SPECIAL ISSUES IN SEMICONDUCTOR MANUFACTURING

V

EECS 290W Class Projects Reports, Spring 1994

Professor:

Costas J. Spanos

Students:

James Chen, Michael Cohn, Mark Hatzilambrou,
Antonio Miranda, and Richard Schenker

SPECIAL ISSUES IN SEMICONDUCTOR MANUFACTURING

V

EECS 290W Class Projects Reports, Spring 1994

Professor:

Costas J. Spanos

Students:

James Chen, Michael Cohn, Mark Hatzilambrou,
Antonio Miranda, and Richard Schenker

ELECTRONICS RESEARCH LABORATORY

Preface

This is the fifth annual edition of the 290W report. This edition includes descriptions of projects completed during the Spring semester of 1994, in the context of the graduate course "Special Issues in Semiconductor Manufacturing". Five students have participated, and according to the course requirements, these students worked with me on their projects during the last six weeks of the semester.

Each of the presented projects covers at least one novel aspect of semiconductor manufacturing. The first discusses experimental modeling of micro self-assembly. The second deals with the control of pulsed laser energy for photolithographic applications. The third presents a detailed statistical process control scheme for a polysilicon etcher. The fourth solves the problem of statistically valid worst case characterization of BSIM3 models. Finally, the last projects investigates the possibility of process characterization based on high level integrated circuit behaviors.

It is my hope that these reports will add to our understanding of semiconductor manufacturing. My thanks go to the 290W students whose work made this document possible. I am also grateful to the personnel and management of the Berkeley Microfabrication laboratory for their help with the experimental part of the projects presented here.

Costas J. Spanos

August, 1995

Table of Contents

Electrostatic Self-Assembly of Microfabricated Structures: Factorial Optimization

Michael Cohn

The term "self-assembly" has been applied to spontaneous ordering processes such as crystal and polymer growth. In these processes, random thermal vibration causes a system of atoms or molecules to evolve in a way that minimizes its potential energy. Recently, this principle has been proposed for use in microelectronics manufacturing, for rapid assembly of large-scale hybrids. A demonstration system has been constructed. The rate of self-assembly in this system is analyzed with factorial experiments. Results are employed to optimize performance and model system behavior.

1.0 Introduction

Self-assembly has been proposed for the manufacture of hybrid circuits incorporating large numbers of devices [1], [2]. For example, one may wish to bond approximately 100 GaAs lasers onto a microprocessor die to provide optical interconnects; alternately, single-crystal silicon FETs could be placed on a plastic substrate for active-matrix liquid crystal displays.

The focus here will be on rapidly placing a number of small (400x400x200 micron) devices at predetermined sites on a substrate. To accomplish this, electrodes are lithographically defined on the substrate, so that an applied voltage results in electrostatic forces which can attract small particles in the vicinity of the electrodes. The substrate is vibrated to defeat static friction and to bring devices near the electrodes. The apparatus is diagrammed in Figure 1.
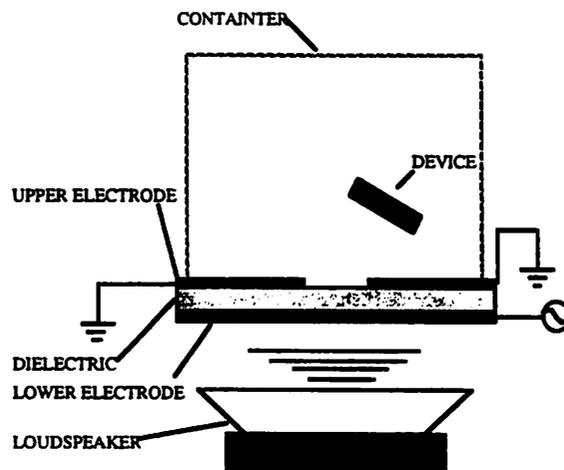
FIGURE 1. Apparatus.

The electrode design is shown in Figure 2. This design represents a parallel-plate capacitor, in which the upper plate has a small aperture. Fringing fields will attract a neutral particle by inducing electrical polarization. The force on the particle then parallels the field gradient. Binding of a single particle in the aperture shields subsequent particles from the field.

This concept borrows from thermodynamics and simulated annealing principles. As in these areas, equilibrium behavior can be modeled straightforwardly, depending only the potential energy change on binding and the average particle velocity. However, kinetics -- the rate of assembly in this case -- depend on many variables. These include the overlap of the electrode's potential well with the particle's velocity-space distribution. Orientational asymmetry of the particle and well must also be taken into account, as well as the particle's average spatial distribution within the container. Since the dependence is in general both complex and sensitive, assembly rate is a worthwhile candidate for statistical analysis and optimization.

2.0 Methodology

The binding time, τ, was measured as a function of electrode voltage V and vibrational amplitude A. Other parameters, such as vibrational frequency, were held constant.
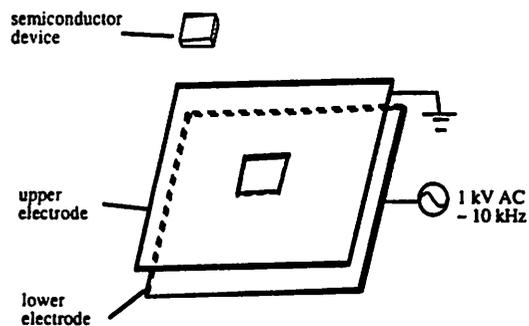
FIGURE 2. Electrode configuration.

Since the range of binding times extended below 1 second, an optical sensor was employed in conjunction with a recording oscilloscope to accurately time these events. (See Figure 3.) An AC high-voltage supply was employed to counter the effects of static charge buildup on the particle. Initial trials established that τ was highly sensitive to both control variables; therefore, measures were taken to ensure stability and repeatability. Electrode voltage was stabilized using a self-oscillating driver on the primary of the step-up transformer, in conjunction with an automatic gain control (AGC) circuit. (See Figure 3.) The loudspeaker was driven from the source output of an HP3562A Digital Signal Analyzer (402.6 Hz sinewave), via a power amplifier. The RMS electrode voltage was found to be stable to 0.06%, and the speaker drive to 0.33%, over a 10 minute period. Both noise levels were significantly below the sensitivity of the experiment.
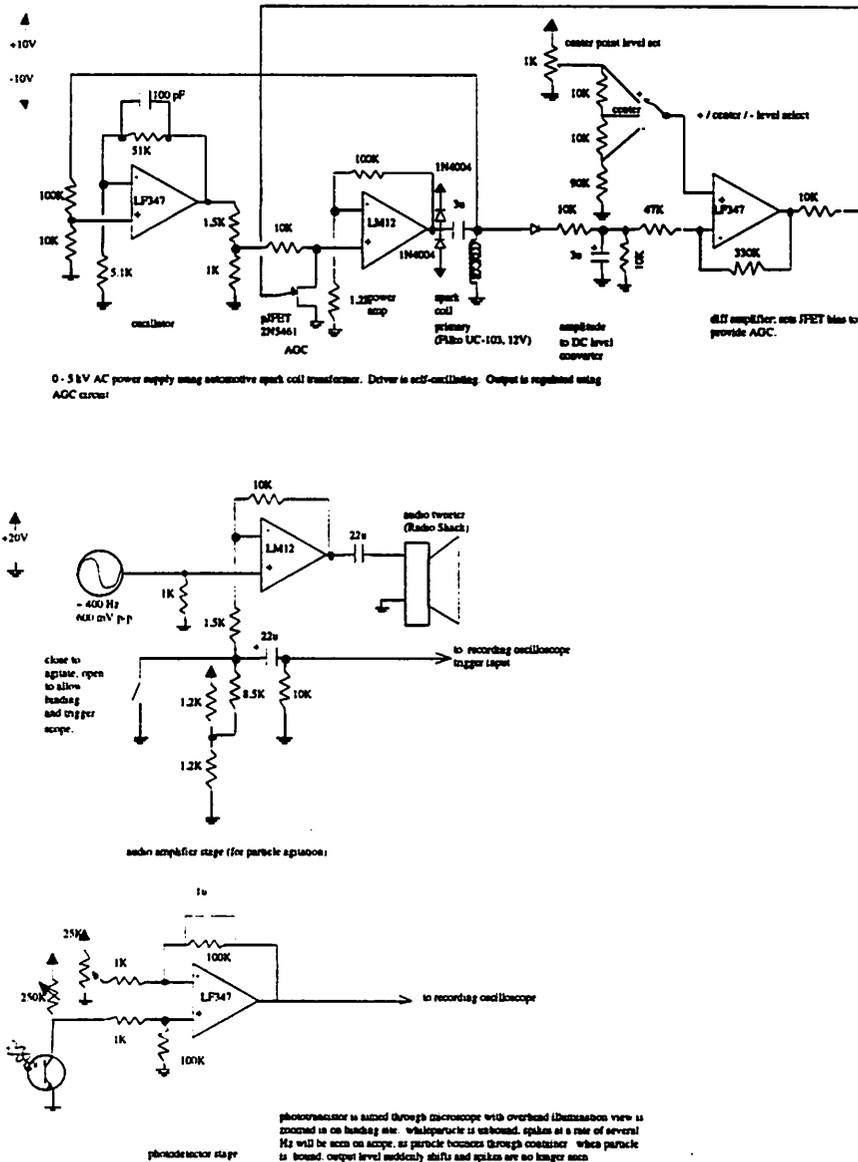
FIGURE 3. Schematics for high-voltage power supply, speaker amplifier, and optical sensor circuit.

The initial centerpoint was chosen to yield convenient binding times, on order of 30 seconds, so as to allow an acceptable number of trials. Also, electrode voltage had to be minimized (≤ 4 V at the testpoint), because otherwise the dielectric tended to break down in a matter of minutes otherwise.

A two-level factorial was subsequently run over both control variables, including the centerpoint (5 points total). 25 measurements were collected at each setpoint, on the basis of initial estimates of effects and the standard error. The magnitude and significance of each effect, as well as the interaction, were evaluated. A subsequent run of 70 measurements was performed at a separate setpoint to obtain more information on the distribution and to evaluate autocorrelation. A second five-point experiment was performed to optimize the process (minimize τ).

3.0 Results and Discussion

The data from the first factorial and the long run are given in Table 1.

	set point	1	2	3	4	5	run of 70	run of 70 cont	run of 70 cont
	<τ>	**28.63**	**58.27**	**27.50**	**70.34**	**10.86**	**42.49**		
	σ	**23.90**	**46.41**	**22.68**	**64.60**	**11.17**	**33.80**		
		14.530	89.172	19.170	69.214	9.094	18.060	106.250	23.187
		32.720	36.010	3.052	153.870	3.174	5.594	9.875	26.656
		5.000	74.800	51.697	28.380	3.967	18.590	71.531	45.406
	5 3	30.031	16.480	60.730	68.298	25.330	17.030	17.970	2.594
↑	1	1.812	98.633	7.568	50.354	1.221	70.312	169.940	43.437
V	2 4	64.000	52.856	57.312	77.637	23.500	60.031	19.719	17.530
	A->	34.060	130.740	16.720	110.050	38.330	59.800	27.562	23.060
		15.010	12.330	14.890	116.940	7.385	29.312	11.870	44.469
		30.580	26.670	48.650	24.840	31.130	46.562	90.750	30.406
		31.070	45.959	7.202	58.716	0.794	100.810	38.781	14.250
V (volts)	A (mV)	8.484	71.411	8.057	82.703	19.470	7.281	30.875	47.437
2.800	590	16.720	206.300	13.370	270.810	13.060	33.562	8.094	33.500
2.575	575	91.736	60.364	35.710	91.248	2.502	26.969	48.094	86.062
2.340	560	36.930	7.385	13.730	2.625	3.784	62.312	51.969	3.812
		29.600	82.031	2.197	49.969	1.831	96.219	56.750	62.562
		12.820	33.870	53.772	23.130	5.798	18.469	14.750	10.660
		8.057	13.610	63.538	164.730	4.639	23.156	40.062	50.344
		5.249	107.790	16.110	164.730	8.850	52.250	96.094	7.344
		13.060	52.063	2.747	61.523	3.540	17.310	42.312	29.906
		23.990	22.888	65.002	13.306	10.070	74.219	4.469	15.500
		12.760	7.507	56.213	8.179	5.676	70.094	114.370	
		44.860	72.205	39.49	2.625	35.03	108.810	60.500	
		18.921	87.708	17.82	27.1	6.287	105.440	49.094	
		43.150	8.301	8.423	28.748	3.418	15.090	3.031	
		90.576	39.73	4.395	8.789	3.662	23.156	11.060	

TABLE 1. Data from the first factorial. All measurements given in seconds.

As expected, the data followed a distribution which was essentially exponential, shown in Figures 4 and 5. This type of distribution is familiar from processes such as radioactive decay and (first-order) chemical reactions. For a generalized "reaction"

$$A \Rightarrow B$$

the probability of the reaction occurring in a time Δt is some fixed p. Then,

$$\frac{d[A]}{dt} = [A]\frac{p}{\Delta t}$$

Solving, we finde that the number of events occurring at time t is expected to be

$$-\frac{d[A]}{dt} = \frac{p}{\Delta t}[A]_0 \exp\left(-\frac{p}{\Delta t}t\right)$$
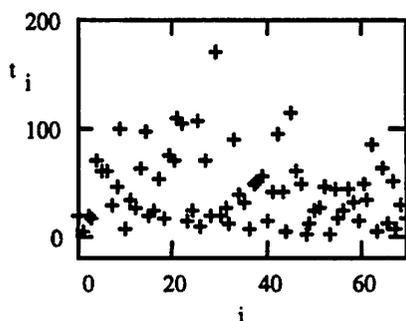
FIGURE 4. Time series from the run of 70. Binding time (seconds) vs. run number.
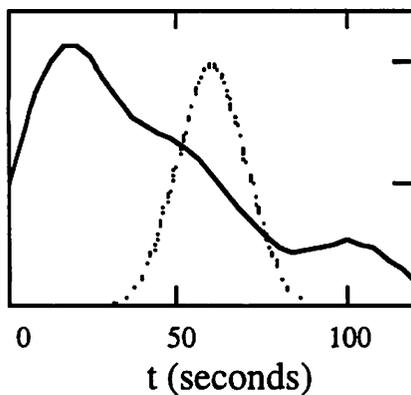
FIGURE 5. Histogram of untransformed data, showing sampling kernel (dotted line -- see Figure 8).
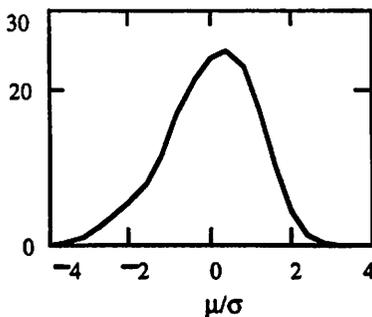
FIGURE 6. Histogram of log-transformed data.

In agreement with the exponential model [3], the standard deviation is not constant, but rather tracks the mean. Accordingly [4], the variance is stabilized by a log transformation. The averages and variances of the six transformed data sets are shown in Table 2

setpt	1	2	3	4	5	run of 70
$<\ln(\tau)>$	3.004	3.713	2.861	3.729	1.891	3.390
$\sigma(\ln(\tau))$	0.928	0.936	1.077	1.226	1.042	0.938

TABLE 2. Averages and variances for log-transformed data.

The significance of the various possible effects is evaluated on the basis of the transformed data, as required by the IIND assumption [5]. These are calculated below in Table 3. Note that the voltage effect was most significant -- beyond the 1% level. The amplitude and interaction effect were both significant at between the 5% and 1% level.

src of var	sum sqr	DOFs	Mean Sqr	Fo				y1..	140.1
A	6.1	1	6.1	**5.26**	5%	$Fo_{0.05,1,96} \sim 3.94$		y2..	164.8
V	45.2	1	45.2	**39.09**	1%	$Fo_{0.01,1,96} \sim 6.9$		y.1.	186.0
Interaction	5.7	1	5.7	**4.92**	5%			y.2.	118.8
Error	111.0	96	1.2					y...	304.8
Total	168.0	99						SSsubt	57.0

TABLE 3. Significance of effect calculations for the first factorial experiment.

Employing a basic regression model, $\ln(\tau)$ can be approximated by.

$$\ln(\tau) \approx E_0 + E_V(V - V_0) + E_A(A - A_0) + E_{A \cdot V}(A - A_0)(V - V_0)$$

The four parameters are extracted using four of the five setpoints, i.e.. the vertices of the square.

E_0	3.049
E_V	-2.689
E_A	0.986
$E_{V \cdot A}$	0.954

TABLE 4. Parameters of the linear model of $\ln(\tau)$ for the first factorial experiment.

Residuals are scatter-plotted as a function of setpoint, and also plotted in histogram and normal-scale form below in Figures 7-9.
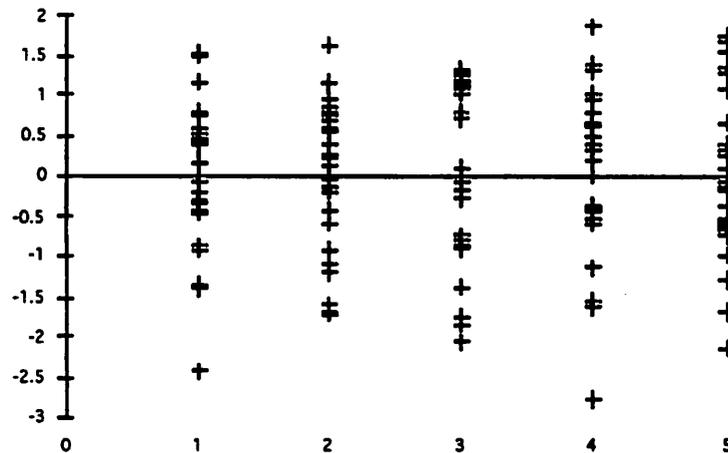
FIGURE 7. Residuals of the log-transformed data scatter-plotted versus setpoint.
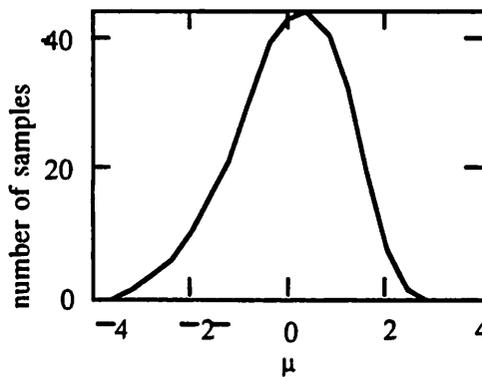
FIGURE 8. Histogram of the residuals of the log-transformed data (kernel width is 0.8). [1]

---

1. Note that the standard deviation of the data is almost exactly 1 (actually, 1.026; average is $3.2*10^{-7}$). The histogram plots the function

$$f(\mu) = \sum_i h(\mu - v_i)$$

where h is the normalized gaussian kernel, and v_i is the ith data point. This method acts to low-pass filter the data, and avoids aliasing which may occur with the more common square-wave filter [6].
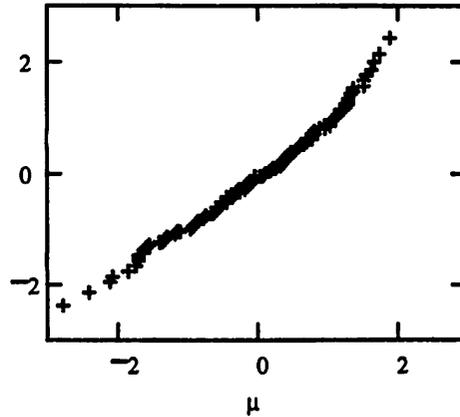
FIGURE 9. Normal-scale plot of the residuals of the log-transformed data.

Note that the centerpoint was not included in the parameter calculation, but nevertheless fits the model well, as evidenced by the small average of the residuals (Figure 8). The slight non-linearity of the normal-scale plot is not surprising, since the distribution of the original data is somewhat unusual. Specifically, the untransformed distribution is neither normal nor perfectly exponential, in that it takes on a zero value at t=0 because the particle is initially free (see Figure 5). Thus, the log transform may overcompensate slightly for the kurtosis.

3.1 Second Factorial Experiment -- Performance Optimization

The results for the second 5-point factorial are summarized in Table 5 below. The effective centerpoint is set as near as possible to #5 of the first factorial, though the exact setpoint was lost because of interceding repairs to the apparatus. Note, however, the significant improvement in binding time -- approximately 4.5 seconds, versus 11 in the previous experiment, comparing averages for the best setpoints in each. This is not unexpected, since a higher electrode voltage and less vibration favors capture of the particle.

set pt.	1	2	3	4	5
$<\tau>$	15.372	34.617	14.215	89.494	4.367
$\sigma(\tau)$	14.475	32.086	13.637	76.429	4.589
$<\ln(\tau)>$	1.969	3.071	2.217	4.083	1.073
$\sigma(\ln(\tau))$	1.985	1.087	1.090	1.016	0.883

TABLE 5. Summary of the results from the second factorial experiment.

3.2 Physical Model

The statistical results place constraints on the physical model. Specifically, the first experiment only shows an interaction in the transformed data; no interaction is seen in the untransformed data (see Table 6 below). This suggests that the variables appear as separate terms in a sum.

src of var	sum sqr	DOFs	Mean Sqr	Fo		
A	5151.3	1	5151.3	**2.96**	10%	Fo 0.1,1,96 ~ 2.76
V	50905.3	1	50905.3	**29.23**	1%	Fo 0.01,1,96 ~ 6.9
Interaction	130.7	1	130.7	0.08		
Error	167182.5	96	1741.5			
Total	223369.7	99				

TABLE 6. Significance of effects for untransformed data in the first experiment.

The classic model for this type of trapping event [7], [8], assumes particles of concentration n traveling with mean velocity v in the vicinity of a trap with areal cross-section σ. Conventionally, the trapping rate R is expressed

$$R = nv\sigma$$

τ in the experiment corresponds to the reciprocal of R. n is assumed constant, and v is assumed proportional to A. σ is estimated as follows: the electrostatic trapping site is modeled as a one-dimensional wire of length L, at a potential V (corresponding to the one-dimensional edges of the aperture in the upper electrode). The electric field will vary with radial distance from such a wire as

$$|E| \propto \frac{V}{r}$$

The potential energy of the particle is given by $-\varepsilon|E|^2 V_p$, where ε is the particle's dielectric constant and V_p its volume. For the particle to be captured, its net energy must be negative. Thus,

Potential Energy $\leq$ -Kinetic Energy

$$\left(\frac{V}{r}\right)^2 \propto \frac{1}{2}mv^2 \propto v^2 \propto A^2$$

and

$$r \propto \frac{V}{A}$$

σ is then approximated as a rectangle of height r and width L. Thus,

$$R = nv\sigma \propto nA\left(\frac{V}{A}L\right) \propto V$$

and

$$\tau \propto \frac{1}{V}$$

· This explains the relatively strong V effect, though not the small but significant A effect.

A simple way to introduce an A effect without an $A*V$ interaction is to posit another cross-section, σ', such that σ and σ' appear as series "conductances":

$$R = nv\sigma_{total}$$

$$\sigma_{total} \equiv \frac{1}{\dfrac{1}{\sigma} + \dfrac{1}{\sigma'}}$$

Then,

$$\tau = \frac{1}{R} = \frac{1}{nv\sigma_{total}} = \frac{1}{nv}\left(\frac{1}{\sigma} + \frac{1}{\sigma'}\right)$$

$$\propto \frac{1}{A}\left(\frac{A}{VL} + \frac{1}{\sigma'(A)}\right)$$

$$\propto \frac{1}{VL} + \frac{1}{A\sigma'(A)}.$$

The A effect occurs with the experimentally observed sign if, e.g., $\sigma' \propto 1/A^2$, yielding

$$\tau \propto \frac{1}{VL} + A$$

where σ' is assumed dependent on A only. The required separation of V and A into two terms is provided by this form.

The "second cross-section" assumed here could arise from the interaction of the particle with the floor of container, e.g. if the particle approached within a diameter of the floor, it would collide and be slowed. Alternately, the particle might be interacting with the three dimensional shape of the binding site, i.e.. as a peg in a hole, independent of the electric fields present. In this case, the cross-section could define a range of permissible particle orientations.

In the second experiment, it is the log transformation that removes the interaction, and we therefore expect A and V to occur as a product (see Tables 7 and 8 below). Note that the amplitude is decreased and electrode voltage increased as compared to the first factorial. Thus, the term in A above might become insignificant. In addition, $r=V/A$ could grow to exceed L, in which case σ behaves as

$$\sigma \propto r^2 = \left(\frac{V}{A}\right)^2$$

Then

$$R = nv\sigma \propto nA\frac{V^2}{A^2} \propto \frac{V^2}{A}$$

and

$$\ln(\tau) = \ln\left(\frac{1}{R}\right) = -2\ln(V) + \ln A + constant$$

which displays the experimentally determined A, V independence under log transformation.

A	29.1	1	29.1	27.79	1%	E_V	-3.87
V	93.4	1	93.4	89.32	1%	E_A	2.17
Interaction	0.1	1	0.1	0.10		E_{AV}	0.13
Error	100.4	96	1.0			eff error	0.97
Total	222.9	99		Fo 0.01,1,96 ~ 6.9			

TABLE 7. Parameter analysis for the second experiment, log-transformed data.

src of var	sum sqr	DOFs	Mean Sqr	Fo			
A	26183.9	1	26183.9	14.80	1%	E_V	-105.53
V	69601.6	1	69601.6	39.33	1%	E_A	64.72
Interaction	12672.7	1	12672.7	7.16	1%	E_{AV}	-45.03
Error	169868.9	96	1769.5			eff error	22.59
Total	278327.1	99		Fo 0.01,1,96 ~ 6.9			

TABLE 8. Parameter analysis for the second experiment, untransformed data.

4.0 Conclusion

In summary, the factorial technique yielded a performance improvement of some 540%, comparing $1/\tau$ at the initial centerpoint to the optimal result in the second trial. These types of experiments can also build intuition regarding system behavior. The "second cross-section" model, for example, does not occur in standard thermodynamics or physical chemistry; rather, the statistical analysis seems to motivate it directly.

Further experiments will search for an optimal operating point, as well as collect data to substantiate the models. One key point which must be addressed is stability of the apparatus against dielectric breakdown, which necessitates replacement of the electrode assembly and randomly changes the effective operating point. A stable electrode assembly could be modeled by finite element analysis to provide the field distribution; in conjunction with direct measurements of V, v, and particle mass, this would enable quantitative modeling.

5.0 References

[1] Cohn, M.B.; Kim, C.J.; Pisano, A.P. Self-assembling electrical networks: an application of micromachining technology. (TRANSDUCERS '91. 1991 International Conference on Solid-State Sensors and Actuators. Digest of Technical Papers (Cat. No.91CH2817-5), San Francisco, CA, USA, 24-27 June 1991). New York, NY, USA: IEEE, 1991. p. 490-3.

[2] Yeh, H.-J.J.; Smith, J.S. Integration of GaAs vertical-cavity surface emitting laser an Si by substrate removal.   Applied Physics Letters, 21 March 1994, vol.64, (no.12):1466-8.

[3] Introduction to statistical quality control / Douglas C. Montgomery. 2nd ed. New York: Wiley, c1991, p. 47.

[4] Statistics for experimenters : an introduction to design, data analysis, and model building / George E. P. Box, William G. Hunter, J. Stuart Hunter. New York : Wiley, c1978.

[5] Box, p. 232.

[6] Fearing, R.S. Tactile sensing mechanisms. International Journal of Robotics Research, June 1990, vol.9, (no.3):3-23.

[7] Thermal physics / Charles Kittel, Herbert Kroemer. 2d ed. San Francisco: W. H. Freeman, c1980.

[8] Physics of semiconductor devices / S.M. Sze. 2nd ed. New York : Wiley, c1981.

Forecasting and Control of Pulsed Laser Energy using an Integrated Moving Average Model

Richard Schenker

A 213nm pulsed Nd-YAG laser is currently being used for the characterization of Deep-UV damage of fused silica. The need for nearly constant pulse energies is of extreme importance in order to determine damage rates as a function of pulse energy density. Analysis of past pulse energy measurements showed a non- stationary series with both long term drift and short term oscillations of average pulse energy. Differencing of the measured pulse energies was successful in producing a stationary data series where Box-Jenkins modeling could be applied. An ARIMA(0,1,1) model proved robust in tracking the average pulse energy as it oscillated in time. The model was implemented in computer code to forecast future pulse energies so as to block future pulses outside a desired energy range. The ARIMA model blocked about 43% of pulses outside the set range in one test, while only blocking 8% of the pulses inside the range. An EWMA control chart was also configured to detect long term shifts in the pulse energy mean. It proved to be an efficient method of monitoring drifts in the average energy without excessive sensitivity to short term energy fluctuations.

1.0 Motivation

Lithography has been consistently moving towards using shorter and shorter wavelength light sources. Deep-UV sources are already being used in industrial IC fabrication processes and will undoubtedly be the main line lithography tool of the future because of their ability to print smaller features. One obstacle to wide spread application of UV lithography is the damage which UV radiation causes to fused silica, the primary material used in UV optics. A pulsed quintupled Nd-YAG laser of wavelength 213nm is being used in the Berkeley microlab to irradiate fused silica with relatively energetic pulses in order to better understand the mechanisms behind various damage processes [1].

It is of high importance to find the damage rates of fused silica as a function of laser pulse energy density. Since a two photon process is believed to contribute to the damage mechanisms, the damage rates as a function of total energy delivered increase for higher pulse energy densities. Unfortunately, the pulse energy delivered from the Nd-YAG laser is difficult to control. Heating effects in the YAG rod and in the non-linear optics used to quintuple the laser frequency cause large variations in pulse energy. These variations take the form of both an apparent random distribution around a temporary average energy and often a drifting of average power over the span of several hundred laser pulses. In order to improve the accuracy of the determination of damage rates as a function of pulse energy, a system needed to be designed to limit the number of laser pulses delivered to the sample which were significantly different from the desired pulse energy.

The nature of the variations in laser pulse energy is such that each reading statistically depends on previous readings. In other words, the data is autocorrelated and not independently and identically distributed according to a simple normal distribution. Autoregressive Integrated Moving Average (ARIMA) models are often quite robust in predicting the future value of autocorrelated data. Applying an ARIMA model to predict future energy values provides the ability to either block subsequent pulses which are predicted to be outside a desired energy range or to adjust in real time the inputs which effect pulse energy to make subsequent energies closer to the desired value. Unfortunately, real time adjustment of pulse energy is not a viable solution. The only method to change the laser power without altering the beam direction or shape is achieved by rotating a waveplate which is placed before the final non-linear crystal. This crystal mixes the second and third harmonics of the original 1064nm radiation to produce 213nm radiation. The efficiency of the mixing is a strong function of the relative phases between the two harmonics so a slight rotation of the waveplate dramatically alters the beam energy. Consequently, this waveplate can only adjust the average beam energy by coarse amounts. Furthermore, the sensitivity of pulse energy to changes in waveplate orientation varies with thermal conditions and is difficult to predict due to the complexity of the physical situation.

The ability to filter or block future pulses is available in that the computer which monitors the pulse energy can also be programmed to close a shutter which then blocks the beam from irradiating the sample. The pulse energy could still be measured when the shutter is closed because the shutter would be positioned after the initial sampling of the beam power as can be seen in Figure 1. The computer will use an ARIMA model to predict the subsequent pulse energy and control the blocking shutter based on that prediction. Coarse adjustment of the beam power using the rotating waveplate will only be used when an Exponentially Weighted Moving Average control chart, also programmed into the computer, indicates that the average pulse energy has drifted outside of specified limits.

2.0 Analysis of Past Measurements

2.1 Basic Trends in Laser Pulse Energy

Figures 2, 3 and 4 show the typical variation of laser pulse energy with time. The variation in pulse energy can be said to consist of three components:

- Random Variation about Current Average

- Short term oscillation of Average Energy

- Long term drift of Average Energy

Figure 2 contains a linear regression fit of all 2800 data points. The positive slope of this fit corresponds to a long term drift of the average laser energy independent of the short term oscillations. Figure 3 is a control chart of the residues from the linear fit subtracted from the measured energy. Only one data point is outside the three sigma limits (sigma calculated from data), which is fewer than would be expected from a normal distribution of the data. One would expect on average seven points to fall outside the control limits if the data was normally distributed about the linear regressive fit. It appears that the calculated sigma of the pulse energy (0.063) is inflated in terms of its applicability for the control chart due to the short term oscillations of the pulse energy. In order to more precisely estimate the sigma for the control chart, a fit of the short term oscillations of the average energy is required in order to track only the random fluctuations of pulse energy.

Figure 4 which shows a shorter time interval and best illustrates the oscillations in pulse energy. These oscillations, which are not strictly periodic, seem to result in either a maximum or minimum of laser energy roughly every 300 pulses. The heating and cooling cycles of the non-linear crystals used in the system are a possible assignable cause for these oscillations. ARIMA models, in general, are only applicable when there is definite assignable cause for the autocorrelation of the data, such as sampling at a high frequency. But, since the basic characteristics of the pulse energies has consistently been observed to be similar to that in Figure 2, 3, and 4; it is logical to assume that some physical process, albeit not precisely known, is behind the non-normal behavior and hence can be predicted with an ARIMA model.

2.2 EWMA Control Chart

Because of the difficulty in adjusting the output pulse energy of the laser, it is only desired to attempt to adjust the energy for long term drifts in the laser power. A method of monitoring the average energy which is not overly sensitive to the short term oscillations in energy is needed. A Moving Average chart computes the average of the last n data points in order to detect small shifts in process mean. A Moving Average 'window' of at least 300, the approximate period of the short term oscillations, would be required to sufficiently reduce the sensitivity to the short term oscillations. The real time storage of the 'window' of data points is slightly time consuming for the PC controller used in this experiment, which must perform all its operations during the 0.1 second period of the pulsed laser. Fortunately, an Exponentially Weighted Moving Average (EWMA) chart does not need to store many past measurements and has essentially the same performance in detecting small shifts in mean as the generic Moving Average chart.

The EWMA chart monitors the entire process history but places more weight on the more recent measurements. The weighting decreases geometrically as the data points move farther back into time. The estimated current mean, wt, is given by the equation:

$$w_t = \lambda z_t + (1 - \lambda) w_{t-1} \tag{1}$$

where z_t is the last data measurement and λ is the geometric weighting factor. A small geometric weighting factor of 0.01 was chosen so that a large degree of averaging would take place in order to reduce mean fluctuation due to solely short term oscillations. The control limits for an EWMA chart generally are equal to plus and minus $3\sigma * (\lambda/(2-\lambda))^{1/2}$ around the desired mean. Since it is already conceded that the pulse energy average oscillates with time, the control limits will be expanded by a full sigma in each direction. This gives the control limits as:

$$UCL = Center + \sigma + 3\sigma * (\lambda/(2-\lambda))^{1/2} \tag{2}$$

$$LCL = Center - \sigma - 3\sigma * (\lambda/(2-\lambda))^{1/2} \tag{3}$$

These limits hence are set up to test for situations where the current average is outside plus or minus one sigma from the desired center energy. If the actual mean was just below one sigma above the center, a false alarm would occur with a probability of 0.00135.

2.3 Attempt to Fit ARMA model to Data

The fitting of a time series model is usually performed with the aid of the Autocorrelation Function (acf) and the Partial Autocorrelation Function (pacf) of past measurements. Figures 5 and 6 show the acf and pacf respectively for the previously presented data calculated using BLSS (The Berkeley Interactive Statistical System). The standard method used in forecasting, Univariate

Box-Jenkins analysis or ARIMA analysis, requires that the data series be 'stationary' for the analysis to be valid [2]. According to Pankratz, a stationary series has a mean, variance, and acf that are essentially constant through time. Simple inspection of the data shows that the mean is not constant with time indicating that the data series is non-stationary. A more formal check of stationarity uses the values of the coefficients in the acf and pacf functions. For an ARMA(2,q) model, if any of the following three conditions is not met the data is not stationary.

$$|\phi| < 1 \tag{4}$$

$$\phi_2 + \phi_1 < 1 \tag{5}$$

$$\phi_2 - \phi_1 < 1 \tag{6}$$

ϕ_i are the pacf coefficients An ARMA(1,q) has only the condition, $|\phi| < 1$, for stationarity. But, the above rules are mute due to the form of the estimated acf function which does not fall to zero for several hundred lag spacing. Virtually all acf's which correspond to physical situations with a stationary mean and variance fall to zero after only lag 5 or 6 [2]. It is clear that the data series is non- stationary so a transformation is needed before a time series model can be implemented.

2.4 Differencing of Data Series

Differencing is one method to make a time series stationary. This is achieved by simply calculating the difference between adjacent data points in the data series, essentially differentiating the series. Figures 7 and 8 show the first differences of the previously plotted data. Note that both the long term drift and short term oscillations of values are no longer observable. Figures 9 and 10 are the calculated acf and pacf of the first difference data. The acf function falls to zero effectively after the first autocorrelation coefficient, which is consistent with a stationary series. The pacf coefficients also conform to the three conditions for stationarity listed above for a ARMA(2,q) model. Since, the acf and pacf functions are consistent with stationary data series, and because the plots of the first difference portray no indications that either the variance or mean is time varying, one can conclude that the first difference is a stationary series. Therefore, No further differencing is needed.

2.5 Choice of ARIMA Model

The general form of an ARIMA model is the following.

$$y_t = \mu + \phi_1 z_{t-1} + \phi_2 z_{t-2} + ... + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - .... \tag{7}$$

where y_t is the next value, μ is the mean, the z are previous values in the series, ε_t is the normally distributed error from the predicted model, ε_{t-x} are the past errors in the model prediction, and θ and ϕ are coefficients to be fitted for the given data series. An ARIMA(p,d,q) model has p autocorrelation terms ($\phi_x z_{t-x}$), q moving average terms ($\theta_y \varepsilon_{t-y}$), and has had the data differenced d times. It has already been determined that our data series had to be differenced once to make it stationary. The choice of p and q was made by using the Box-Jenkins modeling procedure [2]. The acf and pacf of the first differenced data was compared to theoretical acf's and pacf's of stationary processes. In general, ARIMA(1,d,0) processes, autoregressive processes, have a pacf which contains only one significant term and an acf with terms that slowly decay down to zero. ARIMA(2,d,0) processes have two significant pacf terms and a similar exponentially decaying acf. ARIMA(0,d,q) (q=1 or 2) processes, or moving average processes, on the other hand, have only q significant acf terms and a pacf which slowly decays exponentially to zero. Processes with

both a non-zero q and p, also called mixed processes, have theoretical acf's and pacf's, both of which, have several significant terms.

The acf and pacf shown in Figures 9 and 10 strongly follow that of an ARIMA(0,d,1) model because of the single significant acf term and the exponentially decaying pacf. This would result in the following model for the first differenced data: $y_t = \mu + \varepsilon_t - \theta_1\varepsilon_{t-1}$ The mean of first differenced data was calculated to be 4.35 x 10^{-5}, which will be considered small enough to be set to zero for convenience. One can calculate the theoretical acf of the $y_t = z_t = \varepsilon_t - \theta_1\varepsilon_{t-1}$ model. The acf coefficients are just the correlation coefficients for the data series ρ_k, where $\rho_k = 1/\sigma_z^2 * E(z_t, z_{t-k})$. Expanding this with the model $y_t = z_t = e_t - \theta_1\varepsilon_{t-1}$ gives:

$$\rho_k = 1/\sigma_z^2 \, E[(\varepsilon_t - \theta_1\varepsilon_{t-1})(\varepsilon_{t-k} - \theta_1\varepsilon_{t-1-k})] \tag{8}$$

$$\rho_k = 1/\sigma_z^2 \, [E(\varepsilon_t \varepsilon_{t-k} - \theta_1\varepsilon_{t-1}\varepsilon_{t-k} - \theta_1\varepsilon_t \varepsilon_{t-1-k} + \theta_1^2\varepsilon_{t-1}\varepsilon_{t-1-k})] \tag{9}$$

$$\rho_k = 1/\sigma_z^2[E(\varepsilon_t \varepsilon_{t-k}) - \theta_1 E(\varepsilon_{t-1}\varepsilon_{t-k}) - \theta_1 E(\varepsilon_t \varepsilon_{t-1-k}) + \theta_1^2 E(\varepsilon_{t-1}\varepsilon_{t-1-k})] \tag{10}$$

The estimated value of $\varepsilon_x\varepsilon_y$ for x not equal to y is zero because the estimated value of the product of two random numbers is zero. So, the above equation is only non-zero when k=1 which gives:

$$\rho_k = -1/\sigma_z^2 \, \theta_1 E(\varepsilon_{t-1}\varepsilon_{t-1}) = -\theta_1 * \sigma_\varepsilon^2/\sigma_z^2 \tag{11}$$

for k=1 and $\rho_k = 0$ for k>1. Notice that the variance of z is not equal to the variance of the error of the model prediction. One can find σ_z^2 in terms of σ_ε^2 using $z_t = \varepsilon_t - \theta_1\varepsilon_{t-1}$ to find that;

$$\sigma_z^2 = (1 + \theta_1^2) \, \sigma_\varepsilon^2 \tag{12}$$

This finally gives

$$\rho_1 = -\theta_1/(1 + \theta_1^2). \tag{13}$$

The pacf coefficients can be found by applying the algorithms used to calculate the pacf for measured data to simulated data generated using the model. The first coefficients in the acf and pacf are always equal.

The choice of θ_1 for the model was done using the formula $\rho_1 = -\theta_1/(1 + \theta_1^2)$ and setting it equal to -0.49, the calculated ρ_1 for the first difference data. This gave a θ_1 of 0.8 and a final ARIMA(0,1,1) model of:

$$y_t = z_t = \varepsilon_t - 0.8\varepsilon_{t-1}. \tag{14}$$

2.6 Applying ARIMA Model to Past Measurements

Figures 11, 12 and 13 show the results of applying the above model to the first difference of the previously described data and then integrating back to get a predicted pulse energy. As can be seen, the model nicely follows the general trend of the pulse energy, including the short term oscillations. Only random deviations about the model are observed; this is best shown in Figure 12. Figure 14 shows the residual between the model and measured data. The residues appear to be

identically and independently distributed about zero with a normal distribution. Five points fall outside the three sigma control limits, which is close to the seven which would be predicted from an arbitrary normal distribution. The calculated standard deviation of 0.0445 does not appear to over estimate the true sigma of the process. The model plots and the residual plot confirm that the choice of the ARIMA model is a valid one, eliminating all but random deviations from the data series.

3.0 Implementation of Time Series Filter and EWMA Control

After the basic model of laser pulse energy was established, the model was implemented in the actual damage exposure system. Figure 1 shows the setup which is entirely controlled by a personnel computer. The shutter has not yet been installed. The software package Windows for Data provided the framework for the code. Some of the commands in Windows for Data are different from standard C language commands.

The different operations of control of a damage experiment are illustrated in Figure 15. In Manual Mode, data points are stored only when the operator instructs the computer to do so by means of a keystroke. Stat Mode stores the last 2500 points in an array energys_pd continuously. When Stat Mode is exited a new coefficient for the ARIMA(0,1,1) model, macoeff, is calculated using the stored pulse energies. An iterative procedure finds θ_1 which best satisfies the equation $\rho_1 = -\theta_1/(1 + \theta_1^2)$. If Stat Mode is not entered the coefficient value remains at its default of 0.8, the value calculated from the past data analyzed above. Automatic Mode is the mode used during the majority of damage characterization experiments. It stores averaged measurements periodically without operator assistance. It also continuously implements the ARIMA(0,1,1) model by predicting the next pulse energy. The code for predicting the next pulse energy is run after every laser pulse is fired and measured:

```
diff=detectors_pd[2]-hold1;

err=diff-pred;

pred=macoeff*err*(-1);

predb=detectors_pd[2]+pred;

hold1=detectors_pd[2];
```

`detectors_pd[2]' is the measured pulse energy, `hold1' is the previous pulse energy, `diff' is the difference between the current and previous pulse energies, `err' is the error in prediction of the current difference, `pred' is the predicted next difference in pulse energies, and `predb' is the next prediction pulse energy.

If the predicted next pulse energy is outside of the range specified at the beginning of running the code, an output bit on a PC board is toggled to produce a positive five volt value. This voltage can be then used to close a shutter, which blocks that pulse from hitting the sample. Automatic Mode also calculates the EWMA for the pulse energy. In calculating the control limits, the sigma of the process was assumed to scale with the magnitude of the average pulse energy. If the EWMA falls outside of the previously calculated control limits an alarm is generated. A negative energy reading on the PC screen indicates to the operator that an alarm is present. The operator should adjust the rotating waveplate so that it returns the pulse energy to the desired value when an alarm is generated.

4.0 Evaluation of Operation of Time Series Filter and EWMA control

After successfully compiling the modified damage experiment control code, the code was run to test its performance. A desired pulse energy of 0.75 was set as the goal, with the allowed prediction interval set to 0.67 to 0.83. An acceptance chart could have also been used to setup control limits. Stat Mode operation was entered and a moving average coefficient of 0.81 was output. The code was modified to save the predicted energy values and every laser pulse energy. Figures 16 and 17 show both the measured energy values and the values predicted by the real time computer control. As with the reference data, the model was successful in tracking the average mean of the pulse energy. The program successfully toggled the output bit when the energy was predicted to be outside the set range. Figure 18 shows a histogram of the measured energies and the pulse energies which would not have been blocked by the shutter. As can be seen, the program was successful in filtering out many of the shots far from the center value, while allowing almost all shots near the center value to pass through. 115 of the 260 pulses with energy measured above 0.83 were blocked while 110 of the 262 pulses below were successfully predicted. Only 117 of the remaining 1478 shots within the 0.67 to 0.83 range would have been incorrectly blocked and most of those were near the edge of the range.

No EWMA control alarms were generated during the test. Subsequent tests did later show that an EWMA alarm was generated when appropriate. Figure 19 plots the EWMA for the above mentioned test. As can be seen, the EWMA never falls outside the control limits, which is consistent with the plot of pulse energy which does not appear to show significant long term drift of laser energy.

5.0 Conclusions

Differencing of the measured pulse energies was successful in producing a stationary data series where Box-Jenkins modeling could be applied. An ARIMA(0,1,1) model proved robust in following the average pulse energy as it oscillated in time. The model was implemented in computer code to forecast future pulse energies so as to block future pulses outside a desired energy range. The ARIMA model blocked about 43% of pulses outside the set range in one test, while only blocking 8% of the pulses inside the range. A EWMA control chart was also configured to detect long term shifts in the pulse energy mean. It proved to be an efficient method of monitoring drifts in the average energy without excessive sensitivity to short term energy fluctuations. Future work should involve the determination of an assignable cause to the short term pulse energy fluctuations. More extensive modeling of the physical processes in the non-linear crystals should also be performed to gain a better understanding behind the causes of the erratic energy output.

6.0 References

[1] Schenker, Richard, "Issues in the Industrial Application of Deep-UV Photolithography Sources," masters report-UC Berkeley, Dept. of Elect. Eng., May 1994.

[2] Pankratz, Alan, Forecasting with Univariate Box-Jenkins Models: Concepts and Cases, 1983.

Damage Exposure System with Computer Controlled Shutter

Pre-Sample
Pulse Energy
Photodiode

Computer Controlled
Shutter

Attenuation

Adjustable
Sample
Stage

To Transmission
and Birefringence
Instrument

5th Harmonic
Nd-YAG
(213nm)
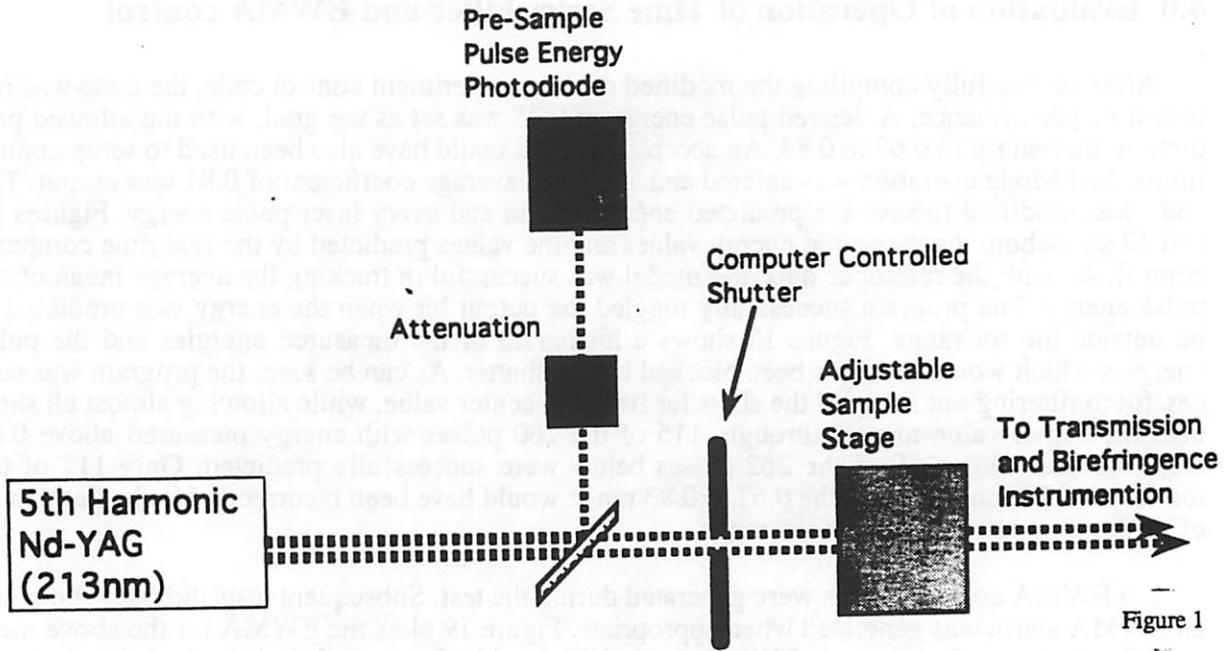
Figure 1

Laser Pulse Energy vs. Time

linear regression (0.62049 + 3.01*10^-5 * Pulse Number)

4/94

Laser Pulse Number
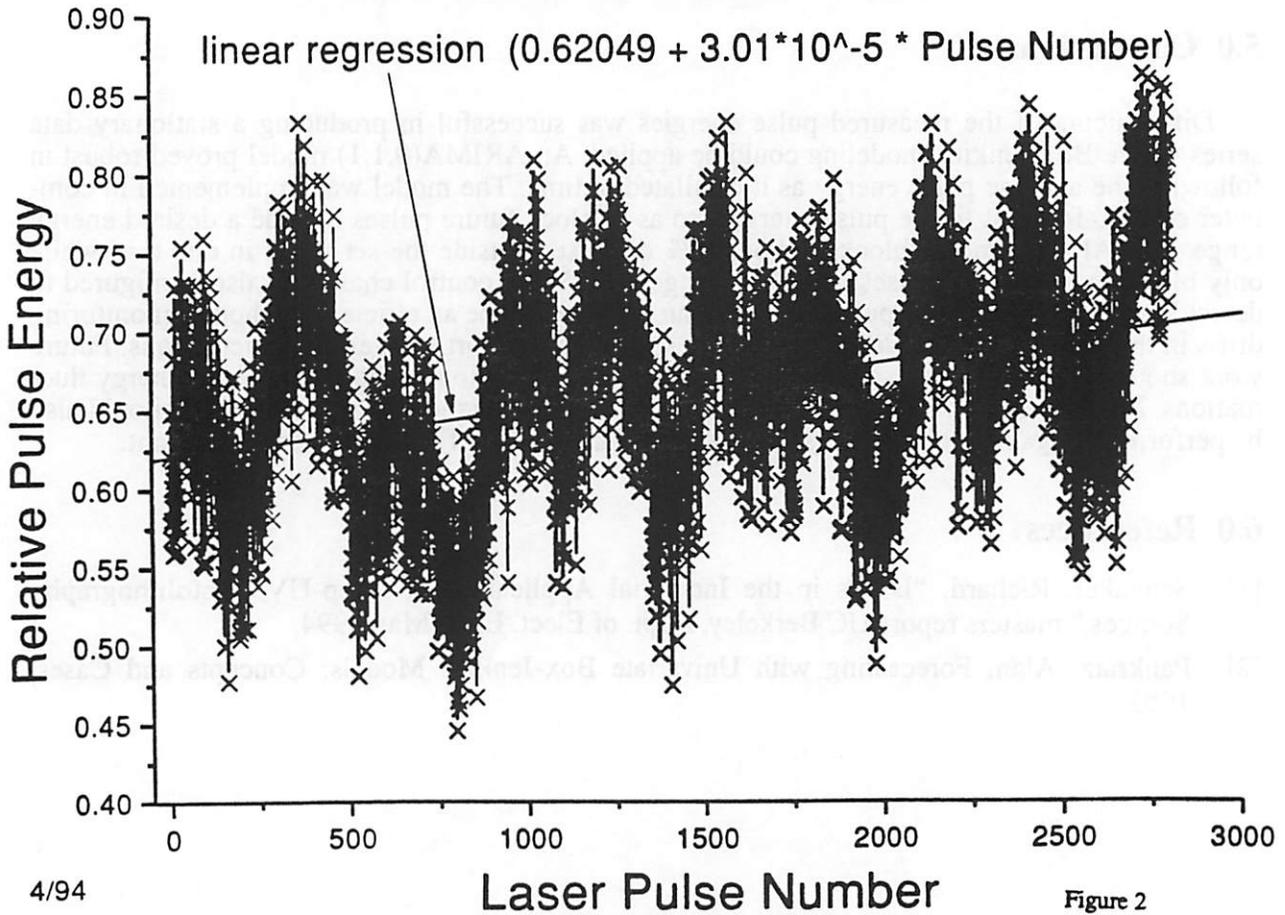
Figure 2

Residues for Linear Regression Fit of Laser Pulse Energies
Control Chart (3 Sigma Limits)

Figure 3

Laser Pulse Energy vs. Time

Figure 4

4/94
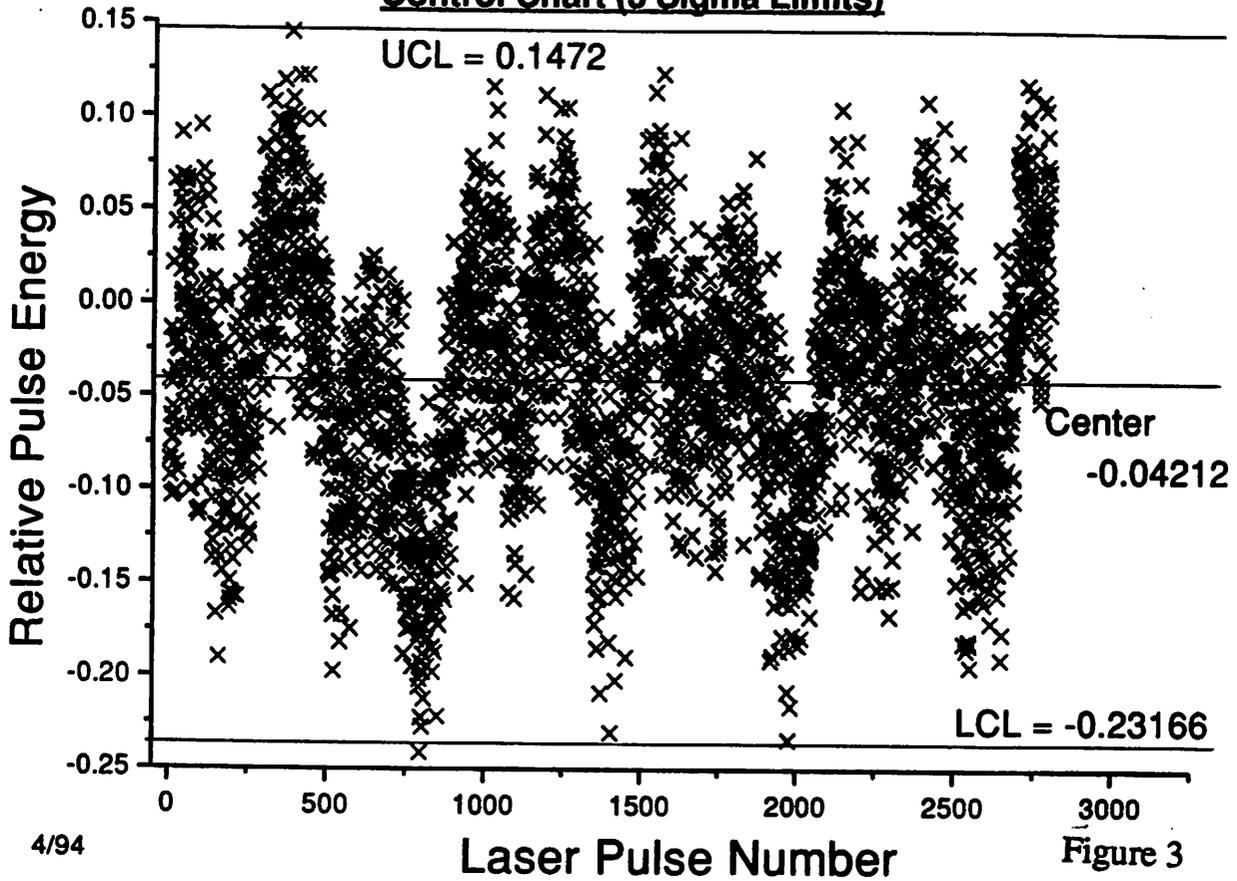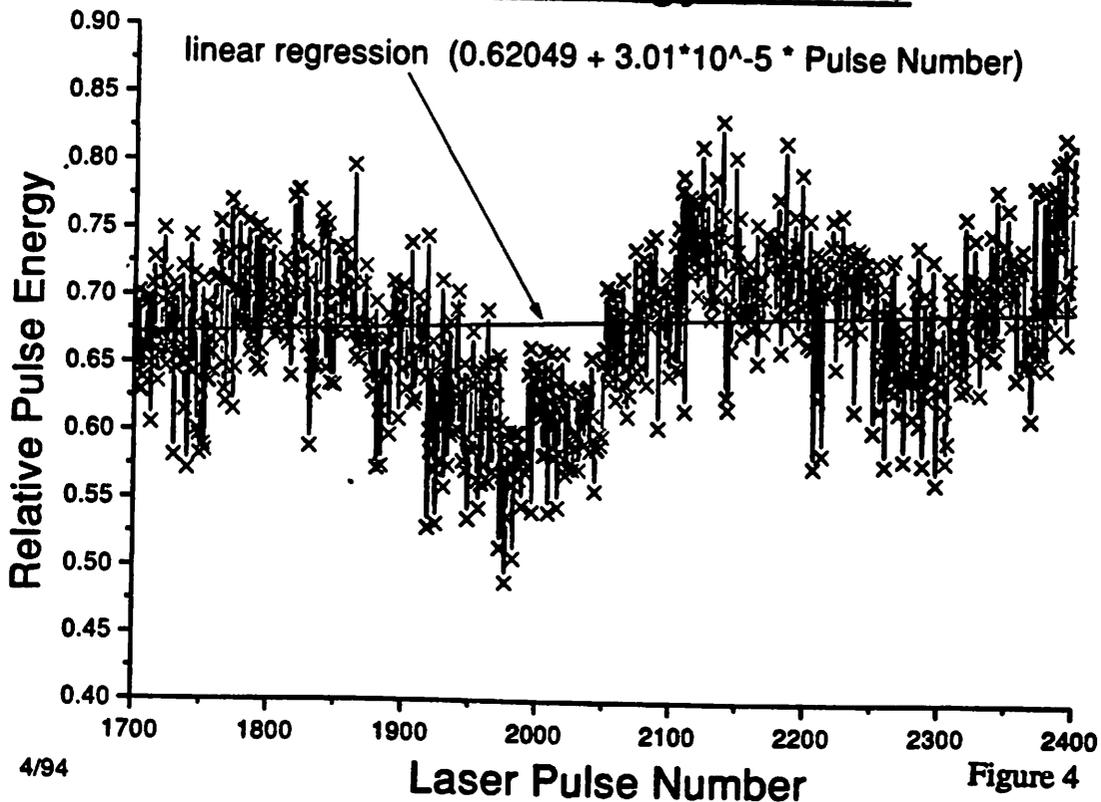
Autocorrelation Function for Laser Pulse Energy
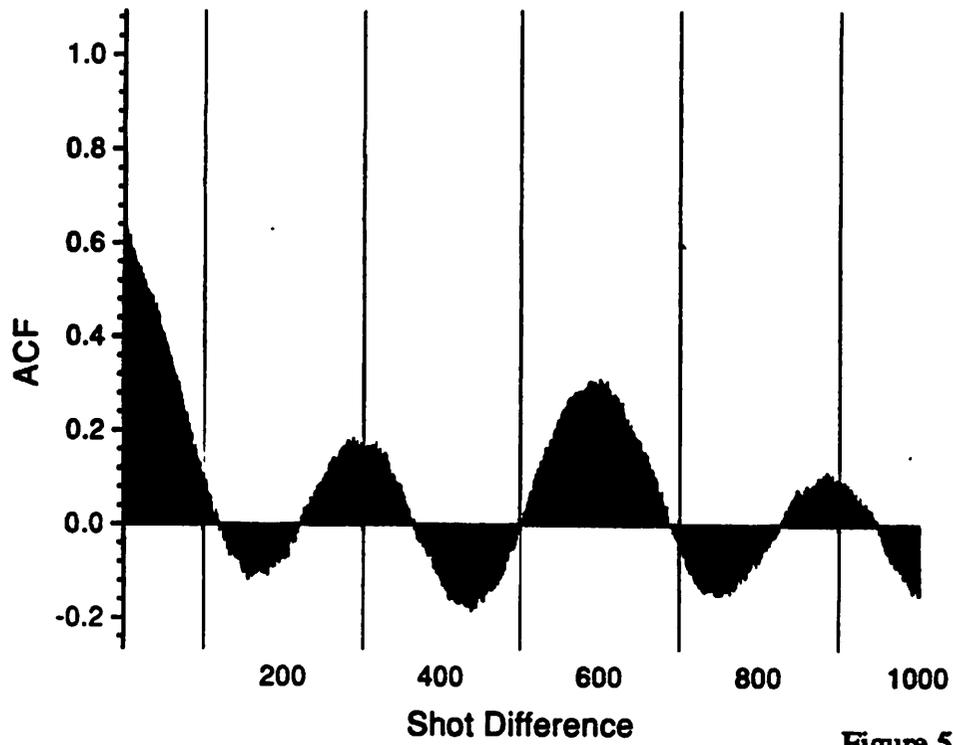
Figure 5

Partial Autocorrelation Function for Laser Pulse Energy
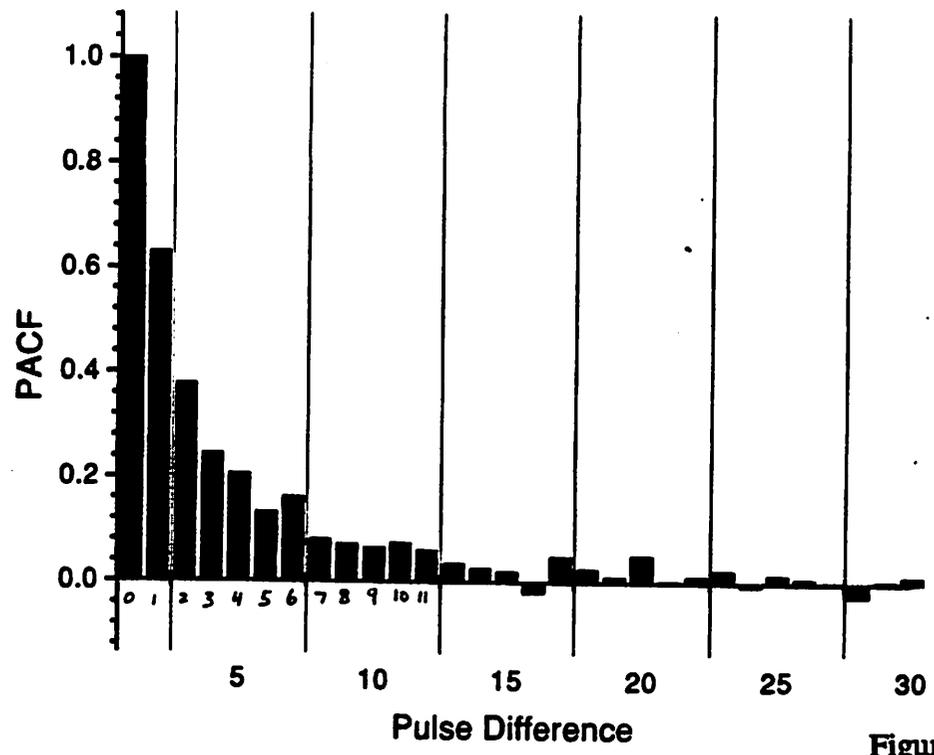
Figure 6

First Difference for Laser Pulse Energy vs. Time
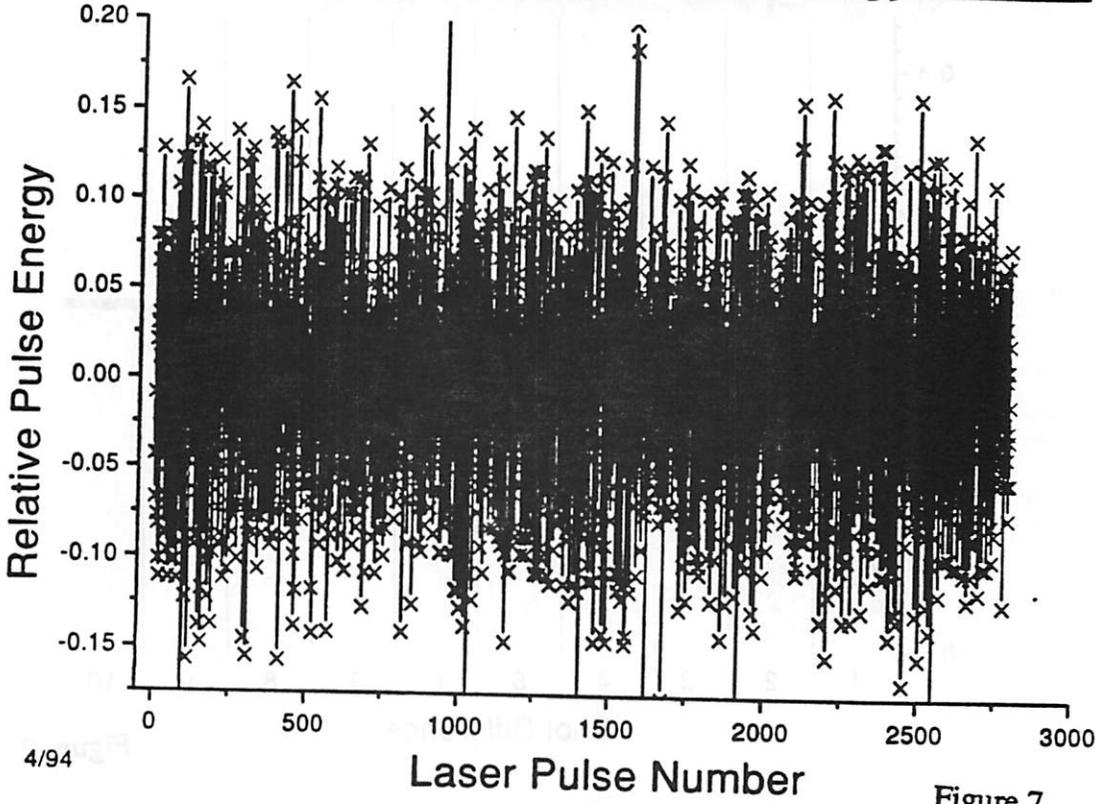
Figure 7

First Difference for Laser Pulse Energy vs. Time
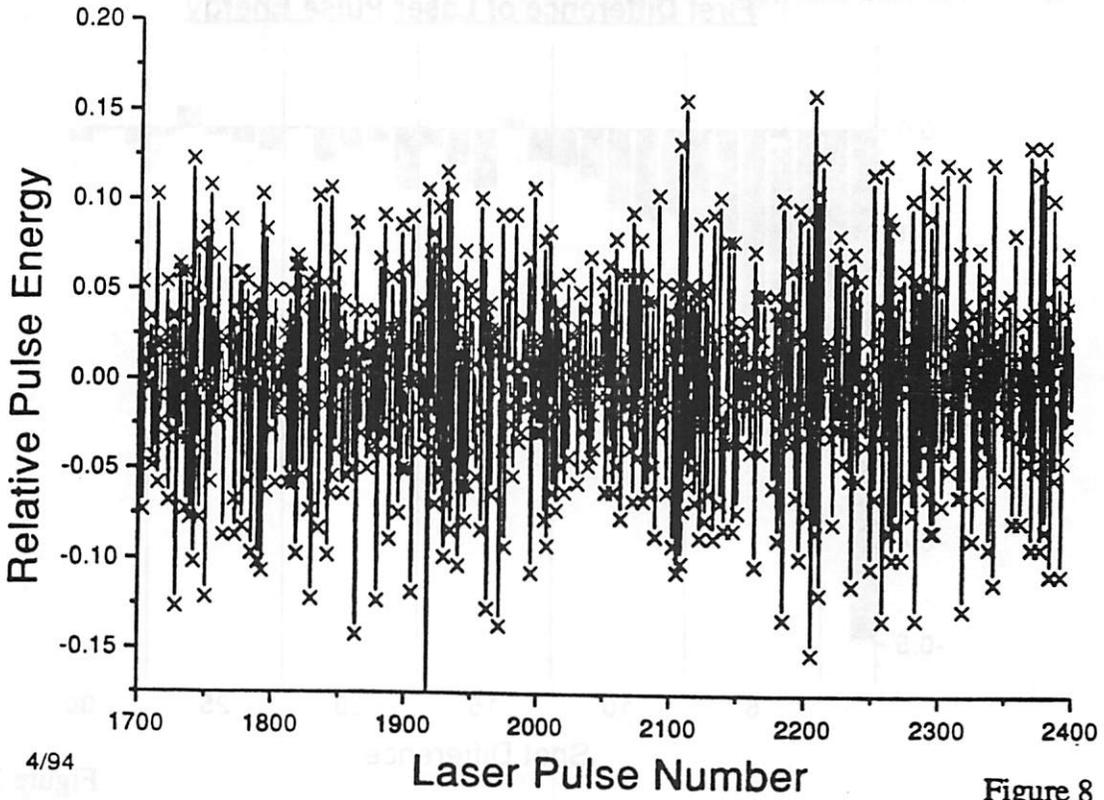
Figure 8

Autocorrelation Function for First
Difference of Laser Pulse Energy
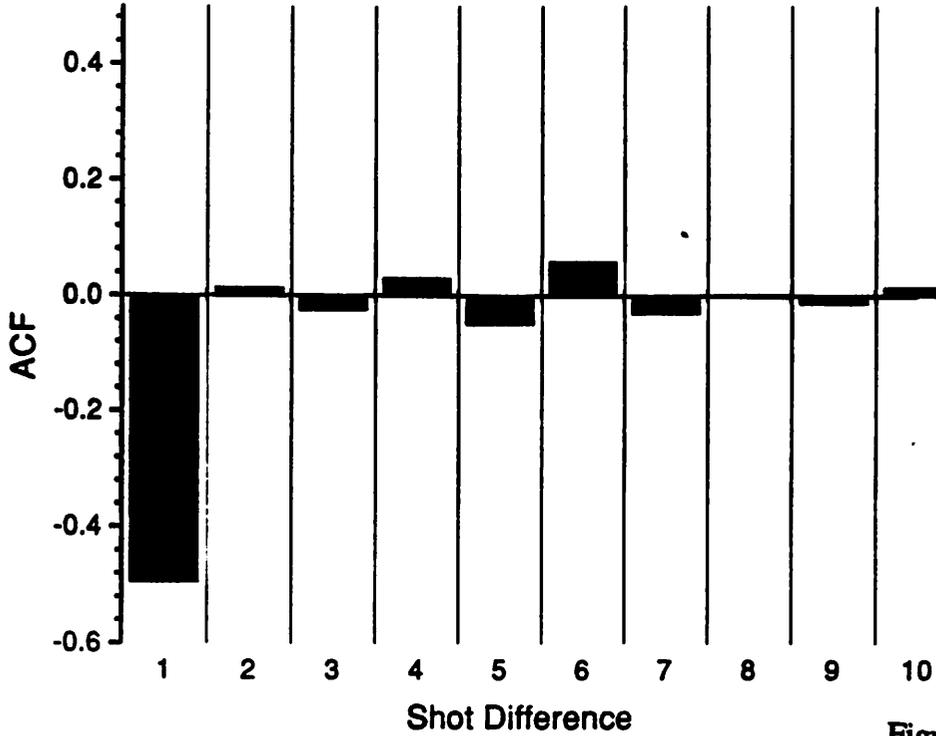
Figure 9

Partial Autocorrelation Function for
First Difference of Laser Pulse Energy
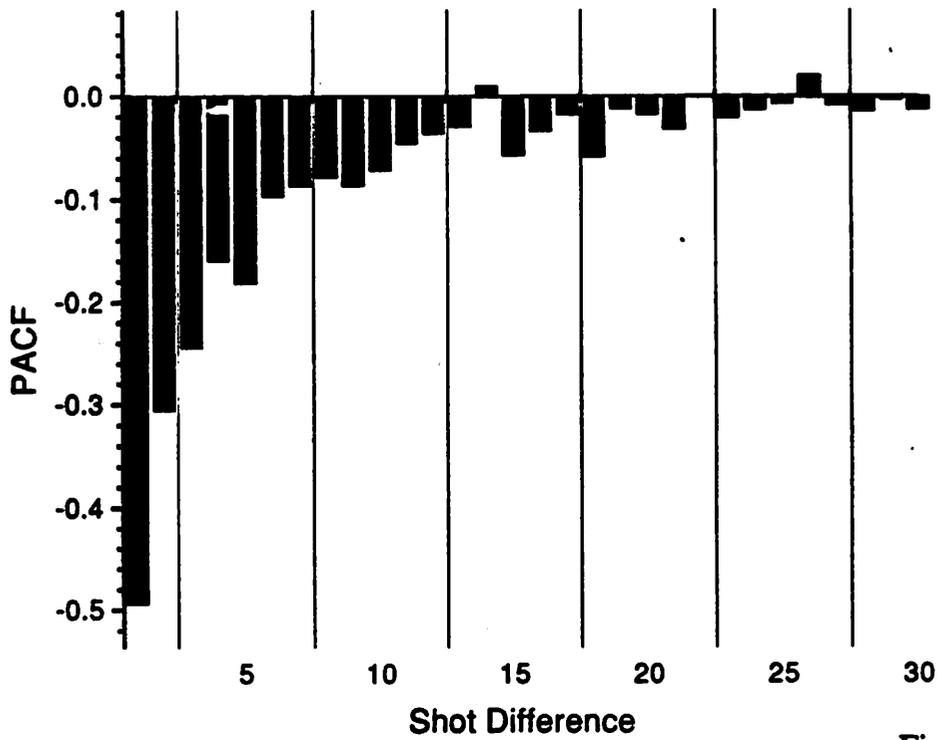
Figure 10

Laser Pulse Energy and Predicted Energy from
Moving Average Model Applied to First Difference
of Laser Pulse Energy vs. Time

4/94

Figure 11

Laser Pulse Energy and Predicted Energy from
Moving Average Model Applied to First Difference
of Laser Pulse Energy vs. Time

4/94

Figure 12

First Difference of Laser Pulse Energy
and Moving Average Model Prediction vs. Time

MA(1) Model Prediction

$$w(t) = -0.8 * a(t-1)$$

Laser Pulse Number

Figure 13

4/94

Residues between First Difference for Laser Energy
and Moving Average Model Prediction vs. Time

UCL = 0.1339

Center = 0.0003

LCL = -0.1333

Standard Deviation = 0.04453

Laser Pulse Number

Figure 14

4/94
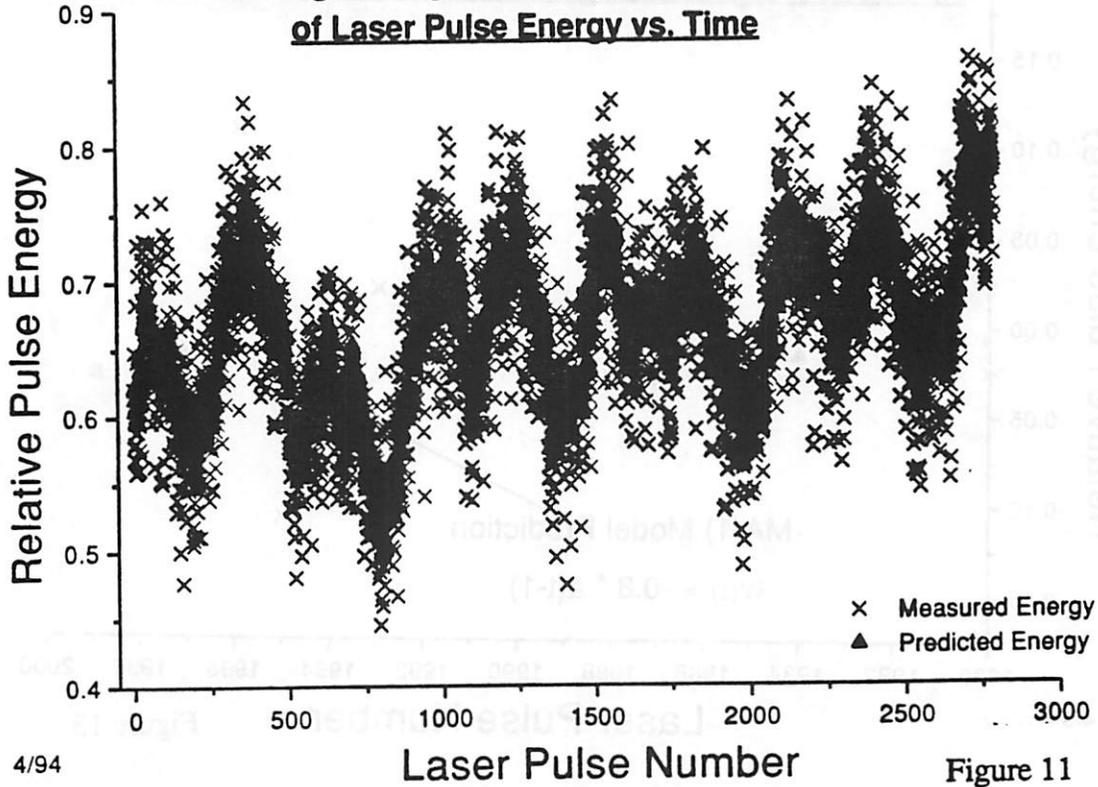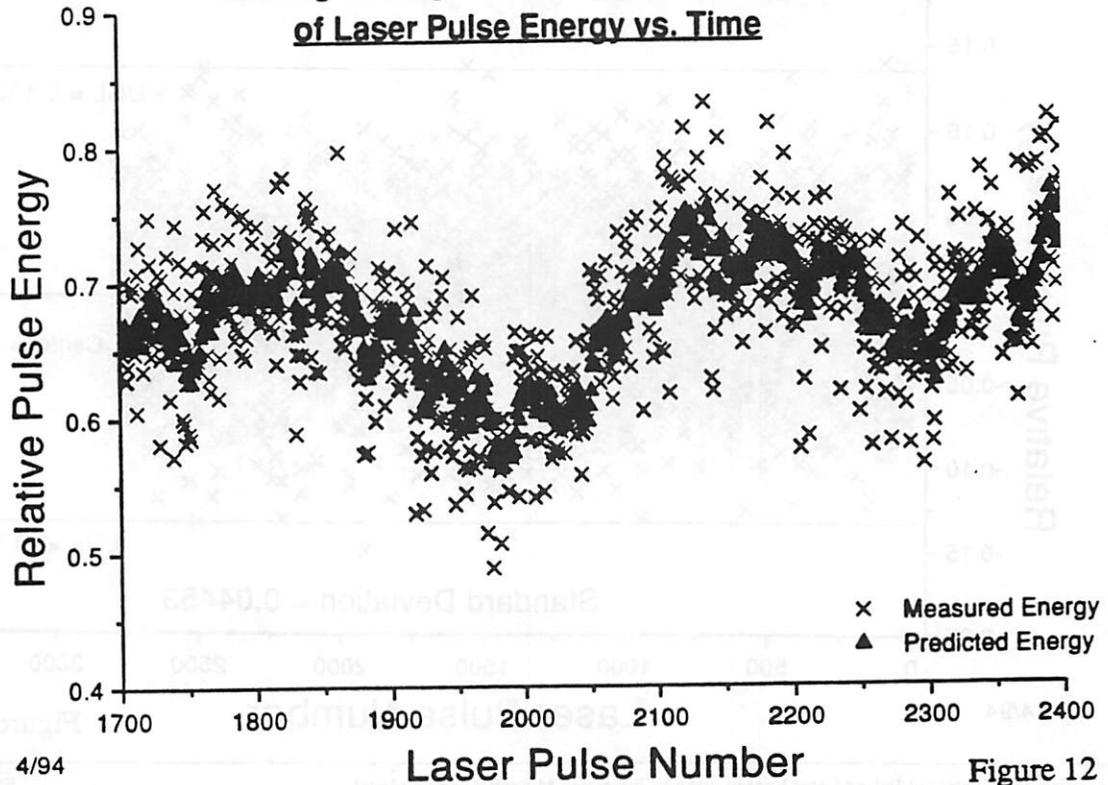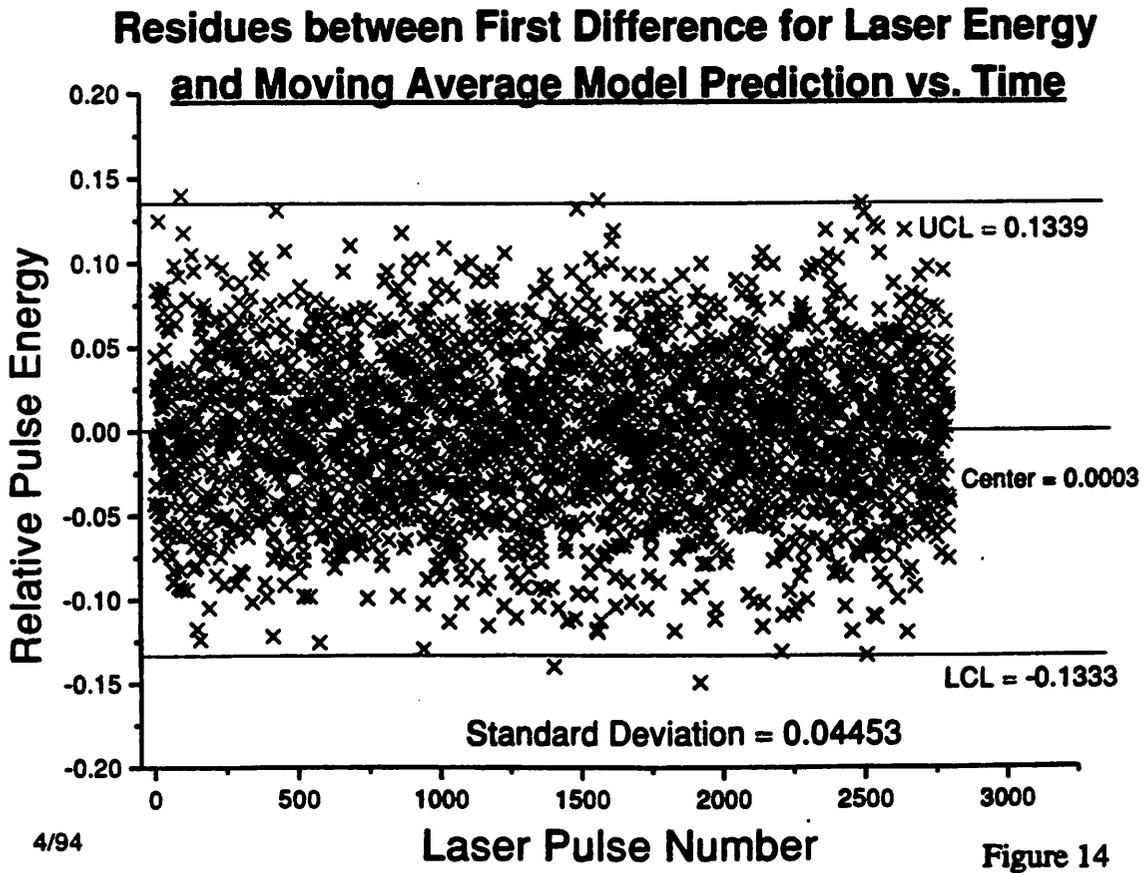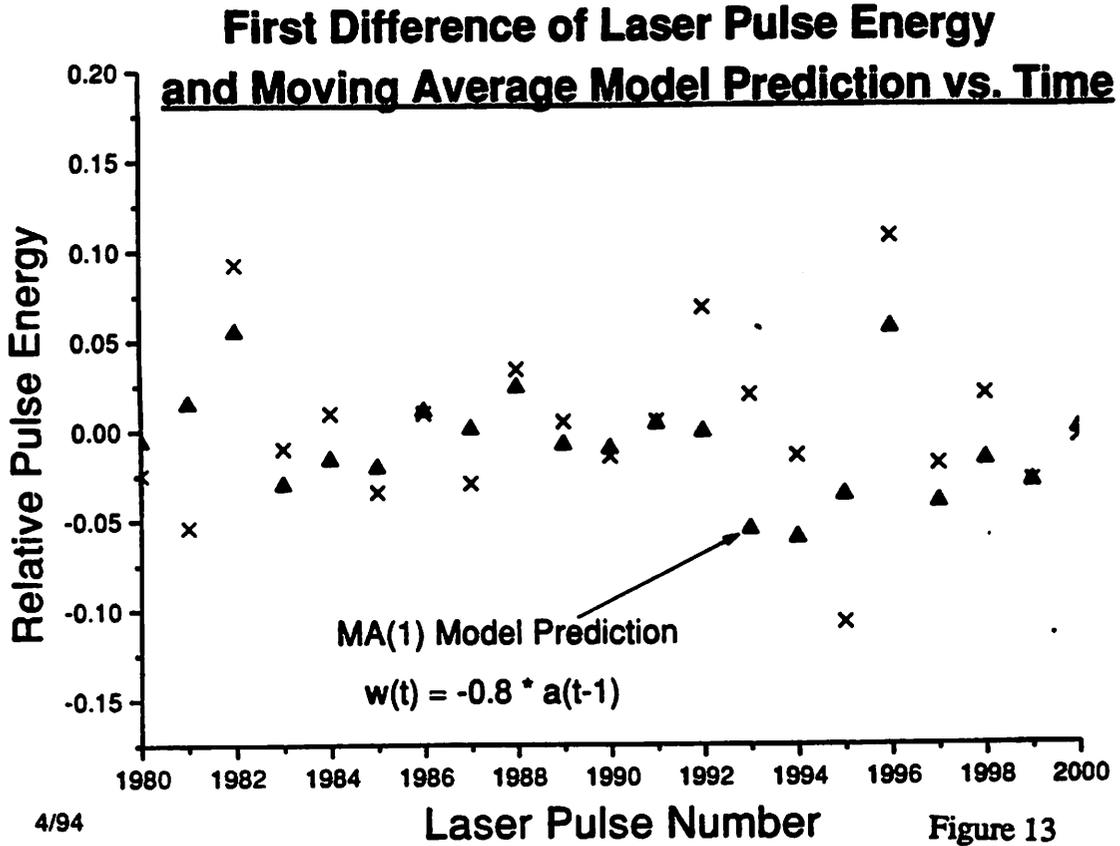
Basic Computer Control of 213nm Radiation Damage Experiments

Manual Mode
Firelaser and Reads Data
Prints values to Screen
Typing 's' toggles between
  Bir. and Tran. Measurements
Typing 'o' toggles between
  Counting and Not Counting
  Pulses

Typing 'f'

Stat Mode
Firelaser and Reads Data
Prints values to Screen
Typing 's' toggles between
  Bir. and Tran. Measurements
Saves last 2500 Data Points

Typing anything
but 'w'

Typing anything but 's','m','f','o'

Stalled Mode
Firelaser
Shows last Data Set
  Saved on Screen
Typing 'w' Ends
  Experiment

Typing anything but 's'
During Transition the
ARIMA(0,1,1) Coefficient
is Calculated from past
Data

Typing 'm'

Typing 'm'

Automatic Mode
Fires, Reads and Displays Data
Saves Data Set Every 30 Seconds
Typing 'o' toggles Counting of Pulses
Closes Shutter if Predicted Value
  from ARIMA(0,1,1) model outside
  preset Range
Gives Alarm if EWMA outside of
  Control Limits

Figure 15

Results of Application of Time Series Filter
Measured versus Real Time Predicted Values
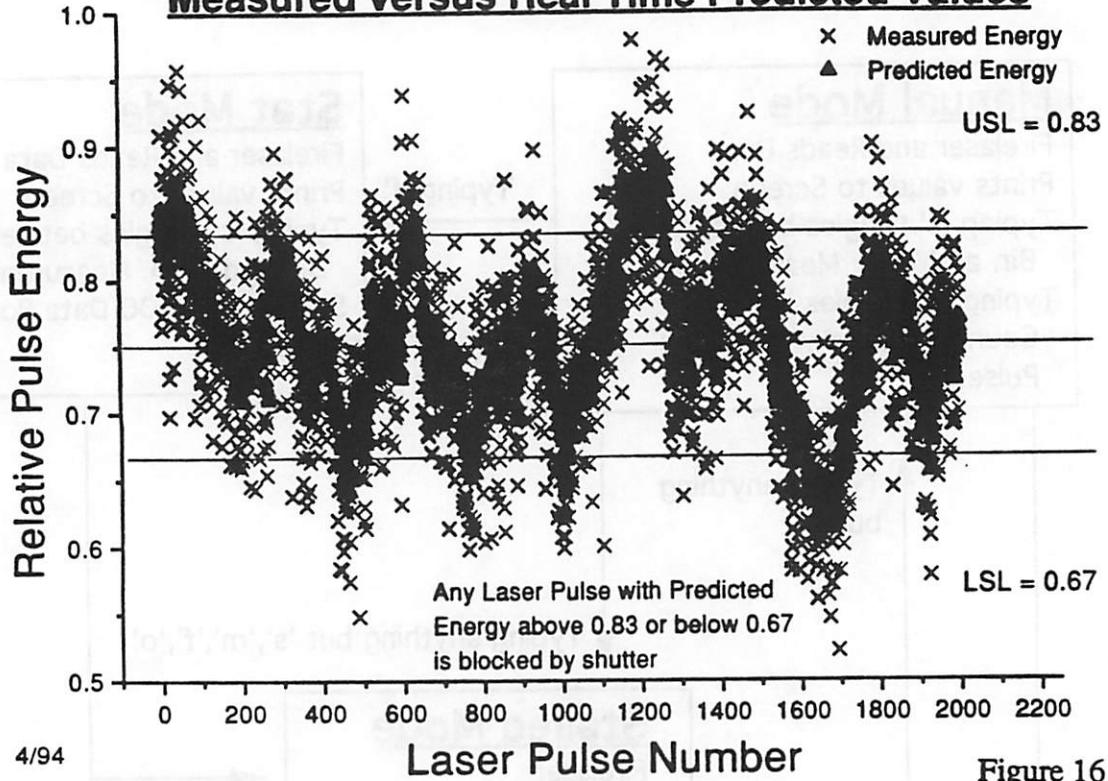
Figure 16

Results of Application of Time Series Filter
Measured versus Real Time Predicted Values
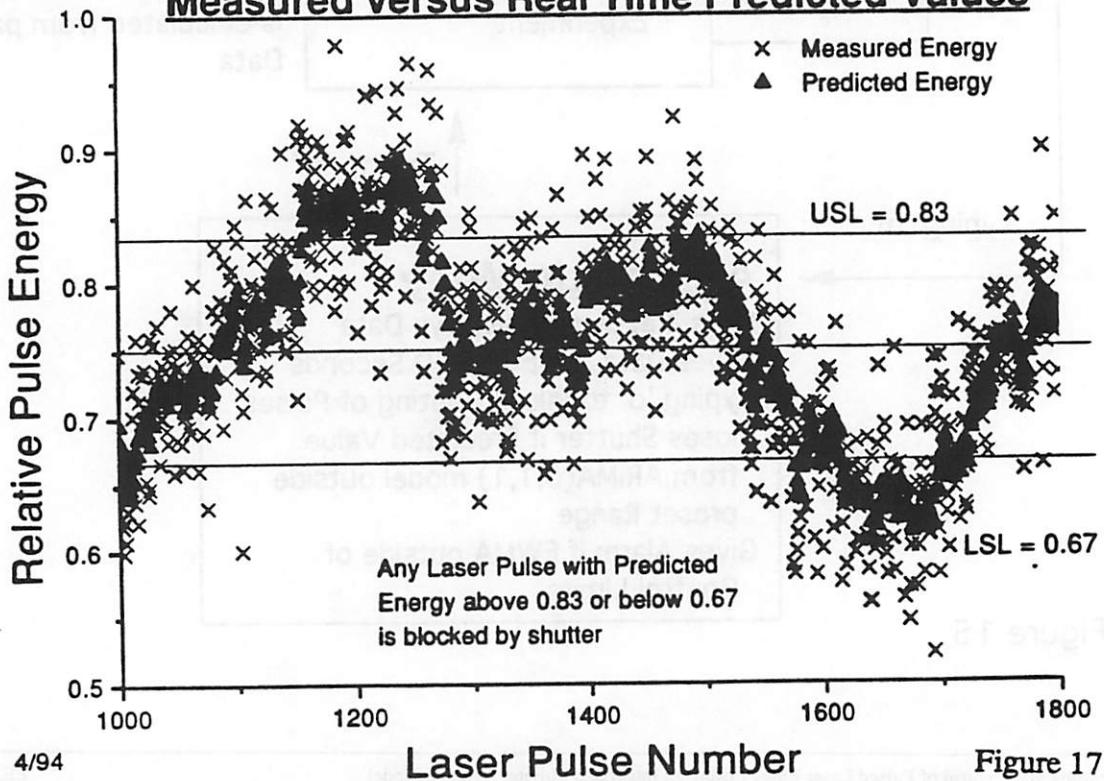
Figure 17

Distribution of Pulse Energies in Timer Series Filter Test
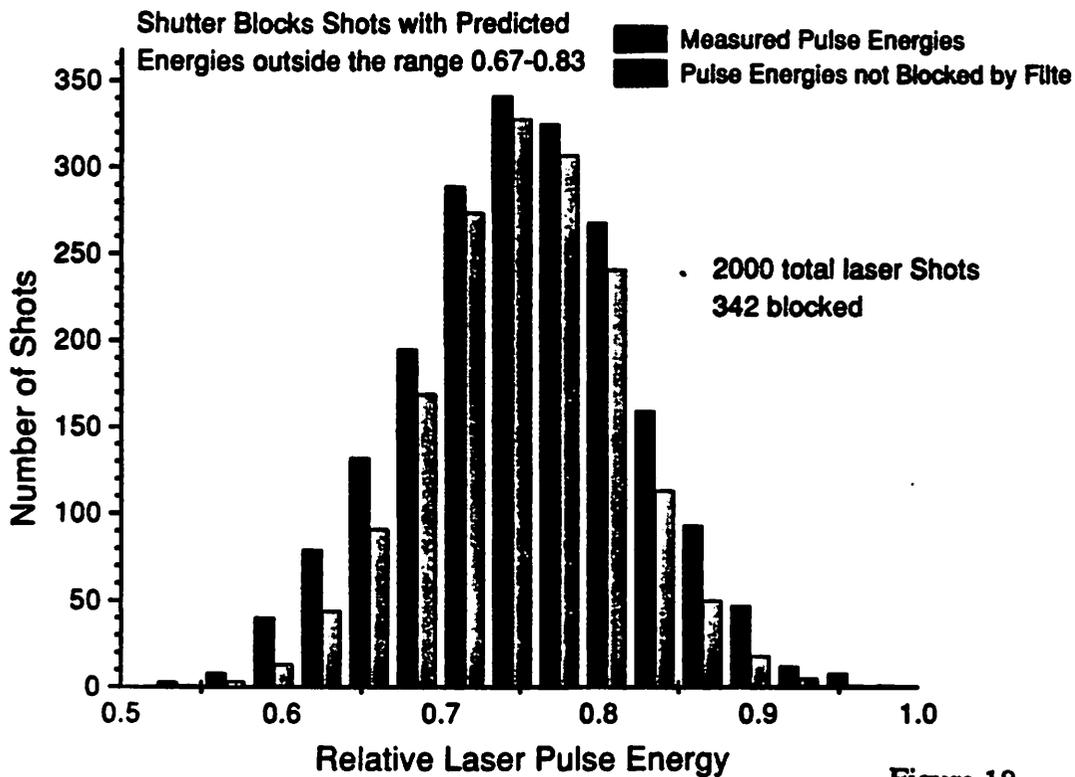
Figure 18

EWMA for Time Series Filter Test
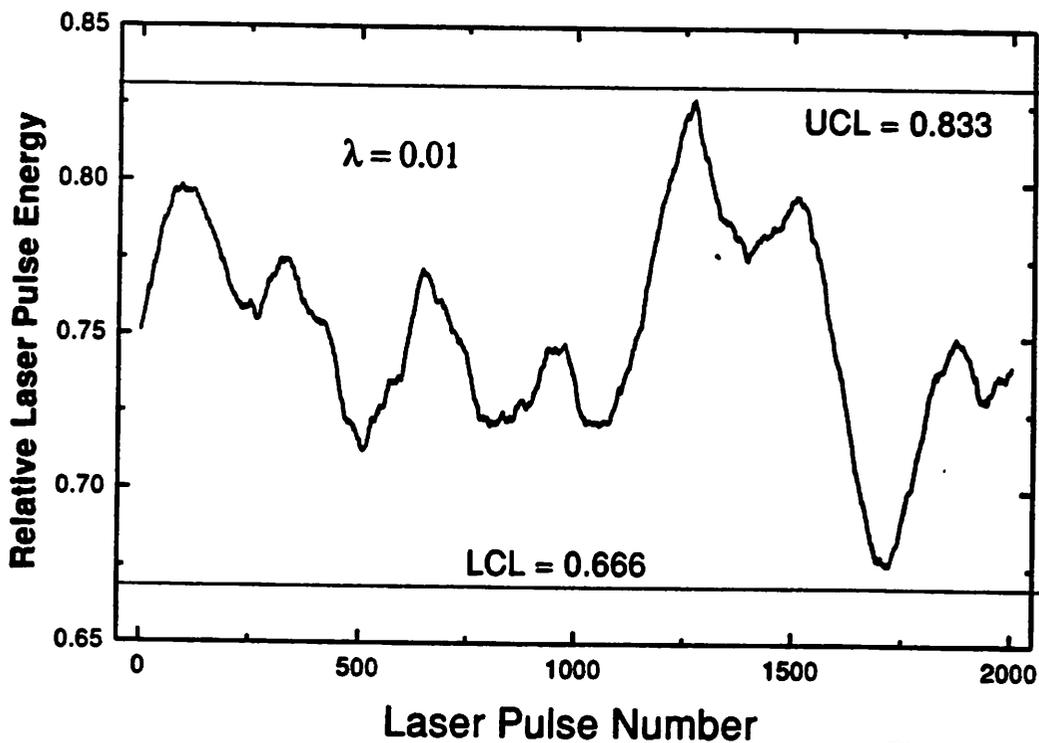
Figure 19

Developing a Statistical Process Control Method for the LAM Rainbow 4400 Plasma Etcher

Antonio J. Miranda

Control charts for individuals are investigated and developed to form a statistically robust monitoring process for the LAM Rainbow 4400 Plasma Etcher, located in the Microlab at the University of California, Berkeley. Much information can be collected from the etcher, including real time data from the Real Power Monitor, the HYT Particle Counter, and the SECII communications. It is desired that the control charts developed, monitor the essential aspects of the etcher, and give a good indication of the process etch variability.

1.0 Introduction

Control Charts are used in many environments in order to monitor the stability of a given process. One area which can benefit from judicious use of control charts is in the manufacturing of integrated circuits. Statistical Process Control (SPC), with all its various tools, is useful in achieving the process stability, and improving the capability through reduction in variability, required to remain competitive in the semiconductor industry. In the Microlab, the Lam Rainbow plasma etcher (LAM4) is just such a piece of equipment which can benefit greatly from SPC.

Presently, the microlab's monitoring procedure for LAM4 consists of etching 3 patterned wafers which have a layer of silicon dioxide(~1000 Angstroms) and polysilicon(~5000 Angstroms) grown on them. Pre- and post-etch polysilicon thicknesses are measured at five locations on the wafer (top, center, flat, left, right), using the Nanospec. The etching process consists of a pre-etch and a main etch step. The pre-etch step is done to remove any native oxide which may have formed on the wafers. This is done using an SF_6 feed gas. The main etch step is carried out using a Cl_2/He mixture. The input settings are summarized in Table 1.

The microlab tracks the etch rate for the main etch process only. This assumes a negligible amount of poly is etched during the pre-etch step. The average etch rate, wafer-to-wafer and within-wafer etch rate uniformities are presently monitored. The etch rate uniformity is calculated as follows:

$$\%uniformity = (Highest\ Rate - Lowest\ Rate)/\ (Highest\ Rate + Lowest\ Rate) * 100 \qquad (1)$$

The control standards used to determine if LAM4 is in control are:

- Etch Rate: 5500 A/min +/-10%

- Wafer-to-Wafer Etch Rate Uniformity: +/-10%

- Within-Wafer Etch Rate Uniformity: +/-10%

This method is fine for general analysis and control. However, in order to use the plasma etcher to its fullest extent, statistically defined control charts should be used to give a better indication of the etcher's performance. Control charts when defined and used appropriately give valuable visible feedback, which is not obtained using the present method. The objective of this project is to define appropriate control charts for use in the microlab, which are statistically sound, robust, and easily implemented. To this end, I will investigate all data available from LAM4. This includes not only the monthly process monitor results, but also the real time data collected.

Pressure (mtorr)	425
RF power (watts)	275
Gap (cm)	0.800
Cl_2 flow (sccm)	180
He flow (sccm)	400
He Clamp (torr)	8.0
Time (sec)	30

TABLE 1. Recipe for the main etch step of the process monitor test.

2.0 Methodology

2.1 Choosing the Proper Type of Control Chart

Since the process monitoring test is only run once a month on three wafers, it would not make sense to use the normal $\bar{X}$-R charts to monitor the parameters. A more appropriate chart is *the control chart for individuals*. This is the appropriate chart to use because presently data is collected too infrequently, and it would not be acceptable to have run lengths longer than one to catch processes which are out of control. The process variability is determined by using the moving range of two successive points. The moving range is defined as $MR_i = |x_i - x_{i-1}|$. In the control chart for individuals, the parameters are:

- UCL $= \bar{X} + 3\overline{MR}/d_2$

- Center Line $= \bar{X}$

- LCL $= \bar{X} - 3\overline{MR}/d_2$

$\bar{X}$ is the total average, $\overline{MR}$ is the moving range average, and d_2 is a statistically defined constant which depends on the sample size. The moving range could also have been plotted, but it would not have been correct in this case. Since the individual points are actually averages for the three wafers, the measurements are not necessarily independent and therefore, would not be correct to plot the moving range.

2.2 Choosing the Appropriate Parameters to Control

One of the challenges when first developing control charts, is choosing the proper parameters and sub-grouping the data correctly. Another obstacle is having the historical data available to set up the control charts. Both of these proved to be difficult issues to tackle for the LAM 4. Since the system has only been in operation for a little over a year very little historical data was available.

The process monitor test began in June 1993, and therefore only 10 months of data is available. Even worse, no historical data is available for the real time signals monitored by the plasma etch sensors. Therefore, what is presented here is the basis for the control chart, to be used in the future. In addition to the historical data, a few monitor wafers were run to begin building a database of real time signals, including SECSII, RPM and the particle counter.

2.2.1 Etch Rate: Grand Average

The average etch rate for each process monitor test is determined by taking the average of all measurements taken from the wafers. Since at this point the data appears to be random there is no reason to alter the way the average etch rate is computed. This should be investigated more closely when at least 30 to 35 points are available.

2.2.2 Etch Rate Uniformity: Wafer-to-Wafer

When monitoring the wafer-to-wafer uniformity the present test used equation 1 to determine the wafer-to-wafer uniformity. Presently this is the best method to use. However, there is reason to believe that these average values are not independent.

Recall how the monitor wafers are created. One of the process steps is to grow approximately 5000 Angstroms of poly-silicon. This is typically done in an LPCVD furnace, in the microlab (tylan 11). There is strong evidence that the furnace displays deterministic behavior from wafer-to-wafer. One of the main reason for this, is the gas depletion effects in the tube. This deterministic effect has been shown to effect the resistivity, and thus the doping concentration of the poly-silicon. If this deterministic effect is indeed true, then the etch rate may be greatly altered from run-to-run depending on the position the wafer held in the tylan furnace. Unfortunately, at present time the position which the wafer held in tylan 11 is not kept track of. There is no way of knowing, from the historical data, whether or not the variability in wafer-to-wafer etch rate uniformity is due to limitations in the plasma etcher or deterministic behavior from the tylan furnace. Therefore, the control charts developed for the wafer-to-wafer uniformity may have wider control limits then should be expected for our process.

Two things can be done to determine if the etch rate is correlated to the position the wafer held in the tylan furnace. First, keep track of the position the wafer held during the poly deposition. This may be done by scribing a number on each wafer and tracking this number using a log. This way future runs in the plasma etcher can be analyzed for any correlations to the LPCVD furnaces. Alternatively, if the process engineers feel confident that the deterministic effects are negligible for adjacent wafers, then no tracking is needed as long as three adjacent wafers are used in running each monthly monitor test. The control charts have been developed assuming the second approach was inherently used.

For wafer-to-wafer uniformities what is typically monitored is the range of etch rates per wafer. Therefore, the wafer-to-wafer uniformity control chart was calculated by taking the difference between the largest and smallest etch rate wafer averages.

2.2.3 Etch Rate Uniformity: Within Wafer

The issue of independence and sub-grouping had to be investigated closely for the within wafer uniformity. Presently the within wafer uniformity is determined as stated in eq. (1). Even though the deterministic effect which may have caused exaggerated control limits in the wafer-to-wafer parameter does not seem to have a dramatic effect, the measurements do not look to be independent. Instead, what seems to be occurring is what is termed the *bull's-eye* effect. This is

attributed to spacial non-uniformities, in the density of ions, in the sheath. This results in etch rates which vary as you move out from the center of the wafer. To determine if this effect was significant, a paired t-test was performed on the average of all center points verses the average of the four outer points for all wafers. The results in Table 2, illustrate that this effect is significant and should be accounted for. Even at a confidence level of 1%(from table: t-value = 3.646), the null hypothesis can easily be rejected (H_0: The average of the center measurements is the same as the four outer measurements).

Number of Points	30
Avg. of Difference	282.8
t-statistic	9.2128
99% Conf. Level	3.646
Conclusion	Reject H_0

TABLE 2. Paired t-test on the difference between outer four points and the center etch rate measurement.

A within wafer control chart can then be created by taking the difference between the average of the center measurements and the four outer values for the three wafers, and dividing by the smaller of the two values. This will give a value for the percent uniformity within each wafer.

2.2.4 Real Time Signals

Even though the real time signals are collected automatically, no historical data is available. This is attributed to a limit in available memory space, and is an engineering decision to determine when the data should be deleted from the database. Some monitor wafers were run in order to begin building up a database of information, but since only one run has been made it is not practical to try and perform any statistical analysis. However, in the future the data should be collected so that trends and overall averages can easily be extracted. This may lead to the need for regression charts, if the data has a slope during each wafer run. In addition, only a few parameters should be kept track of, in order to make the database more manageable. For instance, maybe only the RPM and particle counter data should be collected, and the SECSII information discarded. This may need to be done because the SECSII data is coupled to the input settings and, therefore, is not independent and will not give a true indication of the etcher's stability. A more detailed analysis should be performed when sufficient data is available.

3.0 Results

The next step in setting up the control charts is to do an initial plot of each parameter. The data calculated for each control chart, as well as the date of when each run was made and the number of wafers etched are summarized in table 3.

Date	# of wafers	Grand Average	Wafer-to-Wafer	Within Wafer
June 21, 1993	2	5276	580	0.0509
August 25, 1993	1	5204	----	0.0683
September 14, 1993	3	5079	116	0.0356
October 19, 1993	3	4329	206	0.0098
November 16, 1993	3	5210	182	0.0489
December 17, 1993	3	5352	77	0.0442
January 28, 1994	3	5554	16	0.0660
February 28, 1994	3	5636	222	0.0414
April 22, 1994	3	5311	121	0.0114
May 1, 1994 (set1)	3	5400	229	0.0936
May 1, 1994 (set2)	3	5040	780	0.1241

TABLE 3. Data used to build preliminary control charts.

3.1 Plotting the Control Charts: First Pass

The control charts presented in this section are preliminary, since the control limits should not be fixed until more data has been collected. Figure 1 shows the results for the three control charts as determined from the historical data. Western Electric rules were applied to determine if any alarms were present.

3.2 Assignable Causes for Out of Control Points

A couple of initial alarms are visible on the control charts. Using the problem log on the wand, possible reasons for the alarms were found.

In the overall average etch rate control chart, point 4 dropped below the lower control limit. During this time period, September - October 1993, The Lam Rainbow Plasma Etcher experienced several different problems. The major problem was that the Freon mass controller drifted and the flow of Freon in the chamber was not zero. This was traced back to loose +/-15 V_{dc} connectors for the gas control PCB. The problem was not detected until the flow exceeded the threshold manually set on LAM 4. However, before the problem was corrected a wafer broke in the chamber and it had to be vented. Therefore, the chamber was probably cleaned while it was opened to remove any residue which may have accumulated on the walls and electrodes. This can obviously effect the etch rate of the system. Another problem which was logged on the wand was a gap error. If the gap between the wafers is not set correctly, the etch rate may be altered depending on the dimensions which actually exist.

The wafer-to-wafer uniformity etch rate also displayed an alarm. In this case the first month of the monitor test was greater than the upper control limit. Once again the chamber was opened a

couple of days before the monitor test was run because a wafer was stuck in it. The chamber, in all likelihood, was wiped down while it was open. This once again could have caused a shift in the etch rate uniformities. Problem with the Helium clamp also arose when the system was being brought back on line. This was corrected by simply cleaning all electrical connections leading to the clamp.

The last control chart, the within wafer uniformity, also had a point out of control. This error occurred in the last run, which was recently performed (May 1, 1994). No problem logged on the wand could be assigned to this alarm. Therefore, either a shift is taking place in the etcher or this may well be the natural variation of the system. At this point this can not be determined, because enough data points do not exist. This should be monitored closely in the future.

3.3 Revised Control Charts

The control charts were re-evaluated with the points which could be explained removed when determining the control limits. The new control charts, shown in figure 2, now have all its points in control. These control charts are only the basis for charts to be built upon in the future. If a shift does exist for the within wafer uniformity control chart, then new control limits should be determined from the point in time the shift occurred.

4.0 Implementation

The control charts developed here should be implemented automatically, in order to make it simple. At present time the microlab is in the process of automating their statistical process control charts. This is being done using a combination of script files and UNIX interface and data retrieval commands. The software may be found on the argon cluster by typing *spcwand*. The program from the lam4 monitor test should retrieve all data stored through the nanospec interface and calculate the following values:

- Total average etch rate = average of all readings

- Wafer-to-wafer uniformity = Highest etch rate wafer average - lowest etch rate wafer average

- Within wafer uniformity = (Average etch rate of Center pts. - Average etch rate of Outer pts.) / (the minimum of the two readings)

These values should then be put into their respective control charts and compared to the control limits.

5.0 Conclusions

Control charts have been developed to effectively and efficiently monitor the LAM Rainbow plasma etcher. The control charts monitor the total average etch rate, wafer-to-wafer uniformity and within wafer uniformity etch rates. Real time signals were also investigated, however, since no historical data was available, only recommendations were made. The control charts proposed should provide a more rigorous indication of the ethers capabilities. It should be noted, however that the control limits presented here should not be fixed at this point, because sufficient data is yet not available.

[1]   D.C. Montgomery, *Introduction to Statistical Quality Control*, John Wiley & Sons, New York, 1985.

[2]   D. Hebert et al., *The Microlab Manual*, Ch. 1.14.

[3]   D. M. Manos and D. L. Flamm, *Plasma Etching an Introduction*, Academic Press, Inc., New York, 1989.

[4]   Maria Perez, Microlab, private conversation.
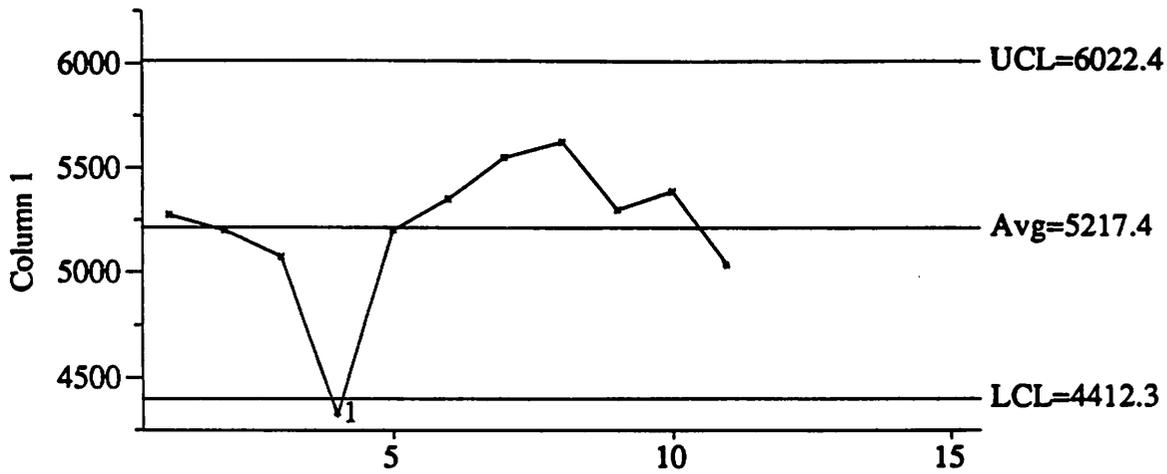
(a) Total average etch rate
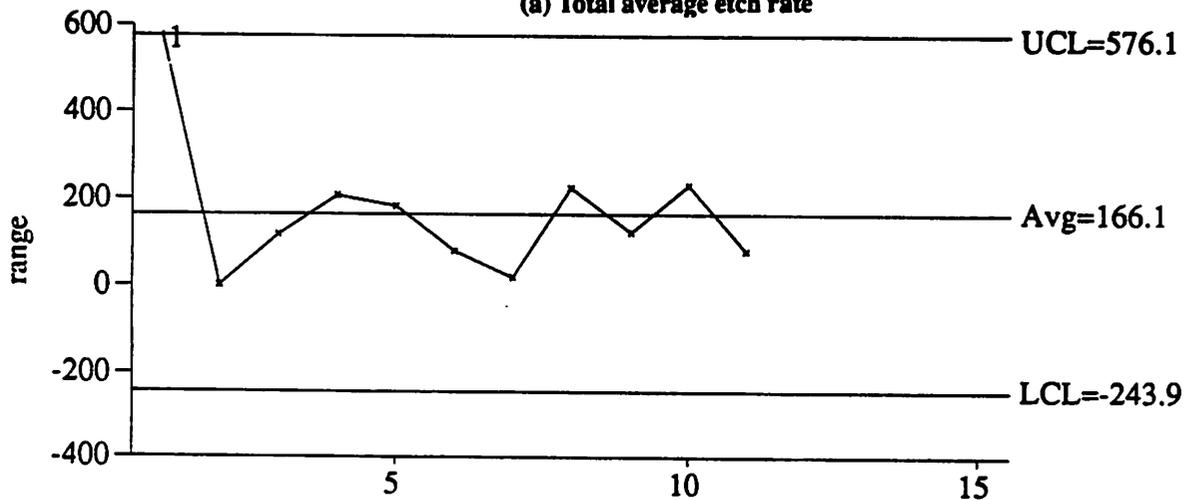
(b) Wafer-to-wafer etch rate uniformity
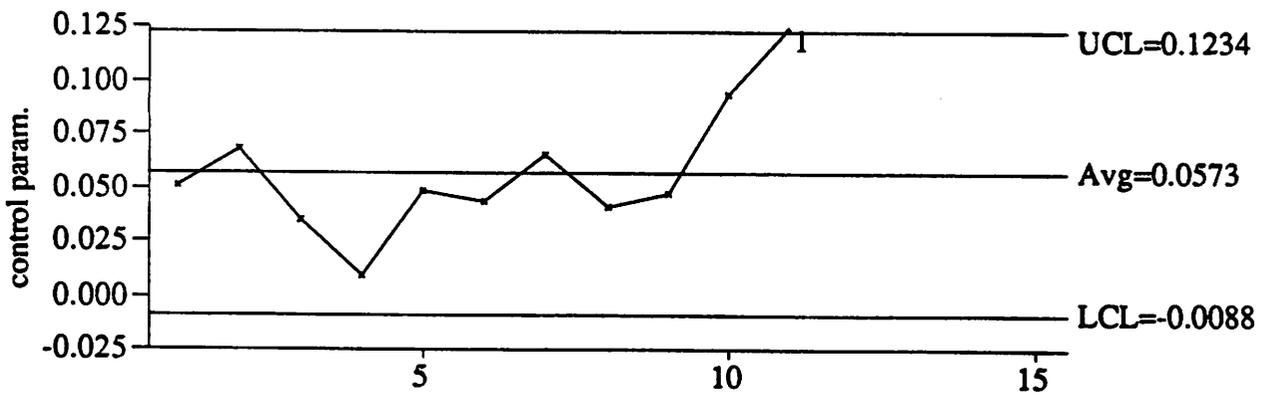
(c) Within wafer etch rate uniformity

FIGURE 1. Control Charts with all data points included
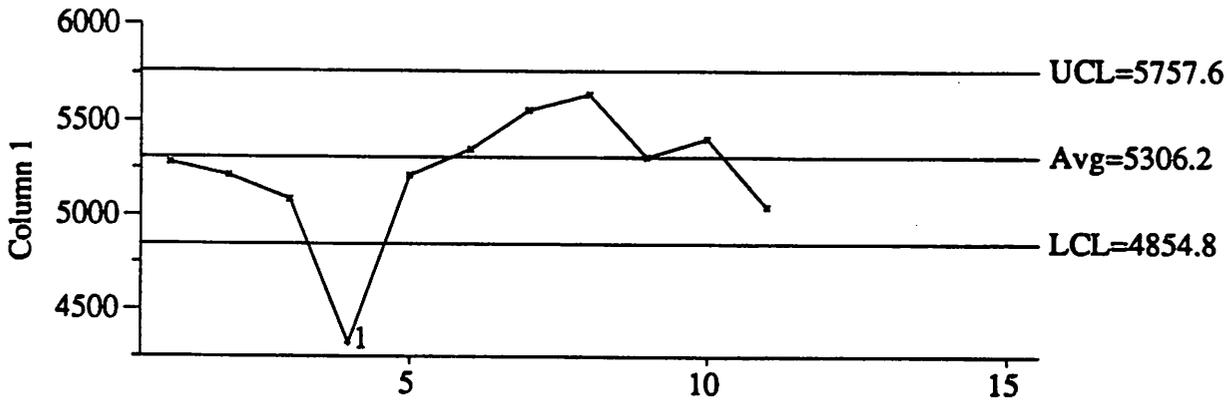
(a) Total average etch rate
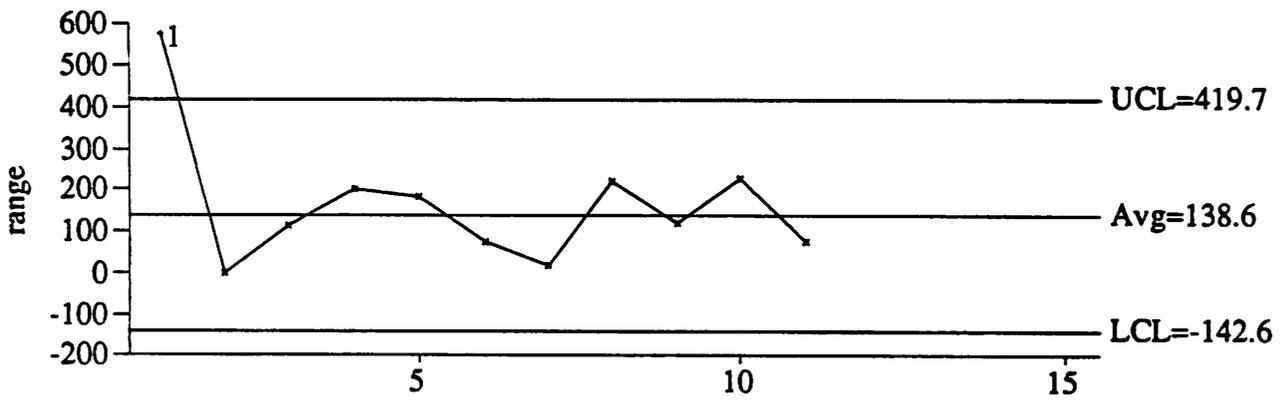
(b) Wafer-to-wafer etch rate uniformity
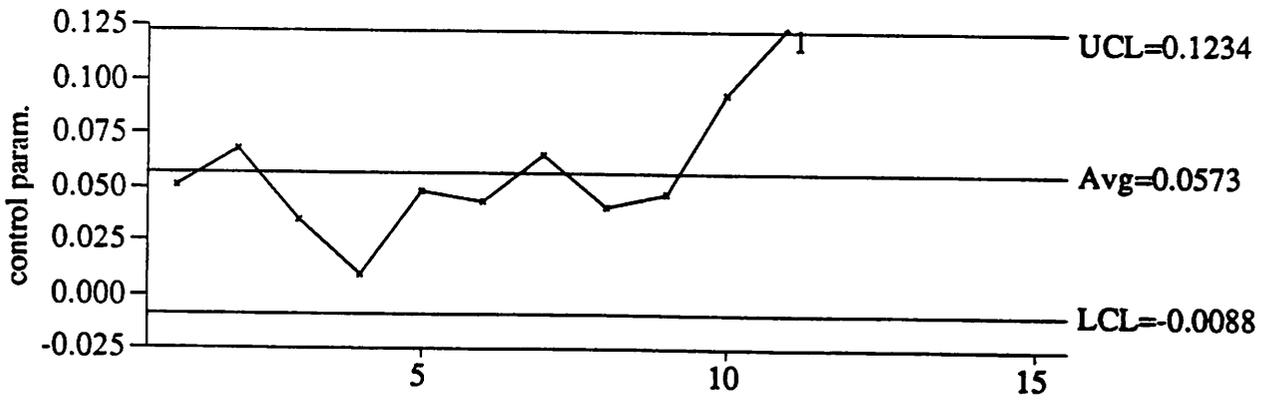
(c) Within wafer etch rate uniformity

FIGURE 2. Control charts for data excluding points with assignable causes

Predicting Worst Case SPICE Files Using BSIM3

James C. Chen

In this work, we present a methodology to predict worst and best case SPICE files using a subset of the entire BSIM3 device model parameter set. This methodology takes advantage of the physical robustness of BSIM3 to simplify the traditional analysis using Monte Carlo simulations. The results show that this methodology is sensitive to a the number of Monte Carlo simulation points as well as the goodness of fit of linear regression models.

1.0 Introduction

1.1 The Problem Defined

It is well known fact that the production of IC's suffers from process fluctuations, which in turn manifest themselves in devices, ultimately affecting circuit performances. The attempt to characterize this process variation necessitated the methodology of SPICE Worst, Best, and Nominal files. These files (also called process "corners") are given to the circuit designer in order to ensure that all their designs will assume proper operation.

In order to track device variability (and hence circuit performance spreads), traditional methods for determining process corner files rely on the explicit use of Monte Carlo Analysis. Data from a test run of a given process is collected to estimate the mean and sigma values for device model parameters (and their correlations) for the device model to be used. A Monte Carlo analysis is executed and a particular circuit performance variable of interest (Id_{sat}, τ_p, power, etc.) is examined. Worst case files are then chosen at the $+/-3\sigma$ levels for these circuit variables. Various modifications of this methodology have been proposed but most, if not all, share the same problems.

These problems are all concerned with the efficiency and physical robustness under which specific variations in process variables (e.g. T_{ox}, L_{eff}, W_{eff}, N_{peak}) are translated into device model parameters variations. This difficulty arises because traditional device models attempt to capture device behavior through the use of fitting parameters to match non-physical equations to actual device I-V curves. These fitting parameters seldom have no physical meaning outside of the simulation domain. Hence, it is very difficult to reconcile real, physical data such as Rds (parasitic resistance) collected from fab electrical tests to specific device model parameters.

1.2 Justification for BSIM3

Given such difficulties many present methodologies are now using so-called *physical* models to bridge the gap between the process domain and device domain. These models range from very simple models modeling only the on-state transistor current Id to many complex proprietary formulations[1,2,3]. Still, the availability of an accurate, *deep-submicron* (down to L= 0.25μm), physical model that is computationally efficient and readily available in SPICE has been sparse. BSIM3 (Berkeley Short-Channel IGFET Model)[4] fulfills this requirement and was chosen for this study.

In BSIM3, there are a total of 33 DC model parameters of which 7 are the basic parameters. These 7 parameters, in addition to having a physical meaning, are responsible for determining the majority of device characteristics. The model has been formulated in a way that most of the remaining 26 parameters are either functions of these basic parameters or contribute little to first order effects. A summary of these 7 parameters is given in Table 1 along with their physical meanings.

SPICE Parameter	Physical Meaning	Units
VthO	Threshold Voltage @ V_{bs} = 0v for Large W and L	V
Tox	Thickness of Gate Oxide	M (meters)
dl	Channel Length Reduction on One Side	M
dw	Channel Width Reduction on One Side	M
K1	First-Order Body Effect Coefficient (proportional to N_{peak})	$V^{1/2}$
K2	Second-Order Body Effect Coefficient (related to N_{sub})	-
RdsO	Parasitic Resistance Between Source and Drain	ohms

TABLE 1. Basic Parameters for BSIM3

2.0 Methodology

The goal of this experiment was to use the above 7 basic parameters to generate the Worst and Best case SPICE file for the average propagation delay (τ_p) and average power dissipation (P_{avg}) for a 21 stage ring oscillator. This particular circuit was chosen for its ease in evaluating both speed and power dissipation, two of the most critical circuit performance variables in digital circuit design.

The methodology required the use of Principal Component Analysis (PCA) to resolve the possible correlation among the basic BSIM3 device model parameters. This is important because PCA will enable us to carry out the remaining analysis with **independent** pseudo-variables which greatly simplify mathematical equations and geometrical concepts. This transformation of actual SPICE model variables into pseudo-variables is simple; each pseudo-variable is a linear combination of the SPICE variables. Thus as a result of PCA, we have:

$$PV_1 = P_{1,1}{}^*VthO_{nmos},+...P_{1,6}{}^*RdsO_{nmos}+P_{1,7}{}^*VthO_{pmos},+...P_{1,12}{}^*RdsO_{pmos} \qquad (1)$$

$$PV_n = P_{n,1}*VthO_{nmos}+...P_{n,6}*RdsO_{nmos}+P_{n,7}*VthO_{pmos}+...P_{n,12}*RdsO_{pmos}$$

The variables PV_i are the n pseudo-variables (also called principal components) resulting from the PCA analysis. Each successive PV explains an incremental amount of the variance in the original data set until the last pseudo-variable, PV_n, at which point all the variation in the data set has been accounted for. These pseudo-variables can now be used in a Monte Carlo experiment to build a simple linear regression model for each of our circuit performance variables. These models will be of the form:

$$\tau_p = f(PV_1, PV_2, ...PV_n) \quad \text{and} \quad P_{avg} = f(PV_1, PV_2, ... PV_n) \tag{2}$$

Once these equations are known, a non-linear optimizer is used to find the maximum or minimum values for each of the above two objective equations separately. These optimized values of τ_p and P_{avg}, hence, will be our "worst" and "best" case values. For each of these values there will be a unique set of corresponding pseudo-variables (the optimization is done in the rotated space of the pseudo-variables). Once these PV's are determined, the PCA equations in (1) above can be then used to transform them into meaningful BSIM3 basic model values corresponding to our worst and best case files.

Before we employ the use of the non-linear optimizer for the equations defined in (2) above, we must define the constraints of our optimization problem. These constraints arise from the multivariate normal distribution of our pseudo-variables. Each PV is normally distributed since it is a linear combination of the basic BSIM3 parameters, each is assumed to follow a normal distribution (See Figures 1 through 4). When we lump all n PV's together to form a multivariate distribution and decide on a specific probability contour, we produce the constraint for our non-linear optimization. Mathematically, this implies:

$$(x - \mu)^T \Sigma^{-1}(x - \mu) = \text{constant} \tag{3}$$

The left hand side of equation (3) above defines the multivariate distribution of our PV's: $(x - \mu)$ is a n x 1 matrix of our pseudo-variables minus its mean and Σ is its n x n covariance matrix. These variables follow a multivariate Gaussian distribution. The constant is usually set to some probability value which corresponds to the pth cumulative percentile.

Graphically, this procedure can be visualized with the help of Figure 5 for the case of two PV's (n=2). Our equation for the constraint would resemble that of an ellipse with the boundaries of the ellipse bounding the pth cumulative percentile of the multivariate distribution.
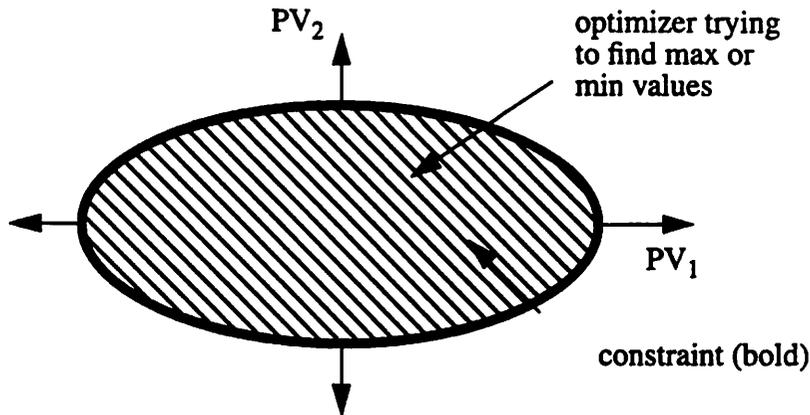
FIGURE 5. Graphical Representation of Optimizing a Nonlinear Function of Two Variables

3.0 Implementation

BSIM3 .model files for both NMOS and PMOS were extracted from a 0.8μm process. There were a total of 40 SPICE .model files extracted from one wafer corresponding to 40 individual dies. From these files the estimate for the mean and sigma values for each of the basic BSIM3 parameters in Table 1 were calculated, with the exception of the variable T_{ox}. T_{ox} was extracted with a range of 5 Angstroms across the wafer. Such a small variation could be considered to be insignificant compared to the variation seen for the other basic parameters. As a result, the total number of basic parameters was 12, 6 for each type of MOS device.

The 12 variables exhibited some correlation between one another. For instance, dw_{nmos} and dw_{pmos} had a correlation coefficient (ρ) equal to 0.79, while the correlation between K1 and K2 for both NMOS and PMOS was -0.92 and -0.94, respectively. The results of PCA (see Table 2) show how 12 pseudo-variables could be used to explain 100% of all the variance in our original data. Even though the percentage of variance explained by each successive principal component decreases rather sharply, it is important to keep all 12 pseudo-variables, to ensure that there are enough linear equations to map from pseudo-variable space directly to real SPICE parameter space and vice versa.

This proved to be important for our Monte Carlo Analysis which followed. Random numbers (from a normal distribution) were generated in pseudo space which were then transformed back into real values for the 12 BSIM3 basic parameters to be used in SPICE. Thus for every Monte Carlo point, 6 BSIM3 parameters were varied for each type of device. The remaining 27 DC parameters were fixed at their values extracted from the center of the die. This is because the physical nature of BSIM3 and the assumption that the variation in these 7 (now 6) basic model parameters will explain the most significant part of the variation in the current drive of the device.

	PV1	PV2	PV3	PV4
Vth0(nmos)	-1.492108e-3	5.905668e-1	-8.517796e-2	3.163409e-2
dl(nmos)	3.726877e-1	-1.878210e-1	2.927029e-1	-2.505171e-2
dw(nmos)	2.931581e-1	2.712105e-1	1.378165e-1	-4.257756e-1
K1(nmos)	3.106211e-1	3.815624e-1	1.784557e-1	3.653242e-1

	PV1	PV2	PV3	PV4
K2(nmos)	-2.723438e-1	-2.953161e-1	-1.287262e-1	-4.852363e-1
Rds0(nmos)	2.487619e-1	-3.782492e-1	2.874006e-1	-3.97424e-2
Vth0(pmos)	3.205826e-1	6.753631e-3	-4.538006e-1	-1.145988e-1
dl(pmos)	2.08159e-1	-2.142144e-1	4.554936e-1	8.914575e-2
dw(pmos)	3.576423e-1	1.397914e-1	9.897135e-2	-3.829252e-1
K1(pmos)	3.682049e-1	-1.663373e-1	-3.84664e-1	1.697973e-1
K2(pmos)	-3.366706e-1	2.356324e-1	4.412387e-1	-1.695413e-1
Rds0(pmos)	-1.403111e-1	-1.364805e-1	8.491283e-2	4.704298e-1
Variance	4.17309	2.006184	1.603642	1.496005
% of total	34.77575	16.7182	13.36369	12.46671
Cumultative %	34.77575	51.49396	64.85765	77.32436

	PV5	PV6	PV7	PV8
Vth0(nmos)	-4.703119e-1	1.058689e-2	-3.84398e-1	3.520794e-1
dl(nmos)	-1.631401e-1	-1.673190e-1	-2.566214e-1	-1.088275e-2
dw(nmos)	-8.240403e-2	3.275626e-1	2.382398e-1	2.215665e-1
K1(nmos)	2.078682e-1	-4.247258e-2	-4.318194e-2	6.734491e-2
K2(nmos)	-3.536356e-1	6.153219e-2	-1.281194e-1	2.166942e-1
Rds0(nmos)	3.417080e-1	3.130584e-1	-4.244618e-1	4.104748e-1
Vth0(pmos)	-3.091586e-2	1.473436e-1	-5.434611e-1	-5.569698e-1
dl(pmos)	-5.496431e-1	-3.209041e-1	5.966834e-2	-2.131889e-1
dw(pmos)	1.340821e-2	2.469910e-1	3.569030e-1	-2.885404e-1
K1(pmos)	-1.630712e-1	-6.439068e-2	1.989695e-1	3.553272e-1
K2(pmos)	9.227592e-2	6.226508e-2	-2.423637e-1	-1.554870e-1
Rds0(pmos)	-3.493127e-1	7.531812e-1	8.098749e-2	-1.322661e-1
Variance	7.832995e-1	7.58259e-1	4.157197e-1	3.098587e-1
% of total	6.5275	6.31883	3.46433	2.58216
Cumultative %	83.85186	90.17069	93.63502	96.21717

	PV9	PV10	PV11	PV12
Vth0(nmos)	6.020433e-2	2.341407e-1	-2.862604e-1	-9.768466e-2
dl(nmos)	-7.790794e-1	8.937161e-2	3.975985e-2	-4.907228e-2
dw(nmos)	-6.709257e-2	-6.442109e-1	2.758212e-2	4.141946e-3
K1(nmos)	8.636655e-2	7.179719e-2	4.534965e-1	5.682694e-1
K2(nmos)	1.650754e-2	1.660058e-1	3.759891e-1	4.745300e-1
Rds0(nmos)	3.197551e-1	9.999898e-2	-1.821015e-1	-4.510925e-2
Vth0(pmos)	1.364084e-1	-1.922844e-1	1.015581e-1	3.064068e-4
dl(pmos)	4.651098e-1	-1.507048e-1	-4.952238e-2	2.323582e-2
dw(pmos)	8.862903e-2	6.396207e-1	-8.038769e-2	-2.952102e-2
K1(pmos)	1.079737e-1	9.326037e-2	4.798739e-1	-4.612373e-1
K2(pmos)	6.444173e-2	4.222229e-2	5.320785e-1	-4.725918e-1
Rds0(pmos)	-1.278026e-1	1.379345e-2	5.384005e-2	1.985559e-2

	PV9	PV10	PV11	PV12
Variance	2.313289e-1	1.579808e-1	5.335946e-2	1.126996e-2
% of total	1.92774	1.31651	0.44466	0.09392
Cumultative %	98.14491	99.46142	99.90608	100

TABLE 2. Principal Component Loadings

The variations in the 27 other parameters are not as significant. The center die was picked under the assumption that the process was "centered" at this point.

A 21 stage ring oscillator with a fanout of two per stage was then simulated and its τ_p and P_{avg} measured for every Monte Carlo simulation point (50 points in all). Simple linear regression was subsequently performed on the 12 independent PCA pseudo-variables and the dependent circuit variable (τ_p or P_{avg}). The result is a relationship between a circuit performance variable and a set of pseudo-variables, which is just another method of representing real SPICE device parameters. Another linear model with quadratic pseudo-variable dependence was also formulated but the fit of the model (see Section 4.0) was no better than the one below. Explicitly, these equations are the following:

$$\tau_p \text{ (psec)} = 181.80 - 2.051PV_1 + 0.955PV_2 - 0.865PV_3 + 0.547PV_4 + 1.024PV_5 + 1.823PV_6 +$$
$$1.567PV_7 + 1.06PV_8 + 2.455PV_9 - 0.956PV_{10} - 1.410PV_{11} + 9.259PV_{12} \qquad (4)$$

$$P_{avg} \text{ (mW)} = 13.170 + 0.116PV_1 - 0.156PV_2 + 0.162PV_3 - 0.009PV_4 - 0.0646PV_5 - 0.217PV_6 -$$
$$0.0877PV_7 - 0.0705PV_8 - 0.119PV_9 - 0.0202PV_{10} - 0.0715PV_{11} - 0.052PV_{12} \qquad (5)$$

As previously stated, it is necessary for us to maximize or minimize these τ_p and P_{avg} with respect to some constraint. A good discussion of the formulation of this constraint can be found in [5,6]. The important point to remember is that this constraint is a multivariate distribution of pseudo-variables resulting from the previous Principal Component Analysis. Since the PCA component loadings are calculated with normalized BSIM3 parameters, the pseudo-variables have zero means as well. Equation (3) then simplifies to:

$$(\mathbf{x})^T \Sigma^{-1}(\mathbf{x}) = \text{constant} \qquad (6)$$

with $\mu = 0$. The covariance matrix, Σ, also assumes a simple form containing only non-zero values in its main diagonal entries due to the orthogonality of the pseudo-variables. Hence, our constraint is the following:

$$(PV_1)^2/4.173 + (PV_2)^2/2.00618 + (PV_3)^2/1.6036 + (PV_4)^2/1.496 + (PV_5)^2/7.833e\text{-}01 + (PV_6)^2/$$
$$7.583e\text{-}01 + (PV_7)^2/4.157e\text{-}01 + (PV_8)^2/3.099e\text{-}01 + (PV_9)^2/2.313e\text{-}01 + (PV_{10})^2/1.58e\text{-}01 +$$
$$(PV_{11})^2/5.336e\text{-}02 + (PV_{12})^2/1.127e\text{-}02 = 21.03 \qquad (7)$$

The value of the constant used was 21.03 corresponding to 12 degrees of freedom (12 pseudo-variables) at a cumulative 95% value for a Chi- Squared distribution.

Once equations (4), (5), and (7) have been obtained, they were used with a general non-linear optimizer called "hanpal." The output of hanpal is hopefully the set of "worst" and "best" case parameters (in pseudo space) corresponding to a unique set of actual BSIM3 SPICE parameters (the basic 12 parameters). The C input file used in hanpal is given in the Appendix.

4.0 Results and Discussion

The results of the optimized worst and best case files for the ring oscillator circuit were compared to the actual performance distribution obtained by simulating the same circuit 40 times will complete BSIM3 .model decks extracted from each individual die. The results are summarized in Table 3.

File	Predicted Value	Location in Actual Distribution ($\bar{\sigma}$)
Best Case P_{avg} (minimum)	11.82mW	-3.40
Worst Case P_{avg} (maximum)	14.86mW	+3.11
Best Case τ_p (minimum)	172.46 psec	-0.68
Worst Case τ_p (maximum)	202.19 psec	+4.35

TABLE 3. Comparison of Predicted Worst and Best Case versus Actual

Several observations can be made. First, we see immediately that our methodology was more precise at predicting the +/- 3σ values for P_{avg} than for τ_p. Secondly, prediction of worst case files for P_{avg} falls a little beyond the 3σ limits, but this error is symmetric. This is not the case for τ_p where the minimum τ_p is severely under-estimated and the maximum τ_p over-estimated. These errors can be traced to two main steps in our methodology.

First, in the formulation of the equations (4) and (5) above we assumed that a 50 point Monte Carlo analysis was adequate in generating enough data points for performing the linear regression. The "rule of thumb" concerning the convergence of Monte Carlo is 100-150 simulation points. Thus our Monte Carlo might not have given us enough data points to adequately bound the distribution of both circuit performance variables. This is a failing not the methodology but of the resources available at the time of the project. Certainly, more points in the Monte Carlo analysis could be performed.

If inadequate coverage of the Monte Carlo analysis was the only plausible explanation of the errors in our methodology, then we would expect that *both* sets of predictions would be similarly in error. This is not true and leads us to suspect that there are other underlying factors at work. One possibility could be the poor model given by our linear regression model (see ANOVA Tables 4 and 5).

As we can see, the result of our linear regression was far more satisfactory in for the prediction of P_{avg} than for τ_p. We evaluate this based upon the value of R-squared and the F-value. R-squared is the regression sum of squares divided by the total sum of squares and represents the percent variation in the response explained by the regression model. The results of the linear regression on P_{avg} returned a R-squared value of 0.94. Although a large R-squared value usually denotes a good fit of the model, this unfortunately is not always so, due to the fact that R-squared usually increases as more terms are added to the model. A better assessment is to use the adjusted R-squared value which was 0.92 in this case.

Since the R-squared statistics is not a good assessment of overall fit, the F-value is used. The F-value is the ratio of the regression mean square to the residual mean square and is a measure of the significance of the fit as a whole. The F-value is usually used in a significance test of hypothesis to determine the probability of whether observing a F-value as large or larger than the one observed (assuming that all the coefficients are zero)[7]. Thus a high F-value corresponding to a small significance level are good indicators of the goodness of fit. For P_{avg}, this was indeed the case for F-value was equal to 48.48 and the significance level was determined to be zero.

The ANOVA table tells another story for τ_p. The adjusted R-squared coefficient was 0.699842 and the F-value was 10.52. Even though the significance level of the F-value was small (1e-08) the fit of this particular linear regression can be considered poor relative to that for P_{avg}. A new linear regression with quadratic independent terms was performed in the hopes of improving fit, but this was also unsuccessful for similar values for R-squared and F-value were obtained. Subsequent results from hanpal optimization on this regression relation were therefore not accurate.

Error Source	Sum of Sq.	DF	F	Sig. Level
Regression	12.788268	12	48.48271	0
Residual	0.813290	37		
R-squared (adjusted)			0.920814	
Std. Dev. of Regression			0.148259	

TABLE 4. ANOVA Table for Pavg Model

Error Source	Sum of Sq.	DF	F	Sig. Level
Regression	1924.79443	12	10.520635	1e-08
Residuals	564.108798	37		
R-squared (adjusted)			0.699842	
Std. Dev. of Regression			3.9046369	

TABLE 5. ANOVA Table for τp Model

5.0 Conclusions

The methodology presented in this paper allows for the prediction of worst and best case SPICE .model files based upon the analysis of a subset of the BSIM3 parameter set (known as the basic parameter set). However, the results point to two limitations of this methodology. First, enough points in the Monte Carlo analysis must be performed to adequately allow the model of the circuit performance variable to converge to its actual distribution. Secondly, care must be taken in performing/evaluating the linear regression. It has been shown that satisfactory results will only be obtained if the fit is good.

As a parting observation, it is important to understand that the generation of worst and best case SPICE files requires an intimate knowledge of the device model. BSIM3 is a good illustration of this concept. This model was formulated and implemented into SPICE with as many physical relationships as computationally practical. Because of the prevalence of the underlying physics involved, most of the variation in a device performance can and should be only significantly influenced by the most physical parameters such as those in Table 1. Therefore, it naturally follows that our methodology is only concerned with this subset of basic parameters. Conceivably, all 33 parameters could be included in the analysis with PCA, Monte Carlo, linear regression, and hanpal optimization but the increase in computational cost would be enormous.

Acknowledgments: The author would like to thank the help of many whom have given their time and shared their knowledge on this project. Those who come to mind are: Sovarong Leang for his help in introducing the hanpal optimization program, Sherry Lee for her assistance with the statistical package SPlus, Eric Boskin for his many references on Multivariate Statistics and another statistical package, RS1. Finally I would like to thank Professor Spanos for giving me many suggestions/advice concerning my methodology as well as for spending a memorable afternoon with me debugging the intricacies of gcc compilation of the hanpal optimizer. Thank you all very much.

<u>References</u>

[1] Herr, Norman, John J. Barnes, "Staistical Circuit Simulation Modeling of CMOS VLSI," IEEE Transactions on Computer-Aided Design, Vol. CAD-5, No. 1, January 1986, pp. 15-22.

[2] Yang, Ping, Dale E. Hocevar, Paul C. Cox, Charles Machala, and Pallab K.Chatterjee, "An Integrated and Efficient Approach for MOS VLSI Statistical Circuit Design," IEEE Transaction on Computer-Aided Design, Vol. CAD-5, January 1986, pp. 5-14.

[3] Spanos, Costas, Lecture and Slides on "IC manufacturability and the Interaction between Fab Lines and Designers," Berkeley Education Series for EE290W, Spring 94.

[4] Huang, Jian Hui, Z. H. Liu, M. C. Jeng, K. Hui, M. Chan, P. K. Ko, and C. Hu, "BSIM3 Manual Version 2.0," (C) 1994, Electronic and Research Laboratory, Department of Electrical Engineering and Computer Sciences, University of California at Berkeley.

[5] Mardia, Kent, Bibby, **Multivariate Statistics**, Academic Press 1979, pp. 36-41

[6] Morrison, Donald, **Multivariate Statistical Methods**, 3rd Edition, McGraw-Hill 1990, pp. 85-90

[7] Box, G.E. Hunter, W.G. Hunter, J.S. Statistics for Experimenters, (C) 1978 by John Wiley & Sons, Inc. New York, New York.

Figure 1

Figure 2

Mean = .7409
SD = .0114
N = 40

Vth0 (V) (NMOS)

Mean = 1.0155
SD = .02276
N = 40

K1 (NMOS)

Mean = .015
SD = .017
N = 40

dl (um) = .5 * Total Δ L (NMOS)

Mean = -.1179
SD = .0059
N = 40

K2 (NMOS)

Mean = .3786
SD = .026
N = 40

dw (um) = .5 * Total Δ W   (NMOS)

Mean = 56.99
SD = 6.14
N = 40

Rds0 (ohms) (NMOS)

Figure 3

Figure 4

Mean = -.9368
SD = .0139
N = 40

Vth0 (V) (PMOS)

Mean = .4685
SD = .01447
N = 40

K1 (PMOS)

Mean = .1107
SD = .0117
N = 40

dl (um) = 0.5*Total Δ L (PMOS)

Mean = 0.02537
SD = .00658
N = 40

K2 (PMOS)

Mean = .45789
SD = .0317
N = 40

Mean = 87.35
SD = 13.8
N = 40

Rds0 (ohms) (PMOS)

Appendix

```c
#include <stdio.h>
#include <strings.h>
#include <math.h>

void funccall(FILE *output, FILE *input, double x[], double *funct,
double g[], double icon[], double econ[], double gric[][20],
double gric[][20], int *varnumber, int *innumber, int *eqnumber,
short interactive, short debug, int Groupno, int *noofgroups, short *definp)
{

FILE *fp;
double cost1, cost, dummy;
static double constraint, f1,f2,f3,f4,f5,f6,f7,f8,f9,f10,f11,f12;
static double c1,c2,c3,c4,c5,c6,c7,c8,c9,c10,c11,c12;

if ((!definp) goto define;
/*.................................................................*/
start:
/* calculate the cost function and its derivatives */

/* cost = constraint + f1*x[1] + f2*x[2] + f3*x[3] + f4*x[4] + f5*x[5] + f6*x[6] +
*/

/* cost = -constraint - f1*x[1] - f2*x[2] - f3*x[3] - f4*x[4] - f5*x[5] - f6*x[6]
*/

/*
cost1 = constraint + f1*x[1] + f2*x[2] + f3*x[3] + f4*x[4] + f5*x[5] + f6*x[6]
cost = 1/(constraint + f1*x[1] + f2*x[2] + f3*x[3] + f4*x[4] + f5*x[5] + f6*x[6]
*/

/*
g[1] = -f1/(cost1*cost1);
g[2] = -f2/(cost1*cost1);
g[3] = -f3/(cost1*cost1);
g[4] = -f4/(cost1*cost1);
g[5] = -f5/(cost1*cost1);
g[6] = -f6/(cost1*cost1);
*/

g[7] = -f7/(cost1*cost1);
g[8] = -f8/(cost1*cost1);
g[9] = -f9/(cost1*cost1);
g[10] = -f10/(cost1*cost1);
g[11] = -f11/(cost1*cost1);
g[12] = -f12/(cost1*cost1);
*/

/* Scaling the cost function */
*funct = cost;

/*.......................... Inequality constraints .....................*/

icon[1] = x[1]*x[1]/c1 + x[2]*x[2]/c2 + x[3]*x[3]/c3 + x[4]*x[4]/c4 + x[5]*x[5]/c5
icon[2] = 21.03 - x[1]*x[1]/c1 - x[2]*x[2]/c2 - x[3]*x[3]/c3 - x[4]*x[4]/c4 - x[5]

gric[1][1] = 2*x[1]/c1;
gric[1][2] = 2*x[2]/c2;
gric[1][3] = 2*x[3]/c3;
gric[1][4] = 2*x[4]/c4;
gric[1][5] = 2*x[5]/c5;
gric[1][6] = 2*x[6]/c6;
gric[1][7] = 2*x[7]/c7;
gric[1][8] = 2*x[8]/c8;
gric[1][9] = 2*x[9]/c9;
gric[1][10] = 2*x[10]/c10;
gric[1][11] = 2*x[11]/c11;
gric[1][12] = 2*x[12]/c12;

gric[2][1] = -2*x[1]/c1;
gric[2][2] = -2*x[2]/c2;
gric[2][3] = -2*x[3]/c3;
gric[2][4] = -2*x[4]/c4;
gric[2][5] = -2*x[5]/c5;
gric[2][6] = -2*x[6]/c6;
gric[2][7] = -2*x[7]/c7;
gric[2][8] = -2*x[8]/c8;
gric[2][9] = -2*x[9]/c9;
gric[2][10] = -2*x[10]/c10;
gric[2][11] = -2*x[11]/c11;
gric[2][12] = -2*x[12]/c12;

/*.....................................................................*/
goto end;

define:
/*.....................................................................*/

/* Setting the upper and lower limits of the parameters */
/* for the function */

/*........ for template 1_3 ...........*/
/*   for power optimization  */

/* constraint = 13.688229;
f[1] = 0.152963;
f[2] = -0.153118;
f[3] = 0.169631;
f[4] = 0.002184;
f[5] = -0.174360;
f[6] = -0.196719;
f[7] = -0.095136;
f[8] = -0.176160;
```

```
c6 = .758259;
c7 = .4157197;
c8 = .3098587;
c9 = .2313289;
c10 = .1579808;
c11 = .05335946;
c12 = .01126996;

/* setting the start-up values */
x[1] = -1.0;
x[2] = -1.0;
x[3] = -1.0;
x[4] = -1.0;
x[5] = -1.0;
x[6] = -1.0;
x[7] = -1.0;
x[8] = -1.0;
x[9] = -1.0;
x[10] = -1.0;
x[11] = -1.0;
x[12] = -1.0;

*varnumber = 12;
*innumber = 2;
*eqnumber = 0;
(*definp) = 1;
(*noofgroups) = 1;

goto start;
/*******************************************************/
end;

return;
}

void save(FILE *output, int groupno, double x[], double funct, int varnumber)
{
double y[4], y1, y2;
int i=1;

output = fopen("hanpalE.data", "w");
y[1] = x[1]/100.0 + 1.57;
/* y[1] = -0.030842 + 1.0254152*y1; */
y[2] = x[2]*50 + 1200;
/* y[2] = 20.822585 + 0.976473S*y2; */
y[3] = x[3];

for (i=1; i<=varnumber; i++) {
    fprintf(output," %d\t%lf\n", i, y[i]);
}
fclose(output);
}
```

```
f9 = -0.013922;
f10 = -0.092715;
f11 = -0.039833;
f12 = 0.124771;
;

/* for delay optimization */
*/

/* constraint = 175.79501;
f1 = -2.178302;
f2 = 0.813690;
f3 = -1.035651;
f4 = 0.670933;
f5 = 1.436104;
f6 = 1.983524;
f7 = 1.627088;
f8 = 0.810502;
f9 = 2.187712;
f10 = -0.478262;
f11 = -1.747272;
f12 = 9.833871;
*/
/***** for template 0_4 ************/

/* for power optimization */
/* constraint = 13.169852;
f1 = 0.115934;
f2 = -0.156639;
f3 = 0.161607;
f4 = -0.009022;
f5 = -0.064623;
f6 = -0.216347;
f7 = -0.087720;
f8 = -0.070496;
f9 = -0.119468;
f10 = -0.020215;
f11 = -0.071489;
f12 = -0.052022;
*/

/* for delay optimization */
constraint = 181.801018;
f1 = -2.051040;
f2 = 0.954933;
f3 = -0.864776;
f4 = 0.547431;
f5 = 1.024806;
f6 = 1.822507;
f7 = 1.566505;
f8 = 1.060189;
f9 = 2.455392;
f10 = -0.956480;
f11 = -1.409511;
f12 = 9.258759;
*/
/***********************************************/

/* for the constraint */
c1 = 4.173090;
c2 = 2.006184;
c3 = 1.603642;
c4 = 1.496005;
c5 = .7832995;
```

A Methodology for Process Parameter Extraction from Ring Oscillator Period Measurements

Mark Hatzilambrou

In order to ease the extraction of key circuit performance related process parameters, a suite of ring oscillator test circuits whose performance is easily measurable is modeled through simulations. These models are then used to extract the values of the parameters from results generated via simulation at arbitrary points in the parameter space.

1.0 Introduction

It is widely known that speed performance variations in digital MOS logic circuitry is attributable to variations in several key device metrics, i.e. transistor effective channel length (Leff), transistor effective channel width (Weff), device threshold voltages (Vt), and gate oxide thickness (Tox). It has been shown that varying the component parameters of these few metrics in simulation can adequately describe speed performance variation across the range of process variation.

Monitoring the variation in these device parameters directly is very expensive in terms of time and resources; thus, acquisition of this data for each wafer processed is prohibitive. Electrical measurements, however, are much less expensive than the standard SEM and more destructive measurement techniques used to measure the key components parameters. An electrical means of measuring the effects and extracting the values of these parameters would be helpful for SPC, binning, and statistics acquisition for circuit design.

This study explores the possibility of extracting parameter values from ring oscillator(RO) period measurements. In order to extract parameter values from electrical measurements which confound effects of these parameters, an overdetermined system is required. Therefore a suite of circuits is modeled in order to extract parameter values through least squares fitting. An analysis of the models is conducted to discover deficiencies in the methodology.

2.0 Methodology

Eight ring oscillators designed so that they have different sensitivity to seven key parameters are simulated across the range of parameter component variation. Simple additive models are built to relate RO periods to the seven process components. These models are then used to estimate parameter component values through a least squares fit to results from simulations, which are conducted at arbitrary points in the parameter space. The validity of the overall methodology can be determined by the accuracy with which it extracts these parameter values.

The methodology can be divided into four basic steps: selection of process, model, and parameter components; selection of test circuits; modelling circuit performance in terms of process parameters; and parameter extraction from the suite of circuit models.

2.1 Process and Parameters

The hSpice circuit simulator presents a range of choices of device models, of which MOS Level3 was selected. Although hSpice Level3 model is an empirical model which relies upon parameter fitting in order to achieve circuit accuracy, good results have been achieved in simulating process variation using this model by varying only the key device parameters[1].

We used Leff component parameters XL (poly gate length variation) and LD (diffusion of source and drain under gate poly, separate for both NMOS and PMOS); Weff parameter XW (transistor width variation); Tox parameter TOX (gate oxide thickness); and Vt parameter DELVT0 (zero-bias gate threshold variation, separate for NMOS and PMOS) to model variations in Leff, Weff, Tox, and Vt for both NMOS and PMOS devices. Although some of these parameters are in reality highly correlated (e.g. Tox and Vt) so that independent variation of these parameters can result in combinations not seen in production, there is no loss of generality in allowing them to vary independently in this simulated experiment. The MOSIS 1μ process models are used, although source and drain static overlap capacitances have been removed and are calculated dynamically[2].

There are a total of seven parameters, which are listed in Table 1, covering component variation for both NMOS and PMOS devices. The range of variation in the MOSIS process was assumed and is also listed in Table 1 along with the full device models.

Table 1: Parameter Variations and MOSIS 1μ hSpice Level3 MOS Models

Parameter Variation							
	Diffusion Under Gate		Gate Poly	Gate Ox	Threshold Voltage		Width
	(PLd)	(NLd)	(XL)	(Tox)	(PVt)	(NVt)	(XW)
.param	PDELLD	NDELLD	DELXL	DELTOX	PDELVT	NDELVT	DELXW
variation	+/-0.1μ	+/0.1μ	+/-0.1μ	+/-20A	+/-0.2V	+/-0.2V	+/-0.3μ

NMOS Model:
.MODEL NMOS NMOS LEVEL=3 PHI=0.600000 XJ=0.150000U TPG=1 VTO=0.7894 DELTA=5.3480E-01 KP=1.2023E-04 UO=588.4 THETA=1.2810E-01 RSH=2.2190E+00 GAMMA=0.5931 NSUB=4.4240E+16 NFS=2.9700E+12 VMAX=1.7610E+05 ETA=4.1800E-02 KAPPA=1.4750E-01 CGBO=3.4802E-10 CJ=1.4276E-04 MJ=0.7052 CJSW=4.5285E-10 MJSW=0.351440 PB=0.800000 TOX=MODELTOX DELVTO=NDELVT LD=NMODLD XL=DELXL XW=MODELXW

PMOS Model:
.MODEL PMOS PMOS LEVEL=3 PHI=0.600000 XJ=0.150000U TPG=-1 VTO=-0.8682 DELTA=5.9780E-01 KP=3.5553E-05 UO=174.0 THETA=1.9370E-01 RSH=1.8010E+00 GAMMA=0.4694 NSUB=2.7710E+16 NFS=4.8760E+12 VMAX=8.6470E+05 ETA=1.1730E-01 KAPPA=9.9050E+0 CGBO=3.7779E-10 CJ=5.5753E-04 MJ=0.4566 CJSW=7.1088E-11 MJSW=0.013373 PB=0.850000 TOX=MODELTOX DELVTO=PDELVT LD=PMODLD XL=DELXL XW=MODELXW

.param MODELTOX='1.69E-08 + DELTOX'	.param PMODLD='1.123E-07 + PDELLD'
.param MODELXW='-0.3624E-6 + DELXW'	.param NMODLD='1.463E-07 + NDELLD'

2.2 Ring Oscillator Circuits

In order to provide a set of models that form an overdetermined system in terms of the seven parameters, eight ring oscillators were designed to be variously sensitive to these parameters. Threshold voltage sensitivity was emphasized by stacking gates in series, and gate oxide sensitivity was changed by increasing fanout due to gate loading on each stage.

Table 2 summarizes the structure of the test circuits. Single stack indicates simple inverters, NAND-NOR cascade is made up of standard NAND and NOR gates, and Triple Stack Inverters indicates a series of 3 devices in each the NMOS and PMOS legs of the inverter. Fanout indicates the loading on the output of a gate in terms of its own gate loading. A full spice deck for model 1 is included in the appendix.

Table 2: Ring Oscillator Test Circuits

Model	1	2	3	4	5	6	7	8
Gate Structure	Single Stack Inverter	Single Stack Inverter	Single Stack Inverter	NAND-NOR Cascade	NAND-NOR Cascade	NAND-NOR Cascade	Triple Stack Inverter	Triple Stack Inverter
Fanout	1	4	7	1	4	7	1	2

2.3 Model Development

A simple and numerically efficient model is desired relating ring oscillator period to the values of the key parameters for each of the eight circuits. A full factorial experiment was run in simulation by varying the parameters by the extremes of the process listed in Table 1. From the results of these simulations, the Yates algorithm can be used to determine the factor effects, and the resulting model can be verified by comparing residual sum of squares to model sum of squares.

It should be noted that the models are constructed in terms of the deviations of parameter values from nominal values; therefore each model consisting only of main effects for the seven parameters would contain eight terms, one for each factor effect, and one grand average. A model could also be derived from the factorial data using a least squares fit; alternatively, simulations could have been run for points throughout the parameter space rather than just at the extremes of the factorial, from which curvature could be more directly modelled. However, inadequacy of the models developed from the factorial can be deduced from confirmation runs.

2.4 Parameter Extraction

Once models have been developed for all eight circuits, given the measured periods of the eight circuits fabricated, or in this case simulated, at a point in the parameter space, parameter values can be extracted using a least squares fit[3]

$$b = [X^T X]^{-1} X^T y \qquad (1)$$

where b is the resultant vector of parameter values, X is the matrix of main effects of these parameters, and y is the difference between the measured periods and the grand averages. In order for the fit procedure to be well conditioned, the columns of X should be as linearly independent as possible.

3.0 Model Results

The main factor effects of the 8x128 simulation runs are displayed in Table 3. Note that all factor effects are calculated in terms of the normalized parameter range +/-1. These main factor effects are all more significant than any of the interaction terms, with only a few interaction terms being of the same order of magnitude as the smallest main effect. This implies that for the circuits selected, the parameters have independent effects upon the ring oscillator period.

Testing the adequacy of the model is somewhat complicated by the fact that our experiment is deterministic involving no measurement error and no error due to variation in other process parameters. Some of this uncertainty may be included by fitting only a four-term model involving only the grand average and the three most significant main effects. By treating the variations in the resulting parameters as replicated runs, an estimated variance may be arrived at and goodness of fit deduced for the four term model. Note that the extremes of variation will tend to overestimate the actual variation, so care must be taken in drawing conclusions. An F-test using the resulting mean-square ratios indicates that the 4-term model is highly significant. The R^2 value of ~0.85 for the four term model indicates a reasonably complete fit given the noise of the system. Residual plots indicate that the residuals are randomly distributed in their deviation from zero, which we might expect with a good fit to a deterministic experiment.

The 8-term model displays an R^2 value of 0.98, which indicates a fairly complete fit. However, it is difficult to draw definite conclusions given the complete lack of any noise information whatsoever. Noise estimates could be incorporated by running replicated experiments with Monte Carlo selection of further key parameters, but a more physical device model than the one we have used here would probably be necessary to yield adequate results. There is no possibility that our models for our circuits are overfit, considering the deterministic nature of the experiment.

Table 3: Factor Effects from Full Factorial Runs

Model	Model Coefficients								Adjusted R^2	
	Avg	PLd	NLd	XL	Tox	PVt	NVt	XW	8term	4term
1	9.509	-1.06	-1.10	2.00	-0.71	-0.24	0.32	-0.26	.98	.86
2	24.41	-3.29	-2.92	5.37	-1.92	-0.62	0.83	-0.64	.98	.87
3	39.24	-5.46	-4.70	8.71	-3.16	-0.94	1.33	-.101	.98	.87
4	11.14	-1.02	-1.21	2.42	-0.92	-0.23	0.37	-0.24	.98	.85
5	31.05	-3.87	-3.66	7.23	-2.87	-0.65	1.05	-0.65	.98	.85
6	50.50	-6.84	-6.07	11.99	-4.85	-1.07	1.78	-1.07	.98	.85
7	21.19	-2.19	-2.31	4.60	-1.96	-0.41	0.75	-0.27	.98	.83
8	28.65	-3.09	-3.11	6.33	-2.71	-0.65	0.99	-0.32	.98	.83

4.0 Confirmation Run Results and Discussion

Using the methodology described in section 2.4, parameter values were extracted and compared with the known input values. Confirmation runs were performed at arbitrary points in the parameter space (not at points examined during the factorial experiment) in order to test the validity of the models throughout the parameter space. As a less rigorous test of the model and methodology, an attempt was made to extract points that were part of the initial factorial experiment. The results of these least-squares extractions are shown in Table 4, with the first four pairs of rows listing data points internal to the parameter space and the final four pairs of rows listing "corner" points in the factorial set. Note that all values are in terms of normalized parameter variation.

The results show a remarkable inadequacy in either the models or the methodology. We can see that although an occasional extracted parameter is close to its actual value (such as the case with PLd and PVt in the first example), it is very often astronomically incorrect. Calculation of predictions and residuals using the extracted values of the parameters indicates that the least-squares fit has indeed done a good job. In fact, the high error in values is more indicative of a system that is overfit with the high values attributable to insignificant terms.

If our models were perfect, we would expect that extraction of parameters for points on the factorial would yield exact results. The fact that our extraction for such points does not yield results which are close to, or even in the direction of, our actual values, indicates that perhaps the incompleteness of the models is not the culprit.

Fits using the 4-term model, and fits to the three most significant parameters using the 8-term models with known values of the remaining parameters yielded similarly unsatisfactory results.

Table 4: Actual Versus Extracted Normalized Parameter Variation

	PLd	NLd	XL	Tox	PVt	NVt	XW
Actual	0	0	0	0	0	0	0
Extracted	-0.065	1.31	1.065	0.29	-0.027	-2.77	-2.44
Actual	0.99	-0.12	0.40	-0.42	-0.84	-0.41	0.94
Extracted	0.55	4.43	1.96	2.35	4.00	12.14	-1.38
Actual	-0.94	0.85	0.65	0.70	-0.19	0.08	-0.55
Extracted	-0.608	-1.416	-0.66	0.13	0.036	0.182	-0.748
Actual	-0.66	-0.16	0.15	-0.39	0.96	0.66	0.73
Extracted	-8.43	24.80	14.39	12.51	-12.43	-8.14	24.99
Actual	+1	+1	-1	+1	+1	-1	-1
Extracted	0.512	3.733	-0.997	-2.29	0.064	-1.216	-3.32
Actual	+1	+1	-1	+1	+1	+1	-1
Extracted	0.814	1.539	-1.968	-2.94	-0.447	-2.52	-2.912

Table 4: Actual Versus Extracted Normalized Parameter Variation

Actual	+1	-1	-1	-1	-1	-1	+1
Extracted	1.275	-3.96	-0.837	1.787	2.632	-0.795	4.313
Actual	-1	+1	+1	+1	-1	+1	+1
Extracted	-1.098	2.184	1.106	1.472	-1.206	4.731	1.398

Equation 1 leads us to examine our coefficient matrix X. It was noted above that in order for the fit to be numerically insensitive to the matrix inversion operation, the columns of X must be as close to orthogonal as possible. The columns of X represent the parameter coefficient columns of our models, and colinearity of these columns is a measure of the covariance of parameters with respect to model outputs. A measure of the colinearity can be calculated by taking the inner product of the normalized columns with one another. This colinearity is listed in Table 5 as the angular difference between the direction of the parameter vector in parameter 7-space. We see that these vectors are nearly colinear, with the greatest deviation being 16.3 degrees, and typical deviations around five degrees. By analyzing the rows of X, we find a similar relationship between circuit models, indicating that the circuits themselves do not differ very much in their sensitivity to the various parameters. The colinearity in rows and columns of our coefficient matrix magnifies any imperfections in modelling our deterministic system, yielding highly inaccurate results.

Table 5: Angular Deviation Between Parameter Columns (Degrees)

	PLd	NLd	XL	Tox	PVt	NVt	XW
PLd	0	3.7	174.9	8.0	5.6	174.2	5.3
NLd		0	178.0	5.3	4.2	177.6	11.3
XL			0	176.6	174.5	1.8	167.0
Tox				0	8.3	176.2	16.3
PVt					0	175.1	11.3
NVt						0	166.5
XW							0

5.0 Conclusions

It has been suggested that deducing process variation and process performance from the frequency of test ring oscillators is difficult because ring oscillators confound all parameters. However, this is not necessarily the major problem. Confounded effects may be extracted given a system that adequately spans the parameter space. Designing ring oscillators so that the desired parameters are represented in adequately orthogonal models appears to be a key.

[1] "A Method For Modeling the Manufacturability of IC Designs", Boskin et al, Electronics and Research Laboratory, Department of Electrical Engineering and Computer Sciences, University of California at Berkeley.

[2] HSpice User's Manual, Volume2:Elements and Models, (C) 1992 Meta-Software, Inc., Chapters 6 and 7.

[3] Box, G.E. Hunter, W.G. Hunter, J.S. Statistics for Experimenters, (C) 1978 by John Wiley & Sons, Inc. New York, New York, p. 501.

6.0 Appendix

The following is the hspice input deck for model1. Parametric definitions are found at the end of the file.

```
*************************************************************************
.SUBCKT inv VDD GND in out
M1 VDD in out VDD PMOS W=15.0U L=1U AD=60.0P PD=38U AS=60.0P PS=38U
M2 GND in out GND NMOS W=5.0U L=1U AD=20.0P PD=18.0U AS=20P PS=18.0U
C1 in out 1.0F M=FANOUT
C2 out GND 19.0F M=FANOUT
*** Fan-out Loading Transistors
* Mpl1 VDD out nl1 VDD PMOS W=15.0U L=1U AD=60.0P PD=38U AS=60.0P PS=38U
* + M='FANOUT-1'
* Mnl1 GND out nl1 GND NMOS W=5.0U L=1U AD=20.0P PD=18.0U AS=20P PS=18.0U
* + M='FANOUT-1'
* Cl1 nl1 GND 20PF
* + M='FANOUT-1'
*** Node Listing for subckt:  inv
** GND Node 0 is the global ground node
** GND GND!
** in input
** out output
** VDD Vdd!
.ENDS

****** top level cell
xinv31 VDD 0 31 1 inv
xinv30 VDD 0 30 31 inv
xinv29 VDD 0 29 30 inv
xinv28 VDD 0 28 29 inv
xinv27 VDD 0 27 28 inv
xinv26 VDD 0 26 27 inv
xinv25 VDD 0 25 26 inv
xinv24 VDD 0 24 25 inv
xinv23 VDD 0 23 24 inv
xinv22 VDD 0 22 23 inv
xinv21 VDD 0 21 22 inv
xinv20 VDD 0 20 21 inv
xinv19 VDD 0 19 20 inv
xinv18 VDD 0 18 19 inv
xinv17 VDD 0 17 18 inv
xinv16 VDD 0 16 17 inv
xinv15 VDD 0 15 16 inv
xinv14 VDD 0 14 15 inv
xinv13 VDD 0 13 14 inv
xinv12 VDD 0 12 13 inv
xinv11 VDD 0 11 12 inv
xinv10 VDD 0 10 11 inv
xinv9 VDD 0 9 10 inv
xinv8 VDD 0 8 9 inv
xinv7 VDD 0 7 8 inv
xinv6 VDD 0 6 7 inv
xinv5 VDD 0 5 6 inv
xinv4 VDD 0 4 5 inv
xinv3 VDD 0 3 4 inv
xinv2 VDD 0 2 3 inv
xinv1 VDD 0 1 2 inv
```

```
***** models for nmos and pmos mosfets *****************************
.MODEL NMOS NMOS LEVEL=3 PHI=0.600000 XJ=0.150000U TPG=1
+ VTO=0.7894 DELTA=5.3480E-01 KP=1.2023E-04
+ UO=588.4 THETA=1.2810E-01 RSH=2.2190E+00 GAMMA=0.5931
+ NSUB=4.4240E+16 NFS=2.9700E+12 VMAX=1.7610E+05 ETA=4.1800E-02
+ KAPPA=1.4750E-01
+ CGBO=3.4802E-10 CJ=1.4276E-04 MJ=0.7052 CJSW=4.5285E-10
+ MJSW=0.351440 PB=0.800000
+ TOX=MODELTOX
+ DELVTO=NDELVT
+ LD=NMODLD
+ XL=DELXL
+ XW=MODELXW

.MODEL PMOS PMOS LEVEL=3 PHI=0.600000 XJ=0.150000U TPG=-1
+ VTO=-0.8682 DELTA=5.9780E-01 KP=3.5553E-05
+ UO=174.0 THETA=1.9370E-01 RSH=1.8010E+00 GAMMA=0.4694
+ NSUB=2.7710E+16 NFS=4.8760E+12 VMAX=8.6470E+05 ETA=1.1730E-01
+ KAPPA=9.9050E+00
+ CGBO=3.7779E-10 CJ=5.5753E-04 MJ=0.4566 CJSW=7.1088E-11
+ MJSW=0.013373 PB=0.850000
+ TOX=MODELTOX
+ DELVTO=PDELVT
+ LD=PMODLD
+ XL=DELXL
+ XW=MODELXW

***** Independent sources ********
Vdd VDD 0 dc=vccv

***** Analysis ********
.param vccv=5V
.dcvolt v(1)=0V
.op all 35ns
.tran .1ns 20ns
.MEASURE TRAN first TRIG v(15) VAL=('.5*vccv') RISE=1
+           TARG v(15) VAL=('.5*vccv') RISE=2

***** Fanout Parameter ********
.param FANOUT=1
***** Parameters for model changes ********
.param DELXW=-0.30E-6
.param NDELVT=-0.2V
.param PDELVT=-0.2V
.param DELTOX=-0.20E-08
.param DELXL=-0.10E-6
.param NDELLD=-0.10E-6
.param PDELLD=-0.10E-6

.param MODELTOX='1.69E-08 + DELTOX'
.param PMODLD='1.123E-07 + PDELLD'
.param NMODLD='1.463E-07 + NDELLD'
.param MODELXW='-0.3624E-6 + DELXW'
.END
```