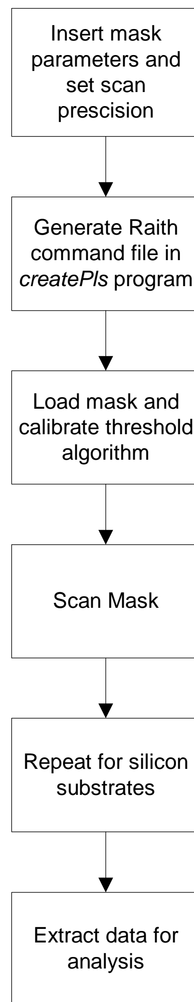


Figure 4.1 The data capture process.

4.1.1 The Measuring Process, Scan Parameters and Command File Generation

The Raith EBL system executes its operation from a file containing a list of commands called a position list. In order to instruct the Raith to scan all the markers on a mask/substrate, the known parameters of the layout of the mask are entered into the *CreatePls* program (Appendix A) along with parameters that govern the resolution and dimensions of the scans. This is illustrated in Figure 4.2.

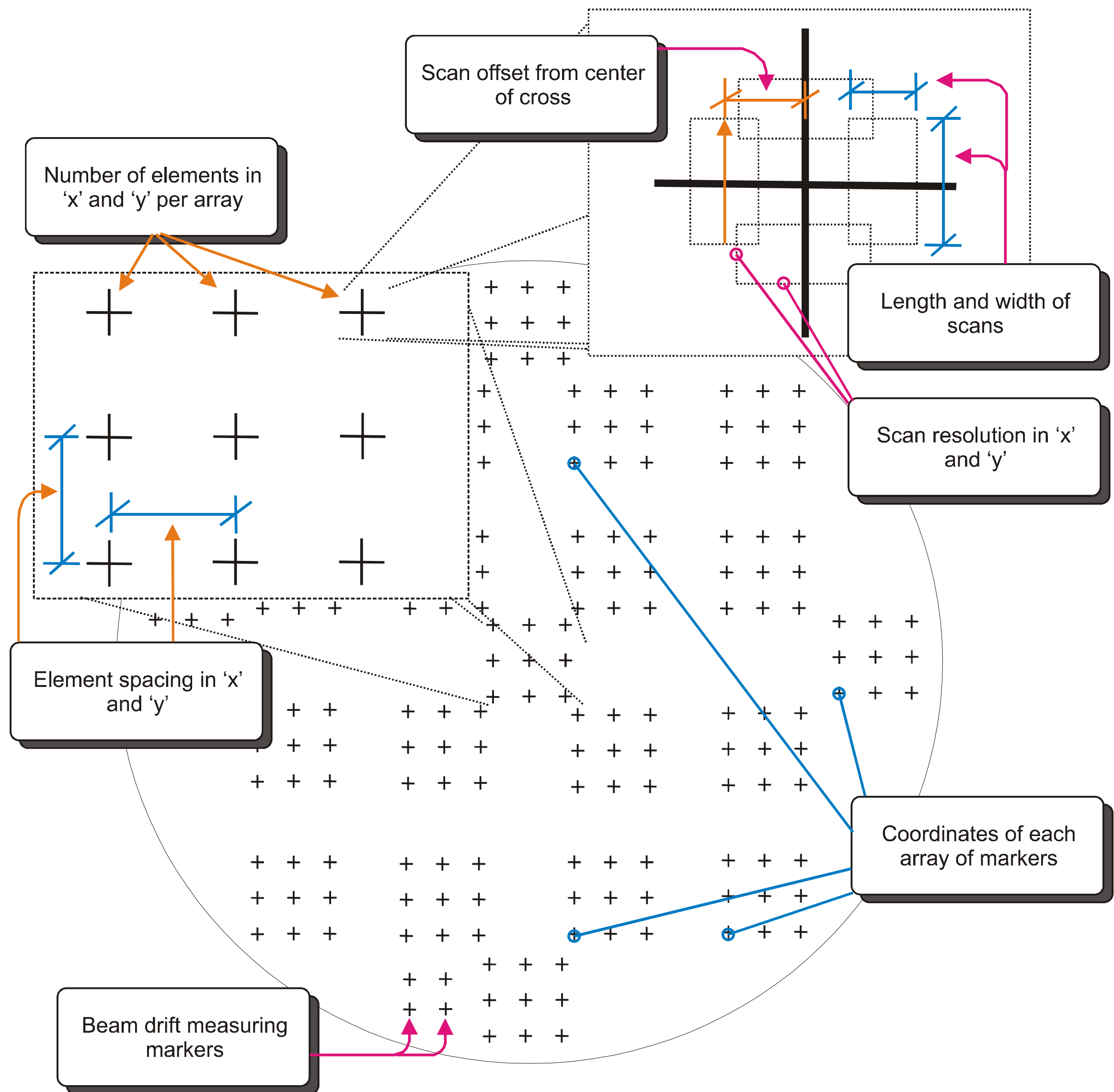


Figure 4.2 Parameters used in the *CreatePls* program to generate Raith command files 'position lists'. Parameters not shown are the frequency of the beam drift measurements and a number for pre scan beam drift measurements.

Additionally two more parameters are passed to the *CreatePls* program: an initial number of beam drift measurements to provide a means to only measure beam drift in a session, and the frequency of beam drift measurements when scanning the whole mask/substrate. The latter parameter made it necessary to develop a program to generate the Raith command files. This is because the Raith software is only capable using a 'matrix copy' to generate repeated units of scan commands. It is not able to copy a command a given number of times to the same position, nor is it able to insert commands to be at a set frequency relative to the scans of the general cross markers.

In fact, for each marker a set of five command lines are needed which are treated separately in the Raith software. Beam drift measurements could be inserted manually however the quickest way was to generate the distortion measurement marker (general marker) scans is by: using a matrix copy then dragging a beam drift measurement from the bottom of the position list up through several hundred lines to the relevant position, as new commands always appear at the bottom of the list, finally repeating this a hundred or more times.

What was needed was a program to treat as separate entities the scans of general markers and beam drift markers, and a way to alter the frequency of the beam drift measurement. The *CreatePls* program enabled this, and made the placement of arrays flexible, and the scan length, width and resolution easily changeable. The array placement was no longer restricted to that of a square array and therefore enabled better coverage with measurement markers of the 18mm circular sample area. The ability to alter the scan length ensures marker arm stays within the scan with the increasing effect of beam drift, and altering the width changes the length of the marker arm feature edges measured – mitigating line edge roughness by averaging (see Figure 3.7). The final parameter, that of scan offset, allows altering the distance of the scan from the expected centre of the cross. This is useful as discussed in the previous chapter it is necessary to scan the developed PMMA rather than the etched tungsten.

In order to measure a cross marker arm, the microscope exposes it to an electron beam as it scans. As discussed in chapter three, this breaks down the polymer chains not only making them more readily dissolved away in for development, it also creates more spaces for electrons to pass through the resist and reach the underlying tungsten and be detected via Rutherford back scattering. This results in a higher intensity in these areas reducing contrast in subsequent scans. Figure 4.3 shows this for an early beam drift measurement marker. Being able to alter the scan offset means that for repeated scans of a single mask, 'fresh' areas on the cross marker arms can be scanned rather than continued degrading of scan quality per subsequent scan.

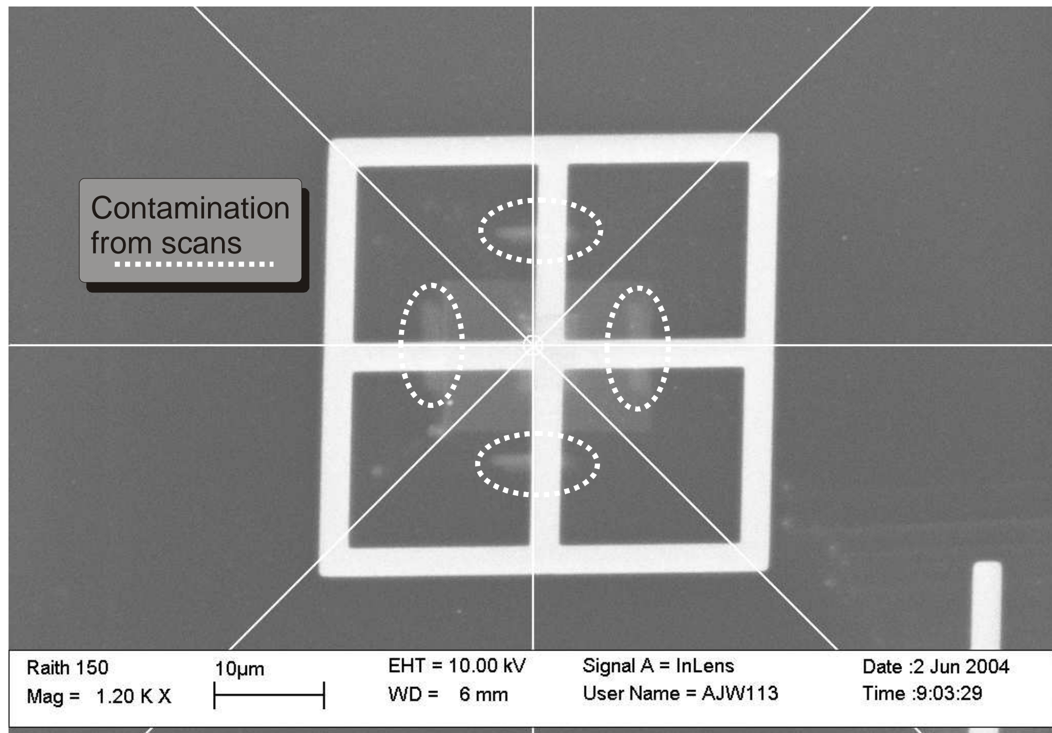


Figure 4.3 Early beam drift measurement marker. Contamination from scans can be seen as lines across the marker arms.

Once the parameters of the mask are entered, along with the resolution, dimension and offset of the scans plus the frequency of beam drift measurements, the Raith position list file is generated and is ready to be uploaded into the Raith system. A section of a typical Raith position list file is provided as an example in Appendix D.

With the command file uploaded into the Raith, and a patterned sample in the chamber the electron beam column is excited with a 10kV acceleration voltage and a 30µm aperture is exchanged in. The Raith is then set up for measurement, with such steps taken as correcting the aperture alignment, stigmation, focus levelling and importantly a coordinate system is assigned to the orientation of the patterned sample. Not only must this be aligned such that the cross markers will be able to be measured in their expected locations (give or take a few microns for distortion) it also determines the amount of shift and rotation that will need to be applied to the data when it is processed. A detailed overview of the Raith EBL's setup and use is given in Chapter 3. With setup complete, the Raith EBL is able to be used for metrology by use of a software module to detect edges of features through a form of image processing.

4.1.2 The Threshold Algorithm

The threshold algorithm is the crux of the measuring process that recognises features from one- or two-dimensional images taken of the sample surface. Such images are the result of a single scan across a feature or multiple scans which are then converted to a graph of intensity over the length of the scan. It is the changes in intensity – brought about by the presence of features – in this line scan that the threshold algorithm will return data from. An example of this is depicted in Figure 4.4 with a cross marker feature overlaid with intensity line scans.

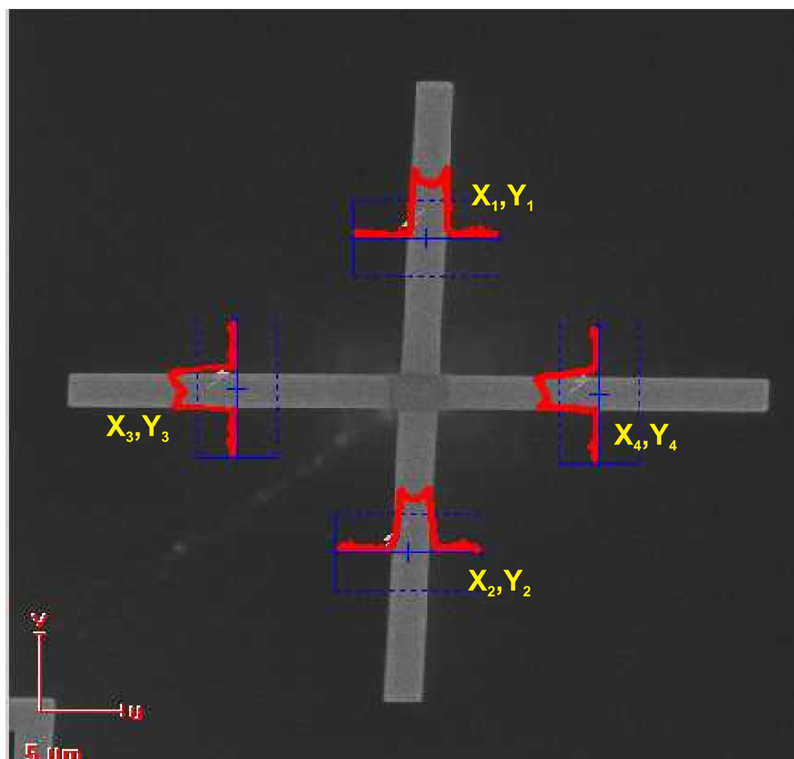


Figure 4.4 A marker cross with executed scans overlaid. Coordinate numbers represent order of scan execution.

For data capture, the primary parameters from the position list are the ‘x’ and ‘y’ coordinates found from each scan of the four marker cross arms. For a horizontal cross arm measurement (e.g. X_3,Y_3) the scan is vertical and the ‘x’ coordinate (along the scans axis) is a given from the known location of where the scan takes place, the ‘y’ coordinate however is found from the threshold algorithm (Figure 4.5).

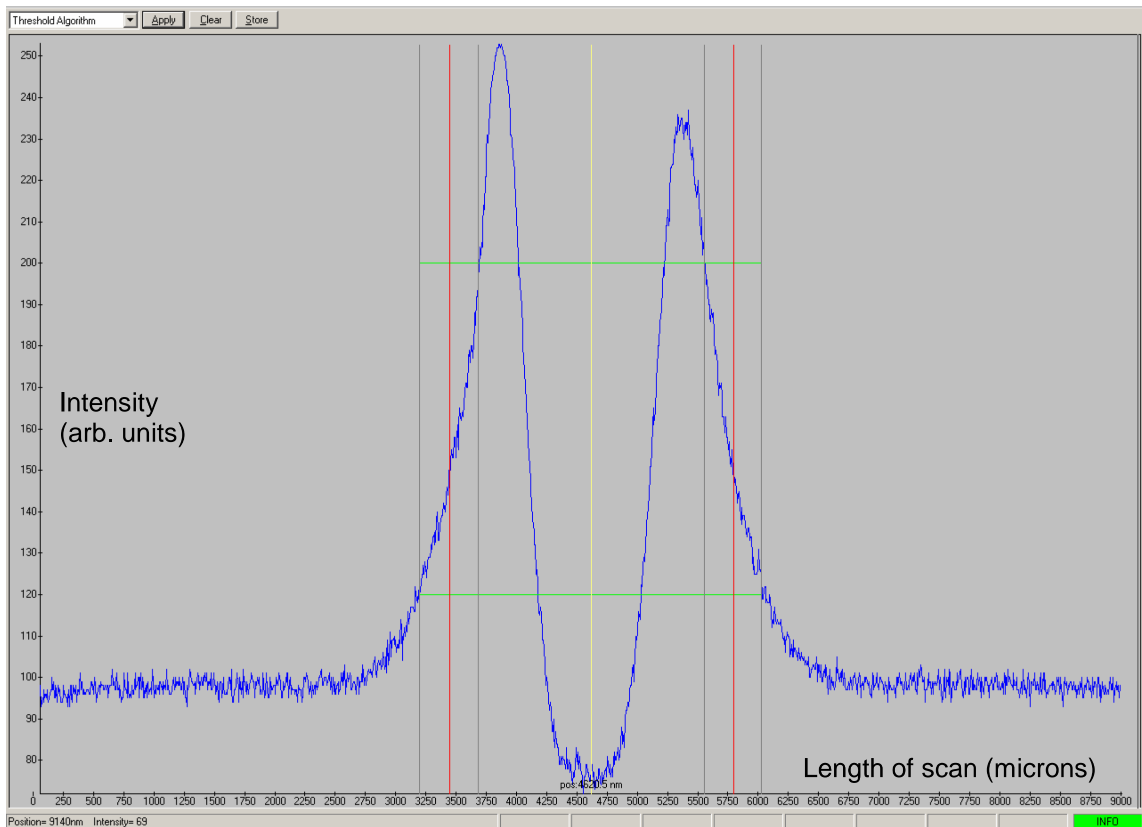


Figure 4.5 Intensity line scan of a cross marker arm showing threshold algorithm position markers. Green lines indicate the threshold interval in which line edges are weighted. The two red lines are where the algorithm has determined the edges of a marker arm and are stored as position 1 and 2. The yellow line is the calculated mid point of a marker arm stored as position 3.

The threshold algorithm searches for an intensity change between two values making a threshold a preset number of times in from the left or right of an intensity line scan graph. When it finds an intensity change within this threshold it records the two points along the scan at which this occurred. The threshold is shown by the green lines in Figure 4.5 and the points at which it is crossed are shown by grey lines. The midpoint between these two grey lines is calculated and it is this position which is determined as one of the edges of a cross marker feature. Once these edges are found from either side of the intensity graph the midpoint of the cross arm feature is calculated by simple averaging, which gives us the 'y' coordinate for a horizontal cross arm feature and vice versa for a vertical cross arm. The algorithm finally stores three values, the positions of the edges and the calculated midpoint [47].

Initially it was not thought that calibrating the threshold algorithm for each scan was necessary. This was because the algorithm could be set to use a threshold set by intensity percentages relative to the intensity graph found. These settings (which can be seen in the right hand side of Figure 4.6 in the threshold box) will always return results, as for a scan; the software determines the window of intensity between the highest and lowest values so there will virtually always be lines filling the window.

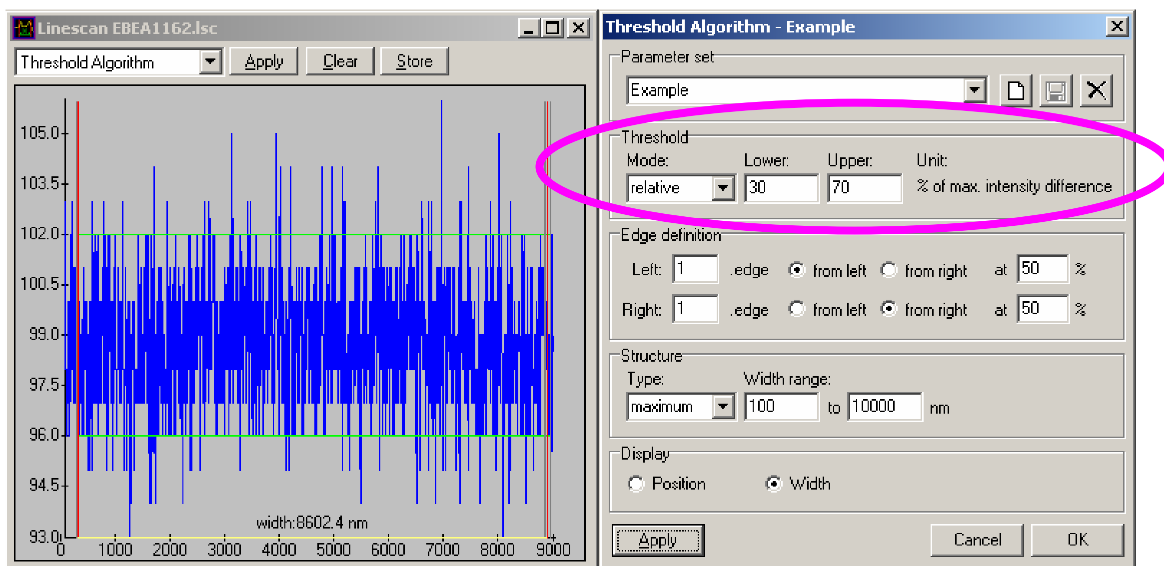


Figure 4.6 Operation of threshold algorithm using relative threshold values. The scan above is of a relatively uniform intensity because the cross marker arm in this case was missing. The threshold algorithm finds intensity changes within a threshold set to relative intensities of an intensity graph and these are recorded as line edges giving erroneous data.

This means for a relatively uniform intensity, such as the graph shown in Figure 4.6 in which the cross marker arm was missing, line edges and position are found by the algorithm and returned. Its operation is correct according to its settings but it returns a meaningless result in this case where the aim is to find successfully patterned line features. Using this result would introduce error on the order of half the scan length. This is because the intensity graph of a relatively uniformly bright area approximates noise and an intensity change within threshold will be approximately the same distance from either side, therefore the centre of the scan will always be returned. With a typical scan being of length 8μm this would introduce an error of up to 4μm per missing marker arm.

The solution is to use absolute values of intensity for the threshold values. The result of which is shown in Figure 4.7 where, because the intensity does not vary between the absolute values of 120 and 200, no line edges are found and the scan is returned as error, as desired.

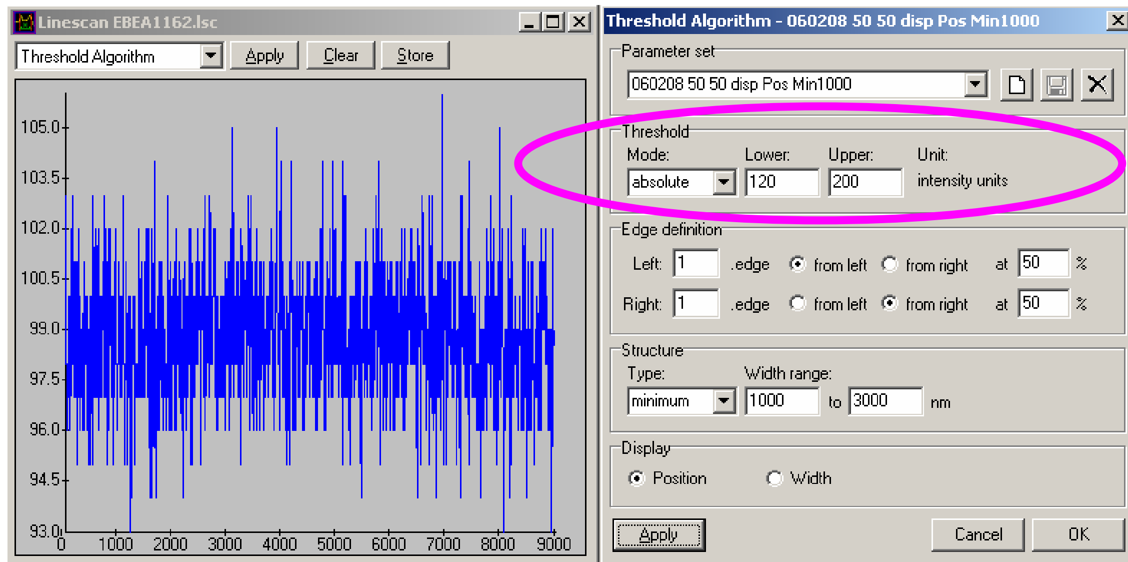


Figure 4.7 Absolute threshold values used to avoid false positive identification of line edges in a uniform intensity line scan (due to a missing cross arm).

To find the threshold in absolute values, a number of test scans are taken at the start of each major scan to observe the normal window of intensity. Once contrast and brightness have been set on the Raith system to provide a good exposure (typically 50% brightness and 40% contrast), intensity values between 120 and 200 units are used. Another issue is that where some cross marker arms are dimmer on one feature edge with respect to the other. This would presumably be because of thicker PMMA in these areas causing a darker surface. When the surface is darker on one side, it falls short of the intensity threshold and the algorithm moves on, not determining the feature edge, to find an intensity change that does fall within the threshold. This can result in the false positive of Figure 4.8 in which the line edges determined do not match the actual cross arm feature. To avoid this issue the minimum width setting can be used to reject such an instance as the algorithm misses the first edge the resultant width will be less than 1μm the nominal width of the cross markers.

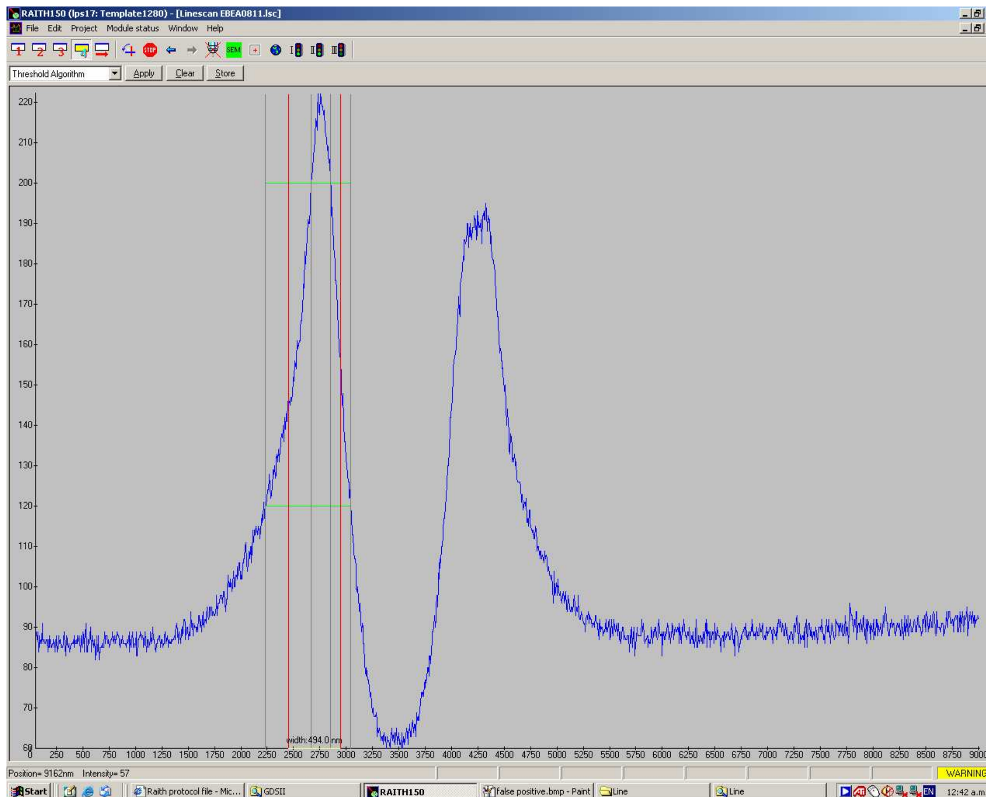


Figure 4.8 False positive identification of feature edges in a line scan. In this case, one side of a line feature has a lower intensity than that of average, it then falls beneath the absolute threshold values, the algorithm searches further along from right to left finding an intensity change within threshold. The result is that the midpoint is determined at the edge of the line. This can be avoided using the minimum width parameter.

However, not all marker arms are scanned successfully. Some markers are within thicker PMMA and result in an overall lower intensity such as shown in Figure 4.9. These fall outside of the threshold values that will work for the rest of the markers and therefore are not scanned successfully. Such results represent false negatives. If missing marker arms were not an issue threshold values based on relative intensity would return data for these ‘dark’ marker arms. However such arms are few in number (ranging from 0-1 per mask, typically 2 per sample) and for the overall analysis in this case the loss in accuracy from a few false negatives far outweighs the inaccuracy which would result from using relative threshold values which, although including such dark marker arms, would add an error of half the scan length (as mentioned earlier).

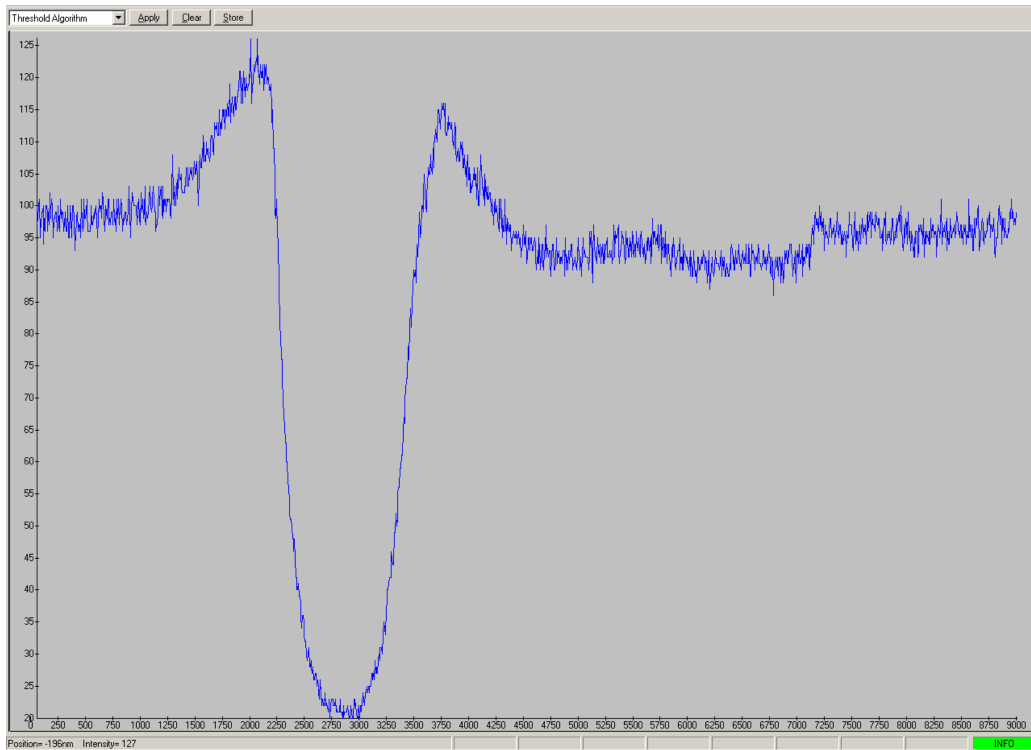


Figure 4.9 Thicker PMMA has resulted in an intensity line scan graph that is below the normal threshold window defined by absolute values yielding scan error. As a feature is clearly documented and this result is a false negative.

Once a successful scan is completed the position of the sides of the detected line and the line's centre position are recorded in the position list file. With all four cross arms scanned, yielding the midpoint coordinates of each, the centre of the measurement marker is found through calculation. The accuracy of this has been found to be up to 63nm, however this is provided no dust particles were present during the initial scan of the mask membrane. The results and discussion of such are presented in later chapters. Now that the raw data has been obtained it must then be processed in order to compensate for various error-producing phenomena: from mundane misalignment of the patterned samples to missing markers and the unpredictable spiralling drift of the electron beam.

4.2 Data Processing

In order to achieve the aim of this measuring process, that of revealing pattern distortion, the data must have the various systematic errors compensated for so that one data set may be compared to the other. The procedure for analysing the data is shown below in Figure 4.10.

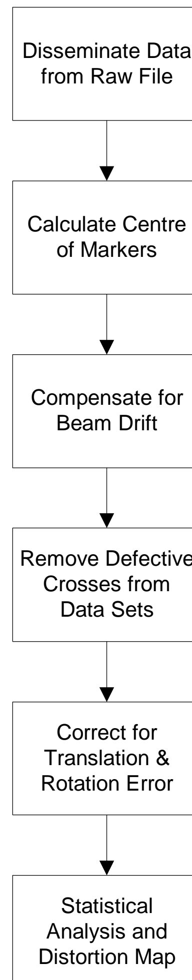


Figure 4.10 The data analysis process.

The position list is read and processed by two programs used consecutively. The first is *ProcessPlsJava* (Appendix A) which accounts for the first four steps of the data analysis process. It calculates the centres of the crosses, corrects measured positions for the drift of the electron beam and compensates for missing markers on a substrate or on the mask. The remaining steps are performed by the program *ProcessPlsMatlab* written in Matlab (Appendix B). This corrects for misalignment of the coordinate system during measurement and generates statistics and a map of distortion.

4.2.1 Raw Data Processing and Marker Feature Centres

The first step of the data analysis process is to gather the relevant information from the raw position list file. This file can contain many parameters of the EBL system including the working distance of the microscope, the filename and path containing the line scan for a measurement, to dose factors and step sizes for writing patterns. Each line of the file represents a separate command, containing space for all the parameters each separated by a comma. The necessary parameters are then obtained by the *ProcessPlsJava* program (see Appendix A). Each scan has the following additional parameters stored: the time stamp of the measurement scan, whether there was an error, the type of scan, and a preset text field labelling the scan. This text field is set to 'beam drift measurement' or 'general measurement' using the C*reatePls* program (see Appendix A). The recorded 'x' and 'y' coordinates are then used as pairs for the horizontal arms and the vertical arms of a marker. For these pairs, the line equations are calculated and the intersection of which yields the centre of the measurement marker (Figure 4.11).

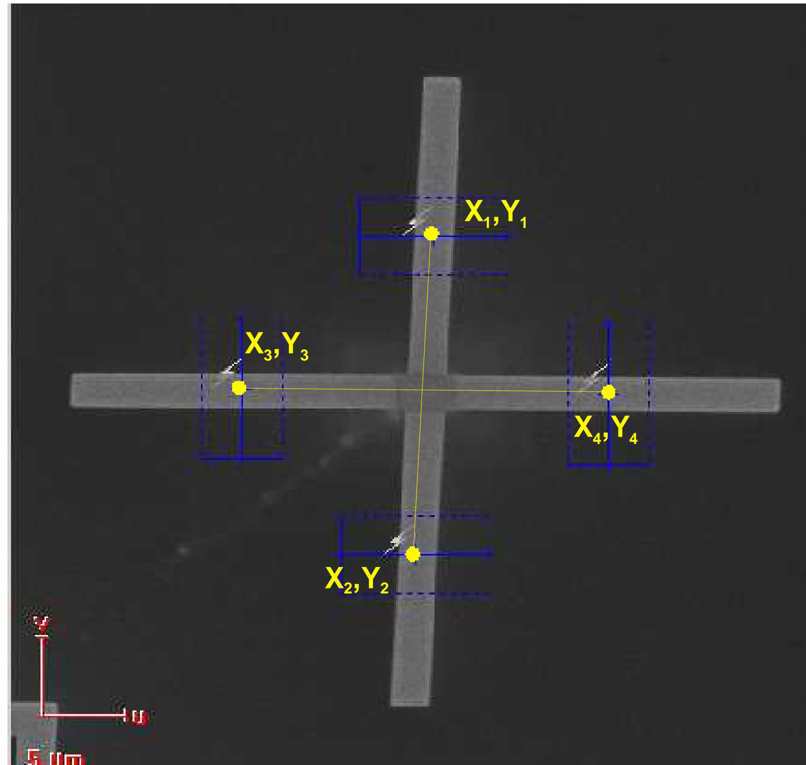


Figure 4.11 Cross with superimposed measured coordinates. Lines represent line equations found from the scanned coordinates. The cross centre is found from the line intersection.

If any scans are missing – due to a cross arm being malformed, or a marker missing altogether perhaps due to dust, the marker is flagged as missing so that a missing marker in the sample is not compared against one present in the mask or vice versa – by which a marker missing from a scan the mask would be able to be patterned to the substrate. This has occurred and is likely to be the result of a viable marker being missing from the mask scan due to a particle present on its surface preventing its successful scan but later shifting so that the marker was patterned to the substrate. The *ProcessPls* program is able to detect a missing marker by checking the error flag of each scan which is set by the threshold algorithm when no features are found. Once the drift of the electron beam is compensated for the *ProcessPlsJava* program deletes the erroneous missing cross data from sample and its corresponding marker in the mask data and vice versa.

4.2.2 Compensating for Electron Beam Drift

Due to uneven and unpredictable heating in the electron acceleration column and imperfect electron 'optics' the beam emitted from an electron microscope drifts; pointing at a varying location in the 'x', 'y' plane presented to it. For example the stage remains stationary but the beam may draw a spiral. The Raith 150 electron beam lithography system has been ascertained to have a drift of not more than 1μm per hour [41]. Measurement of the Raith 150 is this department yields a beam drift of 0.86μm per hour from a total drift of 1.96μm over 2 hours and 16 minutes (Figure 4.12). This is less than that reported in [41] and could be found to be lower still by allowing 3 hours for the sample and stage to be in thermal equilibrium with that of the chamber as in [41].

In order to mitigate the effects of beam drift the same marker can be repeatedly scanned in between measurements and thus the apparent differences in measurement can be taken as drift. Initially beam drift was compensated using the beam drift measurements with no consideration of time – this meant that for a given number of cross measurements e.g. 10 taken in the fartherest corner of the sample were compensated in the same fashion as measurements taken directly beside the beam drift measurement marker.

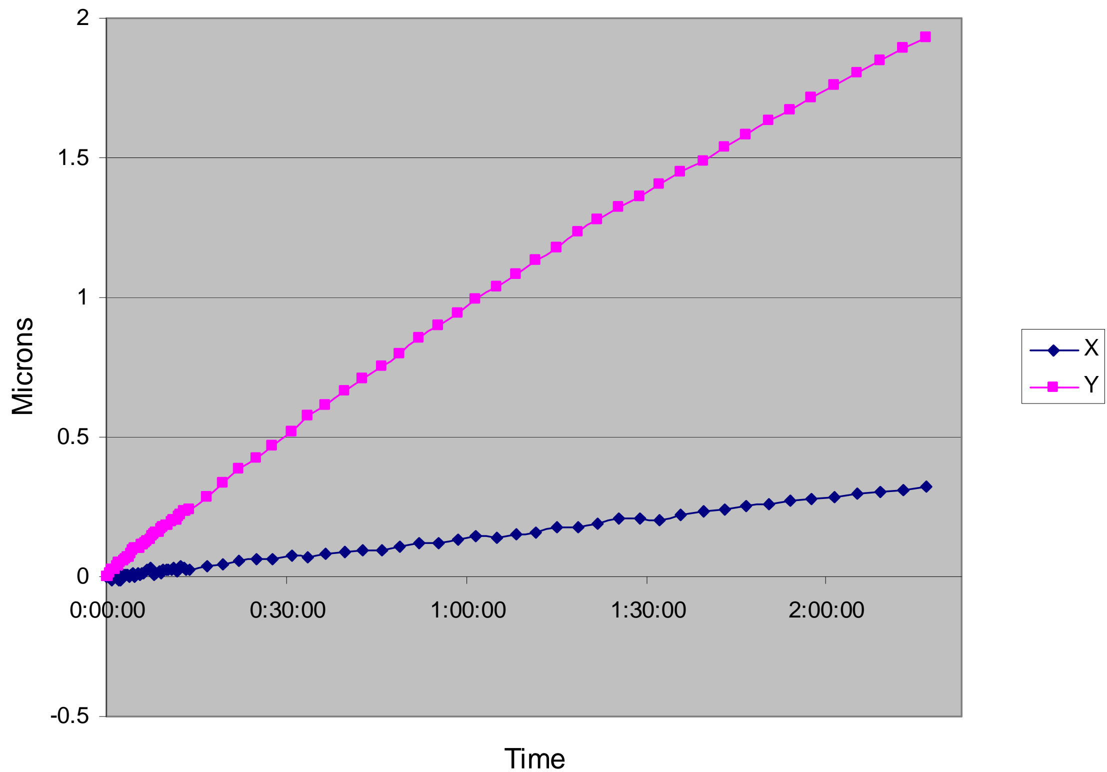


Figure 4.12 Electron beam drift over time in 'x' and 'y' for the Raith 150 EBL.

The result of this is that the fartherest corner measurements would be over compensated as with the time taken to travel back from corner the measurements the beam would drift further and this extra drift was subtracted from the measurements skewing the results. With the use of the scan timestamps the times at which general measurements were taken and beam drift measurements taken meant that the beam drift could be linearly interpolated with respect to time between two sets of beam drift measurements increasing the accuracy of the compensation.

From these measurements the drift of the electron beam can be plotted with respect to time. This is shown Figure 4.13 with the drift of the beam in 'x' and 'y' over a period of 26 minutes. The curve plotted is not smooth and it appears to be made of line segments; this is due to the use of linearly interpolated values. For a more detailed view Figure 4.14 is taken from the data set of Figure 4.13 to serve as a magnification of. Here it is apparent that the linear interpolation of the beam drift is an

approximation of the actual drift. The beam drift measurements are labelled with arrows, and the other points are the interpolated values. Accuracy of beam drift compensation could be increased by taking more frequent beam drift measurements, by use of non linear interpolation or by driving the microscope to each scan coordinate to reduce intra-field distortion[41].

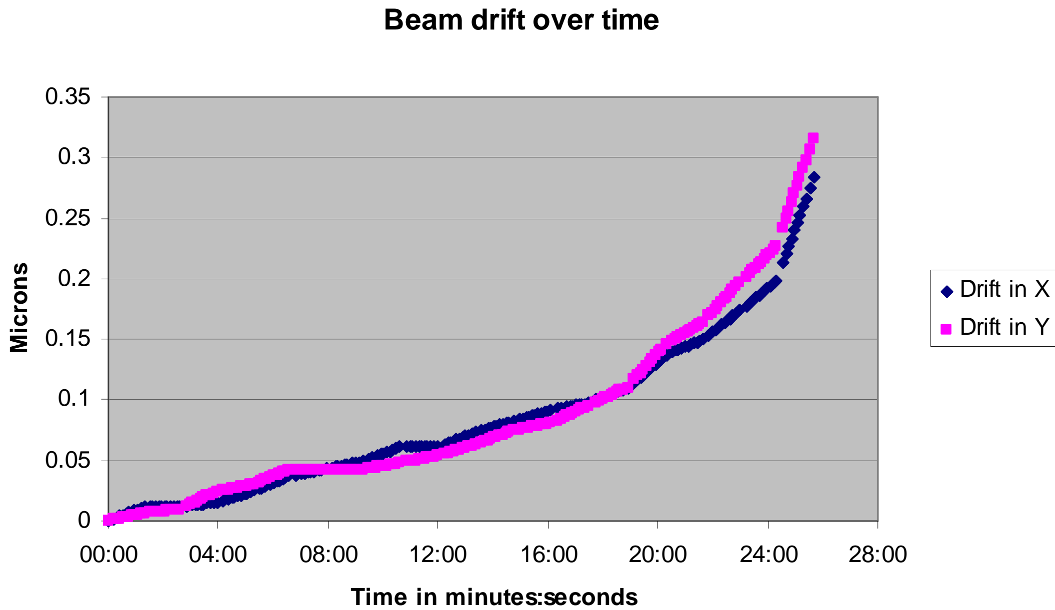


Figure 4.13 Beam drift in 'x' and 'y' over the course of scanning a marker pattern array on a sample.

From Figure 4.14 with a curve fitted through the beam drift measurement points as the likely path of the drifting beam in 'x' it is estimated that measurement uncertainty due to such remnant beam drift error is less than 20nm, so such additional steps were not implemented.

4.2.3 Removal of Defective Markers

For a given experiment it was typical that not all markers were successfully scanned from the silicon substrate, and occasionally one marker was found to be defective from the mask. These defective markers created a mismatch between that of the mask and substrate marker sets, hampering distortion measurement that requires the successful measurement between a marker on the mask and where it was patterned on the substrate.

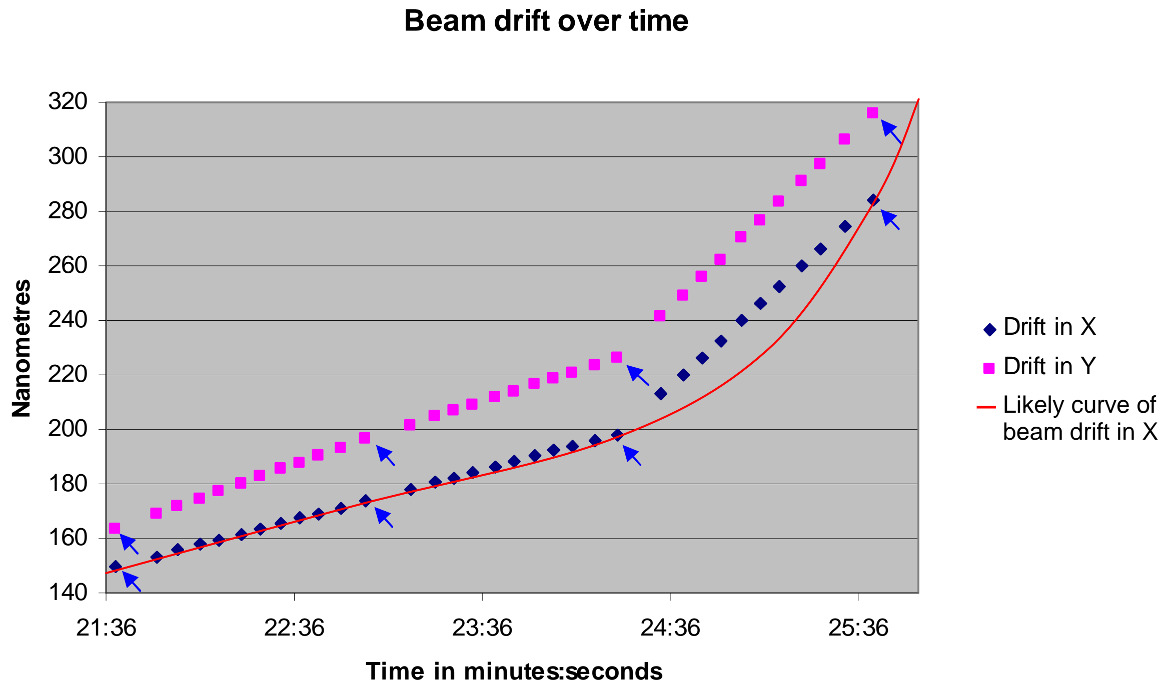


Figure 4.14 Magnification of Figure 4.13 to show interpolation. Arrows indicate points where actual beam drift measurements were taken, with interpolated beam drift points in between. The time delay to drive between beam drift measurements and cross measurements can be seen. The likely curve of beam drift in 'x' is overlaid in red to show the interpolation error.

Individual markers failed to scan for a number of reasons: dust on the mask/substrate during photolithography, obscuring mask features; particles present in the resist during spin-on to the substrate, leaving unexposable points; imperfections of the mask during its patterning through particles in the electron resist from spin-on or from dust on the mask during patterning; false negatives given by the threshold algorithm as discussed in section 4.1.2 and finally, dust obscuring the scanning of markers on the mask or substrate preventing measurement. Nearly all the missing markers were missing from the substrate, mainly from failed pattern transfer and for the mask, occasionally one missing marker. For a complete set of markers as that shown in Figure 4.15 (189 markers) typically 10-15 markers were found to be defective for a scan of a patterned silicon substrate.

In order to correct the mismatch between the data sets, a defective marker in one data set must be removed along with its corresponding marker from the other data set. A routine was written in the *ProcessPlsJava* program to this end leaving matched data sets with successful measurements, which could then be compared to one another.

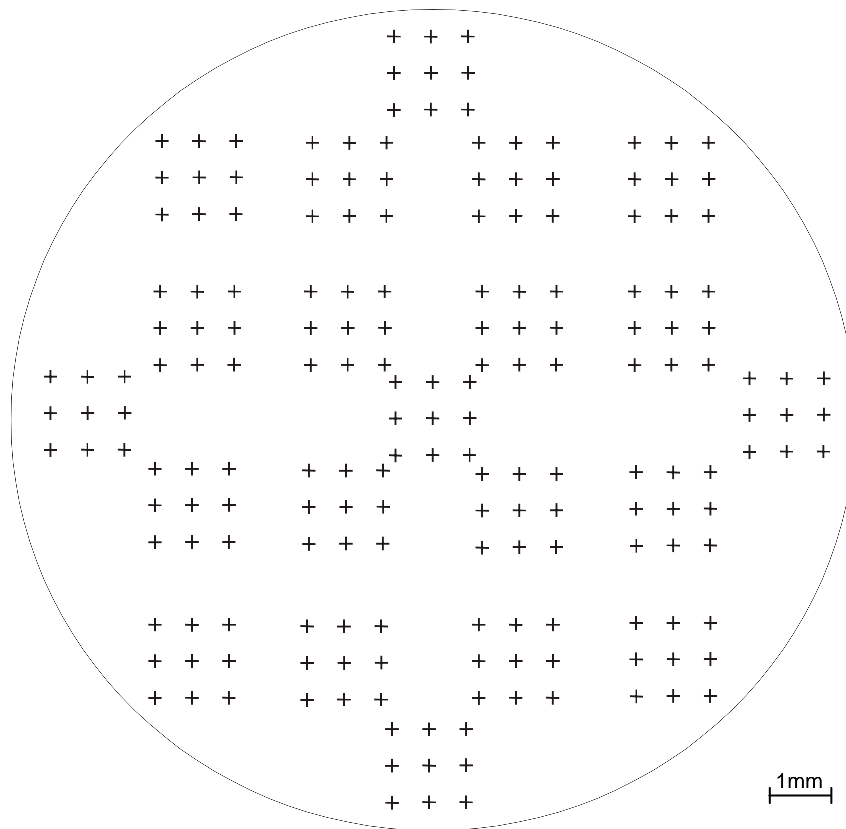


Figure 4.15 The marker array pattern used for measurement experiments. It is symmetrical about its coordinate axes.

4.2.4 Correcting for Scan Shift and Rotation Error

Upon compensating for beam drift, the data sets must be corrected for scan shift and rotation error, which is the misalignment of the measuring scan's coordinate system to that of the marker patterns on the samples. This occurs as it is not possible to obtain perfect alignment for a number of reasons which follow. When a scan is initiated, the 'x' axis (and hence the 'y' axis orthogonally) is aligned using markers that are maximally apart along the bottom of the pattern to obtain alignment with minimum error. However the error is still large and occurs in the determination of these points (or any other marker) due to a variety of sources: the alignment of the electron write field to that of the precision stage, the quality of the patterning, drift in the beam both when the marker was patterned and while it is being scanned and finally the pattern distortion that is the investigation of this thesis if it is one of the patterned silicon substrates to be scanned.

With all these sources, error is inevitable and can be separated into two forms: first, that of shift, where the origin is translated from where it is on the marker pattern thus lending a false translation to the marker pattern and second, misalignment of the coordinate axes resulting in a perceived rotation of the data points. These types of systematic error are illustrated in Figure 4.16 for a scan of a data set to the actual position of its marker features.

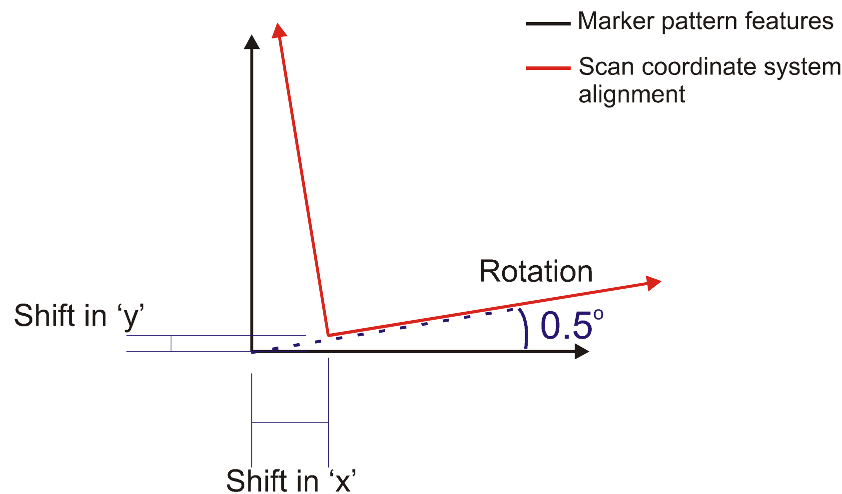


Figure 4.16 Scan shift and rotation error; due to imperfect alignment of scanning coordinate system (red) to that of the marker pattern features (black).

In order to correct for these errors when the data on any one marker centre cannot be relied upon, the measurements of all the markers are analysed simultaneously. The crux of this process is finding the centroid of each data set, the centre of mass of a two dimensional plane. Correction is then achieved by mapping the data points onto each other by first shifting each data set so that the centroid and origin coincide and then rotating one data set onto the other to find the minimum error to a very high precision.

In order to find the centroid, two lines of least squares are found through the data sets one with respect to the 'x' axis and the other with respect to the 'y' axis; the intersection of these gives the centroid. The centroid is able to be found this way as a line of least squares provides the line that bisects the data such that the points are evenly weighted away from it – a line with even 'mass' on either side of it in a two dimensional plane. If another line of least squares could be found, the intersection of this one and the first would yield the centroid. Thus finding two lines of least squares one with respect to the 'x' axis the other with respect to the 'y' provides the means to

this end. Shift correction is now achieved by shifting each data set so that the centroid coincides with the origin. This is demonstrated in Figure 4.17 using a test data set of points that were shifted and rotated to create a second set of points giving simulated mask and patterned silicon data points respectively. For clarity the shift and rotation were made large with a translation of 3μm and rotation of 15 degrees.

Upon correcting for shift (Figure 4.17(g)) one data set must be rotated to map onto the other. The patterned silicon data set is chosen to map onto the mask data set and the mapping angle calculated using an incremental numerical algorithm. This algorithm calculates the mapping angle by finding the minimum displacement between the two data sets by successively searching over a narrower angle in successively finer increments. The algorithm receives an initial angle to search over, i.e. 20 degrees, it then calculates the displacement between the two data sets in increments of 10^{-N}, where N is from 1 through to a preset precision i.e. 20 incrementing over successive loops. The displacement is calculated by: first rotating the silicon data set, and then subtracting the mask data set points from the sample data set points yielding a data set with the displacement components in 'x' and 'y' for each of the silicon data points. This is then converted to polar coordinates to create a data set with the absolute displacement of the data. The sum of the absolute displacement is then stored in an array alongside the angle of rotation currently being tested. Once the range of angles has been tested for, the angle that yielded the minimum displacement is used as the basis of the next range of angles to search over where the range is the angle $\pm \frac{1}{2}10^{-M}$ where M is equal to N for this loop. The next loop searches over this range but in increments of 10^{-N} where N is now 2. The next successive loops search over ranges of 10^{-N-1} in increments of 10^{-N} up to N=20. Once the algorithm has completed calculation the compensation angle is applied to the data and the data will have been corrected for rotation to a high degree of precision (Figure 4.17(h)).

Note that with good computing power, calculating to high precision – incrementing angles from 10^{-1} to 10^{-20} over 20 loops with each loop executing 20 sets of calculations, some 400 sets of calculations – the computing time is a fraction of a second, so seemingly gratuitous precision is a matter of 'why not?'

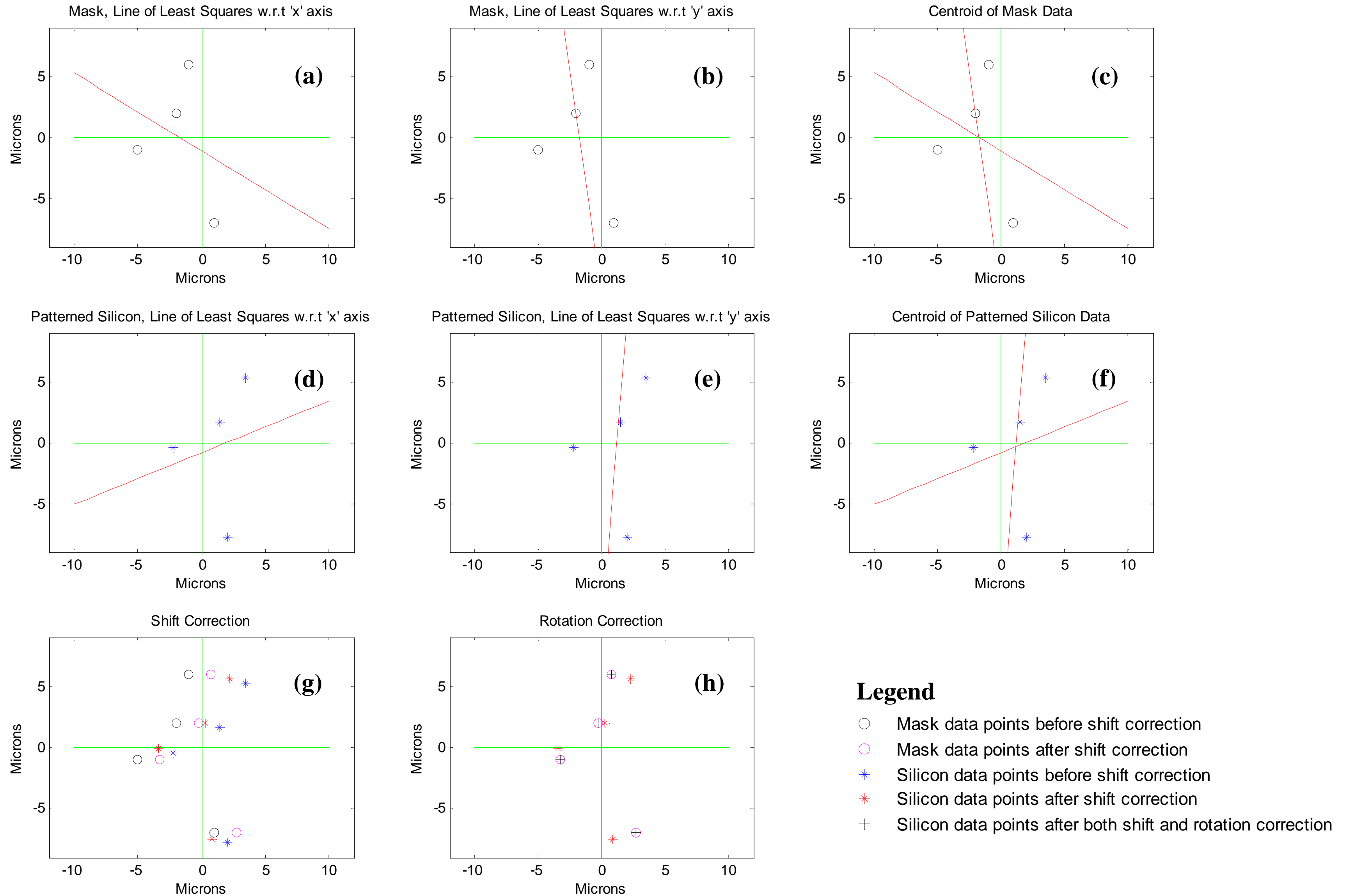


Figure 4.17 Centroid determination, shift and rotation correction demonstrated using a test data set. Figures 'a-c' and 'd-f' show the determination of the centroid for the mask and patterned silicon data sets respectively. Lines of least squares are shown in red. Figure 'g' depicts the translation in the data points as they are shifted such that the centroid coincides with the origin achieving shift correction. Figure 'h' shows the patterned silicon data points mapped exactly onto their mask counter parts upon achieving full correction.

The performance of shift and rotation correction was tested by duplicating a data set and adding Gaussian noise to one copy. This gives two data sets that have not been shifted or rotated with respect to one another. The shift and rotation correction algorithms should ideally not apply any shift or rotation compensation. Any shift and rotation that is applied should show the error of the algorithms. Measured distortion is on the order of 1µm (as shown in the next chapter) to three standard deviations (or three sigma for short); thus a sigma of 0.33µm was chosen for the noise. The result of 1000 trials for a data set of 176 markers gives the error of the applied shift correction to be a sigma of 25nm in 'x' and 'y' and a sigma of 2.7×10^{-4} degrees for the rotation correction error. Given that the added noise or added error had a sigma of 330nm, a resultant sigma of 25nm for the shift error represents at least a factor of 10 recovery from the introduced error. This is why so many measuring points were chosen across the surface of the samples and adding more points would further improve the performance of the algorithms.

4.2.5 Results: Statistical Information and Maps of Distortion

Once the data has been compensated for beam drift, rotation and shift error it is easily made ready for comparison of mask to substrate marker displacement and the gathering of statistical information. The substrate marker positions are subtracted from the mask marker positions to reveal the displacement from what was originally written to the mask to that patterned. The displacement in 'x' and 'y' are then stored in separate arrays and used with the Matlab commands mean() and std() to obtain the average and standard deviation of the data. This can then be displayed as a histogram to show the displacement distribution such as that shown in Figure 4.18.

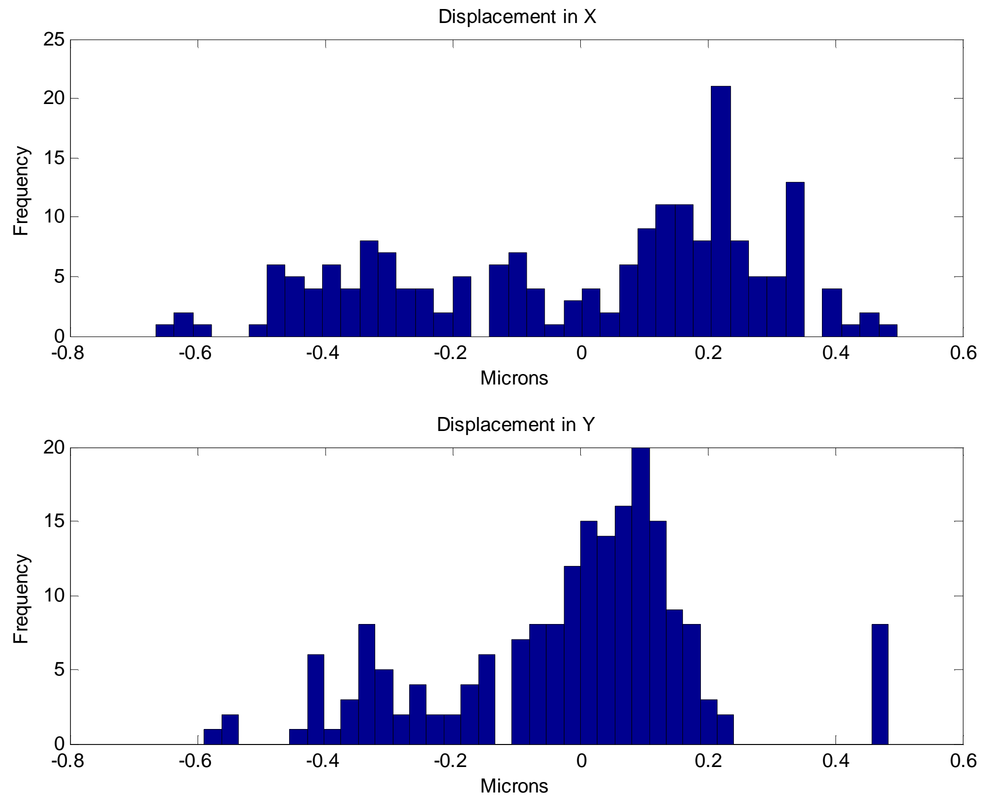


Figure 4.18 Displacement histograms.

Next, the difference in position of the sample markers with respect to the mask markers is plotted as a vector map using Matlab's quiver() function. This function has the mask marker positions and the displacement data passed to it; the mask marker positions are used as the start points of the arrows, and the displacement data for the direction and magnitude of the arrow e.g.

```
quiver(maskmarkerX,maskmarkerY,displacementX,displacementY)
```

this is shown in Figure 4.19 where the arrows indicate the displacement of the markers in the substrate relative to that of the mask, creating a map of the distortion. With scaling of the arrows the displacement is greatly amplified – approximately 500 times, which effectively illustrates the distortion.

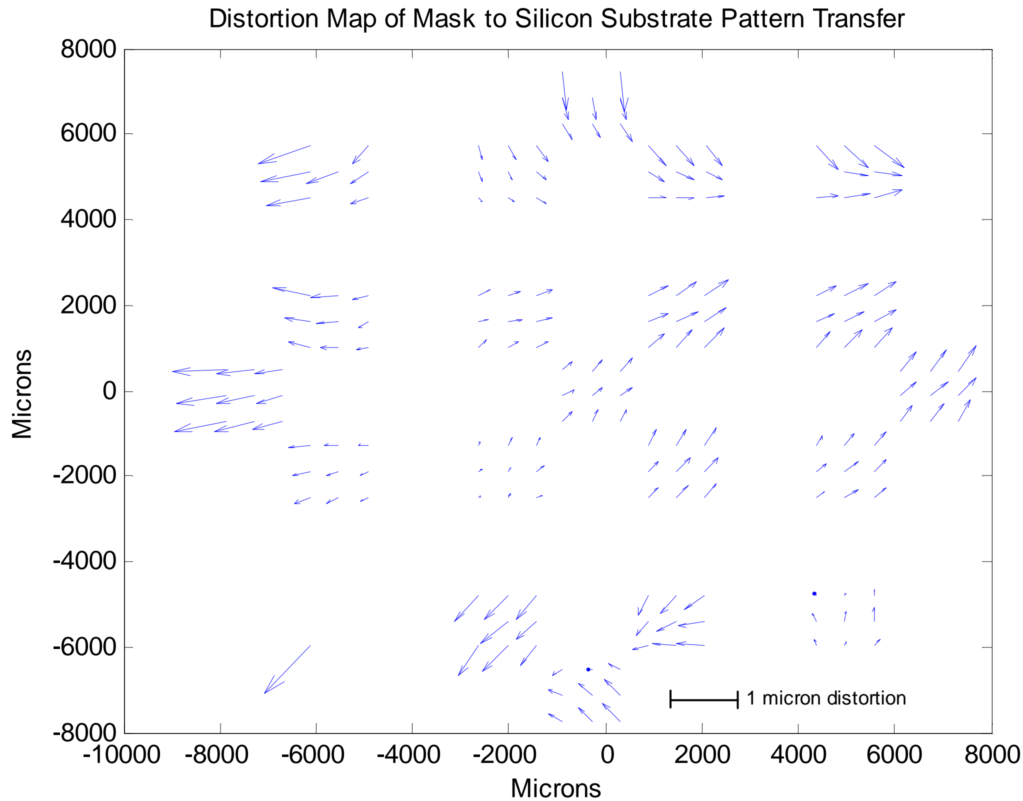


Figure 4.19 Distortion map showing the displacement between markers in the silicon substrate and that of the mask. The arrows indicate the 'movement' of the substrate markers.

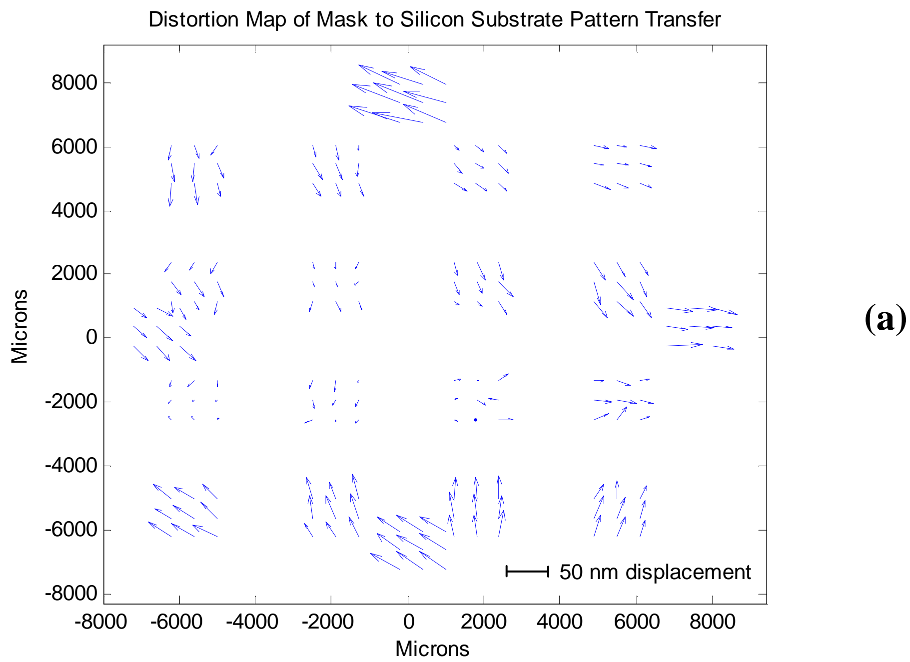
Finally with the distortion map produced, the nature of the distortion can be seen. In this chapter the two major stages of data capture and data processing have revealed the process by which the pattern distortion of an ENFOL conformable mask to substrate is ascertained. What is the amount of this distortion? What is the nature of the distortion thereby revealing its origin? The answers to these questions and whether the distortion can be compensated for are revealed in the following chapters.

Chapter 5: Experimental Results

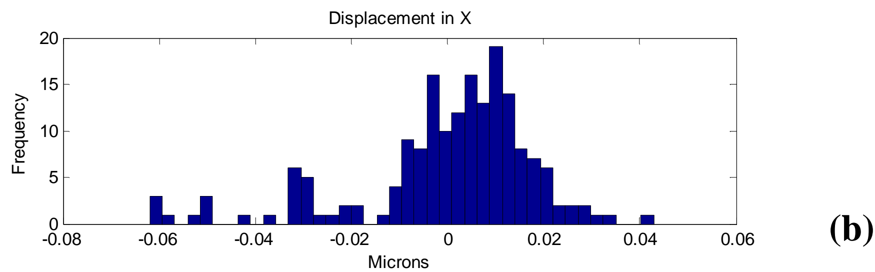
In this chapter the distortion results of mask-to-substrate patterning using a conformable contact mask are presented. The first section gives an analysis of accuracy of the measuring process itself through successive scans of the same sample, and reveals the first evidence of particle contamination affecting results. The next section presents the initial results of mask-to-substrate patterning showing a systematic distortion; this spurred the development of PDMS spacer masks, the first successful implementation of which is presented in the third section. Several issues were raised with the use of the PDMS spacer mask, some of which only became apparent with the implementation of the next design of the PDMS mask mount. In the final section, the original mask mount, which was being used for comparison to the PDMS spacer results, supersedes them through the use of 50µm separation of the mask and substrate during patterning and a higher vacuum. This results in the lowest distortion to date.

5.1 Measuring Accuracy

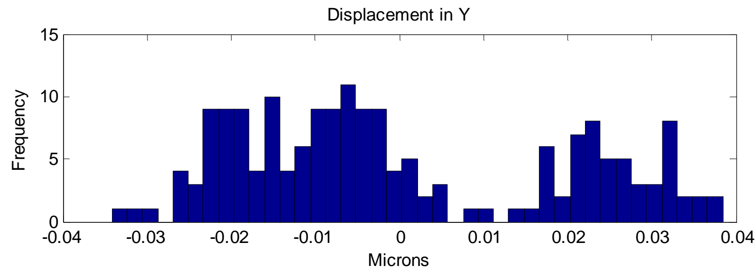
Uncertainty is inevitable in any measuring system and is the limit of its precision. In order to have a reliable measuring system, this limit of precision must be known as this gives confidence that a certain measured quantity can be taken within a known tolerance. In order to quantify the uncertainty of this particular measuring process one sample was repeatedly scanned. The apparent change in the placement of the markers as measured between these scans will yield the measuring error. The sample chosen to be scanned multiple times was an ENFOL mask, as opposed to the alternative of scanning a patterned sample. This turned out to be crucial in discovering one effect of particle contamination, which will be outlined in the subsection following. The result, depicted in Figures 5.1(a-c), shows the error in measurement found as a distortion map, distortion distribution and statistics respectively. The distortion is non-uniform and is below 60nm in any direction with the maximum errors occurring at the top and the bottom of the mask.



(a)



(b)



	'x' (nm)	'y' (nm)
Mean	0.00	0.00
Sigma	18.8	19.0
Three-Sigma	56.4	57.0

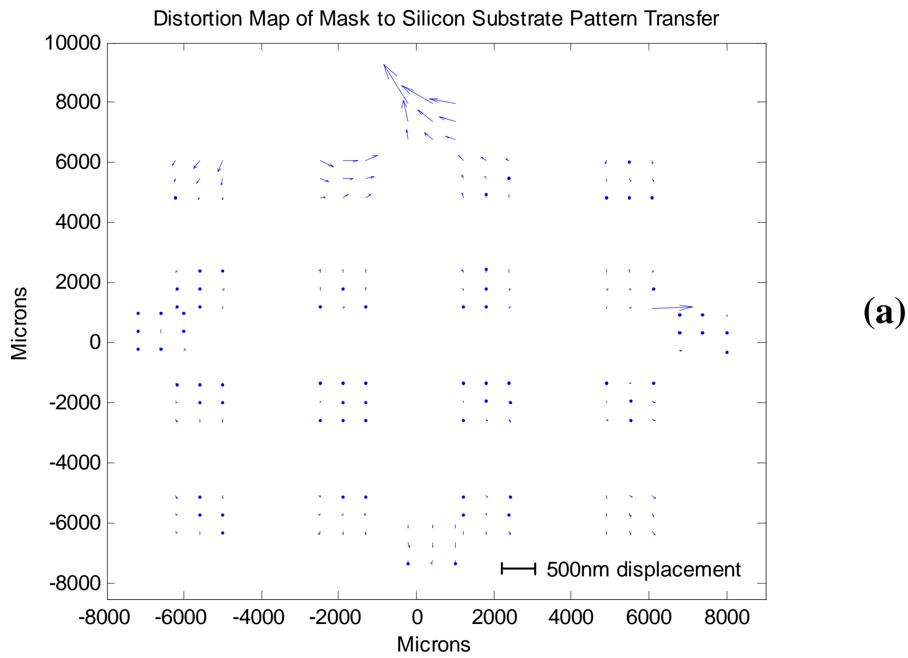
(c)

Figure 5.1 Measurement Error: (a) a distortion map showing the apparent movement of markers between two consecutive scans of the same mask. (b) Displacement distribution in 'x' and 'y' and (c) 'x' and 'y' distortion distribution statistics.

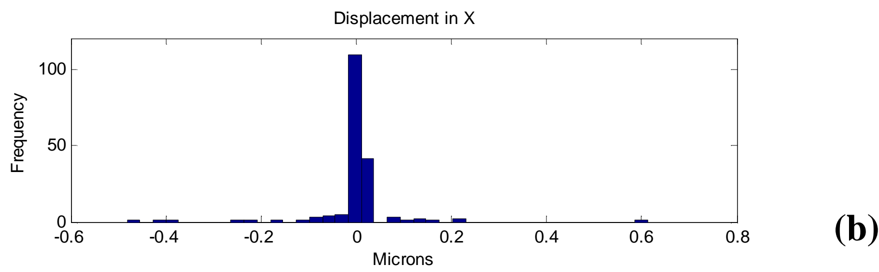
The amount of distortion is measured as the displacement in the 'x' and 'y' component directions and the distribution of the distortion presented as a histogram. The distortion is then presented as a mean plus three standard deviations (three sigma) result which encompasses 99.73% of the results as distributed in an assumed normal distribution. This is an accepted standard of quality for manufacturing and gives a high confidence level that the amount of distortion will be equal or less than the given value. The measurement error has been found to be a three sigma result of 56nm in 'x' and 57nm in 'y' absolute. This has then quantified the accuracy of measurement process and the results from gained from this process could be considered to be reliable to within +/- 60nm. However this is only the case if there was no particle contamination during the scanning of the mask, and following discussion in the next subsection the results in this chapter need to be taken with due consideration.

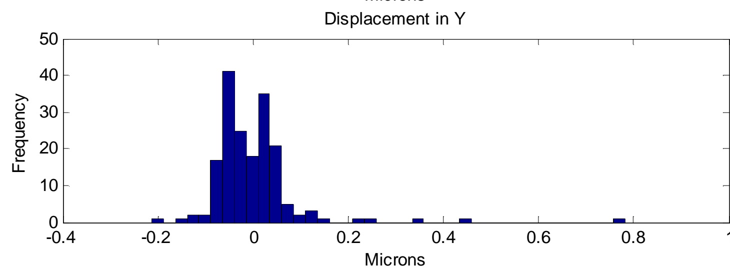
5.1.1 Effect of Particulate Contamination

Upon measuring the same mask repeatedly, it was necessary to unload and reload the mask between successive measurements over two days in order to accommodate other users of the system. In fact this was fortunate as without such, this effect may not have been considered. The result is shown in Figure 5.2(a) with a prominent localised distortion in the shape of a swirl at the top of the distortion map. During the repeated measurement of the mask, 5 scans were taken of the mask on the first day and 5 scans on the second day. This effect was shown with any of the scans taken on the second day that were compared to the first, and only such. Therefore there was a change in the surface topology of the thin membrane-like mask between successive loadings into the EBL system. The only plausible explanation is that of a particulate contaminant beneath the mask and the EBL stage. In particular with the distortion at the top of the pattern the particle would have been very near where the mask was clipped into place on the EBL stage, and thus its effect was more pronounced due to the pressure of the clip. Close inspection of the data reveals the peak displacement due to the dust particle is 0.81μm, significantly larger than the measuring uncertainty of a few tens of nanometres.



(a)





(b)

	'x' (nm)	'y' (nm)
Mean	0.00	0.00
Sigma	86.9	93.7
Three-Sigma	261	281

(c)

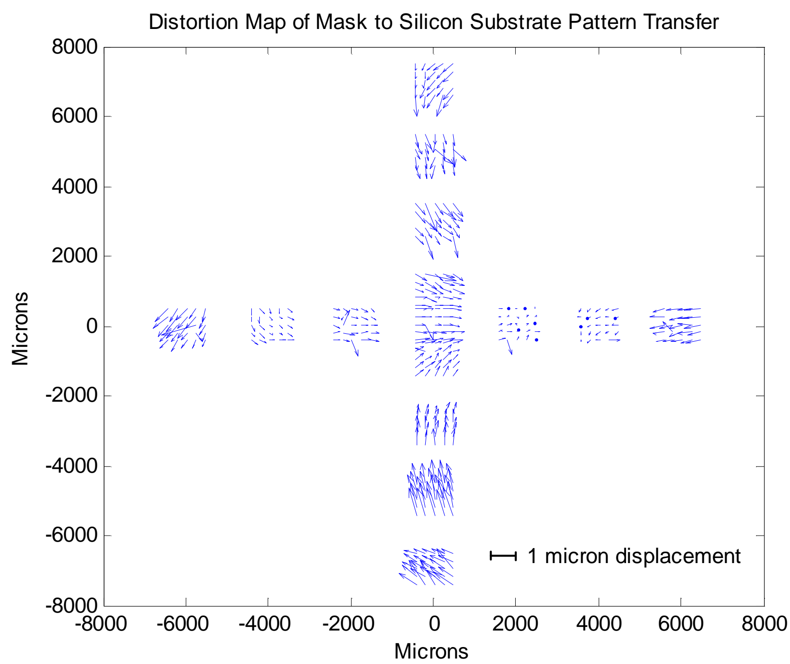
Figure 5.2 Effect of a dust particle. (a) A large localised distortion can be seen at the top of the distortion map. (b) Displacement distribution in 'x' and 'y' and (c) 'x' and 'y' distortion distribution statistics.

It can be seen that the majority of the distribution in ‘x’ is very confined, and in ‘y’ is from -80nm to 60nm, and therefore without the particulate contamination the result would be much the same as in Figure 5.1. From this result, it can be considered that any mask scanned could have been affected by particulate contamination at the mask scanning stage. This must be taken into account when considering the results gained from the measuring process. The issue of particulate contamination is in fact paramount to this work with further evidence discussed in section 6.3.2, and the next chapter dedicated to modelling its effect on the mask membranes. As an aside, a method to detect contamination at the initial mask scanning stage is outlined in the Future Work section of the final chapter.

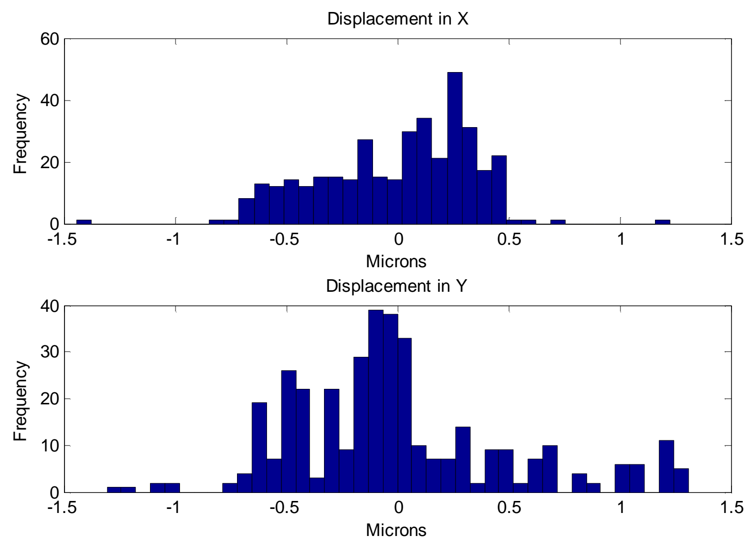
5.2 Initial ENFOL Mask Patterning Distortion

The first ENFOL masks were made using 150µm thick circular glass cover slips that were glued onto 18mm diameter holes drilled into finely ground mask plates used as holders to support the membrane like mask. The first measurement patterns to be used were designed on a presumption that there may be radially uniform distortion from the centre outwards, and thus were designed with arrays at increasing distances from the centre of mask, outwards. This produced the cross shaped mask array pattern seen in Figure 5.3(a), however with the twist like pattern that can be seen in the lower portion of the distortion map it was evident that the distortion was not uniform across the surface of the pattern and for further measurement the mask design was altered so that the much more of the 18mm diameter surface area (given by the hole in the mask holder) would be sampled to more clearly indicate the distortion (Figure 4.15).

The main result from the first experiments where ENFOL-patterned substrates were compared to the first masks was that the distortion measured was very large. The three sigma results of distortion seen in Figure 5.3(c)) being 1.02µm in ‘x’ and 1.48µm in ‘y’ is much larger than that desired for using ENFOL to produce sub 100nm features.



(a)



(b)

	'x' (microns)	'y' (microns)
Mean	0.00	0.00
Sigma	0.302	0.493
Three-Sigma	1.02	1.48

(c)

Figure 5.3 (a) Distortion Map of an early result showing twist distortion in lower portion of pattern, from this it was thought that a PDMS mask could prevent such twisting. (b) Displacement distribution in 'x' and 'y' and (c) 'x' and 'y' distortion distribution statistics.

At this early juncture the same mask had produced similar twist like patterns in the lower extremity of the marker array pattern. It was thought that the process of gluing the thin mask to the mask holder using UV-curing adhesive may be responsible for the skewing of the mask as the glue dried unevenly. Thus a new method of mask mounting was pursued, as described in the next section.

5.3 PDMS Mask Mounts

In the literature [31] successful contact lithography masks have been made that were attached to a mask holder using an Ultem plastic annulus. It was thought that the use of such a toroidal shaped spacer between the glass mask plate holder and the thin glass masks would allow some give during the vacuum process. From the experience of the first mask it was thought that this would reduce the stretch and hence distortion that the glass cover slip would otherwise experience when hard glued directly to the glass mask holder as in the original mask mounts. The high relief of the spacer reminiscent of a mesa (a table top like geographical feature) is depicted in Figure 5.4.

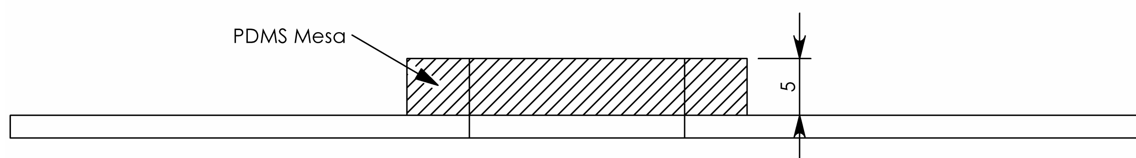


Figure 5.4 The first PDMS mesa mask mount.

The material chosen for the spacer was poly-dimethyl-siloxane (PDMS) silicone rubber. Not only is PDMS flexible, it readily adheres strongly to glass by van der Waals forces, and if treated in a plasma asher can be made to chemically bond to glass. It also can be made to varying viscosity through altering the ratio of curing agent (hardener). A soft consistency was desired ratios up to 1:60 curing agent to silicone elastomer were experimented with. Such ratios provided the consistency of a gel and it was found that a ratio of 1:30 was found to give a soft consistency and still hold vacuum. Additionally the type of glass cover slip used for the masks was changed from 150µm thick circular glass cover slips that of 100µm thick square glass cover slips; thus improving the conformability of the mask.

5.3.1 The First PDMS Mask Mounts

At first it was envisaged to have a mesa-like PDMS spacer that was quite thick to have as much ‘give’ as possible. However, the mesa depicted in Figure 5.5 was too high at 5mm for the Karl Suss MA6 mask aligner. This is because the balloon of the sample holder was unable to make contact with the surrounding glass which would normally create a vacuum seal because of the large gap created.

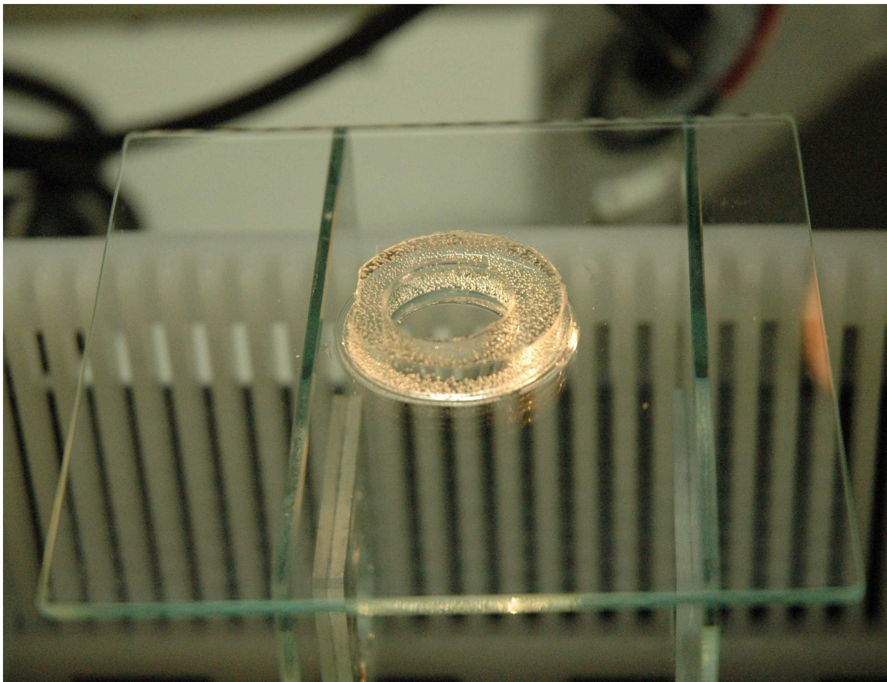


Figure 5.5 First PDMS mount mask holder. PDMS mesa was too high for mask aligner sample holder balloon to make contact with surrounding glass to establish a hermetic seal. Note: later PDMS mounts were adequately degassed and as opposed to the above with plenty of bubbles.

Therefore a new design of mask holder was devised in which the toroidal mesa was surrounded by plastic so that the mask aligner balloon could create a seal to it (Figure 5.6, Figure 5.7). The plastic chosen was taken from a mask plate holder box with 2mm thick material and when the holder was constructed, resulted in that the mesa was slightly shorter by 0.25mm than the surrounding plastic. This difference in height resulted in the mask being pulled down onto the silicon substrate under vacuum.

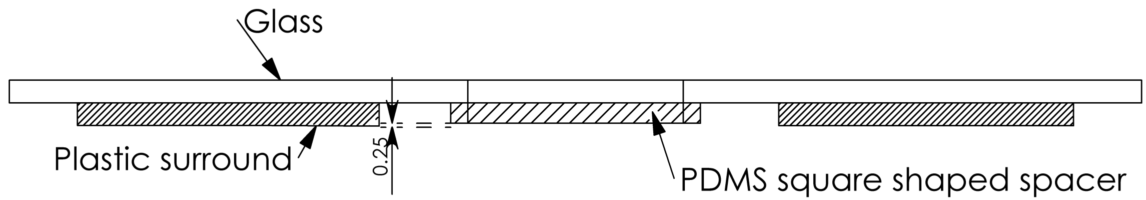


Figure 5.6 Schematic of PDMS mount mask with pull down due to a difference of 0.25mm between the height of the plastic surround and the of the central mesa.

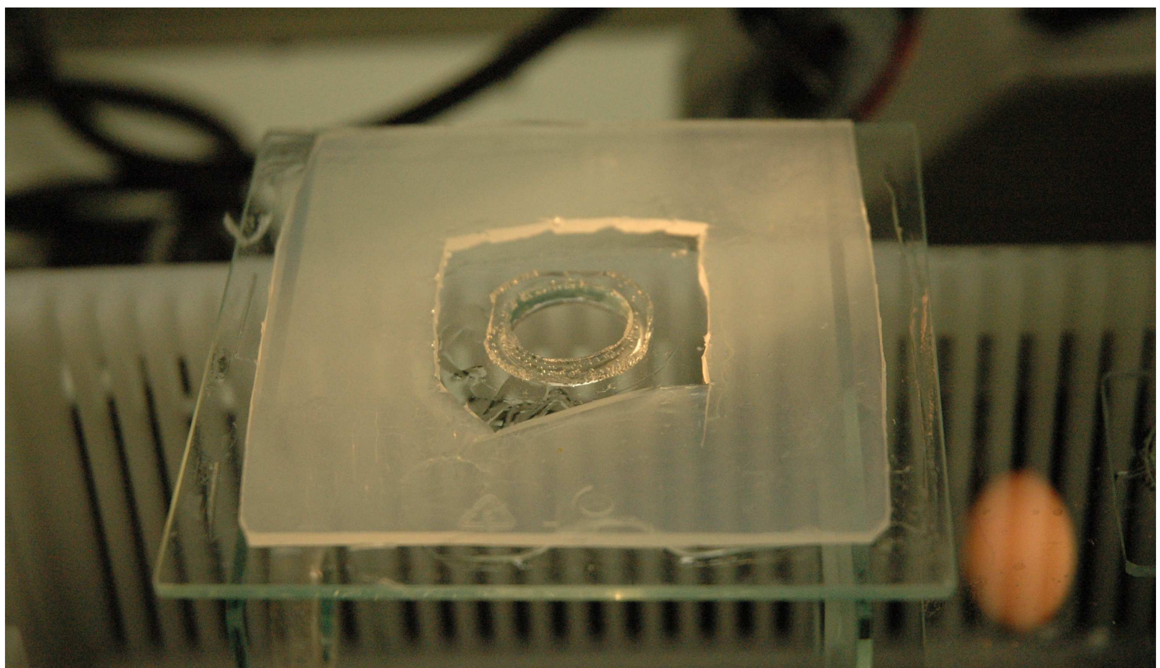


Figure 5.7 PDMS mask pull down mount. The surrounding plastic ensures that the mask aligner sample holder balloon can reach a flat plane to seal against, and the PDMS mesa is slightly shorter resulting in a 'pull' down of the mask under vacuum to the silicon substrate beneath.

One advantage of such a 'pull' down is that it ensures that the mask isn't being pushed up into the PDMS by the substrate during the mask aligners wedge error correction. This is when the mask is lowered and pushed up against the substrate in order to align the mask and substrate in the 'z' axis. Thus if a mask on a PDMS holder were allowed align in this fashion, the PDMS would be squashed as the mask aligner applied pressure and would remain squashed throughout the exposure, negating any beneficial effect of the flexibility of the PDMS mount.

Another advantage is that there would be opportunity for all of the air to be removed from the chamber, which may not be the case if the mask was already in contact with the silicon beneath before vacuum as bubbles of air could remain as the glass cover slip mask adhered via Van der Waals forces to the photoresist-covered sample beneath. Therefore a slight ‘pull’ down could result in the best intimate contact of mask to substrate (Figure 5.6).

The result from this mask was very promising (Figure 5.9(a)) because it yielded a uniform distortion. A radially symmetric systematic distortion that if repeatable could perhaps be compensated for by patterning inversely distorted features on the mask thus cancelling the apparent distortion. This gave hope for a successful resolution to this investigation. The distortion however was somewhat larger than previous (Figure 5.9) with a three sigma result of 1.55μm in ‘x’ and 1.67μm in ‘y’.

5.3.2 The Embedded PDMS Mesa Mask

The design of the PDMS mask mount was altered so that the plastic would not be necessary. This was enabled by making the glass mask holder such that the mesa could be embedded into it (Figure 5.8). It was thought that this design would be easier to manufacture as it didn’t require adhesion of the plastic surround, and the PDMS spacer could be easily poured into a ready made mold (Figure 5.10).

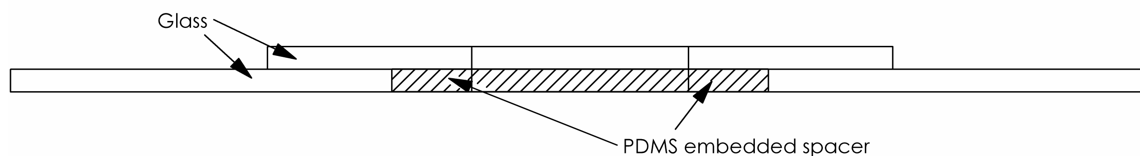
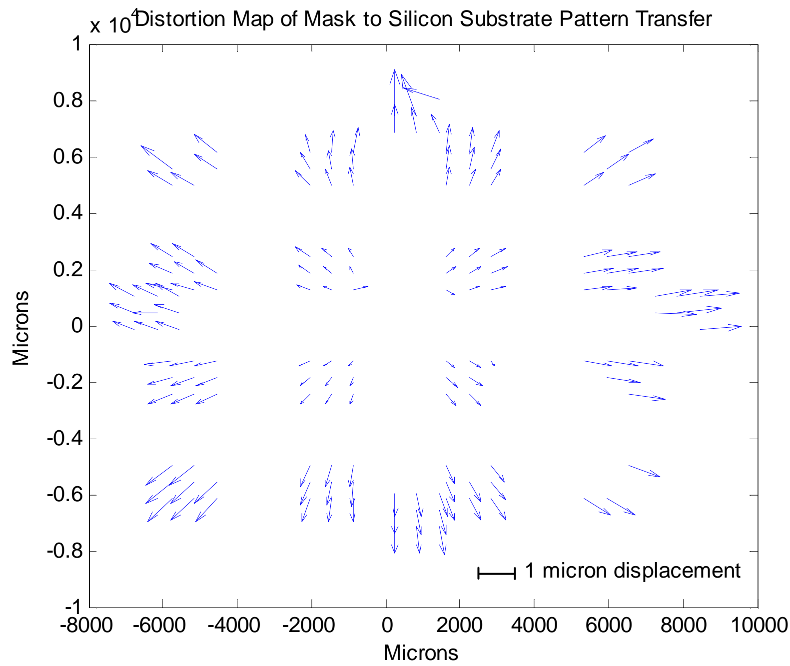
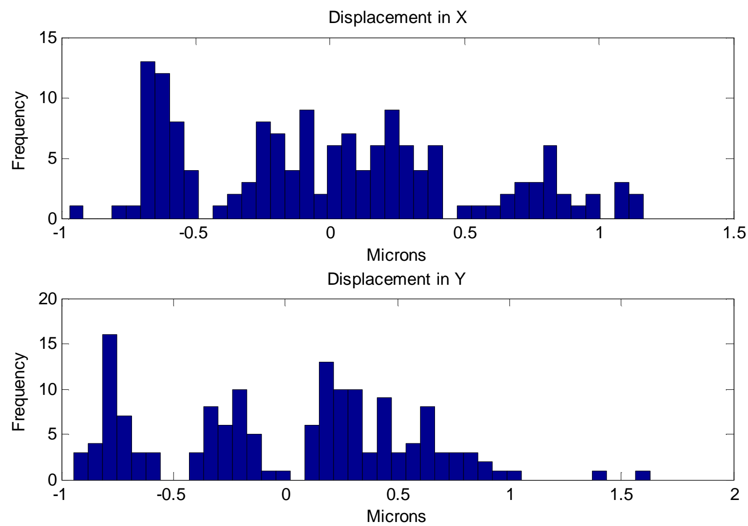


Figure 5.8 Schematic of embedded PDMS mask mount.



(a)



(b)

	'x' (microns)	'y' (microns)
Mean	0.00	0.00
Sigma	0.516	0.558
Three-Sigma	1.55	1.67

(c)

Figure 5.9 PDMS mount mask with 'pull' down. (a) A radially symmetric systematic distortion can be seen. This uniform distortion result gave hope that in-plane distortion could be eliminated. (b) Displacement distribution in 'x' and 'y' and (c) 'x' and 'y' distortion distribution statistics.

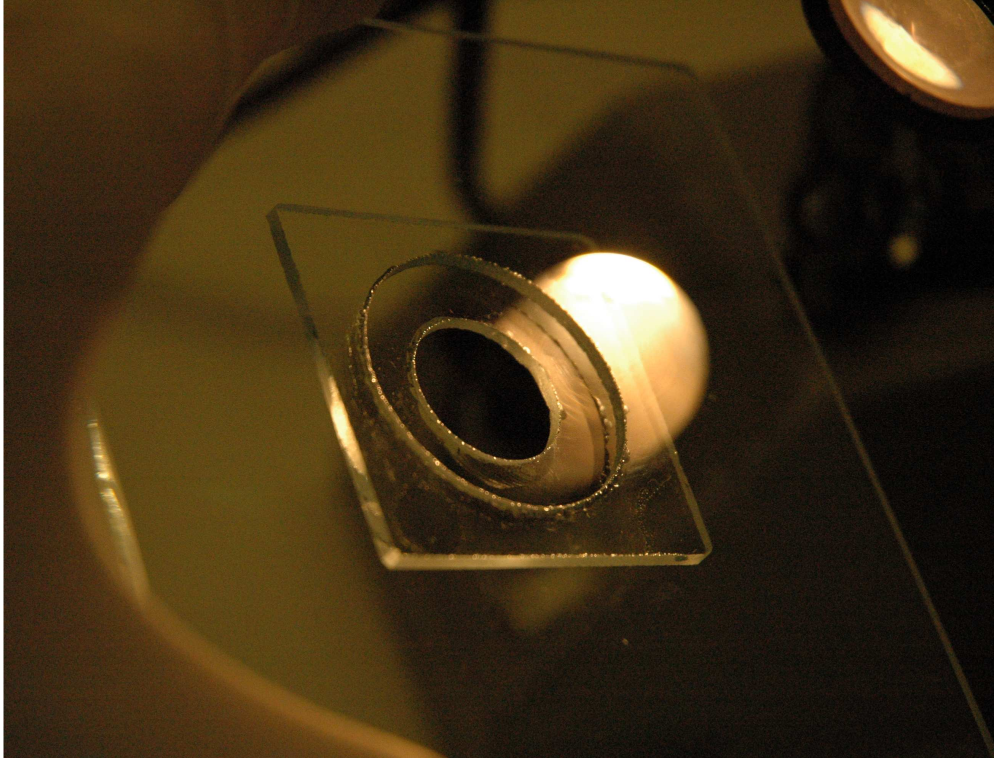


Figure 5.10 Glass mask mount for embedded PDMS mesa.

However it was this design that clarified issues and showed the previous mount to be superior. This is because the use of the plastic surround higher than the of the mesa ensured the PDMS wouldn't be squashed during the mask aligner's wedge error correction (see section 3.1.2) and its design also meant that the PDMS wouldn't 'kiss' the substrate holder sealing in air – preventing a the creation of a vacuum between the mask and substrate. The prevention of the PDMS mount being squashed during wedge error correction was crucial to the success of the previous mount, as any mask mount spacer giving additional pressure onto the mask and substrate, would not only potentially seal in air preventing vacuum, but would negate any positive effect of the PDMS spacer as it would already be squashed up and preventing any desired beneficial flexible properties.

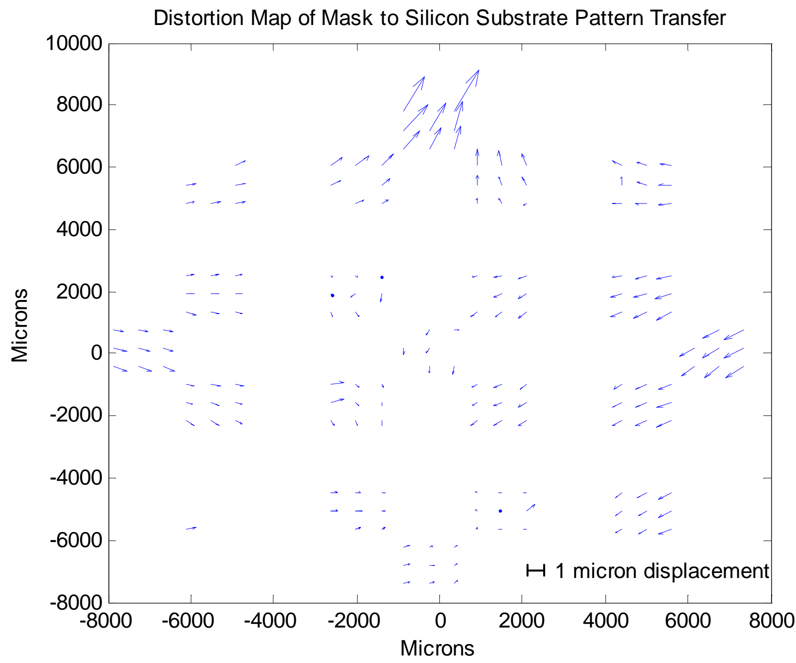
In an attempt to overcome the shortcomings of this mask design an alteration to the process was made for mask once these issues were realised. To avoid the mesa being squashed during wedge error correction, stacks of 2 samples high were placed on a larger substrate holder, this was to create a pull down effect onto the central sample. However,it was difficult to obtain a reliable vacuum as many holes had to be covered in the substrate holder which was designed for a 3" wafer, and the central sample was not held down by a separate vacuum as it would be normally to hold it in place.

More Evidence of Dust.

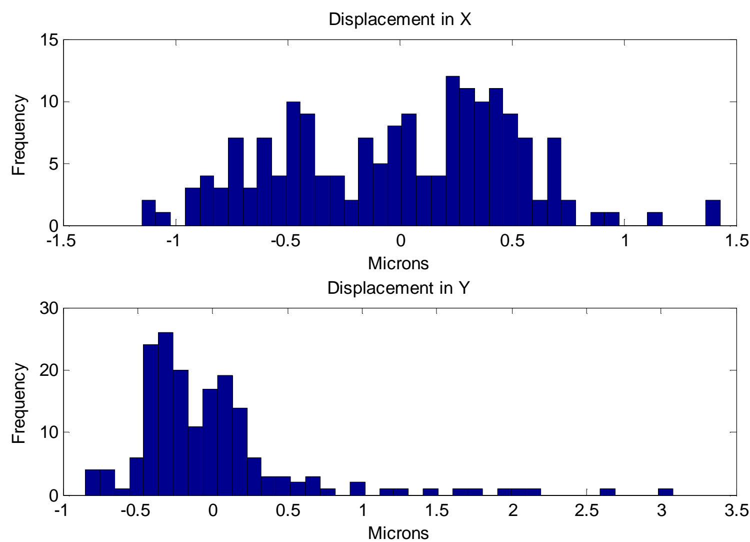
It was at this point of testing the mask holder with a clear cover slip in place of the mask (to check that it wouldn't break under vacuum with the new mask holder) that Newton's rings were observed – spectrum of colours interference pattern due to the glass being distorted in the z direction creating a small gap on the order of the wavelengths of visible light at a varying height and thus displaying varying colours – which are a definite indication of dust present on the sample. Despite an extensive cleaning process the effect of dust could still be seen between the clear cover slip and substrates.

Once the clear glass cover slip dummy mask survived the vacuum process a real mask was mounted. However, this mask lasted for one sample before breaking in the post vacuum process despite this being set to the maximum time available setting on the machine to gently ramp down the vacuum. The results from this mask are a significantly larger distortion than that previously (Figure 5.11)

The distortion resulting from this mask with three sigma displacements of 1.54µm in 'x' and 1.77µm in 'y' is similar in magnitude to that of the previous PDMS mask design albeit with a non uniform distortion. However, while manufacturing and testing the performance of this embedded PDMS mesa mask mount, it was decided to test the hypothesis of that this mask mount would be superior to that of the original mask mount (sans PDMS) and compare the two qualitatively. It was found that the hypothesis was wrong and that original mask mount yielded a much lower distortion. Thus, work on the PDMS mount masks ceased.



(a)



(b)

	'x' (microns)	'y' (microns)
Mean	0.00	0.00
Sigma	0.515	0.590
Three-Sigma	1.54	1.77

(c)

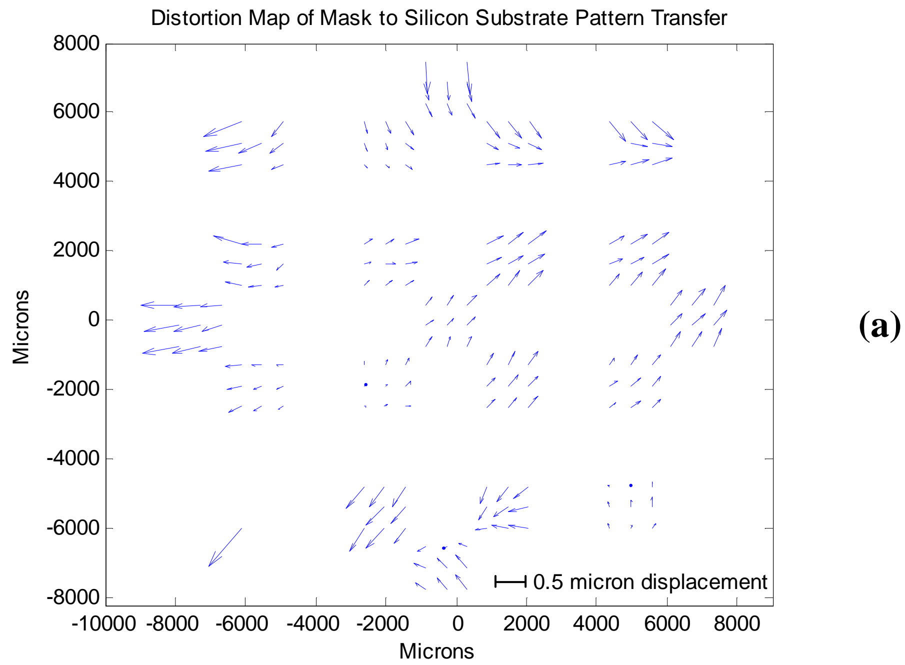
Figure 5.11 Embedded PDMS mount mask. (a) Distortion is non uniform as opposed to the previous PDMS mask mount design. (b) Displacement distribution in ‘x’ and ‘y’ and (c) ‘x’ and ‘y’ distortion distribution statistics.

5.4 Original Mask Mount with 50μm Separation

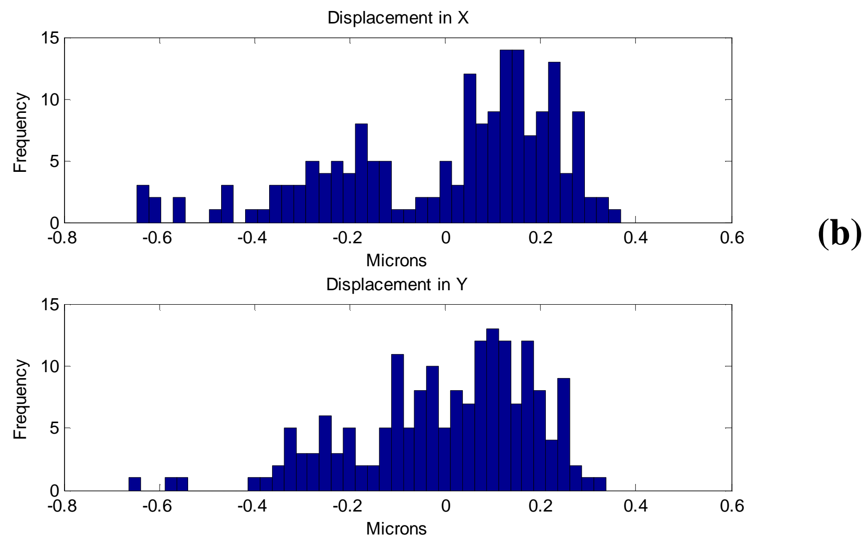
Masks were mounted once again using UV adhesive directly to the glass mask plate holders with 18mm holes drilled in them however there were two important differences. The first was that this time the vacuum was set to hard as opposed to soft as with the first results. And second it was experimented to adjust the wedge error correction offset to provide separation between the mask and the substrate thus creating a pull down effect. A number of exposures were taken at varying separations in 'z' of 0, 25μm and 50μm.

The lowest distortion result overall was found for a separation of 50μm and is shown in Figure 5.12. The amount of distortion measures at a three sigma displacement in 'x' of 0.709μm and 0.560μm in 'y'. The distortion in Figure 5.12 (a) is non-uniform showing several seemingly localised distortions. This result showed that the PDMS mount masks offered no improved over the original mask mount design, therefore ceasing development of the PDMS mount masks.

The other separation heights gave higher overall distortion measurements: a 25μm separation in 'z' resulted in three sigma displacements of 1.09μm in 'x' and 0.523μm in 'y' (Figure 5.13) and for a separation of 0μm the three sigma displacements are 0.904μm 'x' and 1.36μm 'y' (Figure 5.14). With positive separation heights yielding decreased distortion this supports the hypothesis that a pull-down effect is beneficial by allowing air to escape that could otherwise be trapped between mask and substrate via Van der Waals forces if the mask were brought into contact with the substrate prior to vacuum.



(a)

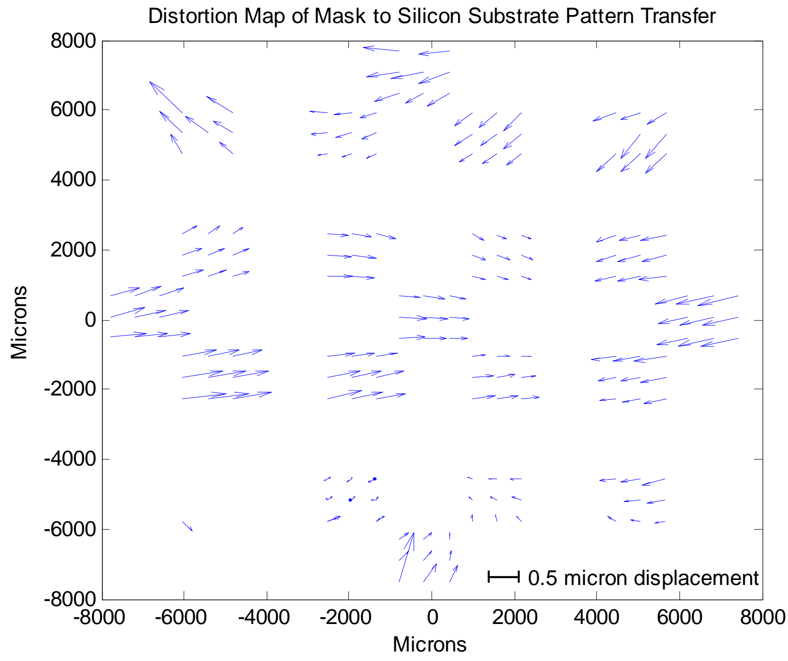


(b)

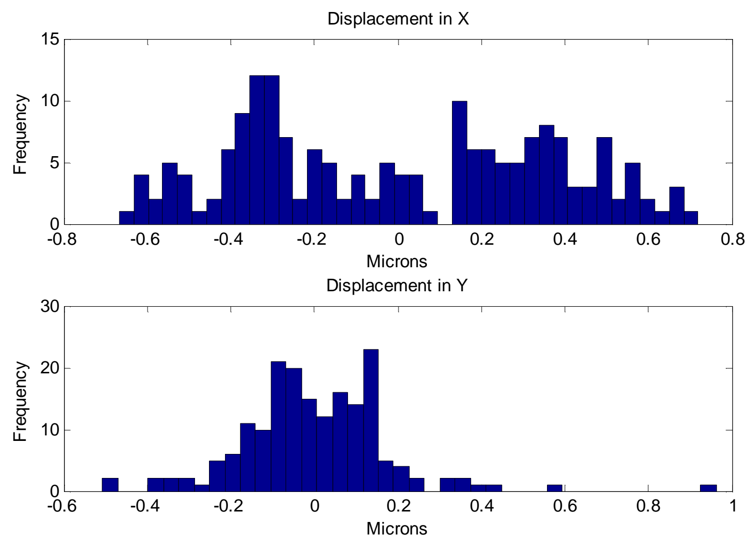
	'x' (microns)	'y' (microns)
Mean	0.00	0.00
Sigma	0.236	0.187
Three-Sigma	0.709	0.560

(c)

Figure 5.12 (a) 50μm separation original mask mount distortion map. Overall distortion is non-uniform with multiple localised distortions. (b) Displacement distribution in 'x' and 'y' and (c) 'x' and 'y' distortion distribution statistics.



(a)

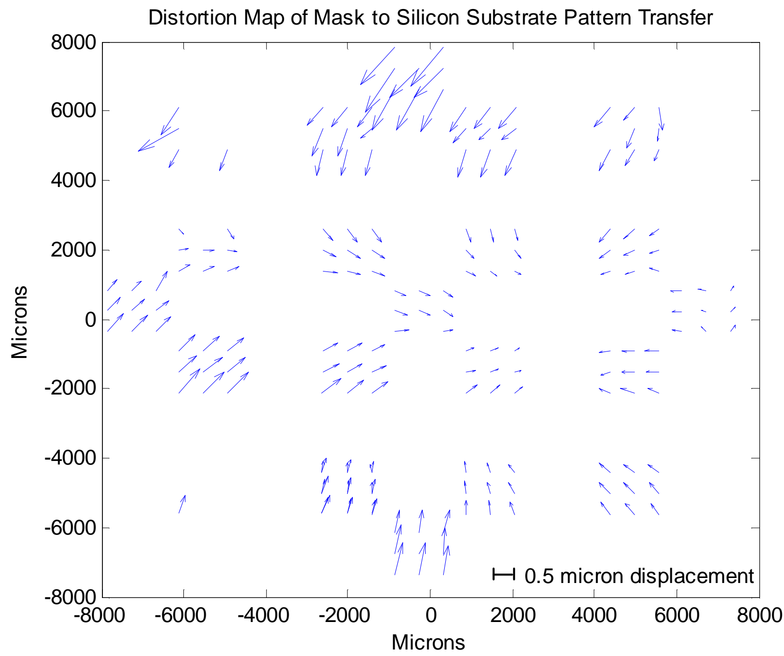


(b)

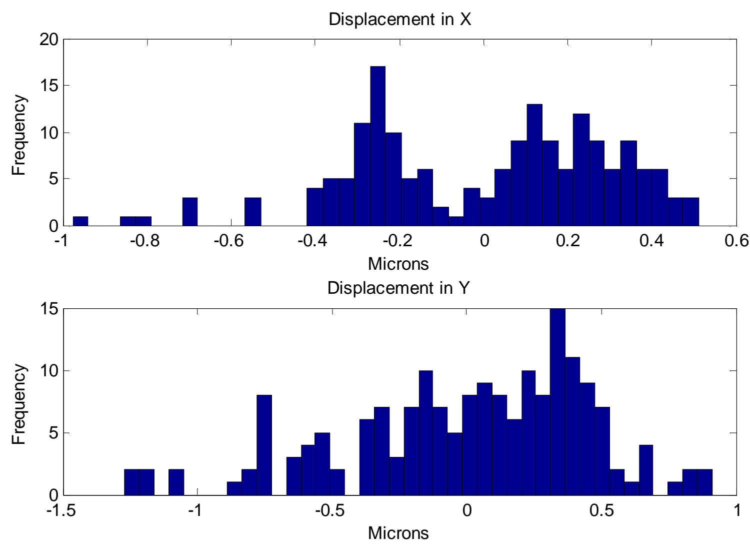
	'x' (microns)	'y' (microns)
Mean	0.00	0.00
Sigma	0.362	0.174
Three-Sigma	1.09	0.523

(c)

Figure 5.13 (a) 25µm separation original mask mount distortion map. Overall distortion is non-uniform with multiple localised distortions. (b) Displacement distribution in 'x' and 'y' and (c) 'x' and 'y' distortion distribution statistics.



(a)



(b)

	'x' (microns)	'y' (microns)
Mean	0.00	0.00
Sigma	0.301	0.455
Three-Sigma	0.904	1.36

(c)

Figure 5.14 (a) 0µm separation original mask mount distortion map. Overall distortion is non-uniform with multiple localised distortions. (b) Displacement distribution in 'x' and 'y' and (c) 'x' and 'y' distortion distribution statistics.

5.5 Summary

The aim of the work was to measure the in-plane distortion of substrates patterned by conformable masks for ENFOL. The main result found is that there is large in-plane distortion in the form of random patterns of localised displacements. The most likely cause is that of particulate contamination, as particle contaminants would stretch and distort the mask membrane to varying degrees (according to their size) and provide the random distribution of localised distortions seen.

An alternative hypothesis would be that of localised thermal effects causing distortion, however this is not supported by the observation of Newton's rings observed as soon as the mask and substrate made contact when a clear test mask was used, thus without instantaneous heat transfer, thermal effects can be ruled out. Another hypothesis is the possibility of out-gassing of the photoresist beneath the mask while in contact causing bubbles beneath the membrane, however such bubbles would take time to form, again the observation of Newton's rings as soon as contact was made also rules out this hypothesis.

The accuracy of measurement of the in-plane distortion was established at 60nm; however this was dogged by further evidence of particulate contamination as shown by the measurement of a single large distortion upon reloading the mask into the EBL system. Overall, the amount of in-plane distortion from ENFOL substrates measured has been on the order of 1µm, this is much larger than what was measured in [31] of 58nm over $2cm^2$ for a mask of similar thickness and makeup. In order to rigorously test that particulate contamination is the cause of the large distortion, the mechanical effects of representative dust particles will be simulated in the next chapter, such that if the magnitude of the distortion resulting from the dust particles matches that observed experimentally, it will prove beyond reasonable doubt the case for particulate contamination.

Chapter 6: Mechanical Analysis

As presented in Chapter 4 a method for measuring the in plane distortion of conformable masks has been demonstrated. This method was developed in order to determine the distortion from the use of ENFOL masks which are by necessity, conformable. What was found from the measurements of ENFOL patterned substrates in Chapter 5 is that the patterns have localised distortions. There is evidence to suggest that this is from significant particle contamination. However in order to prove the case for particle contamination and rule out the possibility of thermal effects, the effect of particles representing the size of dust contaminants beneath the membrane like ENFOL mask will be modelled, such that if the results give in plane distortions of the same magnitude to those observed, it will prove the case for particle contamination and hence reveal the origin of the disordered distortion.

This chapter is divided into two sections: the first of which outlines the method of analysis used to model the mask membrane and the procedures involved in building the model such as assigning material parameters and load conditions. It also describes assumptions made in order to create the model, the issue of the use of units in the simulation software and the justification for the choice of an axisymmetric model. The last section presents the results of modelling the system including the area of the mask available to be in intimate contact, permitting ENFOL exposure and the verdict on whether or not particle contamination is the cause of the observed distortion.

6.1 Finite Element Method Analysis Using Abaqus/CAE

The system to be modelled is that of a pressurised circular membrane against a flat surface upon which a representative dust particle has been placed. The circular glass membrane is thin at 100μm and is 18mm in diameter, and from this surface a precision of 10nm is desired in the measurement of the distortion resulting from the interaction with the particle. To provide an accurate simulation of the system the finite element method (FEM) of numerical calculation was chosen as it is the standard scheme of computation for the modelling of physical systems [48]. FEM calculation

requires that entities to be simulated are divided up into a number of small elements (or discretizations) upon which shape functions are applied. These shape functions approximate physical relationships such as temperature, mechanical stress, or mechanical displacement; the solution to these equations yields the physical quantity (i.e. displacement) for a given discretization point (node) for Lagrangian (nodal) finite elements [49].

6.1.1 Abaqus FEM Analysis Software

The Abaqus 6.4.1 FEM analysis software system was chosen for the simulation. This is available at the University of Canterbury from the Civil Engineering department. Abaqus 6.4.1 comprises a set of four programs: Abaqus/Standard for general purpose FEM analysis; Abaqus/Explicit for dynamic FEM analysis; Abaqus/Viewer for graphical representation of simulation results; and Abaqus/CAE. Abaqus/CAE provides a complete environment for creation and simulation of models calling the other programs for analysis and display of results (Figure 6.1). Abaqus/Standard and Abaqus/Explicit can be called directly (without Abaqus/CAE) so that so that advanced users can construct their models in other packages and subsequently call the Abaqus calculation engines to model their systems. Abaqus/Viewer could then be used independently by such users to read the output from the analysis engines and disseminate desired data.

6.1.2 Assumptions

Before describing the procedure of creating the model, it is necessary to state assumptions made on the nature of the model (in order to produce it) and yet have it still be a fair representation of the real system. The first assumption is that of a perfect vacuum beneath the glass membrane, the simplifies the creation of the model and although in the real system the vacuum isn't perfect, it is negligible compared to atmospheric pressure being applied to the other side of the membrane.

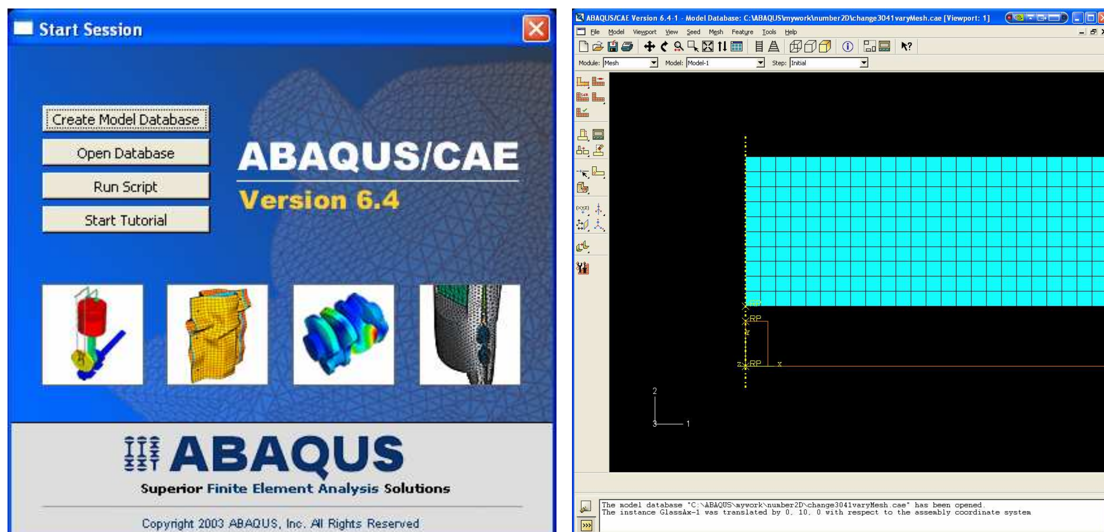


Figure 6.1 (a) Abaqus/CAE startup dialog (b) Abaqus/CAE integrated environment.

Secondly to simplify the analysis, the dust particle was made rigid so that the particle would be of a known size at all times. Similarly the silicon substrate base was modelled as a rigid platform, which is reasonable as the silicon would not deform much, especially when placed upon the thick metal base of the sample holder.

For reasons outlined in the following text it was necessary to make the separation of the membrane from the surface to be the same height of the particle being used so that the two surfaces were in contact at the start of the analysis. It is assumed that this will not significantly affect the measurement of the distortion from the particle as any stretch resultant from the separation height of the membrane will be a localised effect near the edge of the membrane some millimetres away from the particle and its own local distortion effect. The results in the final section of this chapter (Figures 6.8-6.11) bear out this assumption with Figure 6.11 showing that the effect this will have will be on the order of 10nm, which is much less than the distortion caused by the particle being studied.

Because rigid bodies were simulated, the surfaces needed to be in contact at the start of the simulation. Otherwise the simulation failed as the analysis package was unable to model the membrane and rigid bodies without some of the element nodes intruding inside the body of the other. This is termed node over-closure and occurs as the

simulation package tries to incrementally model the motion of the membrane to find compatible geometry. Placing the surfaces such that they are in contact initially prevents node over-closure as the incremental analysis is able to cope without large relative motion of the surfaces. Thus this means that the membrane is not able to be placed at an arbitrary height with respect to the particle beneath, it also means that the membrane can only be at the height that it is in the experimental assembly for a 50μm particle.

The fourth assumption is that the friction between the surface of the glass and the photoresist can be approximated using a coefficient of friction (μ) equal to 1.8, as higher values caused the simulation to fail. Systems with high degrees of intimate contact have coefficients of friction ranging from 5 through to 32 [50] for relatively low pressure (8kPa). This system has a good degree of intimate contact, with high pressure (~101kPa) and strong Van der Waals forces between the photoresist and glass; therefore the coefficient of friction is expected to be much higher than 1.8. The effect of this lower coefficient will serve to decrease the maximum distortion experienced by the membrane due to reduced anchoring of the membrane to substrate.

The final assumptions are that the model is analysed at U.S. standard room temperature (295K) and that the system is isothermal i.e. all entities within the system remain at a constant temperature which will be 295K.

6.1.3 Creating a System of Units within Abaqus/CAE

The Abaqus Analysis user's manual states that "Abaqus has no built in units, except for rotation". "Therefore, the units chosen must be self-consistent, which means that derived units of the chosen system can be expressed in terms of the fundamental units without conversion factors" [51]. This means that when entering dimensions or parameters into Abaqus/CAE there are only number fields for arbitrary units, and the user must keep track of the units (and scale of i.e. milli, micro, nano) that they are using. For instance if a user were to decide to use micrometers and needed to use units of pressure as well, the user would have to covert the units of pressure to be in terms of micrometers instead of meters to keep the units consistent. Why does Abaqus use such a system instead of being able to assign prefixes to SI units?

It is because Abaqus/CAE uses the ACIS geometry engine to represent a part [51]. The analysis manual states that “ACIS is an object-oriented toolkit designed for use as a geometry engine for modelling applications and is considered the industry standard for geometry modelling. The ACIS format uses a tolerance of 10^{-6} to define a geometric entity, such as a vertex or an edge.” Because of this tolerance feature sizes are restricted to a minimum of 10^{-3} units to allow for node and element dimensions that are within ACIS tolerance. There is also an upper limit to using ACIS that of 10^5 units. With such limitation in the geometry engine it can be seen why Abaqus leaves it up to the user (particularly micro-engineers) to specify a unit range that encompasses the magnitude of their model.

For the system to be modelled, a precision of at least 10nm was desired. As the restriction due to ACIS is 10^{-3} of the units used this gives the units to be in tens of micrometres. This would be a cumbersome unit to work with, switching to units of microns gives a lower limit of 1nm and an upper limit of 10^5 = 100,000μm of which the diameter of the membrane at 18,000μm is within range. Therefore the fundamental unit chosen was the micron. This however meant that Young’s modulus (needed for the property of the stiffness the glass) and atmospheric pressure needed to be in terms of microns. The standard unit for these quantities is the Pascal, equal to 1 N/m^2. In order to express this in microns pressure will take the form of $N/\mu m^2$ and therefore dividing pressure in Pascals by a factor of $1x10^6$ will give the necessary quantity. The rescaled units are displayed in Table 6.1.

Table 6.1 Standard Unit and Rescaled Unit Systems.

	Standard Units (metres, kilograms, seconds)	Rescaled Units (microns, kilograms, seconds)
Diameter of Membrane	0.018 metres	18000 microns
Atmospheric Pressure	101325 Pascals = N/m^2	0.101325 $N/\mu m^2$
Young’s Modulus for Borosilicate Glass	$6.28x10^{10}$ Pascals	62800 $N/\mu m^2$

The consistency of the units was tested by creating two models of the same system one using standard units and the other the rescaled units. A square shaped glass rod of length 0.1m with sides of width 0.01m was created for the first model. The Young's modulus for borosilicate glass (the mask membrane glass) and Poisson's ratio (0.200) was applied to the rods material property. A reverse pressure load of 1 atmosphere was applied at one end of the rod while the other end was completely constrained. Meanwhile the beam was recreated to be of the same magnitude in the scaled unit system. The length of the rod was then 100,000μm and 10,000μm wide. The rescaled Young's modulus was applied to the material and a reverse pressure load again of 1 atmosphere applied, but using the scaled units of pressure. The results can be seen in Figure 6.2 and Figure 6.3 where the same amount of displacement has been found in each simulation thus confirming that the rescaled units are consistent with the standard units, and thus with reality. It may be worthy to note that the other property necessary for the specifying the stiffness of the glass, its Poisson's ratio, does not need to be scaled as it is unitless by definition.

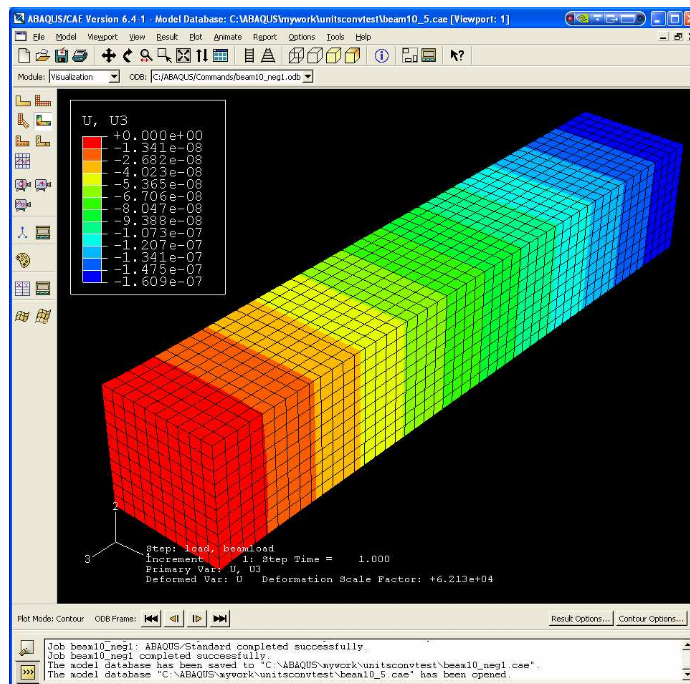


Figure 6.2 0.1metre length beam under reverse pressure load of 1 atmosphere. The legend shows the displacement in the 'z' component. The maximum displacement in 'z' is -1.6x10^{-7}m or 0.16μm.

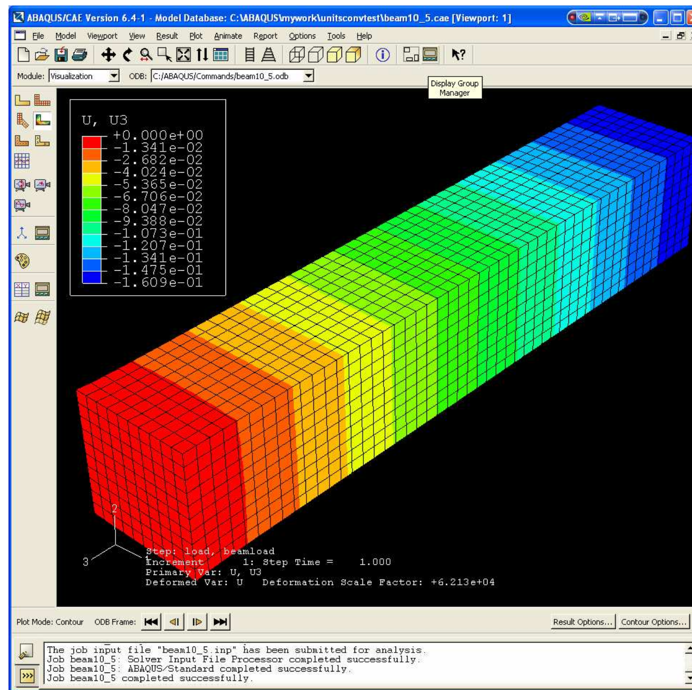


Figure 6.3 The same beam as in Figure 6.2, but in terms of the rescaled units in terms of microns (the legend is in microns). 100,000µm length beam under the same pressure load has a displacement of -0.16µm, matching the result in Figure 6.2 and thus passing the test for unit consistency.

6.1.4 3-Space, 2-Space and Axisymmetric

When it came to building the model in the Abaqus/CAE software package, the first decision to be made was that of what space within which to build the model. There are three spaces to be chosen from: three dimensional space, two dimensional space and axisymmetric. The initial decision was to build and simulate in 3D, as this is the most familiar and therefore seemingly the easiest to understand. However it is the most computationally expensive and the most prone to instability. The 3D simulation of the 18mm diameter membrane, measuring down to the nanometre scale requires measurements to be taken over 254mm^2 or 254x10^{6}µm^2. Solutions investigated included partitioning the membrane so that elements of different sizes could be used with higher density elements where greater precision was desired. With the amount of

area to be covered and the precision desired, the 3D system qualified as a large scale simulation. The author in [48] states that "the simulation of large deformations continues to be challenging as the currently available software in the market still faces difficulties" and therefore it is possible that the nature of the system may not have been favorable to 3D simulation using the Abaqus analysis engines.

Alternatively the system could be built in two dimensional space. This would mean building the membrane as a slice upon a 2D plane. The slice of membrane in turn approximates a very large sheet going into and out of the flat 2D plane. However the system to be modeled is that of a *circular* thin glass membrane, upon which the distortion of only one particle needs to be measured. The final space perfectly fits this, as it provides a space that is symmetrical about a central axis, modeling the circular membrane and upon placing the particle in the centre of the membrane, the distortion is radially symmetrical about the axis of symmetry as it would be in the real case Figure 6.4.

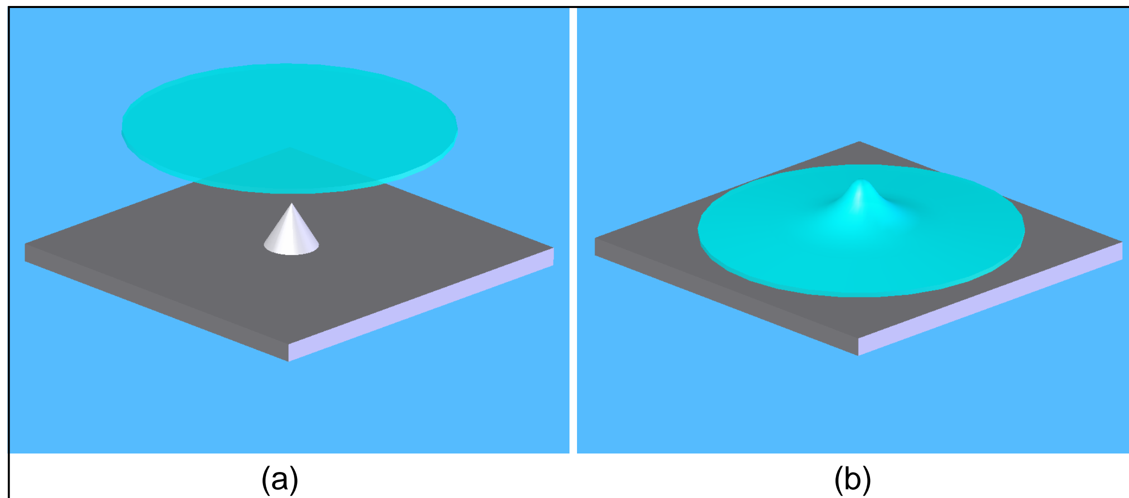


Figure 6.4 (a) Mask membrane and particle upon silicon substrate. (b) Axially symmetric distortion of mask membrane resulting from interaction of centrally located particle.

Therefore the axisymmetric modeling space was chosen to build the system within, and without any drawbacks, as despite being restricted to placement of objects along the axis of symmetry only one particle needs to be modeled and centrally placed it may be. Additionally the axisymmetric space requires the least computation time permitting a higher seed count, and thus obtaining higher precision.

6.1.5 Model Creation Using Abaqus/CAE

Abaqus/CAE is divided up into several modules the majority of which are used to construct the model, with the final two for submission/monitoring and evaluation of the simulation respectively. The modules encapsulate logical steps in the simulation process such as: defining the geometry of the system, assigning physical properties, specifying physical constraints, applying loads and generating a mesh that defines the number and placement of the simulation elements. Upon completion of the construction modules, an input file is generated by the job module to Abaqus/Standard or Abaqus/Explicit. The modules generate an output database which is then opened by the Abaqus viewer module (which incorporates the use of the Abaqus/Viewer program).

There are seven modules used in building the model. The first is the part module which allows the user to create entities in 2D, 3D or axisymmetric and is used to edit and manage the created features. The glass membrane was formed as an axisymmetric deformable solid of radius 9000µm and depth 100µm. An analytical rigid axisymmetric surface represented the silicon base and rigid particle.

The next module, the property module, is used to define materials, such as iron, aluminum or glass. Using the property manager within the module the material properties of glass were entered, specifically the parameters for the borosilicate glass of the ENFOL masks: Young's modulus equal to 62800 N/µm^2 in the rescaled units (section 6.1.3) and a Poisson ratio of 0.2. A section was then assigned to the glass membrane and the property of borosilicate glass assigned to it.

The parts are all brought together upon being instantiated into the assembly in the assembly module. Parts are instanced into a common space in the assembly module. Parts can be rotated and translated to build up the assembly at will. Parts can also be placed by placing position constraints so that specified faces, edges or vertices are made to align to one another. A model may have many parts and these can be instantiated multiple times into the assembly, however there may be only one assembly per model. The assembly space is what all further steps apply to and therefore if a model has only one part it must still be instantiated into the assembly in

order for the generation of the model to continue. Within this module the silicon base was instantiated with the dust particle placed on top, upon which the glass membrane was instantiated bringing it directly on top of the particle and in contact.

The fourth module is the step module which facilitates the creation of different stages in a simulation. For example if a given membrane initially at rest had differential pressure applied followed by heating some time later – the step module would be used to create the different steps their nature – i.e. ramping the pressure and their time of execution: temporal position and duration. The step module gives the user the important control of selecting which information of the FEM simulation are recorded to the output database – the database that will hold the results of the simulation. This module was used to create the pressure load step, upon which the pressure is ramped up to atmospheric pressure.

The fifth module, interaction is used to define the contact between surfaces magnitude and type of friction or lack thereof. If any surfaces are to come in contact their interaction must be described in order to bring about a valid analysis result. In this case this module was used to create a Lagrange multiplier friction interaction between the surfaces, using a coefficient of 1.8 to approximate the friction between the surfaces as discussed in section 6.1.2.

The load module is used to define forces to the system, such as a pressure load on a membrane, to boundary conditions; such as translational and rotational constraints to the edges, faces, vertices of parts. An example is that of a truss beam anchored by a pin joint, the end of the beam is translationally constrained but is free to rotate about the axis of the pin. The load module was used to constrain the edge of the glass membrane in 'x' and 'z' and to completely constrain the motion of the silicon base and dust particle rigid bodies (Figure 6.5). It was then used to apply the atmospheric pressure load to the upper surface of the mask membrane, applying a pressure of 0.101325 N/μm^2 in the scaled units. Figure 6.5 shows the pressure load applied to the glass membrane, depicted as purple arrows and the boundary conditions constraining the membrane and rigid bodies – the orange and blue arrows.

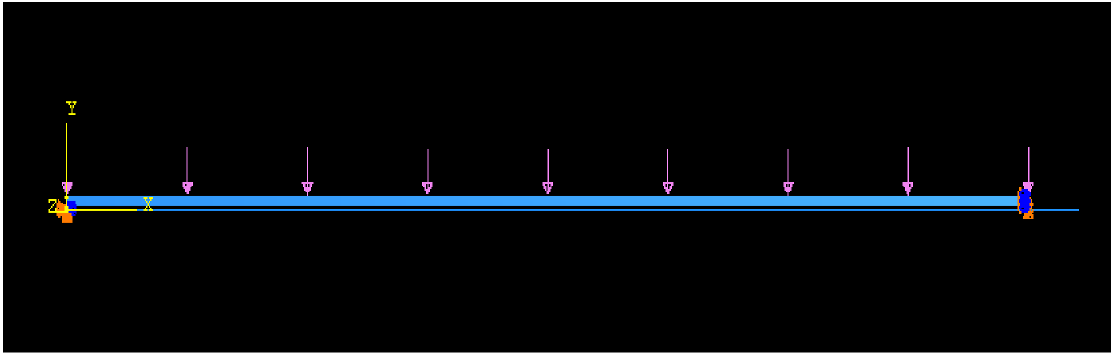


Figure 6.5 Pressure load (purple arrows) and boundary conditions (orange and blue arrows) applied to the glass membrane and rigid bodies in the load module.

The final model construction module is that of the mesh module which defines the number, placement and shape (type) of the elements used to create the FEM model of the system to be simulated. There are a few steps to creating a mesh, which begins with selecting the shape of the element, of which the quadrilateral shaped elements were chosen for this model, alternatives are quad-dominated and triangular shapes. Next the technique of generating the placement of the elements is chosen, of which the structured technique was chosen as the shape of the membrane to be meshed is simple and rectangular. Alternative techniques include sweep for objects with rotational symmetry and the free techniques when it is not possible to mesh with the first two techniques.

Upon selection of the mesh controls, seeds are then assigned to the model at a given spacing, or number along a specified edge (or section of a part). The seeds specify the size of the elements such that when the mesh is then generated, the program attempts to place the elements as close to the specified seeds as possible. This is shown in Figure 6.6(a) with the placement of the seeds, and the resultant elements generated with the mesh is generated in Figure 6.6(b). Additionally, the mesh module is able vary the size of elements over a single part through the use of partitioning. Once partitioned a part may be seeded unequally to produce the desired variation in element sizes. For this system the glass membrane was seeded with 10µm spaced seeds obtaining 10µm quadrilateral structured elements totaling 9000 elements. Figure 6.7 shows a partial view of the meshed membrane as the elements are too fine for screen capture when showing the entire membrane.

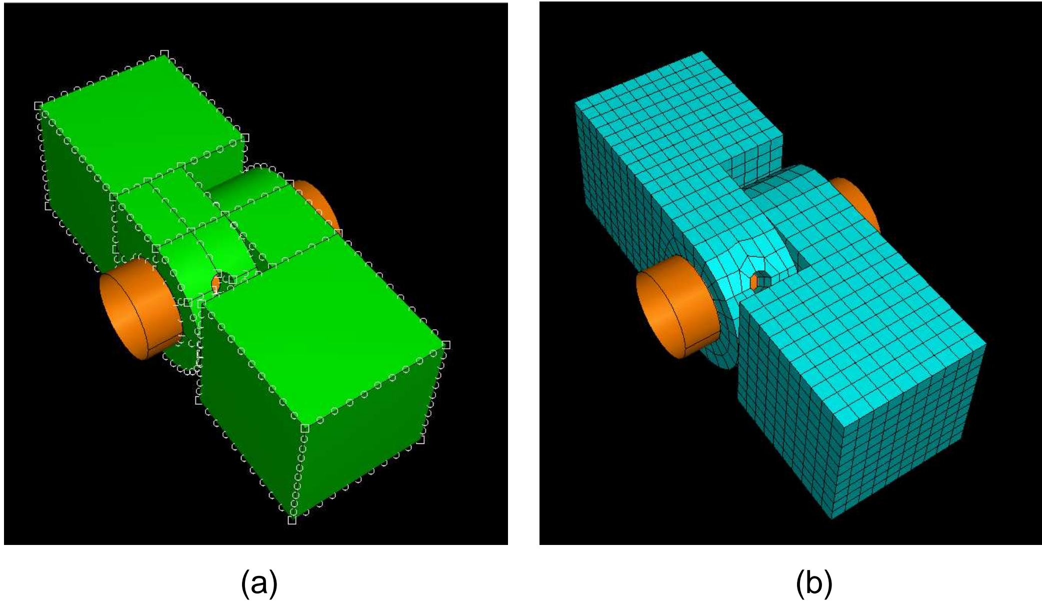


Figure 6.6 Example of seeding: (a) Seeds for the placement of elements are placed on the model. (b) The mesh is generated with the elements being placed as closely as possible to the pattern dictated by the seeds.

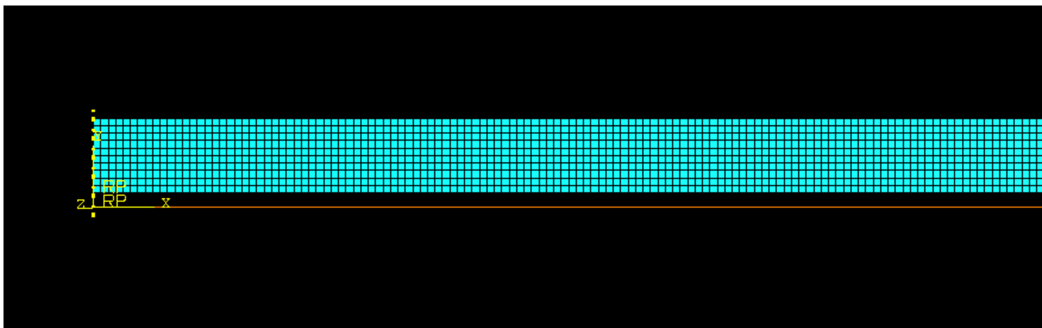


Figure 6.7 Partial view of meshed glass mask membrane. Elements are 10x10µm.

Upon completion of the construction modules the completed model of the system is ready to be submitted to the Abaqus/Standard and/or Abaqus/Explicit FEM analysis engines. The job module is used for this purpose and is able to keep track of multiple output databases for a single model, such that many results generated from one model can be kept together.

6.2 Simulation Results

6.2.1 Distortion effects from singular dust particles

Dust particles that are present as contamination are generally too small to be seen by the naked eye as any large ones that are seen are promptly dealt with, such particulates are on the order of 50µm or less. In this section the results from simulating the effect of dust particles of 40µm, 30µm and 20µm on the ENFOL mask membrane respectively are presented. The first result in Figure 6.8 shows a negative distortion in 'x' to be of 1.05µm for a 40µm high particle.

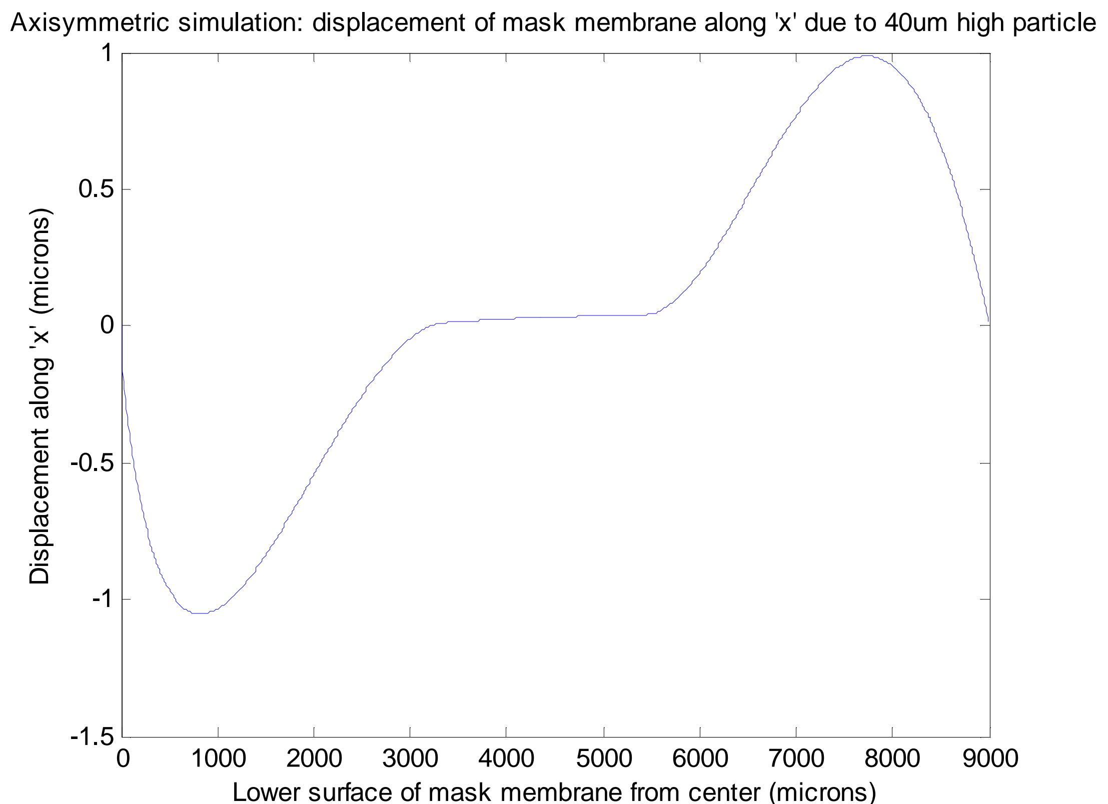


Figure 6.8 The axisymmetric simulation of a 40µm high particle underneath an ENFOL mask. A distortion of -1.05µm is evident from the particles presence. Also the distortion from the raised edge of the mask can be seen.

This result agrees with observed distortion of magnitude 1µm seen on the first mask mounts in Figure 5.3 and embedded PDMS mount in Figure 5.11. It also agrees on the spread of the distortion of ~3mm from the particle, as this is of the same magnitude of

spread as that seen in Figure 5.2 of a single localised distortion and within the range of distortion spreads seen in other distortion maps in Chapter 5. The other distortion evident in Figure 6.8 is the stretch from the edge of the mask; although raised the same height as the particle from the surface, the membrane from the edge requires a slightly further distance to recover ~3.5mm. The next particle of 30μm was modelled with the results in Figure 6.9.

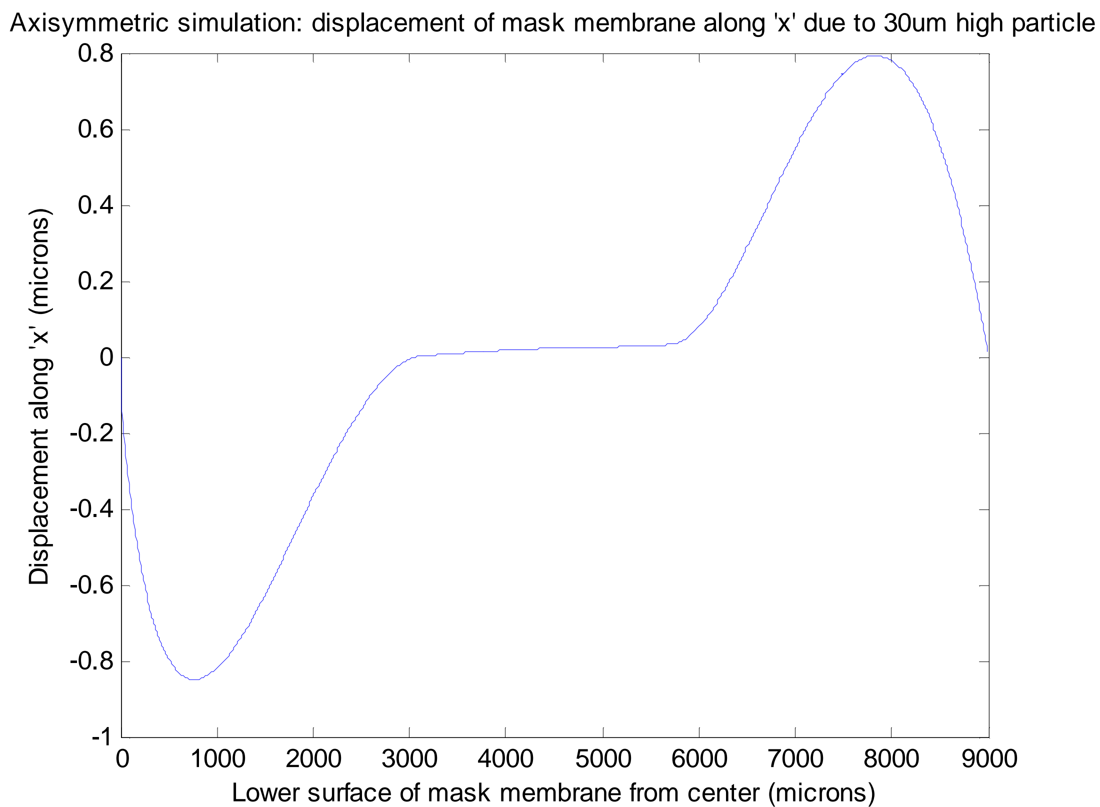


Figure 6.9 Axisymmetric simulation of a 30μm high particle underneath an ENFOL mask. A distortion of -0.85μm is evident from the particles presence.

The result in Figure 6.9 of 0.850μm matches the magnitude of the distortion of a suspected singular particle measured between subsequent scans of the same mask (Figure 5.2 in section 5.1.1) of 0.81μm. The spread of the distortion, slightly decreased from that in Figure 6.8, again agrees with that seen in the experimental results in Chapter 5.

The next particle to be modeled is 20µm high and the results of which are shown in Figure 6.10. The magnitude of this distortion at 0.625µm again agrees with results, specifically it is of the same magnitude as the highest distortion experienced (~0.6µm) in the lowest distortion result observed in Figure 5.12 for the original mask mount with 50µm offset in ‘z’. Also the spread of the distortion decreasing with the height of the particle again agrees with the magnitude of that observed in the experimental results.

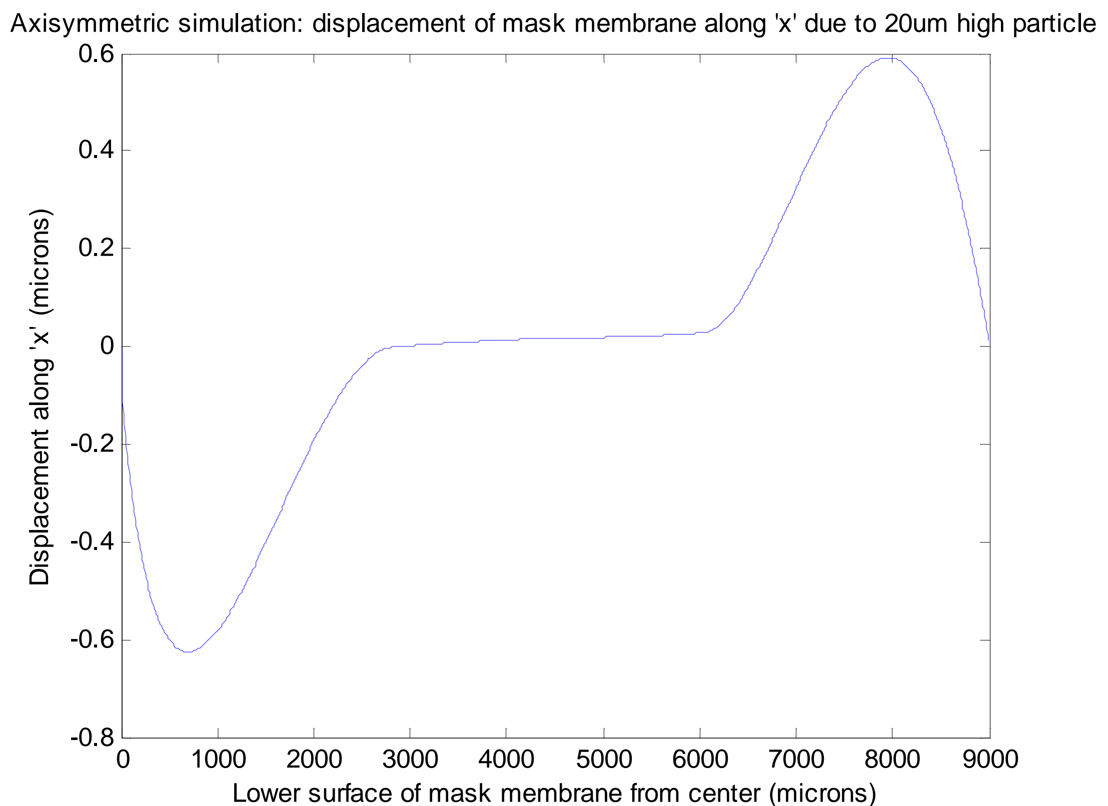


Figure 6.10 Axisymmetric simulation of a 20µm high particle underneath an ENFOL mask. A distortion of -0.625µm is evident from the particles presence.

If it were desired, even closer matches could be made to the observed distortions by adjusting the sizes of the particles by increments of microns. However, this is not necessary, the notion itself indicates that the simulations agree with the observed results and thus the case for particulate contamination could be considered proved beyond reasonable doubt.

6.2.2 Residual distortion in ‘x’ of membrane due to friction

A significant observation is that the area between the high distortions for each of the three simulations (where the membrane is in contact with the substrate) is not perfectly flat; there is a residual positive distortion in ‘x’. This is due to friction anchoring the membrane as it descends upon the photoresist covered substrate. This is compared with a frictionless simulation resulting in no distortion in ‘x’ where the membrane was in contact. The displacement in ‘x’ is shown alongside displacement in ‘z’ to indicate where the mask is in contact with the substrate (Figure 6.11) using data from the three simulations of 20, 30 and 40μm separations. An increase of 10nm of distortion in ‘z’ can be seen per additional 10μm initial mask substrate separation in ‘z’.

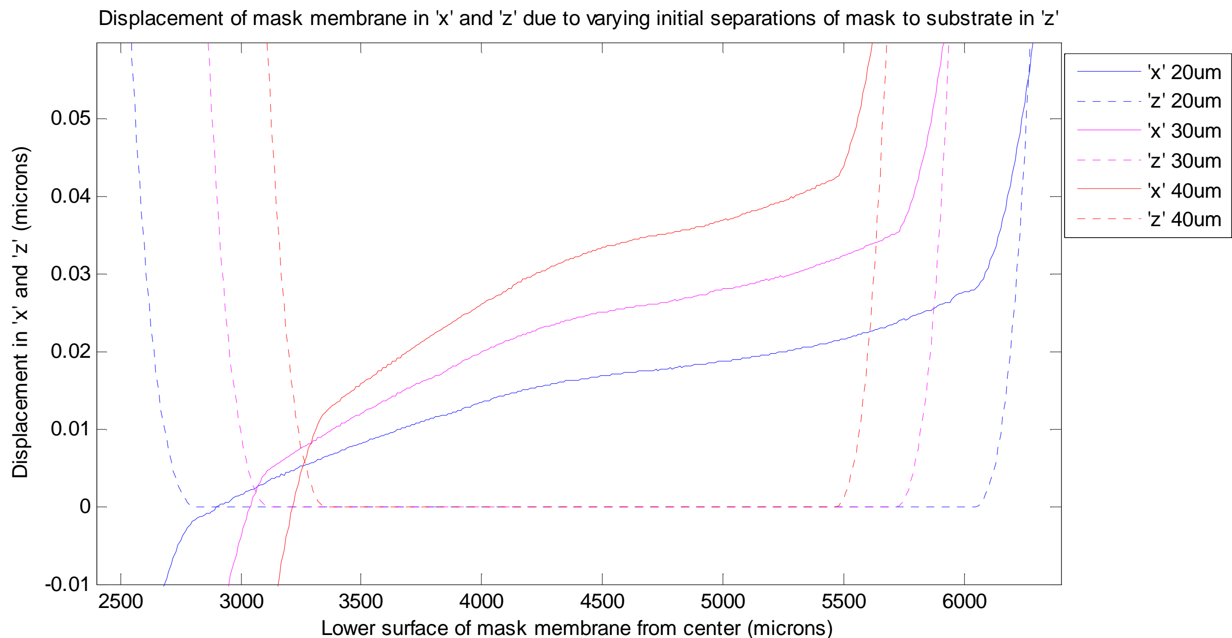


Figure 6.11 Displacement of mask membrane in ‘x’ and ‘z’ due to varying initial separations of mask to substrate of 20,30 and 40μm in ‘z’. Residual displacements of mask in ‘x’ (solid lines) can be seen where mask is in contact with substrate, due to initial separation height in ‘z’ stretching mask membrane. Displacement in ‘z’ is depicted (dotted lines) to indicate contact of mask with substrate.

Note: the sharp drop off to the left of the data lines due to the particles obscures the effect somewhat, however attempting to remove the particles to isolate the residual stretch resulted in simulation failure – possibly due to more elements being in contact complicating the model.

The cause of this residual stretch of some tens of nanometers is simple: the membrane mask stretches down toward (is pulled-down to) the substrate, and while it does, it will stretch evenly in ‘x’. However, once the membrane comes into contact with underlying resist it will adhese to it – retaining stretch (and hence distortion) in ‘x’ from pull-down. It is this residual stretch that is seen in the simulation results (Figure 6.11) and this will be a glimpse of the true distortion of the ENFOL mask. As discussed earlier (in section 5.3.1) a pull-down is necessary to rid the space beneath the mask of potential air pockets, however, from this observation, this comes at a cost of in-plane distortion as a function of separation height. The distortion predicted by these simulations is on the order of a few tens of nanometers and will be a uniform, radially symmetric distortion. Such a uniform distortion will be able to be designed for; with features patterned on the mask with an inverse distortion, to the end that features are patterned with correct geometry upon transfer to the substrate.

The last section of this chapter deals with the usability of the mask area for ENFOL exposure. It follows up on the observation that a lot of the mask area is lost from having had the edge of the mask raised off the surface of the silicon beneath, by only a few tens of microns.

6.2.3 Usable Mask Surface Area

Upon seeing how far the in-plane distortion extends from the particle, some 3mm from the 30µm particle (Figure 6.9), it was also noticed how far the distortion from the edge of the mask is. The distortion from the edge is due the edge being 30µm offset in ‘z’ from the surface of the silicon. As discussed previously in Chapter 6 it is necessary for there to be such an offset to facilitate a pull-down of the mask to substrate to eliminate any pockets of air that may remain prior to vacuum. With an offset needed there will always be a certain amount of unusable surface area extending from the edge of the mask holders hole to the centre of the mask dependent on the height of the offset. This amount of area that is not in intimate contact with the mask preventing ENFOL exposure is revealed through simulation.

Thus the displacement data in ‘z’ was gathered (Figure 6.12) and it can be seen that distortion free area in ‘z’ matches the low distortion zone in ‘x’ (Figure 6.9). Thus the usable area of an ENFOL mask with a 30μm offset in ‘z’ is the central area with a radius of 5.73mm or 11.46mm in diameter. Experiments with ENFOL masks (described in section 5.4) used an offset of 50μm. It is desired to observe the relationship of offset to usable surface area and this can be found through simulation. To this end, offsets of ranging from 10μm through 50μm were analyzed with the results presented in Figure 6.15 [52].

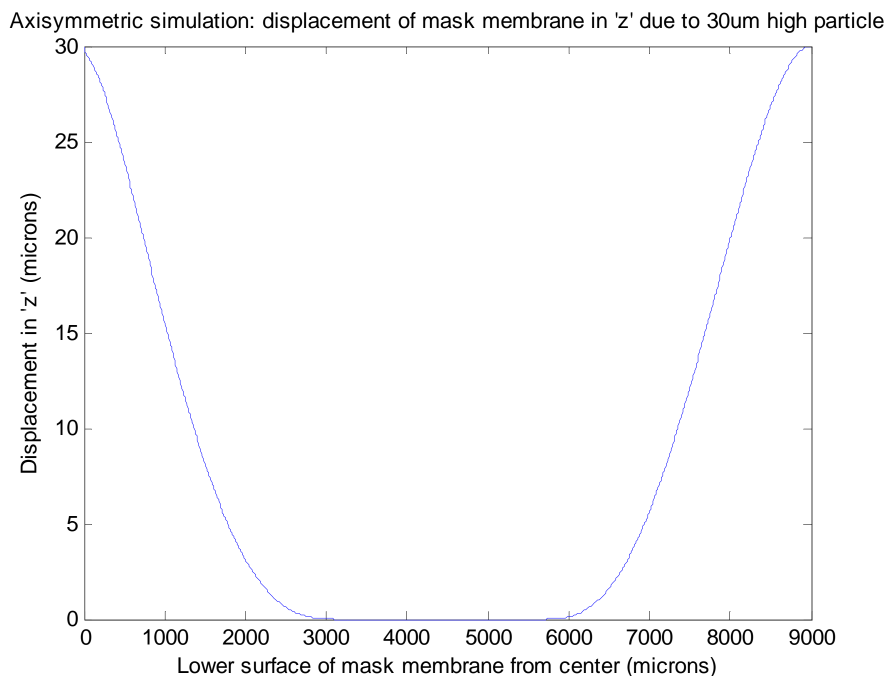


Figure 6.12 Simulation of a 30μm high particle and displacement in ‘z’. It can be seen that from 30μm topological difference, either due to the dust particle or the edge of the mask that the membrane takes thousands of microns to come back into intimate contact with the substrate.

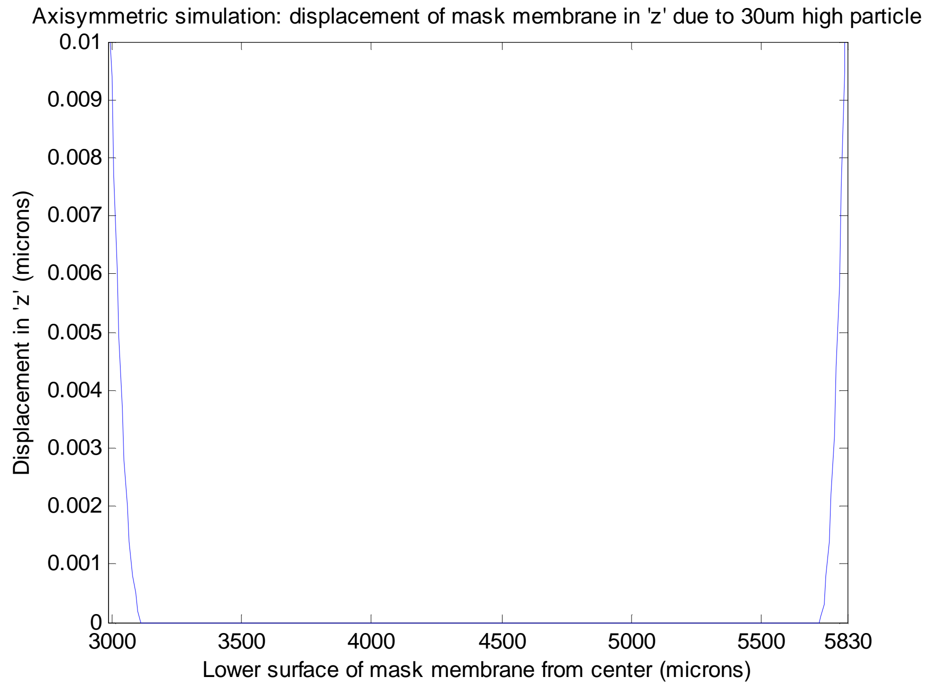


Figure 6.13 Area of mask where contact is within 10nm in 'z'. This represents only one third of the masks radius.

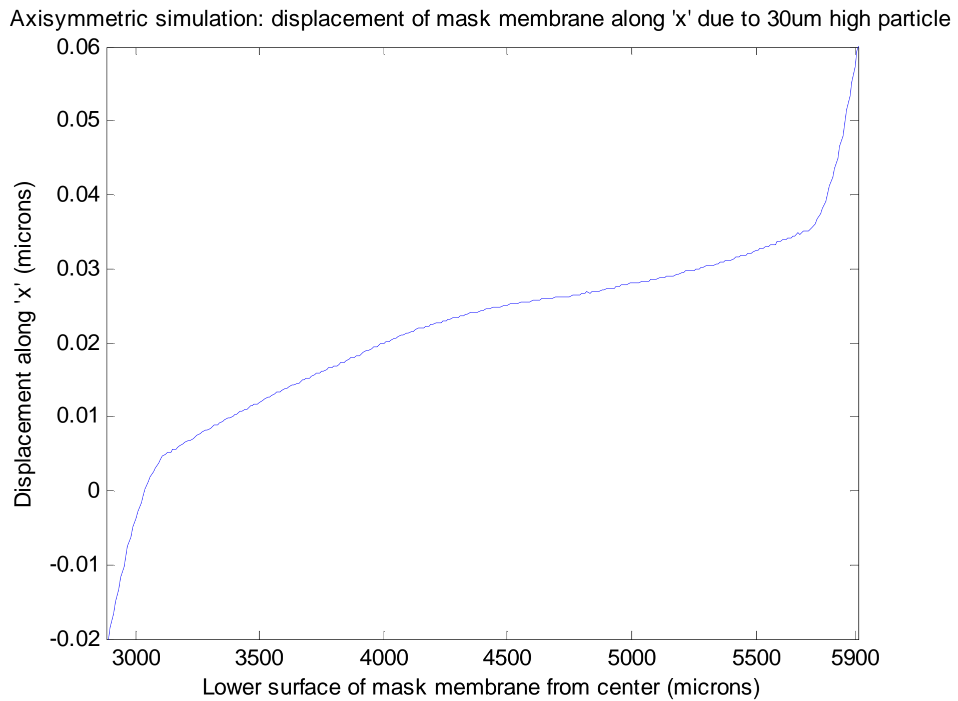


Figure 6.14 Area of mask membrane with distortion within +/-40nm in 'x' for 30µm topological interference.

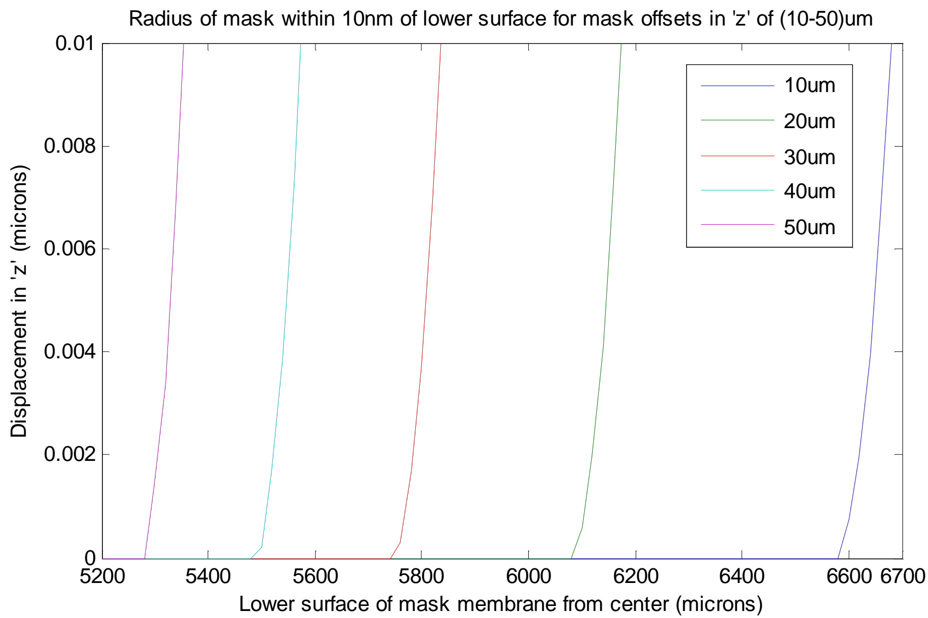


Figure 6.15 Usable radius of mask for varying offsets in 'z'.

The relationship is non-linear, it can be seen that varying the offset from 50µm to 10µm increases the usable mask radius from 5300-6600 which is ~1300µm. The usable surface area goes up from 88mm^2 to 137mm^2.

In order to find the optimum offset, further experimentation, free from particulate contamination would yield a minimum offset maximizing the usable surface area while still eliminating any air pockets – the limiting factor. The elimination of the air pockets could be verified by patterning features beyond the diffraction limit to utilize the evanescent near field and systematically placing these over the surface.

In summary, the use of FEM simulation has uncovered three important results: the first and foremost is that evidence has been gathered that proves the case for particle contamination beyond reasonable doubt; the second is that the stretch of the mask membrane due to necessary initial mask-to-substrate separation, has been modeled, revealing the true distortion of the mask; finally the usable surface area of an ENFOL mask with given initial separation in 'z' has been predicted. The next chapter is final chapter in this thesis, outlining future work and concluding statements upon the results uncovered in this work.

Chapter 7: Conclusions and Future Work

In this thesis the in-plane pattern distortion resulting from the use of conformable masks for Evanescent Near Field Optical Lithography (ENFOL) was studied. This was of interest as by ascertaining the distortion resulting from the use of such conformable masks, the distortion could be compensated for such that predictable patterning within tolerance would be possible. ENFOL is a promising technique of low-cost high resolution lithography pioneered here at the University of Canterbury. It has immediate practical relevance for use in research laboratories and garners commercial interest from Japanese electronics giant Canon. ENFOL and enhancing technologies are active research areas and recently significant advances have been made by researchers at Canon with the production of a prototype stepper and the highest resolution results patterned to date [30]. ENFOL achieves sub diffraction limited resolution through the use of the evanescent near field, which decays within a few tens of nanometres beneath a mask. This necessitates the use of conformable masks for intimate contact.

7.1 A Process for the Measurement of Distortion

The methodology to measure distortion was as follows: ENFOL masks were manufactured and patterned by Electron Beam Lithography (EBL) with arrays of marker features across their surface area. The positions of the mask markers were then measured and the masks used to pattern silicon substrates by use of ENFOL. Upon measuring the position of the markers on the patterned substrates the position data of the mask and substrate markers were analysed revealing the distortion. The key enabling apparatus for measurement was the Raith 150 EBL as it possesses a precise laser-interferometer stage, allowing accurate measurement over a large surface area (tolerance 2nm) and metrology software module allowing the automated measurement of features and their position.

Arrays of cross marker features 20μm high and arms 1μm thick were patterned in 40nm tungsten on 100μm thick borosilicate glass cover slips. Prior to etching the

tungsten the marker features were measured in the EBL using a command file generated using custom software. This was necessary as the built-in Raith software was unable to generate the commands needed for compensating for the drift of the electron beam. This involved the repeated measurement of assigned beam drift measurement markers and inserting the commands to measure them within the thousands of commands making up the position list. Software written in Java was coded for this purpose.

With measurements of both the mask and substrate marker arrays the corresponding data sets were analysed using two programs: the first written in Java and second in Matlab. The first program compensated for beam drift using linear interpolation with respect to the time the measurements were taken and removed the data from unpatterned marker features. The second program then corrected for the systematic errors of shift and rotation resulting from imperfect alignment during setup of the measurement scans.

The performance of these stages of correction is as follows: measurements of beam drift markers at a frequency of one beam drift measurement per ten normal measurements showed an interpolation error of less than 20nm per marker. This was adequate in the context of measuring distortion on the order of 1µm as was the case. The performance of the shift and rotation correction was tested with adding Gaussian noise (1µm three sigma) added to a measured data set of 176 markers. 1000 trials were conducted and the result was the shift and rotation demonstrated a factor of ten recovery from the noise. Addition of more markers would further increase this performance.

With development of data acquisition and processing methods complete, the performance of the measuring system overall was testing by multiple measurements of the same sample. This yielded the result that the measuring process is capable of measuring down to a three sigma tolerance of 60nm. This result demonstrated that the development of a measuring process for in-plane distortion had been successful.

7.2 Distortion results: Particulate Contamination

The testing of the measurement system however shed light on a problem that proves to be pervasive throughout these experiments. Measurements of the same mask across loadings of the EBL showed a large localised distortion of magnitude 0.81μm in the upper portion of the mask. This could only have been caused by topographical change in the mask and fits the effect of a particle trapped beneath the mask. In fact mechanical modelling of the system confirms this to be the case for a 30μm high particle.

The effect of particulate contamination could be seen across the distortion maps of ENFOL mask patterning with multiple localised distortions randomly distributed. The magnitude of the distortions was on the order of 1μm. Initial results suggested problems with the mask, and alterative poly-dimethyl-siloxane (PDMS) spacer mask mounts were experimented upon. These were not found to be of advantage, and with a return to the original mask mount the lowest distortion results were found to be 0.709μm three sigma in 'x' and 0.560 three sigma in 'y'.

A further observation supporting contamination was that of Newton's rings observed as soon as contact of an unpatterned test mask membrane was made with samples in the mask aligner. This ruled out thermal effects and outgassing as the topological change was instantaneous. With the random distribution of large distortion measured across the patterning of substrates, observation of Newton's rings, a large localised distortion across measurements of a single mask in the EBL, and 10-15 markers failing to pattern per sample, the case for particulate contamination being the cause of the observed distortion was strong.

This was proved beyond a doubt with the use of Finite Element Method (FEM) mechanical modelling of the system giving distortions matching those observed experimentally. An axially symmetric model of the glass membrane was created with 10μm mesh to conforming against rigid bodies under atmospheric pressure, as in the real system. The result of a 40μm particle gave an in-plane distortion of 1.05μm with a spread of distortion (at least 10nm) over 3mm. This confirms that observed

experimentally with spreads over a few mm and distortions of magnitude 1μm. A particle 30μm high yielded a distortion of 0.85μm over less than 3mm matching the large distortion observed under testing of the measurement process. Further simulation of a particle 20μm high yielded a magnitude of 0.625μm distortion, matching the lowest distortion result found experimentally.

The aim of this thesis was to measure the in-plane distortion resulting from ENFOL patterning of silicon substrates through use of conformable masks. To this end a sophisticated measurement methodology was successfully implemented capable of 60nm precision. The true nature of the distortion of ENFOL mask to substrate was hampered by the effects of particulate contamination. However, the nature of the distortion was glimpsed in the results of the mechanical modelling with friction. This can be seen as residual stretch in the mask of the order of a few tens of nanometres. This stretch is a function of initial separation height, increasing by 10nm with each increases of 10μm in initial separation.

Carrying out this work in an environment with a lower degree of particulate contamination would serve to measure the distribution of this distortion experimentally, which, mechanical modelling suggests is radially symmetric and hence would be able to be corrected for.

7.3 Future Work

When a large distortion was measured across multiple scans of the same mask between loadings of the EBL, it was realised that the initial scan of the mask was unreliable and accuracy of in-plane distortion measurement only capable of high precision if the mask was unaffected by contamination under its initial measurement. A way of determining which masks had been affected by contamination would serve to restore reliability of measurements in environments affected by particulate contamination. Comparing the resulting measurement of the mask features against a distortion free pattern would reveal large distortions from particles, however there is the possibility the distortions could hide amongst the effect of beam drift – during

exposure. Beam drift can be compensated for during measurement but drift during exposure unmeasured is an unknown quantity.

Beam drift is of the order of 1μm per hour in magnitude, with exposure of the mask occurring over 40 minutes significant drift could occur and mask distortion caused by particles when comparing the mask markers to an ideal pattern. To this end, measuring the beam drift during exposure would serve to give knowledge upon where the mask markers will end up.

A single beam drift measurement marker would be needed and this would consist of a marker patterned on a separate sample placed nearby the mask to be patterned. In order to instruct the Raith to take frequent measurement of the marker, the exposure pattern would have to be divided up to be exposed piece wise to allow the insertion of measurement commands between exposures. Armed with the knowledge of the beam drift during exposure an algorithm would be coded to compute the likely location of beam drifted patterned markers. Further computation would analyse the data on a marker by marker basis to check it is within the computed location within tolerance.

Thus with this knowledge the detection of contamination at the measurement stage would be able to be realised, and thus reliable measurements of ENFOL masks able to be taken in an environment affected by contamination.

References

[1] I. M. Ross, "The invention of the transistor," *Proceedings of the IEEE*, vol. 86, pp. 7-28, 1998.

[2] "Julius Edgar Lilienfeld ", 2005 March 9th http://chem.ch.huji.ac.il/~eugeniik/history/lilienfeld.htm.

[3] F. Faggin, "The Intel 4004: A testimonial from Federico Faggin, its designer, on the first microprocessor's thirtieth birthday," 2005 March 8th http://www.intel4004.com/sgate.htm.

[4] R. M. Holt, "F14 “TomCat” MOS-LSI Microprocessor Chip Set," presented at Vintage Computer Festival, Santa Clara Convention Center, 1998 available at http://www.microcomputerhistory.com/.

[5] Wikipedia, "F14 CADC --- Wikipedia, the free encyclopedia," 2005 March 3rd. http://en.wikipedia.org/wiki/F14_CADC.

[6] Wikipedia, "Intel 4004 --- Wikipedia, the free encyclopedia," 2005 March 3rd. http://en.wikipedia.org/wiki/Intel_4004.

[7] S. J. McNab, *Evanescent near-field optical lithography : overcoming the diffraction limit : a thesis presented for the degree of Doctor of Philosophy in Electrical and Electronic Engineering at the University of Canterbury, Christchurch, New Zealand / Sharee J. McNab.* Christchurch New Zealand: University of Canterbury, 2001.

[8] Yager, "Nanotechnology: A Brief Overview," vol. 2004: Barrett Research Group, Department of Chemistry, McGill University, Montréal, 2004.

[9] http://www.smt.zeiss.com, "Carl Zeiss SMT ships EUV Optics for World’s First, 0.25 NA, Full Field EUV Alpha Tool by ASML," 2005.

[10] Y. Chen and A. Pepin, "Nanofabrication: Conventional and nonconventional methods," *Electrophoresis*, vol. 22, pp. 187-207, 2001.

[11] P. Naulleau, K. A. Goldberg, J. P. Cain, E. H. Anderson, K. R. Dean, P. Denham, B. Hoef, and K. H. Jackson, "Extreme ultraviolet lithography capabilities at the advanced light source using a 0.3-NA optic," *Ieee Journal of Quantum Electronics*, vol. 42, pp. 44-50, 2006.

[12] K. Suzuki, N. Hirayanagi, T. Fujiwara, A. Yamada, J. Ikeda, T. Yahiro, S. Kojima, J. Udagawa, H. Yamamoto, N. Katakura, M. Suzuki, T. Aoyama, H. Takekoshi, T. Umemoto, H. Shimizu, S. Fukui, S. Suzuki, T. Okino, Y. Ohkubo, T. Shimoda, T. Tanida, Y. Watanabe, Y. Kohama, K. Ohmori, F. Mori, S. Takemoto, T. Yoshioka, H. Hirose, K. Morita, K. Hada, S. Kawata, Y. Kakizaki, and T. Miura, "Full-field exposure performance of electron projection lithography tool," *Journal of Vacuum Science & Technology B*, vol. 22, pp. 2885-2890, 2004.

[13] Wikipedia, "Nanoimprint Lithography --- Wikipedia, the free encyclopedia," 2005 March 4th. http://en.wikipedia.org/wiki/Nanoimprint_lithography.

[14] M. D. Austin, H. X. Ge, W. Wu, M. T. Li, Z. N. Yu, D. Wasserman, S. A. Lyon, and S. Y. Chou, "Fabrication of 5 nm linewidth and 14 nm pitch features by nanoimprint lithography," *Applied Physics Letters*, vol. 84, pp. 5299-5301, 2004.

[15] M. Konijn, M. M. Alkaisi, and R. J. Blaikie, "Nanoimprint lithography of sub-100 nm 3D structures," *Microelectronic Engineering*, vol. 78-79, pp. 653-658, 2005.

[16] T. Ito, M. Ogino, T. Yamada, Y. Inao, T. Yamaguchi, N. Mizutani, and R. Kuroda, "Fabrication of sub-100 nm patterns using near-field mask lithography with ultra-thin resist process," *Journal of Photopolymer Science and Technology*, vol. 18, pp. 435-441, 2005.

[17] S. J. McNab and R. J. Blaikie, "Contrast in the evanescent near field of lambda/20 period gratings for photolithography," *Applied Optics*, vol. 39, pp. 20-25, 2000.

[18] B. Schechter, "Bright new world," in *New Scientist*, vol. 178, 2003, pp. 30.

[19] X. G. Luo and T. Ishihara, "Surface plasmon resonant interference nanolithography technique," *Applied Physics Letters*, vol. 84, pp. 4780-4782, 2004.

[20] M. Chown, "Tight fit - Sloshing electrons help light wriggle through the most minute gaps," in *New Scientist*, vol. 157, 1998, pp. 6.

[21] T. Ono and M. Esashi, "Subwavelength pattern transfer by near-field photolithography," *Japanese Journal of Applied Physics Part 1-Regular Papers Short Notes & Review Papers*, vol. 37, pp. 6745-6749, 1998.

[22] R. R. Kunz, M. Rothschild, and M. S. Yeung, "Large-area patterning of similar to 50 nm structures on flexible substrates using near-field 193 nm radiation," *Journal of Vacuum Science & Technology B*, vol. 21, pp. 78-81, 2003.

[23] H. I. Smith, "Low cost nanolithography with nanoaccuracy," *Physica E-Low-Dimensional Systems & Nanostructures*, vol. 11, pp. 104-109, 2001.

[24] J. G. Goodberlet, "Patterning 100 nm features using deep-ultraviolet contact photolithography," *Applied Physics Letters*, vol. 76, pp. 667-669, 2000.

[25] H. Schmid, H. Biebuyck, B. Michel, O. J. F. Martin, and N. B. Piller, "Light-coupling masks: An alternative, lensless approach to high-resolution optical contact lithography," *Journal of Vacuum Science & Technology B*, vol. 16, pp. 3422-3425, 1998.

[26] M. Paulus, H. Schmid, B. Michel, and O. J. F. Martin, "Contrast mechanisms in high-resolution contact lithography: A comparative study," *Microelectronic Engineering*, vol. 57-8, pp. 109-116, 2001.

[27] The EAM mask using near field interference has demonstrated sub-diffraction limited resolution - 50nm linewidths from 220nm deep UV. However interference lithography cannot produce truly arbitrary features at the nano scale

[28] M. Paulus, B. Michel, and O. J. F. Martin, "Near-field distribution in light-coupling masks for contact lithography," *Journal of Vacuum Science & Technology B*, vol. 17, pp. 3314-3317, 1999.

[29] R. J. Blaikie, M. M. Alkaisi, S. J. McNab, D. R. S. Cumming, R. Cheung, and D. G. Hasko, "Nanolithography using optical contact exposure in the evanescent near field," *Microelectronic Engineering*, vol. 46, pp. 85-88, 1999.

[30] T. Ito, T. Yamada, Y. Inao, T. Yamaguchi, N. Mizutani, and R. Kuroda, "Fabrication of half-pitch 32 nm resist patterns using near-field lithography with a-Si mask," *Applied Physics Letters*, vol. 89, 2006.

[31] J. G. Goodberlet and B. L. Dunn, "Deep-ultraviolet contact photolithography," *Microelectronic Engineering*, vol. 53, pp. 95-99, 2000.

[32] M. M. Alkaisi, R. J. Blaikie, and S. J. McNab, "Nanolithography in the evanescent near field," *Advanced Materials*, vol. 13, pp. 877-+, 2001.

[33] O. J. F. Martin, "Surface plasmon illumination scheme for contact lithography beyond the diffraction limit," *Microelectronic Engineering*, vol. 67-8, pp. 24-30, 2003.

[34] D. O. S. Melville and R. J. Blaikie, "Super-resolution imaging through a planar silver layer," *Optics Express*, vol. 13, pp. 2127-2134, 2005.

[35] D. O. S. Melville, R. J. Blaikie, and C. R. Wolf, "Submicron imaging with a planar silver lens," *Applied Physics Letters*, vol. 84, pp. 4403-4405, 2004.

[36] D. O. S. Melville, R. J. Blaikie, and M. M. Alkaisi, "A comparison of near-field lithography and planar lens lithography," *Current Applied Physics*, vol. 6, pp. 415-418, 2006.

[37] as the electrons have enough thermal energy to escape their atoms

[38] "Electron Microscopy." Lincoln, Nebraska: Center for Materials Research and Analysis University of Nebraska-Lincoln, 2004.

[39] The electron beam results in other types of interaction such as back scattered electrons, which are yield dependant on atomic mass resulting in contrast from elemental composition. However the Raith 150 EBL is not equipped with detectors for such.

[40] *Ultra high resolution electron beam lithography and metrology tool Raith 150*: Raith Product Information Brochure, 2006.

[41] J. G. Goodberlet, J. T. Hastings, and H. I. Smith, "Performance of the Raith 150 electron-beam lithography system," *Journal of Vacuum Science & Technology B*, vol. 19, pp. 2499-2503, 2001.

[42] D. O. S. Melville, *Planar Lensing Lithography: Enhancing the Optical Near Field. A thesis presented for the degree of Doctor of Philosophy in Electrical and Electronic Engineering at the University of Canterbury, Christchurch, New Zealand*. Christchurch New Zealand: University of Canterbury, 2006.

[43] K. Salaita, S. W. Lee, X. F. Wang, L. Huang, T. M. Dellinger, C. Liu, and C. A. Mirkin, "Sub-100 nm, centimeter-scale, parallel dip-pen nanolithography," *Small*, vol. 1, pp. 940-945, 2005.

[44] Geometric data stream II (GDSII) is an industry standard 2D graphical data format to describe lithographic mask patterns.

[45] C. Reyes-Betanzo, S. A. Moshkalyov, A. C. Ramos, J. A. Diniz, and J. W. Swart, "Study of conditions for anisotropic plasma etching of tungsten and tungsten nitride using SF6/Ar gas mixtures," *Journal of the Electrochemical Society*, vol. 149, pp. G179-G183, 2002.

[46] Talk with Dr. Ali Azam of Canesis Network, September 2004

[47] The algorithm can be set to either return the midpoint of a feature or the width calculated from the edges.

[48] K. Mohamed, "Resist Deformation in Nanoimprint Lithography," in *Electrical and Computer Engineering*. Christchurch: University of Canterbury, 2005, pp. 116.

[49] M. Kaltenbacher, *Numerical Simulation of Mechatronic Sensors and Actuators*, 1 ed: Springer, 2004.

[50] C. Majidi, R. E. Groff, Y. Maeno, B. Schubert, S. Baek, B. Bush, R. Maboudian, N. Gravish, M. Wilkinson, K. Autumn, and R. S. Fearing, "High friction from a stiff polymer using microfiber arrays," *Physical Review Letters*, vol. 97, pp. -, 2006.

[51] "ABAQUS Analysis User's Manual," ABAQUS, Inc, 2003.

[52] Simulations without particles failed upon addition of friction. Therefore the results in this figure were obtained for a frictionless model.

Appendix A

CreatePls and ProcessPlsJava referenced in this thesis refer to programs written in Java to respectively generate measurement position lists in the Raith 150 and process acquired data. When *ProcessPlsJava* was coded it was built upon *CreatePls* as the later program had several classes that could be reused making coding of *ProcessPlsJava* expedient. Thus *CreatePls* actually refers to the usage of *ProcessPlsJava* in generation mode.

The source code for the *ProcessPlsJava* program in its entirety with all sub classes, is of length which would consume ~50 pages prohibiting its inclusion here. However the *ProcessPlsJava* class by itself which is the main program referencing other classes (routines) is included later in this appendix to outline the functionality and order of processing in the program.

A brief description of usage follows:
Position lists must be generated first for measurement of marker positions. The program accepts input parameters such as the resolution and dimension of measurement scans, beam drift measurement frequency, and coordinates of the marker arrays. These parameters are specified in an input parameters file, an example of which is provided on the next page of this appendix.

The program is then commanded to read in the input parameters file and generate a position list ready for use in the EBL. Once data has been acquired the position list and its associated input parameter file is needed along with its mask/sample counterparts and all four files are read in to *ProcessPlsJava* for processing.

Further detail can be found in the included example parameters file and *ProcessPlsJava* source code provided over the next few pages in this appendix; for those interested, the Java source code of *ProcessPlsJava* in its entirety will be available on a CD from my supervisor, Professor Richard Blaikie.

Example Measurement Parameters File
This file is used to generate Raith 150 position lists, producing a list of commands to measure arrays of patterned markers with beam drift compensation measurements. It is also later used to discern data contained in such lists for data processing.

In order to scan samples patterned by masks - which are laterally inverted multiple the following parameters in 'u' by -1

element_spacingU
pattern_coordU (all appearances of - one for every marker array coordinate)

Note the only lines that concern the program in these files are those with colons in them, any lines without colon characters will be ignored.

Specify the offset of the scan along the cross marker arm w.r.t. to point driven to by the EBL
scan_offset: 0.01

Specify the dimension in microns of the scans in 'u' and 'v'.
The main dimension is along the length of the scan and
the side dimension determines the breadth of the scan (used
for averaging out line edge roughness)
scan_dimMainU: 9.0
scan_dimSideU: 5.0
scan_dimMainV: 9.0
scan_dimSideV: 5.0

Resolution of scans in 'u' and 'v' for main and side dimensions
scan_resMainU: 2048
scan_resSideU: 16
scan_resMainV: 2048
scan_resSideV: 16

Switch to copy the scan parameters in 'u' to 'v'
UVScansDimResSame: false

Specify the spacing and number of elements arrays of marker
features consist of.
element_spacingU: 0.6
element_spacingV: 0.6
element_numberinU: 3
element_numberinV: 3

Specify the location of the 2 markers used for beam drift measurement
beam_drift_cross1coordU: 0
beam_drift_cross1coordV: 0
beam_drift_cross2coordU: 0.6
beam_drift_cross2coordV: 0

Order a number of beam drift measurements before measuring

the marker arrays to show the state of the beam drift prior to
measurement in greater detail. Alternatively this can be used to
set up scans that just measure beam drift alone
beam_drift_init_measuremnts: 30

Specify the frequency at which beam drift measurements are taken
with respect to measurement of markers in the arrays.
beam_drift_frequency: 10

Specify coordinates of marker arrays (bottom left hand corner of)
Center
1
pattern_coordU: 0
pattern_coordV: 0
2
pattern_coordU: 3.7
pattern_coordV: 0
3
pattern_coordU: 7.4
pattern_coordV: 0
4
pattern_coordU: 11.1
pattern_coordV: 0
5
pattern_coordU: 0
pattern_coordV: 3.7
6
pattern_coordU: 3.7
pattern_coordV: 3.7
7
pattern_coordU: 7.4
pattern_coordV: 3.7
8
pattern_coordU: 11.1
pattern_coordV: 3.7
9
pattern_coordU: 0
pattern_coordV: 7.4
10
pattern_coordU: 3.7
pattern_coordV: 7.4
11
pattern_coordU: 7.4
pattern_coordV: 7.4
12
pattern_coordU: 11.1
pattern_coordV: 7.4
13
pattern_coordU: 0
pattern_coordV: 11.1
14
pattern_coordU: 3.7

pattern_coordV: 11.1
15
pattern_coordU: 7.4
pattern_coordV: 11.1
16
pattern_coordU: 11.1
pattern_coordV: 11.1
17
pattern_coordU: -1
pattern_coordV: 6.0
18
pattern_coordU: 13
pattern_coordV: 6.0
19
pattern_coordU: 6.0
pattern_coordV: 13
20
pattern_coordU: 6.0
pattern_coordV: -1

/***

ProcessPlsJava

for metrological use of the Raith 150 Electron Beam Lithography System

Written by: Alan James Wright
Created: 25 Jan 2005
Last updated: 10 March 2007

See below for program description followed by usage

****
Program description:

Generation of measurement command files:
This program generates .pls files to command the Raith to measure arrays
of marker patterns with beam drift compensation measurements.
The arrays are scanned according to coordinates specified in an
input parameter file. The input file also specifies the frequency
of beam drift measurements and the location of markers to be used
for measuring beam drift.
(see the bottom of this source file for an example input parameter file)

Processing of measurement data files:
The second part of this program is the processing of the resulting
measurements made and saved to the generated position list. The position
list for both the mask and sample being compared are input into
this program along with their original input parameter files.
The data is compensated for drift of the electron beam through
linear interpolation of the beam drift measurements. Files are output
detailing the beam drift calculations for the mask and sample data sets
in the form of:

{mask input parameter file name}_{sample input parameter file name}_BD_calculation_Mask.txt

and one for the sample beam drift calculations.
The program then removes data from unpatterned markers in one data
set and its counterpart in the other data set. This prevents attempting
to measure distortion between two markers where the measurements for
one are missing. A single txt file is output detailing the markers that
were removed from both data sets:

{mask input parameter file name}_{sample input parameter file name}__MissingCrosses.txt

The last part of this program outputs the beam drift compensated
and unpatterned marker removed data and outputs it to a file
that can be read by Matlab where ProcessPlsMatlab.m is used to
apply shift and rotation correction to the data and thus complete
processing of the data. Maps of distortion and statistics are then
readily produced and gathered.

****
Usage:
This program is executed via command line interface on machines
installed with a java 2 runtime environment

For Generation:
java ProcessPls InputParameterFileName GENERATE

For Processing:
java ProcessPls MaskInputParameterFileName MaskPositionList.pls
SampleInputParameterFileName SamplePositionList.pls

Note: the later is entered a single command line instruction

****
Notes

The original input parameter files are used in the processing of
position lists generated from their parameters as this gives the
program information on how data is organised in the position list
without having to work backwards and determine which measurements
are beam drift and thus establish the frequency of beam drift
measurement and thus how to apply the interpolation.

```
*/
public class ProcessPlsJava
{
		public static void main(String args[])
		{

			//flags_______________________________________________________
			boolean debug = false;
			boolean GenerationMode = false;

			//Final vars_______________________________________________________
			final int MAX_LINES = 20000;

			/*external class instantiations______________________________*/
			check_file checkFile = new check_file();
			read_file readFile = new read_file();
			write_file writeFile = new write_file();
			obtain_parameters obtainParameters = new obtain_parameters();
			getMeasValue getMeasrValue = new getMeasValue();
			gen_pls_lines genPlsLines = new gen_pls_lines();
			genMatlabMFiles genMatlabMFiles = new genMatlabMFiles();

			//for mask
			m_parameters mParam = new m_parameters();//the class is also for generation of pls files
			mParam.init(debug);
			m_values mValues = new m_values();
			mValues.init(debug);
			obtain_pls_measurements obtain_pls_measurements  = new obtain_pls_measurements();
			crossCentres crossCentres = new crossCentres();
			beamDrift beamDrift = new beamDrift();

			//for sample
			m_parameters mParam_S = new m_parameters();//the class is also for generation of pls files
			mParam_S.init(debug);
			m_values_S SValues = new m_values_S();
			SValues.init(debug);
			obtain_pls_measurements_S obtain_pls_measurements_S  = new obtain_pls_measurements_S();
			crossCentres_S crossCentres_S = new crossCentres_S();
			beamDrift_S beamDrift_S = new beamDrift_S();


			/*Operations_____________________________________________________*/
			 /*Echo Program Name*/
			System.out.println("\nMetrology Position List Data Generator/Processor\n" +
			"Written by: Alan Wright\n\n" +
			"For position list generation\n" +
			"Usage: java ProcessPls InputParameterFileName GENERATE\n" +
			"\nFor processing scanned mask and sample position lists\n" +
			"Usage: java ProcessPls MaskInputParameterFileName MaskPositionList.pls\n" +
			"SampleInputParameterFileName SamplePositionList.pls\n" );

			//check there are at least 2 cmd line args
			if(args.length < 2)
			{
				System.out.println("\n Error: this program requires at least two command\n" +
				"line arguments for Generation mode. See Usage above");
				System.exit(1);
			}

			//Check second command line argument is GENERATE if so place program in pls generation mode
			String Arg2 = args[1];
			if(Arg2.equalsIgnoreCase("GENERATE"))
			{
				GenerationMode = true;
			}

			if(!GenerationMode)
			{
				//check there are at least 4 cmd line args
				if(args.length < 4)
				{
					System.out.println("\n Error: this program in processing mode requires " +
						"\n at least four command line arguments:\n two input parameter files " +
						"and two for the sample and mask position list files.\n See Usage above.\n " +
						"This program will now terminate");
					System.exit(1);
				}
			}

			 /*check input parameter file */
			String input_param_file[] = new String[] {" "};
```

```
input_param_file[0] = args[0];//copy first cmd line arg

if(checkFile.file_check_ok(input_param_file,debug)== false)
{
		System.out.println("!! Input parameter file error");
		System.exit(1);
}

/*Obtain the number of lines in the first block of the position list
this value will then be used to skip straight to the data of the position list
being processed*/
 /*check file for first block of position list*/
String first_block_file[] = new String[] {" "};
first_block_file[0] = "first_block"; //first block of position list, filename
if(checkFile.file_check_ok(first_block_file,debug)== false)
{
		System.out.println("!! first block file error");
		System.exit(1);
}

 /*read in first block file*/
String output_file_lines[] = new String[MAX_LINES]; //read the file to this array of strings
int num_lines_first_block = 0;
try
{
		num_lines_first_block = readFile.read_in_file(first_block_file,output_file_lines,debug);
		if(debug){System.out.println("First block number of lines is: " + num_lines_first_block);}
}
catch(Exception e){e.printStackTrace(System.err);}


 /*debugging*/
if(debug)
{
		System.out.println(output_file_lines[1]);
		System.out.println("Just before write file in main, "
				+ "num_lines_first_block = " + num_lines_first_block);
}

/* The INPUT measurement parameters file must be the same file that was used
to create the position list file originally. This ensures the correct parameters
are used, and the bonus that heaps of code gets reused! */
 /*read in the measurement parameters file*/
String input_file_lines[] = new String[MAX_LINES];
int num_input_file_lines = 0;
try
{
		num_input_file_lines = readFile.read_in_file(input_param_file,input_file_lines,debug);
}
catch(Exception e){e.printStackTrace(System.err);}

 /*obtain measurement parameters to program variables.*/
if(obtainParameters.measurements(mParam,input_file_lines,num_input_file_lines,debug)== false)
{
		System.out.println("!! error obtaining measurement parameters");
		System.exit(1);
}

//POSITIONLIST GENERATION ONLY: START
if(GenerationMode)
{
		 /*generate the positionlist lines for measuring*/
		int num_plslines = 0;
		num_plslines = genPlsLines.generate(mParam,output_file_lines,num_lines_first_block,false);
		if(num_plslines < 0)
		{
				System.out.println("!! error generating position list");
				//System.exit(1);
		}

		 /*write out the first_block and the generated lines to the position list file*/
		String output_pls_filename = input_param_file[0] + "_unscanned.pls";
		try
		{
				writeFile.write_to_file(output_pls_filename,output_file_lines,num_lines_first_block
				+ num_plslines,debug);
		}
		catch(Exception e){e.printStackTrace(System.err);}

		System.out.println("\nPosition list generation complete. \n" +
		"Output file is \"" + output_pls_filename + "\" ");
```

```
}

//POSITIONLIST GENERATION ONLY: FINISH


//PROCESSING ONLY: START

else
{
//Begin Processing Mask Data

		/*check input position list file */
		String input_pls_file[] = new String[] {" "};
		input_pls_file[0] = args[1];//copy second cmd line arg
		if(checkFile.file_check_ok(input_pls_file,debug)== false)
		{
			System.out.println("!! Input position list file error");
			System.exit(1);
		}

		/*From here I will read in the position list file - and remove the first block
		leaving only the data*/
		String input_pls_lines[] = new String[MAX_LINES];
		int num_input_pls_lines = 0;
		try
		{
			num_input_pls_lines = readFile.read_in_file(input_pls_file,input_pls_lines,debug);
		}
		catch(Exception e){e.printStackTrace(System.err);}

		/*remove first block (header information) by copying only data section to the
		pls_lines array*/
		String pls_lines[] = new String[MAX_LINES];
		System.arraycopy(input_pls_lines,num_lines_first_block,
					pls_lines,0,num_input_pls_lines - num_lines_first_block);
					/* Use of System.arraycopy() :(Object src,  int  srcPos,
										Object dest, int destPos,
						int length);*/

		 /*debugging*/
		if(debug)
		{
			for(int loop = 0; loop < (num_input_pls_lines - num_lines_first_block); loop++)
			{
				System.out.println(pls_lines[loop]);
			}
			System.out.println("Just before obtaining measurement values of pls file in main, "
				+ "num_lines_first_block = " + num_lines_first_block);
		}


		/*Calculate the correct number of lines for the obtain_pls_measurements class*/
		int num_data_lines = num_input_pls_lines - num_lines_first_block;
		if(debug){System.out.println("Num data lines is: " + num_data_lines );}

		 /*Obtain the pls measurement values into variables within this program*/
		if(obtain_pls_measurements.measurements(mValues,getMeasrValue,pls_lines,num_data_lines,debug)== false)
		{
			System.out.println("!! error obtaining measurement values");
			System.exit(1);
		}

		/*Begin processing - There are a specified (within input parameter file) number of initial
		beam drift measurements before beam drift compensation is to begin as cross measurements are
		taken. Identifying the id number of the first cross measurement will determine which beam
		drift measurements to use. - - don't need to do this the input parameter file has the number
		of initial beam drift measurements.*/

		/*Get cross centers*/
		if(crossCentres.calculate(mValues,debug)== false)
		{
			System.out.println("!! error obtaining cross centres");
			System.exit(1);
		}

		/*Compensate for beam drift - deal with any missing beam drift crosses first
		record these calculations for inspection - file output*/
		String BDcomp_lines[] = new String[MAX_LINES];
		if(beamDrift.compensate(mValues,mParam,BDcomp_lines,debug)== false)
		{
			System.out.println("!! error computing beam drift compensation");
```

```
		System.exit(1);
	}



//Begin Processing Sample Data

	 /*check input parameter file */
	String input_param_file_S[] = new String[] {" "};
	input_param_file_S[0] = args[2];//copy third cmd line arg

	if(checkFile.file_check_ok(input_param_file_S,debug)== false)
	{
		System.out.println("!! Sample input parameter file error");
		System.exit(1);
	}

	/*check input position list file */
	String input_pls_file_S[] = new String[] {" "};
	input_pls_file_S[0] = args[3];//copy fourth cmd line arg
	if(checkFile.file_check_ok(input_pls_file_S,debug)== false)
	{
		System.out.println("!! Sample Input position list file error");
		System.exit(1);
	}

	/* The INPUT measurement parameters file must be the same file that was used
	to create the position list file originally. This ensures the correct parameters
	are used, and the bonus that heaps of code gets reused! */
	 /*read in the measurement parameters file*/
	String input_file_lines_S[] = new String[MAX_LINES];
	int num_input_file_lines_S = 0;
	try
	{
		num_input_file_lines_S = readFile.read_in_file(input_param_file_S,input_file_lines_S,debug);
	}
	catch(Exception e){e.printStackTrace(System.err);}

	 /*obtain measurement parameters to program variables.*/
	if(obtainParameters.measurements(mParam_S,input_file_lines_S,num_input_file_lines_S,debug)== false)
	{
		System.out.println("!! error obtaining measurement parameters");
		System.exit(1);
	}

	/*From here I will read in the position list file - and remove the first block
	leaving only the data*/
	String input_pls_lines_S[] = new String[MAX_LINES];
	int num_input_pls_lines_S = 0;
	try
	{
		num_input_pls_lines_S = readFile.read_in_file(input_pls_file_S,input_pls_lines_S,debug);
	}
	catch(Exception e){e.printStackTrace(System.err);}

	/*remove first block (header information) by copying only data section to the
	pls_lines array*/
	String pls_lines_S[] = new String[MAX_LINES];
	System.arraycopy(input_pls_lines_S,num_lines_first_block,
			pls_lines_S,0,num_input_pls_lines_S - num_lines_first_block);
			/* Use of System.arraycopy() :(Object src,  int  srcPos,
							Object dest, int destPos,
				int length);*/

	 /*debugging*/
	if(debug)
	{
		for(int loop = 0; loop < (num_input_pls_lines_S - num_lines_first_block); loop++)
		{
			System.out.println(pls_lines_S[loop]);
		}
		System.out.println("Just before obtaining measurement values of sampe pls file in main, "
			+ "num_lines_first_block = " + num_lines_first_block);
	}

	/*Calculate the correct number of lines for the obtain_pls_measurements class*/
	int num_data_lines_S = num_input_pls_lines_S - num_lines_first_block;
	if(debug){System.out.println("Sample Num data lines is: " + num_data_lines_S );}

	 /*Obtain the pls measurement values into variables within this program*/
```

```
if(obtain_pls_measurements_S.measurements(SValues,getMeasrValue,pls_lines_S,num_data_lines_S,debug)== false)
                    {
                              System.out.println("!! error obtaining sample measurement values");
                              System.exit(1);
                    }


                    /*Begin processing - There are a specified (within input parameter file) number of initial
                    beam drift measurements before beam drift compensation is to begin as cross measurements are
                    taken. Identifying the id number of the first cross measurement will determine which beam
                    drift measurements to use. - - don't need to do this the input parameter file has the number
                    of initial beam drift measurements.*/

                    /*Get cross centers*/
                    if(crossCentres_S.calculate(SValues,debug)== false)
                    {
                              System.out.println("!! error obtaining sample cross centres");
                              System.exit(1);
                    }

                    /*Compensate for beam drift - deal with any missing beam drift crosses first
                    record these calculations for inspection - file output*/
                    String BDcomp_lines_S[] = new String[MAX_LINES];
                    if(beamDrift_S.compensate(SValues,mParam_S,BDcomp_lines_S,debug)== false)
                    {
                              System.out.println("!! error computing sample beam drift compensation");
                              System.exit(1);
                    }

          //Combined Mask and Sample Processing

                    /*Deal with missing crosses*/
                    String MC_lines[] = new String[MAX_LINES];
                    if(missingCrosses.share(mValues,SValues,MC_lines,debug)== false)
                    {
                              System.out.println("!! error sharing missing crosses");
                              System.exit(1);
                    }

                    /*Sample - write out the beam drift calculations for inspection*/
                    if(debug){System.out.println("LINE COUNT BD FILE IS: " + SValues.BDFileLineCnt);}
                    String BDFileName_S = input_param_file[0] + "_" +
                    input_param_file_S[0] + "_BDcalculation_Sample.txt";
                    try
                    {
                              writeFile.write_to_file(BDFileName_S,BDcomp_lines_S,SValues.BDFileLineCnt,debug);
                    }
                    catch(Exception e){e.printStackTrace(System.err);}

                    /*Mask - write out the beam drift calculations for inspection*/
                    if(debug){System.out.println("LINE COUNT BD FILE IS: " + mValues.BDFileLineCnt);}
                    String BDFileName = input_param_file[0] + "_" +
                    input_param_file_S[0] + "_BDcalculation_Mask.txt";
                    try
                    {
                              writeFile.write_to_file(BDFileName,BDcomp_lines,mValues.BDFileLineCnt,debug);
                    }
                    catch(Exception e){e.printStackTrace(System.err);}

                    /*write out the missing cross share for inspection*/
                    if(debug){System.out.println("LINE COUNT MC FILE IS: " + mValues.MCFileLineCnt);}
                    String MCFileName = input_param_file[0] + "_" +
                    input_param_file_S[0] + "_MissingCrosses.txt";
                    try
                    {
                              writeFile.write_to_file(MCFileName,MC_lines,mValues.MCFileLineCnt,debug);
                    }
                    catch(Exception e){e.printStackTrace(System.err);}



                    /*Generation of matlab code files (m files) */
                    String GenM_lines[] = new String[MAX_LINES];
                    if(genMatlabMFiles.engage(mValues,SValues,GenM_lines,debug)== false)
                    {
                              System.out.println("!! error generating Matlab m files");
                              System.exit(1);
                    }

                    /*write out the missing cross share for inspection*/
```

```
                        if(debug){System.out.println("LINE COUNT GenM FILE IS: " + mValues.GenMFileLineCnt);}
                        String GenMFileName = "MF" + input_param_file[0] + "_" +
                        input_param_file_S[0] + ".m";
                        try
                        {
                                writeFile.write_to_file(GenMFileName,GenM_lines,mValues.GenMFileLineCnt,debug);
                        }
                        catch(Exception e){e.printStackTrace(System.err);}


                }//end of else (non generation mode)



        }
}
/*

Example Measurement Parameters File

(included here in the actual source code is an example parameters file
 It was removed from this location and provided before the source code,
 to be in line with the operation description provided at the start of this appendix)

*/
```

Appendix B

ProcessPlsMatlab.m

This Matlab program is the second of two parts in the analysis of data gathered of the mask and sample data sets. The previous program *ProcessPlsJava* (Appendix A) corrected the data for beam drift and unpatterned markers. This program continues where *ProcessPlsJava* left off with correcting the data for shift and rotation systematic error and then proceeds to produce distortion maps, displacement distribution histograms and displacement statistics – yielding the results of the distortion measurement.

The use of the program requires the entering the name of the output file from *ProcessPlsJava* to pass the input data into the program.

```
clear; %variable reset
clc; %screen clear

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% ProcessPlsMatlab.m

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Shift, rotation correction for distortion mapping and stats
%
%Alan James Wright
%Last modified Mar 18 2007
%
%This program is used to correct for shift and rotation systematic errors
%between mask and substrate data sets. Shift correction is carried out by
%finding a least squares fit of a data set with respect to the 'x' axis and
%then the 'y' axis. the intersection of these lines provides the centroid
%of the data set. Shift correction is achieved by remapping the centroids
%of the data sets to coincide.
%Rotation correction proceeds by iteratively (with increasing smaller
%amounts) rotating one data set about the other to find the angle resulting
%in the least error between the two sets.
%
%With Shift and Rotation correction performed this program outputs
%a distortion map, displacement distribution histograms for 'x' and 'y' and
%a statistical summary of 'x' and 'y' displacements
%
%Input file needed:
%This program requires the 'x' and 'y' data positions for the mask and
%substrate provided by ProcessPlsJava
%e.g. mask1_sample1.m
%variables used Y_ValsMask, Y_ValsSample for 'y' and viceversa for 'x'
%
%Note: this program laterally inverts the sample data so that it matches
%the features scanned on the mask. Disable the inversion line of code if
%analysing two scans of one sample i.e. to test the accuracy of measurement
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

PerformInversionOfSampleData = 1;

%Get Mask and Sample data... Data must be entered by running m file for
%particular mask and sample first.
%This will be the file output by ProcessPlsJava in the form of
% mask1_sample1.m
```

```
%Place data m file here:
060405maskB2_060506samA7.m


% due to the photolithographic pattern transfer the sample is a mirror
% image of the mask (about the y axis) therefore we do this below
if(PerformInversionOfSampleData == 1)
   X_ValsSample = (-1)*X_ValsSample;
end


%Change of angle (range) to search in order to correct for rotational error
%i.e. to search over 20 degrees enter 10 for  -10 to +10 degrees.
searchAngle = 20;

%Rotation correction iteration level. Precision of angle correction.
Precision = 20; %increment to 10^-20 degrees

%Print distortion map with scalebar? 1 : 0
PrintScaleBarFigure = 0;


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


onesMask = ones(length(X_ValsMask),1);

onesSample = ones(length(X_ValsMask),1);


% Set axis
DataAxis = [-10000 10000 -10000 10000];
% Set range of x values
x = [-10000:100:10000];


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Mask least squares fit w.r.t. 'x' axis
%
% Horizontal m*x + c = y, solve for m and c
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

A = [X_ValsMask, onesMask];
b = [Y_ValsMask];

termA = inv(A'*A);
termB = A'*b;

coeffs = termA*termB;
m = coeffs(1);
c = coeffs(2);

% Horizontal plot
[sx,sy] = size(x);
xaxis = zeros(sy);
yaxis = zeros(sy);
yMaskHoriz = m * x + c;

mMaskHoriz = m;
cMaskHoriz = c;

% Horizontal calculate error
xcoord = X_ValsMask;
ycoord = m * xcoord + c;
error = sum((ycoord - Y_ValsMask).^2);

%Output least squares for w.r.t 'x' for mask data set
figure(1);
subplot(3,3,1);
plot(X_ValsMask,Y_ValsMask,'ko');
axis(DataAxis)
hold on;
plot(x,xaxis,'g-');
plot(yaxis,x,'g-');
plot(x,yMaskHoriz,'r-');
```

```
title('Mask, Line of Least Squares w.r.t ''x'' axis');
    xlabel('Microns');
    ylabel('Microns');


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Mask least squares fit w.r.t. 'y' axis
%
% Vertical m*x + c = y, solve for m and c
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

A = [Y_ValsMask, onesMask];
b = [X_ValsMask];

termA = inv(A'*A);
termB = A'*b;

coeffs = termA*termB;
m = 1 / coeffs(1);
c = -coeffs(2)/coeffs(1);

% Vertical plot
[sx,sy] = size(x);
xaxis = zeros(sy);
yaxis = zeros(sy);
yMaskVert = m * x + c;

mMaskVert = m;
cMaskVert = c;

% Vertical calculate error
ycoord = Y_ValsMask;
xcoord = (ycoord - c)/(m);
error = sum((xcoord - X_ValsMask).^2);



%Output least squares for w.r.t 'y' for sample data set
subplot(3,3,2);
plot(X_ValsMask,Y_ValsMask,'ko');
axis(DataAxis)
hold on;
plot(x,xaxis,'g-');
plot(yaxis,x,'g-');
plot(x,yMaskVert,'r-');
title('Mask, Line of Least Squares w.r.t ''y'' axis');
    xlabel('Microns');
    ylabel('Microns');

%Output intersection of least squares lines for sample data set (Centroid
%location)
subplot(3,3,3);
plot(X_ValsMask,Y_ValsMask,'ko');
axis(DataAxis)
hold on;
plot(x,xaxis,'g-');
plot(yaxis,x,'g-');
plot(x,yMaskHoriz,'r-');
plot(x,yMaskVert,'r-');
title('Centroid of Mask Data');
    xlabel('Microns');
    ylabel('Microns');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Sample least squares fit w.r.t. 'x' axis
%
% Horizontal m*x + c = y, solve for m and c
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

A = [X_ValsSample, onesSample];
b = [Y_ValsSample];

termA = inv(A'*A);
termB = A'*b;
```

```
coeffs = termA*termB;
m = coeffs(1);
c = coeffs(2);

% Horizontal plot
[sx,sy] = size(x);
xaxis = zeros(sy);
yaxis = zeros(sy);
ySampleHoriz = m * x + c;

mSampleHoriz = m;
cSampleHoriz = c;

% Horizontal calculate error
xcoord = X_ValsSample;
ycoord = m * xcoord + c;
error = sum((ycoord - Y_ValsSample).^2);



%Output least squares for w.r.t 'x' for sample data set
subplot(3,3,4);
plot(X_ValsSample,Y_ValsSample,'b*');
axis(DataAxis)
hold on;
plot(x,xaxis,'g-');
plot(yaxis,x,'g-');
plot(x,ySampleHoriz,'r-');
title('Patterned Silicon, Line of Least Squares w.r.t ''x'' axis');
    xlabel('Microns');
    ylabel('Microns');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Sample least squares fit w.r.t. 'y' axis
%
% Vertical m*x + c = y, solve for m and c
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

A = [Y_ValsSample, onesSample];
b = [X_ValsSample];

termA = inv(A'*A);
termB = A'*b;

coeffs = termA*termB;
m = 1 / coeffs(1);
c = -coeffs(2)/coeffs(1);

% Vertical plot
[sx,sy] = size(x);
xaxis = zeros(sy);
yaxis = zeros(sy);
ySampleVert = m * x + c;

mSampleVert = m;
cSampleVert = c;

% Vertical calculate error
ycoord = Y_ValsSample;
xcoord = (ycoord - c)/(m);
error = sum((xcoord - X_ValsSample).^2);



%Output least squares for w.r.t 'y' for sample data set
subplot(3,3,5);
plot(X_ValsSample,Y_ValsSample,'b*');
axis(DataAxis)
hold on;
plot(x,xaxis,'g-');
plot(yaxis,x,'g-');
plot(x,ySampleVert,'r-');
title('Patterned Silicon, Line of Least Squares w.r.t ''y'' axis');
```

```
    xlabel('Microns');
    ylabel('Microns');

%Output intersection of least squares lines for sample data set (Centroid
%location
subplot(3,3,6);
plot(X_ValsSample,Y_ValsSample,'b*');
axis(DataAxis)
hold on;
plot(x,xaxis,'g-');
plot(yaxis,x,'g-');
plot(x,ySampleHoriz,'r-');
plot(x,ySampleVert,'r-');
title('Centroid of Patterned Silicon Data');
    xlabel('Microns');
    ylabel('Microns');


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Find intercepts for Mask and Sample best fit lines. i.e. the Centres of
% Mass.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

interceptMaskX = (cMaskHoriz - cMaskVert) / (mMaskVert - mMaskHoriz);
interceptMaskY = mMaskHoriz*interceptMaskX + cMaskHoriz;

interceptSampleX = (cSampleHoriz - cSampleVert) / (mSampleVert - mSampleHoriz);
interceptSampleY = mSampleHoriz*interceptSampleX + cSampleHoriz;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Perform shift correction - move the Centres of Mass to the origin - thus
% the rotation centre and Centre of Mass are coincident.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

X_ValsMaskShifted = X_ValsMask - interceptMaskX;
Y_ValsMaskShifted = Y_ValsMask - interceptMaskY;

X_ValsSampleShifted = X_ValsSample - interceptSampleX;
Y_ValsSampleShifted = Y_ValsSample - interceptSampleY;


%Output shift correction subplot within figure 1
subplot(3,3,7);
plot(X_ValsSample,Y_ValsSample,'b*');
axis(DataAxis)
hold on;
plot(x,xaxis,'g-');
plot(yaxis,x,'g-');
plot(X_ValsMask,Y_ValsMask,'ko');
plot(X_ValsMaskShifted,Y_ValsMaskShifted,'mo');
plot(X_ValsSampleShifted,Y_ValsSampleShifted,'r*');
title('Shift Correction');
    xlabel('Microns');
    ylabel('Microns');


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Perform angle correction
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Rotate sample in small steps
% initial range to search over i.e. 10 degrees - from -5 to +5 degrees
% in steps of 1 degree, calculate total displacement for each angle and store
% the angle producing the least displacement.
% loop again with over a range with about the angle of minimum displacement
% but with the range decreased by a factor of ten, and similarly for the step increment
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

rangeL = -0.5*searchAngle;
rangeH = 0.5*searchAngle;

numberLoops = 0; %count total number of loops

for N = 1:Precision
```

```
    counter = 1; %number of loops for each precision number

    stepIncrement = 10^-N; %decrease step increment by a factor of ten each successive loop

    for stepSize = rangeL:stepIncrement:rangeH


        [ThetaSample,RhoSample] = cart2pol(X_ValsSampleShifted,Y_ValsSampleShifted);
        ThetaSampleRotated = ThetaSample + stepSize*(pi/180); %try adjusting by current angle in the stepsize within
the range
        [X_ValsSampleRotated,Y_ValsSampleRotated] = pol2cart(ThetaSampleRotated,RhoSample);

        %calulate the displacement magnitudes
        X_ValsDisplacement = X_ValsSampleRotated - X_ValsMaskShifted;
        Y_ValsDisplacement = Y_ValsSampleRotated - Y_ValsMaskShifted;

        [ThetaDisplacement,RhoDisplacement] = cart2pol(X_ValsDisplacement,Y_ValsDisplacement);

        %save the current angle being used to this array
        ErrorDisplacement(counter,1) = stepSize;
        %find the sum of the displacement magnitudes
        ErrorDisplacement(counter,2) = sum(RhoDisplacement);
        counter = counter + 1;

        numberLoops = numberLoops + 1;

    end

    %find the minimum sum(displacement rotation) from those calculated for
    %the angles tried above
    [A,index] = min(ErrorDisplacement(:,2));
    bestRotation = ErrorDisplacement(index,1); %store this to bestRotation angle.

    %change the range to further narrow in on the minimum by one tenth the
    %range smaller than before.
    rangeL = bestRotation - 10^-N;
    rangeH = bestRotation + 10^-N;
end

[A,index] = min(ErrorDisplacement(:,2));
bestRotation = ErrorDisplacement(index,1)

%Perform the angle correction from compensating angle found above
[ThetaSample,RhoSample] = cart2pol(X_ValsSampleShifted,Y_ValsSampleShifted);
ThetaSampleRotated = ThetaSample + bestRotation*(pi/180);
[X_ValsSampleRotated,Y_ValsSampleRotated] = pol2cart(ThetaSampleRotated,RhoSample);


%Output Rotation Correction subplot within figure 1
%figure;
 subplot(3,3,8);
 plot(X_ValsSampleShifted,Y_ValsSampleShifted,'r*');
 axis(DataAxis)
 hold on;
 plot(x,xaxis,'g-');
 plot(yaxis,x,'g-');
 plot(X_ValsMaskShifted,Y_ValsMaskShifted,'mo');
 plot(X_ValsSampleRotated,Y_ValsSampleRotated,'k+');
 title('Rotation Correction');
    xlabel('Microns');
    ylabel('Microns');
 hold off;


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Distortion map output section
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Calculate displacement vectors for distortion map output.
X_ValsDisplacement = X_ValsSampleRotated - X_ValsMaskShifted;
Y_ValsDisplacement = Y_ValsSampleRotated - Y_ValsMaskShifted;

%Output distortion map using matlabs velocity vector map (without scale bar)
figure(2)
quiver(X_ValsMaskShifted, Y_ValsMaskShifted, X_ValsDisplacement, Y_ValsDisplacement);
```

```
    title('Distortion Map of Mask to Silicon Substrate Pattern Transfer');
    xlabel('Microns');
    ylabel('Microns');

%The following figure is to print a 1um velocity vector to facilitate a
%scale (of distortion) bar for the distortion map
if(PrintScaleBarFigure == 1)

    %Coordinates of scale bar arrow
    X_ValsMaskSB = [X_ValsMaskShifted;2000];
    Y_ValsMaskSB = [Y_ValsMaskShifted;-7500];
    %magnitude of scale bar arrow
    X_ValsDispSB = [X_ValsDisplacement;1];
    Y_ValsDispSB = [Y_ValsDisplacement;0];

    figure(4)
    quiver(X_ValsMaskSB, Y_ValsMaskSB, X_ValsDispSB, Y_ValsDispSB);
        title('Distortion Map of Mask to Silicon Substrate Pattern Transfer');
        xlabel('Microns');
        ylabel('Microns');
        gtext('1um displacement')
end


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Displacement distribution histograms output section
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

figure(3)
%output 'x' displacement distribution histogram
subplot(2,1,1)
Hist(X_ValsDisplacement,40)
     title('Displacement in X')
     xlabel('Microns')
     ylabel('Frequency')
%output 'y' displacement distribution histogram
subplot(2,1,2)
Hist(Y_ValsDisplacement,40)
     title('Displacement in Y')
     xlabel('Microns')
     ylabel('Frequency')


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Statistics output section
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Output statistical summary for 'x' and 'y'
%mean, standard deviation (sigma), mean+3*sigma
X_Disp_Results = [mean(X_ValsDisplacement),std(X_ValsDisplacement),mean(X_ValsDisplacement) +
3*std(X_ValsDisplacement)]
Y_Disp_Results = [mean(Y_ValsDisplacement),std(Y_ValsDisplacement),mean(Y_ValsDisplacement) +
3*std(Y_ValsDisplacement)]

%Distortion absolute stats
AbsoluteResults = [mean(RhoDisplacement),std(RhoDisplacement),mean(RhoDisplacement) +
3*std(RhoDisplacement)]
```

Appendix C

ProcPMatlabSelfTest.m

This Matlab program is tests the precision of the shift and rotation correction as implemented in ProcessPlsMatlab.m (Appendix B). The data from the marker pattern of one mask was duplicated with gaussian noise added to one copy. The untouched copy was then analysed against the noisy copy. Ideally such processing should yield no shift or rotation correction as there has been no net translation or rotation of the marker pattern with the added noise. Inevitably some shift and rotation correction is applied in error which is saved to an array. The addition of noise and correction is repeated over 1000 trials.

```
clear; %variable reset
clc; %screen clear

%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% ProcPMatlabSelfTest.m

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Test of shift, rotation correction for distortion mapping and stats by
% noise trials
%
% Alan James Wright
% Last modified Mar 19 2007
%
% This program is to ascertain the performance of shift and rotation
% correction as implemented in ProcessPlsMatlab.m .
%
% This is performed by duplicating a data set of measurement marker
% coordinates and then adding gaussian noise to one data set with a
% given standard deviation.  Thus with no overall translation or rotation of
% noisy data set with respect to its healthy duplicate, idealy there should
% not be any rotation or shift correction applied to the data sets.
%
% 1000 trials were undertaken and the statistics gathered on any rotation
% and shift correction applied.
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Shift, rotation correction for distortion mapping and stats

%Change of angle (range) to search in order to correct for rotational error
%i.e. to search over 20 degrees enter 10 for  -10 to +10 degrees.
searchAngle = 20;

%Number of trials to conduct of adding noise to the mask data set to test
%the accuracy of the shift and rotation correction.
numberOfTrials = 1000;

%Load data mask scan data
maskDataForNoiseTest.m

onesMask = ones(length(X_ValsMask),1);

%For Loop to conduct trials of adding noise to the mask data set to test
%the accuracy of the shift and rotation correction.
for noiseTrial = 1:numberOfTrials
```

```
%____________________________________________

%Noisy bit
noiseX = 0.33*randn(length(X_ValsMask),1); %1* this will give me 1um standard dev. else 0.1* will give me 100nm
std dev.
noiseY = 0.33*randn(length(X_ValsMask),1);

%SAMPLE DATA
onesSample = ones(length(X_ValsMask),1);

X_ValsSample = X_ValsMask + noiseX;

Y_ValsSample = Y_ValsMask + noiseY;

% Set axis
DataAxis = [-10000 10000 -10000 10000];
% Set range of x values
x = [-10000:100:10000];


%%%%%%%%%%%%%%%%%%% End of Data %%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Mask least squares w.r.t. 'x' line
%
% Horizontal m*x + c = y, solve for m and c
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

A = [X_ValsMask, onesMask];
b = [Y_ValsMask];

termA = inv(A'*A);
termB = A'*b;

coeffs = termA*termB;
m = coeffs(1);
c = coeffs(2);

% Horizontal plot
[sx,sy] = size(x);
xaxis = zeros(sy);
yaxis = zeros(sy);
yMaskHoriz = m * x + c;

mMaskHoriz = m;
cMaskHoriz = c;

% Horizontal calculate error
xcoord = X_ValsMask;
ycoord = m * xcoord + c;
error = sum((ycoord - Y_ValsMask).^2);



%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Mask least squares w.r.t. 'y' line
%
% Vertical m*x + c = y, solve for m and c
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

A = [Y_ValsMask, onesMask];
b = [X_ValsMask];

termA = inv(A'*A);
termB = A'*b;

coeffs = termA*termB;
m = 1 / coeffs(1);
c = -coeffs(2)/coeffs(1);

% Vertical plot
[sx,sy] = size(x);
```

```
xaxis = zeros(sy);
yaxis = zeros(sy);
yMaskVert = m * x + c;

mMaskVert = m;
cMaskVert = c;

% Vertical calculate error
ycoord = Y_ValsMask;
xcoord = (ycoord - c)/(m);
error = sum((xcoord - X_ValsMask).^2);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Sample least squares w.r.t. 'x' line
%
% Horizontal m*x + c = y, solve for m and c
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

A = [X_ValsSample, onesSample];
b = [Y_ValsSample];

termA = inv(A'*A);
termB = A'*b;

coeffs = termA*termB;
m = coeffs(1);
c = coeffs(2);

% Horizontal plot
[sx,sy] = size(x);
xaxis = zeros(sy);
yaxis = zeros(sy);
ySampleHoriz = m * x + c;

mSampleHoriz = m;
cSampleHoriz = c;

% Horizontal calculate error
xcoord = X_ValsSample;
ycoord = m * xcoord + c;
error = sum((ycoord - Y_ValsSample).^2);


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Sample least squares w.r.t. 'y' line
%
% Vertical m*x + c = y, solve for m and c
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

A = [Y_ValsSample, onesSample];
b = [X_ValsSample];

termA = inv(A'*A);
termB = A'*b;

coeffs = termA*termB;
m = 1 / coeffs(1);
c = -coeffs(2)/coeffs(1);

% Vertical plot
[sx,sy] = size(x);
xaxis = zeros(sy);
yaxis = zeros(sy);
ySampleVert = m * x + c;

mSampleVert = m;
cSampleVert = c;

% Vertical calculate error
ycoord = Y_ValsSample;
xcoord = (ycoord - c)/(m);
error = sum((xcoord - X_ValsSample).^2);
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Find intercepts for Mask and Sample best fit lines. i.e. the Centres of
    % Mass.

    interceptMaskX = (cMaskHoriz - cMaskVert) / (mMaskVert - mMaskHoriz);
    interceptMaskY = mMaskHoriz*interceptMaskX + cMaskHoriz;

    interceptSampleX = (cSampleHoriz - cSampleVert) / (mSampleVert - mSampleHoriz);
    interceptSampleY = mSampleHoriz*interceptSampleX + cSampleHoriz;

    ShiftInX(noiseTrial) = interceptSampleX - interceptMaskX;
    ShiftInY(noiseTrial) = interceptSampleY - interceptMaskY;


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Perform shift correction - move the Centres of Mass to the origin - thus
    % the rotation centre and Centre of Mass are coincident.
    X_ValsMaskShifted = X_ValsMask - interceptMaskX;
    Y_ValsMaskShifted = Y_ValsMask - interceptMaskY;

    X_ValsSampleShifted = X_ValsSample - interceptSampleX;
    Y_ValsSampleShifted = Y_ValsSample - interceptSampleY;



%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Perform angle correction

    % Rotate sample in small steps
    % initial range to search over 10 degrees - from -5 to +5 degrees
    rangeL = -searchAngle;
    rangeH = searchAngle;

    Precision = 20; %increment to 10^-15 degrees
    numberLoops = 0; %count total number of loops

    for N = 1:Precision

        counter = 1; %number of loops for each precision number

        stepIncrement = 10^-N;

        for stepSize = rangeL:stepIncrement:rangeH


            [ThetaSample,RhoSample] = cart2pol(X_ValsSampleShifted,Y_ValsSampleShifted); %cart2pol
            ThetaSampleRotated = ThetaSample + stepSize*(pi/180); %try adjusting by current angle in the stepsize
within the range
            [X_ValsSampleRotated,Y_ValsSampleRotated] = pol2cart(ThetaSampleRotated,RhoSample);

            %calulate the displacement magnitudes
            X_ValsDisplacement = X_ValsSampleRotated - X_ValsMaskShifted;
            Y_ValsDisplacement = Y_ValsSampleRotated - Y_ValsMaskShifted;

            [ThetaDisplacement,RhoDisplacement] = cart2pol(X_ValsDisplacement,Y_ValsDisplacement);

            %save the current angle being used to this array
            ErrorDisplacement(counter,1) = stepSize;
            %find the sum of the displacement magnitudes
            ErrorDisplacement(counter,2) = sum(RhoDisplacement);
            counter = counter + 1;

            numberLoops = numberLoops + 1;

        end

        %find the minimum sum(displacement rotation) from those calculated for
        %the angles tried above
        [A,index] = min(ErrorDisplacement(:,2));
        bestRotation = ErrorDisplacement(index,1); %store this to bestRotation angle.
```

```
        %change the range to further narrow in on the minimum by one tenth the
        %range smaller than before.
        rangeL = bestRotation - 10^-N;
        rangeH = bestRotation + 10^-N;
    end

    [A,index] = min(ErrorDisplacement(:,2));
    bestRotation = ErrorDisplacement(index,1);

    bestRotationArray(noiseTrial) = bestRotation;

end

%output results%
results = [mean(ShiftInX), std(ShiftInX); mean(ShiftInY), std(ShiftInY); mean(bestRotationArray),
std(bestRotationArray)]
```

Appendix D

This appendix consists of an example of a Raith 150 position list file. The data stored in the position list is a comma delimited set of fields, which field is which and their order of display is given the [COLUMNS] section at the beginning of the file. Important measurement parameters include the 'u' and 'v' ('x' and 'y') coordinates of the scans in fields 7 & 8; threshold algorithm result (error or success) in field 10; whether or not the measurement is a beam drift compensation measurement or general marker in field 12; scan dimension and resolution (fields 16-19) and time stamp in the third to last field. Note: the command 'STAY' in field 13 is used to instruct the EBL to remain stationary and scan the cross marker arms within its write field once it has driven to the approximate centre of a cross marker.

```
[HEADER]
FORMAT=IXYZRTUVWATC,Options,65,Type,17,Size-U,11,Size-V,11,Points-U,11,Points-
V,11,Dir,5,Avg,5,Pos1,17,Pos2,17,Pos3,17,Link,9,File,257,Layer,65,Area,49,DoseFactor,11,Dwelltime,11,Stepsize,11,SplDwell,11,SplStep,11,DotDwell,11,SplDot,5,Time,17,Timestamp,17,Method,17,Dot,16
WAFERLAYOUT=DEFAULT.WLO
LotID=
WaferID=
Slot=
MinimizeWin=FALSE

[COLUMNS]
No.=WIDTH:5,!VISIBLE
ID=WIDTH:5,VISIBLE
X=WIDTH:10,!VISIBLE,DIM:mm,SHOWDIM
Y=WIDTH:10,!VISIBLE,DIM:mm,SHOWDIM
Z=WIDTH:10,!VISIBLE,DIM:mm,SHOWDIM
R=WIDTH:10,!VISIBLE,DIM:deg,SHOWDIM
T=WIDTH:10,!VISIBLE,DIM:deg,SHOWDIM
U=WIDTH:10,VISIBLE,DIM:mm,SHOWDIM
V=WIDTH:10,VISIBLE,DIM:mm,SHOWDIM
W=WIDTH:10,!VISIBLE,DIM:mm,SHOWDIM
Attribute=WIDTH:11,VISIBLE,DEFAULT:A
Template=WIDTH:11,VISIBLE,DEFAULT:UV
Comment=WIDTH:20,VISIBLE
Options=WIDTH:33,VISIBLE
Type=WIDTH:17,VISIBLE
Size-U=WIDTH:11,!VISIBLE,DIM:µm,SHOWDIM
Size-V=WIDTH:11,!VISIBLE,DIM:µm,SHOWDIM
Points-U=WIDTH:11,!VISIBLE
Points-V=WIDTH:11,!VISIBLE
Dir=WIDTH:3,!VISIBLE
Avg=WIDTH:5,!VISIBLE
Pos1=WIDTH:17,VISIBLE,DIM:µm,SHOWDIM
Pos2=WIDTH:17,VISIBLE,DIM:µm,SHOWDIM
Pos3=WIDTH:17,VISIBLE,DIM:µm,SHOWDIM
Link=WIDTH:5,VISIBLE
File=WIDTH:32,VISIBLE
Layer=WIDTH:16,VISIBLE
Area=WIDTH:49,!VISIBLE
DoseFactor=WIDTH:11,VISIBLE
Dwelltime=WIDTH:11,!VISIBLE,DIM:ms,SHOWDIM
Stepsize=WIDTH:11,!VISIBLE,DIM:µm,SHOWDIM
SplDwell=WIDTH:11,!VISIBLE,DIM:ms,SHOWDIM
SplStep=WIDTH:11,!VISIBLE,DIM:µm,SHOWDIM
DotDwell=WIDTH:11,!VISIBLE,DIM:ms,SHOWDIM
SplDot=WIDTH:2,!VISIBLE
Time=WIDTH:17,VISIBLE
Timestamp=WIDTH:17,!VISIBLE
Method=WIDTH:17,!VISIBLE
Dot=WIDTH:4,!VISIBLE
[DATA]
0,0.000000,0.000000,0.000000,0.000000,0.000000,0.700000,-0.700000,6.000000,A,UV,"Drive Command",,,,,,,,,,,,,,,,1.000,,,,,,,,,,
1,0.000000,0.000000,0.000000,0.000000,0.000000,0.700000,-0.685000,6.000000,LS,UV,"BD1 measure",STAY,EBEAMSCAN,8.0,5.0,2048.0,16.0,U,1,699.170,701.308,700.237,,%UserRoot%LINE\EBEA1647.LSC,,,1.000,,,,,,,1119162952,06-19-05 19:35:5,Thr:,
2,0.000000,0.000000,0.000000,0.000000,0.000000,0.700000,-0.715000,6.000000,LS,UV,"BD1 measure",STAY,EBEAMSCAN,8.0,5.0,2048.0,16.0,U,1,698.748,700.830,699.787,,%UserRoot%LINE\EBEA1648.LSC,,,1.000,,,,,,,1119162954,06-19-05 19:35:5,Thr:,
3,0.000000,0.000000,0.000000,0.000000,0.000000,0.685000,-0.700000,6.000000,LS,UV,"BD1 measure",STAY,EBEAMSCAN,5.0,8.0,16.0,2048.0,V,1,-701.027,-699.041,-700.036,,%UserRoot%LINE\EBEA1649.LSC,,,1.000,,,,,,,1119162955,06-19-05 19:35:5,Thr:,
4,0.000000,0.000000,0.000000,0.000000,0.000000,0.715000,-0.700000,6.000000,LS,UV,"BD1 measure",STAY,EBEAMSCAN,5.0,8.0,16.0,2048.0,V,1,-701.436,-699.430,-700.433,,%UserRoot%LINE\EBEA1650.LSC,,,1.000,,,,,,,1119162955,06-19-05 19:35:5,Thr:,
5,0.000000,0.000000,0.000000,0.000000,0.000000,0.700000,-0.600000,6.000000,A,UV,"Drive Command",,,,,,,,,,,,,,,,1.000,,,,,,,,,,
6,0.000000,0.000000,0.000000,0.000000,0.000000,0.700000,-0.585000,6.000000,LS,UV,"BD2 measure",STAY,EBEAMSCAN,8.0,5.0,2048.0,16.0,U,1,699.153,701.336,700.245,,%UserRoot%LINE\EBEA1651.LSC,,,1.000,,,,,,,1119162961,06-19-05 19:36:0,Thr:,
7,0.000000,0.000000,0.000000,0.000000,0.000000,0.700000,-0.615000,6.000000,LS,UV,"BD2 measure",STAY,EBEAMSCAN,8.0,5.0,2048.0,16.0,U,1,698.720,700.839,699.779,,%UserRoot%LINE\EBEA1652.LSC,,,1.000,,,,,,,1119162961,06-19-05 19:36:0,Thr:,
8,0.000000,0.000000,0.000000,0.000000,0.000000,0.685000,-0.600000,6.000000,LS,UV,"BD2 measure",STAY,EBEAMSCAN,5.0,8.0,16.0,2048.0,V,1,-601.083,-599.021,-600.052,,%UserRoot%LINE\EBEA1653.LSC,,,1.000,,,,,,,1119162962,06-19-05 19:36:0,Thr:,
9,0.000000,0.000000,0.000000,0.000000,0.000000,0.715000,-0.600000,6.000000,LS,UV,"BD2 measure",STAY,EBEAMSCAN,5.0,8.0,16.0,2048.0,V,1,-601.460,-599.414,-600.437,,%UserRoot%LINE\EBEA1654.LSC,,,1.000,,,,,,,1119162963,06-19-05 19:36:0,Thr:,
10,0.000000,0.000000,0.000000,0.000000,0.000000,0.000000,0.000000,6.000000,A,UV,"Drive Command",,,,,,,,,,,,,,,,1.000,,,,,,,,,,
11,0.000000,0.000000,0.000000,0.000000,0.000000,0.000000,0.015000,6.000000,LS,UV,"Cross measure",STAY,EBEAMSCAN,8.0,5.0,2048.0,16.0,U,1,-1.308,1.673,0.181,,%UserRoot%LINE\EBEA1887.LSC,,,1.000,,,,,,,1119163399,06-19-05 19:43:1,Thr:,
12,0.000000,0.000000,0.000000,0.000000,0.000000,0.000000,-0.015000,6.000000,LS,UV,"Cross measure",STAY,EBEAMSCAN,8.0,5.0,2048.0,16.0,U,1,-1.769,1.264,-0.253,,%UserRoot%LINE\EBEA1888.LSC,,,1.000,,,,,,,1119163399,06-19-05 19:43:1,Thr:,
13,0.000000,0.000000,0.000000,0.000000,0.000000,-0.015000,0.000000,6.000000,LS,UV,"Cross measure",STAY,EBEAMSCAN,5.0,8.0,16.0,2048.0,V,1,-1.545,1.276,-0.136,,%UserRoot%LINE\EBEA1889.LSC,,,1.000,,,,,,,1119163400,06-19-05 19:43:2,Thr:,
14,0.000000,0.000000,0.000000,0.000000,0.000000,0.015000,0.000000,6.000000,LS,UV,"Cross measure",STAY,EBEAMSCAN,5.0,8.0,16.0,2048.0,V,1,-2.006,0.907,-0.550,,%UserRoot%LINE\EBEA1890.LSC,,,1.000,,,,,,,1119163401,06-19-05 19:43:2,Thr:,
15,0.000000,0.000000,0.000000,0.000000,0.000000,-0.600000,0.000000,6.000000,A,UV,"Drive Command",,,,,,,,,,,,,,,,1.000,,,,,,,,,,
16,0.000000,0.000000,0.000000,0.000000,0.000000,-0.600000,0.015000,6.000000,LS,UV,"Cross measure",STAY,EBEAMSCAN,8.0,5.0,2048.0,16.0,U,1,-601.176,-598.411,-599.795,,%UserRoot%LINE\EBEA1891.LSC,,,1.000,,,,,,,1119163406,06-19-05 19:43:2,Thr:,
17,0.000000,0.000000,0.000000,0.000000,0.000000,-0.600000,-0.015000,6.000000,LS,UV,"Cross measure",STAY,EBEAMSCAN,8.0,5.0,2048.0,16.0,U,1,-601.717,-598.784,-600.253,,%UserRoot%LINE\EBEA1892.LSC,,,1.000,,,,,,,1119163407,06-19-05 19:43:2,Thr:,
18,0.000000,0.000000,0.000000,0.000000,0.000000,-0.615000,0.000000,6.000000,LS,UV,"Cross measure",STAY,EBEAMSCAN,5.0,8.0,16.0,2048.0,V,1,-1.569,1.224,-0.173,,%UserRoot%LINE\EBEA1893.LSC,,,1.000,,,,,,,1119163408,06-19-05 19:43:2,Thr:,
19,0.000000,0.000000,0.000000,0.000000,0.000000,-0.585000,0.000000,6.000000,LS,UV,"Cross measure",STAY,EBEAMSCAN,5.0,8.0,16.0,2048.0,V,1,-1.805,0.658,-0.574,,%UserRoot%LINE\EBEA1894.LSC,,,1.000,,,,,,,1119163409,06-19-05 19:43:2,Thr:,
20,0.000000,0.000000,0.000000,0.000000,0.000000,-1.200000,0.000000,6.000000,A,UV,"Drive Command",,,,,,,,,,,,,,,,1.000,,,,,,,,,,
21,0.000000,0.000000,0.000000,0.000000,0.000000,-1.200000,0.015000,6.000000,LS,UV,"Cross measure",STAY,EBEAMSCAN,8.0,5.0,2048.0,16.0,U,1,-1201.172,-1198.495,-1199.836,,%UserRoot%LINE\EBEA1895.LSC,,,1.000,,,,,,,1119163414,06-19-05 19:43:3,Thr:,
```