

Copyright © 1991, by the author(s).
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

**INVESTIGATION OF OPTICAL PHENOMENA
IN PHOTOLITHOGRAPHY AND OPTICAL
METROLOGY USING MASSIVELY PARALLEL
SIMULATION**

by

Karim H. Tadros

Memorandum No. UCB/ERL M91/72

28 August 1991

COVER PAGE

**INVESTIGATION OF OPTICAL PHENOMENA
IN PHOTOLITHOGRAPHY AND OPTICAL
METROLOGY USING MASSIVELY PARALLEL
SIMULATION**

by

Karim H. Tadros

Memorandum No. UCB/ERL M91/72

28 August 1991

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

TITLE PAGE

**INVESTIGATION OF OPTICAL PHENOMENA
IN PHOTOLITHOGRAPHY AND OPTICAL
METROLOGY USING MASSIVELY PARALLEL
SIMULATION**

by

Karim H. Tadros

Memorandum No. UCB/ERL M91/72

28 August 1991

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

Investigation of Optical Phenomena in Photolithography and Optical Metrology Using Massively Parallel Simulation

Karim H. Tadros

Department of Electrical Engineering and Computer Sciences,
University of California at Berkeley

July 23, 1991

ABSTRACT

A massively parallel computer algorithm that solves the Maxwell equations has been evaluated based on performance and suitability to analyze electromagnetic scattering from nonplanar topography. The program, **TEMPEST** (an acronym that stands for "Time-domain Electromagnetic Massively Parallel Evaluation of Scattering from Topography"), mates the inherent parallel nature of electromagnetic wave propagation with the power of the massively parallel computer architecture of the Connection Machine. The convergence and accuracy of the algorithm and its applicability to reflective notching and optical linewidth measurements are considered in detail. Efforts undertaken to extend the algorithm's capabilities to include illumination at oblique angles of incidence and implement symmetric boundary conditions are also presented. The algorithm has been tested for scalability in solving different problem sizes and running on different machine sizes. A very significant finding indicates that the simulations required constant time per iteration for problems ranging from a few thousand unknowns up to one million, providing the ratio between the problem size and the number of processors is kept constant. Convergence in the time-domain was found to be dominated by the physical process of multiple scattering from the given topography. The results of two detailed simulation studies show the suitability of **TEMPEST** in analyzing electromagnetic scattering issues in photolithography and optical metrology. One study investigates the role of wafer topography in generating reflective notching during resist exposure and gives insight to the possible mechanisms that cause notching and the capability of various methods to reduce the undesirable effects. A study of optical linewidth metrology of polysilicon gates illustrates how the optical image profile of the gate structure is affected by small variations in material thicknesses, the index of refraction, and the focus offset of the imaging microscope by comparing simulated images with images measured using an NBS-type optical microscope.

Acknowledgments

I extend sincere appreciation to the many individuals with whom I have had the great benefit of learning. Most notably, I thank Professor Andrew Neureuther for his constant encouragement and guidance. I thank both Dr. Roberto Guerrieri and John Gamelin for providing technical discussions regarding the details of the unique algorithm they developed. For their technical and moral support and their friendship I extend a warm thanks to my colleagues: Takeshi Doi, John Hutchinson, Rich Ferguson, Kenny Toh, Alex Wong, Ed Scheckler, Nelson Tam, Alfred Wong, Robert Wang. I am deeply grateful for my parents, my brothers and especially my wonderful wife who instilled in me the will to pursue new challenges.

The financial support provided by the SRC / SEMATECH Center of Excellence on Optical Lithography and Pattern Transfer (88-MC-500) is gratefully acknowledged.

Table of Contents

Chapter 1: Introduction	1
References	4
Chapter 2: Performance Evaluation	5
2.1. Scaling of Problems and Machine	5
2.2. Convergence in the Time-Domain	7
2.3. Examples	9
References	11
Figures	12
Chapter 3: Reflective Notching	15
3.1. Introduction	15
3.2. TEMPEST and Testing	16
3.3. Energy Coupling	18
3.4. Resist Surface Refraction	19
3.5. Surface Grain Size	20
3.6. Specular Structures	20
3.7. Role of Material Properties	22
3.8. Conclusions	23
References	25
Figures	27

Chapter 4: Linewidth Metrology of Polysilicon Gates	32
4.1. Introduction	32
4.2. Overview of Experiment and Issues	33
4.3. Thin Film Characterization	35
4.4. Optical Image Measurement Using an NBS System	36
4.5. Simulations	37
4.6. Image Post Processing	39
4.7. Comparisons of Simulation and Experiment	40
4.7.1. Line Edge	40
4.7.2. Line Width	40
4.7.3. Focus	41
4.8. Conclusions	42
References	43
Figures	45
Chapter 5: Extensions to the Algorithm	49
5.1. Oblique Incidence	49
5.2. Symmetric Boundary Conditions	52
References	55
Figures	56
Chapter 6: Conclusions	59
APPENDIX – TEMPEST User’s Guide	

Chapter 1

Introduction

Electromagnetic scattering from wafer topography and diffraction through apertures in photomasks have been key considerations in advancing from one generation to the next in semiconductor manufacturing. Computer simulation of these optical phenomena is playing an increasingly important role in the development of today's semiconductor processes. Numerical models of wave propagation must be based on more rigorous models in order to accurately predict the effects of interference and scattering while still providing reasonably quick results. This document examines the capabilities of a rigorous numerical approach which through parallelism has both the accuracy and speed to address these fundamental issues in electromagnetic scattering and optical imaging.

TEMPEST^{†1,2,3} is a massively parallel computer solution of the Maxwell electromagnetic wave equations using a time-domain, finite-difference method similar to the formulation proposed by Yee⁴ and extended by Wojcik.⁵ This rigorous algorithm mates the inherent parallel nature of electromagnetic wave propagation with the power of the massively parallel computer architecture of the Thinking Machines Corp. Connection Machine. Electromagnetic wave propagation and scattering are simulated by solving the wave equations explicitly on a physical grid of processors where each processor corresponds to a uniformly-spaced grid node in the simulated structure. The problem is solved in the time-domain rather than in the frequency domain to limit processor communication to nearest neighbors; hence, eliminating the need for a matrix solution.

TEMPEST simulations have been used to analyze electromagnetic scattering and optical imaging issues in the areas of photolithography,⁶ optical metrology,⁷ alignment mark signal quality,⁸ and signal integrity through photomasks.⁹ TEMPEST is capable of simulating electromagnetic scattering from

[†] "TEMPEST" is an acronym that stands for "Time-domain Electromagnetic Massively Parallel Evaluation of Scattering from Topography."

arbitrarily nonplanar and inhomogeneous topography while monitoring the dynamic changes in the optical constants of resist. Simulations are currently limited to transverse-electric (TE) polarized illumination at normal incidence impinging upon 2-dimensional periodic, isolated, or symmetric structures.

In chapter 2, the TEMPEST program is evaluated based on its merits as a massively parallel algorithm by measuring how its performance scales with machine size and problem size. An assessment of the algorithm's effective parallelism is made based on the sequential contribution introduced by the boundary conditions. Also, the issue of simulation convergence in the time-domain is illustrated by means of an example.

Two detailed simulation studies which examine electromagnetic scattering issues in photolithography and optical metrology are presented and illustrate TEMPEST's capabilities. First, Chapter 3 investigates the role of wafer topography in causing reflective notching of the photoresist during exposure and explores the possibility of reducing the effects through the introduction of special properties of resists and antireflection coating materials. The emphasis is on examining physical scattering mechanisms such as focused specular reflections from highly reflective substrates, reflections from substrate grains, resist thickness interference effects, and focusing of incident light by the resist surface curvature. The relative effectiveness of increasing resist contrast and absorption, and reducing substrate reflectivity in an attempt to minimize the effects of notching is addressed.

Chapter 4 presents a study of electromagnetic scattering and optical imaging issues related to optical linewidth measurements of polysilicon gate structures. Careful characterization of thin film optical parameters and rigorous electromagnetic scattering simulations are used to predict and systematically investigate optical image profiles of polysilicon / oxide / silicon line structures. A spectral component-weighting technique is applied to synthesize optical microscope images based on the diffraction efficiencies from normally incident illumination. Simulated image profiles of isolated

edges and $1.2\ \mu\text{m}$ wide lines are then compared with images obtained using an NBS-type optical microscope. The dramatic effects caused by an 8 nm variation in polysilicon thickness on the electric field distribution within the gate, on reflectivity, and on the image profile are illustrated. Also, the effect of focus position is shown by comparing measured and simulated image profiles at different focus offsets.

Chapter 5 presents two efforts undertaken to extend the capabilities of the algorithm. First, in order to predict the effects of high numerical apertures and partial coherence, **TEMPEST** must be capable of simulating illumination with rays incident at oblique angles. The formulation of a forcing function that allows obliquely incident illumination is presented for the special case of a periodic structure and specific angles of incidence. The difficulty in implementing off-axis illumination is discussed. Second, by taking advantage of the symmetry found in a given structure, a higher grid density or larger structure can be simulated. The formulation of symmetric boundary conditions is presented and their effectiveness is illustrated with a graphical example.

References

1. J. Gamelin, "Simulation of Topography Scattering for Optical Lithography with the Connection Machine," M.S. Thesis, Memorandum No. UCB/ERL M89/71, University of California, Berkeley, May 1989.
2. J. Gamelin, R. Guerrieri and A.R. Neureuther, "Exploration of Scattering from Topography with Massively Parallel Computers," *J. Vac. Sci. Technol. B.*, vol. 7, no. 6, pp. 1984-1990, Nov/Dec 1989.
3. R. Guerrieri, K. Tadros, J. Gamelin, and A.R. Neureuther, "Massively Parallel Algorithms for Scattering in Optical Lithography," *IEEE Trans. CAD*, to appear.
4. K. S. Yee, "Numerical Solution of Initial Boundary Value Problems involving Maxwell's Equations in Isotropic Media," *IEEE Trans. Antennas Propagat.*, vol. AP-14, pp. 302-307, May 1966.
5. G. Wojcik, D. Vaughn and L. Galbraith, "Calculation of Light Scatter from Structures on Silicon Surfaces," *Proc. SPIE: Lasers in Microlithography*, vol. 772,, pp. 21-31, 1987.
6. K. Tadros, A.R. Neureuther, J. Gamelin and R. Guerrieri, "Investigation of Reflective Notching with Massively Parallel Simulation," *Proc. SPIE: Optical/Laser Microlithography III*, vol. 1264, pp. 322-332, March 1990.
7. K. Tadros, A.R. Neureuther and R. Guerrieri, "Understanding Metrology of Polysilicon Gates through Reflectance Measurements and Simulation," *Proc. SPIE: Integrated Circuit Metrology, Inspection and Process Control V*, vol. 1464, March 1991.
8. A. Wong, T. Doi, D. Dunn, and A.R. Neureuther, "Experimental and Simulation Studies of Alignment Marks," *Proc. SPIE: Optical/Laser Microlithography IV*, vol. 1463, March 1991.
9. T. Doi, K. Tadros, B. Kuyel, and A.R. Neureuther, "Edge-Profile, Materials and Protective Coating Effects on Image Quality," *Proc. SPIE: Integrated Circuit Metrology, Inspection and Process Control V*, vol. 1464, March 1991.

CHAPTER 2

Performance Evaluation

TEMPEST has previously been tested for numerical accuracy by comparing the diffraction efficiencies of both 1-dimensional and 2-dimensional structures with values computed using well-established analytical techniques.¹ Accuracy comparable to the fully rigorous methods was achieved. In this chapter, the algorithm is evaluated based upon its ability to scale with machine size and problem size and on its ability to maintain parallelism. Also, the issue of steady-state convergence in the time-domain is examined. The ability of TEMPEST to simulate electromagnetic scattering is illustrated with examples of imaging through a photomask and projection printing in the presence of specular reflections from curved substrates.

2.1. Massively Parallel Performance

Several issues unique to the massively parallel approach have prompted performance testing of the algorithm within the specific context of its implementation on the Connection Machine.[†] First, the benefit of the parallel approach was quantified. The same problem was simulated on a number of different grid sizes, by increasing the number of processors while maintaining the ratio of problem size[‡] to the total number of available processors. In Connection Machine jargon, the machine size was varied while maintaining a constant *VP* ratio* (*i.e.* ratio of virtual to physical processors). The CPU time required to complete 1000 iterations of the simulation was recorded for each of the different

[†] For detailed information on the Thinking Machines Corp. Connection Machine, the reader may refer to the CM Technical Reference Manual.² For information on the specific implementation of the TEMPEST algorithm on the Connection Machine, the reader may refer to the original formulation by Gamelin.³

[‡] The term "problem size" in this context is equivalent to "total number of processors in the simulation," where each processor corresponds to one grid node.

* Each processor in the Connection Machine is capable of emulating some V virtual processors. The memory of the original physical processor is split equally among the V virtual processors. The ratio of virtual-to-physical processors is termed the "*VP* ratio."

problem sizes (Figure 1). One iteration represents one time step (dt) in wave propagation. Each curve in Figure 1 represents a different ratio between problem size and available resources (*i.e.*, a different *VP* ratio). Since for any given curve, the problem size can increase without any significant increase in the CPU time, the performance of the algorithm improves proportionally with problem size.

The ability to operate with virtual processors offers the tradeoff of solving larger problems with the same number physical resources at the cost of increased computation time. This tradeoff was measured by recording the time required to complete 1000 iterations for a number of different *VP* ratios (Figure 2). The different curves represent different machine sizes. The fact that all three curves fall relatively close to each other indicates that the effect of using virtual processors is independent of the amount of available resources. This graph indicates that for *VP* ratios up to 4, CPU time is constant and is dominated by system overhead. For larger *VP* ratios, the curves are sub-linear indicating that the system overhead is still a significant component of the total computation time. This nonlinearity is a function of the Connection Machine operating system and not of the TEMPEST algorithm. Earlier predictions¹ of system performance based on simulations running with a *VP* ratio of 16 on the 1K prototype Connection Machine at Berkeley assumed linear scaling. The fact that no significant degradation in system performance exists for simulations operating with *VP* ratios up to 4 also explains the mistaken prediction.

A major concern during the early development of the absorbing boundary conditions for this algorithm was the need to minimize the amount of time required to update the boundary nodes.³ This implied that the equations to compute the fields at the absorbing boundaries should be as similar as possible to the equations for the nodes in the bulk of the domain. For each dissimilar computation, the bulk nodes would have to sit idle, thus reducing the overall parallelism of the algorithm. This is particularly important for the isolated version of TEMPEST which utilizes absorbing boundary conditions at all four boundaries. Since each boundary must be treated separately because of its orientation to the bulk nodes, the sequentiality introduced in the boundary conditions is magnified by

four in the simulations of isolated structures. The actual time required to update the fields in the boundary nodes of the isolated version was compared to the time needed to update the nodes in the bulk of the domain. These timing measurements were made for a number of different problem sizes. As seen in Figure 3, roughly 75% of the computation time was found to be consumed by the absorbing boundary conditions. The remaining 25% was consumed by the bulk nodes. The total time considered here does not include the overhead involved in loading and unloading the processors. If one considers that for large problems, the number of computations performed by the boundary nodes is negligible compared to the number performed by the bulk nodes, the percent of computational resources actually used by this algorithm is 25%. Note, however, that the Connection Machine reaches its peak performance when updating the inner nodes, which implies that the sustained performances are still very high.

2.2. Convergence in the Time-Domain †

One aspect of this algorithm is that the steady-state solution of the electromagnetic fields cannot be expressed in closed form. Steady-state convergence in the time-domain depends on the physical process of multiple reflections from the specified topography. The time to reach steady-state varies with the material constants, physical dimensions and the geometries of the scatterers. Steady-state is achieved when all transient waves die out. TEMPEST determines when the system has converged by comparing the electric field value from one period to the next at every node along a horizontal cutline near the top boundary in the simulation domain. The maximum difference at any node along the cutline cannot exceed a specified fraction of the incident field amplitude. This condition can be expressed as

$$\max_{i=0 \rightarrow x_{\max}} \left| E_z^t(i,j) - E_z^{t+Nd}(i,j) \right| < f_{error} \times \left| E_z^{inc} \right|$$

† Takeshi Doi is gratefully acknowledged for his assistance in implementing and testing the convergence criterion.

where t is the instant in time, Ndt is the period of the incident wave, f_{error} is a user definable error tolerance, and j is held constant, indexing the row corresponding to the cutline. To ensure maximum sensitivity, the fields are checked beginning at a time t_0 , when the incident electric field is a maximum at the cutline. After several periods of excitation have been allowed to propagate, this criterion is monitored and must be satisfied for three consecutive periods to ensure the system has converged to its steady-state condition.

The three structures in Figures 4(a), 4(b) and 4(c) were used to test the convergence in the time-domain. The structures consisted of vacuum, resist, and silicon layers, each with a different resist thickness. The dotted line at the top of the domain indicates the location where the electric fields were monitored. A normally incident wave was excited along the top boundary of the domain. The root-mean-square difference in electric fields along the cutline was plotted after each cycle of excitation (Figure 5). Two peaks in the difference curves were observed. The first corresponds to the incident wave reflecting from the resist surface and returning to the cutline. The second peak corresponds to the reflection from the resist/silicon interface. As the resist thickness increases, the optical path length from the cutline to the vacuum/resist interface is reduced while the optical path length to the resist/silicon interface increases since resist is optically slower than vacuum. This results in the first peak occurring earlier in the simulation while the second peak occurs later. It can be seen from this simple study that the time to reach steady-state is dependent upon the physical process of multiple scattering from the specific topography.

For more complicated (*i.e.*, nonplanar, inhomogeneous) structures, the time to reach steady-state will vary. In fact, even for the previous set of structures (Figure 4), had there not been attenuation in the resist and silicon, secondary reflections between the two interfaces could have created more transients requiring even longer for the system to reach steady-state. Based on the experience of hundreds of simulations, roughly 80% of the simulations performed required less than 50 cycles of excitation to reach steady-state.

2.3. Examples

A key advantage to the massively parallel, time-domain approach is in its ability to provide a physical view of the scattering effects with no additional effort. This “picture” can be of the instantaneous electric field at any time or the maximum electric field during steady-state (*i.e.*, depiction of standing waves). To illustrate, Figure 6 displays the maximum, steady-state electric field obtained from a simulation of electromagnetic diffraction through the photomask structure displayed in Figure 7. The simulation was performed using an illumination wavelength of 0.248 μm . The periodic structure is a quartz plate ($n = 1.50$) coated with 800 \AA thick chrome (modeled as a perfect conductor) with an aperture 0.25 μm wide. A normally incident plane wave with its electric field parallel to the edge of the chrome aperture was applied at the top boundary. The wave then propagated into the domain and scattered from the material interfaces. From the 3-dimensional plot of the maximum, steady-state electric field, a number of wave scattering effects are observed. First, light was able to pass through the mask only where there was no chrome. The radiation that did pass through the wavelength-sized aperture spread in a fashion very similar to Huygen’s point source principal. The amount of radiation reaching the regions immediately beneath the chrome but far from the aperture was minimal. Standing waves were found in regions above the chrome due to the highly reflective chromium layer. The amplitude of the electric field inside the quartz layer is noticeably less than the amplitude in the vacuum region above the quartz due to the difference in dielectric constant. And scattering from the edges of the aperture caused a modulation of the standing waves above the mask. TEMPEST simulations similar to this were used to perform an in-depth study of the impact of chrome-edge profiles, phase shifter materials, protective mask coatings, and reflective masks on projection printed image quality.⁴

The second example consists of examining the difficulty found in imaging contact holes in negative photoresist over slightly dished topography.⁵ The problem is believed to be associated with specular reflection from the substrate redirecting the incident energy into nominally unexposed areas.

A typical geometry is shown in Figure 8 where the curvature has been designed to focus incoming light at a point 1.0 μm above the bottom of the dish. Incident images for 1.0 μm contact holes in positive and negative mask polarity were generated using SPLAT⁶ and input to TEMPEST. A wavelength of 436 nm, numerical aperture of 0.38 and partial coherence factor of 0.5 were used. In this regime the square root of the scalar intensity from SPLAT is a reasonable approximation to the vector electric field.

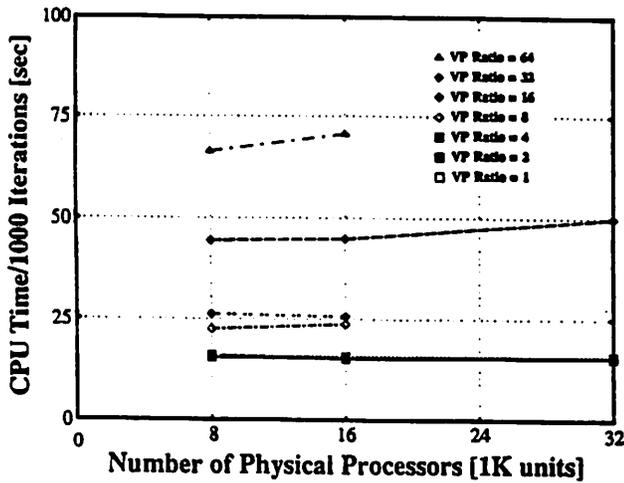
This simulation used a refractive index of $n = 4.73$, $k = -0.136$ for the polysilicon. The resist was modeled using the Dill's ABC resist parameters. The values used for bleachable absorption (A), unbleachable absorption (B), and resist sensitivity (C), were 0.51, 0.031, and 0.013 respectively. These values correspond to KTI-820 positive photoresist. The value $n=1.67$ was used for the real part of the resist refractive index. The same resist parameters were used to transfer both the positive and negative tone images since the primary interest is in the scattering effects.

The images were then propagated in two-dimensions by TEMPEST to bleach photoresist on the curved polysilicon topography. The contours shown in Figure 8 are for equal photoactive compound concentrations after 70 mJ/cm^2 of exposure of KTI-820 resist. For the positive resist where the central region is illuminated the curvature helps focus the light back into the illuminated area. For the negative resist shown in Figure 9 the substrate specularly reflects the light into the dark central portion giving unwanted exposure.

References

1. J. Gamelin, R. Guerrieri and A.R. Neureuther, "Exploration of Scattering from Topography with Massively Parallel Computers," *J. Vac. Sci. Technol. B.*, vol. 7, no. 6, pp. 1984-1990, Nov/Dec 1989.
2. *Connection Machine Model CM-2 Technical Summary*, Thinking Machines Corporation, April 1987.
3. J. Gamelin, "Simulation of Topography Scattering for Optical Lithography with the Connection Machine," M.S. Thesis, Memorandum No. UCB/ERL M89/71, University of California, Berkeley, May 1989.
4. T. Doi, K. Tadros, B. Kuyel, and A.R. Neureuther, "Edge-Profile, Materials and Protective Coating Effects on Image Quality," *Proc. SPIE: Integrated Circuit Metrology, Inspection and Process Control V*, vol. 1464, March 1991.
5. Joe Langston. private communication
6. K.H. Toh and A.R. Neureuther, "Identifying and Monitoring Effects of Lens Aberrations in Projection Printing," *Proc. SPIE: Optical Microlithography VI*, vol. 772, pp. 202-209, 1987.

CPU Time vs. Number of Processors



CPU Time vs. VP Ratio

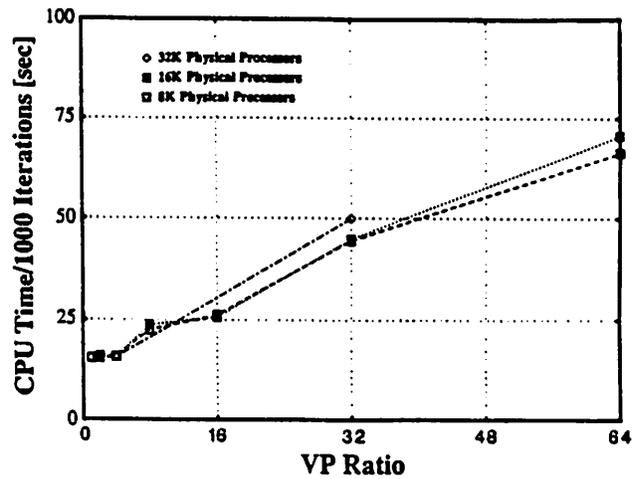


Figure 1. CPU time to compute 1000 iterations of the simulation versus the number of physical processors (*i.e.*, machine size). Each curve represents simulations running on machines operating at the same virtual-to-physical (*VP*) ratio. Note that for *VP* ratio = 32, the total number of processors used in the simulations ranged from $8K \times 32 = 256K$ to $32K \times 32 = 1M$. The curves for *VP* ratios 1, 2 and 4 are equivalent.

Figure 2. CPU time to compute 1000 iterations of the simulation for three different *VP* ratios on different machine sizes.

CPU Time for Boundary and Bulk Nodes

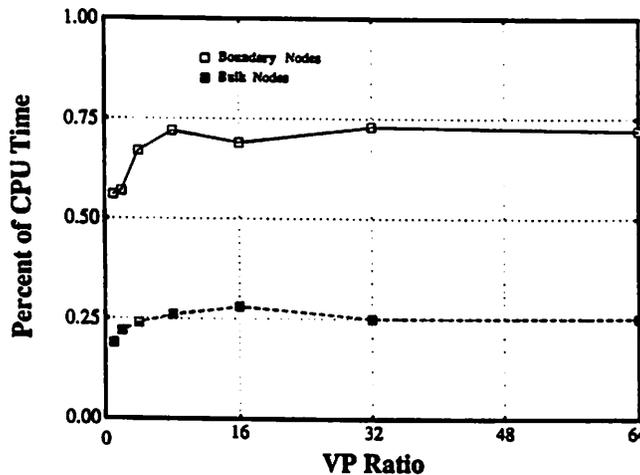


Figure 3. Percent CPU time spent updating the boundary nodes and the bulk nodes for a number of different problem sizes on the same machine.

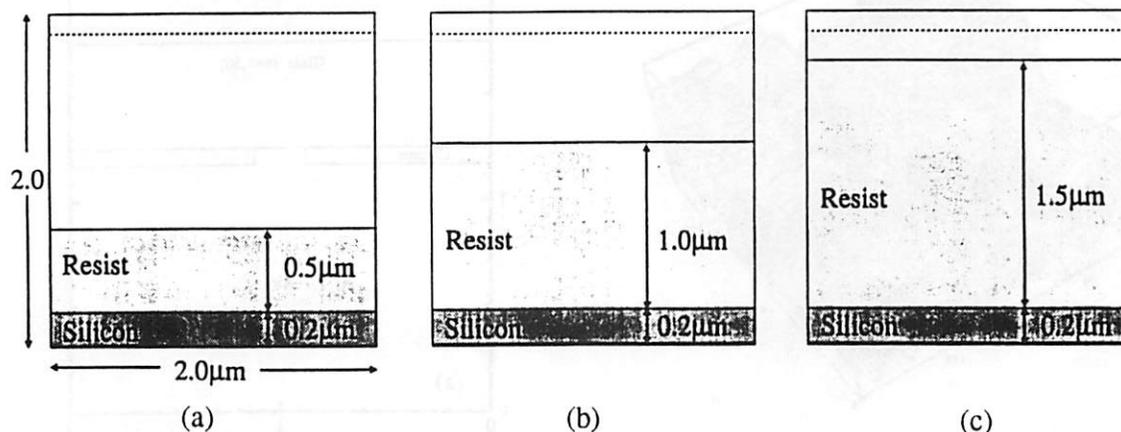


Figure 4. Test structures comprised of vacuum ($n = 1.0$, $k = 0.0$), resist ($n = 1.68$, $k = -0.028$), and silicon ($n = 4.82$, $k = -0.117$), used to investigate the time to reach steady-state. Each contains a different thickness of resist: (a) $0.5 \mu\text{m}$, (b) $1.0 \mu\text{m}$, and (c) $1.5 \mu\text{m}$. The dotted line indicates where the electric field was monitored at each cycle.

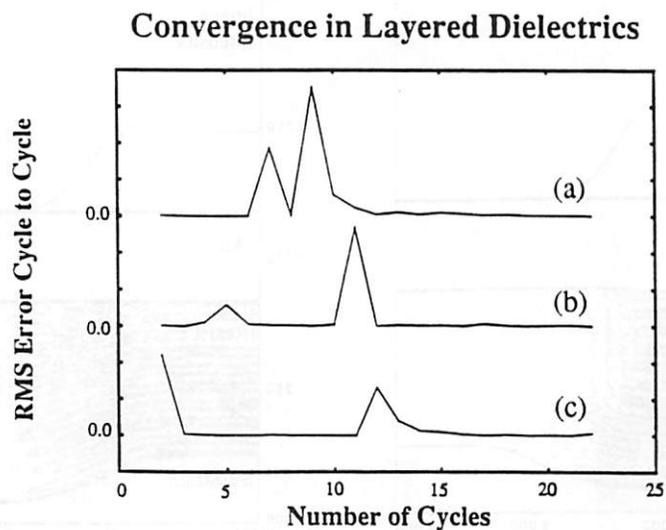


Figure 5. Root-mean-square difference of the electric field value from cycle to cycle at each node along the cutline (Figure 4) for each of the three resist thickness: (a) $0.5 \mu\text{m}$, (b) $1.0 \mu\text{m}$, and (c) $1.5 \mu\text{m}$.

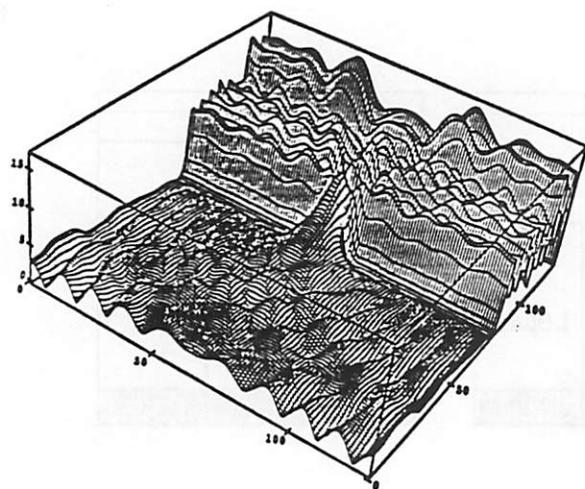


Figure 6. The maximum steady-state electric field after simulating illumination ($\lambda = 246$ nm) through the photomask depicted in Figure 7.

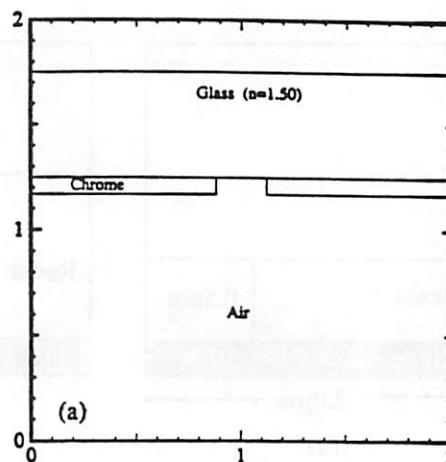


Figure 7. The photomask structure used to generate Figure 6. Photomask structure: air ($n=1.0$) / quartz ($n=1.5$) / chrome (perfect conductor) / air. The aperture is $0.25 \mu\text{m}$ wide and the wavelength used in the simulation is 248 nm

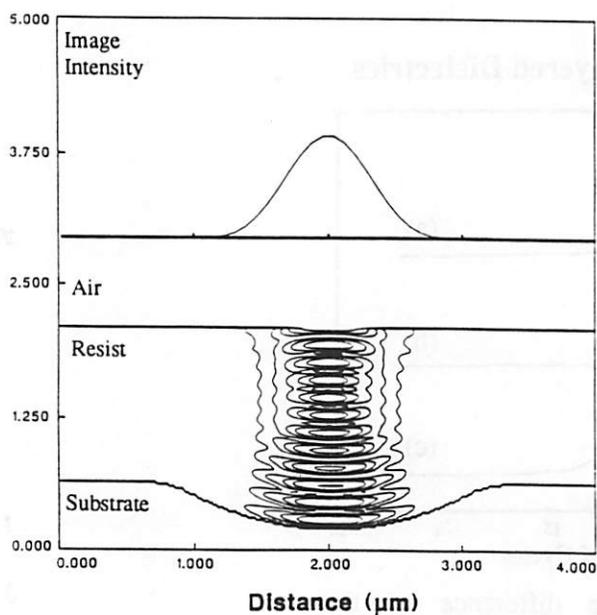


Figure 8. Bleaching of positive photoresist with an incident image (top) showing focusing due to the specular reflection from the substrate. The contours are for equal phototactic compound concentrations at 15% intervals.

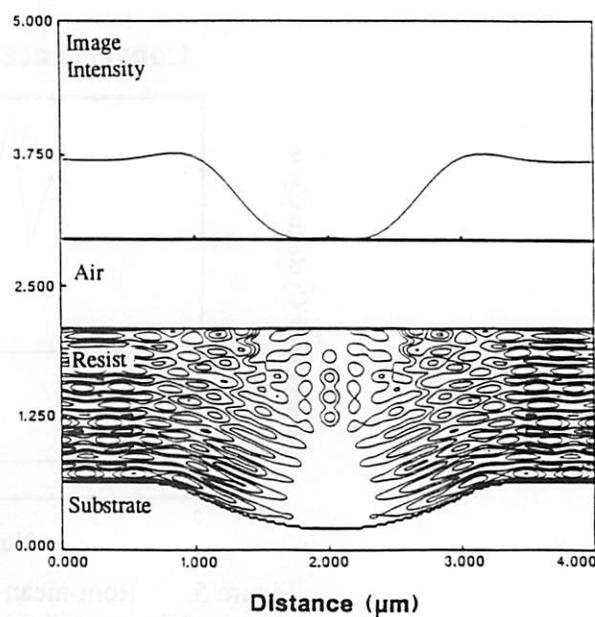


Figure 9. Bleaching of negative photoresist showing specular reflection of the illumination from the dished topography into the masked region.

Chapter 3

Reflective Notching[†]

3.1. Introduction

As the technology of integrated circuit fabrication continues to push the limits of optical resolution and the efficient usage of already crowded real estate, wafer topography is becoming increasingly nonplanar. As a consequence, photolithography must contend with topographical features that cause specular reflection of the incident illumination into unexposed regions of the photoresist. Development of the latent images reveals structures that have been severely notched or have significantly deviated from their intended dimensions. Examples of linewidth variations at steps have been described by Widman.¹ Recent examples investigating the role of resist curvature, resist material characteristics, and antireflection coatings can be found in work by Guibert,² Housley,³ Karnett,⁴ White,⁵ Bruce⁶ and Nölscher.⁷

Resist notches or dimples which can be attributed to focused specular reflections are of particular concern. Notching of this type can be seen in Figure 18 of Guibert.² A very distinctive hole in the top of a resist profile can be found in Figure 13a of Karnett.⁴ Figure 1(a)⁸ shows notching in the side of a resist line passing over a bowl-shaped region on a reflective substrate. Positive-type resist was used and the underlying layer is Al-Si-Cu. By using a resist with a different contrast or by adding dye to the original resist, the reflective notching is reduced as shown in Figures 1(b) and 1(c) respectively.

Computer simulation of the photolithography process has become a fundamental tool for predicting the complex interaction of projected optical images with photoresist exposure response and predicting the resulting developed resist profiles. Simulation of reflective notching, however, is a particularly difficult problem because the incident electromagnetic radiation must be allowed to bounce

[†] This chapter is extracted from the author's published paper "Investigation of Reflective Notching with Massively Parallel Simulation" *Proc. SPIE: Optical/Laser Microlithography III*, March 1990.

around inside structures with irregular dimensions on the order of several wavelengths in size. In addition, a wide variety of refractive indices are encountered and in the case of resist exposure, the dynamic change in the refractive index of the photoresist during the exposure process must be incorporated.

A new class of photolithography simulators has emerged to take on these challenging problems. A two and three-dimensional, time-domain, finite-element (FE) and finite-difference (FD) algorithm developed by Wojcik⁹ was used to simulate scattering from latex spheres of sub-micron diameter on a silicon half space. Matsuzawa¹⁰ developed a two-dimensional, frequency-domain solution of the Helmholtz equation using a finite-element/boundary-element method to simulate reflective notching near a perfectly conducting stepped substrate. Urbach and Bernard¹¹ have developed a two-dimensional, frequency-domain, finite-element/boundary-value algorithm which is suitable for examining imaging with partial coherence in the presence of underlying topography. In this study, the results of photolithography simulations of scattering from topography using TEMPEST are presented.

The emphasis of this study is on systematically studying mechanisms which affect imaging over topography. Energy coupling variations due to photoresist thickness changes are examined by exposing a linearly tapered, isolated region of photoresist on top of a flat silicon substrate. The effects of resist curvature are then studied by examining a worst-case situation of a "bubble" of resist refracting the incident illumination into a small region in the resist. Specular reflections from substrate grain structures are investigated by analyzing the relative effects of grain dimensions. Focused specular reflections are then examined by analyzing effective local exposure variations and developed resist profiles of lines imaged near a perfectly conducting parabolic reflector. Finally, the role of resist contrast and absorption as well as substrate reflectivity in reducing these effects are briefly explored.

3.2. TEMPEST and Testing

To facilitate meaningful comparisons of the effects caused by different mechanisms, a single set of exposure parameters for the resist and imaging tool were chosen for use in the simulations presented throughout this study. The ABC exposure parameters for KTI-820 positive resist at g-line and the associated development rate parameters for KTI-932 developer are given in Table 1. The characteristics of the g-line imaging system are given in Table 2. The choice of high numerical aperture emphasizes the differences which occur due to wave propagation at off-axis angles. For convenience, the substrate was modeled as a perfect conductor, except where noted.

To demonstrate the accuracy of TEMPEST for resist bleaching the simulation results of resist profiles on planar substrates were compared with those from SAMPLE exposure. Figure 2 presents simulations based on exposure/bleaching calculations made by (a) TEMPEST and (b) SAMPLE of a 1.25 μm line imaged in 1.0 μm of KTI-820 resist over a flat substrate. The incident field used by TEMPEST was calculated by SAMPLE assuming a 1.25 μm line in a 4.0 μm pitch. This image was input to TEMPEST at a height of 0.1 μm above the resist surface. The photoactive compound (PAC) concentrations throughout the resist layer was then determined by TEMPEST. This involved simulating Maxwell's equations for the dynamically bleaching resist material on a rectangular grid of 128 vertical by 256 horizontal nodes. For each of the bleaching steps a coherent, transient, incident wave was introduced and followed to steady-state convergence. After 13 bleaching cycles the PAC concentration matrix was transferred to SAMPLE for the development simulation. The resist dissolution for the PAC matrix from TEMPEST, Figure 2(a), shows a more rapid clearing to the substrate in the exposed regions, than in Figure 2(b) from SAMPLE. This quicker development may be more physically correct since TEMPEST allows for lateral as well as vertical wave propagation through the resist medium.

Comparison of results have also been made on nonplanar substrates with other rigorous two-

dimensional simulators. TEMPEST has been shown to be accurate when compared with results of integral equation analysis of diffraction gratings.¹² Gamelin¹² also produced simulation results similar to Matsuzawa's¹⁰ two-dimensional simulation of electromagnetic scattering from a reflective stepped substrate. As an additional example, Figure 3 shows a simulation similar to that made by Urbach and Bernard¹¹ in their Figure 12 for the formation of a latent image on top of a bump made of a perfect conductor. The image contours are generally similar. However, an exact comparison cannot be made since TEMPEST does not currently include wavefront phase shifting for focusing within the resist and averaging for partial coherence. The effect of neglecting these phenomena in the investigation of the more basic reflective notching mechanisms is small.

The computational effort required to solve the class of reflective notching problems in this study is on the order of 1 GFLOP-min. The Connection Machine CM-2 has a computational speed of 2GFLOPs for 64K processors. For the examples in this paper, 32,000 virtual nodes were used on 1,000 actual processors. The run-time was 33 minutes for a total of 13 resist bleaching steps. Thus, one electromagnetic analysis took about 80 MFLOP-mins and the full exposure/bleaching required 1 GFLOP-min.

3.3. Energy Coupling

The accurate control of resist thickness is a significant challenge since only a $\lambda_{resist}/4$ fluctuation in resist thickness, in theory, can cause as much as a factor of 8.1 difference in absorbed energy when imaging above a silicon substrate. This difference in exposure can mean the difference between clearing a contact hole and only putting a dent in it. Figure 4 shows the variation in PAC concentration caused by the standing wave patterns that form in a slowly tapered resist on top of a silicon substrate. The resist was designed to vary in thickness by one $\lambda_{resist} = 0.2595 \mu\text{m}$ from left to right. Large exposure variations occur vertically throughout the resist and a noticeable exposure variation also occurs laterally. Note that there is a skew in the standing wave pattern due to refraction

at the resist surface. The tilt in the surface together with the thickness of the resist caused lateral displacement of the locations where good and poor exposure occurred at the bottom of the resist layer.

A quantitative interpretation of the TEMPEST results can be made in order to determine a local effective exposure relative to the same energy illuminating resist on a matched substrate in a clear field region. The electric field amplitude used as the incident field by TEMPEST is described by the equation:

$$E = \sqrt{2I(x)i_{scale} \frac{\eta_o}{n}}$$

where $I(x)$ is the normalized incident intensity from the SAMPLE aerial image calculation, i_{scale} is an arbitrary scaling factor used to obtain a desired peak intensity, and η_o/n is the characteristic impedance of the incident region with refractive index n . For typical values of $\frac{\eta_o}{n} = \frac{377}{1.68}$ and $i_{scale} = 0.1$ (as used in these simulations), the nominal electric field amplitude in resist over a matched substrate is 6.7. The effective local exposure is given by the square of these electric field values.

For the tapered resist simulation, the electric field amplitude in the resist varied in the neighborhood of the substrate from a peak of 9.40 to a null of 3.71, separated vertically by one quarter wavelength. The adjacent poor coupling region varied in amplitude from 7.12 to 2.78. This gives a local effective exposure relative to a matched substrate of 2.0, 0.31, 1.1, 0.17 respectively. The exposure thus varies by a factor of 6.5 vertically and by a factor of 1.8 horizontally. The difference in exposure variation from the theoretical value of 8.1 for a substrate with refractive index $n_{silicon} = (4.73 - i0.136)$ is due to refraction at the resist surface and possibly lateral spreading of the electromagnetic radiation as it travels through the resist. A small percentage of the deviation can also be attributed to absorption of the energy by the photoresist as it bleaches.

3.4. Resist Surface Refraction

Film curvature is unavoidable when working with thin films over nonplanar topography.⁵ This curvature in a photoresist surface is capable of redirecting the incident energy according to Snell's law of refraction and thus affecting the nominal exposure within the resist film. Figure 5 shows the contour lines of the electric field as it travels through a layer of resist with its surface designed such that normally incident illumination will converge upon the region in the lower left corner of the cross section. The resist was considered to be infinitely thick to prevent formation of standing waves. The left boundary is an absorbing boundary so that there is no contribution from the left side as there would be with a symmetric boundary condition. The refractive focusing effect of this quarter circle geometry enhanced local exposure by a factor of 2.2 over the nominal exposure.

3.5. Substrate Grain Size

Photolithography of sub-micron features must contend with substrate grain structures that are becoming a larger percentage of the device feature size. Spurious reflections from these grain structures can cause significant linewidth deviation if the substrate is of a highly reflective material. To analyze the effects of grain size, we simulated imaging of a 1.25 μm line near a perfectly conducting grain (Figure 6(a)). The grain is modeled simply as a triangular bump with a side length of λ_{resist} , sloped at 20° from horizontal, and positioned at the proper distance from the nominal line edge to cause the reflected energy to illuminate the lower half of the line.

Figure 6(b) shows the contours of post-exposure PAC concentration for this geometry. It is evident that energy has been directed into the area that would nominally be dark under the mask. Correspondingly, the developed profile for this geometry, Figure 7(a), exhibits significant line-edge asymmetry. Figure 7(b) shows the developed profile for a grain with the same side slope (20°) but with only $\lambda_{\text{resist}}/2$ side length. Even this minute structure begins to have a noticeable effect on profile asymmetry.

3.6. Specular Structures

As a result of the ever-increasing nonplanar wafer topography, the incident illumination in an exposure process can be specularly reflected in lateral directions. In certain cases, these reflections will travel into regions intended to be unexposed. The net effect of these specular reflections is to partially expose desired resist features and disturb their line-edge profile upon development.

To simulate the ultimate worst-case situation, a perfectly conducting substrate was designed in the shape of a half-parabola, as shown in Figure 8, with its focus located at the surface of the resist, 0.2 μm inside the nominal left line edge. Using the unique property of the parabola that all coaxial rays incident upon its inner surface are reflected to its ‘‘focus’’, an attempt was made to ‘‘blow a hole’’ in the top surface of the photoresist line. As seen in the normalized, post-exposure PAC concentration contours, Figure 9(a), the incident illumination left a trail of PAC variations as it made its way to the focus. At the surface near the focus, it not only exposed all the PAC, but it also incurred internal reflection and proceeded downward into the masked region of the resist.

At the focal point located under the mask, the effective exposure is a factor of $\left(\frac{20.75}{6.7}\right)^2$ or about 10 times the nominal clear field exposure. But for the simulation of a flat substrate, Figure 2(a), the exposure 0.2 μm under the mask is roughly $\left(\frac{3.2}{6.7}\right)^2 = 0.25$. Hence, due to the focused specular reflection by the substrate followed by a second reflection at the resist/air interface, the region under the mask at the focus has received an effective exposure more than $\left(\frac{20.75}{3.2}\right)^2 = 40$ times the intended exposure. Reflection at the resist/air interface ($\Gamma = \frac{1.68 - 1.0}{1.68 + 1.0} = 0.254$) causes $(0.254)^2 = 6.44\%$ of the energy incident upon the interface to bounce back into the resist. This corresponds to $\left(\frac{20.75}{3.2}\right)^2 \times 0.0644 = 2.7$ times more exposure under the mask than occurred for a flat substrate. The developed profile for this simulation, shown in Figure 9(b), is badly distorted and shows a ridge of

resist left untouched by the reflected energy as it bounced up then down through the masked region of the resist.

To verify that the notching is due to the downward internally reflected energy as well as the upwardly focused energy, the mask was shifted so that the focus would be located at $0.1\ \mu\text{m}$ outside the nominal left edge. The plot of PAC contours for this simulation, Figure 10(a), confirms the hypothesis. Variations in the PAC concentration are seen through the unexposed resist region. The developed profile, Figure 10(b), now shows a clear development track along a diagonal path within the resist profile from upper left to lower right. Comparing the 20 second development contour with the same contour in Figure 9(b) shows a greater retention of the line shape at the top as expected since the focus is outside the line in Figure 10(b). However, both cases show that energy reflected from the surface causes significant exposure in the intended masked regions of the resist.

Though the example of the parabolic reflecting geometry had a reasonably large vertical dimension ($0.5\ \mu\text{m}$), similar reflective notching effects can be seen with topography of the same curvature but with very small vertical heights. The geometry in Figure 8 was modified by reducing the vertical height of the substrate from $0.5\ \mu\text{m}$ to $0.0625\ \mu\text{m}$, which is on the order of one $\lambda_{\text{resist}}/4$. Even a curved step of this size with $\frac{1}{3}$ the collection area, is enough to cause significant developed resist line asymmetries compared to a flat topography as can be seen in Figure 11.

3.7. Role of Material Properties

Since it is not always possible to prevent the formation of specular reflectors in the course of a given process flow, certain techniques can be employed to help ensure control of feature size. Such measures may include adding dye to resist to increase absorption or using a resist with a different contrast between development rates of exposed and unexposed regions. Reducing substrate reflectivity is a viable solution but it usually requires additional steps in the process flow to deposit then remove an antireflection coating. The results of applying these techniques to the profiles previously considered

are discussed below.

To model increased resist contrast, the R_2 development rate parameter (unexposed development rate) was decreased by one order of magnitude to 0.0006183 (Table 1). The PAC matrix from the worst-case simulation, Figure 9, was then developed using the modified rate parameters. The resulting line profile, Figure 12(a), regained much of the intended line features but still showed significant notching near the designed focus point.

Increasing the resist absorption was implemented by increasing the exposure parameter, B , by one order of magnitude to $0.31 \mu\text{m}^{-1}$ (Table 1). Using the parabolic geometry as before, the simulation of the exposure was re-run using the modified exposure parameters. The developed profiles from this simulation, Figure 12(b), revealed improved feature retention compared to the original simulation, but still exhibited notching near the focus region.

The previous simulations involving the parabolic geometry were performed using a perfectly conducting substrate which has a reflection coefficient (Γ) equal to 1. This value is similar to what can be expected for highly reflective materials such as aluminum. Substrate reflectivities less than 1 were simulated to observe their impact upon effective exposure near the parabolic geometry. Figure 13(a) shows the developed profiles for the parabolic geometry using a substrate with $\Gamma = 0.48$ ($n = 4.73$, $k = -0.136$: silicon). Figure 13(b) shows the developed line profile for a simulation with substrate reflectivity $\Gamma = 0.25$ ($n = 2.8$, $k = -0.02$: hypothetical material). Figures 14(a) and 14(b) show development profiles of resist lines near grain-like structures comparable to those of Figure 7 but for a silicon substrate. These lower reflectivity materials greatly reduced the notching and show the importance of reducing substrate reflectivity to values equal to or less than that of silicon.

3.8. Conclusions

The effects of wafer topography on optical lithography can be rank ordered based on the degree to which the physical mechanisms affect the local effective exposure. As a point of reference,

interference effects within the resist on silicon substrates produce a peak-to-peak variation in local effective exposure from about 2.0 to 0.3 in regions of good energy coupling and from about 1.1 to 0.17 in regions of poor energy coupling. This results in a total variation in effective exposure on the order of 6.5 due to vertical standing waves and a factor of 1.8 horizontally due to thickness changes from even to odd multiples of $\lambda_{resist}/4$. The positions for good and poor coupling are determined by refraction along diagonal paths through the resist material. Focusing caused by refraction due to surface curvature of the resist showed a relatively minor influence which was about a factor of 2 for a large quarter circle lens. Focused specular reflections from the substrate caused the greatest damage by magnifying the effective local exposure by a factor of 10 in a clear field region and by more than a factor of 40 in a dark field region. This problem is compounded by a secondary reflection of energy at the resist/air interface back down into resist features. The dominant effect of substrate grains was to produce specular reflections which became significant for grains with side length on the order of $\lambda_{resist}/2$ or larger.

Simulation of different methods to reduce reflective notching revealed that increasing resist contrast helps but cannot ensure proper resist feature retention. Increasing resist absorption by adding dye appeared to be a little more effective. By far, the most effective method of reducing reflective notching was found to be decreasing substrate reflectivity. In general, a reflection coefficient of the substrate seen from within the resist which is less than or equal to that of silicon (0.48) should be sufficient to prevent reflective notching. These rules-of-thumb are based on 2-D simulations and with careful interpretation in scaling on the basis of area and reflectivity, should provide some guidance for dealing with 3-D effects in practice. The results presented in this study are very encouraging and indicate that it is worthwhile to pursue generalizing the massively parallel time-domain approach.

Special thanks is extended to John Frank of SEMATECH for discussions on observations of reflective notching and for providing the SEM photographs in Figure 1. Thanks also to Richard Ferguson for modifying the SAMPLE program to handle a resist development rate matrix from TEMPEST, which involves nonplanar topography.

References

1. D. W. Widman and H. Binder, "Linewidth Variations in Photoresist Patterns on Profiled Surfaces," *IEEE Trans. Elec. Dev.*, vol. ED-22, no. 7, pp. 467-471, July 1975.
2. J. C. Guibert, M. Chevalier, D. Leti and P. Paniez, "An Improved Bilayer PMGI(1) Lithographic Process," *Microelectronic Engineering*, vol. 6, pp. pp. 487-494, 1987.
3. J. C. Housley and D. J. Williams, "Dyes in Photoresists: Today's View," *Semiconductor International*, pp. 142-144, April 1988.
4. M. P. Karnett and M. C. Sarnoff, "Optimization and Characterization of Single Layer Resist," *Proc. SPIE*, vol. 1088, pp. 324-338, 1989.
5. L. K. White, "Film Curvature Effects for Optical Resists Patterned on Topography," *Proc. SPIE*, vol. 1086, pp. 535-542, 1989.
6. J. A. Bruce, R. K. Leidy, M. S. Hibbs and B. J. Lin, "Characterization and Prediction of Linewidth Variation Due to Thin Film Interference Effects," *Proc. KTI Micro. Sem.*, pp. 1-13, 1989.
7. C. Nö Ischer, L. Mader and M. Schneegans, "High Contrast Single Layer Resists and Antireflection Layers - An Alternative to Multilayer Resist Techniques," *Proc. SPIE*, vol. 1086, pp. 242-250, 1989.
8. John Frank, SEMATECH, private communications.
9. G. Wojcik, D. Vaughn and L. Galbraith, "Calculation of Light Scatter from Structures on Silicon Surfaces," *Proc. SPIE: Lasers in Microlithography*, vol. 772., pp. 21-31, 1987.
10. T. Matsuzawa, A. Moniwa, N. Hasegawa and H. Sunami, "Two-dimensional Simulation of Photolithography on Reflective Stepped Substrate," *IEEE Trans. CAD*, vol. 6, no. 3, pp. 446-451, 1987.

11. H. Urbach and D. Bernard, "Modeling Latent-image Formation in Photolithography, Using the Helmholtz Equation," *J. Opt. Soc. Amer.*, vol. 6, no. 9, pp. 1343-1356, Sept 1989.
12. J. Gamelin, R. Guerrieri and A.R. Neureuther, "Exploration of Scattering from Topography with Massively Parallel Computers," *J. Vac. Sci. Technol. B.*, vol. 7, no. 6, pp. 1984-1990, Nov/Dec 1989.

Table 1: Photoresist simulation parameters for exposure of KTI-820 positive photoresist and development using KTI-932 developer

KTI-820		KTI-932	
A	0.51	R_1	0.1143
B	0.031	R_2	0.001683
C	0.013	R_3	4.667
		R_4	0.10
		R_5	0.45
		R_6	0.30
Resist thickness		1.0 μm	
Refractive Index		1.68	

Table 2: Optical imaging parameters

Wavelength	0.436 μm
Numerical Aperture	0.6
Coherence factor	0.7
Defocus	0.0 μm
Exposure Dose	50 mJ/cm^2
Mask Pattern	1.25 μm line (isolated)

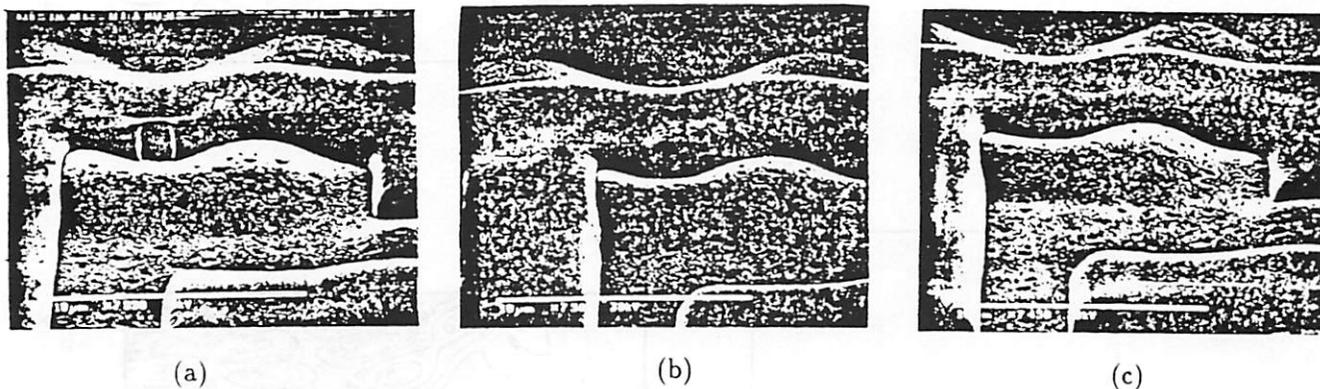


Figure 1: SEM photographs of notching of positive photoresist due to specular reflections from a bowl-shaped region in an underlying Al-Si-Cu layer for (a) resist #1, (b) resist #2 and (c) resist #1 with dye.

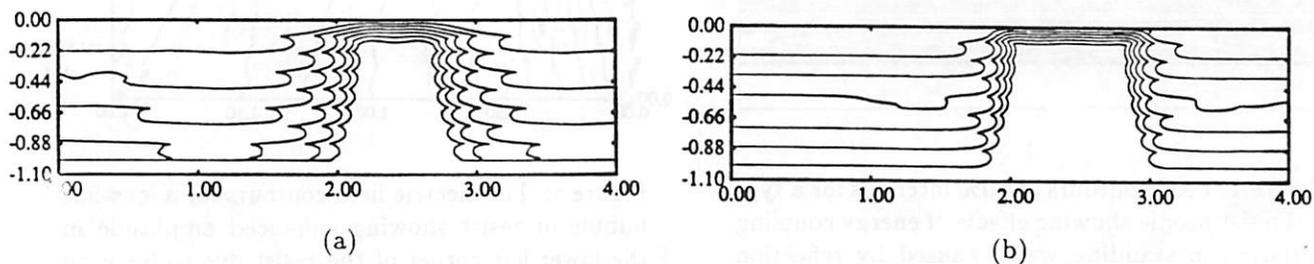


Figure 2: Development profiles of a 1.25 μm line in 1.0 μm of KTI-820 resist on top of a flat perfect conductor for simulation by (a) TEMPEST and (b) SAMPLE. Development times are 20, 40, 60, 80, 100, and 120 seconds. Detailed simulation parameters are given in Tables 1 and 2.

Figure 3: PAC contours at 20% intervals for comparison with Figure 12 of Urbach and Bernard [10]. The topography is a $0.3\mu\text{m}$ high perfectly conducting bump, $0.48\mu\text{m}$ wide at top (this is a direct measurement from their Figure 12 which differs from their quoted value of $0.6\mu\text{m}$), $0.72\mu\text{m}$ wide at bottom centered in $1.2\mu\text{m}$ periodic domain. Resist is $1.2\mu\text{m}$ thick, $n'=1.68$, $A=0.54\mu\text{m}^{-1}$, $B=0.06\mu\text{m}^{-1}$, and $C=0.014\text{cm}^2\text{sec/mJ}$. Exposed using $\lambda=0.4358\mu\text{m}$, $\text{NA}=0.6$, and $\sigma=0.5$, focal plane at $0.1\mu\text{m}$ above resist surface and a dose of 74.3mJ/cm^2 split into 5 bleaching steps. Total computation time running on 32,000 nodes took about 9 minutes.

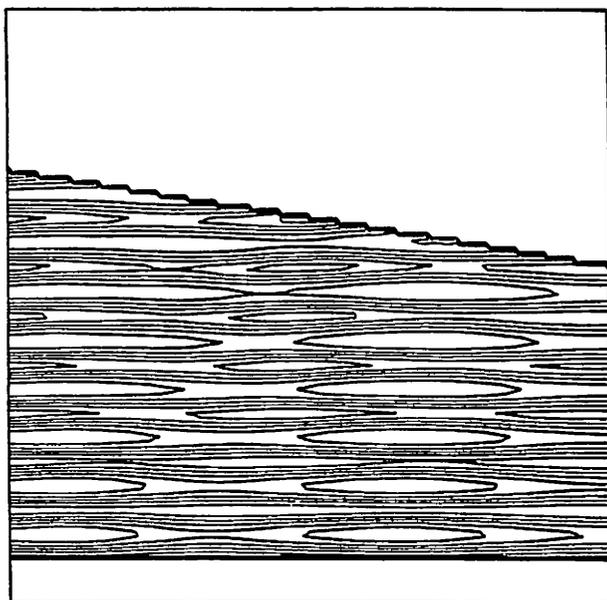
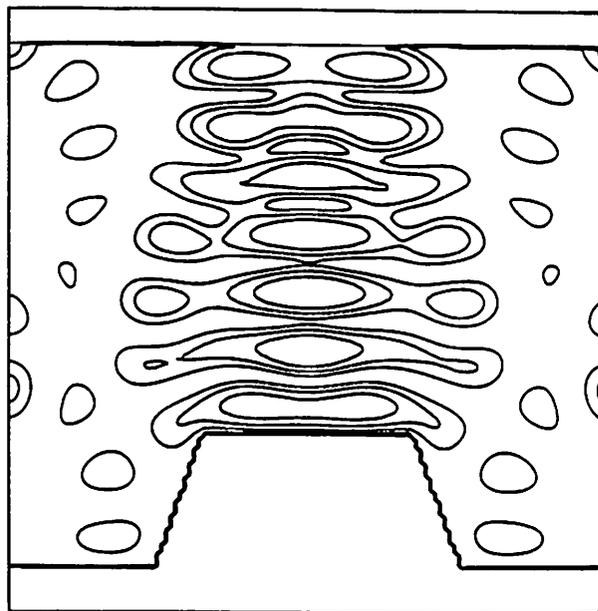


Figure 4: PAC contours at 10% intervals for a typical resist profile showing effects of energy coupling variation in standing waves caused by reflection from the silicon substrate ($n=4.73$, $k=-0.136$). Resist varies linearly in thickness from $4\lambda_{\text{resist}}$ to $3\lambda_{\text{resist}}$.

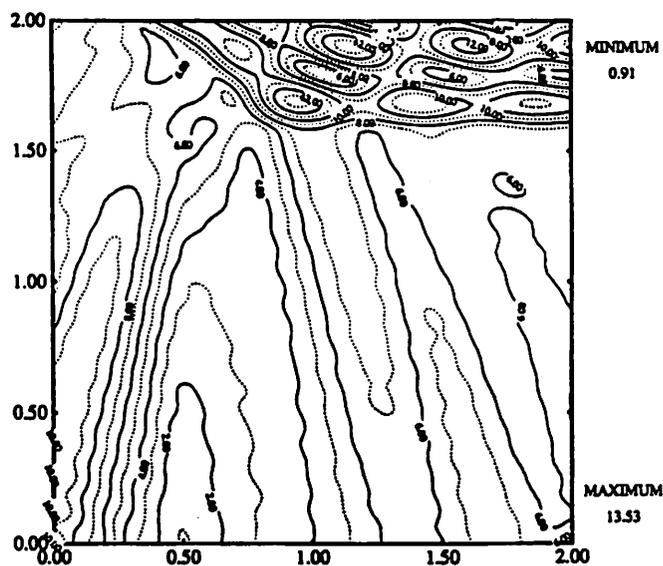


Figure 5: The electric field contours for a lens-like bubble of resist showing enhanced amplitude in the lower left corner of the resist due to focusing of the incident radiation by the resist surface. The substrate is matched to prevent standing waves.

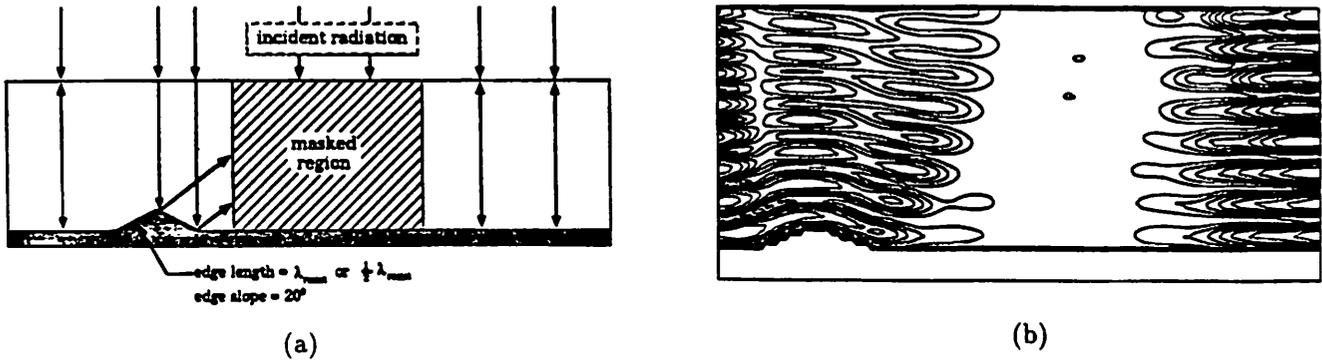


Figure 6: (a) Topography used in simulating specular reflection from substrate grain structures into nearby resist features. (b) PAC contours after exposure with grain side length = λ_{resist} .

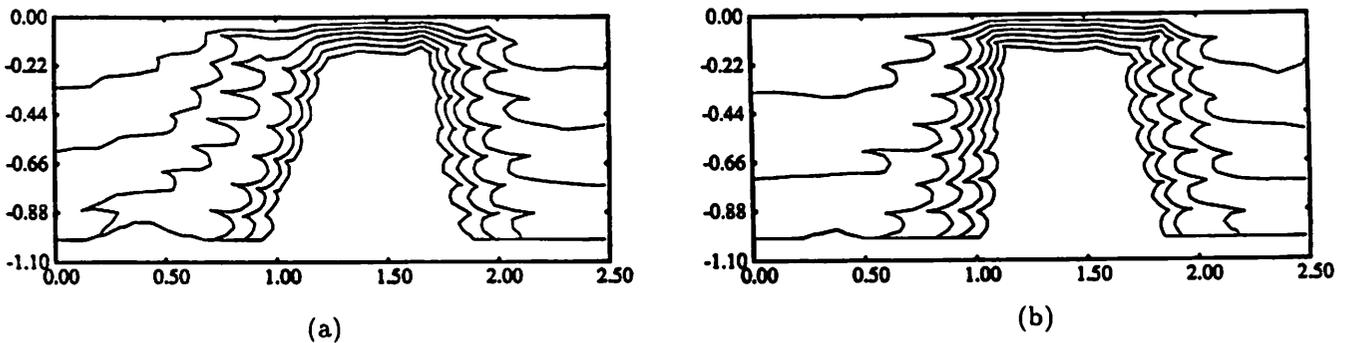


Figure 7: Developed resist profiles showing asymmetries caused by specular reflection from substrate grains with (a) side length = λ_{resist} and (b) side length = $\lambda_{resist}/2$.

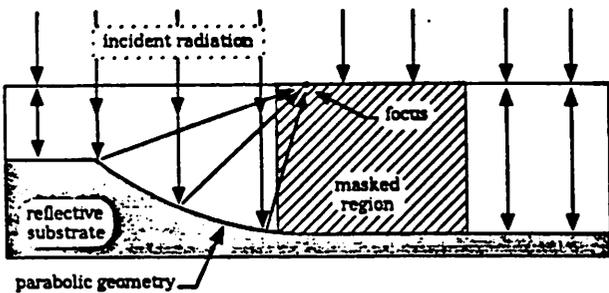


Figure 8: Simulation topography showing worst-case situation of a perfectly conducting parabolic substrate with focus point at $0.2 \mu\text{m}$ inside nominal left edge of the masked region.

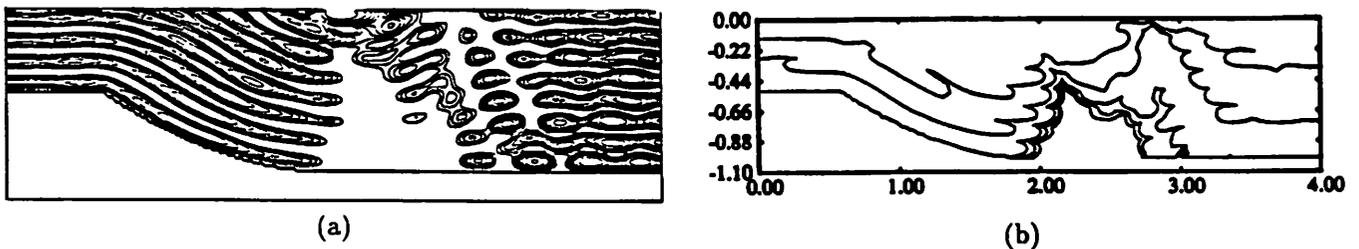


Figure 9: Reflective notching, caused by internal reflection at resist/air interface, from worst-case topography: (a) Post-exposure PAC concentration contours, (b) Development profiles at 20, 40, 60, 70, and 80 seconds.

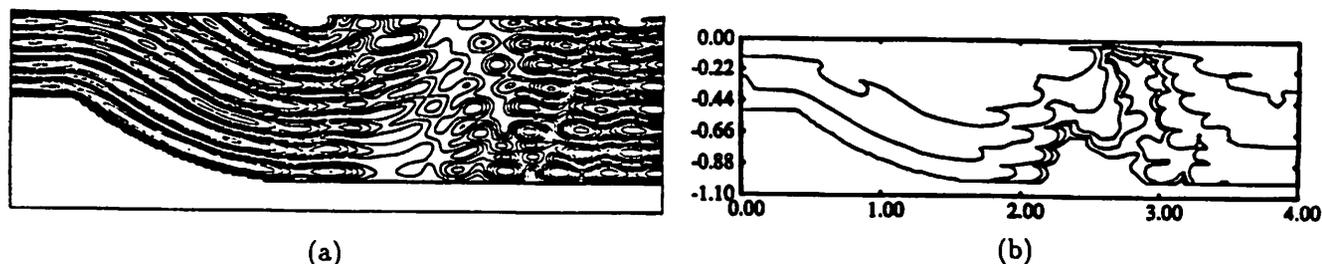


Figure 10: Same perfectly conducting parabolic substrate as in Figure 9, but with mask shifted so that focus point is $0.1 \mu\text{m}$ outside nominal left edge. (a) Post-exposure PAC concentration contours, (b) Development profiles at 20, 40, 60, 70, 80 seconds.

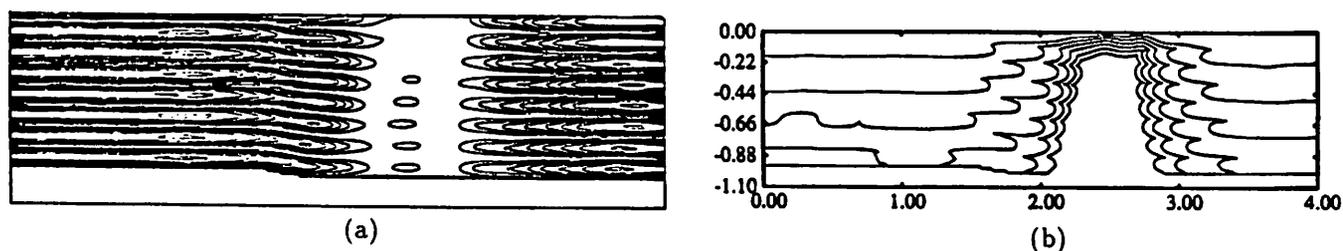


Figure 11: Reflective notching for parabolic substrate with $1/3$ the collection area of Figure 8: (a) Post-exposure PAC concentration contours, (b) Development profiles at 20, 40, 60, 80, 100, and 120 seconds.

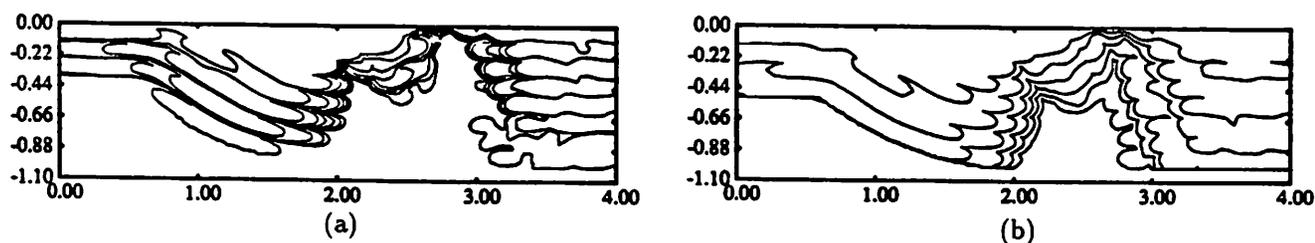


Figure 12: Reduction of reflective notching in Figure 9: (a) increased resist contrast (by decreasing development rate parameter R_2 to 0.0001638) and (b) increased resist absorption (by increasing exposure parameter B to $0.31 \mu\text{m}^{-1}$).

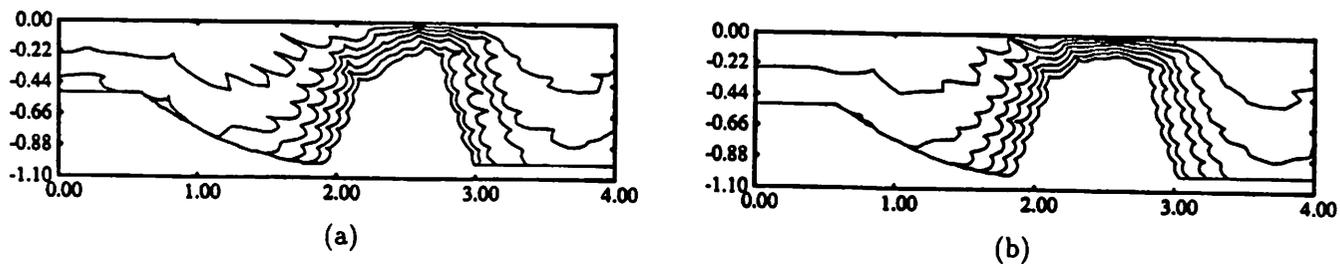


Figure 13: Reduction of reflective notching in Figure 9 through the use of substrate material with lower reflectivity (a) $\Gamma = 0.48$, (refractive index: $n=4.73$, $k=-0.136$, silicon at g-line) and (b) $\Gamma = 0.25$, (refractive index: $n=2.8$, $k=-0.02$).

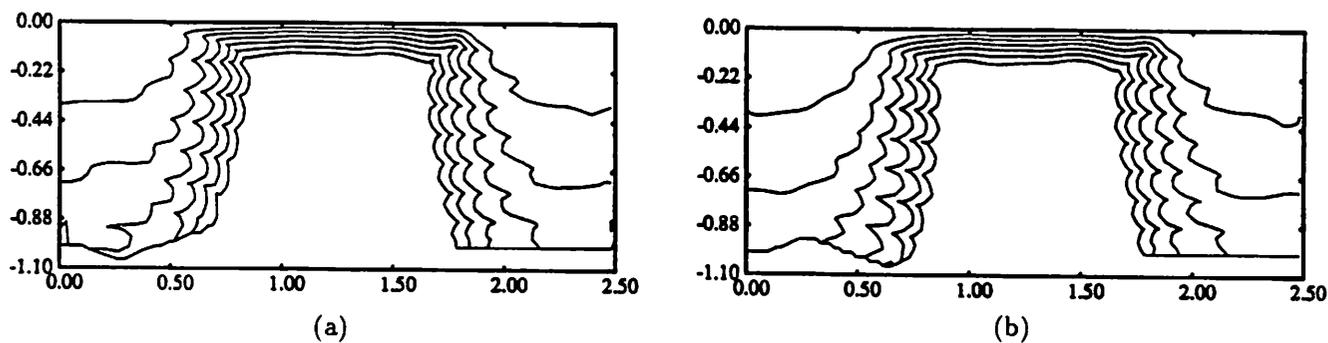


Figure 14: Reduction of specular reflection from grains by using a silicon substrate ($n=4.73$, $k=-0.136$). Development profiles for nominal left line edge falling (a) on grain peak and (b) in grain valley. Profiles correspond to 20, 40, 60, 80, 100, and 120 seconds.

Chapter 4

Linewidth Metrology of Polysilicon Gates[†]

4.1. Introduction

Maintaining the critical dimension of a polysilicon gate is of obvious importance in integrated circuit manufacturing since the dimension of the transistor gate is a primary component in the functionality of any logic device. Yet the optical image profile of polysilicon on oxide structures is extremely sensitive to the thickness and refractive index of the polysilicon and oxide layers. The relatively high refractive index of the polysilicon is a primary factor in the formation of sharp standing waves in the polysilicon layer. Variations in either the thickness or refractive index of the polysilicon layer can cause significantly different image profiles to be observed in linewidth measurements on nominally similar samples. These effects are particularly noticeable when narrow band and highly coherent illumination are used such as in the type of optical microscope suggested by the National Bureau of Standards for performing metrology tasks.¹ Alternative optical metrology schemes have been developed such as confocal scanning, differential interference contrast and differential phase contrast.² These schemes introduce more averaging over wavelengths and angles but are still limited by the same phenomena of plane wave interaction with topographical features which is not well understood.

Computer modeling is playing an increasingly important role in understanding the important factors in optical imaging and metrology. Early modeling of optical imaging, based on scalar wave theory, was limited to simulating highly coherent illumination of structures not thicker than one fourth the illuminating wavelength.³ .4 .5 Nyssonen¹ presented a more general method based on a modified form of Burkhardt's⁶ .7 vector waveguide theory that takes into account the scattering caused by multiple

[†] This chapter is extracted from the author's published paper "Understanding Metrology of Polysilicon Gates Through Reflectance Measurements and Simulation", *Proc. SPIE: Integrated Circuit Metrology Inspection and Process Control V*, March 1991.

reflections in the layered media comprising the line structure. Yuan⁸ has modeled some of the more recent schemes of optical metrology using the same modified waveguide theory and extended it to include both TE and TM polarizations. Very rigorous approaches in frequency domain and time-domain have been introduced by Wojcik.⁹ At Berkeley, we have been using a time-domain approach specially formulated for massively parallel computers in a program called TEMPEST.¹⁰

In this study the rigorous electromagnetic scattering simulator, TEMPEST, is used to explore the role of material properties and geometrical parameters in optical measurements of optically thick films such as in polysilicon gate structures. First, the problems encountered in comparing simulated images with results from an initial experiment are briefly described. It is then shown how accurate knowledge of layer thicknesses (polysilicon and oxide) and their complex index of refraction are key factors in calculating the correct image profile. The issues of simulation convergence, and off-axis illumination estimates are then addressed. Finally, a careful comparison of simulated and experimental images is made.

4.2. Overview of Experiment and Issues

A standard process was used to fabricate an initial set of wafers with structures similar to polysilicon gates. The process was designed to produce nominally 1.2 μm wide patterned gates in 450 nm thick polysilicon deposited on 37 nm of thermally grown oxide. The oxide thickness measured using a Nanometrics Nanospec proved to be uniform across the wafer and from wafer-to-wafer. Chemical vapor deposition was used to deposit the degenerately doped n-type polysilicon. Thickness measurements using the Nanospec showed poor polysilicon thickness uniformity. Measurements ranged from 405 nm to 465 nm. The remaining processing involved resist patterning and plasma etching of the polysilicon. SEM photographs of the line cross section indicated that the plasma etching resulted in nearly vertical polysilicon side-wall angles with an overetch bias of about 0.1 μm on each edge. The SEMs also showed that small cones of polysilicon "grass" remained. A

cross section of the etched wafers is drawn in Figure 1(a) and 1(b) with grass.

Reflected intensity versus position was then measured using a Nanometrics Nanoline linewidth measurement system in a specialized mode. The Nanoline was modified by placing a 550 nm filter with 10 nm bandwidth in front of the tungsten light source, restricting the illumination to the smallest aperture stop (about a 15° cone angle), and recording the analog intensity while scanning across the structures. When operated in this mode, the Nanoline is similar to the NBS-type microscope described by Nyysönen.¹ The absolute reflectivity levels for large areas were then calibrated by measurement on a Nanospec. Simulations were carried out using the cross section shown in Figures 1(a) and 1(c) for a line/space array and an isolated step respectively. Figures 2(a) and 2(b) show simulated image profiles overlaying the measured profiles of polysilicon to oxide steps on two different wafers, labeled 11A and 3A. A similar comparison of the optical images was made with measurements of a periodic array of nominally equal lines and spaces and is shown in Figure 3(a) and 3(b).

This initial comparison of simulation and experiment was encouraging. Wafer 3A with a large phase shift in reflectivity near the step from polysilicon to oxide regions showed a larger dip than wafer 11A in both simulation and experiment. The array results for wafer 11A also showed quite good agreement. The low oxide reflectivity for wafer 3A was attributed to polysilicon “grass” and adding this grass to simulation showed lower oxide reflectivity in Figure 3(b), curve “c”.

Studying the results in detail, however, pointed out the need to very carefully check several issues in both experiment and simulation and resulted in a second experiment and simulation study. The issues which needed to be addressed included eliminating the polysilicon “grass” and oxide punch-through during plasma etching. Careful measurement of the polysilicon reflectivity versus wavelength in the area of the test structure was also deemed necessary to determine a consistent set of values for polysilicon thickness, refractive index and extinction coefficient. A study of the effect of polarization and focus as well as more detailed image measurement and computer overlay of results

were also needed. The accuracy of TEMPEST simulations for high refractive index layers capable of supporting source-free solutions of Maxwell's equations also need checking. Finally, image post-processing to account for off-axis illumination and to reduce ringing in the simulated image was also needed. The approaches taken to address these key issues will be described in the following sections.

4.3. Thin Film Characterization

A new set of wafers was fabricated using the same mask as before with the intention of improving the polysilicon edge profile and thin-film quality. The over etching of polysilicon was reduced to prevent substrate attack and a final buffered hydrofluoric acid dip was added to remove residue from the plasma etching. As in the initial wafer set, the plasma etch resulted in an overetch with a 0.1 μm bias on each line edge. Experience from the first simulations indicated that much greater attention was needed in determining the polysilicon and oxide layer thicknesses and the complex refractive index of the polysilicon. The initial oxide thickness was measured to be 37 nm and very uniform. The polysilicon extinction coefficient and thickness were estimated from reflectance measurements made with a Nanometrics Nanospec/DUV microspectrophotometer which provided values of relative reflectance versus wavelength over the range of 200 to 900 nm. As seen in Figure 4, the reflectivity for the polysilicon/oxide stack is cyclic. The minima in the data correspond to stack thicknesses that are multiples of a half wavelength. For wavelengths shorter than about 400 nm, the absorption in polysilicon becomes dominant and the reflectivity fluctuations dampen. The small rise then fall of the reflectivity in the wavelength region between 200 and 400 nm is a result of the reflectivity values being referenced to aluminum.

Two wafers, labeled 1A and 2A, with different polysilicon thicknesses were examined in detail. Due to the plasma etching and the HF dip, the oxide thickness in the cleared regions was 8 nm for wafer 1A and 15 nm for wafer 2A. This indicates a moderate amount of oxide etch and explains the relatively high oxide reflectivity observed when making post-processing reflectivity measurements with

the spectrophotometer. The locations of the local minima and the taper in the oscillation were used to accurately estimate the thickness and extinction coefficient of the polysilicon. According to Jellison and Modine,¹¹ the refractive index of silicon at the microscope's operating wavelength (546.1 nm) is $n = 4.093$, $k = -0.031$. The oxide refractive index was assumed to be 1.46 based on a handbook of optical constants¹² and varied by only 0.5% over a 100 nm range surrounding the measurement wavelength. It was then assumed that the real part of the refractive index for polysilicon was the same as that for silicon. Smith charts for the oxide and polysilicon were then used to find the theoretical reflection coefficient in the polysilicon. Comparison with the relative phase of maxima and minima in the reflectivity data gave the polysilicon thickness. Using this technique, polysilicon thicknesses of 484 nm and 476 nm were computed for wafers 1A and 2A, respectively. A Tencor Instruments Alphastep profilometer was also used to check the approximate polysilicon thickness taking into account the slight oxide overetch.

The polysilicon extinction coefficient was estimated by comparing the maximum reflectivity measured near the operating wavelength with the theoretical maximum assuming no attenuation. By knowing this ratio and the thickness of the polysilicon (z), a value for the extinction coefficient can be computed using

$$\frac{R_{measured}}{R_{theory}} = e^{-2 \left[\frac{2\pi}{\lambda} \right] kz}$$

This gave an extinction coefficient $k = -0.043$. Some error is introduced into this technique by assuming that the oxide/silicon phase shift is independent of wavelength and by using the silicon refractive index for polysilicon. But these approximations result in only a few percent error.

4.4. Optical Image Measurement Using an NBS System

Optical images of the polysilicon gate structures were made using the well characterized bright-field, incident-light, scanning slit, optical microscope at VLSI Standards, Inc. in Mountain View,

California.¹³ This system consists of a monochromatic illumination source operating at 546.1 nm (mercury green line) passing through a 19 nm bandwidth filter and provides a numerical aperture of 0.20 for the illumination optics and 0.95 for the collection optics. *Best focus* was determined in a manner similar to the method for thin film structures which consists of finding steepest slope at the line edge in the image profile.¹⁴ Though this method is not very accurate when used for thick films, it was not critical to focus exactly at the top of the polysilicon line since the image synthesis algorithm could vary focus offset until a best match was found. Measurements were made on the same die used to collect the reflectivity versus wavelength data.

As discussed in section 4.5, the simulations were performed using TE polarized illumination. To investigate the effect that polarized illumination had on the image profile, optical measurements were made using unpolarized and linearly polarized light with the electric field parallel to the line edge (TE polarization). Figure 5 shows measured image intensity profiles with and without TE polarized illumination. Note that when using TE polarized illumination, the image profile showed slightly larger maxima and minima compared to the unpolarized case. Since the optical microscope is normally operated without a polarizer, the experimental data reported throughout the remainder of this study was collected using unpolarized illumination. Thus the simulated images based on TE polarization will have slightly enhanced maxima and minima.

4.5. Simulations

TEMPEST was used to simulate the electromagnetic scattering from the nominally 1.2 μm wide polysilicon gate structures using 128K processors allocated in a spatially uniform grid dimensioned as 1024 \times 128 nodes (each processor representing one grid node). This grid was mapped directly onto an 8 μm wide by 1 μm high periodic simulation domain as shown in Figure 6. The system was excited along the top boundary of the simulation domain with a single, normally incident plane wave with the electric field vector parallel to the line edge. The excitation involved stepping the sinusoidal plane

wave in *time* for about 10 full cycles (one cycle corresponds to one wavelength) while computing the electric and magnetic fields at each of the 128K nodes at every time step. A check for steady-state convergence is then performed at the end of each succeeding time cycle until convergence is reached. The CPU time to reach steady-state depends upon the multiple reflections of waves within the structure being simulated as well as the density of the grid nodes.¹⁵ For simulations on a CM-2 with 1K real processors using a virtual processor ratio (*VP*) of 128, the time to reach steady-state was typically 650 seconds for these polysilicon gate structures. Upon reaching convergence, the scattered electric field along the top boundary was used to compute the magnitude and phase of the scattered energy in each of the propagating harmonics. These “diffraction efficiencies” were then used by a post-processing program on a computer workstation to calculate the optical image profile.

Because of the uniform grid and point by point sampling of the material properties used in the current version of TEMPEST, discretization limits the layer dimensions to be integer multiples of the smallest spatial unit, dx . For these simulations, dx was $0.0078125 \mu\text{m} = 8 \mu\text{m}/1024$ grid nodes. This value worked out well for the material thicknesses measured on the wafers. The oxide under the polysilicon was modeled using 5 grid layers to achieve 39.0625 nm (measured as 37 nm). The regions of oxide not covered by polysilicon were modeled using 1 grid layer for wafer 1A (measured as 8 nm) and 2 grid layers for wafer 2A (measured as 15 nm).

Because TEMPEST computes the electromagnetic scattering in the time domain, it provides the benefit of obtaining a picture of the instantaneous electric field and the standing waves at any given time. The complicated scattering caused by the polysilicon gate structure can be viewed in 2-dimensions by plotting the electric field everywhere in the simulation domain. Figure 7 shows that the electric field in the polysilicon has lateral as well as vertical standing waves indicating that the edges of the polysilicon line contribute significant scattering. The standing wave pattern along a vertical cut line in a planar polysilicon / oxide / silicon stack is shown as curve “a” in Figure 8 and the instantaneous electric field is shown as curve “b”. The dashed vertical lines indicate the boundaries

between the layers of air, polysilicon, oxide and the silicon substrate.

The relatively low grid density in the polysilicon ($17 \text{ nodes} / \lambda_{poly}$) results in a sampling error which clips most of the sharp minima in the standing wave pattern. This sampling error results in a small vertical and horizontal shift in the curve of reflectivity versus polysilicon thickness as shown in Figure 9 which compares reflectivity values computed by TEMPEST and values computed analytically using REFLOP,¹⁶ a computer solution of the Fresnel equations. The error increases with increasing refractive index and appears to be related to the accuracy in describing the sharp minima caused by the large reflection coefficient in the polysilicon. A high density of nodes per wavelength in polysilicon (roughly 70 or more) is needed to obtain the precise reflectivity expected from the polysilicon / oxide / silicon stack. To save computer time, the polysilicon refractive index was adjusted by 1%, from 4.093 to 4.14, to tune the experimental thickness-refractive index combination and to operate with $17 \text{ nodes} / \lambda_{poly}$.

4.6. Image Post Processing

An estimate of the optical image with Kohler illumination can be made by assuming that the diffraction efficiencies are invariant with angle of incidence. This simplification is the key to the interchange of the order of integration in Hopkin's formulation for imaging with partially coherent illumination.¹⁷ As in Hopkin's approach, this assumption allows the image to be calculated by weighting each diffracted order by overlapping circles in k -space. A post-processing program for weighting the diffraction efficiencies output by TEMPEST in this manner is described by Wong.¹⁸ Our specific knowledge from TEMPEST in the current version is limited to propagating diffracted orders with k vectors parallel to the surface less than $k_o = 2\pi / \lambda$ corresponding to an NA of unity. Since for our application the sum of the illumination and collection numerical apertures of the optical microscope is $0.2 + 0.95 = 1.15$, a value of 0.20 was used for $NA_{illumination}$ and 0.73 for $NA_{collection}$. These values gave a reasonable compromise between broadening of the width of the main null and the

number of ripples in the optical image.

4.7. Comparisons of Simulation and Experiment

4.7.1. Line Edge

Figure 10 compares the image profiles of an oxide to polysilicon step obtained from simulation and measurement for wafers 1A and 2A. The scale is based on absolute reflectivity. Due to the grains in the polysilicon, small oscillations in its reflectivity distinguish it from the oxide. Note that for wafer 1A the polysilicon reflectivity is greater than that of the oxide while for wafer 2A the opposite is observed due to the difference in polysilicon thickness. In regions well removed from the step, the TEMPEST image matches the correct theoretical reflectivity. The width and ringing of the synthesized image have been balanced by adjusting $NA_{illumination}$ and $NA_{collection}$ as described in section 4.6. The width of the lobes is inversely proportional to $NA_{collection}$ while the amount of ringing decreases as $NA_{illumination}$ increases. From experience, the grazing angle diffracted orders appeared irregular so the sum of $NA_{illumination}$ (0.20) and $NA_{collection}$ (0.73) was limited to 0.93 rather than 1.0.

Compared with the initial simulations of isolated line edges (section 4.2), a considerable improvement is apparent. The agreement is surprisingly good given that only the TE polarization was used and that data from only visible angles from normal incidence was included in synthesizing the image profile. As diffraction efficiencies for evanescent waves, off-axis illumination and the TM polarization become available from TEMPEST, further improvements in image quality are likely.

4.7.2. Line Width

Results of optical measurements of an isolated 1.2 μm line from both wafers are given in Figure 11. The slight variation in polysilicon thickness of about 8 nm results in significantly different optical image profiles. The image behavior on each side of the line edge is nearly identical to the image of an

isolated edge for both wafers. The location of the simulated line edge is indicated by the vertical lines in the figures. It is interesting to note that when the magnitude and phase of the reflection from the polysilicon change, the part of the image corresponding to the correct polysilicon linewidth also changes. In the case of wafer 2A, the simulations of the 1.2 μm line showed a peak in the middle of the line larger than the oxide reflectivity. By reducing the polysilicon thickness from the originally simulated 476 nm to 468 nm, the polysilicon reflectivity was reduced and a center peak resembling that of the measured image was produced. This lower polysilicon reflectivity appears to be due to an actual change in local polysilicon thickness on the wafer which might be expected from the rougher grain structure found on wafer 2A compared to wafer 1A. The superimposed simulated images correspond to linewidths of 1.055 μm and 1.0 μm for wafers 1A and 2A respectively. SEM photographs confirm that the etching process typically undersized the polysilicon but these wafers have not been cross-sectioned pending further optical measurements.

4.7.3. Focus

One of the problems in metrology of patterned thick-film structures is the difficulty in determining precise focal position. Accurate image simulation of such structures can give insight to desirable focus positions and aid in determining accurate linewidth measurements. Simulated images showing the effects of focus variation on the optical image profile for the 1.2 μm line on wafer 1A are shown in Figure 12(a). These images were produced by simply reprocessing the TEMPEST diffraction efficiencies from a single normally incident plane wave on a workstation. The time to calculate each image is only a few seconds and increases slightly with the number of harmonics used. As the focal plane moved up from the substrate, the center lobe in the image profile tended to decrease in intensity while the sidelobes increased. For this particular structure, the width of the center lobe remained relatively constant while the width of the sidelobes increased rapidly with focus change. Experimental results shown in Figure 12(b) agree with these simulations.

4.8. Conclusions

Optical image profiles of polysilicon gate structures measured with an NBS-type optical microscope were compared with image profiles generated using rigorous electromagnetic scattering simulation. Carefully derived values of the polysilicon thickness and the real and imaginary parts of its refractive index were necessary to provide the simulator with an accurate representation of the gate structure. Due to the sharply defined standing wave pattern that forms in the polysilicon layer, a very high grid density (roughly $70 \text{ nodes} / \lambda_{\text{poly}}$ in the polysilicon layer) was required to obtain theoretically correct reflectivities for planar layers. However, tuning the refractive index by 1% allowed a grid density of only $17 \text{ nodes} / \lambda_{\text{poly}}$ to be used. A spectral component-weighting technique used for optical image synthesis made it possible to use TEMPEST simulations of a single normally incident plane wave with only a reasonably small trade-off in edge ringing versus lobe width.

Good agreement for normally incident, TE polarized illumination was observed when comparing with measured image profiles using unpolarized illumination. Further improvements can be expected when off-axis incidence and TM polarization are included. Simulation was very useful in seeing the electric field within the structure and determining the part of the image profile that corresponds to the correct line edge. Analysis of two nominally similar polysilicon on oxide gate structures showed that only 8 nm difference in polysilicon thickness caused a significant change both in the image profiles and in the part of the image associated with the actual linewidth. It is hoped that these initial observations and simulation techniques will help to eventually establish a quantitative understanding of optical linewidth metrology.

Special thanks is extended to VLSI Standards, Inc. in Mt. View, California for use of their equipment to collect the optical image profiles. Thanks also to Nanometrics and the UC Berkeley Microfabrication Lab for helping provide and support the equipment used in this experiment. Takeshi Doi helped with the initial metrology simulations and Alfred Wong contributed the code for weighting the diffraction efficiencies used in the image calculations. Kim Chan and Anthony Pfau's assistance in gathering the reflectivity data is also appreciated.

References

1. D. Nyyssonen, "Calibration of Optical Systems for Linewidth Measurements on Wafers," *Proc. SPIE*, vol. 21, no. 5, pp. 882-887, 1982.
2. P. C. D. Hobbs, R. L. Jungerman and G. S. Kino, "A Phase Sensitive Scanning Optical Microscope," *Proc. SPIE*, vol. 565, pp. 71-80, 1985.
3. D. Nyyssonen, "Optical Linewidth Measurements on Wafers," *Proc. SPIE*, vol. 135, pp. 115-119, 1978.
4. D. Nyyssonen, "Calibration of Optical Systems for Linewidth Measurements on Wafers," *Proc. SPIE*, vol. 221, pp. 119-126, 1980.
5. E. C. Kintner, "Method for the Calibration of Partially Coherent Imagery," *Appl. Opt.*, vol. 17, pp. 2747-2753, 1978.
6. C. B. Burkhardt, "Diffraction of a Plane Wave at a Sinusoidally Stratified Dielectric Grating," *J. Opt. Soc. Am.*, vol. 56, pp. 102-1509, 1966.
7. F. G. Kasper, "Diffraction by Thick, Periodically Stratified Gratings with Complex Dielectric Constant," *J. Opt. Soc. Am.*, vol. 63, pp. 37-45, 1973.
8. C. Yuan and A. J. Strojwas, "Modeling of Optical Alignment and Metrology Schemes Used in Integrated Circuit Manufacturing," *Proc. SPIE: Optical/Laser Microlithography III*, vol. 1264, pp. 203-218, 1990.
9. G. Wojcik, D. Vaughn and L. Galbraith, "Calculation of Light Scatter from Structures on Silicon Surfaces," *Proc. SPIE: Lasers in Microlithography*, vol. 772, pp. 21-31, 1987.
10. J. Gamelin, R. Guerrieri and A.R. Neureuther, "Exploration of Scattering from Topography with Massively Parallel Computers," *J. Vac. Sci. Technol. B.*, vol. 7, no. 6, pp. 1984-1990, Nov/Dec 1989.

11. G. E. Jellison, Jr. and F. A. Modine, "Optical Constants for Silicon at 300 K and 100 K Determined from 1.64 to 4.73 eV by Ellipsometry," Technical Report No. TM-8002, Oak Ridge National Laboratory, February 1982.
12. *Handbook of Optical Constants of Solids*, Academic Press, Inc., Orlando, FL, 1985.
13. R. Monteverde and D. Nyyssonen, "A New Line Width Standard for Reflected Light Inspection," *Proc. SPIE: Integrated Circuit Metrology, Inspection, and Process Control II*, vol. 921, pp. 395-405, 1988.
14. D. Nyyssonen, "Practical Method for Edge Eetection and Focusing for Linewidth Measurements on Wafers," *Opt. Eng.*, vol. 26, no. 1, pp. 81-85, Jan 1987.
15. R. Guerrieri, K. Tadros, J. Gamelin, and A.R. Neureuther, "Massively Parallel Algorithms for Scattering in Optical Lithography," *IEEE Trans. CAD*, to appear.
16. REFLOP is a Fortran program written by W.G. Oldham (UC Berkeley) to solve the Fresnel equations and is found on the SAMPLE distribution tape
17. H. H. Hopkins, "On the Diffraction Theory of Optical Images," *Proc. Royal Soc. Series A.*, vol. 217, no. 1131, pp. 408-432, 1953.
18. A. Wong, T. Doi, D. Dunn, and A.R. Neureuther, "Experimental and Simulation Studies of Alignment Marks," *Proc. SPIE: Optical/Laser Microlithography IV*, vol. 1463, March 1991.

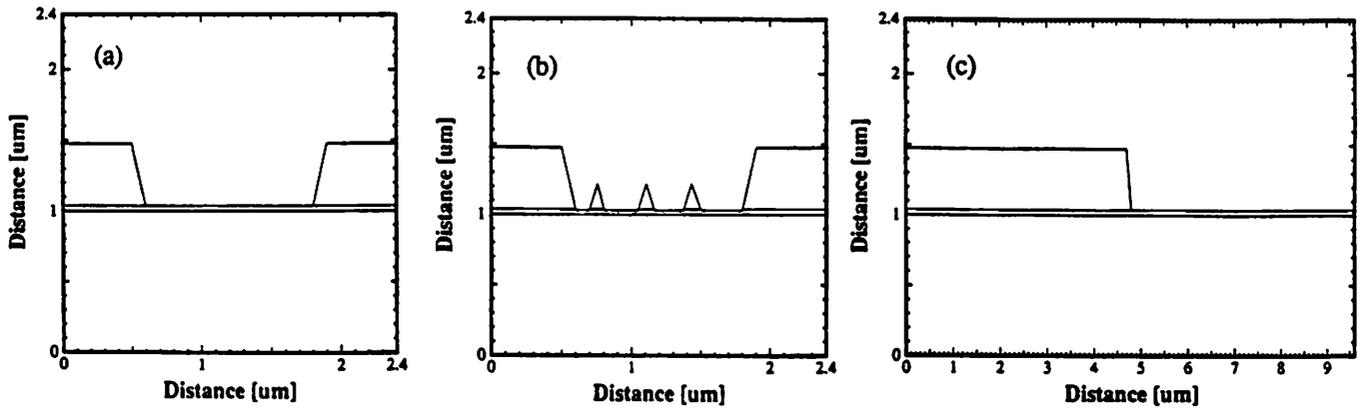


Figure 1. 2-D cross sections of polysilicon/oxide/silicon gate geometries input to *TEMPEST* to simulate electromagnetic scattering: (a) periodic line/space array, (b) line/space array with polysilicon "grass", (c) isolated polysilicon to oxide step.

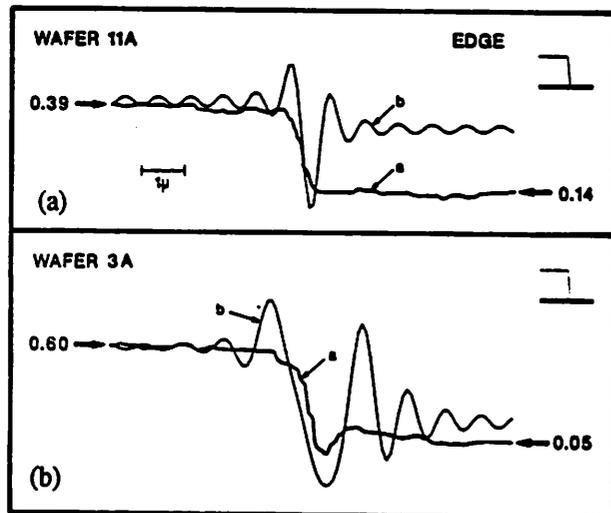


Figure 2. Measured and simulated optical image profiles from an initial study of reflected intensity versus position for a step from polysilicon to oxide as depicted in Figure 1(c) for (a) wafer 11A and (b) wafer 3A. Curve "a" is measured and curve "b" is simulated.

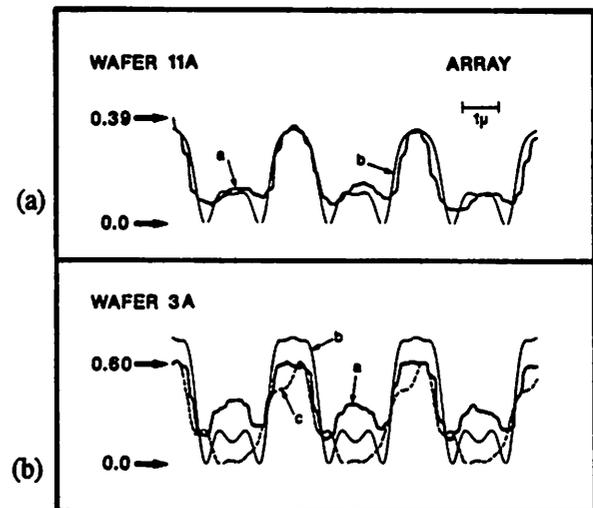


Figure 3. Measured and simulated optical image profiles from an initial study of reflected intensity versus position of an array of nominally equal lines and spaces as depicted in Figure 1(a). (a) wafer 11A, (b) wafer 3A. Curve "a" is measured and curve "b" is simulated. The simulated image profile labeled "c" in graph (b) included polysilicon grass structures as shown in Figure 1(b).

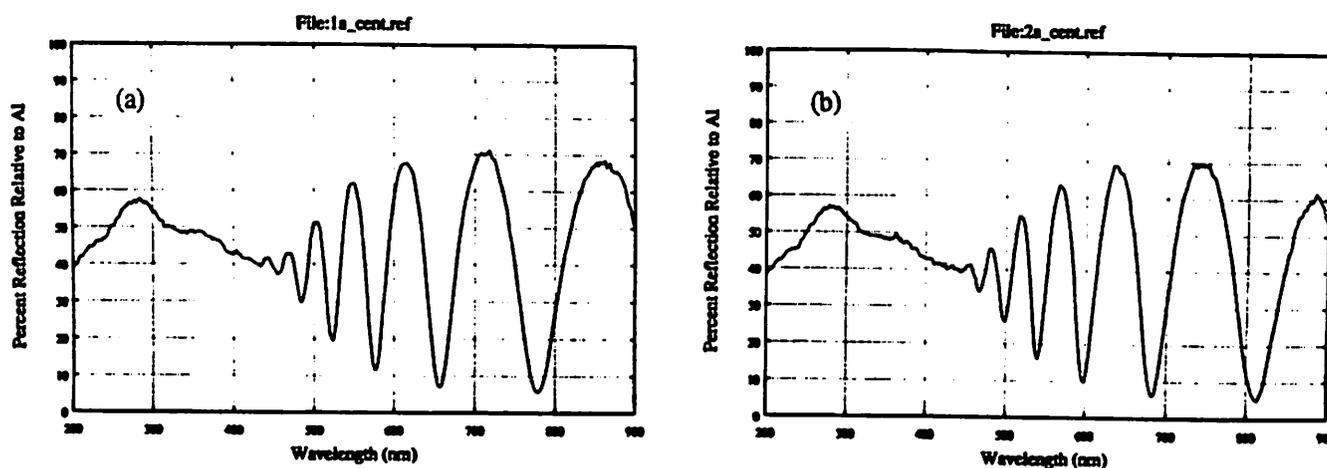


Figure 4. Relative reflectance versus wavelength of a polysilicon/oxide/silicon stack in a second study for (a) wafer 1A and (b) wafer 2A.

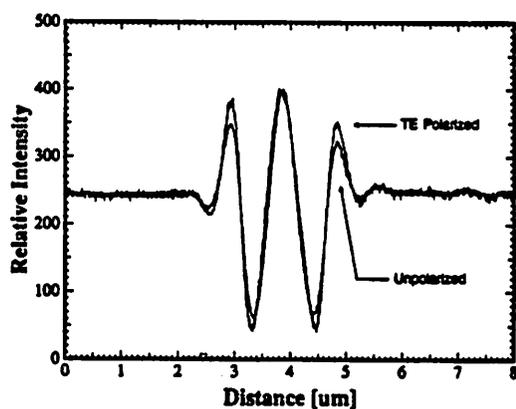


Figure 5. Measured optical image profile of a nominally $1.2 \mu\text{m}$ wide polysilicon gate on wafer 1A using unpolarized and TE polarized illumination.

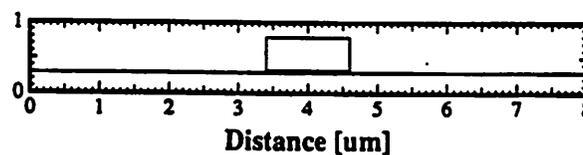


Figure 6. 2-dimensional cross section of a polysilicon gate input to *TEMPEST* for electromagnetic scattering simulation.

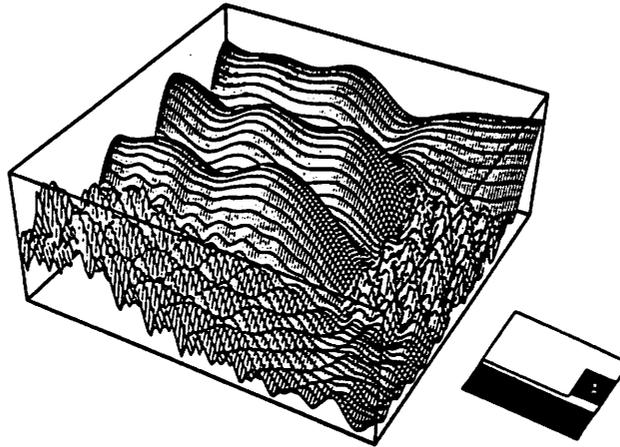


Figure 7. The maximum value of the electric field during steady state at every point in one-half of the 2-dimensional domain shown in Figure 6. The inset diagram shows the polysilicon gate structure as it relates to the orientation of the 3-D plot.

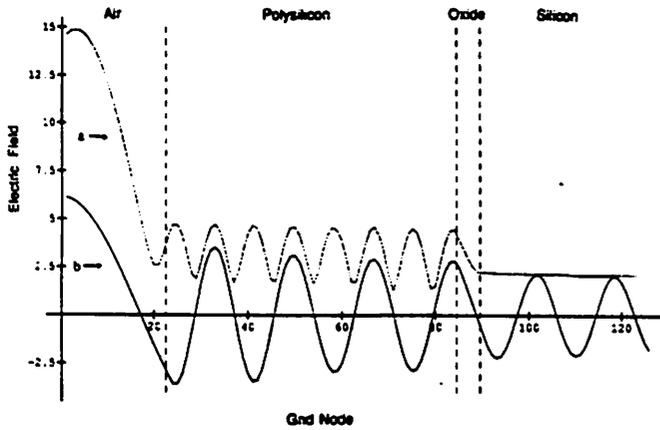


Figure 8. Vertical cutline through a polysilicon/oxide/silicon stack showing the (a) the instantaneous electric field at a particular time and (b) the maximum electric field during steady state.

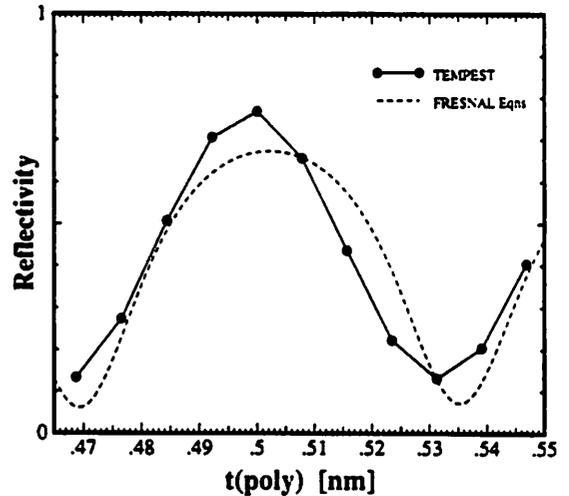


Figure 9. Reflected intensity from a polysilicon/oxide/silicon stack as the polysilicon thickness varies. Calculated using the Fresnel equations (dashed line) and TEMPEST with $17 \text{ nodes} / \lambda_{\text{poly}}$ (solid line). Parameters for oxide: thickness = 39.0625 nm, $n=1.45$; polysilicon: $n=4.14$, $k=-0.031$; silicon: $n=4.093$, $k=-0.031$.

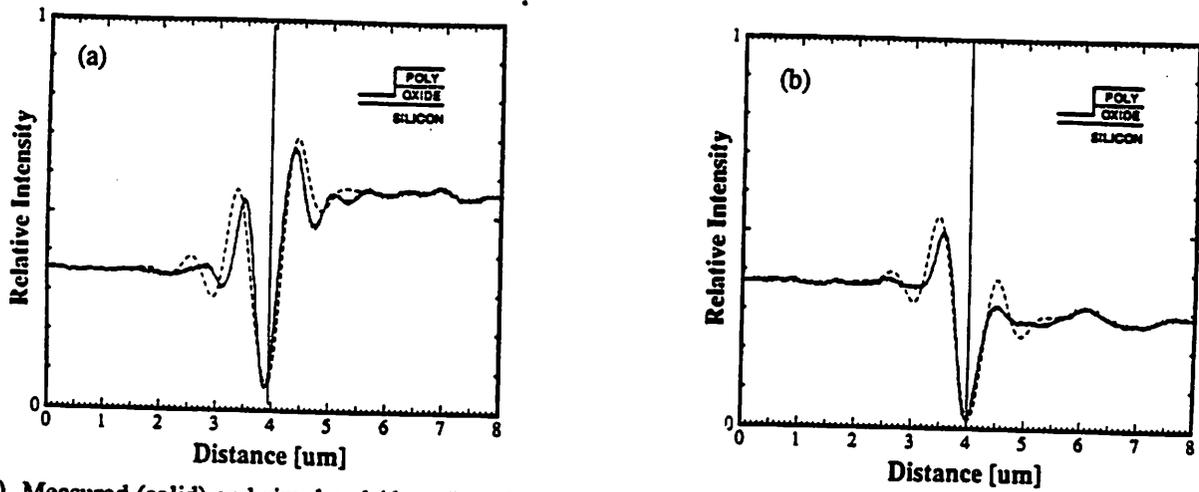


Figure 10. Measured (solid) and simulated (dotted) optical image profiles of a polysilicon to oxide step for wafers (a) 1A and (b) 2A. The solid vertical line denotes the simulated line edge.

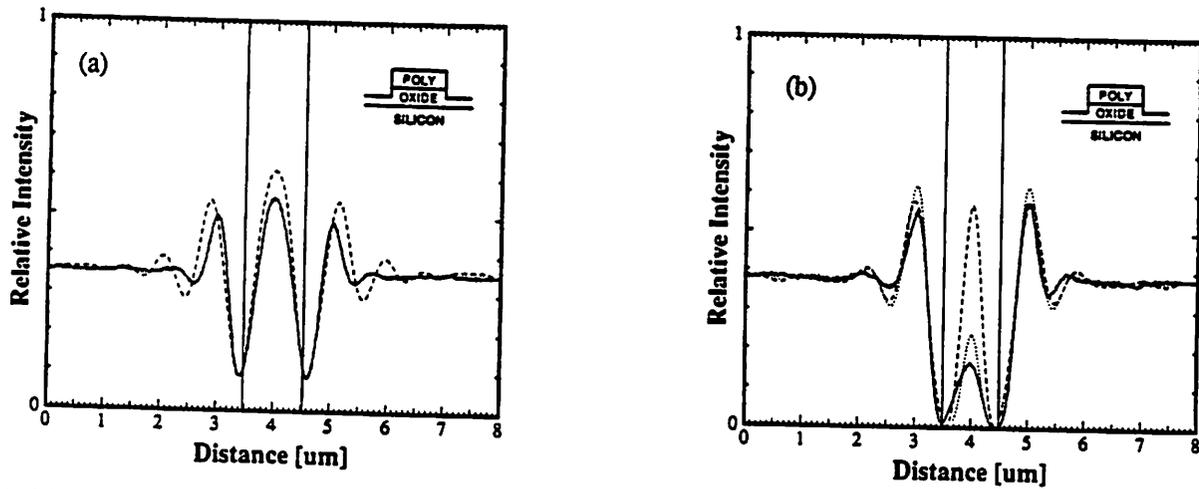


Figure 11. Measured (solid) and simulated (dotted) optical image profiles of a polysilicon gate structure for wafers (a) 1A and (b) 2A. The solid vertical lines denote the simulated lines edges

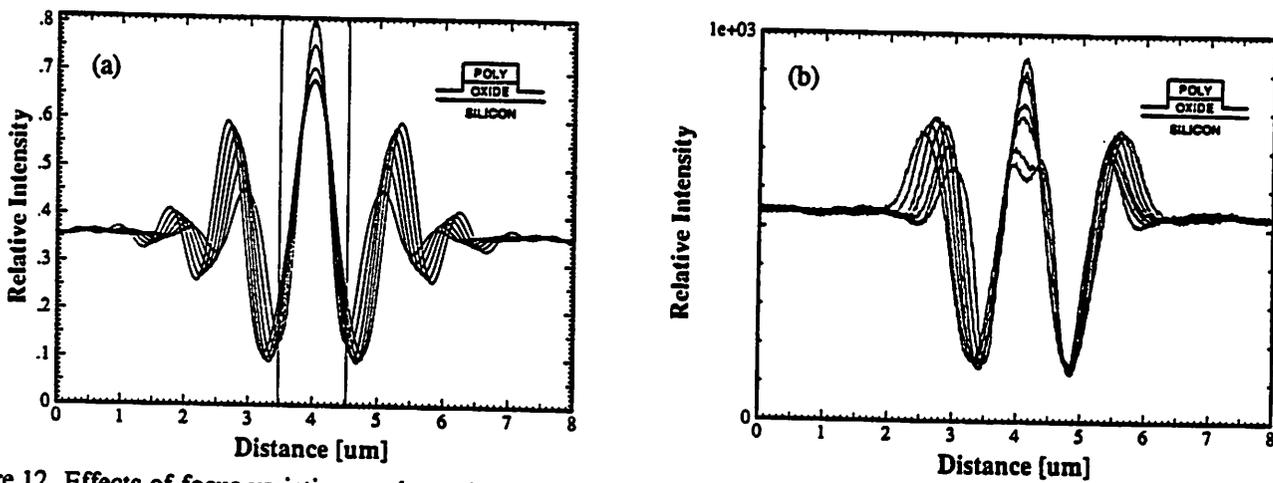


Figure 12. Effects of focus variation on the optical image profile for similar focus offsets obtained by (a) simulation and by (b) experiment.

Chapter 5

Extensions to the Algorithm[†]

Exploratory efforts to enhance TEMPEST's simulation capabilities by implementing obliquely incident illumination and symmetric boundary conditions are addressed in this chapter. Although TEMPEST simulations and experimental comparisons have shown that many optical scattering and imaging issues can be modeled reasonably well with normally incident illumination, the effects of obliquely incident rays must be well understood. A special case of simulating illumination at specific angles is presented. For symmetric structures, the resources needed for scattering analysis can be reduced since only one half of the structure is necessary to define the problem. The boundary equations needed to simulate electromagnetic scattering of structures with a vertical axis of symmetry were formulated and tested. Some of the simulation results are presented.

5.1. Oblique Incidence

Currently, TEMPEST performs electromagnetic scattering simulations using a normally incident plane wave excited with a sinusoidal forcing function along the top boundary of the 2-dimensional simulation domain. However, simulation of off-axis effects with a time-domain, finite-difference method is not trivial since the velocities of the many waves propagating at different angles across the uniform grid cannot be treated independently. This is possibly the cause for errors in synchronizing the analytically calculated forcing function with the numerically propagating field.

Working with periodic structures allows a simplification to be made. At certain angles, a wave illuminating a periodic structure will be in phase from period to period but shifted in time by some

[†] This author extends special thanks to Dr. Roberto Guerrieri of the Universita' di Bologna, Italy, one of the original developers of the TEMPEST massively-parallel algorithm, for providing numerous technical discussions on the algorithm and for initially suggesting the approach taken to implement oblique incidence.

integer multiple of wavelengths. As shown in Figure 1, these angles can be calculated using the equation

$$\theta_{inc} = \sin^{-1} \left[\frac{m\lambda}{L} \right] . \quad (5.1)$$

where m is an integer ($\dots, -2, -1, 0, 1, 2, \dots$), λ is the wavelength, and L is the period of the domain. For these specific angles, modeling is simplified since the incident electric and magnetic field amplitudes at the periodic boundaries are by nature equal but out of phase by an integer number of full wavelengths. Furthermore, since the boundaries of a 2-dimensional grid implemented on the Connection Machine are physically adjacent to each other, no modifications to the field equations along the boundaries are necessary.

In order to achieve an obliquely incident wave, the sinusoidal forcing function applied at the top boundary of the periodic domain in TEMPEST was modified so that each source node would possess an additional phase offset in its calculation of the time-harmonic electric field based on its relative distance from the left boundary. The general form of the incident electric field in TEMPEST is described by the equation

$$E_z(x, t) = A \sin(\omega t - \vec{k} \cdot \vec{r}) . \quad (5.2)$$

Since this forcing function is applied only along the top boundary of the domain, $\vec{k} \cdot \vec{r}$ is simply $k_x x_i$, where x_i represents the location of the i^{th} grid node relative to the left boundary and k_x is the propagation constant

$$k_x = \frac{2}{dx} \sin^{-1} \left[\frac{n_0 dx \sin(\frac{1}{2} \omega dt)}{c dt} \right] \cos(\theta_{inc}) \quad (5.3)$$

given in equation (45) by Gamelin¹ with the addition of the cosine term to take into account propagation at arbitrary angles. To produce an obliquely incident plane wave, the phase offset was applied as a temporal phase shift in the forcing function. This phase shift must vary from zero to mT

seconds along the top boundary. Therefore, t in equation (5.2) now becomes a function of position:

$$t' = f(x_i) = t - mT \frac{x_i dx}{L} \quad (5.4)$$

where t is the instantaneous time, T is the period of the sinusoid, and dx is the spatial discretization.

Note that changing the sign of m will change the orientation of the incident wave. The linearly graded phase shift between the left and right boundaries is considered a time advance if $m > 0$ and a time delay if $m < 0$. The special case of $m = 0$ results in the normal incidence forcing function.

Representing the wave as a time-harmonic plane wave allows the temporal and spatial components to be separated:

$$E_z(x_i, t) = A e^{i \left[\omega \left[t - mT \frac{x_i dx}{L} \right] - k_x x_i \right]} \quad (5.5)$$

$$E_z(x_i, t) = A e^{i(\omega t)} e^{i \left[-\omega mT \frac{x_i dx}{L} - k_x x_i \right]} \quad (5.6)$$

Then at each time step, the field is updated by multiplying the time increment $e^{i(\omega dt)}$ with the accumulated, time-modulated, harmonic field in each processor along the top boundary:

$$E_z(x_i, t_{accum}) = A e^{i \left[\omega \left[t_{accum} - nT \frac{x_i dx}{L} \right] - k_x x_i \right]} \quad (5.7)$$

The cost of performing this phase shift in the forcing function is an additional two parallel variables and a one-time computation of the position-dependent phase term. The actual calculations for computing the forcing function are performed using the trigonometric equivalent equations (expanded using Euler's formula).

A plot of the instantaneous electric field of an obliquely incident wave illuminating a previously unperturbed domain is shown in Figure 2. The simulation was terminated before the wave reached the bottom boundary. In this example, the phase difference between the left and right boundaries was designed to be exactly one wave cycle. There is a noticeable amplitude modulation along the supposedly constant phase fronts. Comparisons of the field values at the left and right boundaries indicated that the fields were in phase just below the top boundary, as expected. But as the wave

propagated into the domain, the difference grew larger. It is suspected that additional transients were generated from the sudden excitation of all the nodes along the top boundary, where each node was excited with an electric field value different from its neighbor. This variable starting value may introduce laterally propagating waves that need to communicate with nodes on either side, an issue not relevant for normally incident waves.

It is also possible that a numerical error is introduced near the top boundary, where the analytical field meets the numerically calculated field, because the effective velocity of rays propagating diagonally across the uniform grid may not match exactly to the analytical wave. This error could be reduced by increasing the grid density. Course grids were used for these tests to reduce the time to simulate and plot the results. A 3-dimensional plot of the electric fields of a *normally* incident plane wave is shown for comparison in Figure 3. An analytically calculated waveform is shown in Figure 4 as an indication of the outcome sought in Figure 2. Further investigation is necessary to remedy the existing situation.

5.2. Symmetric Boundary Conditions

The ability to perform simulations of large structures is limited by the need to maintain a small spatial discretization for accuracy. The number of available processors typically limits the user to perform simulations of structures having minimal dimensions. Taking advantage of the symmetries in the structure can allow an increase in accuracy or simulation of a larger geometry. A symmetric version of **TEMPEST** has been created that handles structures which are symmetric about some vertical axis. This symmetric version is a modification of the isolated version which utilizes absorbing boundaries on all four boundaries. This hybrid version incorporates a symmetric boundary condition on the right (east) boundary and an absorbing boundary condition on the left (west) boundary of the 2-dimensional domain. The north and south boundaries maintain their original absorbing boundary conditions¹ as do all versions of **TEMPEST**.

A symmetric boundary, in the context of 2-dimensional electromagnetic wave propagation, is a null for the magnetic field component parallel to the boundary. The boundary can be treated as a mirror plane in which the propagation vector has a true mirror image. Figure 5 provides a graphical illustration of this concept. The notation is based upon the original discretized formulation developed by Gamelin.² Any wave traveling toward the symmetric boundary at some arbitrary angle, will meet its imaginary mirror image at that boundary. Because the field components of the real and mirror waves are equal, when the fields meet at the boundary, their superposition will either be completely destructive or completely constructive. Specifically, for the east boundary, adding field components vectorially results in a doubling of the electric-field component E_z and the magnetic-field component H_y and a cancellation of the H_x component. In theory:

$$E_z(\text{right}) = E_z(\text{left})$$

$$H_x(\text{right}) = -H_x(\text{left})$$

$$H_y(\text{right}) = H_y(\text{left})$$

where the terms “right” and “left” refer to the side of the boundary on which the field components exist. Applying the TEMPEST discretized equations at the symmetric boundary results in only the H_x component being set explicitly as expected. The two other field components are more interrelated in the curl equations and receive their predicted symmetric boundary values indirectly through the computation of the standard bulk field equations. Using Gamelin’s notation, the electric and magnetic-field equations for the bulk nodes[†] can be applied to the east boundary with a few modifications. The new electric-field equation along the symmetric boundary becomes

$$E_z^{n+1}(i,0) = \alpha E_z^n(i,0) + \beta \left[-H_x^{n+1/2}(i,0+1/2) - H_x^{n+1/2}(i,0+1/2) + H_y^{n+1/2}(i+1/2,0) - H_y^{n+1/2}(i-1/2,0) \right] \quad (5.8)$$

$$= \alpha E_z^n(i,0) + \beta \left[-2(H_x^{n+1/2}(i,0+1/2)) + H_y^{n+1/2}(i+1/2,0) - H_y^{n+1/2}(i-1/2,0) \right] \quad (5.9)$$

The H_x magnetic-field equation along the boundary becomes

$$H_x^{n+1/2}(i,0+1/2) = H_x^{n-1/2}(i,0+1/2) + \frac{\Delta t}{h\mu} (E_z^n(i,0) - E_z^n(i,0+1)) \quad (5.10)$$

[†] The electric and magnetic-field equations for the bulk nodes are defined by Gamelin¹ in equations (15), (16) and (17).

But since the electric fields on either side of the magnetic field null are equal, this equation simplifies to

$$H_x^{n+1/2}(i,0+1/2) = H_x^{n-1/2}(i,0+1/2) \quad (5.11)$$

with a need to define an initial condition: $H_x = 0$ at time $t=0$. The H_y magnetic-field equation is left undisturbed:

$$H_y^{n+1/2}(i+1/2,0) = H_y^{n-1/2}(i+1/2,0) + \frac{\Delta t}{h\mu}(E_z^n(i+1,0) - E_z^n(i,0)) \quad (5.12)$$

However, this field component is effectively double since the E_z components from which it is derived have increase by a factor of two.

To test the implementation of these equations, electromagnetic scattering from the symmetric structure shown in Figure 6 was first simulated using the isolated version of TEMPEST. The structure was then split in half along its axis of symmetry and simulated using the new symmetric version. The number of nodes along the horizontal dimension was reduced by a factor of two for the symmetric case in order to maintain a similar grid density for both simulations. A 3-dimensional plot of the maximum, steady-state electric field is shown for both simulations in Figure 7(a) and 7(b). The electric field only in the left half of the domain is plotted for the isolated simulation (Figure 7(a)) so that it can be compared directly with the plot from the symmetric simulation (Figure 7(b)). Figure 8(a) and 8(b) plot the electric field amplitude along the vertical line of symmetry from the isolated and symmetric simulations, respectively. Comparison of the electric field amplitude along this line showed a maximum difference of 4%. Because the algorithm maps the field components onto a staggered grid structure,¹ an exact comparison of field components computed by the two different TEMPEST versions is difficult. Discrepancies of $1/2 dx$ can result from nominally identical topographies. For this reason, comparisons of the diffraction efficiencies from simulations of isolated and symmetric structures, such as the one shown here, should also be used to verify simulation accuracy. Simulations of other topographies and various materials are also necessary to validate the implementation of these symmetric boundary conditions.

References

1. J. Gamelin, "Simulation of Topography Scattering for Optical Lithography with the Connection Machine," M.S. Thesis, Memorandum No. UCB/ERL M89/71, University of California, Berkeley, May 1989.
2. J. Gamelin, R. Guerrieri and A.R. Neureuther, "Exploration of Scattering from Topography with Massively Parallel Computers," *J. Vac. Sci. Technol. B.*, vol. 7, no. 6, pp. 1984-1990, Nov/Dec 1989.

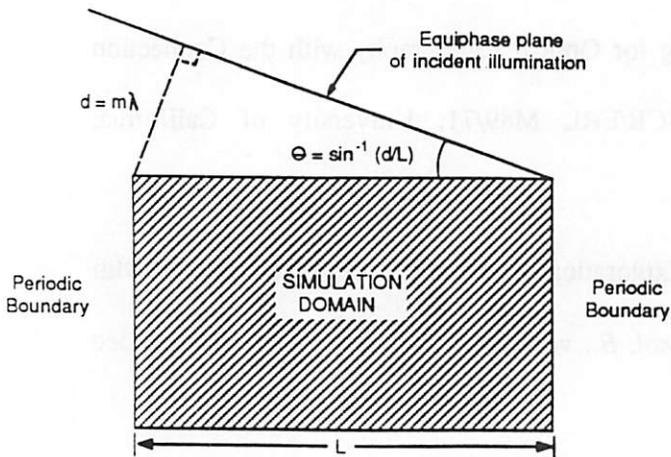


Figure 1. A schematic diagram describing the special case of an obliquely incident plane wave propagating at some angle θ such that the phase of the wave at the left and right boundaries are offset by an integer number of wavelengths.

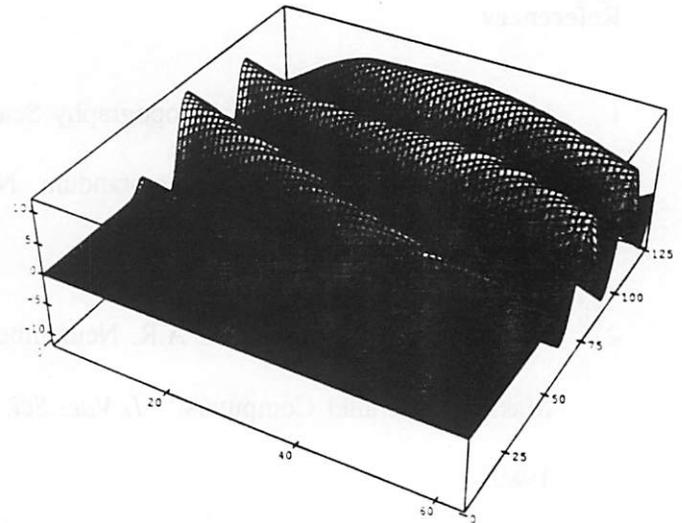


Figure 2. Instantaneous electric field of an oblique plane wave excited along the top boundary of an empty (vacuum-filled) periodic domain designed so that the incident field at the left boundary is exactly one wavelength out of phase from the field at the right boundary. Obvious errors exist.

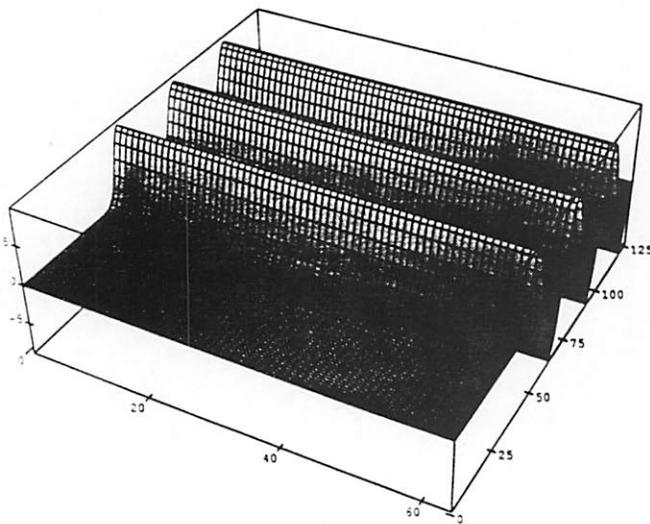


Figure 3. The instantaneous electric field from simulation of a normally incident plane wave entering an empty domain.

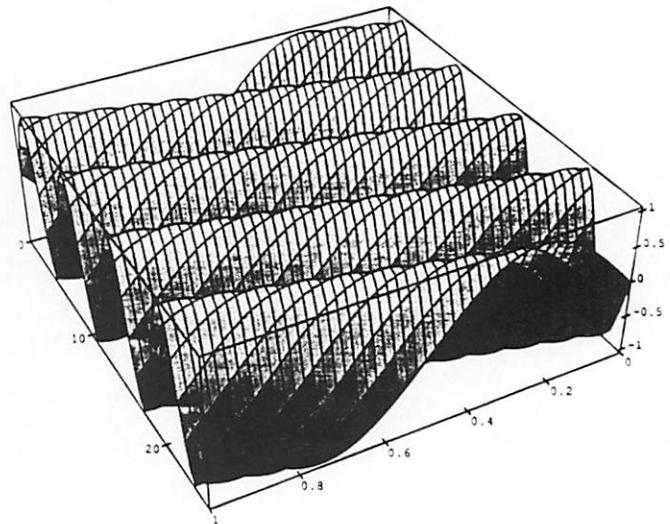


Figure 4. An analytically calculated waveform showing the expected result of obliquely incident illumination as attempted in Figure 2. The orientation of the plot is slightly different from Figures 2 and 3.

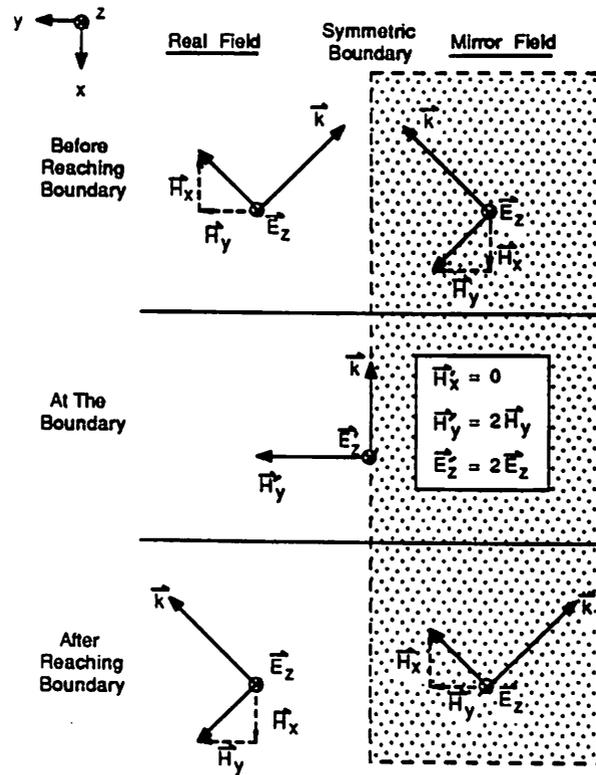


Figure 5. A schematic illustration of a 2-dimensional wave impinging upon a symmetric boundary with the field components shown.

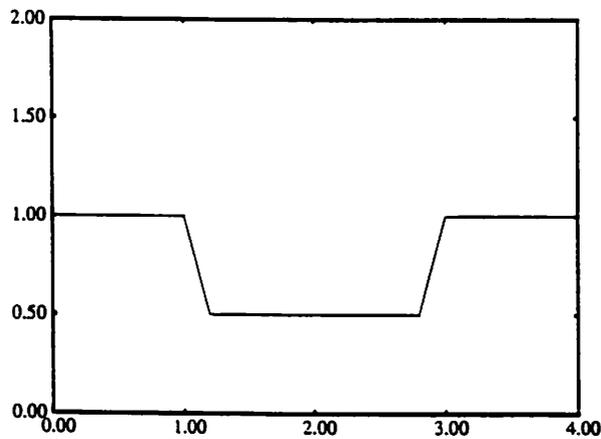


Figure 6. A symmetric trench topography used to test the symmetric boundary conditions. For simplicity, the trench is made of a perfect conductor. Axes units are in microns.

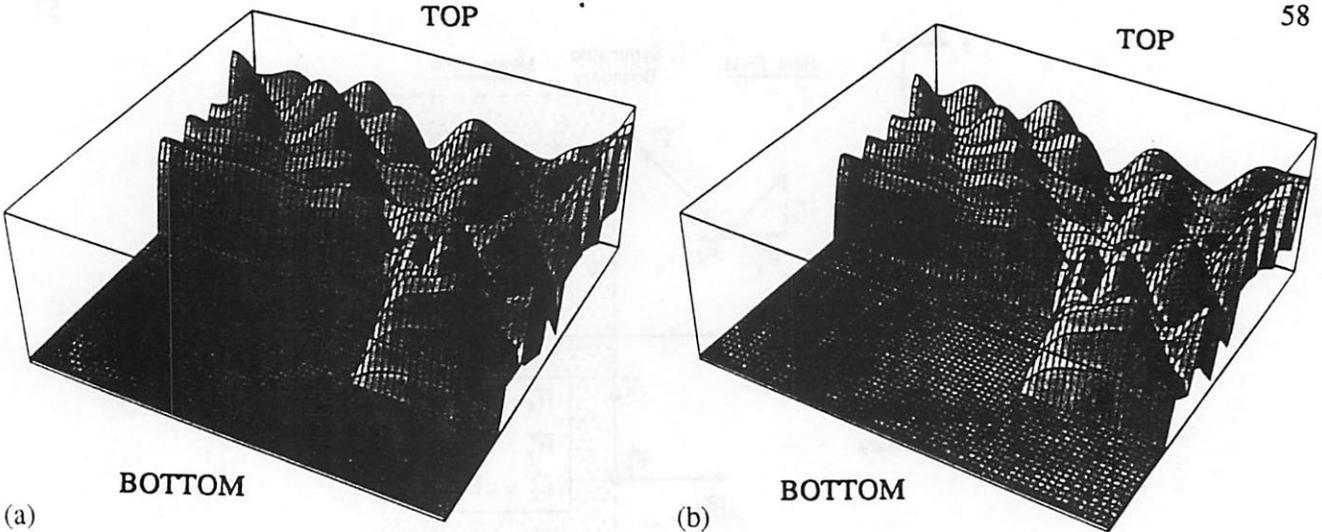


Figure 7. The maximum, steady-state, electric field amplitude everywhere in the simulation domain (Figure 6) after reaching steady-state. (a) fields from half of the domain from the isolated simulation. (b) fields from the symmetric simulation. The isolated structures was simulated using a 256 x 128 grid. The symmetric structure was simulated using a 128 x 128 grid. These grids were sampled at every second row and column for the purpose of generating the plots shown here.

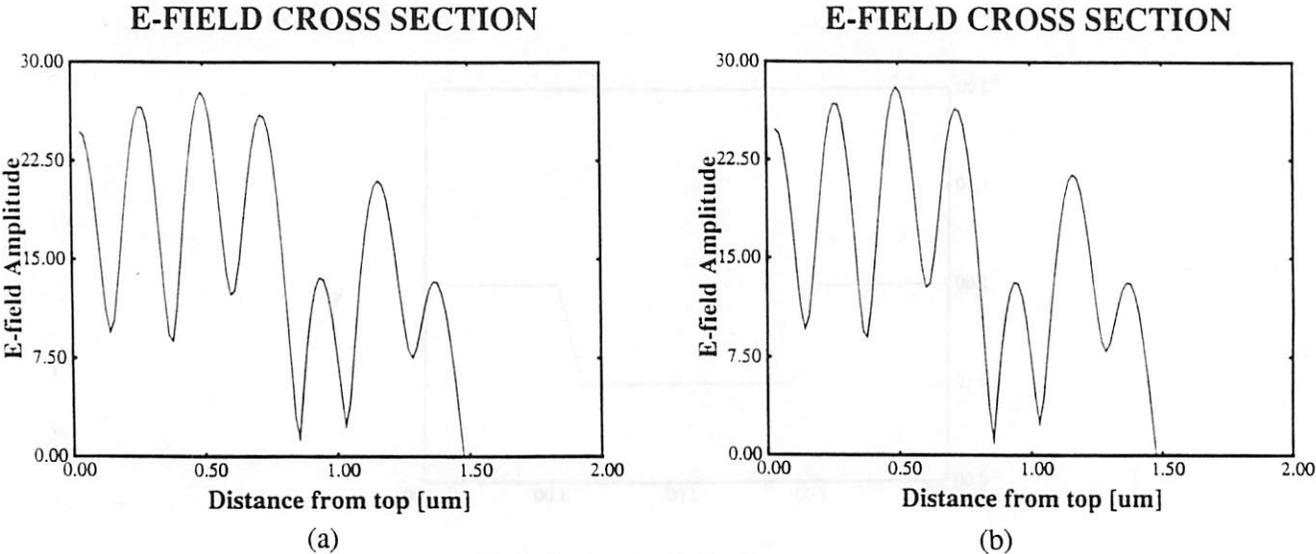


Figure 8. The maximum, steady-state electric field along a vertical cutline (line of symmetry) located at $x = 2.0 \mu\text{m}$ for the structure described in Figure 6. (a) fields from the isolated simulation (b) fields from the symmetric simulation.

Chapter 6

Conclusions

The performance of a massively parallel, electromagnetic scattering simulator, **TEMPEST**, and the suitability of the algorithm to address scattering issues in photolithography and optical metrology have been investigated. A performance evaluation has indicated that the electromagnetic scattering analysis requires constant time per iteration regardless of the problem size providing that the virtual-to-physical processor ratio is kept constant. Hence, the performance of the algorithm improves proportionally with problem size. On the other hand, it was also shown that since the steady-state solution is not in closed form, the time for the system to converge depends upon the time for all transients created by scattering from the topography to die out. Computing the fields along the boundaries presents a trade-off between accuracy and computation time. To effectively propagate waves out to infinity, rigorous calculation of the fields along absorbing boundaries requires three times more effort than calculation of the fields at all other nodes combined. However, because of parallelism, the Connection Machine achieves its peak performance when updating this large number of inner nodes.

The formulation and implementation of two enhancements to the algorithm have been presented. A method of simulating obliquely incident illumination was proposed for plane waves propagating at specific angles of incidence impinging upon a periodic structure. To achieve oblique incidence, a position-dependent temporal phase shift was added to the sinusoidal forcing function. The transients caused by the instantaneous turn-on of this position-dependent forcing function must still be addressed. Also, a symmetric boundary condition was formulated to take advantage of topographies with a vertical axis of symmetry. This boundary condition was implemented and tested in a new "symmetric version" of **TEMPEST**. Further testing of the diffraction efficiencies computed with this version are

necessary to validate its accuracy.

Two in-depth simulation studies have shown TEMPEST's capability to predict the scattering effects of both small and large changes in integrated circuit device topography. An investigation of reflective notching has given insight to the mechanisms which cause this phenomenon and the relative effectiveness of different methods to prevent it. Also, by linking electromagnetic scattering simulations performed by TEMPEST with a specially formulated optical image synthesizer and by carefully determining thin film optical parameters, image profiles of polysilicon gates measured using an NBS-type optical microscope were predicted extremely well. The results of these two detailed studies combined with the massively parallel algorithm's ability to scale with problem size have established the suitability of TEMPEST in analyzing electromagnetic scattering phenomena in photolithography and optical metrology.

APPENDIX

TEMPEST USER'S GUIDE

TEMPEST USER'S GUIDE

Karim H. Tadros
(Professor Andrew R. Neureuther)

Department of Electrical Engineering and Computer Sciences,
and Electronics Research Laboratory
University of California, Berkeley, CA 94720
(415) 642-8897

ABSTRACT

TEMPEST[†] is a massively parallel computer solution of the Maxwell equations used to analyze electromagnetic scattering from arbitrarily nonplanar and inhomogeneous topography. This simulator employs a time-domain, finite-difference algorithm which mates the inherent parallel nature of electromagnetic wave propagation with the power of the massively parallel computer architecture of the Connection Machine. **TEMPEST** simulations have been used to analyze electromagnetic scattering and optical imaging issues in the areas of photolithography, optical metrology, alignment mark signal quality, and signal integrity through photomasks. Simulations are currently limited to transverse-electric (TE) polarized illumination at normal incidence impinging upon 2-dimensional periodic, isolated or symmetric structures.

TEMPEST can be run from a remote site by any user who has obtained a computer account at any one of several Connection Machine public access facilities. The user first formulates the problem and verifies the input topography on a local workstation. The input file is then sent to the remote facility where it is processed. The output data files are then retrieved and analyzed on the local workstation. Optical image profiles can be synthesized from the output data under a variety of imaging parameters using an accompanying workstation-based imaging program. Similarly, resist line-edge profiles can be generated locally using **SAMPLE**.

This user's guide describes the instructions for operating **TEMPEST**, methods of processing its output, and details of its capabilities and its limitations. The input to **TEMPEST** is a rigidly formatted text file requiring 30 different parameters describing the topography as well as the operating conditions. When used for photolithography, a **SAMPLE** or **SPLAT** aerial image profile can be linked to the input and effectively transfer a mask pattern to the photoresist. **TEMPEST** simulations generate diffraction efficiencies for all transmitted or reflected propagating harmonics and 2-dimensional spatial distribution matrices of the electric fields and photoactive compound (PAC) concentrations.

[†] The name "TEMPEST" is an acronym that stands for "Time-domain Electromagnetic Massively Parallel Evaluation of Scattering from Topography".

Table of Contents

1. Introduction	1
2. Applications	2
3. Input Parameters	2
3.1. The “Topog.d” Input File	3
3.2. Line-by-Line Detailed Explanation	4
4. Running TEMPEST	9
4.1. The TEMPEST Code	9
4.2. Compiling and Executing the Code	10
4.3. Virtual and Physical Processors	21
4.4. Attaching and Detaching	21
5. Simulation Details	
5.1. Loading the Topography	23
5.2. The Gaussian Startup	23
5.3. Steady-State Convergence	24
5.4. TEMPEST Limitations	24
5.4.1. Courant Condition	24
5.4.2. Materials	25
5.5. Hints and Explanations	25
5.6. Boundary Conditions	26
5.6.1. Periodic	26

5.6.2. Isolated	26
5.6.3. Symmetric	26
6. A Brief History and Acknowledgements	27
References	
Figures	
Appendix A - Connection Machine Facilities	
Appendix B - Generating "mask.in"	
Appendix C - TEMPEST Image Synthesizer	
Appendix D - TEMPEST Topography Plotter	
Appendix E - A Complete Example	

1. INTRODUCTION

TEMPEST^{1, 2} is a computer solution of the Maxwell electromagnetic wave equations solved in two-dimensions using a time-domain, finite difference technique. The acronym stands for "Time-domain Electromagnetic Massively Parallel Evaluation of Scattering from Topography". This program was developed specifically for use on the Connection Machine, a fine-grain, massively parallel computer architecture designed to handle up to 64K processors simultaneously. Details of the Connection Machine can be found in the CM-2 Technical Summary.³

The current version of **TEMPEST** is written in *LISP (pronounced /star lisp/), a superset of Common LISP⁴ and is designed to run on the Thinking Machines Corp. Connection Machine, CM-2. The Connection Machine is not a stand-alone computer. It acts as a peripheral device that must be feed instructions and performs operations in response. The instructions are fed to it by a hard-wired "front-end system" which, at the time this user's guide was compiled, was required to be one of the following computer workstations:

- SUN-4 workstation running SunOS Sys 4-3.2
- VAX 8000-series minicomputer running VAX ULTRIX Version 2.0 or 2.2
- Symbolics 3600-series LISP machine running Genera 7.2

The SUN-4 and VAX front-end systems are capable of compiling C* and *FORTRAN (parallel extensions of the more common C and FORTRAN programming languages) in addition to *LISP. However, the Symbolics front-end is limited to *LISP only. At the outset of this project, UC Berkeley possessed a prototype CM-2 with only 1K processors, attached to a Symbolics 3600 LISP machine. The LISP machine restricted program development to the use of *LISP, and required all users to share the one single-user terminal. The output files would have to be transferred to a more standard workstation for post-processing. It was not until the Spring of 1989 that the Symbolics front-end was replaced with a SUN-4 workstation. This system allows multiple users to perform program development, input file preparation, program execution, and processing of the output data all on one system, even by remote logging in from a personal workstation.

TEMPEST simulations generate 3 sets of data:

TEMPEST OUTPUT	
Output File	Description
" <i>e-val.o</i> "	2-D matrix representing the spatial distribution of the maximum, steady-state electric field value for all nodes in the simulation domain.
" <i>m-val.o</i> "	2-D matrix representing the spatial distribution of the photoactive compound (PAC) concentration in all material layers of the topography. Only grid nodes located in the resist layer will be nonzero if resist exposure was simulated
" <i>de.o</i> "	diffraction efficiencies for all propagating harmonics based on either the scattered electric field or the transmitted electric field

The *e-val.o* data file is useful mainly for visual inspection with a contour or 3D plotting program. *m-val.o* is generated primarily to be passed on to a resist development model such as **SAMPLE**⁵ for simulation of line edge profiles. However, viewing a contour plot of the spatial distribution of the photosensitizer concentrations can give a quick indication of the exposure effects before performing resist development. The *de.o* file contains some useful information regarding the time and number of iterations to reach convergence as well as its primary data, the magnitude and phase of the diffraction efficiencies. The image synthesis routine (Appendix C) uses these diffraction efficiencies to compute an optical image. The image can be viewed using any standard X-Y plotting program such as **Drawplot**.⁶

The prohibitive expense of high performance computers, such as the Connection Machine, causes some concern of availability. However, any user obtaining an account on any one of several public access sites around the country is capable of running simulations from a local computer system communicating over the Internet. Appendix A lists several of these sites and the appropriate person to contact for information on how to obtain an account. Note that these sites may have restrictions on the usage of their facilities.

2. APPLICATIONS

TEMPEST is capable of simulating a number of optical phenomena occurring in photolithography and optical metrology applications. These include bleaching of photoresist over arbitrary topography, dark-field and bright-field imaging of wafer features (e.g., alignment marks and line structures), generation of image profiles through chrome masks, and calculation of the electric field at all points in the 2-D domain at any instant in time (e.g., electric fields under the gate of an EPROM). Several studies involving these topics have already been performed and the user is encouraged to refer to these publications.^{1, 7, 8, 9, 10, 11} In general, TEMPEST simulations can be placed into two categories:

- 1) lithography – involving dynamic changes during exposure
- 2) imaging / scattering analysis – concerning the far-fields

These categories are distinguished primarily by the desired output. For lithography, the desired output would be the final photoactive compound (PAC) concentrations which are dumped into the file *m-val.o*. The input parameters pertinent for this type of simulation are found in LINES 7-13, 16, and 19 of the input file [see Section 3, "Input Parameters"].

For imaging purposes, the primary output desired would be the diffraction efficiency values written into the file *de.o*. This data is used by the accompanying image synthesis routine to form the bright or dark-field images obtained by a given optical microscope. Details of this imaging program can be found in Appendix C. The other desirable output from an imaging/scattering simulation would be the electric field values in the domain. This data, found in the file *e-val.o*, is generally useful only if viewed graphically using a 3-D or contour plotting package. Vertical and horizontal cutlines of the electric fields are also of interest and can be extracted from the complete 2-D matrix. Note that the electric field components in the file *e-val.o* are not instantaneous values. To obtain the instantaneous fields, the user will need to add one line of code to print the value in the parallel variable *e* from the desired processors at a particular point in time. If TEMPEST is used for imaging/scattering purposes then the energy dose (LINE 8) should be set to 0.0 and none of the layers should be specified as a resist (LINE 19). The other input lines, pertaining to the resist, mentioned in the preceding paragraph, are ignored.

3. INPUT PARAMETERS

TEMPEST requires an input file named *topog.d* to exist in the working directory. Since TEMPEST does not parse its input, the file format of *topog.d* is rigid and requires careful attention so not to omit or rearrange any input parameters. The format of *topog.d* is discussed below in Section 3.1. If a pattern is to be transferred to resist, a file named *mask.in* must also exist in the working directory. This file is generated by SAMPLE's *imagerun* processor or by extracting a cutline from the desired 2-D image generated by SPLAT.¹² Both image generation techniques are explained in Appendix B. Hence, *topog.d* and *mask.in* (not always used) are the only two input files needed by TEMPEST.

Keep in mind that TEMPEST performs absolutely no error checking of the input. The program is far from being user-friendly regarding this point. If a mistake, such as inserting an additional input line, is made, the program will continue reading the input file but all the parameters will be offset by one. Chances are that the simulation will crash somewhere along the way with no explanation as to why, but this is not guaranteed. If the mistake is more severe, such as leaving out an entire set of layer description lines, the program will probably halt when it attempts to read past the end-of-file character. Modifying copies of a working *topog.d* file is one of the best ways to avoid simple mistakes.

Although there are three different versions of TEMPEST, one for each type of boundary condition, all versions use the same input file format. Maintaining all three versions at the same level has not been easy since each version is completely independent of the others. Because a majority of the studies conducted while this program was under development were performed with the version employing periodic boundaries, that version is the most up-to-date of the three. Any discrepancies in the input file format will be noted where appropriate.

3.1. The "Topog.d" Input File

The input text file must be named *topog.d* and must reside in the working directory. This file contains 30 different parameters describing the operating parameters and the geometry of the topography. The file format is rigid. Each line must appear in the order shown. A documented example is shown here.

"TOPOG.D" INPUT FILE		
Input	Line#	Brief Description
256 256	;1	processor grid dimensions [x y in powers of 2]
4.0	;2	width of domain (x-dimension) [μm]
0.436	;3	free space wavelength of incident radiation [μm]
1.0	;4	minimum refractive index (real part only)
1.0	;5	refractive index of incident region (real part only)
0.0	;6	angle of incidence of plane wave [degrees]
0.1	;7	intensity of incident field [W/cm^2]
0.050	;8	total energy dose [J/cm^2]
10	;9	number of energy steps to reach total dose (integer)
0.51	;10	"A" bleachable resist absorption coefficient [μm^{-1}]
0.031	;11	"B" unbleachable absorption coefficient [μm^{-1}]
0.013	;12	"C" bleach rate [cm^2/mJ]
1.0	;13	initial photosensitizer concentration [unitless]
t	;14	compute diffraction efficiencies [t/nil]
refl	;15	diff. eff. of TRANSMitted of REFLECTed fields [trans/refl]
t	;16	apply mask? (assume file "mask.in") [t/nil]
0.0	;17	angle for top boundary absorption [degrees]
3	;18	number of layers in topography (excluding air)
nil	;19a	resist layer? [t/nil]
#C(1.0 0.0)	;20a	refractive index (real and imaginary)
1	;21a	number of <i>segments</i> defining top boundary of layer (integer)
0.0 4.0	;22a1	segment endpoints (x y) [μm]
4.0 4.0	;22a2	segment endpoints (x y) [μm]
t	;19b	<i>This sequence of parameters must be repeated for each layer starting from the top (air) and working down to the substrate</i>
#C(1.68 0.0)	;20b	
1	;21b	
0.0 3.0	;22b1	
4.0 3.0	;22b2	
nil	;19c	
#C(1.463 0.0)	;20c	
5	;21c	
0.0 2.0	;22c1	
1.6 2.0	;22c2	
1.6 1.9	;22c2	

"TOPOG.D" INPUT FILE		
Input	Line#	Brief Description
2.4 1.9	;22c2	
2.4 2.0	;22c2	
4.0 2.0	;22c2	
nil	;19d	<i>last layer: assumed to be substrate</i>
#C(4.356 0.064)	;20d	
5	;21d	
0.0 1.5	;22d1	
1.6 1.5	;22d2	
1.6 1.4	;22d3	
2.4 1.4	;22d4	
2.4 1.5	;22d5	
4.0 1.5	;22d6	
0.01	;23	error factor to determine steady state convergence
1000	;24	required number of iterations before checking for steady state
5000	;25	maximum number of iterations (integer)
10	;26	number of steps in Gaussian startup (integer)
1.0e-10	;27	initial Gaussian value
8	;28	number of harmonics to calculate (integer)
nil	;29	dump cross.section file of index of refraction info [t/nil]
nil	;30	create output files e-val.o and m-val.o [t/nil]

NOTE: Only the first column, shown above, is allowed. No other characters (not even comments) may exist on a line. Blank lines are okay.

3.2. Line-by-Line Detailed Explanation

LINE 1 -- *processor grid dimensions* [x y in powers of 2]

This line must contain 2 integers representing the number of processors to be allocated in the X and Y dimension, respectively. These numbers must be powers of 2. The product of these two numbers cannot exceed *VP* times the number of physical processors to which you are attached on the CM. (*VP* is the ratio of *virtual* to *physical* processors in which the Connection Machine can be configured.) From experience, the CM-2 with 256Kbits of memory per processor is limited to a *VP* ratio of 128. Because of the relatively few parallel variables required, the periodic version of TEMPEST is capable of executing with any *VP* ratio up to and including this maximum. However, the isolated and symmetric versions are limited to a *VP* ratio no larger than 64. In the above example, $256 = 2^8$ processors will be allocated along both the X and Y dimensions. Since the discretization implemented in TEMPEST is uniform and equal in X and Y, the ratio of allocated processors in the X and Y dimensions determines the ratio of physical dimensions in simulated domain. In this case, $X = Y$.

[If the terms "*physical processors*", "*virtual processor ratio*", and "*attaching*" are unfamiliar, please refer to Section 4, "Running The Program", for a more detailed explanation.]

LINE 2 -- *width of domain (x-dimension) [μm]*

This must be a positive number in units of microns. A constraint is placed on the maximum width of the domain in order to maintain a sufficiently high grid density for algorithmic stability. Based on the Courant Condition, [see Section 5.4.1, "Courant Condition"] the grid density must satisfy:

$$\frac{\# \text{ of grid nodes}}{\lambda_{\text{in densest medium}}} > 10 . \quad (\text{from the Courant Condition})$$

This inequality constrains the width of the domain by the following relationship:

$$\frac{\# \text{ of grid nodes}}{\lambda_{\text{in densest medium}}} = \left[\frac{\lambda_{\text{vacuum}} [\mu\text{m}]}{\text{Re}(\text{largest ref. index})} \times \left[\frac{\# \text{ of processors in X dimension}}{\text{width of domain} [\mu\text{m}]} \right] \right]$$

Therefore, the width of the domain is primarily limited by the number of processors in the X dimension which in turn is limited by the physical resources of the CM to which the user has access.

LINE 3 -- *free space wavelength of incident radiation [μm]*

TEMPEST assumes monochromatic illumination. The wavelength, width of the domain, and number of grid nodes define the accuracy of the simulation. [see also LINE 2]

LINE 4 -- *minimum refractive index (real part only)*

This value should always be 1.0 (i.e., air) for the current version.

LINE 5 -- *refractive index of incident region (real part only)*

This value should always be 1.0 (i.e., air) for the current version. This requires that the top layer in the simulation domain always be set to air in the first set of LINES 19-22 where the topography is loaded.

LINE 6 -- *angle of incidence of plane wave [degrees]*

This value should always be 0.0 until TEMPEST is modified to handle obliquely incident illumination. Currently TEMPEST is capable of handling only normally incident illumination.

LINE 7 -- *intensity of incident field [W/cm²]*

This value governs the peak amplitude of the incident electric field based upon the expression

$$E = \sqrt{2I(x)_{\text{scale}} \frac{\eta_0}{n}} .$$

If resist is to be bleached, this value in conjunction with the values of energy dose (LINE 8) and number of dose steps (LINE 9) will be used to calculate the amount of energy in each dose. If the purpose of the simulation is to be for imaging/scattering analysis, then this value should be set such that the amplitude of the electric field inside of the densest layer is not smaller than 1.0e-4. This is so that the value of the electric field is not lost in the six-digit precision.

LINE 8 -- *total energy dose [J/cm²]*

This energy value refers to the resist exposure dose. If no resist is to be exposed then this value must be set to 0.0 and no material layers should be identified as being resist in LINE 19. If a resist layer is specified in LINE 19, then this value should be greater than 0.0. LINE 9 specifies how many EQUAL dose steps to perform. It is recommended that the amount of energy per dose remain smaller than 0.010 J/cm². However, this value may vary depending upon the resist parameters specified in LINES 10-12.

LINE 9 -- *number of energy steps to reach total dose* (integer)

This parameter should be set as 1 if no resist is to be bleached. Otherwise, the total energy dose (LINE 8) is divided into a number of smaller but equal energy doses as determined by this parameter. For each dose step, the electric field is driven until the system reaches steady-state. The PAC value (also termed "M-value") in the resist layer is modified according to Dill's resist model using the energy dose per step as defined by LINES 8 and 9. The electric fields are then zeroed and the incident field is excited once again until steady state convergence. This is repeated 'LINE 9' times.

LINE 10 -- *A parameter of resist bleaching model* [μm^{-1}]

Bleachable Absorption Coefficient

LINE 11 -- *B parameter of resist bleaching model* [μm^{-1}]

Unbleachable Absorption Coefficient

LINE 12 -- *C parameter of resist bleaching model* [cm^2/mJ]

Bleach Rate

These three resist parameters refer to the resist bleaching model proposed by Dill. ¹³ These values are ignored if no layer is specified as being resist. (see LINE 19)

LINE 13 -- *initial photosensitizer concentration*

This is an arbitrary value used to normalize the photoactive compound (PAC) concentration before any resist exposure. This value is typically set to 1.0.

LINE 14 -- *compute diffraction efficiencies* [t/nil]

Since TEMPEST always needs to compute the diffraction efficiencies for one reason or another, this line should always be 't' (t=true, nil=false). Diffraction efficiencies are energy values of the propagating harmonics known to exist for any periodic structure. These values are computed at discrete scattering angles which are based on the period of the structure and the illuminating wavelength. The number of existing harmonics is simply period/wavelength. When simulating isolated structures, the diffraction efficiencies are still computed by assuming the structure is actually periodic.

LINE 15 -- *diffraction efficiencies of TRANSmitted or REFLected fields* [trans/refl]

Since the diffraction efficiencies are used to compute the aerial image through a mask or the dark/bright-field images scattered from a given topography, it is necessary to specify from which fields the diffracted orders are to be calculated. "TRANS" indicates that the calculations should be based on the transmitted electric fields at the bottom of the domain and "REFL" indicates the calculations should be based on the reflected fields at the top of the domain. For the case of REFL (TRANS), the scattered fields used are those located along a horizontal cutline 10 nodes from the top (bottom) of the two-dimensional domain.

LINE 16 -- *apply mask?* [t/nil]

If TEMPEST is being used to expose photoresist, 't' should be used to indicate that a file "mask.in" exists in the working directory which will be applied as an amplitude modulation to the incident plane wave. If no photoresist exposure is needed or if the user desires to "flood expose" the resist, 'nil' should be used instead. A description of how to generate the file "mask.in" is given in Appendix B.

LINE 17 -- *angle for top boundary absorption* [degrees]

The top and bottom boundaries employ Mur's second-order absorbing boundary conditions.¹⁴ This boundary condition is formulated in such a way to absorb radiation impinging upon the boundary at a specific angle more effectively than at other angles. Typically this angle of absorption is left at 0 degrees.

LINE 18 -- *number of layers in topography* (excluding air)

TEMPEST is designed to always expect air as the top layer. However, the total number of layers does not include the air layer. Therefore, if the simulation is for bleaching photoresist on silicon, then the value on this line should be 2 (one for resist, one for silicon, air is assumed). LINES 19-22 must describe the shape of the different layers. Note that when loading the topography, however, the first layer of the topography must define air and must bound the entire domain. [see also Section 5.1, "Loading the Topography"] The other layers must be subsets of this entire domain.

LINE 19 -- *resist layer?* [t/nil]

This determines whether the following layer is to be treated as a dynamically changing resist material with resist parameters (ABC-Dill Model Parameters¹³) specified in LINES 10-12. The parameters in LINES 8, 9, and 13 will also have an effect in exposing this resist. Be sure only one layer is specified as being resist. The code does not make this check for you and undesirable results may appear. There is a version of TEMPEST which is capable of handling contrast enhancement materials (CEM). Contact Professor Neureuther for details on obtaining this version.

LINE 20 -- *refractive index* (real and imaginary)

This is a complex number represented in LISP as #C(*real imaginary*). The imaginary part must be 0.0 or negative. Using a positive number for the imaginary part will cause the program to crash. Also, there is a restriction that the real part of the refractive index must be strictly greater than the imaginary part in magnitude. Mathematically, materials with imaginary part of the refractive index greater than the real part, result in a negative value for the dielectric constant ($\epsilon = \text{Re}(\bar{n}^2) = n^2 - k^2$). As Gamelin² (§5.3) explains, it is not known the exact cause of this instability. For the case of highly conductive materials, it is best to set k just less than n . Experience has shown that for simulations with layers of metal such as aluminum or chrome, setting the refractive index of these layers to #C(1.0e10 -9.999999e9) works well. For materials such as polysilicon which are not highly conductive but possess a $k \geq n$ at the wavelength of interest, it is best to experiment. This limitation is a nuisance that will hopefully be cured in future versions. Until then, simply modify the refractive index and extinction coefficient appropriately.

LINE 21 -- *number of segments defining top boundary of layer* (integer)

The topography is loaded starting with the layer most exposed to air and working downward, layer by layer. [see also Section 5.1, "Loading the Topography"] Each layer is described by a piece-wise linear list of line segments. These line segments are defined by their endpoints. For this reason, the number of segments defining a layer boundary will always be one less than the number of endpoints listed in LINE 22.

LINE 22 -- *segment endpoints* (x y) [μm]

This LINE is repeated as many times as is necessary (as defined in LINE 21) to completely define the layer boundary.

The endpoints must remain within the dimensions specified in LINES 1 and 2. In particular, the first layer specified must be air and must define the entire simulation domain. For example, if the grid dimensions (LINE 1) are 512 by 128 and the width of the domain is 4.0 μm , then the endpoints for air must be (0.0 1.0) and (4.0 1.0) Endpoints for all other layer must fall within these values. TEMPEST will not catch the users mistakes. The simulation may crash for no apparent reason or it may run to completion but yield improper results. It is always a good idea to verify the topography information by running the program "shape" (Appendix D) on the *topog.d* file before running TEMPEST.

⇒ Be sure that the piece-wise linear segments are always increasing in the x-coordinate. If a particularly complex topography is loaded, the user may wish to verify the processors were loaded properly by generating "cross.section". [see LINE 29]

LINE 23 -- *error factor to determine steady state convergence*

The error factor is multiplied by the peak amplitude of the incident electric field to determine the error tolerance which must be met at each node along a horizontal outline near the top (rigidly set to 10 nodes from the top or bottom, depending upon the value for LINE 15) of the domain in order for the system to be considered in steady state.

$$err_tol = error_factor \times amplitude$$

$$|E_{new} - E_{old}| \geq err_tol$$

A value of 0.01 has been working very well.

LINE 24 -- *required number of iterations before checking for steady state*

Each cycle (*i.e.*, one period of excitation) of the incident radiation is divided into an integer number of time steps (*dt*'s). The number of *dt*'s in one cycle of the incident field is printed on the screen at the beginning of program execution (*nperiod* = xxx). Typical values range from 20 to 100. An iteration is defined as one time step. The simulation will excite the incident field with the number of iterations specified on this LINE before beginning to check for steady state convergence. Since checking for steady state consumes CPU time, it is wise not to make the minimum number of iterations less than 1000. It has been determined from experience that the number of iterations a system needs to reach steady state is related to the physical scattering of multiple reflections from the specific topography being simulated. Hence it is more accurate to say that simulations typically require at least 20 cycles of excitation before reaching steady state. More wave cycles are needed if the system is comprised of many dielectric layers having a considerable variance in their index of refraction.

LINE 25 -- *maximum number of iterations (integer)*

The simulation will continue to excite the incident plane wave until steady state is reached or until this user specified number of iterations, whichever comes first.

LINE 26 -- *number of steps in Gaussian startup (integer)*

Since reaching steady state quickly is desirable, a Gaussian curve has been applied to the startup of the incident sinusoid to reduce the transients associated with a step input as occurs with a sinusoid at time $t = 0$. This input LINE allows the user to define how quickly the Gaussian curve climbs. A value of 10 has been found to be reasonably effective. [see also Section 5.4.1, "The Gaussian Startup"]

LINE 27 -- *initial Gaussian value*

A value of 1.0e-10 has been found to give sufficient results. [see LINE 26]

LINE 28 -- *number of harmonics to calculate (integer)*

This is used for calculating the diffraction efficiencies. Always compute at least one harmonic. Since it is typically not necessary to compute the diffraction efficiencies when bleaching photoresist, this value can be set to a minimum. But when performing an imaging/scattering analysis this value must be set to $period / \lambda_{air}$, truncated to the nearest integer. If this value is larger than $period / \lambda_{air}$ or is not truncated to the nearest integer, the program will crash when it attempts to calculate a nonpropagating harmonic. This image synthesis routine (Appendix C) requires all propagating harmonics to be available for computing an aerial image.

LINE 29 -- *dump cross.section file of refractive index [t/nil]*

This parameter is useful for debugging the loading of the processors with the topography information. If this parameter is set to 't', TEMPEST will load the processors of the Connection Machine with the appropriate refractive index values based on the topography information given in LINES 18-22. It will then immediately create a file named "cross.section" which contains an X-Y matrix of the refractive indices as found in each of the processors. This matrix is an EXACT indication of how the processors

were loaded. A message will appear on the screen to inform the user that the refractive index matrix is being written to disk. When it is done, the user may abort program execution by typing CTRL-C. The reason for stopping the program after dumping the matrix is related to the fact that we have been using Mathematica's Density Plotting tool to display the "cross.section" matrix. Examples are given in Appendix D. Mathematica's shading scheme is somewhat backwards for our purpose. The shading is brighter as the magnitude of the value grows larger and black shading is for values near zero. For this reason, we UC Berkeley have typically changed the refractive indices in the input file so that dense materials (such as silicon) will have small numbers (e.g., 1.0) and the air layer will have a large number (e.g., 10.0). Since these values are incorrect for the purposes of performing simulation, as soon as the "cross.section" file is written to disk, type CTRL-C to abort the simulation. The cross-section data has proven to be useful in determining faulty user layout on a number of occasions. [see also Appendix D]

LINE 30 -- create output file *m-val.o* [t/nil]

If this LINE is 't', photoactive compound (PAC) concentration matrix is written to disk in the file *m-val.o* in addition to creating the two files *de.o* (diffraction efficiencies) and *e-val.o* (the peak value in each processor during one cycle of excitation) . If this LINE is 'nil' then only *de.o* and *e-val.o* are created. This parameter is offered since, although the Connection Machine performs its computations in parallel, when communicating to the front end, it must do so in serial fashion. Hence, when dumping the value of a variable in every processor of the simulation domain, it will take a significant amount of time and disk storage. Since there is one floating point value for each node in the domain it is not wise to generate these large files when they are not necessary. This LINE gives the user the option of creating these files or not. (FYI: for a 256x256 domain, it takes roughly 3-5 minutes to generate either *m-val.o* or *e-val.o* on a SUN-4 front-end system.)

4. RUNNING TEMPEST

The different versions of TEMPEST (distinguished by the type of boundary condition employed: periodic, isolated, and symmetric) have not been consolidated into one unified program. In order to simulate an isolated structure, the *LISP[†] files comprising only that unit must be compiled and loaded; likewise for each of the other versions. The primary reason for this separation can be traced to the need for efficient use of parallel variables on the Connection Machine. For the isolated case, almost twice as many variables are needed than for the periodic case. By splitting up the code, less memory is required of the Connection Machine which results in a larger number of virtual processors available for the simulation.

4.1. The TEMPEST Code

Each version of TEMPEST is comprised of several modular *LISP files. Nine different *LISP files make up the periodic version while 10 *LISP files are needed for the isolated and symmetric versions. The files comprising each version of the code are listed in the table below. Each version is distinguished by the filename extension following the root filename.

-ni10	⇒	Periodic
-ni12	⇒	Isolated
-ni13	⇒	Symmetric

[†] Note that the term "*LISP" is specifically used to refer to those aspects of the language or operating environment pertaining to the parallel extension of the more standard "LISP" language and environment. The seemingly interchangeable usage of these two terms in the following sections is not an attempt to confuse the reader. Rather, it is intended to make the reader more aware of what part of the language is responsible for the task at hand.

For example, the "main" file belonging to the periodic version is named "main-ni10.lisp"; "main-ni12.lisp" belongs to the isolated version; and "main-ni13.lisp" belongs to the symmetric version.

TEMPEST VERSIONS				
ROOT FILENAME	PERIODIC (-ni10)	ISOLATED (-ni12)	SYMMETRIC (-ni13)	DESCRIPTION
main	X	X	X	Directs program execution
inn	X	X	X	Guts of TEMPEST, finite-difference code, and variable definitions
in	X	X	X	Reads input files and loads processors
in_mask	X	X	X	Reads mask.in file if necessary
com_e	X	X	X	Updates E&M fields, determines steady-state convergence
com_de	X	X	X	Computes diffraction efficiencies
mesh	X	X	X	Routines called by in-ni10.lisp to load processors
up_index	X	X	X	Updates refractive index of resist if bleaching is desired
out_f	X	X	X	Writes E-field and PAC values to files and dumps matrix of refractive index as a debugging aid
average		X	X	Computes effective refractive index at interface of different layers along boundaries

Section 4.2 explains how each of these files is to be compiled and loaded by displaying a detailed dialog session.

4.2. Compiling and Executing the Code

Since the Common LISP programming environment is unfamiliar to most computer users, a detailed dialog session of how to compile and execute TEMPEST is provided here. LISP is accessed by entering an interpreted environment similar to BASICA on most DOS-based personal computers. From within this environment, the *LISP package must first be linked in. The appropriate *compiler options must be set. Then the TEMPEST code, consisting of 9 different *LISP files (10 files for the isolated and symmetric versions), must be compiled and loaded. This sequence is shown below in the form of a dialog session on the SUN-4 front-end of the Connection Machine at UC Berkeley. The command to enter the *LISP environment is *starlisp*. The command to exit the *LISP environment is (*system::quit*). It will be necessary at times to move back and forth between Unix and the LISP environment without restarting every time. The easiest way to do this is by using the *suspend* (CTRL-Z) and the *foreground* (fg) commands. From inside the LISP environment, wait until the LISP prompt (>) appears then type CTRL-Z to temporarily halt LISP. This will dump you back into Unix. At this point, the input file can be modified with any text editor such as *vi*. When done working in Unix, type "fg" at the Unix prompt. You will be placed back into the LISP environment at the point where it was halted. This feature will be used below in the dialog.

The *LISP environment on the 1K prototype CM at UC Berkeley is configured with very little memory for program execution. An automatic Garbage Collection routine is executed by the system quite often to regain unused memory. To reduce the frequency of these Garbage Collections, a function call to expand user memory is performed at the outset. This is shown below in the dialog. Depending on how other CM front-end systems are configured, this function call for memory expansion may not be necessary.

When setting the compiler options, the default settings are printed automatically (in *italic*). The bold text identifies user input for suggested settings to increase program speed for "production" use. If modifications are made to the *LISP code, a few options should be set to aid in debugging. These are identified below.

The following conventional prompts are used to distinguish to which process the commands are given.

%	unix prompt
>	lisp environment prompt
bold	user input
roman	computer's response
<i>italic</i>	default settings

The dialog session begins at the Unix prompt and assumes the user is in the directory which contains the TEMPEST *LISP files, the *topog.d* input file, and the SAMPLE or SPLAT-generated mask file *mask.in* (if it is to be used).

% starlisp

*{lots of text is printed as *LISP is loaded}*

> (in-package '*lisp)

#<Package "*LISP" 244271E>

> (lucid-common-lisp:change-memory-management :expand 400)

;;; GC: 9178 words [36712 bytes] of dynamic storage in use.

;;; 154660 words [618640 bytes] of free storage available before a GC.

;;; 318498 words [1273992 bytes] of free storage available if GC is disabled.

;;; Expanding Dynamic Memory

T

> (compiler-options :class :all)

Starlisp Compiler Options

Compile Expressions (Yes, or No), currently *Yes*

Warning Level (High, Normal, None), currently *Normal*

Inconsistency Reporting Action (Abort, Break, Error, Cerror, Warn, None), currently *Warn*

Safety (0, 1, 2, 3), currently *1*

Print Length for Messages (an integer, or Nil), currently *4*

Print Level for Messages (an integer, or Nil), currently *3*

Optimize Bindings (No, Cspeed<3, Yes), currently *Cspeed<3*

Peephole Optimize Paris (No, Cspeed<3, Yes), currently *Cspeed<3*

Pull Out Common Address Expressions (Yes, or No), currently *No Yes*

Use Always Instructions (Yes, or No), currently *No Yes*

Machine Type (Current, Compatible, Cm2, Cm2-Fpa, Simulator), currently *Current*

Add Declares (Everywhere, Yes, No), currently *Yes Everywhere*

Use Undocumented Paris (Yes, or No), currently *Yes*

Verify Type Declarations (No, Current-Safety, Yes), currently *Current-Safety*

Constant Fold Pvar Expressions (Yes, or No), currently *Yes*

Speed (0, 1, 2, 3), currently *1 3*

Compilation Speed (0, 1, 2, 3), currently *1*

Space (0, 1, 2, 3), currently *1 3*

Immediate Error If Location (Yes, or No), currently *Yes*

Optimize Check Stack Expression (Yes, or No), currently *Yes*

Generate Comments With Paris Code (Yes, Macro, No), currently *Yes No*

Array size inline limit (an integer, or Nil), currently *Nil*

Expand *Defun and *Let Wrapper (No, Yes), currently *No*

Rewrite Arithmetic Expressions (No, Yes), currently *Yes*

Use Code Walker (Yes, or No), currently *No*
Do the assignment? (Yes, or No) *yes*
>

For users wishing to modify the *LISP code, the following options should be set as shown to aid in debugging:

Warning Level (High, Normal, None), currently *Normal High*
Safety (0, 1, 2, 3), currently *1 3*
Print Length for Messages (an integer, or Nil), currently *4 nil*
Print Level for Messages (an integer, or Nil), currently *3 nil*
Use Code Walker (Yes, or No), currently *No Yes*

If the user is modifying the PARIS[†] code other options can also be specified to aid in debugging:

Use Undocumented Paris (Yes, or No), currently *Yes No*
Generate Comments With Paris Code (Yes, Macro, No), currently *Yes*

These compiler options may be redefined at any time by re-executing the same command. Once these compiler options are set, the user is ready to compile the TEMPEST *LISP files. The user may compile and load each file interactively, or a LISP program can be written using a text editor such as *vi* then loaded and executed from within LISP to automate the procedure. The LISP file is displayed here.

[†] PARIS = PARAllel Instruction Set. This is the Connection Machine's machine language.

```
[file name = "Compile_load.lisp"]
  (in-package '*lisp)

  (defun compile_load()

    (compile-file "inn2-ni10")
    (load "inn2-ni10")

    (compile-file "in-ni10")
    (load "in-ni10")

    (compile-file "main-ni10")
    (load "main-ni10")

    (compile-file "com_de-ni10")
    (load "com_de-ni10")

    (compile-file "com_e-ni10")
    (load "com_e-ni10")

    (compile-file "in_mask-ni10")
    (load "in_mask-ni10")

    (compile-file "mesh-ni10")
    (load "mesh-ni10")

    (compile-file "up_index-ni10")
    (load "up_index-ni10")

    (compile-file "out_f-ni10")
    (load "out_f-ni10")
  )
```

Simply load then execute this file as shown below. Notice that some warning messages may appear during this process. They may be ignored.

```
> (load "Compile_load.lisp")
;;; Loading source file "Compile_load.lisp"
#P"/perseus1/tadros/work_tempest/Compile_load.lisp"

> (compile_load)
;;; You are using the compiler in production mode (compilation-speed = 0)
;;; Generation of argument count checking code is enabled (safety = 1)
;;; Optimization of tail calls is enabled (speed = 3)
;;; While compiling INITIALIZE-BOUNDARY-CONDITIONS0
;;; Warning: Free variable A assumed to be special
;;; Warning: Free variable T0 assumed to be special
;;; Warning: Free variable TT assumed to be special
;;; While compiling INN-LOOP
;;; Warning: Free variable GAMMA assumed to be special
;;; Warning: Free variable C6 assumed to be special
;;; Warning: Free variable CB2 assumed to be special
;;; Warning: Free variable CB1 assumed to be special
;;; Warning: Free variable C2 assumed to be special
```

```

;;; Warning: Free variable C1 assumed to be special
;;; Warning: Free variable CB3 assumed to be special
;;; While compiling INN-LOOP0
;;; Warning: Free variable GAMMA assumed to be special
;;; Warning: Free variable C2 assumed to be special
;;; Warning: Free variable NUMIT assumed to be special
;;; Warning: Free variable N_INIT assumed to be special
;;; Warning: Free variable C1 assumed to be special
;;; Warning: Free variable TMP0 assumed to be special
;;; Warning: Free variable A assumed to be special
;;; Warning: Free variable T0 assumed to be special
;;; Warning: Free variable TT assumed to be special
;;; Writing binary file "inn2-ni10.sbin3"
;;; Loading binary file "inn2-ni10.sbin3"
;;; Reading source file "in-ni10.lisp"
;;; While compiling IN
;;; Warning: Variable TMP1 is bound but not referenced
;;; Writing binary file "in-ni10.sbin3"
;;; Warning: The following functions are not known to be defined:
;;;     PAR_LOAD_PROCESSORS!! was referenced by IN
;;;     INTERPOLATE-GRID was referenced by IN
;;;     UPDATE-PAR!! was referenced by IN
;;; Loading binary file "in-ni10.sbin3"
;;; Reading source file "main-ni10.lisp"
;;; Writing binary file "main-ni10.sbin3"
;;; Warning: The following functions are not known to be defined:
;;;     CROSS_SECTION was referenced by MAIN
;;;     COMPUTE_FIELDS was referenced by MAIN
;;;     COMPUTE_DE was referenced by MAIN
;;;     UPDATE_INDEX was referenced by MAIN
;;;     OUTPUT_FIELDS was referenced by MAIN
;;; Loading binary file "main-ni10.sbin3"
;;; Reading source file "com_de-ni10.lisp"
;;; Writing binary file "com_de-ni10.sbin3"
;;; Loading binary file "com_de-ni10.sbin3"
;;; Reading source file "com_e-ni10.lisp"
;;; While compiling COMPUTE_FIELDS
;;; Warning: Free variable E_PLOT1A assumed to be special
;;; Warning: Free variable E_PLOT1B assumed to be special
;;; Warning: Free variable E_PLOT2A assumed to be special
;;; Warning: Free variable E_PLOT2B assumed to be special
;;; Warning: Free variable E_RMS1 assumed to be special
;;; Warning: Free variable E_RMS2 assumed to be special
;;; Writing binary file "com_e-ni10.sbin3"
;;; Warning: The following function is not known to be defined:
;;;     IN_MASK was referenced by COMPUTE_FIELDS
;;; Loading binary file "com_e-ni10.sbin3"
;;; Reading source file "in_mask-ni10.lisp"
;;; Writing binary file "in_mask-ni10.sbin3"
;;; Loading binary file "in_mask-ni10.sbin3"
;;; Reading source file "mesh-ni10.lisp"
;;; While compiling PAR_LOAD_PROCESSORS!!
;;; Warning: Type declaration of NN assumed to be special

```

```

;;; Writing binary file "mesh-ni10.sbin3"
;;; Loading binary file "mesh-ni10.sbin3"
;;; Reading source file "up_index-ni10.lisp"
;;; Writing binary file "up_index-ni10.sbin3"
;;; Loading binary file "up_index-ni10.sbin3"
;;; Reading source file "out_f-ni10.lisp"
;;; Writing binary file "out_f-ni10.sbin3"
;;; Loading binary file "out_f-ni10.sbin3"
#P"/perseus1/tadros/work_tempest/out_f-ni10.sbin3"
>

```

Now we are ready to attach to the Connection Machine. Before doing so, it is wise to check if any ports are available. To do this, use the command (*cm:finger*).

```

> (cm:finger)

Name of CM: Polydorus      Physical size = 1k, 64k ram.
          NOBODY          PERSEUS.BERKELEY.EDU:0 Not attached
          NOBODY          PERSEUS.BERKELEY.EDU:1 Not attached
NIL
>

```

We see that nobody is attached so we can proceed to attach to the desired number of processors (the default is always the smallest available unit: 8K processors on a 32K machine, 1K processors at UC Berkeley). After attaching with the command (*cm:attach*), a **cold-boot* is performed to reset the processors and initialize all the parallel variables. Notice that the command (**cold-boot*) by default configures the processors as a two-dimensional array with a *VP* ratio of 1.

```

> (cm:attach)
1024

> (*cold-boot)
1024
(32 32)

> (cm:finger)

Name of CM: Polydorus      Physical size = 1k, 64k ram.
          TADROS          PERSEUS.BERKELEY.EDU:0 Sequencer Ports (0) <-- Attached to 1024 physical processors
          NOBODY          PERSEUS.BERKELEY.EDU:1 Not attached
NIL
>

```

At this point, the program is ready to execute. Be sure the input file *topog.d* is correct and in the working directory. If it is not, look ahead to see how CTRL-Z is used to temporarily return to the UNIX environment. The file *mask.in* must also exist in the working directory if it is to be used in the simulation. To start the simulation, invoke the "main" function. The program will automatically re-dimension the processors into the required configuration as defined by the input LINE 1. Many lines of output will appear as the program executes. The LISP prompt will return when the simulation ends.

```

> (main)

simulation grid dimensions (x,y) in powers of 2 x,y = 1024 128
period of domain (um) dx = 0.0078125

```

```

Free-space wavelength of incident radiation (um) lambda = 0.5461
Minimum refractive index Index of refraction of incident region n0 = 1.0
time step is 1.8215944839143638E-17
Angle of incidence for plane wave theta = 0.0
Intensity of the incident field (mW/cm**2) intensity,amplitude = 0.1 8.6802
0852284091
Total energy dose (mJ/cm**2) energy = 0.0
Number of energy steps to total dose en_step = 1
ABC Parameters of resist aa = 0.51 , bb = 0.031 , cc = 0.013
total exposure time is 0. seconds
Initial Photosensitizer concentration mm = 1.
Compute diffraction efficiencies at each step? (t or nil) d_eff = T
Compute reflected or transmitted diff. eff.? (refl or trans) ref_tra = REFL
Mask with input file mask.in? (t or nil) mask = NIL
angle for top-boundary absorption tht is 0.0
nperiod= 100
ready to warm-boot
done warm-boot
number of layers num_layers = 3
Resist Layer? (t or nil) rst = NIL
index of refraction for layer 0 indexn = 1.0 0.0
number of segments in layer 0 number of segments = 1
eps,cond = 1. 0.
Resist Layer? (t or nil) rst = NIL
index of refraction for layer 1 indexn = 4.14 -0.031
number of segments in layer 1 number of segments = 5
eps,cond = 17.138638999999998 885360610656458.
Resist Layer? (t or nil) rst = NIL
index of refraction for layer 2 indexn = 1.461 0.0
number of segments in layer 2 number of segments = 5
eps,cond = 2.1345210000000003 0.
Resist Layer? (t or nil) rst = NIL
index of refraction for layer 3 indexn = 4.093 -0.031
number of segments in layer 3 number of segments = 1
eps,cond = 16.751687999999998 875309415318087.5
Mean Square Error tolerance MSE tolerance = 0.01
Number of initial iterations
wait-steps = 1000
Maximum number of iterations max-it = 4010
Number of initial time steps = 10
Initial value for tail = 1.0E-10
Largest harmonic number to calculate = 14
Dump 'cross.section' file with refractive index info? NIL
Create output file m-val.o and/or e-val.o? NIL
ready for iterations

test = 0.086802
numit = 1112.0 err = 2924.088900
try = 1
sum of okay = 567.0
numit = 1212.0 err = 13.797140
try = 1
sum of okay = 674.0

```

```

numit = 1312.0  err = 7.664220
try = 1
sum of okay = 687.0
numit = 1412.0  err = 6.343640
try = 1
sum of okay = 784.0
numit = 1512.0  err = 7.785161
try = 1
sum of okay = 770.0
numit = 1612.0  err = 47.800087
try = 1
sum of okay = 809.0

```

```

numit = 3212.0  err = 0.369834
try = 1
sum of okay = 1024.0
numit = 3312.0  err = 0.312238
try = 2
sum of okay = 1024.0
sum of okay = 1024.0
numit = 3412.0  err = 0.374982
try = 3
sum of okay = 1024.0
sum of okay = 1024.0
sum of okay = 1024.0

```

\$\$\$ Fields reached steady-state. MSE = 0.37498167 \$\$\$

\$\$\$ Required 3412 iterations \$\$\$

```

diffraction efficiency,phase for order 0 is 2.856350E-1 , 3.033794E+0
diffraction efficiency,phase for order 1 is 5.979605E-3 , 2.948694E+0
diffraction efficiency,phase for order - 1 is 5.979629E-3 , 2.948695E+0
diffraction efficiency,phase for order 2 is 4.807064E-3 , -3.477628E-1
diffraction efficiency,phase for order - 2 is 4.807061E-3 , -3.477647E-1
diffraction efficiency,phase for order 3 is 3.252412E-3 , 2.492445E+0
diffraction efficiency,phase for order - 3 is 3.252430E-3 , 2.492443E+0
diffraction efficiency,phase for order 4 is 1.871640E-3 , -1.128021E+0
diffraction efficiency,phase for order - 4 is 1.871642E-3 , -1.128019E+0
diffraction efficiency,phase for order 5 is 1.164476E-3 , 1.237144E+0
diffraction efficiency,phase for order - 5 is 1.164494E-3 , 1.237138E+0
diffraction efficiency,phase for order 6 is 1.323964E-3 , -2.862597E+0
diffraction efficiency,phase for order - 6 is 1.323941E-3 , -2.862600E+0
diffraction efficiency,phase for order 7 is 2.006065E-3 , -5.460889E-1
diffraction efficiency,phase for order - 7 is 2.006072E-3 , -5.460905E-1
diffraction efficiency,phase for order 8 is 2.586660E-3 , 1.892791E+0
diffraction efficiency,phase for order - 8 is 2.586652E-3 , 1.892794E+0
diffraction efficiency,phase for order 9 is 2.551609E-3 , -1.958540E+0
diffraction efficiency,phase for order - 9 is 2.551622E-3 , -1.958543E+0
diffraction efficiency,phase for order 10 is 1.640728E-3 , 4.440488E-1
diffraction efficiency,phase for order - 10 is 1.640715E-3 , 4.440474E-1

```

```

diffraction efficiency,phase for order 11 is 5.467406E-4 , 2.667494E+0
diffraction efficiency,phase for order - 11 is 5.467432E-4 , 2.667495E+0
diffraction efficiency,phase for order 12 is 2.879736E-5 , -2.869633E+0
diffraction efficiency,phase for order - 12 is 2.879995E-5 , -2.869623E+0
diffraction efficiency,phase for order 13 is 3.284248E-4 , -2.654984E+0
diffraction efficiency,phase for order - 13 is 3.284261E-4 , -2.654984E+0
diffraction efficiency,phase for order 14 is 1.112988E-3 , -1.932374E+0
diffraction efficiency,phase for order - 14 is 1.112989E-3 , -1.932378E+0

```

```
total energy = 0.344037431671044848
```

```
NIL
>
```

The simulation ran to completion (also, steady-state was reached) and created the files *e-val.o* and *de.o*. There are two pieces of useful information found in the *de.o* that are not printed to the screen.

Computing reflected diffraction efficiencies.

total overhead time = 611.13 seconds

If another simulation were to be performed immediately, the user must temporarily exit the LISP environment with CTRL-Z, rename the output data files *e-val.o* and *de.o* so that the next simulation will not overwrite them, modify *topog.d* for the next simulation, then return to the LISP environment and execute "main" again. This is all shown in the on-going dialog below.

```

> ^Z
stopped

% mv de.o de.trial_1

% mv e-val.o e-val.trial_1

% vi topog.d

      {make changes to input file as necessary}

% fg
starlisp

```

Typically, the LISP prompt does not reappear when re-entering LISP from Unix. This is not a problem. It is just a characteristic of the LISP environment. If the user wishes to exit completely from the LISP environment and return to Unix, first the Connection Machine must be detached. Then the LISP environment is exited. The on-going dialog shows how.

```

> (cm:detach)
T

> (system::quit)

%

```

In future sessions, the user is not required to recompile the code unless some modifications were made. Instead, the binary files can be loaded in immediately and all other commands will remain the same with the exception that the compiler options do not need to be set. We have created a simple LISP program similar to the previous

program which was used to compile and load the files comprising TEMPEST, but this one only loads the compiled files. [file name = "Load.lisp"]

```
(in-package '*lisp)

(defun load_tempest ()
  (load "inn2-ni10")
  (load "in-ni10")
  (load "main-ni10")
  (load "com_de-ni10")
  (load "com_e-ni10")
  (load "in_mask-ni10")
  (load "mesh-ni10")
  (load "up_index-ni10")
  (load "out_f-ni10")
)
```

The following dialog shows how to run a new TEMPEST simulation if the code has already been compiled.

```
% starlisp

      {lots of text is printed as *LISP is loaded}

> (in-package '*lisp)
#<Package "*LISP" 244271E>

> (lucid-common-lisp:change-memory-management :expand 400)
;;; GC: 9178 words [36712 bytes] of dynamic storage in use.
;;; 154660 words [618640 bytes] of free storage available before a GC.
;;; 318498 words [1273992 bytes] of free storage available if GC is disabled.
;;; Expanding Dynamic Memory
T

> (load "Load.lisp")
;;; Loading source file "Load.lisp"
#P"/perseus1/tadros/work_tempest/Load.lisp"

> (load_tempest)
;;; Loading binary file "inn2-ni10.sbin3"
;;; Loading binary file "in-ni10.sbin3"
;;; Loading binary file "main-ni10.sbin3"
;;; Loading binary file "com_de-ni10.sbin3"
;;; Loading binary file "com_e-ni10.sbin3"
;;; Loading binary file "in_mask-ni10.sbin3"
;;; Loading binary file "mesh-ni10.sbin3"
;;; Loading binary file "up_index-ni10.sbin3"
;;; Loading binary file "out_f-ni10.sbin3"
#P"/perseus1/tadros/work_tempest/out_f-ni10.sbin3"

> (cm:attach)
;;; Loading source file "/cm/configuration/configuration.lisp"
1024

> (*cold-boot)
1024
```

(32 32)

> (main)

.

.

program execution

.

.

> (cm:detach)

T

> (system::quit)

%

4.3. Virtual and Physical Processors

The Connection Machine was designed with the ability to handle up to 64K (K=1024) processors operating in parallel. These processors are termed *physical* since they exist on silicon. The total number of physical processors is split into four equal units. Each unit is independently controlled by a hardware device called a "sequencer". A sequencer may be used alone or combined in groups of two or four to provide one user with a greater number of physical processors. For a 32K machine, the following possible combinations may exist:

32K Connection Machine Physical Processor Configurations								
Number of users	Unit #1	Unit #2	Unit #3	Unit #4	Corresponding Unit Size			
4	1	1	1	1	8K	8K	8K	8K
3		2	1	1	16K		8K	8K
2		2		2	16K			16K
1			4					32K

Each of the physical processors has 256K bits of internal memory for computations and allocation of variables. These processors can be placed in any k-dimensional configuration. TEMPEST always configures the processors in 2-dimensions. The system software was designed to allow each physical processor to segment its local memory into some 2^N virtual units. Each unit would then represent a virtual processor and would be treated as an independent entity. These virtual processors would have a factor of 2^N less memory than the physical processors. The value 2^N is termed the *VP* ratio (virtual-to-physical processor ratio). A direct result of this reduction in memory per processor, is a limit on how big the *VP* ratio can be. Since the 256K bits of memory must be divided among the 2^N virtual processors, there is a value of N which will result in insufficient memory to allow variables to be allocated or computations to take place. This will be different for different programs. In the case of TEMPEST, the periodic version is limited to a *VP* ratio of 128. The symmetric and isolated versions are limited to a *VP* ratio of 64. The difference is due to the increased number of variables needed for defining the boundary conditions on all four sides of the domain for the latter two versions.

4.4. Attaching and Detaching

Attaching and *detaching* to the Connection Machine is a skill that should be mastered quickly in order to not be intimidated by the those users who hover around the Connection Machine ports waiting to grab any resources given up by a current user. As shown in the dialog, before a simulation can be run on the Connection Machine, you must first "attach" to the machine with the command (*cm:attach*). If none of the four sequencers are available, this command will respond with the message: *cmattach: No sequencers are available*. The command (*cm:finger*) lists each sequencer and its current usage. The same information can be obtained while at the UNIX prompt with the command *cmfinger* or *cmusers*.

```
> (cm:finger)
CM      Seqs Size  Front end I/F  User  Idle  Command
-----
NEARNET-CM2 0-1 16K   cmns-sun  0   rajan  0h 00m "starlisp.patch3"
              rajan  0h 00m "cmattach"
NEARNET-CM2 3   8K   cmns-sun  1   grumpy 1h 15m "cmattach"
NEARNET-CM2 --- ---  cmns-vax  1   (nobody)
NEARNET-CM2 2   8K   cmns-vax  0   linchiu 0h 11m "cmattach"
64K memory, 64-bit floating point
framebuffer on sequencer 0
CMIOCs on sequencers 0 1 2 3
no free sequencers on NEARNET-CM2

>
```

In some instances, a user may forget to detach from the Connection machine while others are waiting to use it. Idle time is printed in the *cmfinger* information as shown above. If another user wishes to use the CM, the proper CM etiquette requires that some attempt be made to contact the user; e-mail, write, and talk are a minimum. Usually, the idle user will detach if he is done. However, if no response is returned within a reasonable amount of time, the command *cmdetach* can be used to forcibly detach any user from the Connection Machine. This command is executed at the UNIX prompt. CTRL-Z can be used to temporarily exit LISP and return to UNIX in order to detach the idle user.

From the (*cm:finger*) information shown above, we see that user GRUMPY has been idle for over one hour. After numerous attempts to contact GRUMPY failed, we detach him. We can then attach to the released resources and begin our simulations.

```
> ~Z
Stopped

cmns-sun.think.com% cmdetach grumpy

CM      Seqs Size  Front end I/F  User  Idle  Command
-----
NEARNET-CM2 0-1 16K  cmns-sun  0  rajan  0h 00m "starlisp.patch3
"
                                     rajan  0h 00m "cmattach"
NEARNET-CM2 3   8K  cmns-sun  1  grumpy  0h 03m "cmattach"
NEARNET-CM2 --- ---  cmns-vax  1  (nobody)
NEARNET-CM2 2   8K  cmns-vax  0  linchih 0h 14m "cmattach"
64K memory, 64-bit floating point
framebuffer on sequencer 0
CMIOCs on sequencers 0 1 2 3
no free sequencers on NEARNET-CM2

cmdetach: You are about to try to detach user "grumpy"
from Connection Machine "NEARNET-CM2".

Proceed? [yn] y

cmns-sun.think.com% fg
/usr/cm/bands/starlisp

(cm:finger)

CM      Seqs Size  Front end I/F  User  Idle  Command
-----
NEARNET-CM2 0-1 16K  cmns-sun  0  rajan  0h 00m "starlisp.patch3
"
                                     rajan  0h 00m "cmattach"
NEARNET-CM2 --- ---  cmns-sun  1  (nobody)
NEARNET-CM2 --- ---  cmns-vax  1  (nobody)
NEARNET-CM2 2   8K  cmns-vax  0  linchih 0h 00m "cmattach"
                                     linchih 0h 00m "main"
64K memory, 64-bit floating point
framebuffer on sequencer 0
CMIOCs on sequencers 0 1 2 3 (seq 3 is free)
1 free seq on NEARNET-CM2 -- 3 -- totalling 8K procs

> (cm:attach)
8192
>
```

Note that most Connection Machine facilities have some kind of scheduling system whereby computer time and resources (*i.e.*, processors) can be reserved in advance. This schedule should be inspected before detaching any user regardless of idle time.

5. SIMULATION DETAILS

5.1. Loading the Topography

The topography must be described with extreme care since the program does no input verification whatsoever. Incorrect loading of the processors with the topography information can easily cause the simulation to crash, hang, or give completely incorrect results. An example of incorrect topography information would be to specify a 2 μm wide domain (LINE 2) but then load a geometry that extends beyond 2 μm or only extends to 1 μm . In this situation, the simulation will run but may never reach steady state or will reach steady state but give incorrect answers. It is always up to the user to verify that the simulation performed correctly by confirming that the diffraction efficiencies are valid and graphically inspecting the *e-val.o* or *m-val.o* files.

The key point to remember when describing the topography is that each segment of the layer surface literally fills the processors beneath it with the material constant (*i.e.*, refractive index) of that layer. Therefore the order in which the layers are defined is very important. When a layer description is loaded into the processors, its material constant will overwrite any existing values for processors which are "covered" by the new layer's surface.

One very important limitation in this loading algorithm is that since all the processors below a given layer segment become part of that layer, movements in the negative x direction will always cause an improper loading of the topography. For the case where a negative movement along x seems necessary, breaking the layer description into different components can usually yield the proper topography while avoiding negative movements. There is no limit to the number of layers that can be specified. The program takes advantage of the natural parallelism found when loading topography having horizontal layer segments by simultaneously loading all the processors beneath the segment. For sloped layer segments, the intersection of the layer boundary and the processor in each column of the grid must be calculated. The time it takes to load the topography only becomes significant for topographies with many sloped segments loaded onto large grids.

Since it is critical that the topography be loaded properly in order that the simulation be correct, there is a two step process of verifying that the input file is correct. Appendix D, "TEMPEST Topography Plotter", describes a program that reads the input file *topog.d*, strips off all input lines except for the topography data, and creates another file that can be plotted with an X-Y plotting program. The user can then visually verify that the topography was designed correctly. However this method cannot catch all topography description errors. Appendix D explains in detail, with examples, how some mistakes can be made and the best way to detect them.

5.2. The Gaussian Startup

The incident excitation is generated at the top boundary of the simulation domain by forcing the electric field to oscillate sinusoidally. To reduce the effects of transients caused by an instantaneous field being applied to a previously unperturbed system, a Gaussian curve is applied to the input to cause a gradual turn-on from a value close to zero (input LINE 27) to the value of the sinusoid. This Gaussian matches the value of the sinusoid and its derivative at the iteration specified in the input LINE 26. The equations describing the Gaussian are explained in detail by Gamelin.² (§3.6) The initial value of the Gaussian is also an input parameter (LINE 27) and is typically specified as 1.0e-10. Figure 1 shows a vertical cutline of the instantaneous electric field after one complete cycle of excitation. The Gaussian curve can be observed at the front of the wave. As explained in the next section, convergence of TEMPEST simulations depends upon scattering from the topography. Hence, reducing the effects of transients caused by a sudden turn-on should reduce the time to reach steady-state.

5.3. Steady-State Convergence

It has been shown that steady-state convergence with this time-domain algorithm depends on the scattering of multiple reflections from the specified topography.⁹ The time to reach steady-state varies with the material constants, physical dimensions and the geometries of the scatterers. Steady-state is achieved when all transient waves die out. TEMPEST determines when the system has converged by comparing the electric field value from one period to the next at every node along a horizontal cutline near the top boundary in the simulation domain. (The cutline will be at the bottom boundary when transmitted analysis is specified.) The maximum difference at any node along the cutline cannot exceed a specified fraction of the incident field amplitude. This condition can be expressed as

$$\max_{i=0-x_{\max}} \left| E_z^t(i,j) - E_z^{t+Ndt}(i,j) \right| < f_{error} \times \left| E_z^{inc} \right| \quad (5.3.A)$$

where t is the instant in time, Ndt is the period of the incident wave, f_{error} is a user definable error tolerance [see LINE 23], and j is held constant, indexing the row corresponding to the cutline. To ensure maximum sensitivity, the fields are checked beginning at a time t_0 when the incident electric field is a maximum at the cutline. After several periods of excitation have been allowed to propagate, this criterion is monitored and must be satisfied for three consecutive periods to ensure the system has converged to its steady-state condition. Based on the experience of hundreds of simulations, roughly 80% of the simulations performed required less than 50 cycles of excitation to reach steady-state.

5.4. TEMPEST Limitations

There are a few issues in the algorithm which place some restrictions on the possible simulations the program can perform. Accuracy and stability are the primary concerns.

5.4.1. Courant Condition

For algorithmic stability, the time and spatial steps, dt and dx must be related by the Courant condition:¹⁵

$$v_{\max} dt \leq \frac{dx}{\sqrt{2}} \quad (5.4.A)$$

where v_{\max} is the maximum phase velocity within the domain. The velocity is computed as c/n where n is the largest index of refraction of any dielectric layer making up the topography and c is the speed of light in a vacuum. The Courant condition effectively prevents the speed of the electromagnetic wave from exceeding the speed of light. This relationship between the time and spatial discretizations is satisfied by choosing the spatial unit dx such that an appropriate grid density is achieved. Typically, a minimum grid density greater than 10 is required:

$$\frac{\# \text{ of grid nodes}}{\lambda_{in \text{ densest medium}}} > 10 \quad (5.4.B)$$

where $\lambda_{in \text{ densest medium}}$ is defined as $\lambda_0 / n_{largest}$. To compute this ratio given in equation (5.4.B), the following relationship may be used:

$$\frac{\# \text{ of grid nodes}}{\lambda_{in \text{ densest medium}}} = \left[\frac{\lambda_{vacuum} [\mu\text{m}]}{\text{Re}(largest \text{ ref. index})} \times \left[\frac{\# \text{ of processors in } X \text{ dimension}}{\text{width of domain } [\mu\text{m}]} \right] \right] \quad (5.4.C)$$

As an example, if the operating wavelength is 546.1 nm and the topography is comprised of air, polysilicon, oxide and silicon, the densest medium is polysilicon ($n=4.14$, $k=-0.032$). The wavelength in poly is then $\frac{\lambda_{air}}{n_{poly}} = \frac{546.1}{4.14} = 131.9$ nm. Using this value, the width of the domain and the number of processors used must be selected to meet the Courant Condition. One choice might be to use a 512 x 256 grid to simulate a 4 μm x 2 μm structure. This will provide a grid density of 16.88 nodes/ $\lambda_{in \text{ poly}}$ in the polysilicon. This meets the minimum requirement of 10. Accuracy will improve as grid density is increased. Note, however, that the

grid density is also constrained by the available resources (*i.e.*, total number of processors). The user must use the number of available processors and virtual processor capability effectively to achieve both stability and accuracy.

5.4.2. Materials

As mentioned in the description of LINE 20, the magnitude of the real part of the refractive index must be greater than the magnitude of the extinction coefficient ($k \geq n$ at the wavelength of interest). The direct effect of ignoring this restriction is a negative value for the dielectric permittivity ($\epsilon = \text{Re}(\tilde{n}^2) = n^2 - k^2$). As a result, the simulation becomes numerically unstable as the electric field begins to grow rapidly. Some hypotheses of the cause for this instability are discussed by Gamelin,² (§5.3)

Large extinction coefficients are often encountered when using highly conductive materials such as aluminum or chromium. In many cases, simulating these metals as perfect electrical conductors (PEC) provides a reasonably accurate approximation. Perfect Electrical Conductors are implemented by using (1.0e10 -9.9999999e9) for the refractive index of that layer. The nature of this material is such that the electric field drops to zero in the layer except for some skin effect. Experimentation with TEMPEST has shown that the user should ensure that at least 4 grid nodes exist within the conductive layer to ensure the E-field drops to 0.0. This 4-layer constraint is necessary for the greater wavelengths (e.g, 640nm). Shorter wavelengths (248nm) do not penetrate the PEC as far.

5.5. Hints and Explanations

There are several simulation issues that are important for understanding how TEMPEST simulation results should be interpreted and practical hints that may help new users to become proficient with TEMPEST more quickly. These are listed below in no particular order.

- The electric field (E_z) is linearly polarized, perpendicular to the 2-dimensional plane in which the topography is defined. Therefore the electric field vector is always parallel to the edge of the given structure (*i.e.*, TE polarization). The magnetic field components (H_x , H_y) are defined in the simulation plane.
- The incident illumination is currently limited to normal incidence. The simulation domain is excited by forcing the electric field along the top boundary to oscillate sinusoidally.
- When the incident plane wave is modulated with an aerial image (such as for bleaching photoresist) the image is assumed in focus at the top boundary since that is where the incident field is defined. By adjusting the topography so that the surface of the resist is physically near the top boundary, the effects of image diffraction will be reduced.
- When bleaching resist, the incident excitation is driven until steady-state is reached. The photactive compound (PAC) concentration and therefore the absorption coefficient in all nodes located within the resist are modified according to Dill's ABC model for the fractional energy dose simulated. The electric and magnetic fields are reset to zero and the process is repeated until the sum of all energy doses equals the total specified in the input file.
- When performing simulations for the purpose of optical imaging, verify that the top surface of the topography is at least 2 to 3 λ / NA from the top boundary so that only the far field is used to compute the diffraction efficiencies which are in turn used to compute an aerial image.

- The accuracy in defining topography dimensions is limited to the spatial discretization unit dx .
- The 3 output files will be overwritten with each simulation run. Prevent loss of data by renaming *e-val.o*, *m-val.o*, and *de.o* after each run.

5.6. Boundary Conditions

The three versions of TEMPEST are completely independent of each other with respect to source code. Each version is distinguished by the type of boundary conditions that it employs along the four different boundaries. A brief explanation of each version is given here.

5.6.1. Periodic

The periodic version applies no specially conditions along the east or west boundary of the domain since the Connection Machine processors along the boundaries of a two dimensional grid are physically adjacent. Hence, with no additional computational effort, a periodic boundary exists. The user must ensure that the topography is defined such that the structure is periodic in nature, *i.e.*, segment endpoints existing at the west boundary must exist at the same vertical location at the east boundary. An inadvertant discontinuity in the topography at the east/west boundary will adversely affect the simulation results. The north and south boundaries apply specially formulated conditions to absorb outwardly propagating waves.

5.6.2. Isolated

The isolated version applies absorbing conditions along the east and west boundaries identical to the absorbing conditions applied along the north and south boundaries. Waves impinging upon any of the boundaries effectively escape the domain and propagate to infinite. No limits are imposed on the shape of the topography. The result of applying four absorbing boundary conditions is increased computational time. It is wise to take advantage of any existing symmetries and reducing so that it may be simulated using either the periodic or symmetric versions.

5.6.3. Symmetric

The symmetric version is a modification of the isolated version. Absorbing boundary conditions are applied along the north, south and west boundaries. Symmetry is defined along the east boundary. The input topography should be the "left half" of the full symmetric structure. The electric and magnetic field matrices generated upon simulation completion correspond only to the half-structure that was simulated.

6. A BRIEF HISTORY & ACKNOWLEDGEMENTS

The material compiled into this user's guide represents the cumulative efforts of many people who have helped evolve this exciting new approach in electromagnetic scattering simulation using massively parallel computing. John Gamelin is credited with developing, debugging and testing the original code. Roberto Guerrieri brought his expertise with massively parallel algorithms to help define an efficient discretization of the Maxwell equations. Andy Neureuther provided wide scale vision with endless ideas of applications, enhancements, and back-breaking tests of its performance. Karim Tadros put the program through its paces by measuring its performance and performing two in-depth studies of optical scattering phenomena. Takeshi Doi and Alfred Wong also helped verify the program's capabilities by tackling issues in electromagnetic scattering and optical imaging at a level not previously investigated with simulation. And now, with several successful studies supporting this unique approach, Alfred Wong has taken on the challenge to extend the program's capabilities while eliminating some of its weaknesses. This work was supported by the SRC / SEMATECH Center of Excellence on Optical Lithography and Pattern Transfer (88-MC-500).

REFERENCES

1. J. Gamelin, R. Guerrieri and A.R. Neureuther, "Exploration of Scattering from Topography with Massively Parallel Computers," *J. Vac. Sci. Technol. B.*, vol. 7, no. 6, pp. 1984-1990, Nov/Dec 1989.
2. J. Gamelin, "Simulation of Topography Scattering for Optical Lithography with the Connection Machine," M.S. Thesis, Memorandum No. UCB/ERL M89/71, University of California, Berkeley, May 1989.
3. *Connection Machine Model CM-2 Technical Summary*, Thinking Machines Corporation, April 1987.
4. Guy Steele, *Common Lisp the Language*, Digital Press, Burlington, MA, 1984.
5. W. G. Oldham, S. N. Nandgaonkar, A. R. Neureuther, and M. M. O'Toole, "A General Simulator for VLSI Lithography and Etching Processes; Part I - Applications to Projection Lithography," *IEEE Trans. Electron Devices*, vol. ED-26, no. 4, pp. 14-16, April 1979.
6. Kenny K.H. Toh. PDRAW, DRAWPLOT, and CONTOUR: X-window plotting programs developed at UC Berkeley and supplied with the SAMPLE, SPLAT, and TEMPEST program tapes.
7. K. Tadros, A.R. Neureuther, J. Gamelin and R. Guerrieri, "Investigation of Reflective Notching with Massively Parallel Simulation," *Proc. SPIE: Optical/Laser Microlithography III*, vol. 1264, pp. 322-332, March 1990.
8. K. Tadros, A.R. Neureuther and R. Guerrieri, "Understanding Metrology of Polysilicon Gates through Reflectance Measurements and Simulation," *Proc. SPIE: Integrated Circuit Metrology, Inspection and Process Control V*, vol. 1464, March 1991.
9. R. Guerrieri, K. Tadros, J. Gamelin, and A.R. Neureuther, "Massively Parallel Algorithms for Scattering in Optical Lithography," *IEEE Trans. CAD*, to appear.
10. A. Wong, T. Doi, D. Dunn, and A.R. Neureuther, "Experimental and Simulation Studies of Alignment Marks," *Proc. SPIE: Optical/Laser Microlithography IV*, vol. 1463, March 1991.
11. T. Doi, K. Tadros, B. Kuyel, and A.R. Neureuther, "Edge-Profile, Materials and Protective Coating Effects on Image Quality," *Proc. SPIE: Integrated Circuit Metrology, Inspection and Process Control V*, vol. 1464, March 1991.
12. K.H. Toh and A.R. Neureuther, "Identifying and Monitoring Effects of Lens Aberrations in Projection Printing," *Proc. SPIE: Optical Microlithography VI*, vol. 772, pp. 202-209, 1987.
13. F. Dill, W. Hornberger, P. Hauge, and J. Shaw, "Characterization of Positive Photoresist," *IEEE Trans. Electron Dev.*, vol. ED-22, no. 7, pp. 445-452, July 1975.
14. G. Mur, "Absorbing Boundary Conditions for the Finite Difference Approximation of the Time-domain Electromagnetic-field Equations," *IEEE Trans. Electromag. Compat.*, vol. EMC-23, no. 4, pp. 377-382, Nov 1981.
15. A. Taflov and M. E. Browdin, "Numerical Solution of Steady-state Electromagnetic Scattering Problems Using the Time-dependent Maxwell's Equations," *IEEE Trans. Microwave Theory Tech.*, vol. MTT-23, no. 8, pp. 623-630, Aug 1975.

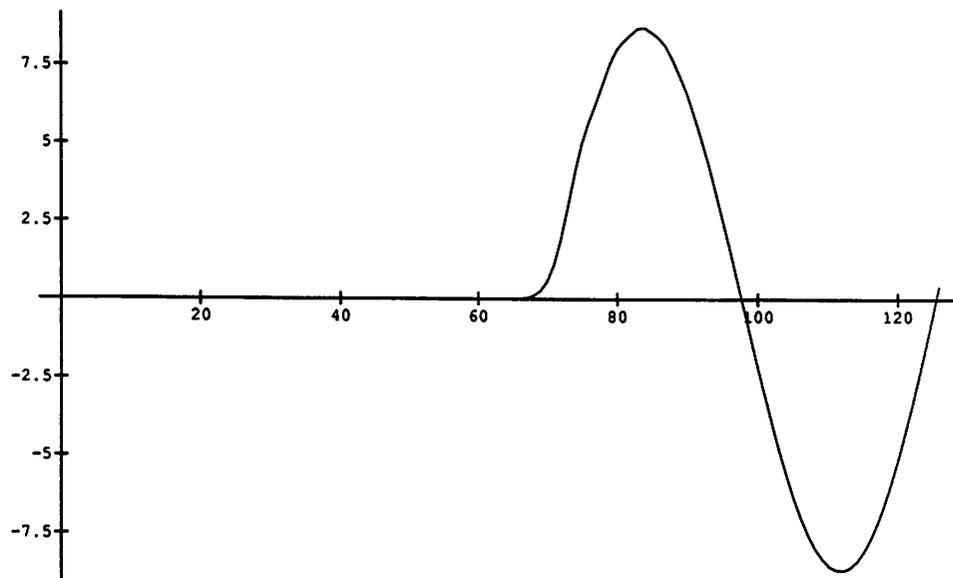


Figure 1 One-dimensional cross section of the incident, sinusoidal electric-field showing the effects of the Gaussian curve applied at the start of the plane wave to reduce transients. The bottom of the domain is located at 0 along the horizontal axis and the top of the domain is located at 128. The electric field has been excited for slightly more than one full wave cycle. [see Section 5.2, "The Gaussian Startup"]

APPENDICES

APPENDIX A

Connection Machine Facilities

The following list of Connection Machine facilities was obtained from David Ray <ray@Think.COM>, Customer Support Group, Thinking Machines Corporation. Already I've found that the facility at NCSA does not consider itself to be a "public" access site. NCSA requires peer review of a written statement describing the academic research for which the requester's CM time will be used. Research other than academic is subject to a fee. The other facilities listed below may require similar methods for obtaining time on their CM. At the time this list was compiled, no information was available regarding the available resources at each facility. From personal experience, I know that the Thinking Machines Corporation Pilot Facility offers a 32K CM-2 accessible by both VAX (Ultrix) and SUN-4 front-end systems.

Thinking Machines Corporation	
Connection Machine Network Server (CMNS) Pilot Facility	
CMNS Facilities Manager	phone: (617) 876-1111 x2121
Thinking Machines Corp.	email: cmns-manager@think.com
245 First Street	uucp: ames!think!cmns-manager
Cambridge, MA 02142-1214	

This facility is supported under terms of DARPA contract DACA76-88-C-0012
05/24/88 for use by the Internet community.

Naval Research Lab [NRL] [Code 51-30]	
Address:	4555 Overlook Ave., SW Washington, DC 20375-5000
Phone:	202-404-7020, -7029
Contacts:	Henry Dardy (202)404-7028 dardy@cmvax.nrl.navy.mil CM-Room - (202) 767-9044 or 45 Charlie Shirron

NASA Ames Research Center	
Address:	Mail Stop T045-1 Moffett Field, CA 94035
Phone:	Bruce Blaylock (415) 604-4400 Dave Gambrel-facilities mgr(-4504)

NCSA -- National Center for Supercomputing Applications

Address: Beckman Institute
National Center for Supercomputing Applications
405 No. Mathews St.
Urbana, IL 61801
Attn: Gerry Quinn 5209 Beckman

Contact: Gerry Quinn (217) 244-1878
<Gquinn@ncsa.uiuc.edu>

Northeast Parallel Architectures Center - NPAC

Address: NPAC - Syracuse University
111 College Place
Science & Technology Bldg.
Machine Room 1-213
Syracuse, NY 13244-1240

Phone: (315) 443-1722
FAX: (315) 443-1973
Computer room: (315) 443-4535

Contact: Don Hewitt, Support, (315) 443-2086 desk 'bdc100289
Use main # (315) 443-1722
don@nova.npac.syr.edu

□

APPENDIX B

Generating "mask.in"

Currently, TEMPEST is limited to simulating transverse electric (TE) polarized illumination impinging at normal incidence from the top of the 2-dimensional simulation domain. Nominally, the excitation is a sinusoidal plane wave having a wavelength defined by the user in LINE 3 of the input file. The plane wave can be amplitude modulated by image profiles computed by SAMPLE or SPLAT to effectively produce a masked pattern. This feature is particularly useful for bleaching photoresist. To use this feature, an image profile must first be generated then placed in a file named *mask.in* in the working directory. This file must be comprised of 50 (x y) data pairs representing a normalized image profile. All other lines in this text file must be deleted.

B.1. SAMPLE-Generated Image

The following is a typical example of a SAMPLE input file to generate a positive tone image of a 1 μm wide line centered in a 4 μm wide periodic domain.

```
###
### 1.00um line imaged in a 4um domain at 0.436 lambda0
###
proj 0.60                ;# numerical aperture
lambda 0.436i            ;# wavelength
parcohref 0 0.7 0        ;# partial coherence and defocus
linespace 1.0 3.0        ;# 1um line, 3um space; line/space pattern
horwindow 4 2.5          ;# define a 4um window with line edge at 2.5um
optimgexp 1 0 1 0 0     ;# generate image intensity data file
imagerun                 ;# run image machine
```

The same image can also be generated by replacing the lines containing *linespace ...* and *horwindow ...* with any the following input lines.

```
line 1.0                 ;# a single line mask of 1.0um
horwindow 4 2.5          ;# define a 4um window with line edge at 2.5um
irregumask 0.0 1.5 1.0 3.0 ;#complex mask
phasemask (1.0 0.0 3.00) (0.0 0.0 1.00) (1.0 0.0 3.00) ;
                        # 1.5um wide transparent regions surrounding
                        1um wide opaque
```

SAMPLE will process the input file and generate a file named "f77punch7" in the working directory. Delete the few lines at the beginning and end of this file that are not part of the 50 (x y) data pairs. Then rename the file to *mask.in* and place it in the TEMPEST working directory.

B.2. SPLAT-Generated Image

Although currently TEMPEST is limited to 2-D propagation of the electromagnetic fields, a more realistic simulation can be achieved for features such as contact holes which add another dimension to the projected image profile. For these instances, the 2-D aerial image simulator, SPLAT, can be used to generate an accurate 2-D image profile from which a one-dimensional outline can be extracted and fed to TEMPEST for scattering analysis. The SPLAT input file for simulating the aerial image of a positive tone 1.0 μm^2 contact hole is given below.

```

# Using SPLAT to generate 2-D image profile for square contact hole.
# Only a 1-D cross section of the image will be used for input to TEMPEST.

Trial 1 :      printlevel = 3                ;# print out full diagnostics
Trial 2 :      lambda = 0.436 um            ;
Trial 3 :      NA   = 0.38                  ;
Trial 4 :      Defocus= 0.0 um              ;
Trial 5 :      Sigma = 0.5                  ;
Trial 6 :      mask = 2.0um x 2.0um        # opaque working area
&              at 0 transmittance          ;
Trial 7 :      cutout = (0.0, 0.0)         # Define contact opening
&              0.5 x 0.5 at 1 transmit    ;# Total of 1x1 opening
Trial 10:                                           ;# Compute Fourier Coeff
Trial 14:      cutline (-2,0) to (2,0)
&              to 'mask.in'                ;# Generate I cutline file
Trial 0 :      end                          ;

```

The second-to-last input line (TRIAL 14) grabs a horizontal cutline across the mask opening and writes the intensity data along that line into the file "mask.in". To incorporate this file into the simulation, LINE 16 of the TEMPEST input file must be set to 't' (*i.e.*, apply mask = true), and the user must make sure that the width of the domain simulated by TEMPEST is identical to the horizontal dimension of the image profile.

B.3. Advanced Uses

With some creativity, a variety of more elaborate excitations can be implemented. For instance, to simulate illumination by a scanning-slit optical microscope, the SAMPLE image of a 1 μm wide space can be used to illuminate the topography at several horizontal locations iteratively. The optical image profiles from each simulation can then be computed using the TEMPEST image synthesizer (Appendix C) and added together to form the final composite aerial image. This was done by Wong[†] for investigating alignment mark issues.

□

[†] A. Wong, T. Doi, D. Dunn, and A. R. Neureuther, "Experimental and Simulation Studies of Alignment Marks," *Proc. SPIE: Optical/Laser Microlithography IV*, vol. 1463, March 1991.

APPENDIX C

TEMPEST Image Synthesizer

(Alfred Wong, 1990)

I. "image" Input and Output Files

To run **image**, the command line should be **image de.xxx** where **xxx** is any extension you want to give to the **de.o** file from **tempest**.

There should also be a file called **spec.xxx** in the same directory in which you are running **image**. The format of this **spec.xxx** file is explained below. The **.xxx** extension must be identical the the **de.xxx** extension.

After the image is calculated, two output files are generated: **image.xxx** which is a data file to be used as input to and X-Y plotting program and **info.xxx** which contains information such as the magnitude of the diffraction efficiencies, focus depth, maximum intensity, etc. This file is not used for any purpose other than being a concise form of the **de.o** file output directly by **TEMPEST**.

II. Description of the "spec.xxx" File

The specification file has five lines:

Line 1 *thetainc*

The angle of incident illumination. [degrees]

Line 2 *intensity*

The intensity of the light in milliJoules per centimeter square [mJ/cm²]. This input is necessary only for normalization.

Line 3 *nail naiu nacl nacu*

These represent the lower and upper limits of the numerical aperture of the illumination and the collection optical systems.

Line 4 *focus_l focus_h focus_delta*

Image computes all the images starting from a focus depth of *focus_l* to *focus_h* in steps of *focus_delta* inclusively. A value zero focus depth refers to the focus plane at the top boundary when using reflected energies and the bottom boundary when using transmitted energies. A negative focus depth refers to a focal plane measured from the boundary toward the inside of the domain, regardless of the reference boundary. **Image** then displays the image at each focal position and then prompts the user to enter the number of the image to be saved. The user may input 0 if no image is to be saved.

Line 5 *adjust_scale factor*

adjust_scale can be one of either two character strings, "normalize" or "multiply".

For "normalize", each of the image intensity value is normalized by the value *factor*. If *factor* is given the value zero, then the image intensity value will be normalized to the maximum of the curve.

For "multiply", each of the image intensity value is multiplied by the value *factor*.

APPENDIX D

TEMPEST Topography Plotter

A small C program was written for TEMPEST users to quickly verify that the lines of the input file describing the geometrical shape of the topography are reasonably correct. This routine is designed to give only a quick idea of what the final 2-dimensional topography will look like based on the segment endpoints in the input file *topog.d*. To use this program, copy the file *topog.d* to *shape.in*. At the unix prompt, execute the shape program by simply typing in its name *shape*. *Shape* will read the file *shape.in* then generate a file called *shape.out* which is ready to be plotted with a X-Y plotting program.

BEWARE: Sometimes, even though the plot of *shape.out* looks good, TEMPEST may not have actually loaded the processors with the topography as the plot might indicate. For this reason, the option to generate *cross.section* [*topog.d* input LINE 29] comes in handy. First and foremost, since TEMPEST employs a uniform and discrete grid, sloped layer boundaries, described by two endpoints with neither of their *x* and *y* components identical, will look like stair steps. Second, any movement in the negative *x* direction will always cause the processors to be loaded incorrectly. Hence, it is strongly recommended that all complex topographies be verified by viewing the *cross.section* file in a density plot (such as in Mathematica) or a similar format.

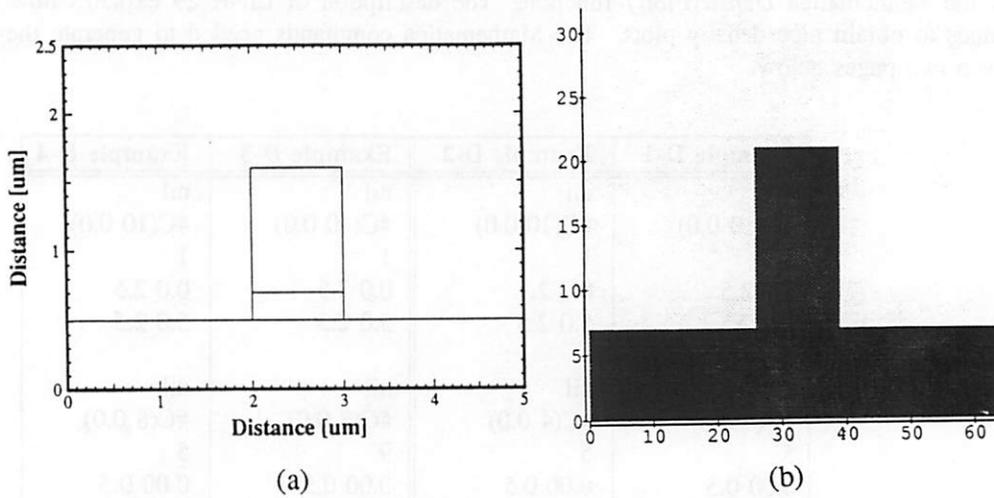
An example of an easily made error in defining the topography is shown below. The topography is of a polysilicon gate structure. The error is in the order in which the poly and oxide layers were specified. First the example input file containing the error is inspected (Example & Figure D-1), then the input file without the error is inspected (Example & Figure D-2). Only that part of the *topog.d* where the topography is defined is shown for clarity. The diagrams Figure D-1(a) and D-2(a) are plots of the output from the TEMPEST topography plotter (*shape*). The second diagram in each set (Figure D-1(b) and D-2(b)) are Mathematica density plots of the *cross.section* file generated by TEMPEST. Because of the way TEMPEST loads each layer, the oxide layer must be specified after the polysilicon layer.

A more complicated example of a topography file designed to represent a self-aligned phase shifting scheme is shown to make one other point. The result from the TEMPEST topography plotter appears to be incorrect when actually, the processors were loaded correctly. For this example, chrome, on a quartz plate, is overetched while being protected by photoresist which acts as the phase shifter. An initial attempt at creating the *topog.d* file might result in the one labeled Example D-3. (The dimensions have been exaggerated for easier viewing.) Figure D-3(a) suggests that the topography looks as if it were designed correctly. However, the density plot (Figure D-3(b)) tells the true story. The processors were loaded incorrectly. The error occurred when defining the phase shifter layer. Two of the segments "moved" in a negative horizontal direction. A correct topography description is given in Example D-4. Note that the output from *shape* (Figure D-4(a)) depicts a very awkward and seemingly undesirable topography. Yet the density plot (Figure D-4(b)) indicates that the processors were loaded properly. Understanding why requires that we understand how the topography loading algorithm works. Section 5.1 describes this process.

The following table contains only the lines comprising the topography description extracted directly from a full *topog.d* file. The refractive index values (denoted by *#C(nn kk)*) have been modified for use by the Mathematica *DensityPlot[]* function. The description of LINE 29 explains how to modify these values to obtain nice density plots. The Mathematica commands needed to generate the density plot are given two pages below.

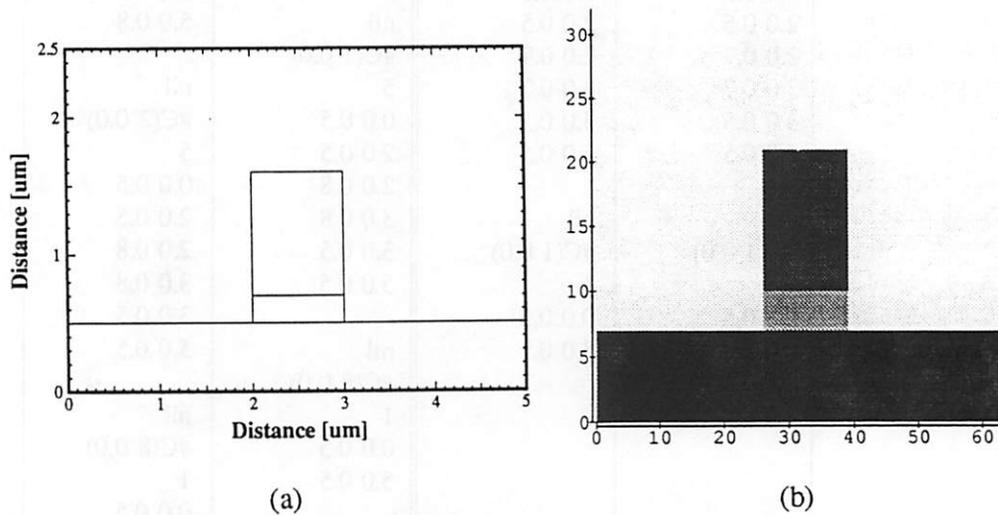
Example D-1	Example D-2	Example D-3	Example D-4
nil	nil	nil	nil
#C(10 0.0)	#C(10 0.0)	#C(10 0.0)	#C(10 0.0)
1	1	1	1
0.0 2.5	0.0 2.5	0.0 2.5	0.0 2.5
5.0 2.5	5.0 2.5	5.0 2.5	5.0 2.5
nil	nil	nil	nil
#C(4 0.0)	#C(4 0.0)	#C(8 0.0)	#C(8 0.0)
5	5	9	5
0.00 0.5	0.00 0.5	0.00 0.5	0.00 0.5
2.00 0.5	2.00 0.5	2.00 0.5	1.25 0.5
2.00 1.6	2.00 1.6	2.00 0.8	1.25 1.5
3.00 1.6	3.00 1.6	1.25 0.8	3.75 1.5
3.00 0.5	3.00 0.5	1.25 1.5	3.75 0.5
5.00 0.5	5.00 0.5	3.75 1.5	5.00 0.5
nil	nil	3.75 0.8	nil
#C(8 0.0)	#C(8 0.0)	3.00 0.8	#C(10 0.0)
5	5	3.00 0.5	1
0.0 0.5	0.0 0.5	5.00 0.5	0.0 0.8
2.0 0.5	2.0 0.5	nil	5.0 0.8
2.0 0.7	2.0 0.7	#C(1 0.0)	nil
3.0 0.7	3.0 0.7	5	#C(2 0.0)
3.0 0.5	3.0 0.5	0.0 0.5	5
5.0 0.5	5.0 0.5	2.0 0.5	0.0 0.5
nil	nil	2.0 0.8	2.0 0.5
#C(1 0.0)	#C(1 0.0)	3.0 0.8	2.0 0.8
1	1	3.0 0.5	3.0 0.8
0.0 0.5	0.0 0.5	5.0 0.5	3.0 0.5
5.0 0.5	5.0 0.5	nil	5.0 0.5
nil	nil	#C(8 0.0)	nil
nil	nil	1	#C(8 0.0)
nil	nil	0.0 0.5	1
nil	nil	5.0 0.5	0.0 0.5
nil	nil	nil	5.0 0.5

Figure D-1



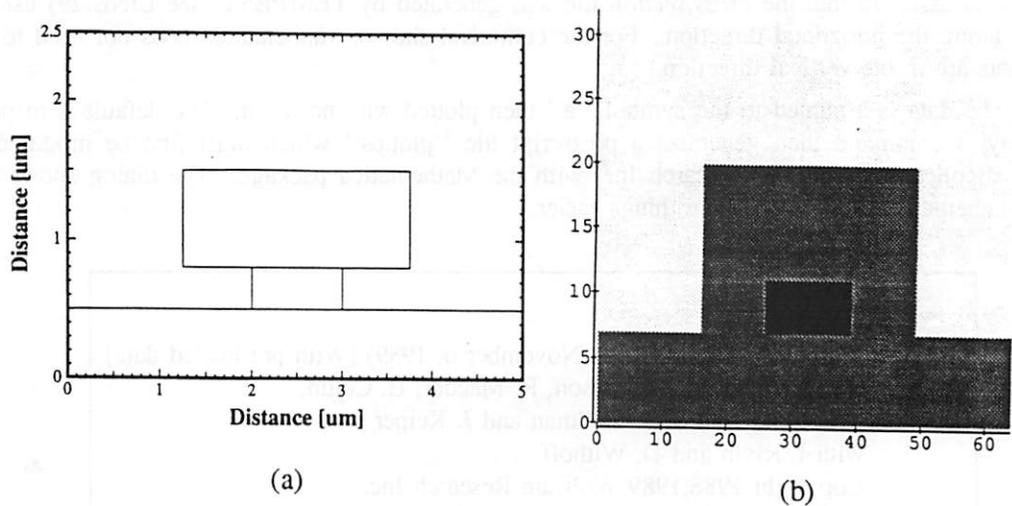
(a) X-Y plot of the topography described in Example D-1 using **shape**. (b) The corresponding Mathematica density plot. This topography was designed **incorrectly**.

Figure D-2



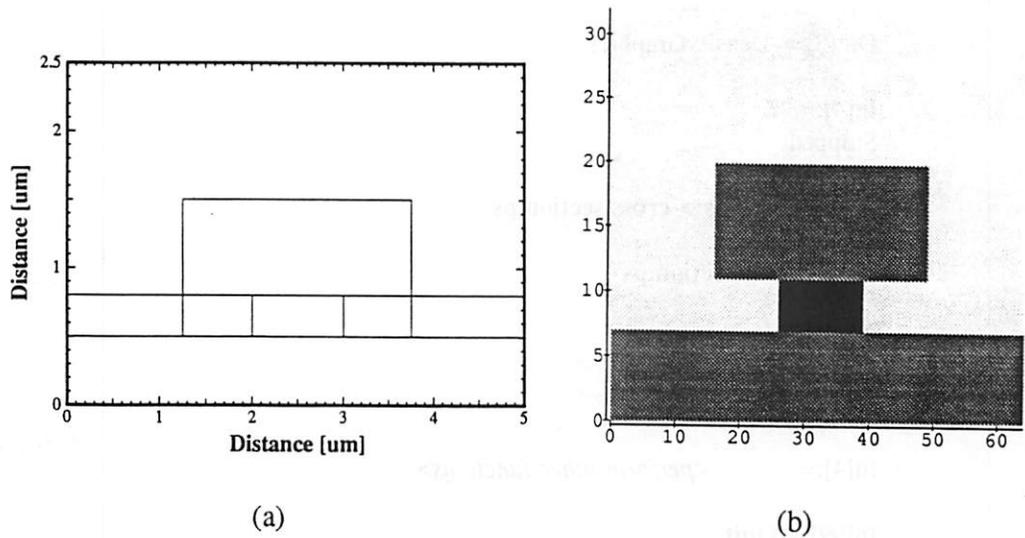
(a) X-Y plot of the topography described in Example D-2 using **shape**. (b) The corresponding Mathematica density plot. This topography was designed **correctly**.

Figure D-3



(a) X-Y plot of the topography described in Example D-3 using **shape**. (b) The corresponding Mathematica density plot. This topography was designed **incorrectly**.

Figure D-4



(a) X-Y plot of the topography described in Example D-4 using **shape**. (b) The corresponding Mathematica density plot. This topography was designed **correctly**.

The following is a brief dialog session showing how the density plots were generated using Mathematica. First, it is assumed that the *cross.section* file was generated by TEMPEST [see LINE 29] using a grid with 64 nodes along the horizontal direction. For the command shown, Mathematica does not need to know how many elements are in the vertical direction.

The data is assigned to the symbol "a" then plotted with no mesh. The default is to print a mesh. The `Display[]` command then generates a postscript file "plot.ps" which must first be modified by the program `psfix`, supplied by Wolfram Research Inc. with the Mathematica package. The dialog shows how to enter and exit Mathematica with `^Z` to make things easier.

```
% math

Mathematica (sun4) 1.2 (November 6, 1989) [With pre-loaded data]
by S. Wolfram, D. Grayson, R. Maeder, H. Cejtin,
  S. Omohundro, D. Ballman and J. Keiper
with I. Rivin and D. Withoff
Copyright 1988,1989 Wolfram Research Inc.
-- X11 windows graphics initialized --

In[1]:= a=ReadList["cross.section",Table[Number,{64}]];

In[2]:= ListDensityPlot[a, Mesh->False]

Out[2]= -DensityGraphics-

In[3]:= Display["plot.ps",%]

Out[3]= -DensityGraphics-

In[4]:= ^Z
Stopped

% psfix plot.ps > cross.section.ps

% lpr cross.section.ps

% fg
math
<press return>

In[4]:=      <perform other functions>

In[99]:= Quit

%
```

□

APPENDIX E

A Complete Example

All the input files needed to make a complete simulation are shown here as a simple case for the new user to test that TEMPEST is operating correctly. Using the *topog.d* shown below, will result in generating the *de.o* file shown on the next page. The diffraction efficiency data from that *de.o* and the *spec.o* files can then be used by the image synthesis routine to create the image shown by Doi[†] T. Doi, K. Tadros, B. Kuyel, and A. R. Neureuther, "Edge-Profile, Materials and Protective Coating Effects on Image Quality," *Proc. SPIE: Integrated Circuit Metrology, Inspection and Process Control V*, vol. 1464, March 1991. in his Figure 6 curve (a).

The *topog.d* file has been written in three columns only in order to keep it all on one page for viewing. Obviously, the same data should be written in one column for the purpose of running the simulation.

TOPOG.D		
512 512	nil	nil
2.0000	#C(1.0 0.0)	#C(1.0 0.0)
0.248	1	5
1.0	0.000 2.000	0.000 1.170
1.0	2.000 2.000	0.875 1.170
0.0		0.875 1.250
0.1	nil	1.125 1.250
0.000	#C(1.50 0.00)	1.125 1.170
1	1	2.000 1.170
0.51	0.000 1.750	
0.031	2.000 1.750	0.01
0.013		1000
1.0	nil	8000
t	#C(4.02 -2.11)	
trans	1	10
nil	0.000 1.250	1.e-10
0.0	2.000 1.250	8
3		

DE.O	
<i>(diffraction efficiency values only)</i>	
\$\$\$ Fields reached steady-state. MSE = 0.6358829 \$\$\$	
\$\$\$ Required 4244 iterations \$\$\$	
Total Dose = 0.	
# iterations = 4244	
diffraction efficiency,phase for order 0 is	1.161704E-2 , 1.551287E+0
diffraction efficiency,phase for order 1 is	9.551030E-3 , -1.795465E+0
diffraction efficiency,phase for order - 1 is	9.551031E-3 , -1.783193E+0
diffraction efficiency,phase for order 2 is	8.669677E-3 , 6.227425E-1
diffraction efficiency,phase for order - 2 is	8.669679E-3 , 6.472862E-1
diffraction efficiency,phase for order 3 is	7.360442E-3 , 2.557419E+0
diffraction efficiency,phase for order - 3 is	7.360443E-3 , 2.594234E+0
diffraction efficiency,phase for order 4 is	5.785786E-3 , -2.429382E+0
diffraction efficiency,phase for order - 4 is	5.785787E-3 , -2.380295E+0
diffraction efficiency,phase for order 5 is	4.462535E-3 , -1.791753E+0
diffraction efficiency,phase for order - 5 is	4.462532E-3 , -1.730396E+0
diffraction efficiency,phase for order 6 is	2.776490E-3 , -2.178977E+0
diffraction efficiency,phase for order - 6 is	2.776492E-3 , -2.105347E+0
diffraction efficiency,phase for order 7 is	1.384628E-3 , 2.169568E+0
diffraction efficiency,phase for order - 7 is	1.384632E-3 , 2.255468E+0
diffraction efficiency,phase for order 8 is	3.952697E-5 , 6.638409E-1
diffraction efficiency,phase for order - 8 is	3.952695E-5 , 7.620135E-1
total energy = 0.09167727663667231	
total overhead time = 222.58 seconds	

SPEC.O	
input parameter	comments (must be left out of file)
0.0	<i>theta_{inc}</i>
0.1	<i>intensity</i>
0.0 0.35 0.00 0.7	<i>NA_{illumin}^{lo} NA_{illumin}^{hi} NA_{collect}^{lo} NA_{collect}^{hi}</i>
-1.25 -1.15 0.01	<i>focus_{lo} focus_{hi} focus_{step}</i>
normalize 1.00	<i>scale factor</i>

□