TOWARDS A COMPLETE PLASMA DIAGNOSTIC SYSTEM

by

Dong Wu Zhao

Memorandum No. UCB/ERL M02/14

1 May 2002

TOWARDS A COMPLETE PLASMA DIAGNOSTIC SYSTEM

by

Dong Wu Zhao

Memorandum No. UCB/ERL M02/14

1 May 2002

TOWARDS A COMPLETE PLASMA DIAGNOSTIC SYSTEM

by

Dong Wu Zhao

Memorandum No. UCB/ERL M02/14

1 May 2002

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

TOWARDS A COMPLETE PLASMA DIAGNOSTIC SYSTEM

by

Dong Wu Zhao

B.S. (University of California, Berkeley) 1996
M.S. (University of California, Berkeley) 1999

A dissertation submitted in partial satisfaction of
the requirements for the degree of
Doctor of Philosophy

in

Engineering-Electrical Engineering & Computer Sciences

in the

GRADUATE DIVISION
Of the
UNIVERSITY OF CALIFORNIA, BERKELEY

Committee in charge:

Professor Costas J. Spanos, chair
Professor Kameshwar Poolla
Professor Laurent El Ghaoui

Spring 2002

The dissertation of Dong Wu Zhao is approved:

C.S. SPANOS 4/17/02
Chair Date

4/25/02
Date

L. E. Ghaoui 4/30/02
Date

University of California, Berkeley

Spring 2002

Towards a Complete Plasma Diagnostic System

By
Dong Wu Zhao

Abstract

TOWARDS A COMPLETE
PLASMA DIAGNOSTIC SYSTEM

by Dong Wu Zhao

Doctor of Philosophy in Engineering-Electrical Engineering & Computer Sciences

University of California at Berkeley

Professor Costas J. Spanos, Chair

We have set up a plasma diagnostic system with three sources of signals, OES, RF, and machine signals. CF_2 OES lines 275 nm and 321 nm are found to be better than any other signals for poly-etch endpoint detection. In addition, excellent statistical models for wafer state prediction are obtained by linear stepwise regression on all available signals. A data exploration system, based on syntactical analysis, is developed for efficiently browsing of the data archive, allowing users unprecedented flexibility in examining the data both qualitatively and quantitatively. Two case studies of syntactic analysis for diagnostics are presented. Finally, the use of low frequency signals for plasma diagnostics in investigated. The syntactic method for analyzing the signals is proposed.

Professor Costas J. Spanos
Dissertation Committee Chair

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I would like to thank my advisor, Professor Costas Spanos, for his careful guidance and generous support for my research.

I acknowledge Qi Li and Jerry Jin of Applied Materials for their insights and expertise in plasma etching. I benefited from discussions with Yuri Trachuk and Georges Gorin of Adavanced Energy.

A "thanks" goes out to present and past members of the Berkeley Computer-Aided Manufacturing (BCAM) group for their friendships: Junwei Bao, Jason Cain, Runzi Chang, Roawen Chen, Weng Foong, Mark Hatzilambrou, Anna Ison, Herb Huang, Nickhil Jakatdar, Michiel Kruger, Jae-Wook Lee, Jeff Lin, Greg Luurtsema, David Mudie, John Musacchio, Xinhui Niu, Manolis Terrovitis, Nikhil Vaidya, Jiangxin Wang, and Haolin Zhang.

This work was funded by the State of California SMART program under research contract SM97-01, and by the following participating companies: Advanced Energy, ASML, Atmel Corp., Advanced Micro Devices, Applied Materials, Asyst Technologies Inc., BOC Edwards, Cymer, Etec Systems Inc., Intel Corporation, KLA-TENCOR, Lam Research Corp., Nanometrics, Inc, Nikon Research Corp., Novellus Systems Inc., Silicon Valley Group. Most of the experimental work took place in the Berkeley Microfabrication Laboratory.

Chapter 1

INTRODUCTION

1.1 Motivation

As the semiconductor processing technology approaches the 0.1 μm feature size and 300 mm wafer diameter, the cost of building a new fabrication plant is rising rapidly (20% per year) [26]. It is predicted that it will take about $10 billion to build a state-of-the-art facility for manufacture in 2005. To remain competitive and manage the escalating cost, the industry has strived to improve feature size, wafer diameter, yield, and equipment utilization. However, the gains from wafer diameter and yield are reaching their practical limits and the new focus is on equipment utilization.

The key to optimize equipment utilization is through process monitoring in order to make sure that wafers are processed properly at each step. However, there are more than 100 manufacturing steps, and it is too costly and time-consuming to measure each wafer after the completion of each step. As of now, people in the industry usually measure and monitor wafers periodically, especially right after performing preventive maintenance and changing machine settings. A final test is performed on each wafer after all the steps. Thus, if an error occurs, it is very likely that many wafers are misprocessed without notice until very late. Because of the late notice, it is very difficult to trace back and locate the faulty step and diagnose the problem. Therefore, one can save considerable resources by monitoring equipments on line, using their real-time signals. In this work we demonstrate that it is possible to do so, with the monitoring of plasma etch signals as an example.

Plasma etching is one of the costliest steps during semiconductor processing. In addition, it is very difficult to control, since the physical mechanism of plasma etching is not well understood. This thesis explores various issues of plasma etch process monitoring, including fault detection and diagnosis, endpoint detection, wafer state prediction.

A fault occurs when there is a sudden change in etching behavior, manifesting through a sudden shift in signal behavior. It can happen due to operator errors, such as no photoresist, undeveloped wafer, and wrong material, or machine errors, such as gas leak, power fault, and pressure fault. Fault detection tools determine the state of the plasma etcher by analyzing the behavior of its real-time signals. Once a fault is detected, the fault diagnosis tools will assign a cause to it, as to assist the process engineer to fix the problem. By detecting the fault early, a process engineer can prevent expensive new wafers from being fetched to the faulty etcher, and correct the fault on a timely basis. Thus, wafer yield and throughput will be enhanced. Also, preventive maintenance (PM) can be scheduled according to fault detection and diagnosis results, and down-time and mean-time-to-repair (MTTR) can be reduced.[15].

An endpoint is reached when the target thin film is etched through. Accurate end pointing has a great impact on controlling the critical feature size. Precise wafer state prediction, for parameters such as uniformity, etch rate, selectivity, and anisotropy, can reduce the need for costly and time-consuming wafer measurement.

A plasma etcher generates a large number of signals suitable for diagnostic purposes. This work examines the combination of heterogeneous signals, in order to extract useful information. Four different sources of diagnostic signals on a plasma etcher are collected and analyzed, including optical emission spectroscopy (OES), RF power information on the fundamental and several harmonics, and the machine signals such as power, chamber pressure, temperature, gas flow rate, etc., and low frequency signals, ranging anywhere from 10 Hz to 10 kHz. Various researchers have investigated the first three kinds of diagnostic signals, for examples, White, et al. [26, 27] and R. Chen [24] on OES, Roth, et al. [45] on RF, Spanos and S. Lee [14, 15] on machine signals. This thesis puts emphasis on exploring and analyzing these three sources of signals. An automated data collection system is set up on a LAM Rainbow 9400 Etcher. An OES sensor and an RF sensor are installed, and the machine signals are collected through the machine built-in SECS II interface. A data exploration system based on syntactic analysis is developed for examining the signals both quantitatively and qualitatively. An endpoint detection sensitivity test is performed on these signals. Some wafer state models are built from them with a few different techniques. Also, a couple of case studies of syntactic analysis in fault detection and diagnosis on the plasma machine signals are presented.

Plasma operation is often associated with low frequency electrical signals. Even though these signals may carry useful diagnostic information, they are not well understood, and have not yet been utilized for this purpose. Recently, Lieberman, et al. observed that some low frequency signals are related to the instability of the plasma, and proposed a physical model for the instability phenomenon [47]. Although some analytical techniques are proposed for analyzing the low frequency signals, the work would not be complete without integration with the physical model. As of now, Lieberman, etc. are still working the instability modeling.

1.2 Thesis Organization

Chapter 2 describes some traits of plasma etch signals which make them difficult to analyze, and then a discussion on previous works by various researchers is presented. Chapter 3 details the hardware setup on the LAM Rainbow 9400 Etcher in the Berkeley Microfabrication Laboratory. Chapter 4 discusses the basic features of the data exploration software and present some case studies. Chapter 5 is about choosing the optimal endpoint detection signals from the available signals on the LAM etcher. Some comments on other endpoint detections works are also presented. Chapter 6 addresses the wafer state modeling results, comparing a few modeling techniques. Chapter 7 talks about the data exploration software's advanced features based on syntactic analysis. The details of syntactic analysis will be discussed. Chapter 8 presents some case studies of fault detection and diagnosis with syntactic analysis on plasma machines signals. Chapter 9 explores the potentials of low frequency analysis. Some analytical techniques are proposed. Chapter 10 concludes the thesis, with some thoughts on the future works.

Chapter 2

BACKGROUND & PREVIOUS WORKS

2.1 Real-time Data of Plasma Etch & Analytical Difficulties

Plasma etching is not a very well understood process. Practical physical models for fault detection and diagnosis are not yet available. Researchers in fault detection and diagnosis so far have used empirical models. Previous works [15] involve modeling of input setting against wafer's output parameters, such as etch rate, uniformity, selectivity, and anisotropy. However, due to machine aging, maintenance and various other effects, input settings do not entirely determine the chamber state. The same settings can result in very different etching behavior. Spanos and S. Lee [14][15] show that the equipment's own electrical and mechanical signals can be modeled as time series and used effectively for fault detection and diagnosis. These real-time signals reflect the chamber state much better than the input settings; they are able to show drift in etching behavior due to machine aging and maintenance.

Usually, when the equipment is just out of control, the malfunction will first manifest itself in the real-time signals, but not much etching damage is done to the wafer yet, and the wafer is still usable if the malfunction is corrected soon enough. As a result, using real-time signals for monitoring the etching process can help prevent misprocessing costly wafers. Hundreds of real-time signals are available for computer analysis via standard communication ports, such as SECS II. An engineer can choose a few of them to monitor the etching process based on experience. Alternatively, one can find out the signals that are sensitive to faults by doing designed experiments. Some of the real-time signals proved useful are RF load, coil position, RF tune vane position, peak-to-peak voltage load impedance, RF phase error, DC bias and endpoint [16]. In this thesis, we will examine signals from other sources as well, such as OES and RF.

Statistical modeling of etching signals has been difficult, due to preventive maintenance (PM), machine aging, chamber memory effects, and other influences [15]. During a maintenance cycle, residue gradually builds up in the chamber. This causes the chamber state and sensor signals to drift slowly.

Notice that a drift in sensor signals does not necessarily correspond to a drift in the chamber state. For instance, as residue accumulates on the sensor window and degrades the transmittance, the intensity of the sensor signals will decrease. Yet, the operation of the equipment is far from being faulty.

If too much residue accumulates in the chamber, the process parameters will be quite different from when the chamber is clean. The aim of PM is to restore the etcher back to the original clean state. However, due to the aging of other parts of the tool, process parameters after a PM will be a little different from those at the beginning of the previous clean cycle; the average level of the signals as well as the variance may change.

One also encounters so-called "memory effects", where, for example, after a signal is unusually high indicating a fault, it will often remain relatively high for a while before returning to the normal level, even after the machine is back in control. This memory effect is very obvious when the machine first starts up. It takes a few runs before the machine reaches its steady operating state, while the signals appear to approach steady state values in an exponential fashion. (See Figure 2.1 and 2.2.)

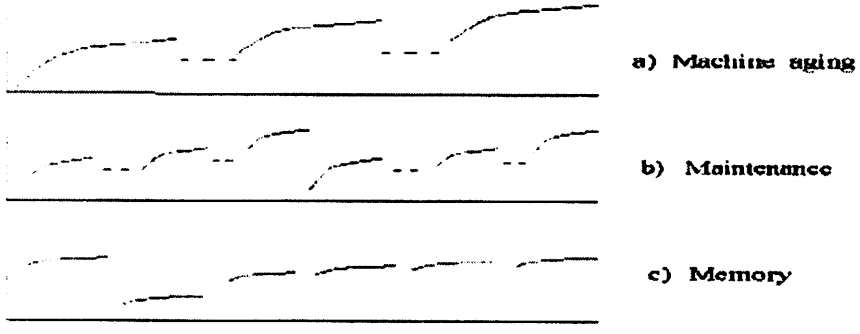


Figure 2.1. Ilustrating the nature of the plasma etching signals. a) Machine aging effect within a PM maintenance cycle. b) Maintenance effect after a PM maintenance procedure. c) Memory effect after a fault occurs.

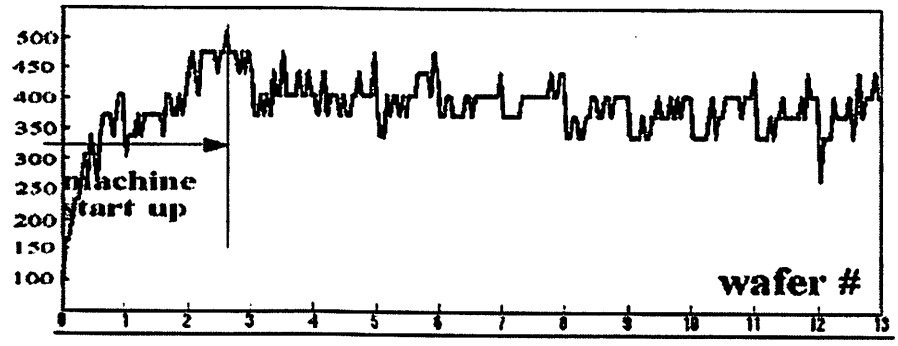


Figure 2.2. The memory effect as seen when the machine starts up.

The above are some of the observations that the author makes on some data obtained from the industry. The thesis investigates signals from heterogeneous sources, and presents syntactic analysis, a novel technique for analyzing the signals,. Some prior works applied analytical techniques on limited observations with mixed success. Before presenting the new technique, it is helpful to go over some of the previous works (hardware setups and analytical techniques), so that we can have a baseline for evaluating the solution that is being presented.

2.2 Real-Time Machine Signals

S. Lee investigates the real-time machine signals with a parallel plate etching system (Lam Rainbow 4400) and a TCP etching system (Lam Rainbow 9400). A parallel plate system's RF source consists of upper and lower electrodes, which produces capacitive plasma discharges. The TCP RF source consists of a planar coils wound from the center to the outer radius of the chamber. The plasma is created when the gas near the coil ionizes due to the induced electric field. The RF sources of both systems are 13.56 MHz AC power generators. TCP systems run at lower pressures and create higher plasma density than parallel plate systems, so they usually achieve better anisotropy, smaller critical dimensions, and faster etcher rates. Two data collection systems are being used, the Brookside LamStation software and the Comdel Real Power Monitor (RPM-1). The LamStation software, which gets data via the SECS II (SEMI Equipment Communication Standard – II) interface, is installed on both the parallel plate etching system and the TCP etching system. The Comdel RPM-1 RF probe, which collects data via its own RS232 interface, is only installed on the parallel plate etching system.

Each etcher has more than 400 signals to choose from for diagnostic analysis. However, many signals do not directly impact the etching chamber. For example, sensor signals for transporting wafers. Also, even under the same recipe, etching behavior of the system changes over time as mentioned in previous paragraphs, so machine settings are not monitored. An F-test is deployed to assess the relevance of the rest of the real-time signals. A factorial experiment is conducted over a certain range on the input settings. Also, the data is collected for a few wafers processed under normal machine setting, or baseline condition.

Then the F-test can be computed as,

$$\frac{s_{fact}^2 / \nu_{fact}}{s_{bas}^2 / \nu_{bas}} \propto F_{\alpha, \nu_{fact}, \nu_{bas}} \qquad (2.1)$$

where s_{fact}^2 is the estimated variance of a signal during the factorial run, ν_{fact} is the degrees of freedom in the factorial experiment; s_{bas}^2 is the estimated variance of the baseline run, ν_{bas} is the degrees of freedom for the baseline condition. The signals with F-statistics above a certain level of significance are used for monitoring the system. Table 2.1 and 2.2 list and describe the signals selected for the parallel plate system, and Table 2.3 lists the signals selected for the TCP system.

LamStation Software	Comdel RPM-1
RF Load Coil Position	RF Power
RF Tune Vane Position	RF Voltage
Peak-to-Peak Voltage	RF Current
Load Impedance	Load Impedance
RF Phase Error	RF Phase Error

DC Bias	DC Bias
Endpoint	

Table 2.1. Real-Time State Signals Collected for the Lam Rainbow 4400.

Signal	Description
RF Tune Vane Position	Position of the tune vane in the matching network of the upper electrode; acts as a variable capacitor
RF Load Coil Position	Position of the load coil position in the matching network of the upper electrode; acts as a variable inductor
RF Load Impedance	Apparent input impedance of the matching network
RF Phase Error	The phase error between the current and voltage (ideally 90 o) at the upper electrode
DC Bias	Measures the potential difference of the electrodes
Peak-to-Peak Voltage	Magnitude of voltage on the electrodes
End Point Data	Reads the intensity of the plasma in the chamber at a particular wavelength
RF Voltage	Root-mean-square (RMS) voltage at the upper electrode
RF Current	RMS current at the upper electrode

Table 2.2. Description of the Real-Time Signals.

Source	Signal	Description
Bottom RF	RF Tune Vane Position	Equivalent position of the tune vane position in matching network of the lower coil
	RF Load Coil Position	Equivalent position of the load coil position in matching network of the lower coil
	Line Impedance	Apparent input impedance of the lower matching network
	RF Phase Error	Phase error between the current and voltage at the bottom coil
	DC Bias	Measures the charge on the electrodes
Top TCP	TCP Tune Vane Capacitor Position	Position of the tune vane capacitor of the matching network for the top coil
	TCP Phase Error	Phase error between the current and voltage at the top coil
	TCP Load Capacitor Position	Position of the load capacitor of the matching network for the top coil
	Line Impedance	Apparent input impedance of the upper matching network
Others	RF Bias	DC bias when both sources are powered
	Endpoint	Reads the intensity of the plasma in the chamber at a particular wavelength

Table 2.3. Real-Time State Signals Collected for the Lam TCP 9600.

Time series models are used to capture the dynamics of the selected real-time signals. First, models are trained to learn the in-control autocorrelation structure from the baseline data. If significant deviation is detected from the baseline model, an alarm is generated.

The time series models used are ARIMA(p,d,q) models, where p is the auto-regressive order, d is the integration order, and q is the moving average order. The ARIMA models for a non-stationary time series X_t can be expressed by the following two equations,

$$\omega_t = -\sum_{k=1}^{p} \phi_k \omega_{t-k} + \sum_{k=1}^{p} \theta_k a_{t-k} \tag{2.2}$$

$$\omega_t = \nabla^d X_t \tag{2.3}$$

where ω_t is the stationary time series after taking the dth difference on the original non-stationary series, with error a_t which is distributed as $N(0,\sigma^2)$.

With the ARIMA model, the prediction of the current stationary series is done by using past observations.

$$\hat{w}_t = -\sum_{k=1}^{p} \phi_k \omega_{t-k} + \sum_{k=1}^{p} \theta_k a_{t-k} \tag{2.4}$$

The actual series is made stationary by taking the dth difference on the raw data as needed, i.e. $\omega_t = \nabla^d X_t$. Then the residual of the time series model is,

$$a_t = \hat{w}_t - \omega_t \tag{2.5}$$

The residual is a zero-mean IIND variable if the tool is in-control.

For some signals, the wafer-to-wafer variation is much greater than the within-wafer variation (Figure 2.3). RTSPC decomposes raw signals into long-term components (wafer-to-wafer) and short-term components (within wafer). Each component is modeled by an ARIMA model.

During production, the wafer-to-wafer averages and the within-wafer trends are filtered by their respective time series model in order to obtain the residuals. Then each component's residuals from different signals is combined into a single score by Hotelling's statistics (Figure 2.4),

$$T^2 = e^T \hat{S}^{-1} e \tag{2.6}$$

where $\hat{S}$ is the estimated covariance matrix of the residuals, which may be computed in an exponential weighted fashion,

$$\hat{S} = \sum_{i=0}^{k} \lambda^i e(k-i) e^T(k-i) \tag{2.7}$$

where k is the user-defined moving window length and λ is the exponential weighting factor.

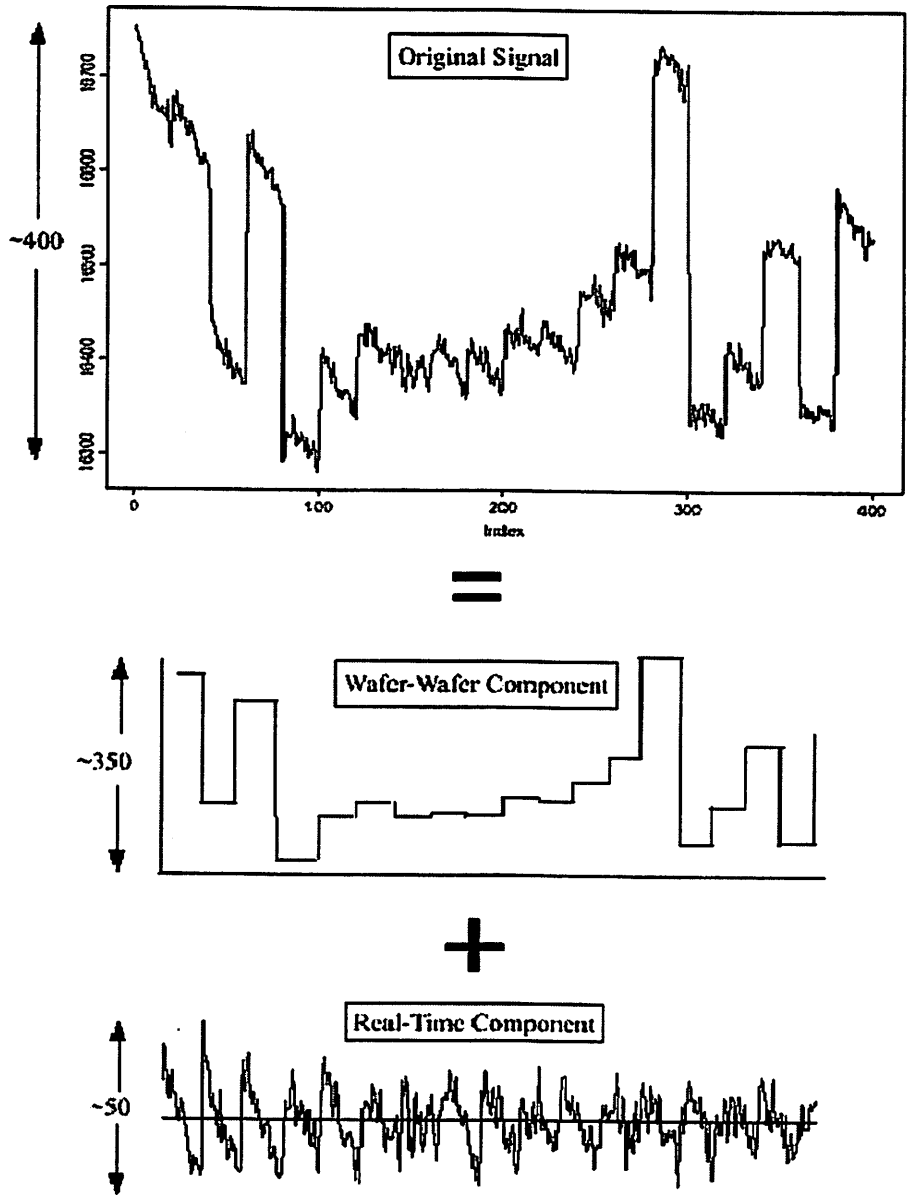


Figure 2.3. Signal decomposition for the impedance signal.

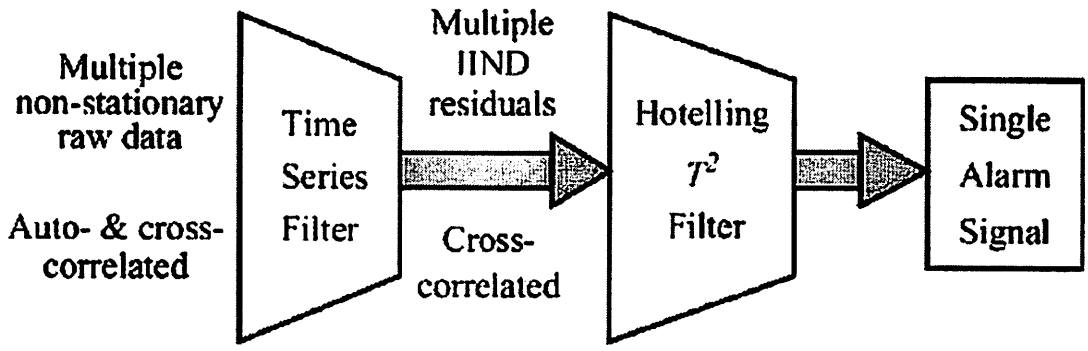


Figure 2.4. Diagram of the real-time SPC scheme.

2.3 OES

White, et al. and R. Chen have investigated the use of OES for plasma diagnostic. Figure 2.5 illustrates the emission process. Gaseous plasma species are elevated to excited states by collision with energetic electrons. As a species drops to a lower energy state, electromagnetic wave is released. Since only excited species can release electromagnetic wave, the observed spectrum reveals density of particles in the excited states, which is only small fraction of total particles, on the order of 10^{-4}. Also on the spectrum, emission from intermediates and products may overlap with that from the intended diagnostic species. As a result, it is necessary to choose selectively from the spectrum for the wavelength in performing diagnostics. We need to choose the ones that are correlate with the diagnostic parameters.

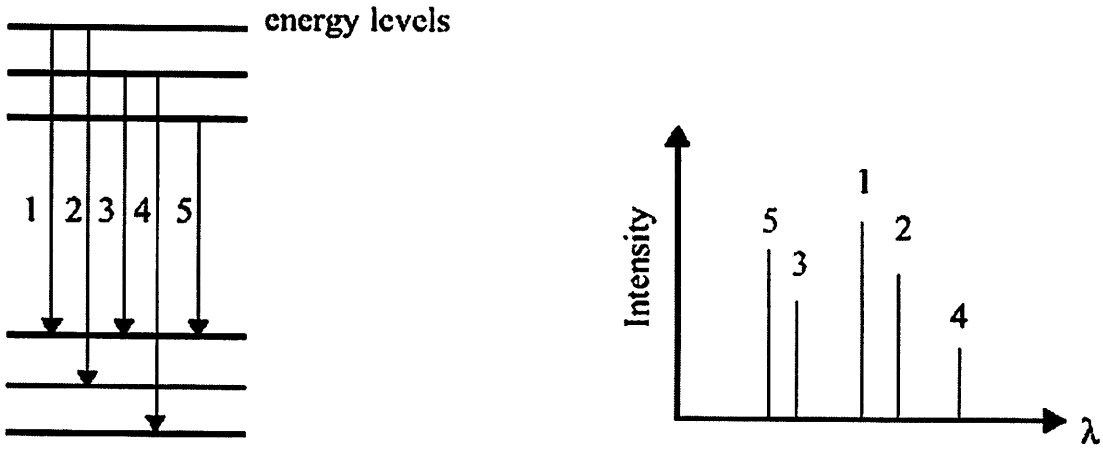


Figure 2.5. Schematic diagram of atomic emission spectrum (from [24]).

Neglecting contribution from other emissions, let us consider only the case that a plasma particle is excited from the ground state i to a state j by an electron collision, and then drops back to state i. The electromagnetic emission intensity can be written as,

$$I(\lambda_{ij}) = NP_{ij}A_{ij}(\lambda_{ij})K \tag{2.8}$$

λ_{ij} is the transition wave-length between state i and state j, N is the ground state density, A_{ij} is the Einstein emission probability, K is a correction factor which describes the effect of view volume and alignment, and P is the electron impact excitation function which represents the probability of exciting the state j by electron impact, starting from the ground state. P is a complex function of electron temperature Te, and is given by Lieberman and Lichtenberg [23].

$$P = \int_0^\infty 4\pi v_e^2 dv_e \sigma_\lambda(v_e) v_e f_e(v_e, T_e, n_e) \tag{2.9}$$

where ve is the electron velocity, σ_λ is the cross section for emission of a photon of wave-length

λ due to electron impact excitement, and f_e is the electron distribution function

which depends on electron temperature and electron density [23].

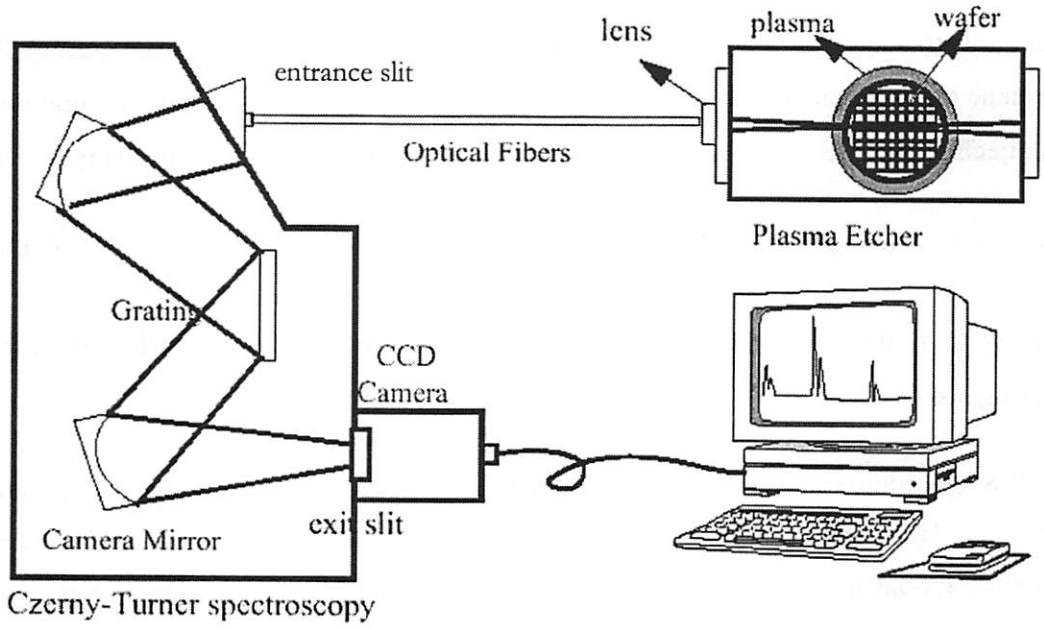


Figure 2.6. Schematic illustration of the OES instrumentation.

Figure 2.6 shows a Czerny-Turner Spectroscopy instrument. Optical emission in the plasma chamber is collected by the lens, transmitted through the optical fiber, imaged onto the entrance slit, dispersed by the diffraction grating system. Then a CCD camera detects the dispersed beam through the exit slot. The diffraction grating is usually a square. The grating equation is,

$$m\lambda = d(\sin\theta_m - \theta_i) \tag{2.10}$$

Where m is the diffraction order, d is the groove separation distance, and θ_m and θ_i are the angles of incidence and diffraction respectively. The grooves are designed to maximize the first-order diffraction (m=1) at a particular wavelength. The wavelength resolution of dispersion can be computed as,

$$\Delta\lambda = \frac{w}{m\upsilon f}\cos\theta_m \tag{2.11}$$

where w is the exit slit width, f is the spectrometer focal length, which is the distance from the exit plane to the last focusing mirror, υ is the groove density. The efficiency of a grating system for collecting light is wavelength-dependent, so the detect optical emission intensity by OES is,

$$I(\lambda_{ij}) = NP_{ij}A_{ij}(\lambda_{ij})Q(\lambda_{ij})K \tag{2.12}$$

Where $Q(\lambda_{ij})$ is the correction factor accounting for the grating system's collection efficiency at wavelength λij.

An OES spectrum contains 500~2000 wavelengths. Thus, an etcher can generate a large amount of OES data. Also, signals from different wavelength are highly correlated. Researchers have used principal component analysis (PCA) to analyze OES data.. The purpose of using PCA is to compress the data and extract relevant information. PCA splits the data matrix into systematic variation (process model) and noise (residual variance). For processing a wafer, data matrix **X** with m rows by n columns (samples by variables), can be expressed as,

$$X = t_1 p_1^T + t_2 p_2^T + \ldots + t_k p_k^T + E = T_k P_k^T + E \tag{2.13}$$

Each variable in X has been centered by subtracting a 1 by n vector of the means of variables, and scaled by d, a 1 by n standard deviation vector. The p_i are called loading vectors, which are eigenvectors of $C = X^T X$, the covariance matrix of X. They are a set of orthonormal vectors; i.e. $p_i^T p_j = 0$ for $i \neq j$, $p_i^T p_j = 1$ for $i=j$. The t_i are called the scores vectors, which for an individual sample, can be computed as,

$$t_i = Xp_i \tag{2.13}$$

And k is the number of principal component (PC) selected, which is less than or equal to the dimension of X, i.e., $k<min(m,n)$. For the highly correlated plasma etching real-time data, the

number of PCs required to adequately capture the systematic variation of a process is far smaller than m and n.

Two statistics are used, "lack of fit" statistics Q and the Hotelling's T^2 statistics. Q is a measure of the amount of variation not captured by the PCA model.

$$Q_i = e_i e_i^T = x_i (I - P_k P_k^T) x_i^T \tag{2.14}$$

where e_i is the ith row of E. T^2 is the measure of the variation within the PCA model,

$$T^2 = t_i (T_k^T T_k)^{-1} t_i^T \tag{2.15}$$

where t_i is the ith row of T_k. Notice that $T_k^T T_k$ is a diagonal matrix due the orthogonality of the $\{t_i\}$ vectors. The diagonal entries of the matrix are eigenvalues of the covariance matrix of X.

The mean vector a, standard deviation vector d and the covariance matrix need to be updated in an exponential weighted way as new process data become available.

$$a(j+1) = \sum_{j=1}^{J} \alpha^j a'(J-j) \tag{2.16}$$

$$d(j+1) = \sum_{j=1}^{J} \gamma^j d'(J-j) \tag{2.17}$$

$$C(j+1) = \sum_{j=1}^{J} \Gamma^j C'(J-j) \tag{2.18}$$

where $a'(J-j)$, $d'(J-j)$, and $C'(J-j)$ are the actual mean vector, standard deviation vector, and covariance matrix respectively, for the jth measurement. α, γ, and Γ are the user-defined exponential weight. J is the window size of the past measurement. Notice that these model parameters depend only on the past observation. The PCA model is recomputed based on the covariance matrix $C(j+1)$, i.e., new loading vector p_i and eigenvalues of the covariance matrix will be obtained. As the new process data

X_{new} become available, it is centered with $a(j+1)$ and scaled with $d(j+1)$. Then new score $t_{i,new}$ can be obtained by Eq. (2.13) with the new loadings. And Q, T^2 can be computed with Eq. (2.14) and (2.15) by replacing X_{new} with X, $t_{i,new}$ with t_i, and $T_k^T T_k$ with the new eigenvalue matrix.

2.4 RF signals

RF signals are not as sensitive to the change in plasma chamber as OES and real-time machine signals. We have not seen the use of RF signals alone for plasma diagnostic purpose. However, we will show in later chapters that RF signals can supply significant diagnostic information.

Let us denote the real-time RF voltage of the powered electrode with respect to ground $v(t)$, and real-time current flowing into the powered electrode $i(t)$. Since $v(t)$ and $i(t)$ contain harmonics of the 13.56 MHz fundamental frequency, they can be expanded into Fourier series,

$$v(t) = \sum V_n e^{j\omega_n t} \tag{2.19}$$

$$i(t) = \sum I_n e^{j\omega_n t} \tag{2.20}$$

where j is the imaginary number, $j^2=-1$; $\omega_n=2\pi nf$ is the angular frequency, V_n and I_n are the Fourier amplitudes at ω_n. The fundamental frequency f is 13.56 MHz.

In studying the plasma impedance's transient behavior, Roth, et al. has used commercial voltage probes Phillips PM 9100, and current probes Pearson 2878 to measure the fundamental frequency and its four harmonics. In our study, we use Advanced Energy's Zscan sensor and software to measure and collect RF voltage and current data at the five frequencies.

2.5 Comments on previous analytical techniques

Previous analytical techniques usually rely on assumptions about the signal. A typical assumption is that the signal is stationary (i.e., the mean value is unchanged, the noise is normally distributed and the variance is constant), or the drift over time is constant, etc. The previous techniques can only deal with the various influence on the signal on a limited scope. That is, a technique may be able to capture

some aspects of the signal well, but fail to address other aspects. For instant, ARIMA time series modeling is able to capture the general dynamic behavior of the signal within a wafer run, or wafer-to-wafer. But it fails to address memory effects (as we have seen during machine startup), or sharp spike in the signal waveform. PCA is able to handle the correlated structure in the signals, but it does not eliminate irrelevant signals that will decrease the significance of the model.

In studies of this nature, it is easy to underestimate the importance of critically examining the data, in the content of the application and the physical model that describes it. Unfortunately, automated data analysis schemes are ill-suited for this time of examination, which has to be performed by a human domain expert. In the following chapters, we will discuss the data archive setup of our diagnostic system first. Then we will present the features of the data exploring software, which allow the researcher to make both quantitative and qualitative observation on plasma etch signals. Also, we will discuss syntactic analysis in depth, which offer great flexibility in handling different influences on the signals.

Chapter 3

THE DATA ARCHIVING SYSTEM SETUP

3.1 Introduction

Many signals from a plasma etcher under operation can be monitored. An automatic data archiving system is set up for the LAM Research Rainbow 9400 Etcher, adapted for 6" wafers, and operating in the Berkeley Microfabrication Laboratory. The system archives three different sources of diagnostic signals, including optical emission spectroscopy (OES), RF power information on the fundamental frequency and several harmonics, and various other machine signals such as power, chamber pressure, temperature, gas flow rate, etc., which are collected via the SECS II interface. The data archiving system is turned on at all times. Every time the etcher starts a wafer run, a set of three time-stamp-synchronized files are created, and the data from each signal source is saved to its respective file. Finally, the data for the diagnostic signals are saved to a network archive file system, available for retrieval from one's workstation.

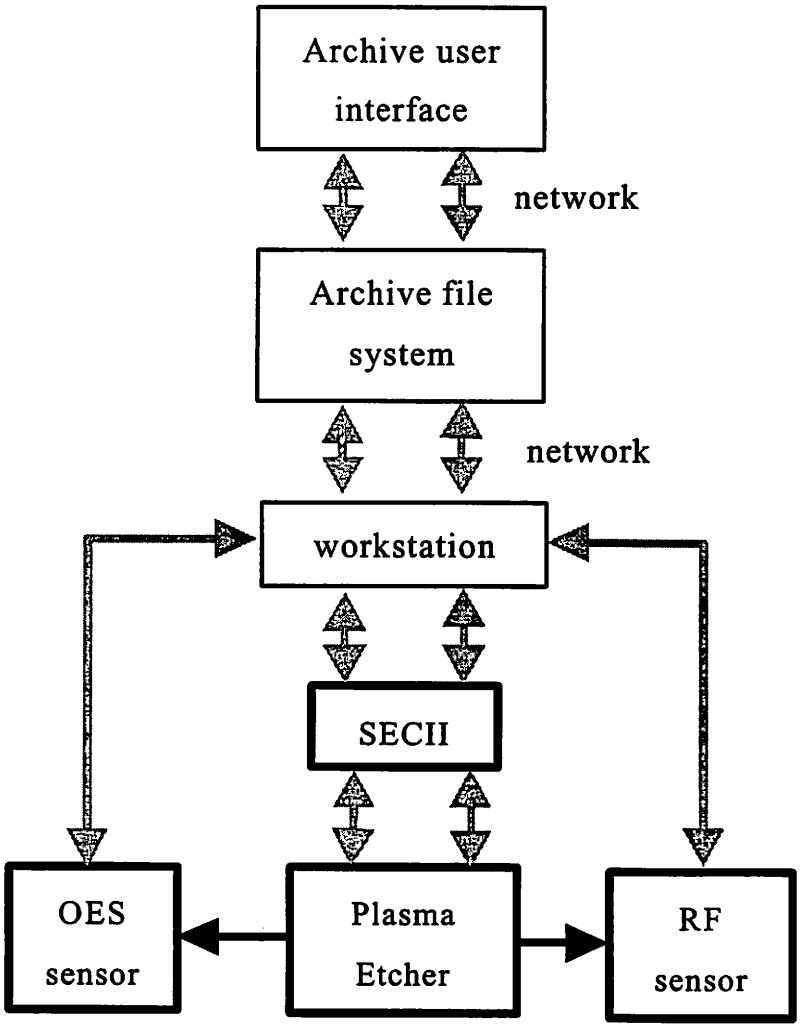


Figure 3.1. Overall setup schematic of the data archiving system.

3.2 Machine Signal Acquisition

The LAM 9400 polysilicon etcher has built-in sensors for its real-time signals. These machine signals, along machine settings, and machine status parameters, such as lot number, vacuum on-off, valve open-close, etc., can be collected by a workstation via the SECS II interface. The Accelar 1200 Network on our system runs at 100 Mbit/sec, and SECS communication ports are set to run at 9600 baud. Thus, server-client network communication will not be the bottle-neck for data transfer from the etcher to the workstation.

3.3 Ocean Optics OES Sensor

The Ocean Optics OES Sensor is PC2000-UV-VIS Fiber Optic Spectrometer with effective range of 200 nm~1100 nm. Its detector consists of a 2048-element linear CCD-array with a grating of 600 lines/mm. The entrance slit is fixed at 25 mm in width, 1000 μm in height. With no moving part, the optical bench is compactly mounted on a PC plug-in 1 MHz ISA-bus A/D card, which fits into a slot in the PC. The spectrometer collects light transmitted from Ocean Optic P400-2-UV/VIS fiber, which is a 2-meter-long, 400-μm-patch fiber. The 74-UV collimating lens, 5 mm in diameter, 10 mm in length, screws on the end of the fiber for measuring optical emission from the LAM etcher window.

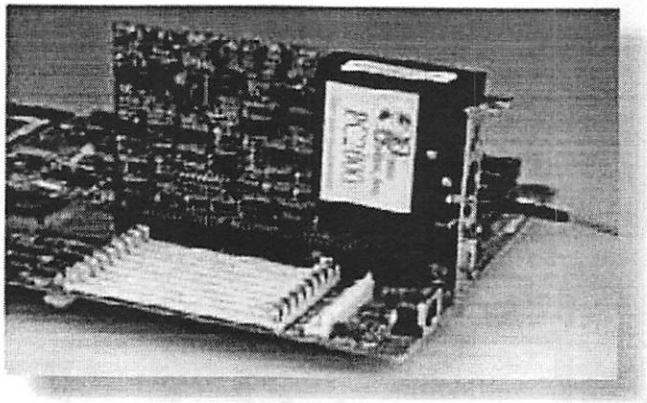


Figure 3.2. Ocean Optics PC2000-UV-VIS Spectrometer is mounted on a PC plug-in card.

3.4 RF Sensor

The RF sensor we use is the Advanced Energy Z-Scan probe, a non-intrusive RF-sensing system that allows accurate real-time measurement under powered conditions. The Z-Scan probe collects voltage,

current, and phase data of the five harmonics of the 13.56 MHz fundamental frequency. The Z-Scan system consists of a sensor, an electronic module, and an analysis software, Z-Ware. The Z-Scan sensor is designed to be inserted between matching network and the process chamber. The electronic module contain A/D converters along with analog processor, and interface boards. The analog board receives the data from the sensor. The A/D converter converts the data into digital format for the processor to read out the sensor input, at 10 readings per second. A RS-232 interface card is for making connection to the PC. Z-Ware contains various analysis features, such as graphical analysis with Smith, polar, and time domain plots. Since we do our own analysis, we just need Z-Ware to output the sensor in ASCII format for our analysis system.

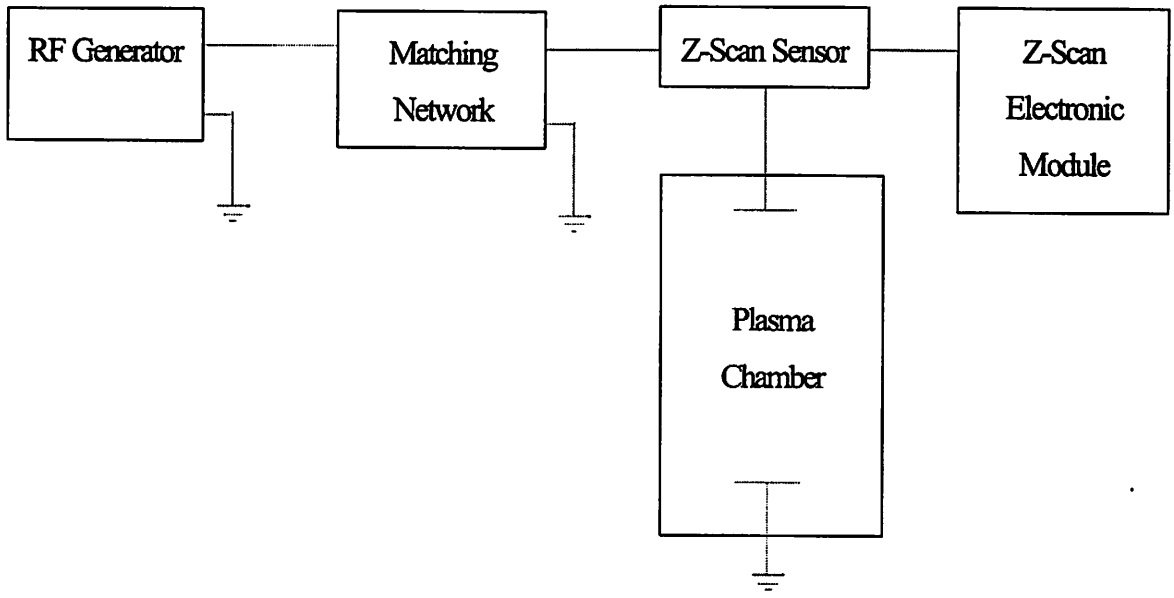


Figure 3.3. Illustration of the placement of the Z-Scan sensor probe.

3.5 The Data Archiving

The data archiving system is developed for automatic data acquisition, storage, and retrieval. The workstation acquires data through various sensors and interfaces. The Archive File System stores the data in a centralized file archival location. The researcher can examine the data files through the

Archive User Interface. The Data Archiving system has five main software components: Custom SECS II Communication Labview VI (Virtual Icon) Interface, Custom Ocean Optics PC 2000 Spectrometer Auto-Archiving Labview Interface, Z-Ware interface, Archival Storage File System, and Custom Archive Retrieval User Interface. The last two components are connected to the data acquisition workstation as shown in Figure 3.1. The first three components resides inside the workstation. These components run continuously. User interaction is not necessary unless the system is down due to network outage or other abnormal events, so that a restart is required.

The Custom SECS II Interface transmits data packets between the connecting workstation and the LAM 5 Etcher. Besides real-time signals, such as power, pressure, gas flow rate, the workstation can fetch additional machine information, such as alarm messages, process chamber status, equipment status, process wafer number.

Ocean Optics PC2000 Interface is a custom-designed Labview VI. It continuously monitors the machine information from the SECS II interface. The SECS II interface and Ocean Optics PC2000 interface share the same Labview front–end control panel, which show the display various machine information (see Figure 3.4). When a wafer is being processed and the plasma is ignited, the VI will acquire a spectrum from the OES sensor through the ISA bus. Also, the workstation will update the various information display on the front-end control panel based on the user-defined integration time. In our case, we set it to be 0.95 second. Due to the overhead of data transmission via the SECS II interface, which run at 9600 baud. The update frequency of the display is about once every 1.9 seconds. There is a graphical display showing the current location of the wafer in the etcher while the wafer is being processed. On the upper right corner, there is the intensity vs. wavelength OES plot, which will be frequently updated when a wafer is being etched in the chamber. On the Lower left corner, there is an array of number entries, which allows users to specify which real-time signals to be saved to the archive. Table 3.1 lists the signals used. The numerical code names can be found in the appendix of the LamStation Rainbow Manual.

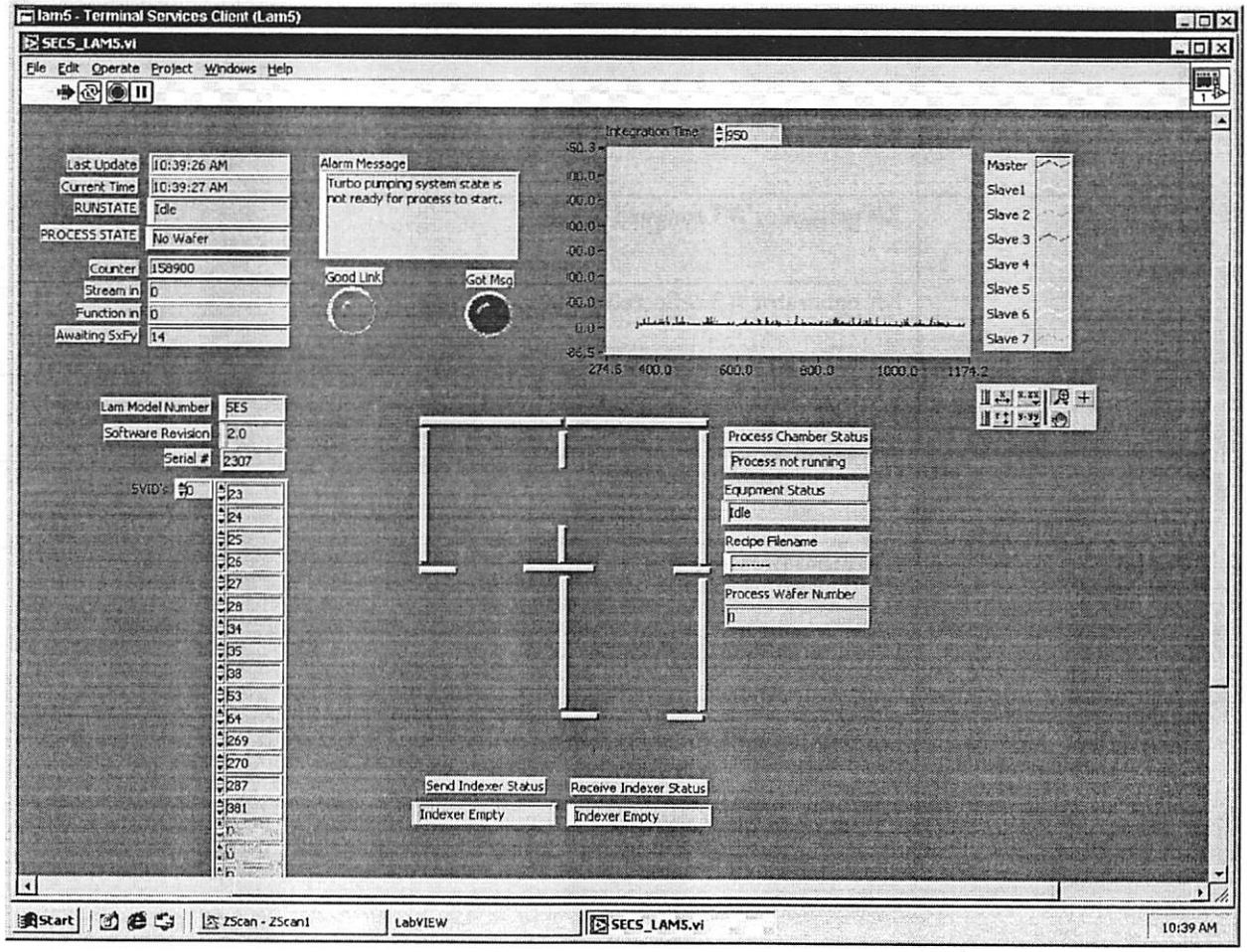


Figure 3.4. The front-end control panel for both SECS II and OES.

Code Name	Signal Name
23	RF power #1
24	RF line impedance #1
26	RF match #1 peak RF voltage
27	RF generator #1 reflected power

28	RF generator #1 forward power
53	RF generator #1 power control
550	RF generator #3 forward power
551	RF generator #3 reflected power set point
552	RF generator #3 forward power set point
241	Chamber pressure set point
242	Chamber pressure
600	Upper chamber temp set point
601	Lower chamber temp set point
612	Upper chamber temp sens
613	Lower chamber temp sens
8	Gas #1 (Cl_2) flow
45	Gas #1 (Cl_2) set point
10	Gas #2 (HBr) flow
46	Gas #2 (HBr) set point
12	Gas #3 (CF_3) flow
47	Gas #3 (CF_3) set point

14	Gas #4 (O_2) flow
48	Gas #4 (O_2) set point
16	Gas #5 (He/Ar) flow
49	Gas #5 (He/Ar) set point
18	Gas #6 (SF_6) flow
50	Gas #6 (SF_6) set point
1	Gas #7 (O_2) flow
41	Gas #7 (O_2) set point
2	Gas #7 (O_2) current sense
42	Gas #8 (CF_4) set point
423	Recipe number
20	RF match #1 tuning position
21	RF match #1 load coil position
564	TCP tuning cap pos
578	TCP match load cap
332	Current recipe step #
3	Gas #8 (CF_4) flow

34	Endpoint detector a (SiCl 405nm)
35	Endpoint detector b (CO 520nm)

Table 3.1. List of selected machine real-time signals for data archiving.

While the Z-Ware software has many data-analysis features, we only need the Z-Ware to convert the RF voltage, current, phase data from the Z-Scan sensor to ASCII format. The final acquiring rate on Z-Ware for the Z-Scan data is about 1 Hz.

The Archive File System Stores the data files acquired by the workstation to a centralized file system location, with a symbolic pathname of '\\skopelos\bcamarchive$\lam5\archive'. Every time a wafer run is finished, a set of three files for the three sources of data is stored. The log file "lam5.log" is updated to record the change. The OES file name format is "lam5_pc2000.*nnn*.dat"; for the machine real-time signal, "lam5_pc2000.*nnn*.dat.SVID"; for Z-Scan data, "lam5_pc2000.*mmm*.dat.ZSCAN"; where *nnn* or *mmm* is the time stamp, the number of seconds past with respect to a fixed time reference. Notice that since both the machine real-time signals and OES data have Labview interfaces, their files can be synchronized to have the same time stamp. At the time of the writing, the Z-Ware has not been synchronized with the Labview interfaces to produce a common time stamp. Some programming is required to group the files together for data retrieval of each wafer run. The researcher can examine the data files from his/her workstation through the Archive User-Interface. The user-friendly interface allow users to view files, retrieve file, store files, or edit files attributes.

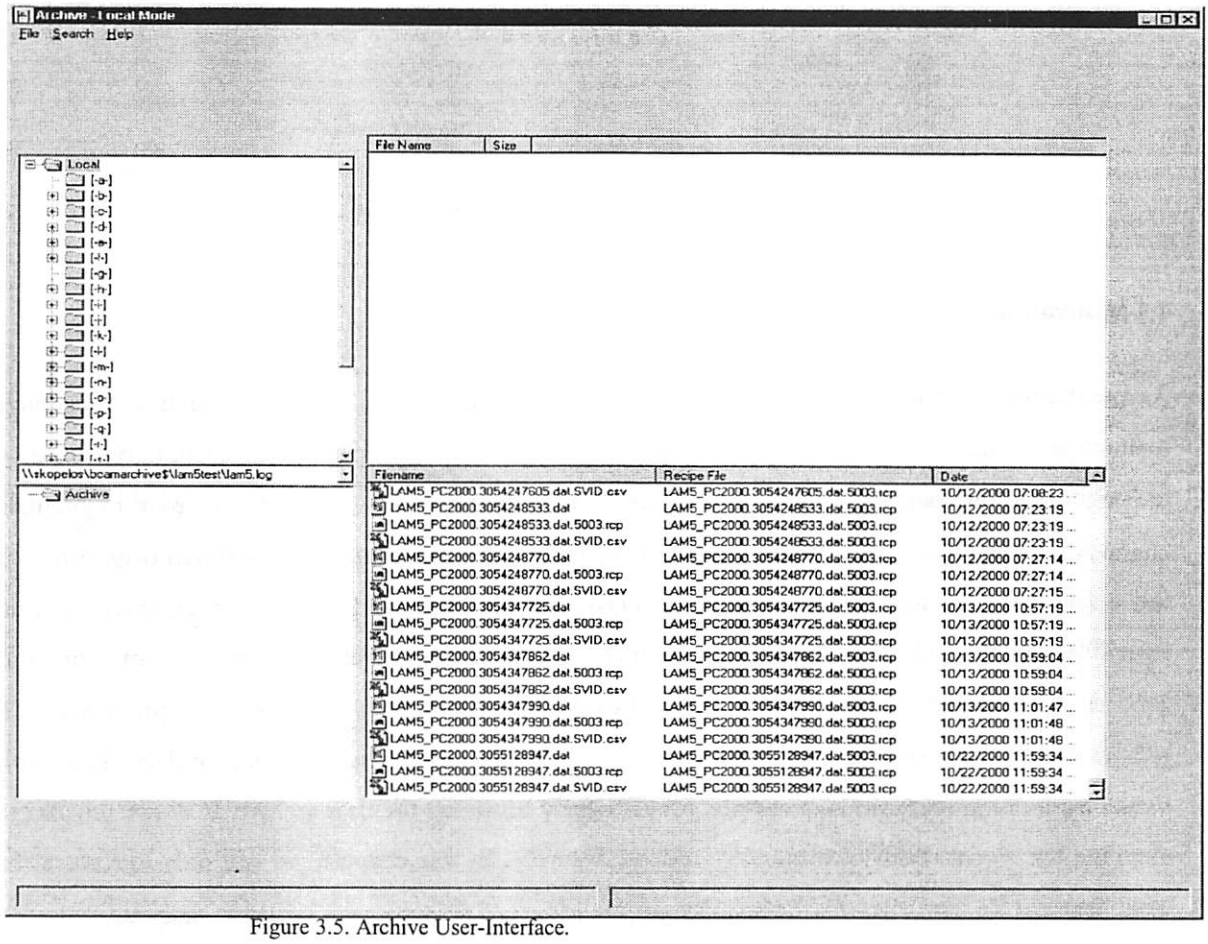


Figure 3.5. Archive User-Interface.

Chapter 4

DATA EXPLOREATION SYSTEM

4.1 Motivation

As mentioned previously, plasma etch signals are subject to various influences, such as preventive maintenance, machine aging, and chamber memory effect. Before proposing any meaningful analytical technique for the signals, a researcher needs to look through the data first in order to mentally characterize the influences. That is, before developing computer routines to perform diagnostics on the signals, s/he has to be able to perform the task with her/his naked eyes. However, there are more than 2000 signals available for investigation on our data archiving system. Also, the system is on at all time. A few hundred kilobytes of data is saved to the file system every time a wafer is processed. It is tedious and time-consuming for a researcher to browse through this huge amount of data. As a result, a data exploration software is developed for efficiently browsing the data archive. It allows the user to examine the signals both quantitatively and qualitatively. In this chapter, we will only introduce the basic features. In later chapters, we will discuss the advanced features with syntactic analysis.

4.2 Features

The software allows the user to retrieve a list of files according to the specified time interval. Then the files are grouped into a wafer list, with three files from each of the three sensor sources for each wafer. The user can indicate a list of signals s/he wishes to investigate. A window can be specified for a particular portion of the etch waveform. The windowed waveforms can be plotted on the screen, wither one after another, or stacking on top of one another. Recipe name distribution can be generated. Also, distribution for the value of the windowed portion of the waveform can be created. Further, wafer data lists can be extracted based on recipe names or values of the windowed portion. Finally, signal-vs.-signal plots can be generated and signal-vs.-signal correlations can be computed. Let us step through the features in the software.

Change exploration settings

This item allows the user to change various inputs of the software, including, time interval, window selection, signal selection. The user can specify the *from-date* and *to-date* of a time interval. With a new time interval, the user can get a complete new list or append the wafer data to an existing list. For the list of wafer data, the software has the options of specifying absolute windows and step window. A different window can be specified for each of the three sources of data. An etch process has a number of steps. For instance, Recipe 5001 has three steps, with step 3 as its main etch step; Recipe 5003 has seven steps, with step 5 as its main etch step. The step number information available is from the SVID machine data. As previously mentioned, due to the different hardware interface, ZSCAN data is not synchronized with SVID and OES data, and have a different sampling rate (1 sample/1 second instead of 1 sample/1.9 second for SVID and OES), so the user can only specify absolute windows for ZSCAN data.

For step window selection, the user need to input step number, delay, and window size. If the window size exceeds the end of the step, or if the user specifies the window size as –1, the data up the last entry of the step will be selected. For absolute window selection, the user just needs to specify the start index and the end index. If the end index exceeds the end of the data, data up to the last entry will be selected. There is a subtle design issue here. For absolute windowing, we do not adapt the step window format which lets the user specify the delay and window size, so that the user can input –1 window size to fetch up until the last data entry. The reason is that when we do windowing, we usually want to get the data from stable region of the same step. The last data entries often correspond to the power-off state and are usually not in the same step we are interested in.

A list of signals can be selected for investigation, with this format: *source1 index1 source2 index2* ... A source is one of the three sources of signals, OES, SVID, and ZSCAN. For OES, the index range from 1 to 2048. Since the OES effective wavelength is 200 nm~1100 nm, to get a wavelength conversion for an index, one can use this formula: wavelength (nm)=200 + index * (1100 – 200)/2048. For SVID, the variable index can be looked up in Table 3.1. The index is the ordinal number on the table. The index for the first variable, RF power #1, code 23, is indexed as "1." For ZSCAN, there are 35 variables for selection, as shown in Table 4.1.

Index	Symbol	Signal Name
N+1	Vrms	Root-mean-square voltage
N+2	Irms	Root-mean-square current
N+3	Phase	Phase between current and voltage
N+4	P	Power
N+5	Z	Impedance
N+6	R	Resistance
N+7	X	Reactance

Table 4.1. ZSCAN signal index.

Where N is from 0 to 4, indicating the *N*th harmonic. N=0 indicates the fundamental frequency. Notice that only the first three variables in the table are independent. The other four are calculated by the Z-Ware software.

Build wafer list & Add wafer data to list

The user can build a wafer data list from scratch or add wafer data to the current existing list by specifying a new time interval. The program first fetches all the data files that fall between the *from-date* and the *to-date*, then builds the list of wafer data records from the files. Each wafer data record consists of three files, with each file from its respective source. While the OES file is synchronized with the SVID file, the ZSCAN file is not synchronized with them, that is, the ZSCAN file has a slightly different time stamp from the other two files. Therefore, the program needs to look through the list of files to find the closest time-stamp match of the ZSCAN file from the other two files.

Generate recipe distribution & Extract wafers on recipe number

The program gets the recipe number from the SVID file from each of the wafer record in the list, and then generates the count for each recipe number. Most of the analysis is done on the wafer data with the same recipe number. It is useful to see the distribution of the recipe number; it helps to decide which recipe's data should be used for analysis, since the more wafer data for a recipe there are, the more significant the analysis is. The program prompts for a recipe number and extract the data with the recipe.

Generate signal value distribution & Extract wafers on signal value

The data window should be specified around the steady region of the waveform. During stable etching, the signal intensity should be more or less steady. However, on a wafer-to-wafer basis, signal intensity may fall into clusters, due to different exposure masks, etching material compositions, equipment status, etc. The user may want to analyze wafer data with the signal value in a certain cluster only. Therefore, it is useful to generate a distribution of the average windowed signal value. In extracting data, the user needs to specify the signal index, and the upper and lower bounds for the signal value.

Generate entire within-wafer plot, windowed within-wafer plot & Concatenated windowed plot

In order to assess the nature of the signal, it is very important for the user to get a visual impression of the waveform. The program can plot the entire waveforms, so that the user can make a judgment on which portion of the signal to be windowed. The windowed within-wafer plot is the plot that shows all the windowed waveforms stacking on top of each other. This can show the trend of the signal during etching, as well as the variation of the similar etch region. The concatenated windowed wafer plot takes all the windowed waveform and connects them back-to-back. This plot can demonstrate the wafer-to-wafer trend of the signal.

Generate cross-wafer plot & Compute signal correlation

A cross-wafer plot is a signal-vs.-signal plot. That is, the program generates all the combinations of two signals from the list of user-selected signals. The average values of the windowed region are

computed for all the wafers. Then, the program generates a X-Y plot for each signal combination for the average values of the windowed region (for example, see Figure 4.8). On the plot, each data point comes from a different wafer. Subsequently, signal-vs.-signal correlation can be calculated for all the combinations. The plots and the respective correlation information are very useful for reducing the number of variables needed for analysis.

Manipulate wafer list

The program allows the user to list all the wafer records, and selectively delete some records. Or the user can move some records to a buffer list, in case they might be useful for later analysis. This feature does not only facilitate the removal outliers, but also allows the user to manipulate the data list based on his/her expert knowledge on the data. There is an interactive windowed within-wafer plot to let the user see the effect of manipulating the wafer list. It generates the same plot as windowed within-wafer plot, only that it puts the windowed waveforms one at a time with user interaction.

4.3 Examples

Wafer state experiment

At the end of February 2001, a wafer state experiment was performed. The machine setting was adjusted in order to achieve different etch rate and uniformity. For details of experimental design and analysis, see Chapter 6. In this example, we want examine the waveform and value spread of some of the significant signals, and perform some correlation calculations. Sixteen wafers were to be processed on February 24. However, an error occurred on the 14th wafer, so that, the remaining 3 wafers were processed on February 28. To get the data of 16 wafers for the exploration software, first, build a list for the time interval of "24-Feb-2001 12:30:00" ~ "24-Feb-2001 13:30:00".

wafer 1	
LAM5_PC2000.983042971.595.dat	24-Feb-2001 12:33:36
LAM5_PC2000.983042971.595.dat.SVID.csv	24-Feb-2001 12:33:36
LAM5_PC2000.983043032.000.dat.ZSCAN.csv	24-Feb-2001 12:48:58
wafer 2	
LAM5_PC2000.983043409.916.dat	24-Feb-2001 12:37:36
LAM5_PC2000.983043409.916.dat.SVID.csv	24-Feb-2001 12:37:37
LAM5_PC2000.983043427.000.dat.ZSCAN.csv	24-Feb-2001 12:48:58
wafer 3	
LAM5_PC2000.983043537.970.dat	24-Feb-2001 12:39:39
LAM5_PC2000.983043537.970.dat.SVID.csv	24-Feb-2001 12:39:40

LAM5_PC2000.983043552.000.dat.ZSCAN.csv	24-Feb-2001 12:48:58
wafer 4	
LAM5_PC2000.983043701.164.dat	24-Feb-2001 12:42:23
LAM5_PC2000.983043701.164.dat.SVID.csv	24-Feb-2001 12:42:24
LAM5_PC2000.983043716.000.dat.ZSCAN.csv	24-Feb-2001 12:48:58
wafer 5	
LAM5_PC2000.983043940.489.dat	24-Feb-2001 12:46:24
LAM5_PC2000.983043940.489.dat.SVID.csv	24-Feb-2001 12:46:25
LAM5_PC2000.983043955.000.dat.ZSCAN.csv	24-Feb-2001 13:18:58
wafer 6	
LAM5_PC2000.983044069.404.dat	24-Feb-2001 12:48:32
LAM5_PC2000.983044069.404.dat.SVID.csv	24-Feb-2001 12:48:32
LAM5_PC2000.983044083.000.dat.ZSCAN.csv	24-Feb-2001 13:18:58
wafer 7	
LAM5_PC2000.983044214.162.dat	24-Feb-2001 12:50:58
LAM5_PC2000.983044214.162.dat.SVID.csv	24-Feb-2001 12:50:59
LAM5_PC2000.983044232.000.dat.ZSCAN.csv	24-Feb-2001 13:18:58
wafer 8	
LAM5_PC2000.983044378.649.dat	24-Feb-2001 12:53:41
LAM5_PC2000.983044378.649.dat.SVID.csv	24-Feb-2001 12:53:42
LAM5_PC2000.983044393.000.dat.ZSCAN.csv	24-Feb-2001 13:18:58
wafer 9	
LAM5_PC2000.983044598.194.dat	24-Feb-2001 12:57:24
LAM5_PC2000.983044598.194.dat.SVID.csv	24-Feb-2001 12:57:23
LAM5_PC2000.983044613.000.dat.ZSCAN.csv	24-Feb-2001 13:18:58
wafer 10	
LAM5_PC2000.983044731.826.dat	24-Feb-2001 12:59:36
LAM5_PC2000.983044731.826.dat.SVID.csv	24-Feb-2001 12:59:37
LAM5_PC2000.983044747.000.dat.ZSCAN.csv	24-Feb-2001 13:18:58
wafer 11	
LAM5_PC2000.983044929.060.dat	24-Feb-2001 13:02:51
LAM5_PC2000.983044929.060.dat.SVID.csv	24-Feb-2001 13:02:51
LAM5_PC2000.983044944.000.dat.ZSCAN.csv	24-Feb-2001 13:18:58
wafer 12	
LAM5_PC2000.983045070.383.dat	24-Feb-2001 13:05:13
LAM5_PC2000.983045070.383.dat.SVID.csv	24-Feb-2001 13:05:13
LAM5_PC2000.983045084.000.dat.ZSCAN.csv	24-Feb-2001 13:18:58
wafer 13	
LAM5_PC2000.983045261.438.dat	24-Feb-2001 13:08:23
LAM5_PC2000.983045261.438.dat.SVID.csv	24-Feb-2001 13:08:24
LAM5_PC2000.983045276.000.dat.ZSCAN.csv	24-Feb-2001 13:18:58
wafer 14	
LAM5_PC2000.983045455.637.dat	24-Feb-2001 13:11:38
LAM5_PC2000.983045455.637.dat.SVID.csv	24-Feb-2001 13:11:39
LAM5_PC2000.983045471.000.dat.ZSCAN.csv	24-Feb-2001 13:18:58
wafer 15	
LAM5_PC2000.983046088.507.dat	24-Feb-2001 13:21:57
LAM5_PC2000.983046088.507.dat.SVID.csv	24-Feb-2001 13:21:58
wafer 16	
LAM5_PC2000.983046179.128.dat	24-Feb-2001 13:23:32
LAM5_PC2000.983046179.128.dat.SVID.csv	24-Feb-2001 13:23:33

According to the experiment logbook, only wafers 2~14 are the ones that took part in the wafer state experiment. So, we go to the Manipulate Wafer List Menu to delete 1, 15, 16. Then, we add more wafer data to the existing 13 records, with time interval "28-Feb-2001 17:30:00" ~ "28-Feb-2001 18:30:00". Upon inspection of the new list, we can find that wafers 14~18 and 22~26 should also be deleted. We are then left with the 16 wafers of the experiment.

wafer 1

LAM5_PC2000.983043409.916.dat	24-Feb-2001 12:37:36
LAM5_PC2000.983043409.916.dat.SVID.csv	24-Feb-2001 12:37:37
LAM5_PC2000.983043427.000.dat.ZSCAN.csv	24-Feb-2001 12:48:58
wafer 2	
LAM5_PC2000.983043537.970.dat	24-Feb-2001 12:39:39
LAM5_PC2000.983043537.970.dat.SVID.csv	24-Feb-2001 12:39:40
LAM5_PC2000.983043552.000.dat.ZSCAN.csv	24-Feb-2001 12:48:58
wafer 3	
LAM5_PC2000.983043701.164.dat	24-Feb-2001 12:42:23
LAM5_PC2000.983043701.164.dat.SVID.csv	24-Feb-2001 12:42:24
LAM5_PC2000.983043716.000.dat.ZSCAN.csv	24-Feb-2001 12:48:58
wafer 4	
LAM5_PC2000.983043940.489.dat	24-Feb-2001 12:46:24
LAM5_PC2000.983043940.489.dat.SVID.csv	24-Feb-2001 12:46:25
LAM5_PC2000.983043955.000.dat.ZSCAN.csv	24-Feb-2001 13:18:58
wafer 5	
LAM5_PC2000.983044069.404.dat	24-Feb-2001 12:48:32
LAM5_PC2000.983044069.404.dat.SVID.csv	24-Feb-2001 12:48:32
LAM5_PC2000.983044083.000.dat.ZSCAN.csv	24-Feb-2001 13:18:58
wafer 6	
LAM5_PC2000.983044214.162.dat	24-Feb-2001 12:50:58
LAM5_PC2000.983044214.162.dat.SVID.csv	24-Feb-2001 12:50:59
LAM5_PC2000.983044232.000.dat.ZSCAN.csv	24-Feb-2001 13:18:58
wafer 7	
LAM5_PC2000.983044378.649.dat	24-Feb-2001 12:53:41
LAM5_PC2000.983044378.649.dat.SVID.csv	24-Feb-2001 12:53:42
LAM5_PC2000.983044393.000.dat.ZSCAN.csv	24-Feb-2001 13:18:58
wafer 8	
LAM5_PC2000.983044598.194.dat	24-Feb-2001 12:57:24
LAM5_PC2000.983044598.194.dat.SVID.csv	24-Feb-2001 12:57:23
LAM5_PC2000.983044613.000.dat.ZSCAN.csv	24-Feb-2001 13:18:58
wafer 9	
LAM5_PC2000.983044731.826.dat	24-Feb-2001 12:59:36
LAM5_PC2000.983044731.826.dat.SVID.csv	24-Feb-2001 12:59:37
LAM5_PC2000.983044747.000.dat.ZSCAN.csv	24-Feb-2001 13:18:58
wafer 10	
LAM5_PC2000.983044929.060.dat	24-Feb-2001 13:02:51
LAM5_PC2000.983044929.060.dat.SVID.csv	24-Feb-2001 13:02:51
LAM5_PC2000.983044944.000.dat.ZSCAN.csv	24-Feb-2001 13:18:58
wafer 11	
LAM5_PC2000.983045070.383.dat	24-Feb-2001 13:05:13
LAM5_PC2000.983045070.383.dat.SVID.csv	24-Feb-2001 13:05:13
LAM5_PC2000.983045084.000.dat.ZSCAN.csv	24-Feb-2001 13:18:58
wafer 12	
LAM5_PC2000.983045261.438.dat	24-Feb-2001 13:08:23
LAM5_PC2000.983045261.438.dat.SVID.csv	24-Feb-2001 13:08:24
LAM5_PC2000.983045276.000.dat.ZSCAN.csv	24-Feb-2001 13:18:58
wafer 13	
LAM5_PC2000.983045455.637.dat	24-Feb-2001 13:11:38
LAM5_PC2000.983045455.637.dat.SVID.csv	24-Feb-2001 13:11:39
LAM5_PC2000.983045471.000.dat.ZSCAN.csv	24-Feb-2001 13:18:58
wafer 14	
LAM5_PC2000.983407267.387.dat	28-Feb-2001 17:41:49
LAM5_PC2000.983407267.387.dat.SVID.csv	28-Feb-2001 17:41:50
LAM5_PC2000.983407282.000.dat.ZSCAN.csv	28-Feb-2001 17:49:54
wafer 15	
LAM5_PC2000.983407394.470.dat	28-Feb-2001 17:43:56
LAM5_PC2000.983407394.470.dat.SVID.csv	28-Feb-2001 17:43:57
LAM5_PC2000.983407409.000.dat.ZSCAN.csv	28-Feb-2001 17:49:54
wafer 16	
LAM5_PC2000.983407522.213.dat	28-Feb-2001 17:46:06
LAM5_PC2000.983407522.213.dat.SVID.csv	28-Feb-2001 17:46:07
LAM5_PC2000.983407538.000.dat.ZSCAN.csv	28-Feb-2001 18:19:56

We know that the wafers are processed with Recipe 5001; we may generate a recipe distribution to verify this. There are three steps in Recipe 5001, with step 3 as the main etch step. Let us first inspect the entire within-wafer plot (Figure 4.1, 4.2).

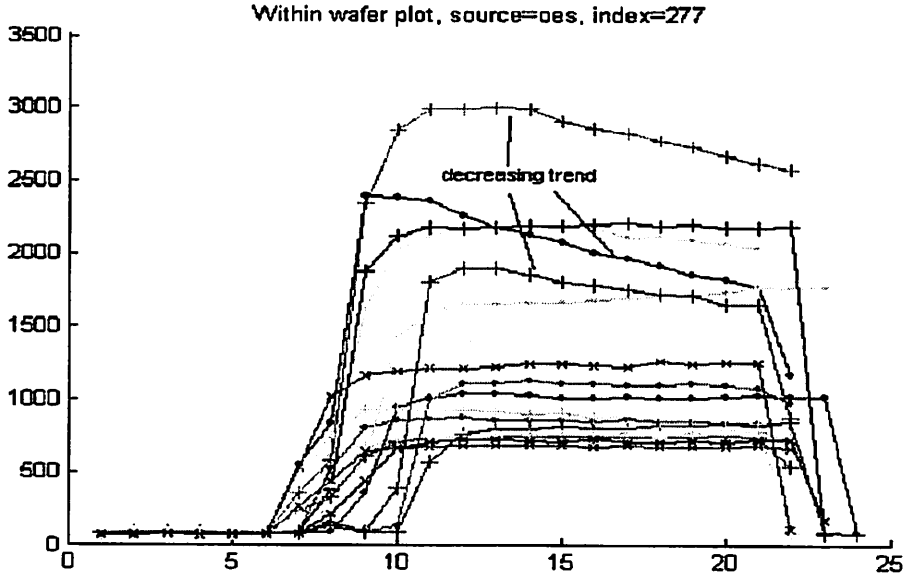


Figure 4.1. The entire within-wafer plot for CF_2 321 nm line.

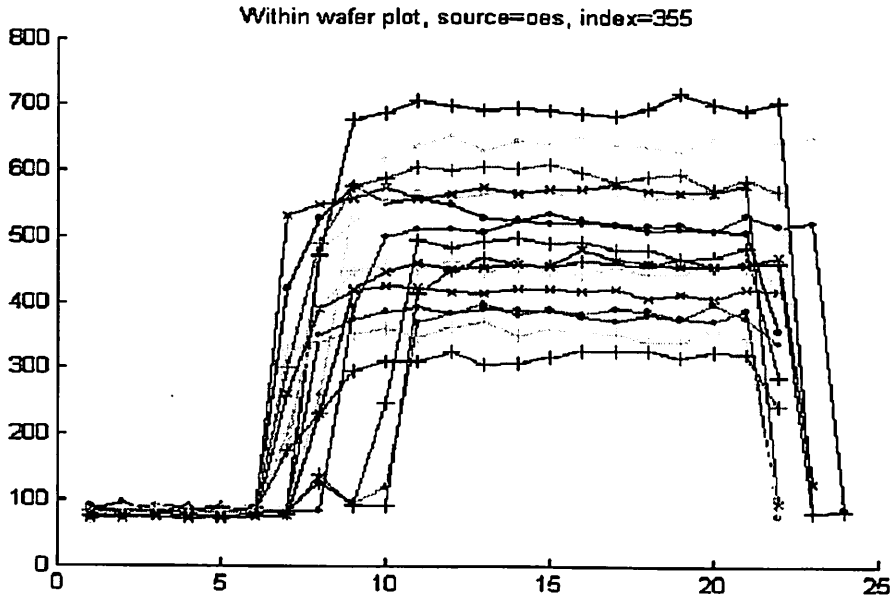


Figure 4.2. The entire within-wafer plot for HBr 355 nm line.

According to the result shown in Chapter 5, the CF_2 321 nm line is effective for endpoint detection, and the HBr 355 nm line is useful for etch rate prediction. As seen in the two plots, the starts of step 3 do not line up, and there is a fall-off for the power-off state in the end. The absolute stable etch window can be chosen to be 13~20. However, upon close inspection, we see that, for the CF_2 line there are three plots that have the decreasing trend. In order to get bigger window size and better average values for computing correlation, we may want to get step windows. Let us examine the windowed within-wafer plot for the entire step 3.

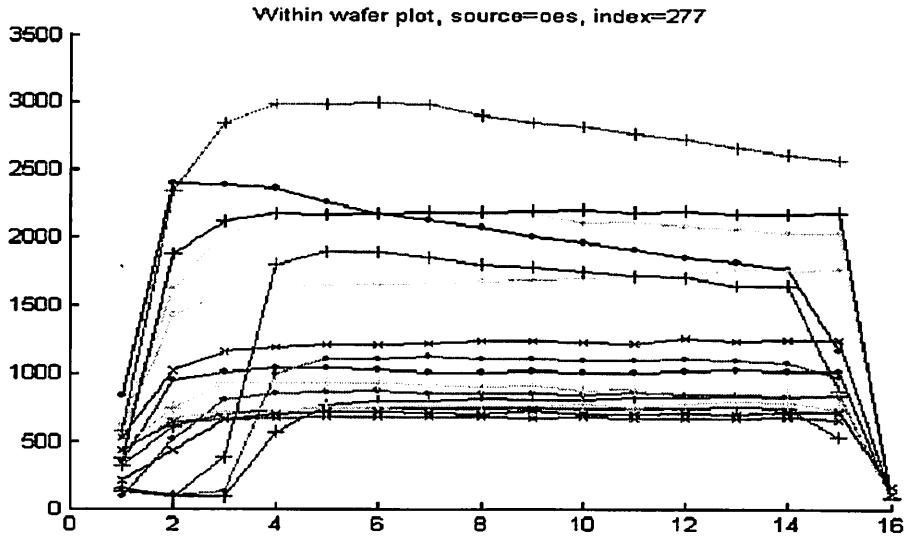


Figure 4.3. Windowed within-wafer plot for CF_2 321 nm line for step 3.

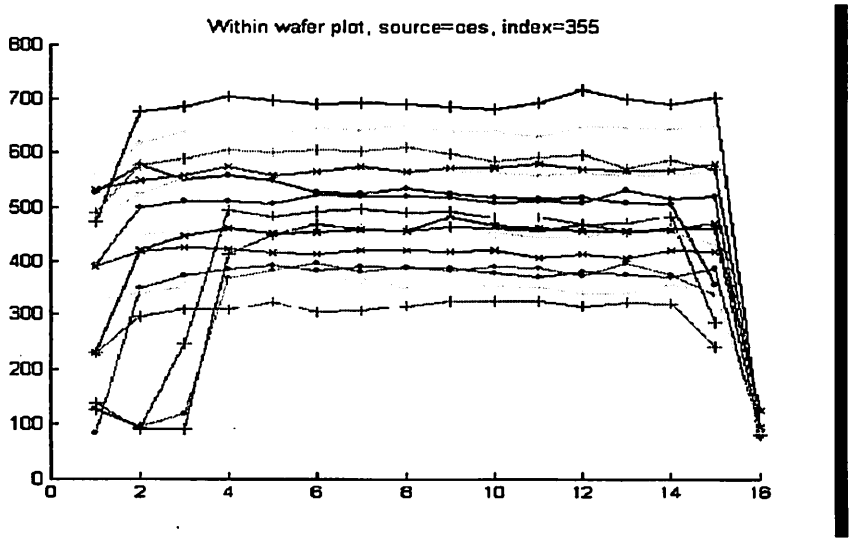


Figure 4.4. Windowed within-wafer plot for HBr 355 nm line for step 3.

We now see that the window size can be a little bit bigger, 5 ~ 14 for step 3. So let the delay be 4, and the size to be 9. We have,

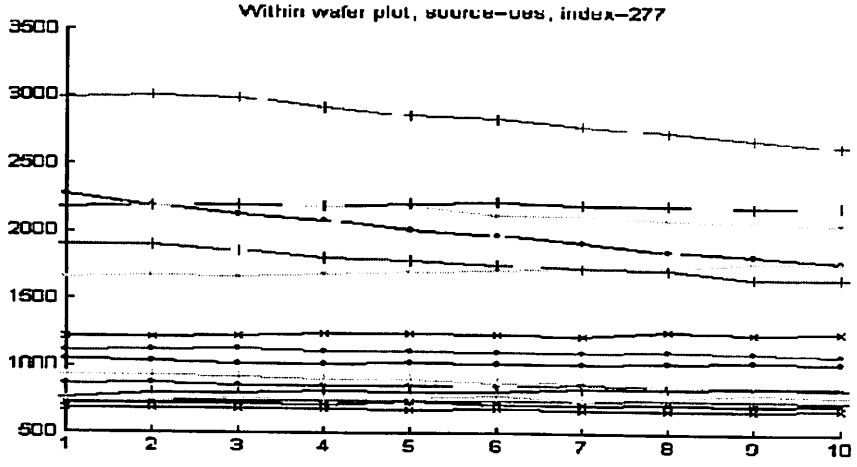


Figure 4.5. Windowed within-wafer plot for CF_2 321 nm line for step 3 size 9.

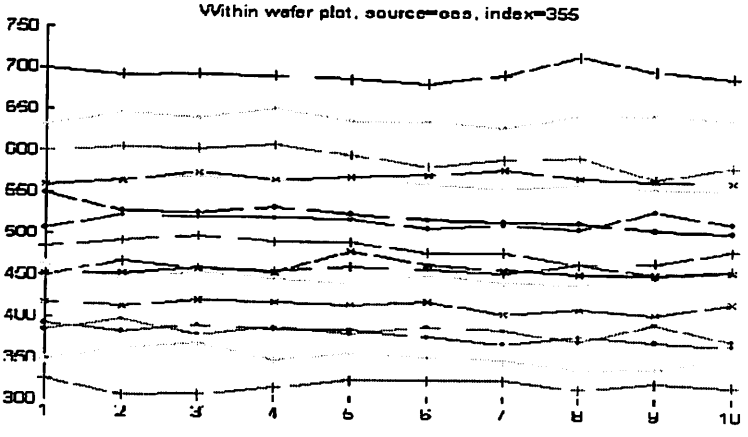


Figure 4.6. Windowed within-wafer plot for HBr 355 nm line for step 3 size 9.

Notice the wafer-to-wafer signal intensity is reasonably well-spread, covering a relatively wide range of value, and not clustering much. This suggests that the experimental design is reasonably sound. Figure 4.7 shows the concatenated plot. From there, we can see that data for wafers 2, 7, 10 have the obvious declining trends.

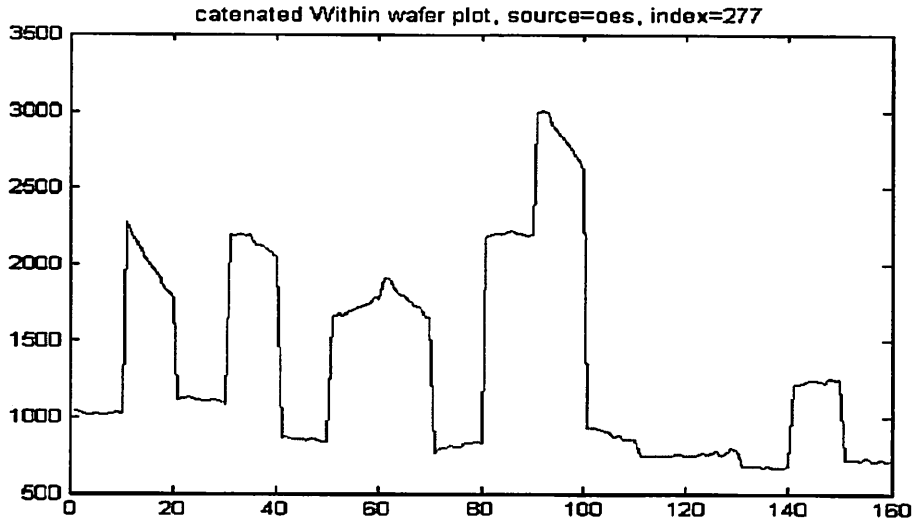


Figure 4.7. Concatenated Windowed within-wafer plot for CF_2 321 nm line.

Lastly, we want to get signal-vs.-signal plots and the correlation coefficient for signal CF_2 275 nm, CF_2 321 nm, and HBr 355 nm. We see that the endpoint detector, CF_2 275 nm and CF_2 321 nm signals have an almost perfect correlation since they are from the same chemical species. On the other hand, HBr 355 nm, the etch rate indicator, is very much correlated with the two CF_2 signals as well.

Correlation between OES 176nm line and OES 277nm line is 0.997439,

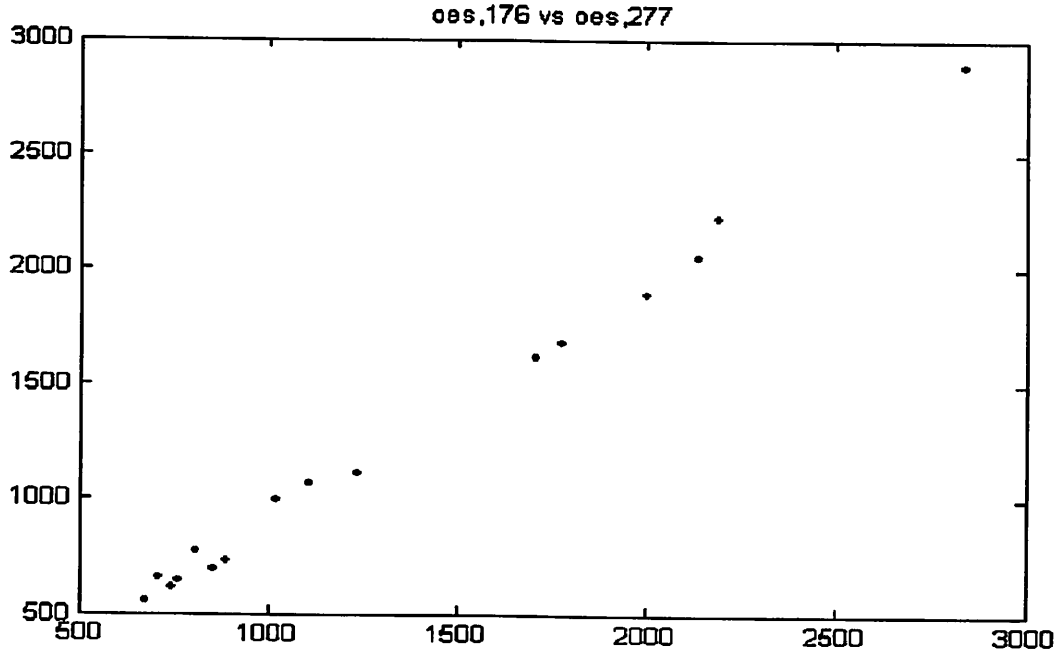


Figure 4.8. Signal-vs.-signal plot for CF_2 275 nm and CF_2 321 nm.

Correlation between 176nm and 355nm line intensities is 0.755675,

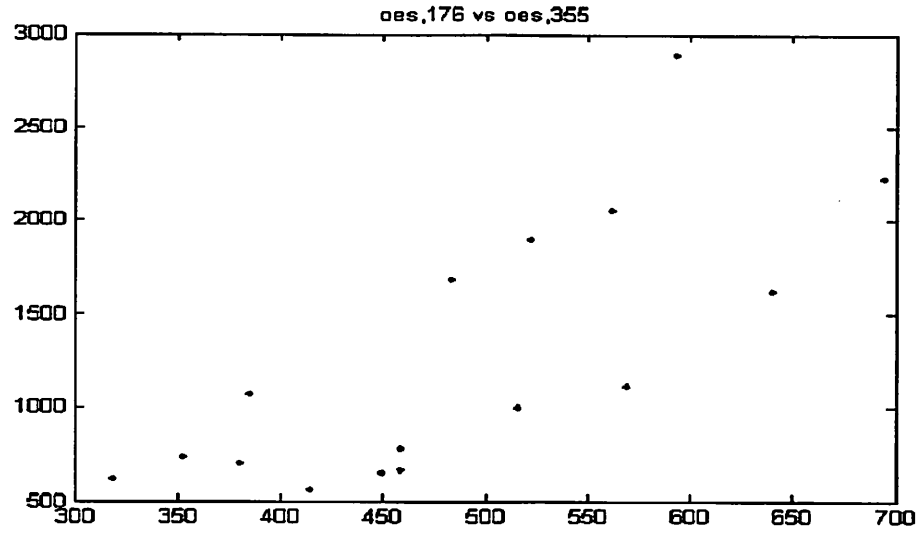


Figure 4.9. Signal-vs.-signal plot for CF_2 275 nm and HBr 355 nm.

Correlation between 277nm and 355 nm line intensities is 0.743946,

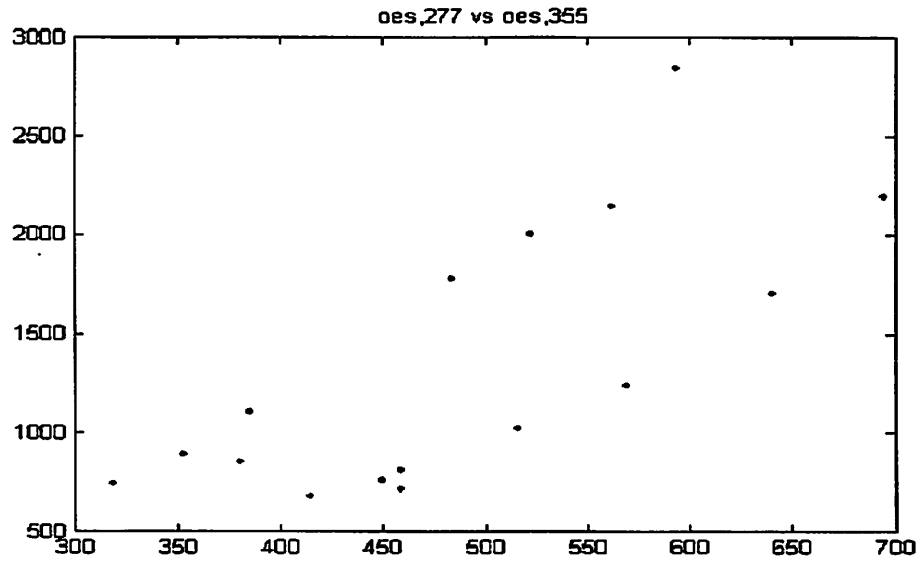


Figure 4.10. Signal-vs.-signal plot for CF_2 321 nm and HBr 355 nm.

The above example shows the basic usage of the software. Next, we will show examples that show the nature of plasma etch signals.

Some plots showing the nature of plasma etch signals

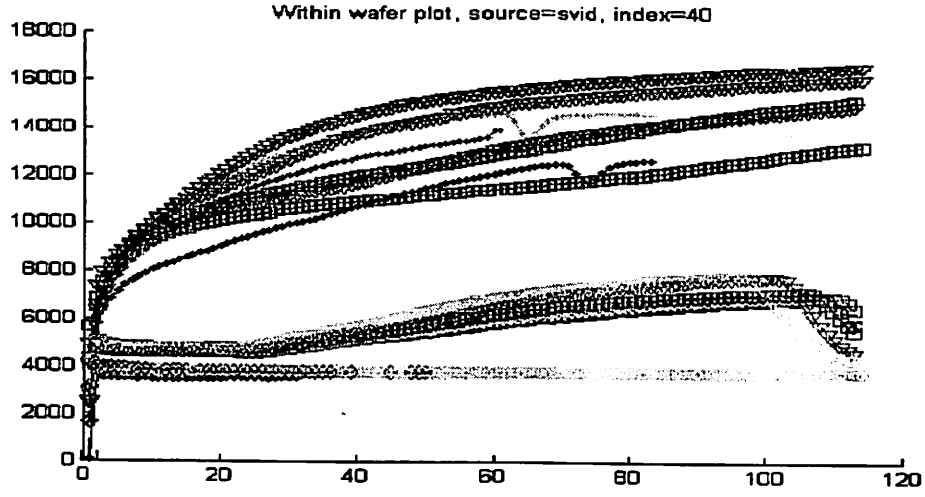


Figure 4.11. Windowed within-wafer plot for machine endpoint SiCl 405 nm.

Figure 4.11 shows the windowed within-wafer plot for machine endpoint SiCl 405 nm, for wafers processed with recipe 5003, from 23-May-2001 to 24-May-2001. The window is the entire step 5. It demonstrates that the wafer-to-wafer signal intensity falls into three clusters. Notice all the waveforms start around the same intensity at 4000, and go up with respect to time at different rate, or even remain constant. Probably the intensity level around 4000 corresponds to oxide etch, and for poly etch, there is usually a thin layer of native oxide on top of the silicon layer. What is shown here is very long etch processes of about 200 seconds (1.9 seconds per sample point). We can infer that the top cluster corresponds to bare silicon etch. The middle cluster has endpoints, so it must be poly etch. However, the uniformity of the poly film must be very bad, so that the oxide-poly etch transition is so slow. The bottom should correspond to oxide etch, since little intensity variation is observed. Also, we should take note the there are two small negative peaks in the top cluster. Previous researchers have not paid

attention to this kind of peaks, which can cause false alarms. Syntactic analysis can recognize them with ease. This will be demonstrated in detail in later chapters.

Figure 4.12 show the plots for wafers processed with Recipe 5001 from 30-May-2001 to 14-Jun-2001. The window is the entire main etch step, step 3. The plot demonstrates the typical chamber memory effect. On a wafer-to-wafer basis, the intensity starts relatively low, and then gradually go up to reach steady value after a few runs. Since the Berkeley Microlab is a research environment, there is no control or consistency over what type of wafers that are being etched. Film thickness varies considerably, as manifesting through the varying duration of the main etch step. Also, the etching film material can be drastically different, as revealed by the different clusters of wafer-to-wafer intensity level and the different etch waveform. Notice that some waveforms have an increasing trend, and some have a decreasing trend, but they all seem to stabilize over time.

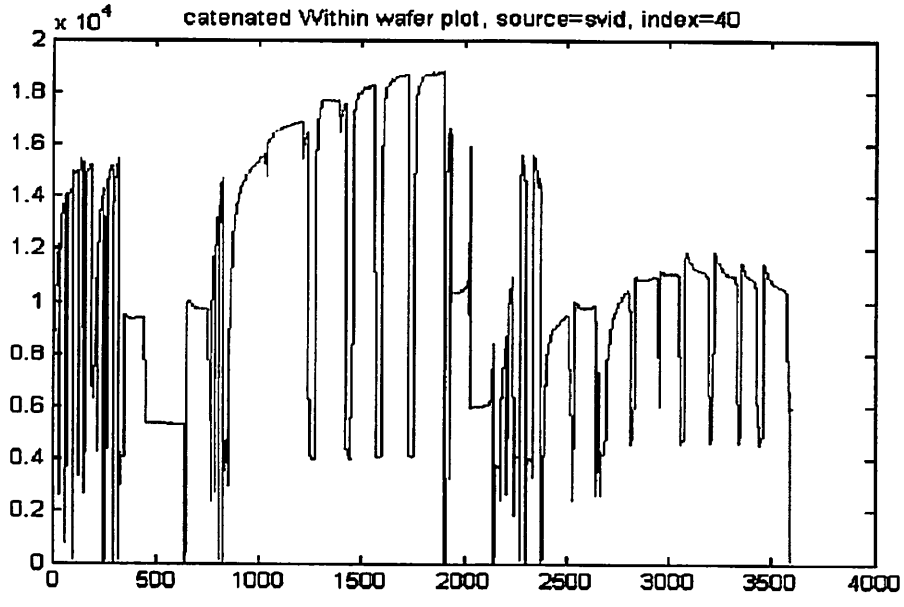


Figure 4.12. Concatenated windowed within-wafer plot for machine endpoint SiCl 405 nm.

Figure 4.13 shows the concatenated windowed plot for some wafer data processed with recipe 5001 around 01-Jun-2001. It demonstrates the chamber memory effect after an extreme event occurs. Although the intensity drops very close to the usual level, still, it takes a few runs for the intensity to stabilize.

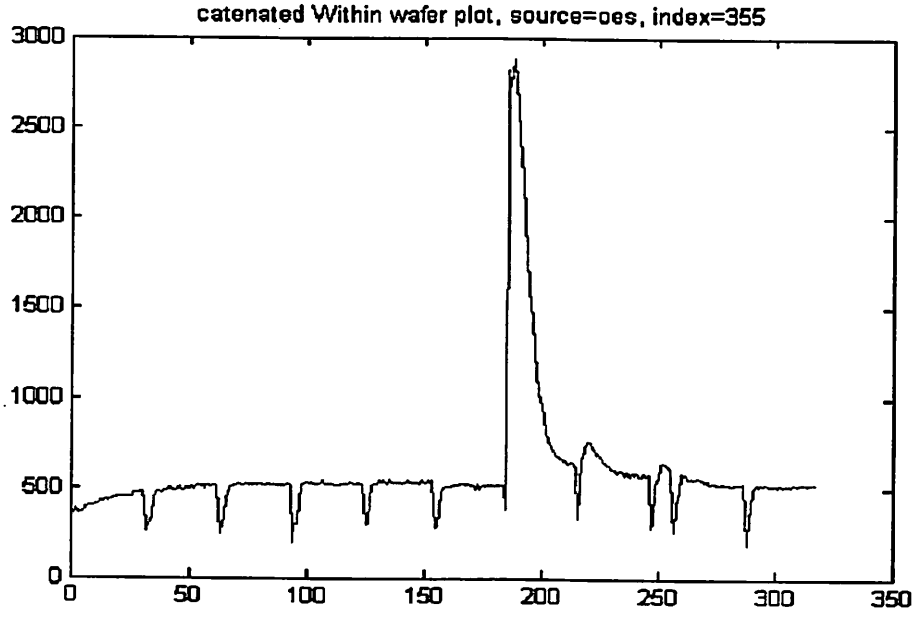


Figure 4.13. Concatenated windowed within-wafer plot for HBr 355 nm, demonstrating the chamber memory effect after the occurrence of a big spike.

Chapter 5

ENTPOINT DETECTION SENSITIVITY EXPERIMENT

5.1 Background

During an etching operation, when the target layer material is clear, the plasma should be stopped to minimized the overetch damage of the underneath layer. The clearing of the target layer typically signals the endpoint of the etch. Due to the non-uniformity of the etch rate and the thin film layer across the wafer, some overetch is often required to make sure that all the exposed area is fully etched.

The timed etch is an older approach for determining endpoint, for instance, [28]. The timed etch approach required pre-etch measurement of the film thickness. Once the wafer is fed into the etcher, the diagnostic system will try to predict the etch rate based on the real-time information, and thus obtain an etch time, which is equal to the measured film thickness divided by the estimated etch rate. The timed etch approach has a few pitfalls. First the pre-etch film thickness measurement may be time-consuming, and has be done manually. Second, non-uniformity in film thickness and etch rate across the wafer presents a difficulty in predicting the time required to fully clear the target film layer. This means that in order to ensure full clearance, additional overetch time would be required. Lastly, due to machine aging and drifting, the equipment and wafer states change over time, and as a result, the etch rate model may require constant update to ensure accurate etch time calculation.

The most popular approach for endpoint detection is to monitor the trace of emission from reactive species or volatile products using OES. Currently, most of these detection methods based on OES use a wavelength corresponding to the chemical species that show an obvious transition at the endpoint. For instance, one could monitor CO emission for oxide or polymer film etch, N_2 or CN for nitride film, SiF or SiCl for polysilicon film, AlCl for Al film. The LAM Research etchers in the Berkeley Microfabrication Laboratory, use the SiCl 405 nm for polysilicon etch endpointing, and the CO 520 nm for oxide etch endpointing. The user is allowed to specify the endpoint criterion on the signal, either based on transition amplitude or on transition slope.

5.2 The Endpoint Detection Experiment

As the lithography exposure area is shrunk down to less than one percent for contact and via etch, there is an increasing need for finding a wavelength with a pronounced endpoint transition. However, plasma etch is a rather complex chemical and physical process. The emission from reactive species or volatile products may not yield the most pronounced transition because they may not be the concentration limiting species for the etch process. Even if they are the concentration limiting species, overlapping bands can blur the transition.

In order to find the wavelength with the most pronounced transition, one needs to carry out a designed experiment. For the design of experiment, the exposure area used were 100%, 40%, 20%, 10%, 5%, 1%. The purpose of this sensitivity study is to determine the signal most sensitive to the endpoint transition for poly etch. That is, we want to be able to find out which signal (or combinations of signals) out of the thousands of available signals, can still show the endpoint transition when the exposure area is very small. Consequently, we choose not to use a contact or via mask for exposure, since non-uniformity in etch rate and film thickness within the wafer will blur the transition. We just do blanket exposure for each die. We know that the diameter of a wafer is 100 mm, the total area of the wafer is 2500π mm^2. And the side of a die is 10 mm, so the area for a die is 100 mm^2. Then we can figure out the number of dies we should have for different percentage of exposure area. For the 100% exposure area, we can just perform blanket etch. There is no photoresist required, and as a result, no patterning is necessary. For other percentage of exposure areas, on descending order, the numbers of dies for exposure are: 32, 16, 8, 4, and 1.

There was a data acquisition error for the 16-die (20% exposure area) wafer. This did not affect the experiment, because the endpoint transition is clearly visible for many signals for the 8-die (10% exposure are) wafer and below. Figure 5.1 shows the best transition from the ZSCAN signals, which is the second harmonic of the voltage reading. The transition is more or less clear for exposure of 32 dies and up. Yet, the stable values for the poly etch and oxide etch are not distinct. The endpoint transition

disappears for exposure of 8 dies and below. Due to their insensitivity, we see that the ZSCAN signals are not suitable for endpoint detection.

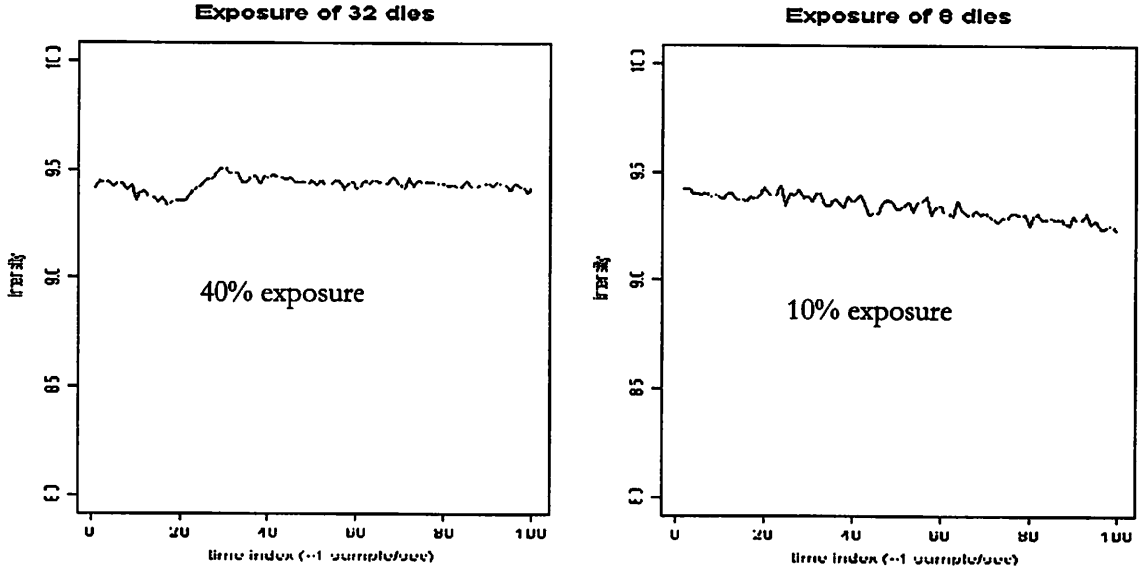


Figure 5.1. ZSCAN 2nd harmonic voltage endpoint plots.

Likewise, it is found that when the exposure area is large, the endpoint transition can be clearly seen from many OES wavelengths below 500 nm. As the exposure area is shrunk, the transition gets more obscure or disappears altogether. It is observed that the best signals for endpoint detection are two CF_2 OES lines, 275 nm and 321 nm. These are the only two signals that the transition is still clearly seen when the exposure area is down to 1%.

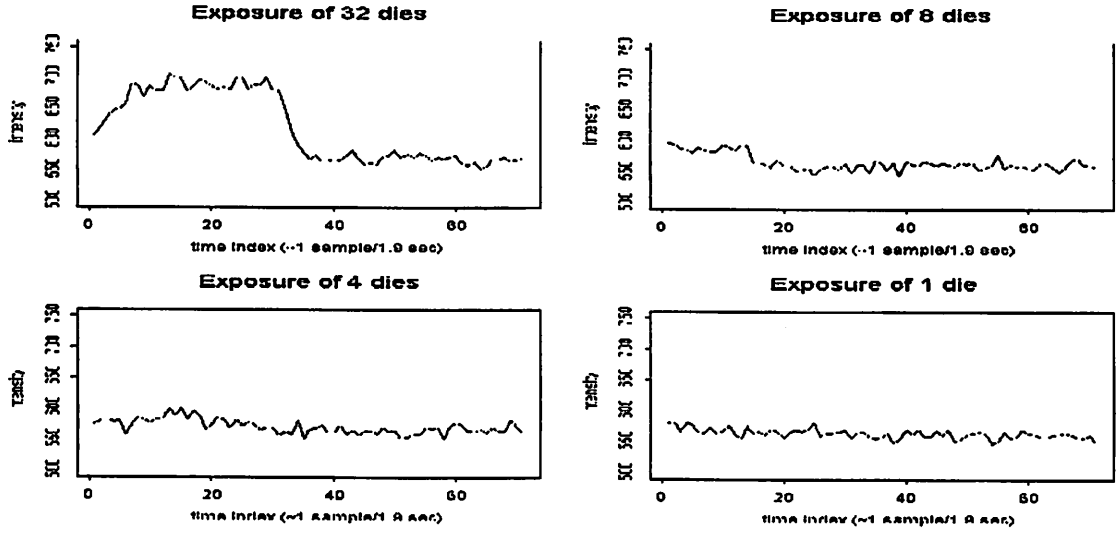


Figure 5.2. The OES endpoint traces of Lam 9400 built-in endpoint detection wavelength SiCl 405nm for different exposure areas.

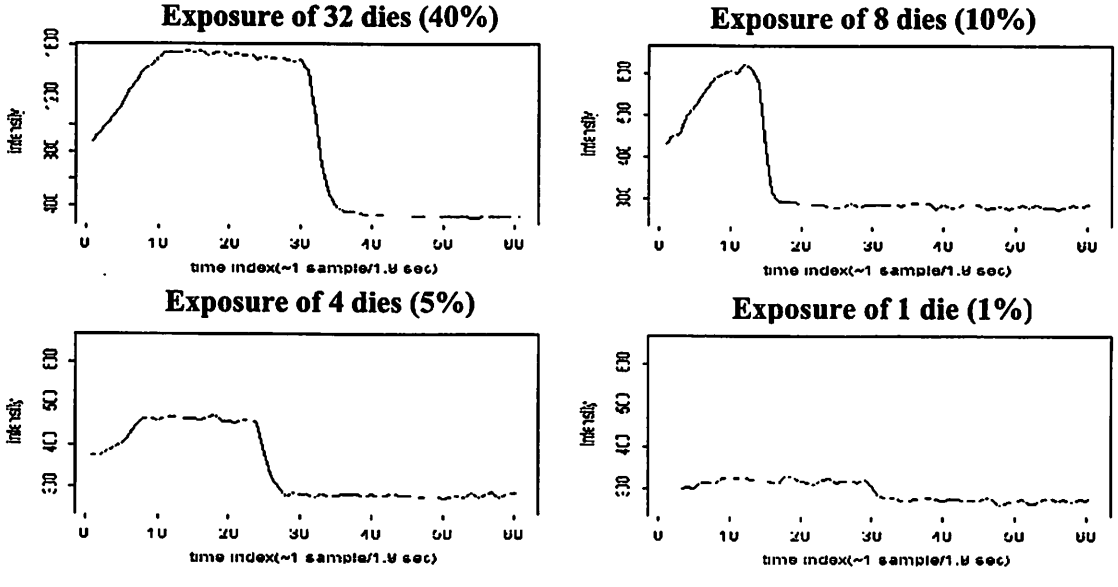


Figure 5.3. The OES endpoint traces of the CF_2 275nm line for different exposure areas.

Figure 5.3 shows the endpoints plots for CF_2 275 nm. One thing to notice is that the signal intensity corresponding to the oxide stays more or less constant for different exposure percentages, and that the signal intensity corresponding to poly etch varies linearly with respect to the amount of exposure area. A comparison for the SiCl 405 nm, which is the LAM 9400 wavelength for poly etch endpoint detection, is shown in Figure 5.2. The transition is more or less obscure for 8-die exposure already, and it is very difficult to locate the transition for the 4-die exposure.

At the time of the endpoint sensitivity experiment, the built-in endpoint signal (SiCl 405 nm) was not included in the SVID data acquisition list. However, we just need to show that the ratio between the signal intensity for poly etch and oxide etch value is greater for the CF_2 wavelengths than that of the built-in endpoint signal. Figures 5.4, 5.5, show some endpoint traces of CF_2 321 nm and the built-in endpoint signal for some wafer runs on 5/28/2001. Also, notice that the waveform of the built-in endpoint signal is much "cleaner" than the CF_2 321 nm wavelength. This is due to the LAM built-in endpoint sensor's aperture is much bigger than that of the OES sensor, and thus achieve much better signal-to-noise ratio for the stable etch region. In Figure 5.4, we see that the poly-versus-oxide ratio for CF_2 is about 1.8, and the ratio for the built-in endpoint signal is about 1.2, as shown in Figure 5.5. This shows that the CF_2 signals are much better endpoint detection signals than the built-in.

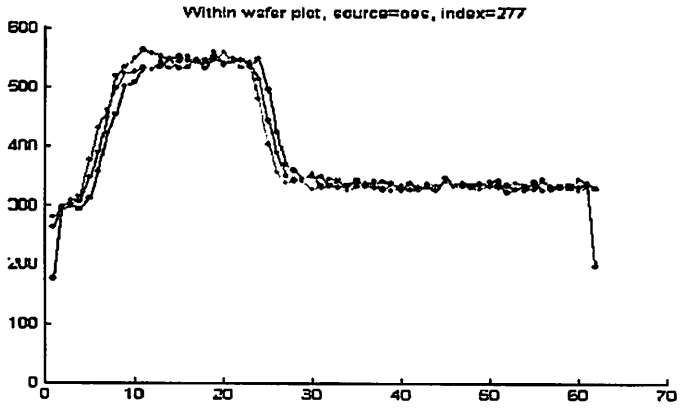


Figure 5.4. The OES endpoint traces of the CF_2 321 nm line for a fixed exposure area.

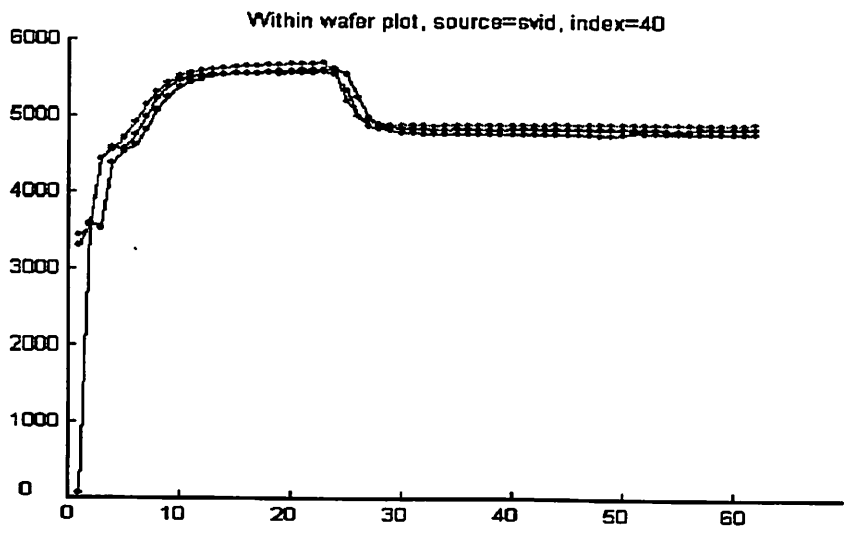


Figure 5.5. The LAM 9400 SVID built-in endpoint traces of the SiCl 405 nm line for a fixed exposure area.

One should take note that this is a poly etch process, which uses Cl_2 gas as the etchant. There is often pre-poly etch step which etches away the native oxide on the top of the poly layer, using CF_4 gas. The CF_2 particles in the plasma during poly etch probably come from the CF_x particles deposited on the chamber walls. The creation mechanism for the CF_2p articles is subject to future investigation. Also, for the future works, we should perform verification runs to assess the endpoint detection accuracy using the CF_2 signals. Currently, for the sensitivity study, the sampling rate is one sample for every 1.9 seconds. The sampling rate should be increased to about 5 Hz, as to minimize unnecessary overetch. Lithography exposure should be done with a via or contact mask with less than 1 percent exposure area. The detection algorithm can be developed using syntactic analysis, which will be illustrated in later chapters.

5.3 Criticism of other OES approaches for endpoint detection

While other researchers have been focusing on using multiple wavelengths for performing endpoint detection, the author believes that the single-wavelength approach should be sufficient for the detection, at least for the polysilicon etch chemistry in question. Other researchers have, one way or the other, avoided searching for the best single wavelength. For very small exposure areas, it would be

very surprising if no single wavelength could provide obvious endpoint transition, and yet, the combination of multiple wavelengths could. It is worthwhile examining some examples of multiple-wavelength approaches in the literature.

White, et al [27] perform PCA on the full OES spectrum on the main etch region. As wafers being processed, and new data coming in, they compute the T^2 value for the combination of the first few principal components. When the T^2 value is abnormally large, that signifies the endpoint is being reached. They apply this approach on SiO_2 etch, varying the lithography exposure area, 100%, 10%, and 1%. They plot T^2 value vs. time for each exposure area, and compare the intensity by plot vs. time for a dominant wavelength (CO 520 nm). They demonstrate that the endpoint transition for T^2 value plot is much more pronounced than that of the dominant wavelength plot. However, White et al did not address the issue of what could be the best dominant wavelength, i.e., the wavelength with the most distinct transition at the endpoint.

The first few principal components capture most of the variance of the entire spectrum due to the highly correlated nature of the OES signals. And usually, the wavelengths with significant endpoint information contain a lot of variation, so that they are heavily weighed in the first few principal components. As the full spectrum approach can work well relatively large exposure area (>10%). However, when the exposure area is very small (<1%), there will be very few wavelengths with significant endpoint transition. As demonstrated in our work, for polysilicon etch with 1% exposure area, only two wavelengths, CF_2 275 nm and 321 nm, contain significant endpoint transition. The full spectrum approach would not yield distinct endpoint transition for small open area.

Yue, et al [29] proposed two steps in selecting wavelengths before performing PCA. First, they divide up the full spectrum into several windows, and use PCA to step through those windows, and then they keep the windows with obvious endpoint transition and discard the ones without. Then recursively, they subdivide up the remaining windows into smaller ones, and further remove irrelevant wavelengths. Next, they use a "sphere" criterion to select the most relevant wavelengths from the remaining from step one. Some principal components with obvious endpoint transition are selected. They examine the sum of square for the loading coefficients in the selected principal components for each wavelength. A few wavelengths with the largest sum of square are finally chosen.

Yue et al's approach substantially reduces the number of wavelengths used for endpoint detection, and thus reduces computation cost and improves model consistency. Yet, they did not do an exhaustive search for the best wavelength for endpoint detection. During the first pass of the wavelength elimination procedure, it is likely that a good endpoint detection signal may fall in one of the window with largely irrelevant signals, so that the PCA could not pick out the transition.

Chapter 6

EQUIPMENT STATE AND WAFER STATE PREDICTION EXPERIMENTS

6.1 Introduction

The purpose of the equipment state experiment is to assess the sensitivity of the sensors to fluctuations of equipment parameters such as power, chamber pressure, gas flow rate, etc. The wafer state prediction experiment is to determine whether the data from the three sensor sources provide significant diagnostic information for predicting wafer states, such as etch rate and uniformity. In other words, the two experiments are used to determine if the combination of the three sources of sensor information can more accurately characterize the equipment and wafer states. If this is not the case, it is not necessary to have all the sensors in place, and thus we can save some hardware cost, disk space, and analysis time.

6.2 Equipment State Experiment

For the equipment state experiment, five parameters are under consideration, HBr flow rate, Cl_2 flow rate, chamber pressure, RF top power, RF bottom power. We vary the machine settings of the parameters, one at a time, and then try to see if the sensors signals fluctuate accordingly. We know that the LAM etcher provides built-in real-time monitoring for the five parameters. The user of the etcher can get both the settings and the actual readings through the SECS II interface. For the equipment state information, the source of machine signals through the SECS II interface is indispensable. Therefore, we only need to assess on the OES and ZSCAN sources. Since we are only interested in the equipment states for this experiment, we want to minimize plasma etching reaction in the chamber. Ideally, we would power up and ignite the plasma without any wafer in the chamber. However, under normal operation, the etcher will not ignite a plasma without wafers. So, we opted to use oxide wafers into the chamber. Since we are using a poly etch recipe with high selectivity to oxide, this diminishes etching when the plasma is ignited.

Each parameter is in turn varied around the center. First, we vary it in an increasing fashion, then vary it away from the center point (see Table 6.2). In this way, we can classify the chamber memory effect, if there is any, during the wafer runs.

Parameter	Center point
HBr flow rate	150 sccm
Cl_2 flow rate	50 sccm
Chamber pressure	12 mTorr
RF top power	300 W
RF bottom power	150 W

Table 6.1. Center point of the equipment machine settings.

Parameter	Run #	Run Sequence
HBr	1	130, 140, 150, 160, 170
	2	150, 140, 160, 130, 170
Cl_2	1	25, 37.5, 50, 62.5, 70
	2	50, 37.5, 62.5, 25, 70
Pressure	1	10, 11, 12, 13, 14
	2	12, 11, 13, 10, 14
RF top	1	250, 275, 300, 325, 350

	2	300, 275, 325, 250, 350
RF bottom	1	130, 140, 150, 160, 170
	2	150, 140, 160, 130, 170

Table 6.2. The run sequence of the equipment state experiment. One parameter is changed at a time; others remain at the center points.

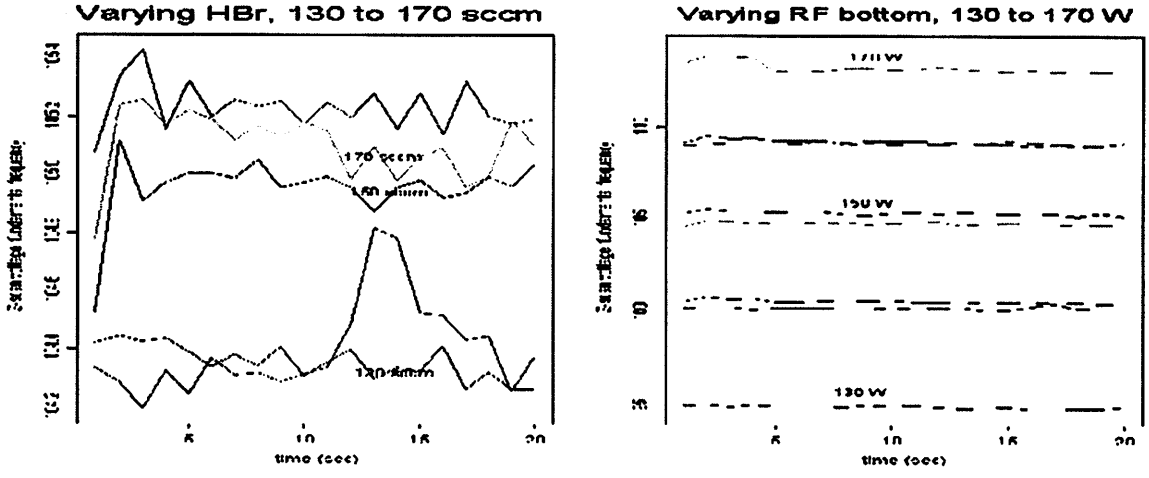


Figure 6.1. Two examples of time series plots of the ZSCAN signals for different machine settings.

All the ZSCAN current, voltage, and phase values on time series are examined. Figure 6.1 shows two examples. From them, we see that the fundamental voltage reading is in linear relationship with the RF bottom power, but it is not sensitive to HBr flow rate deviation. It is found that ZSCAN current, voltage, is sensitive to the setting deviation of chamber pressure, RF top and bottom power, but not sensitive to the deviation of the gas flow rates. The phase reading appears random to any parameter deviation.

For the OES spectrum, wavelengths corresponding to peaks are selected for examination. As shown in Figure 6.2, the intensity value is the average of a window of five sample points. From the plots, we see that OES 797 nm intensity varies linearly with HBr gas flow rate, but varies randomly with RF bottom power. And after examining all the OES peaks, we find out that OES signals are sensitive to all but the RF bottom power. That suggests that RF bottom power may not play much of a role in plasma reaction.

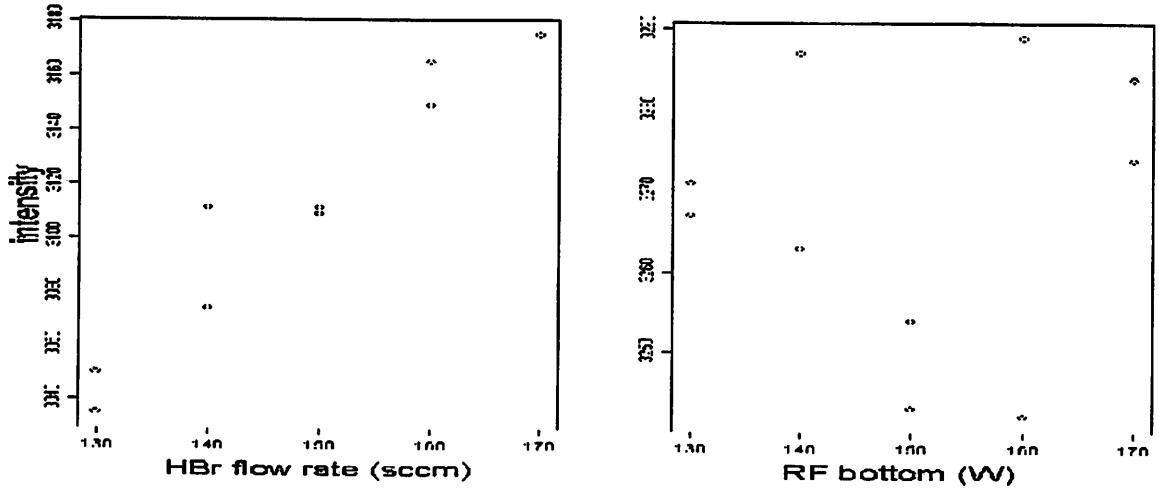


Figure 6.2. Two intensity vs. machine setting plots for OES 797 nm.

Table 6.3 summarizes the qualitative sensitivity study result, demonstrating that OES and ZSCAN are complementary in equipment state modeling.

	HBr	Cl2	Pressure	RF top	RF bottom
ZSCAN	No	Slightly	Yes	Yes	Yes
OES	Yes	Yes	Yes	Strongly	No

Table 6.3. Equipment state sensitivity study summary.

6.3 Wafer State Prediction Experiment

Accurate inline wafer state prediction can reduce the need for costly and time-consuming wafer measurement. The wafer states that we are interested in are etch rate and uniformity. Five machine settings are varied to achieve different etch rate and uniformity on the wafer. The center points and positive and negative deviation of the settings are shown on table 6.4. However, no center point wafer run is needed because machine settings will not be used as terms in wafer state modeling. We only use the three sources of signals. OES, ZSCAN and SVID signals, as the modeling terms. A two level, 2^{5-1} resolution V experiment with I=ABCDE, is designed as shown in Table 6.5. In order to un-confound the blocking effect, a randomization is performed on the running sequence. The randomized sequence is 10, 7, 12, 4, 16, 1, 3, 14, 6, 8, 11, 15, 13, 9, 5, 2 (The numbers are in the first column of the table).

	+	-	Center point
RF top (A)	350	250	300
RF bottom (B)	180	120	150
Cl_2 (C)	75	25	50
HBr (D)	180	120	150
Pressure (E)	14	10	12

Table 6.4. Parameter level assignments.

Run #	A	B	C	D	E
1	+	+	+	+	+
2	+	+	+	-	-
3	+	+	-	+	-
4	+	+	-	-	+
5	+	-	+	+	-
6	+	-	+	-	+
7	+	-	-	+	+
8	+	-	-	-	-
9	-	+	+	+	-

10	-	+	+	-	+
11	-	+	-	+	+
12	-	+	-	-	-
13	-	-	+	+	+
14	-	-	+	-	-
15	-	-	-	+	-
16	-	-	-	-	+

Table 6.5. The design table for the wafer state prediction experiment.

Poly-on-oxide wafers are used. The thickness of undoped poly film is about 3500 angstrom, and the thickness of the oxide film is about 300 angstrom. The wafers are set for 30-second poly etch in the LAM 9400 etcher, which has an etch rate of about 2500 angstrom/min, under the center point settings as shown in Table 6.4. A CMOS gate mask is used for lithography exposure. The GCAW stepper is programmed to perform 32-die exposure runs. Figure 6.3 shows a wafer map. The ones in bold are selected for thickness measurement. For each die, five locations, four in the corner, one in the center, are selected for measurement, as shown in Figure 6.4.

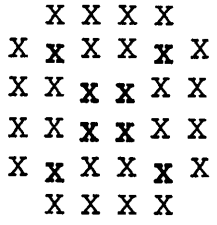


Figure 6.3. Wafer die map. The ones in bold are selected for measurement.

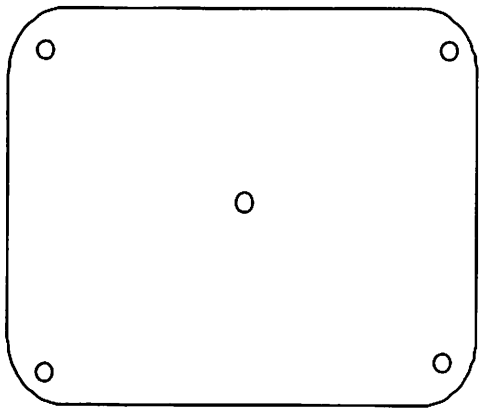


Figure 6.4. The locations selected for thickness measurement within a die.

The CMOS gate mask image on the die is used for finding the central location of the die. For each measurement location, four points in proximity are selected for film thickness measurement by Nanoduv. Two points are in the exposed etched regions; the other two are in the unexposed and unetched regions. Then the etched depth is the sum of the measurements for the unetched regions minus the sum of the measurements for the etched regions, divided by 2. Nanoduv performs the measurement by shining a laser beam onto the wafer surface. The two reflected beams, from the poly and oxide film respectively, interfere with each other, and a sensor captures the interference pattern and calculates the path difference of the two beams, and thus obtaining the film thickness. Then,

Etch rate = average etch depth / .5 [angstrom/min]

Uniformity=(max etch rate – min etch rate)/average etch rate

=(max etch depth – min etch depth)/average etch depth

Once we compute the wafer states and acquire the sensor signal data, we can start to build models from them.

Similar to their endpoint detection work, White, et al [26] have performed PCA on the entire OES spectrum. While PCA can extract most variance from the spectrum using a few principal components, including irrelevant signals will degrade the significance of the model. Draper (1964), Jeffers (1967), and Mansfield (1977) have considered how to select variables based on the PCA loading values of the variables. In order to test the assumption that irrelevant signals degrade the model significance, several stepwise linear regression schemes are deployed.

1) Perform PCA on all signals, and treat all PCs are individual variables, and perform stepwise linear regression on them.

2) Perform PCA on OES signals only, and treat the principal components (PCs), the 43 SVID signals, and 10 ZSCAN signals as individual variables, and perform stepwise linear regression on them.

3) Without performing PCA, treat 2048 OES, 43 SVID, and 10 ZSCAN signals as individual variables, and perform stepwise linear regression on them.

As for the stepwise linear regression, we perform the follow steps,

1) Start with the signal most correlating with the wafer state.

2) Add the next signal which reduce the model prediction error (Cp, consult [48] p216 for definition and explanation) the most.

3) From the current added set of variables, if dropping a variable can reduce the prediction error, remove the variable.

4) Stop if no more variable can be added to increase the model significance ([48], p18) OR no more variable can be added to reduce the model prediction error.

From Table 6.6, we can see that just performing PCA alone yields a better R^2 value than performing PCA on all the signals. This suggests that the irrelevant signals of SVID and ZSCAN in the grand PCA (PCA on all signals) deteriorate the wafer model significance. Due to the large number of OES

signals (2048), comparing to the number of SVID (43), and ZSCAN (10) signals, this deterioration is rather slight. The R^2 value goes from .84 to .81 for the etch rate model, from .90 to .89 for the uniformity model. By the same token, if we just do linear stepwise regression on all the signals, we notice that the model significance improvement is quite substantial; the R^2 value goes from .81 to .99 for the etch rate model, from .89 to .96 for the uniformity model. Yue, et al [30] have attempted to select relevant signals first before applying PCA for wafer state modeling using OES signals. The author takes note of their method and intention. However, with the R^2 values being .96 and .99, which are very close to 1, the margin for improving the model significance is very slim. In other words, even if we gain improvement from additional signal screenings, we cannot really determine if this improvement is statistically significant. The author is well-aware that some schemes of signal selection in addition to PCA might be worth examining for other problems with smaller R^2 values. Nevertheless, for our case, we have a large number of signals for the stepwise regression, giving us very significant terms in the model. We consider that by using the simple linear stepwise regression, the modeling result for this current problem is satisfactory. Besides, the purpose of this work is not to search for the best modeling method for problems with a large number of variables, in a theoretical sense. Rather, we just want to demonstrate that all three sources of signals contribute significant diagnostic information.

Signal source	Wafer state	Significant terms	R^2 (adjusted)
PCA on all signals	ER	PC 1, PC 3	.8122 (.7833)
	U	PC 1, PC 6	.888 (.8708)
PCA on OES alone	ER	PC 1, PC 3	.8383 (.8134)
	U	PC 1, PC 5, I5	.8998 (.8748)
Stepwise Regression on all signals	ER	355 nm, 207 nm, I4, RF load coil pos	.9876 (0.9821)
	U	997 nm, 559 nm, 1049 nm, 974 nm	.9576 (.9422)

Table 6.6. Summary of wafer state modeling, comparing with or without PCA. Note: I5, I4 are the current readings of the fifth and fourth harmonics

of the plasma frequency, respectively. ER = etch rate; U = uniformity; PC = principal component.

Table 6.7 shows the simple linear stepwise regression results on each source alone, and then on all three sources combined. We notice that ZSCAN signals are not sensitive to the wafer states. Probably, this is the reason RF sensors have not been used widely for plasma diagnostics. The popularity of OES sensors is justified since OES model are significantly better than SVID models. As for the stepwise regression on the combination of all sources, we should notice that while uniformity model only takes terms from the OES signals, the etch rate model takes terms from all three sources, and improve slightly, and yet significantly from the OES signals alone. This demonstrates that all three signal sources contribute useful diagnostic information.

Signal source	Wafer state	Significant terms	R2 (adjusted)
OES	ER	355nm Br, 520nm CO, 359nm CN	.9475 (.9394)
	U	997 nm, 559 nm, 1049 nm, 974 nm	.9576 (.9422)
SVID	ER	TCP load cap, Cl2, RF line imp #1	.9082 (.8941)
	U	TCP load cap, HBr stpt, cham press	.8548 (.8325)
ZSCAN	ER	I5	.217 (.161)
	U	V1, I1	.4858 (.4067)
Stepwise Regression on all signals	ER	355 nm, 207 nm, I4, RF load coil pos	.9876 (.9821)
	U	997 nm, 559 nm, 1049 nm, 974 nm	.9576 (.9422)

Table 6.7. Summary of wafer state modeling, comparing different sources.

Chapter 7

DATA EXPLORATION WITH SYNTACTIC ANALYSIS

7.1 Introduction

Syntactic analysis refers to a general pattern recognition technique, which uses formal language paradigms to describe the structure of an object. The basic approach is to decompose the object into sub-patterns of primitives. By some criteria, a symbol is assigned to each primitive, and the symbols are assembled into a sentence. A grammar is a set of syntactic rules for generating sentences, which describes a class of objects. If the sentence encoded from an object is accepted by the grammar, then we consider that the object belongs to the class described by the grammar. Syntactic analysis is widely used for character recognition, especially in the Far East, where syntactic analysis-based Chinese character recognition is an active research area.

Syntactic analysis also has found some success in the medical field, for analyzing electrocardiogram (ECG) signals, in order to determine the status of a patient's heart. If done visually, the procedure is divided into two stages [10]. First, some characteristic features of ECG are recognized, such as the P wave, the PQ segment, the QRS complex, the ST segment, the T wave and the TP segment. Then, the physician measures the features' parameters, such as durations and amplitudes, and interprets these numerical values based on experience and a set of established empirical diagnostic criteria. Due to the massive amount of ECG data, there has been a great interest for computerizing the interpretation process. Many medical researchers have used syntactic pattern recognition techniques to analyze ECG signals [1-5]. The objective is to build an ECG processing system to imitate the physician, and to draw similar judgments about the status of the patient.

ECG signals are similar to the plasma etch signals in some respects. In addition to considerable amounts of noise, their form and size can change over time [3]. Also, like etching signals differing from machine to machine, ECG signals differ considerably from person to person. Syntactic analysis is applicable since it is robust against gross change, and also appeals to intuition. Even if a signal has

been "rubber stretched"(i.e. linearly transformed along the x- and y- axes), if the signal is classifiable by a human expert, then syntactic analysis can usually classify it correctly. For these reasons we think that syntactic analysis holds considerable promise analyzing plasma etching signals.

7.2 A qualitative description of the basic etch waveform

Chapter 2 and Chapter 4 have described the nature of plasma etch signals, and the related analytical difficulties. As pointed out previously, syntactic analysis is able to ignore extraneous influences on the waveforms, and offers great flexibility in capturing both the qualitative and quantitative dynamics of the signals.

Often, for a basic poly etch waveform, we identify the poly etch segment, the oxide etch segment, and the transition between the segments. However, a few variations of the waveform exist. At the beginning and the end of the waveform, there can be power-on and –off transitions, respectively. Sometimes, there is even a stable power-off segment. There is often a thin layer of native oxide on top of the poly film, so we may see an oxide-to-poly transition at the beginning of an etch waveform. If the layer of native oxide is thick enough, there might be a stable oxide etch region before a stable poly etch region. Also, the oxide etch segment may not be clearly defined, and the transition may not be complete. If it is a timed etch and the etch is not through, there will be only a poly etch segment. There will be just a silicon etch segment if it is a dummy run with a bare silicon wafer. There will be just an oxide etch segment if it is a dummy run with an oxide wafer.

The data collected for our analysis comes from a development-oriented process. As a result, users may use masks of various exposure areas, poly think films of different doping concentrations. Different masks will lead the etch signal intensity to fall into distinctly different clusters even for the same material. The different doping concentrations of poly thin films will generate a spread within the cluster. At times, users put wafers with aluminum films into the etcher, and cause intense reaction in the chamber, and the intensity of the etch signal may overshoot the calibration limit.

7.3 Waveform encoding & waveform query

In order to perform qualitative and quantitative data exploration, we need to encode the etch waveforms first. As shown in Figure 7.1, we want to divide a etch signal waveform into stable etch (e.g., poly or oxide etch), and transition (e.g., poly-oxide, or oxide-poly transition) primitives. A stable etch primitive is encoded with "eX", where "e" stands for "etch," and "X" is the material (for Poly, for Oxide, for Aluminum, etc.) symbol. We will discuss in detail how to assign the material symbol later. A transition primitive is encoded as "tYZ", where "t" is for "transition," "Y" is the transition-from material symbol, and "Z" is the transition-to material symbol. Before we divide a waveform into primitives, we do a linear piecewise approximation with an error tolerance of 30 (intensity units).

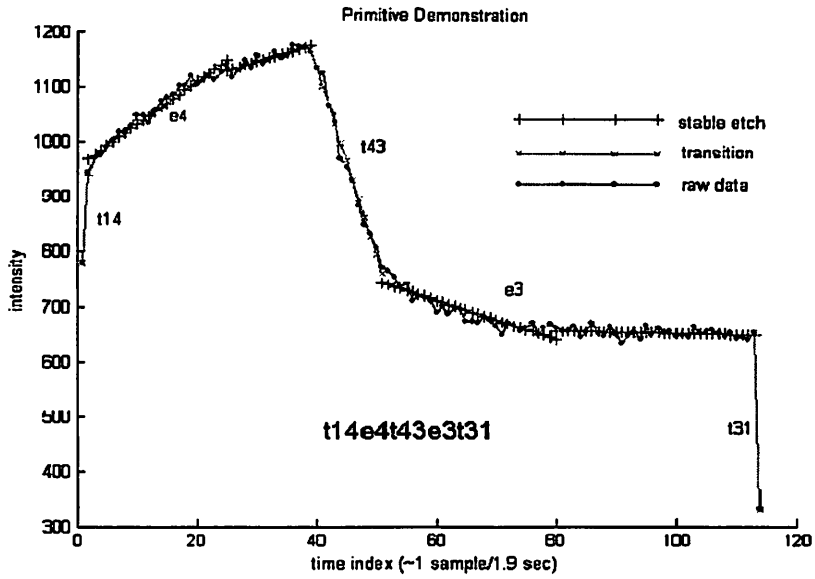


Figure 7.1. Demonstration of the primitives on a typical etch waveform.

The following shows a linear piecewise approximation algorithm. The segment starts from the first data points. The segment keeps growing with successive points until the maximum linear fitting error is greater than the tolerance. The second segment starts with the end of the first segment, and this procedure is repeated until the entire window is represented by linear segments.

Segmentation
Input: Time series $X=\{x_1 \ldots x_n\}$; linear fit error tolerance ε.
Output: List of line segment $L=\{s_1 \ldots s_m\}$.
h=1, i=1, j=1;
j=j+1;

```
while (j<n)
      s=linear model fit on {x_i ... x_j };
      maxerror = max {prediction error of s};
      if (maxerror > ε)
            s=linear model fit on {x_i ... x_{j-1} };
            L=append(L, s);
            i=j-1, j=i+1;
      else j=j+1;
L=append(L, s);
return L;
```

The running time of the above algorithm is n^2. There is a massive amount of data, so we have an interest to speed up the computation. Notice that once we approximate the data points with line segments, the amount of data we need to analyze will be substantially reduced. A faster (n*log n) algorithm for line segmentation is presented next. The basic difference is that, instead of growing the segment point by point, the algorithm grows a line segment exponentially with data points if the linear fit tolerance is not exceeded. Likewise, once it detects that the linear fit tolerance is exceeded, it shrinks the segment in an exponential manner.

Sped-up Segmentation

Input: Time series $X=\{x_1 \ldots x_n\}$; linear fit error tolerance ε.

Output: List of line segment $L=\{s_1 \ldots s_m\}$.

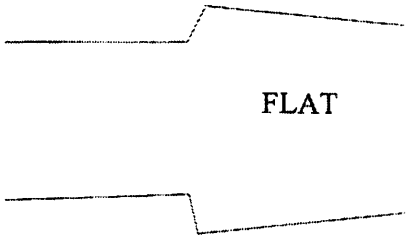
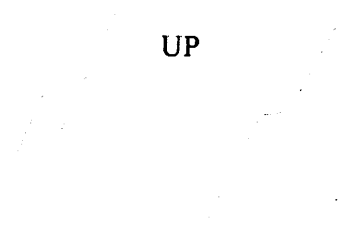
```
k=1; h=1; i=1;
j=i+k;
almostfullidx=0;
while j<=n
   s=linear model fit on {x_i ... x_j };
   maxerror = max {prediction error of s};
  while maxerror< ε
       k=k*2;
       j=i+k;
       exceeded=0;
       if j>n %reaching the end of the curve
            almostfullidx=i+k/2;
            j=n;
            s=linear model fit on {x_i ... x_j};
            maxerror = max {prediction error of s};
            exceeded=1;
            break;
        else
            s=linear model fit on {x_i ... x_j};
```

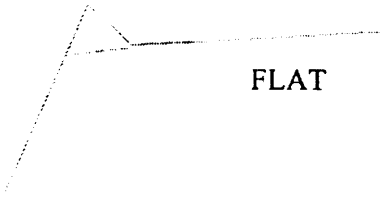
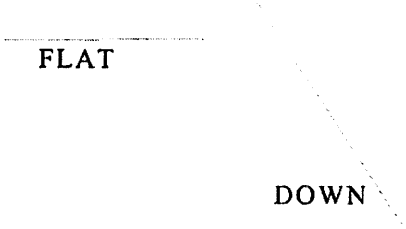
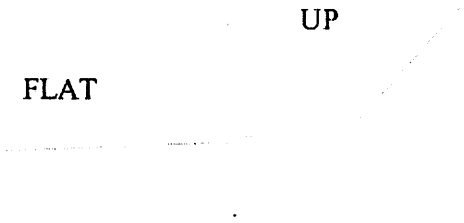
```
            maxerror = max {prediction error of s};

    if maxerror< ε %for break out of reaching the end of the curve
        L=append(L,s);
         break;
    else %adjsize computation
        adjsize=0;
    if j==n AND exceeded
        adjsize=n - almostfullidx;
    else
        adjsize=k/2;
    j=j-adjsize;
    s=linear model fit on {x_i ... x_j};
    maxerror = max {prediction error of s};

    while adjsize>1
        adjsize=floor(adjsize/2);
        if(maxerror<e)
            j=j+adjsize;
        else
            j=j-adjsize;
        s=linear model fit on {x_i ... x_j};
        maxerror = max {prediction error of s};

    adjsize=1;
    if maxerror>e
        j=j-adjsize;
        s=linear model fit on {x_i ... x_j};
    L=append(L,s);
    h=h+1; i=j; k=1; j=j+k; %next segment, notice k reinitialized to 1
return L;
```

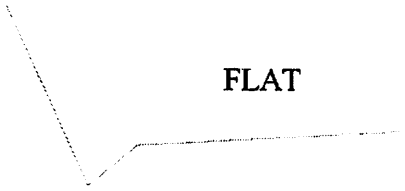
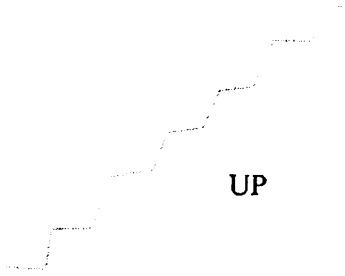
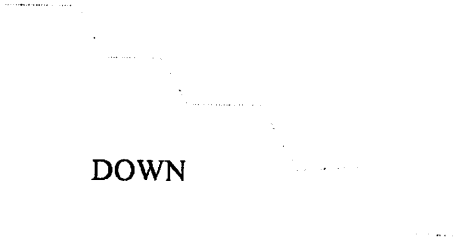
After these computations we group the line segments together based on the slope values. If the absolute slope is less than 20 per sample, we consider the line segments as a flat stable etch region, assigning a slope code of 0. If the slope values are greater than 20, we consider the line segments as an up-transition region, assigning a slope code of 1. If the slope values are less than –20, we consider the line segments as a down-transition, assigning a slope code of –1.

However, due to the irregularity of the signal, this first pass segmentation might lead to many noisy small line segments. Small line segments are those with amplitude smaller than 80, and duration

smaller than or equal to 4 sampling points. Next we show how to group them with bigger line segments or how to filter them out. Table 7.1 lists many of the scenarios that can occur due to noisy segments, and the action we take for each case.

Waveform	Rules Description
 FLAT	A noisy segment lies between two long FLAT segments. We consider all three segments as a FLAT primitive candidate. The noisy segment can be UP or DOWN.
 UP	A noisy segment lies between two big UP segments. We consider all three segments as an UP primitive candidate. The noisy segment can be DOWN or FLAT.

	A DOWN noisy segment lies between a UP segment and a FLAT segment. A horizontal line is drawn through the noisy region, and the region and the FLAT segment are treated as one FLAT primitive candidate.
	A UP noisy segment lies between a FLAT segment and a DOWN segment. A horizontal line is drawn through the noisy region, and the region and the FLAT segment are treated as one FLAT primitive candidate.
	A DOWN noisy segment lies between a FLAT segment and a UP segment. A horizontal line is drawn through the noisy region and the region and the FLAT segment arte treated as one FLAT primitive candidate.

	A UP noisy segment lies between a DOWN segment and a FLAT segment. A horizontal line is drawn through the noisy region, and the region and the FLAT segment are treated as one FLAT primitive candidate.
	Noisy segments occur consecutively, alternating between UP and FLAT primitives. We consider the entire region as UP.
	Noisy segments occur consecutively, alternating between DOWN and FLAT primitives. We consider the entire region as DOWN.

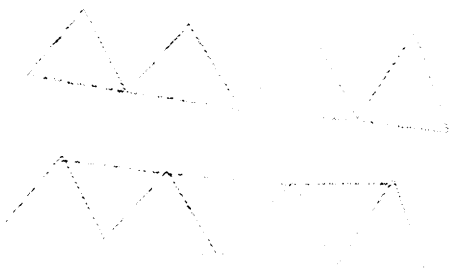
	Noisy segments occur consecutively, alternating between UP and DOWN primitives. We use lines to connect bottoms or tops of the segments. The slope of those lines defines the slope attribute for the region.

Table 7.1. Rules for processing noisy segments.

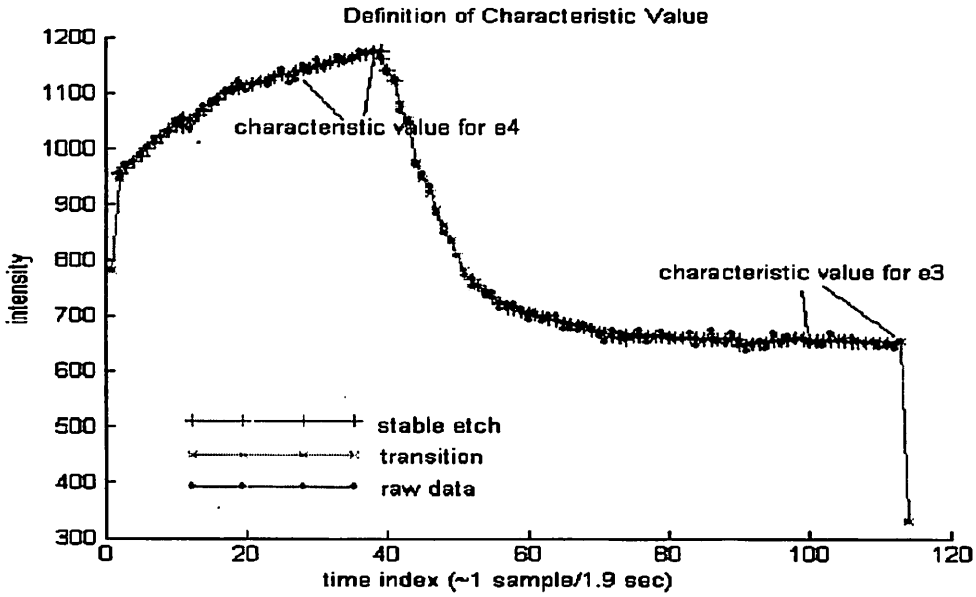


Figure 7.2. Showing how to get the characteristic value on a stable etch region. For illustration purposes, the segmentation criteria are different from the one in figure 7.1.

The numerical attribute for coding the symbol for a certain etching material is the characteristic value of the stable etch waveform. Figure 7.2 shows a typical stable etch waveform, which first increases

rapidly, then stabilizes and reaches a steady state. Due to various reasons, the transient time until reaching steady state can vary considerably. We decided to use the steady state value as the characteristic value of the etch waveform. As we have done the piecewise linear approximation on the waveform already, we select the flattest "significantly long" line segment, and take its average value as the characteristic value. The flattest line segment is the one with the smallest absolute slope value. A "significantly long" line segment is one with more than four samples. Since the sampling rate is about 1.9 sec/sample, the duration of the line segment needs to be greater than 7.6 seconds to be significant.

The reason we want the characteristic line segment to be reasonably long is that there are times when there is a small spike with very short duration in the etch waveform. At the peak of the small spike, the absolute slope value is very close to zero. If we do not require the characteristic line segment to be reasonably long, the peak value of the small spike could be mistaken as the steady etch value. If somehow, due to a noisy waveform, there is no significantly long line segment, then we just choose the longest line segment as the characteristic line segment because when the waveform reaches a steady state, the line segments tend to get longer.

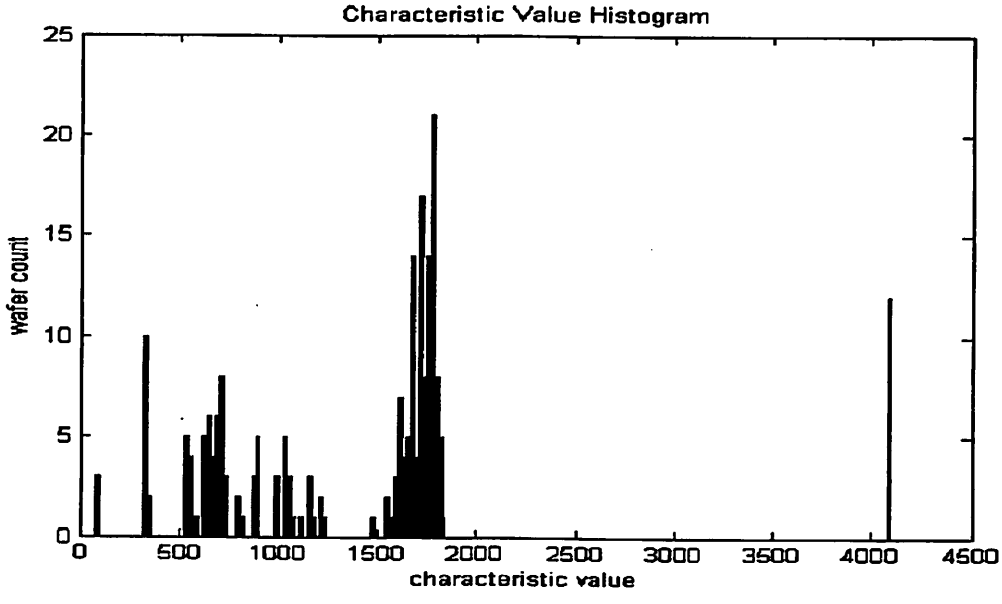


Figure 7.3. The distribution of the characteristic values of stable etch regions for OES 321nm CF_2 line.

Figure 7.3 shows a characteristic value histogram. Table 7.2 shows the code number assignments for different material. On the lower end we have the power-off state. The diagnostic system usually stops sampling when the power is off, so for a power on-off transition, i.e., t1Z or tY1, the sensor may not be able to see the power-off intensity. Nevertheless, we can interpret extremely sharp and large transitions as power on-off transitions. There are two clusters for oxide and poly, respectively, possibly due to the different doping concentration of the material or the different exposure area. Also, there is an overshoot cluster on the high end.

Material Symbol	Range	Etch segment
1	0~200	Power-off
2	200~400	Oxide I
3	400~800	Oxide II
4	800~1400	Poly I
5	1400~2000	Poly II
6	3800~4500	Aluminum

Table 7.2. Material symbol assignment table.

For qualitative data exploration, we often use a search string to test the existence of a portion of the etch waveform, such as the existence of poly etch, and endpoint transition. Once the encoding for the etch waveform is in place, it is saved to the archive with the wafer ID. The user can retrieve the wafer data files as needed. For instance, to focus on poly II etch only, he/she can use "*e5*" as the search string. Figure 7.4 shows some within-wafer plots for the wafers extracted using "*e5*" as the search string. Alternatively, to study wafer runs with poly I-to-oxide II endpoint transition, the appropriate search string will be *e4t43e4*. Notice that the codes for the two stable etches are included in the search string. Then, the user can use the software features, such as within wafer plot, wafer-to-wafer plot, and signal-vs.-signal plot, signal correlation computation, to study the various qualitative and quantitative data properties. Figure 7.5 shows a wafer-to-wafer plot for the wafer runs with poly I to oxide II endpoint transition.

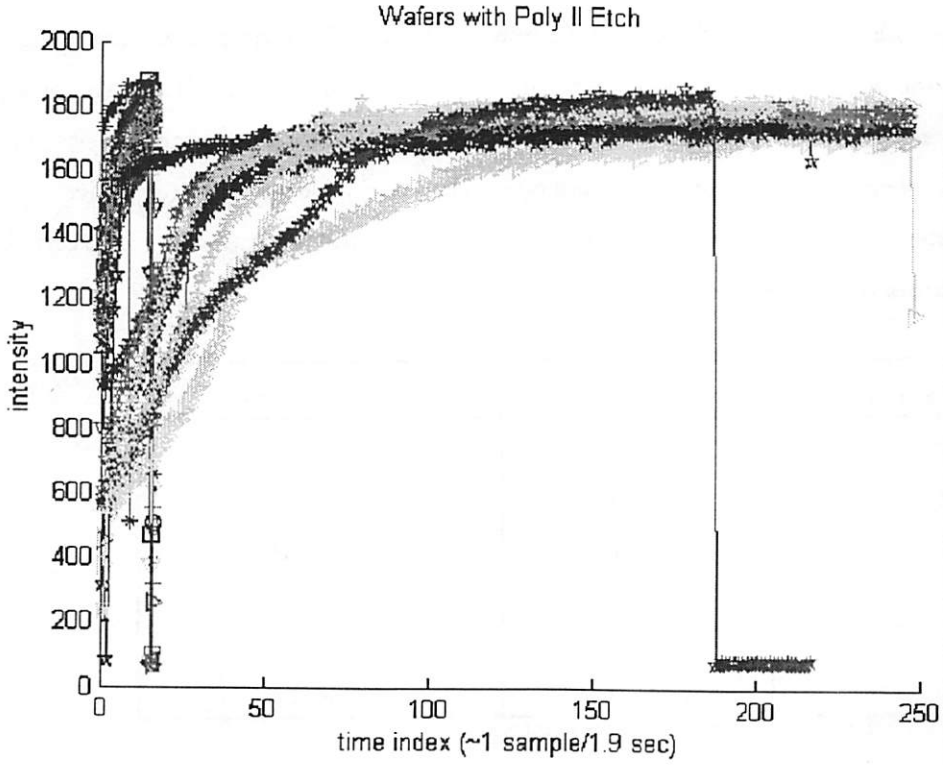


Figure 7.4. Within-wafer plot of runs with poly II etch.

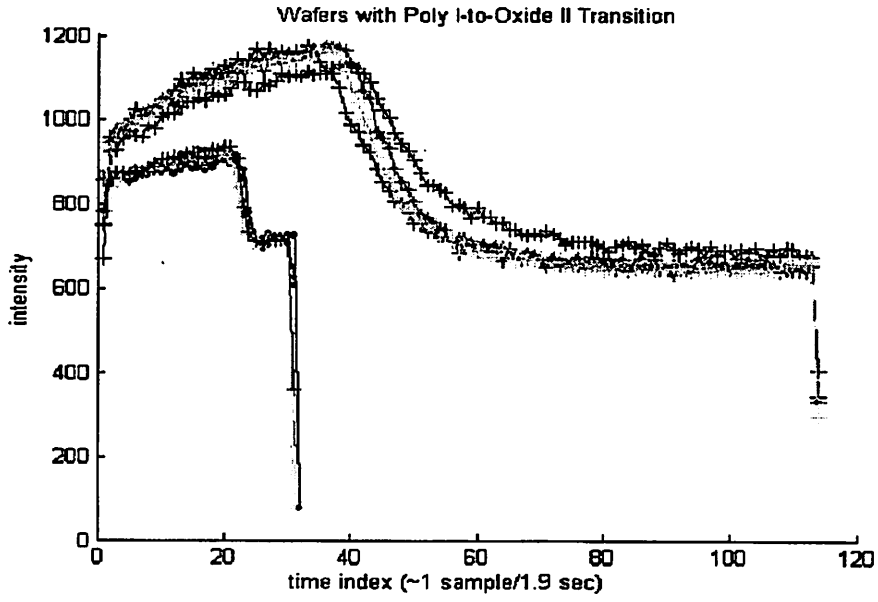


Figure 7.5. Within-wafer plot of runs with poly I to oxide II transitions.

Notice in Figure 7.4 for the poly II etch wafer extraction, the wafers can fall into two clusters based on the duration attribute of the poly II etch. Also, in figure 7.5 for the wafers with endpoint transitions, they can be group into two clusters based on the poly etch duration, etch characteristic value, transition amplitude, or transition duration. In order to further categorize the wafer data, we need to use some quantitative attributes as the criteria.

For quantitative encoding of the stable etch primitive, we are interested in the characteristic value, amplitude, and duration; and for transition primitives, we are interested in the nominal transition amplitude, the actual transition amplitude, as well as the duration. The nominal transition amplitude is the amplitude due to segmentation, which takes the difference between the beginning and terminating points of the transition primitive. The actual transition amplitude is the difference between characteristic values of two stable etch regions (see Figure 7.6). If one of the stable etch regions does not exist, the difference between the characteristic value and one end of the primitive. A power-on or power-off usually does not have a stable region. Often, stable oxide etch region does not exist, since once the endpoint transition is detected, the machine will stop the main etch. If due to some operational errors, neither stable etch region exist, then the actual transition amplitude is equal to the

nominal transition amplitude. If there is a thin native oxide on the top of the poly film, the power-on transition will not have any adjacent stable etch region, because the native oxide etch will exhibit an oxide-to-poly transition. Also, if the native oxide etch is interrupted, then the oxide-to-poly transition will not have any adjacent stable etch region.

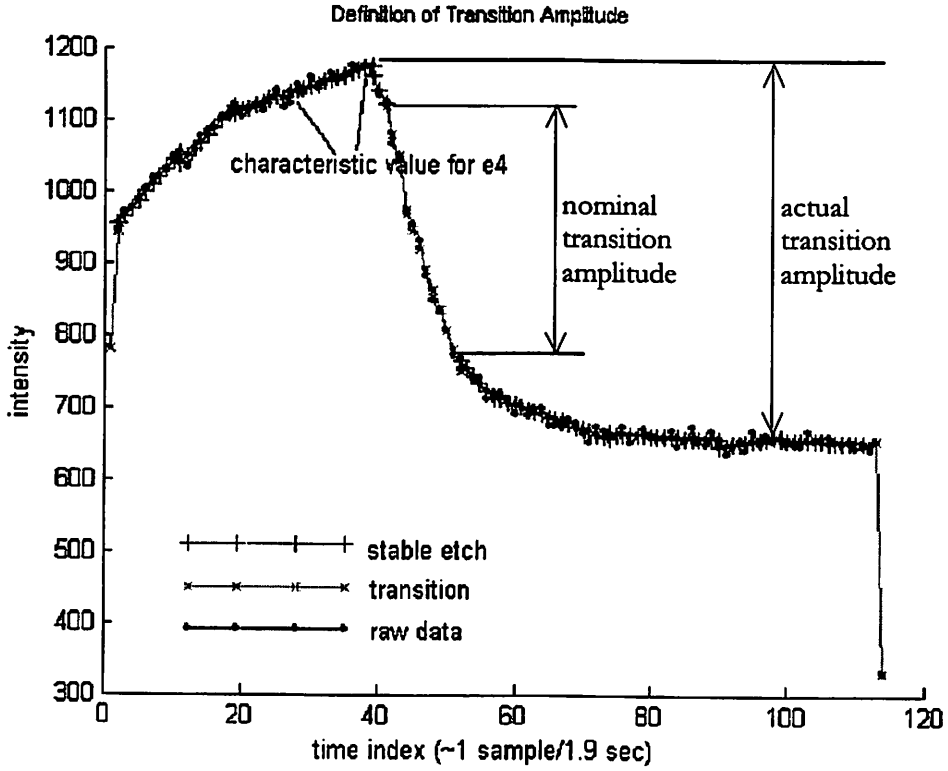


Figure 7.6. Definition of nominal and actual transition amplitude.

Figure 7.7 shows the extraction result based on the poly I etch characteristic value from 1000 to 1200; and Figure 7.8 shows the result based on the characteristic value from 800 to 1000. Likewise, Figure 7.9 shows the extraction result based on the poly II etch duration from 50 to 400; and Figure 7.10 shows the result based on the duration from 1 to 50. We can see that with these numerical criteria we can resolve the respective clusters from the visual inspection of the wafer plot. Lastly, we can do graphical plots, and correlation computations on these numerical attributes. Figure 7.11 shows the distribution plot for the poly II etch characteristic value for the cluster with shorter etch duration.

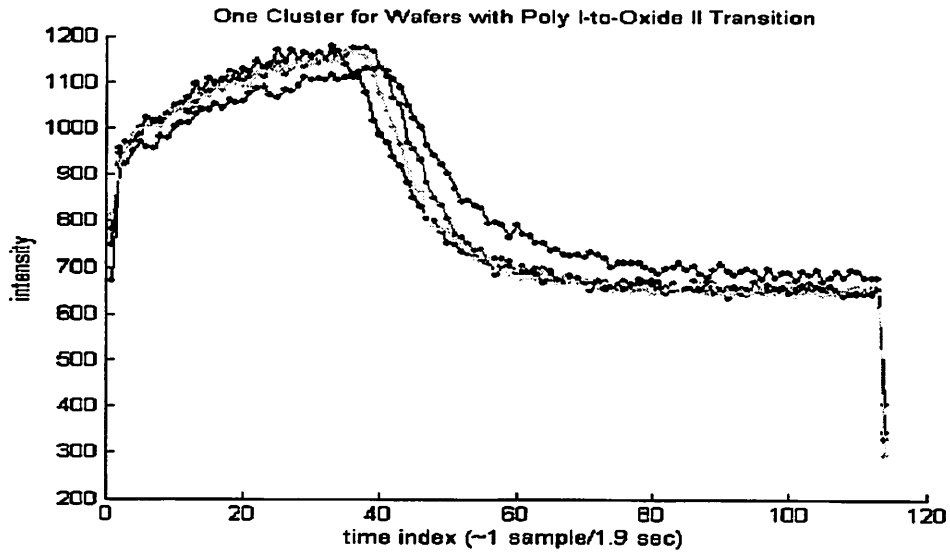


Figure 7.7. The cluster of within-wafer plot of runs with poly I to oxide II transitions for poly I etch characteristic value from 1000 to 1200.

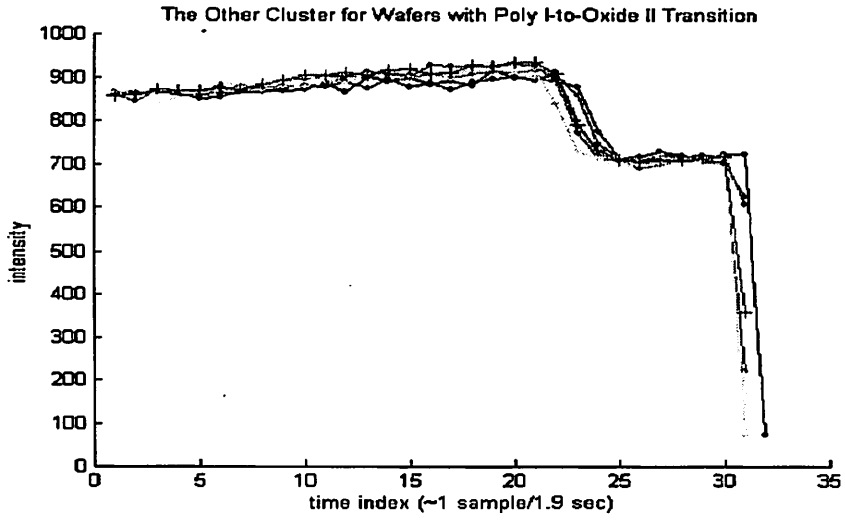


Figure 7.8. The other cluster of within-wafer plot of runs with poly I to oxide II transitions for poly I etch characteristic value from 800 to 1000.

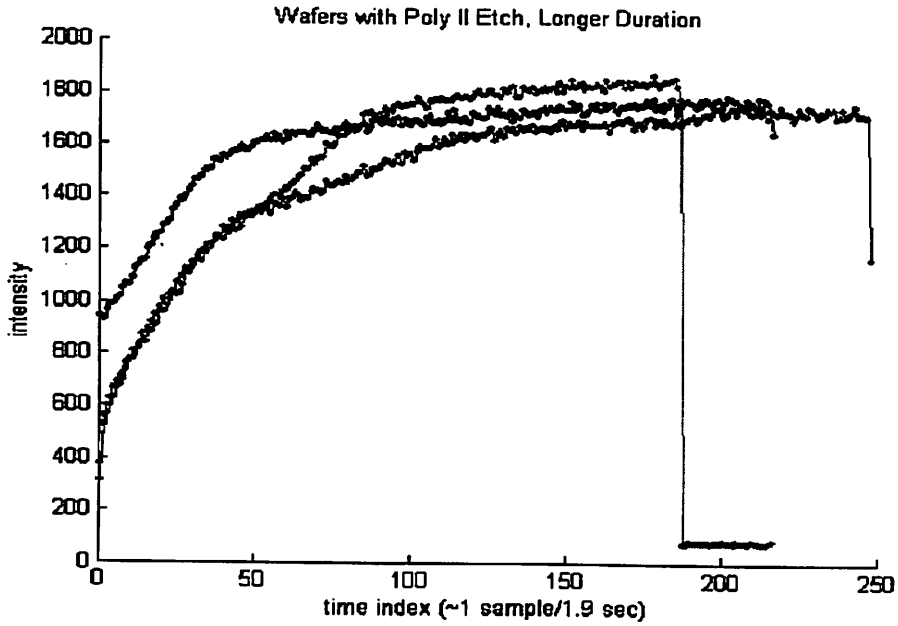


Figure 7.9. The cluster of within-wafer plot of runs with poly II etch duration from 50 to 400.

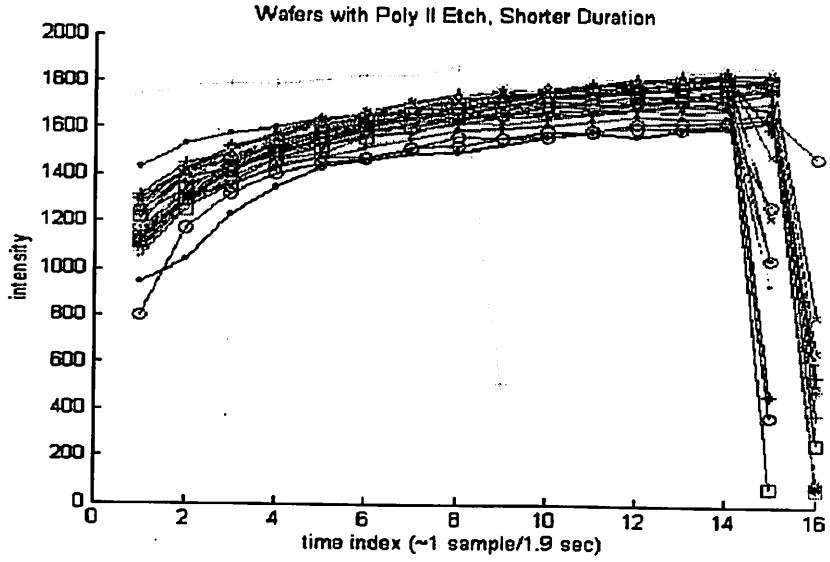


Figure 7.10. The cluster of within-wafer plot of runs with poly II etch duration from 1 to 50.

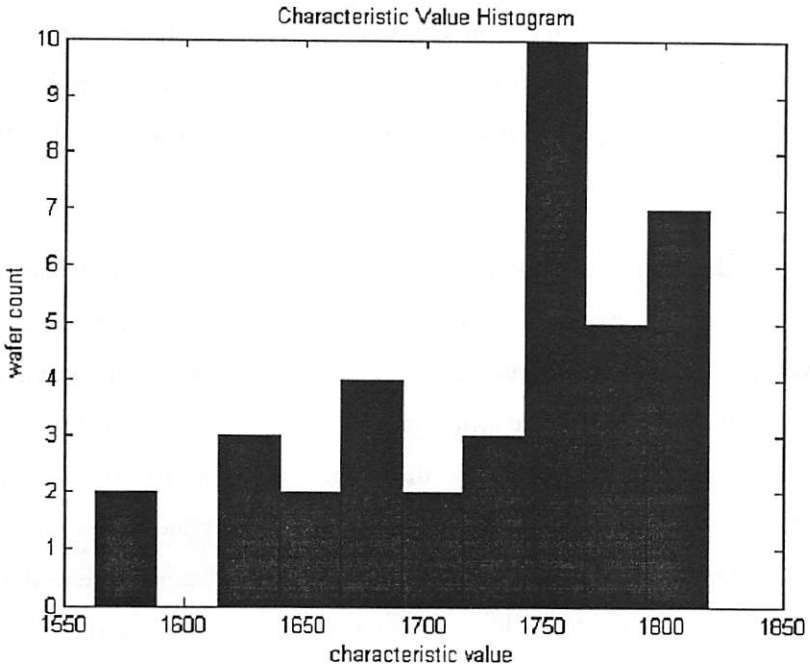


Figure 7.11. The distribution plot for the poly II etch characteristic value for the cluster with shorter etch duration.

7.4 Designing the Syntactic System

Let us examine the issue of how to come up with syntactic rules and parameters for a diagnostic system. For the syntactic rule in this chapter, we use the stable etch and transition primitives to describe the etch waveform. To further characterize the primitives, we extract numerical attributes, such as characteristic value, duration, and amplitude from them. The designed parameters include the criteria for defining the material symbol, the linear tolerance for the first pass segmentation, the slope threshold for defining the stable etch and transition primitives, the small duration and the small amplitude that define noisy line segments.

The author realizes the advantages in completely or partially automating the design process. Complete automation might be possible for simple diagnostic problems. This might be possible, for example, if

we can classify objects correctly using some easily measurable attributes, such as weight, temperature, volume, length, width, and etc. It might also be possible to automate the process of deciding what attributes to use. However, complete automation for plasma diagnostics appears to be impractical due to the complexity of the etch waveform. Complete automation implies coming up with syntactic primitives and parameters without analyzing the etch waveform. The task is equivalent to describing a complicated object without seeing the object.

If we lay down the syntactic rules first, and limit the automated search to just the values of the appropriate parameters, the task is often feasible. Let us use the poly I-to-oxide II transition in Figure 7.5 as an example. Here we will use the syntactic rules from previous work and we are searching for parameters that conform the etch waveforms to the syntactic rules. Just for the investigation, we keep the material symbol criteria constant, and perform a three-point iteration for the rest of the parameters, varying each one by plus/minus 25% of its default center point value. The center points are 30, 4, 20, 80 for linear tolerance, small duration, slope threshold, and small amplitude respectively. Table 7.3 shows all the combinations that conform the etch waveforms to the syntactic rules.

Linear Tolerance	Small Duration	Slope Threshold	Small Amplitude
23	3	15	60
23	3	20	60
23	4	15	60
23	4	20	60
23	5	15	60
23	5	20	60
30	3	15	60
30	3	15	80
30	3	20	60
30	3	20	80
30	4	15	60
30	4	15	80
30	4	20	60
30	4	20	80
38	3	15	60
38	3	20	60
38	4	15	60
38	4	20	60
38	5	15	60

38	5	20	60

Table 7.3. The combinations of parameters that conform the poly I-to-oxide II etch waveforms to the syntactic rules.

This iterative way of searching for parameter is rather computationally intensive, on the order of $m*n^k$, where m is the number of designing wafers; n is the number of parameters; k is the number of points for iteration. In addition, selecting a good combination out of the table remains a problem. A good combination will yield small diagnostic error rates. Automating this choice would require the appropriate quantification of the error rate criterion. Here, the author believes that the human expert's judgment is irreplaceable in making the selection. One should use the combinations one at a time to perform segmentation on some raw etch waveforms (other than the designing etch waveforms), and inspect the segmentation result visually. A good segmentation result usually has the following visual properties:

1) The number of line segment is reasonably small.

2) The line segments approximate the etch waveform reasonably well.

3) The stable etch and transition primitives make sense visually.

However, once the designer performs a visual inspection, the benefit of automation is lost. The designer is better off to just manually vary the parameters and then visually inspect the result. Table 7.4 shows the segmentation result by varying the slope threshold. We see that when the threshold gets to 20 or above, what visually appears to a transition region can segmented to be a stable etch primitive. A slope threshold of 10 or 15 will be acceptable.

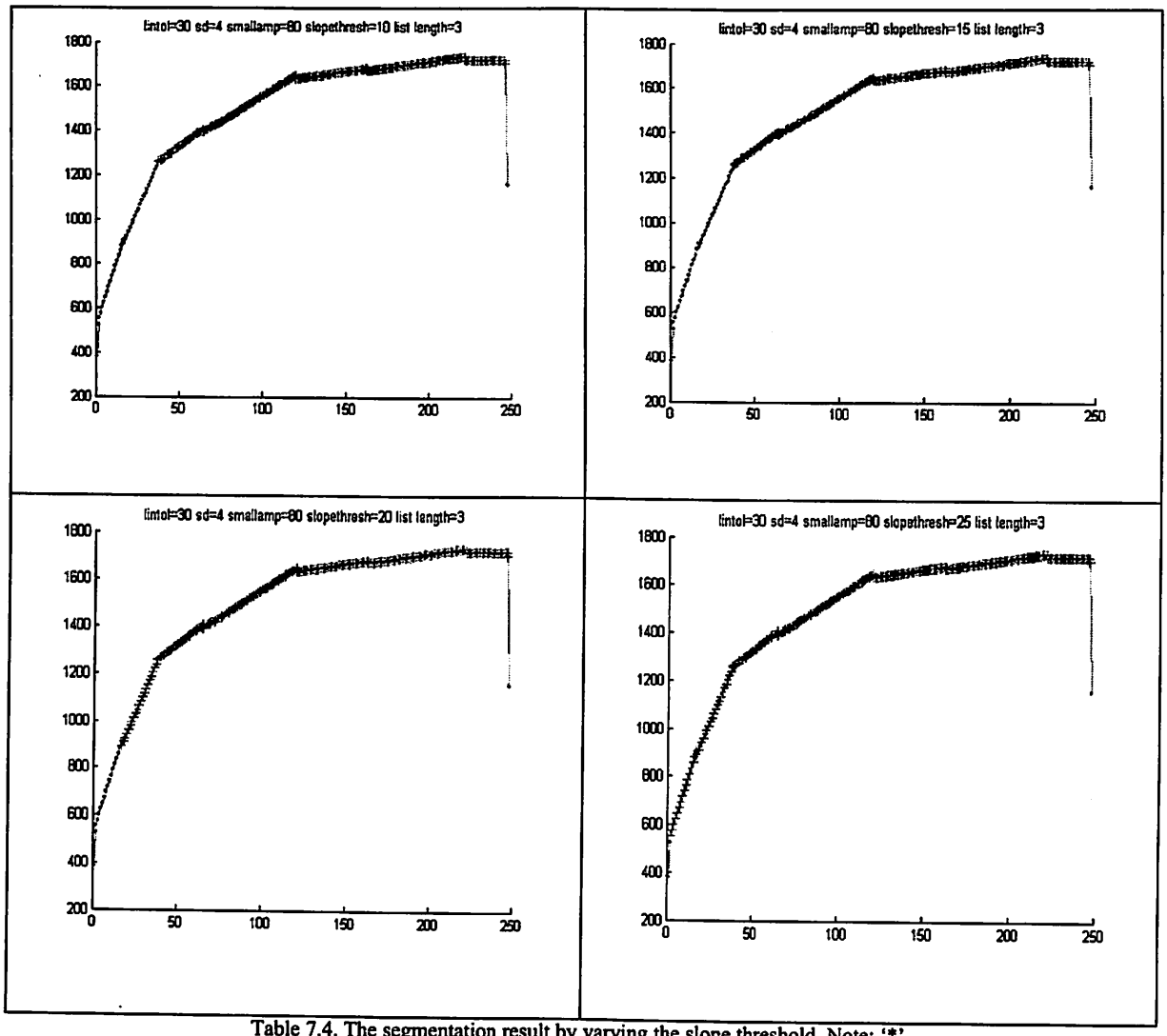
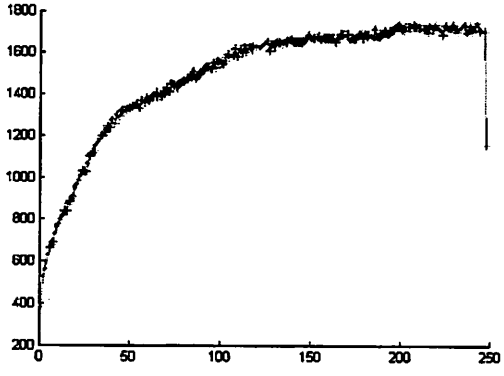
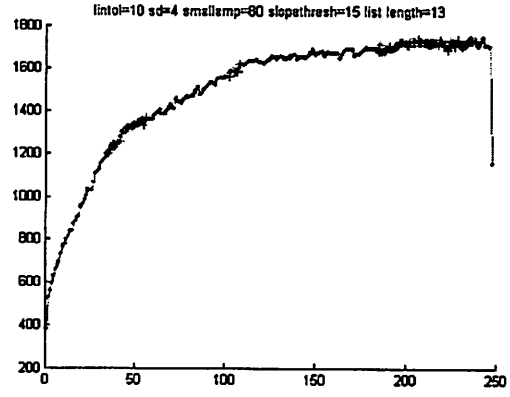
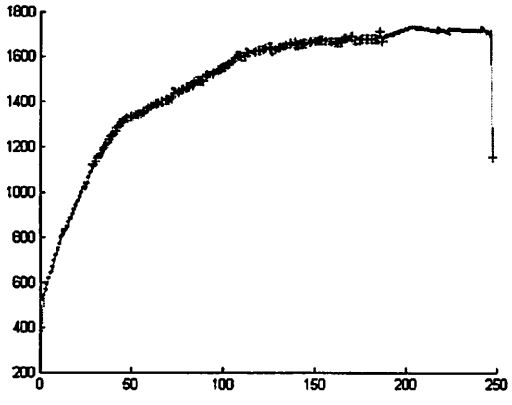
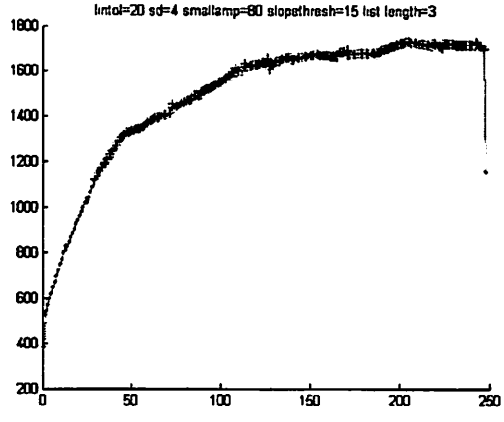


Table 7.4. The segmentation result by varying the slope threshold. Note: '*' = transition, '+' = stable etch. Lintol = linear tolerance, sd = small duration, smallamp = small amplitude, slopethresh =slope threshold.

Table 7.5 shows the segmentation result by varying the linear tolerance. When the tolerance is 10, the segmentation result is wrong even after filtering out any noisy segments. When it is 15, the segmentation result is correct after noisy segment filtering for this sample. However, there are too many line segments, which will induce error in new samples. It is better to use 20 or slightly above for linear tolerance.

Before noisy line segment filtering	After noisy line segment filtering
	
	

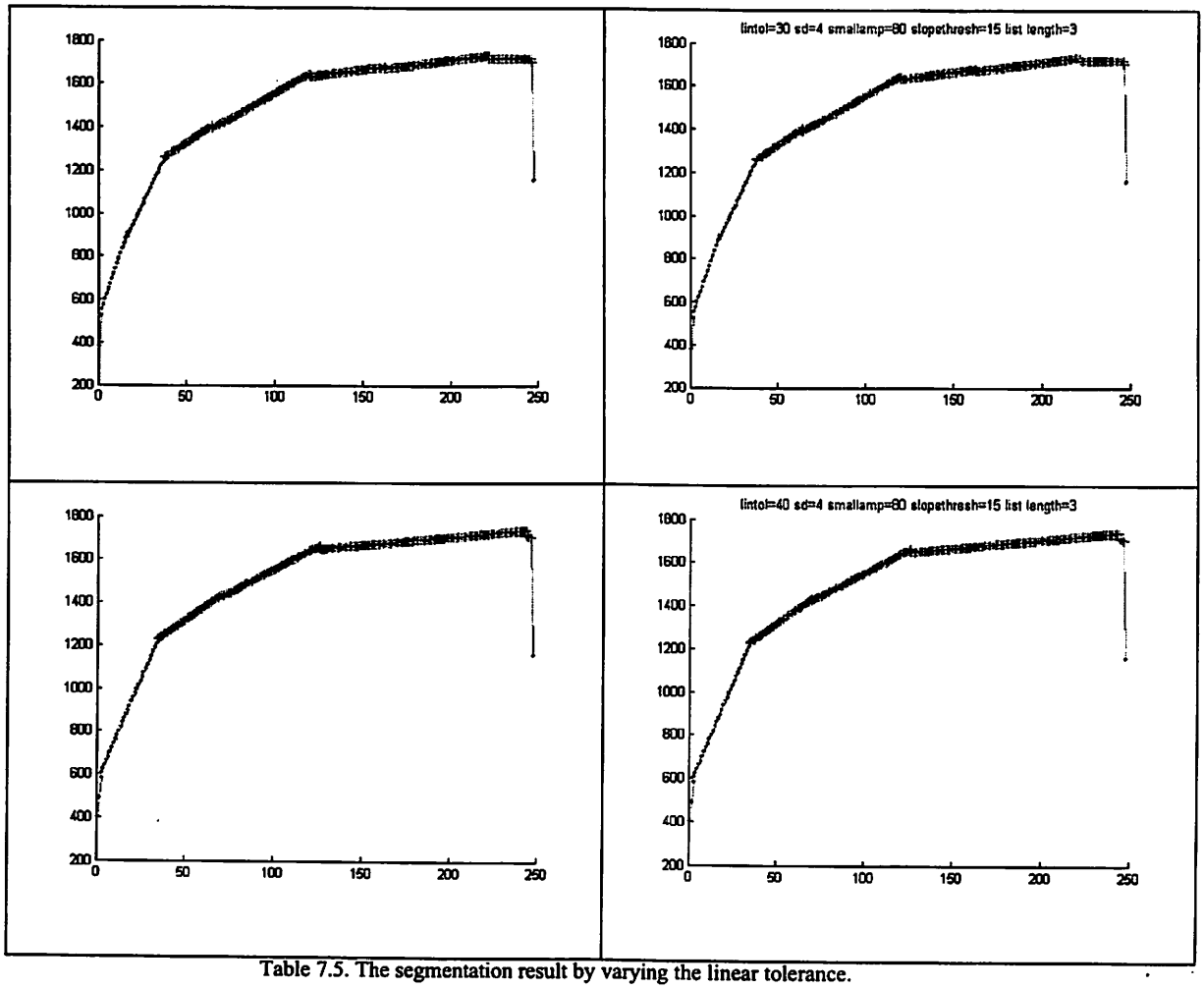


Table 7.5. The segmentation result by varying the linear tolerance.

The following parameter design scheme is recommended:

1) Pick a few representative waveforms. (ones with longer and short durations, big and small transition amplitude, etc).

2) By trial and error, pick a parameter combination that yield correct segmentation for all the sample waveforms.

3) Vary one parameter at a time to fine-tune the combination, as to get a visually acceptable segmentation result.

Although the above scheme is not strictly optimal for diagnostic error rate, it generally yields very small error rate and is applicable to ordinary syntactic parameter design problems.

Chapter 8

TWO CASE STUDIES: FAULT DIAGNOSICS WITH SYNTATIC ANALYSIS

8.1 Introduction

Syntactic analysis offers great flexibility for performing diagnostics. It allows the researcher to select meaningful features and ignore extraneous features or noise. Since the system we have is set up in a research environment, and we do not have control over what wafers to be processed in the etcher, it is not suitable to perform diagnostic analysis on a large scale. We have two data sets. The first data set is the machine signal data of a metal etch marathon run from a manufacturing vendor. The second data set, named "high speed data," was also some machine signal data with sampling rate of about 100 Hz, and was acquired in the Berkeley Microfabrication Lab.

8.2 Metal Etch Marathon Run

The data set consists of real-time signals from more than 1400 wafers. For this analysis, we have chosen the capacitance manometer signal, which reflects the pressure level in the etcher's chamber. The waveform provided by the capacitance manometer is relatively clean, which simplifies visual verification of the analysis.

As mentioned previously, there are several steps in the etching process, including pre-etch of native oxide, main etch, and over-etch. At the beginning of each etching step, it usually takes a few seconds for the etchant gases to stabilize. We usually select the later part of the main etch step for analysis, where the waveform is relatively stable and repeatable. Figure 8.3 shows the "windowing" operation on the capacitance manometer signal. An experienced process engineer can usually tell if etching is faulty by viewing the signal's waveform. For our metal etch marathon data, the commonly seen waveforms are shown in Figure 8.1. We visually classify these signals as either "normal" or of type 1, 2, 3 and 4. Even though we do not have documented faults in this run, types 3 and 4 are most likely faulty. Notice that types 1 and 2 can be viewed as the combination of a normal signal, and a negative

or a positive spike, respectively; they may be considered normal if the spike is small enough. The goal of the analysis is to correctly classify the waveforms.

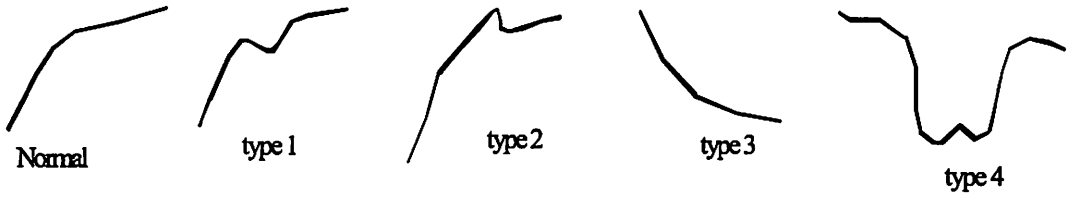


Figure 8.1. Commonly seen waveforms for capacitance manometer in a metal etch marathon run.

A syntactic system for analyzing the etching signal of a capacitance manometer is presented here. The system attempts to discriminate among various waveform types. Figure 8.2 shows the overall block diagram. When a raw signal comes in, the waveform is pre-processed to facilitate further analysis. Then the waveform is encoded into a string of integers. The string is fed into the classifier to determine the fault category. There is also a numerical spike evaluator in the classifier. We will point out its necessity when we talk about the classification result. The major parts of this syntactic analysis system will be described next.

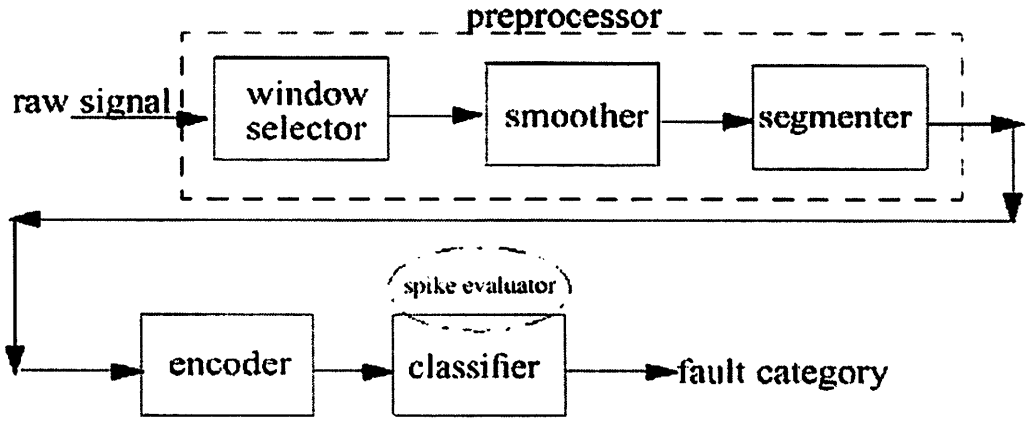


Figure 8.2. Architecture of the overall syntactic system for analyzing the marathon run data.

The preprocessor performs three operations: windowing, smoothing, and segmentation (Figure 8.3). "Windowing" refers to choosing the appropriate time interval for observation during the etch cycle of one wafer. The time window we select is usually the later part of the main etch step. For the capacitance manometer signal, there are two dominant positive spikes (as opposed to the minor ones in the stable region), one big, and one small, before the relatively stable region, so we can define a window after the small spike. Since we do not do any analysis on the random, high frequency noise, we can smooth out the noise of the windowed waveform. We use an algorithm called Locally Weighted Scatter Plot Smoothing [13]. This algorithm attempts to predict each point of the signal by interpolation, by appropriately weighing the nearby raw data. The smoother lets the user specify the fraction of total data used for predicting a particular point; the larger the fraction, the smoother the fit. For the capacitance manometer, a fraction value of 0.2 is appropriate in the sense that this transformation seems to preserve the features that are analyzed later by the segmentation algorithm and the classifier. The smoothed waveform is segmented using the faster algorithm discussed in Chapter 7.

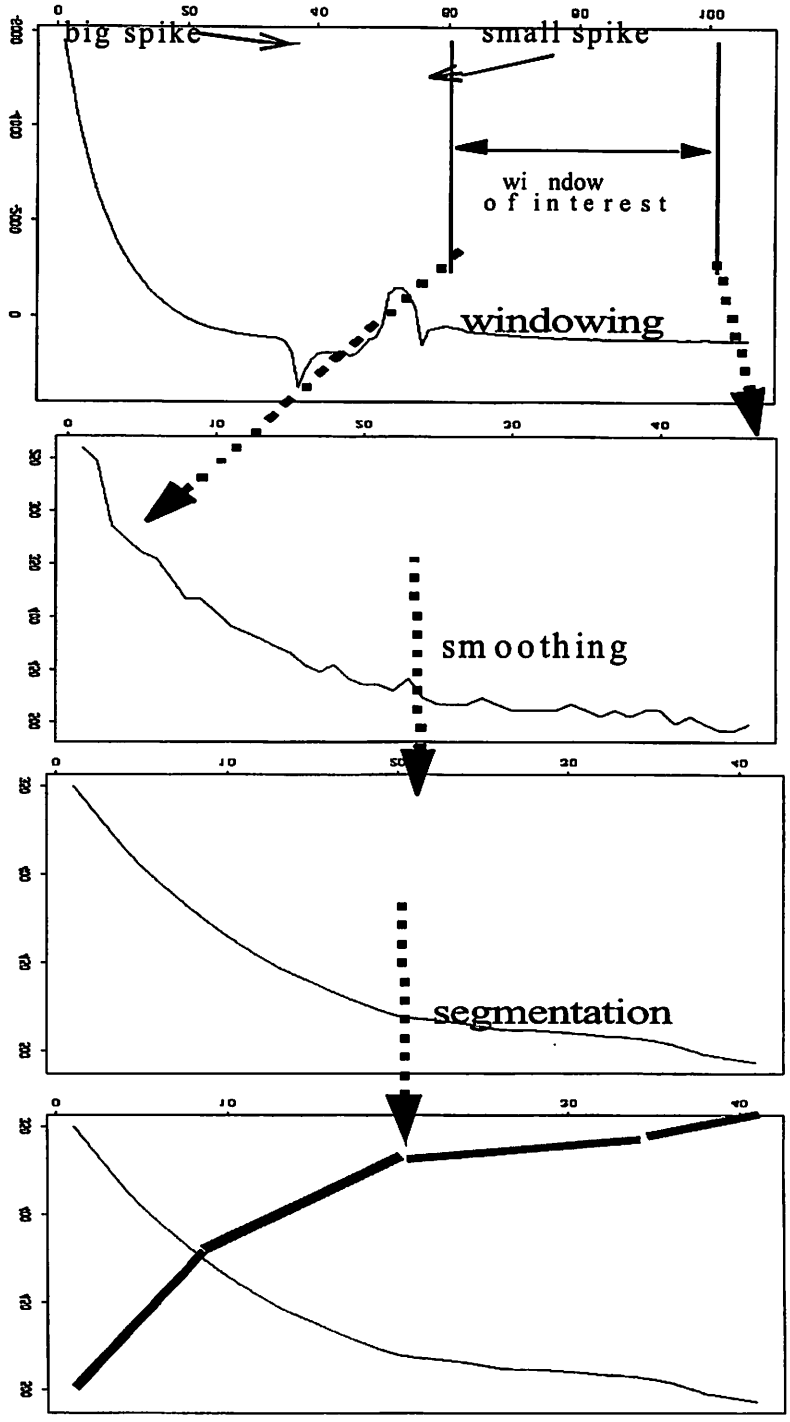


Figure 8.3. The process flow of the preprocessor. The sample rate of the original signal is 2 samples per sec.

Figure 8.4 shows the encoding scheme used to convert the sequence of segments into a string of integers. Five integers are used for encoding the slope of the segments: 2 (fast increasing), 1 (slowly increasing), 0 (almost flat), -1 (slowly decreasing), and -2 (fast decreasing). For the windowed waveform of the capacitance manometer, we consider a segment with a slope of magnitude more than 10 units/sec to be fast changing, less than 4 units/sec to be almost flat, and the in-between values to be slowly changing.

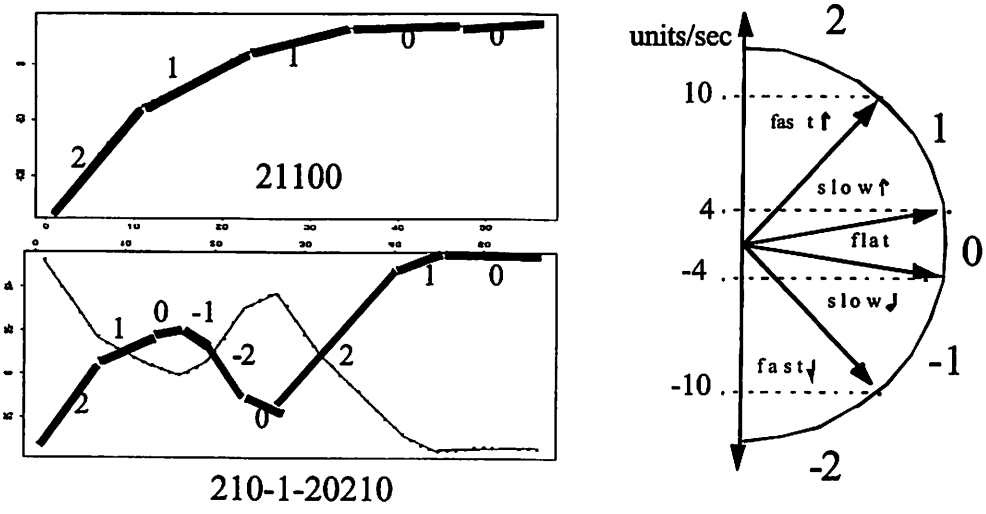


Figure 8.4. The encoding scheme.

The classifier's operation is based on regular expression representation. Regular expressions are used to build the classifier. The expressions are used for matching the encoding string from the raw data. Assuming that x is an integer variable, we show some examples of regular expressions:

x*: zero or more x, i.e., <empty>, x, xx, or xxxxxxx.

x+: one or more x, i.e., x, xx, or xxxxxxx.

x?: zero or one x, i.e.,<empty>, x.

x|y: either x or y.

For example, the following strings are all represented by the same regular expression, 2*1+0*.

2*1+0*: 2222110000, 22221, or 100000.

Based on the process engineer's knowledge on the different waveforms, the classifier can be built to describe the shape of waveforms with one regular expression for each. After an incoming etching waveform is encoded into an integer string, the classifier will try to match the string to one of the regular expressions, and thus determine its category. For example, the following regular expressions can be used to describe the waveforms shown in Figure 8.4:

2+1+0+: describes a curve that first increases rapidly, then stabilizes and finally flattens out. (I.e. the first encoding example 21100 from Figure 8.4)

2+ {1*0*(-1)*(-2)*(-1)*0*1*2*} 1+0+: describes curves that are the sum of the 2+1+0+ curve and a possible negative spike. (I.e. the second encoding example 210-1-20210 from Figure 8.4. Notice that the expression within the curly brackets represents the spike).

However, actual real-time signals may evolve quite a bit over time. The normal waveform may be "stretched" in time or amplitude; the spikes of type 1 and 2 can appear at various times, with varying amplitude and duration, relative to the base waveform. Care should be taken when one derives a regular expression, so that the expression is flexible enough to accept variants of the waveform. Let us discuss in some detail the regular expressions for the five different waveforms in our data.

Regular expressions for five waveform categories

Normal:

The normal waveform has a shape similar to the first example in Figure 8.4. It first increases rapidly, then stabilizes and flattens out. However, expression 2+1+0+ will not be appropriate enough to describe this decreasing trend of positive slope. Due to the "rubber stretching" effect, sometimes slope code of 2 or 1 might not appear in the integer sting. The engineer must exercise discretion in deriving the regular expression. Strings without a "2" or "1" should be accepted. An expression that would accommodate this range of signals is 2*1+0+ | 2+1*0+.

Type 1:

The type 1 waveform is the sum of the normal waveform and a negative noisy peak. Because the amplitudes of the peaks are different, and they are added to an increasing curve, the encoding representation might not contain negative slopes. For instance, see the waveform in Figure 8.5. Also, the peak might appear in any position relative to the normal curve, so it is necessary to consider all scenarios of where the peak appears. The notation of Nxy is used for describing the peak, where N stands for negative peak; x is the slope encoding value before the peak; y is the slope encoding value after the peak. The encoding for the peak is in this format:

(starting segment, left arm, right arm, ending segment)

N_{end} stands for the negative peak occurring at end of the waveform; flat segments do not need to appear after N_{end} .P_{end} is the positive peak defined similiarly. For the peak coding N22, (2 (-2|-1|0|1)+ (0|1|2)* 2), i.e., the negative peak occurring within the fast increasing "2" region, line segments with slope code less than 2 will be considered as a valid left arm; also, it is not necessary to have a right arm.

N22=(2 (-2|-1|0|1)+ (0|1|2)* 2)

N21=(2 (-2|-1|0)+ (0|1|2)* 1)

N11=(1 (-2|-1|0)+ (0|1|2)* 1)

N10=(1 (-2|-1)+ (0|1|2)* 0)

N00=(0 (-2|-1)+ (0|1|2)* 0)

N_{end} =(0 (-2|-1)+ (0|1|2)*)

Type1=2*{N22}?2*{N21}?1*{N11}?1*{N10}?0*{N00}?0*{ N_{end} }?

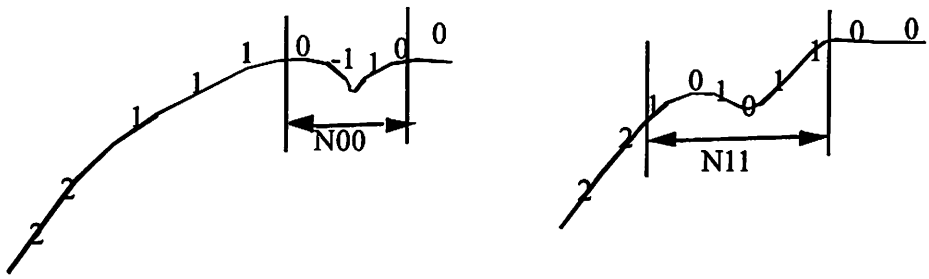


Figure 8.5. Two examples of negative peaks.

Type 2:

The type 2 waveform is the sum of the normal waveform and a positive noisy peak. Similar to the negative-peak type 1 case, it is necessary to consider all scenarios of where the positive peak appears. Notice that there are no P22 and P21. A positive peak has segments with slopes greater than the segments before it. However, as "2" is the largest slope coding value, it is not possible to have a segment with the slope coding value greater than 2. Thus, under this coding scheme, it is not possible to have a possible peak within, or right after a region of segments with coding values "2."

P11=(1 2+ (1|0|-1|-2)* 1)

P10=(1 2+ (1|0|-1|-2)* 0)

P00=(0 (1|2)+ (0|-1|-2)* 0)

P_{end} =(0 (1|2)+ (0|-1|-2)*)

Type2=2*1*{P11}?1*{P10}?0*{P00}?0*{ P_{end} }?

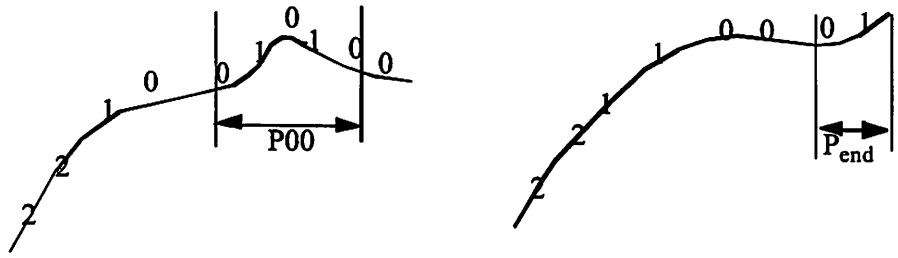


Figure 8.6. Two examples of positive peaks.

Type 3:

The type 3 waveform is more or less the inverted version of the normal waveform. It first decreases rapidly, then stabilizes and flattens out. Again, due to the "rubber stretching" effect, there may not be a "-1" or a "0" in the encoding strings, so the regular expression for type 3 is (-2)+(-1)*0+ | (-2)+(-1)+0*.

Type 4:

The type 4 waveform has a more complicated valley-like shape. There is a bump at the bottom of the valley. The expression is

(-2)*(-1)+0*(-2)+(-1)*0*1*2*1*0*(-1)*(-2)*0*1*2+1*0*.

First-pass classification result

Table 8.1 summarizes the classification result based on the system described above.

Type	Normal	1 & 2	3	4	unknown
Correct	1180	221	9	2	3
Miss	2	0	1	0	--

Table 8.1. Waveform category distribution, first-pass result.

Let us examine this table. There are 3 "unknown" signals that could not be classified as any of the predetermined types. Many normal waveforms may have small spikes; proper smoothing and quantizing prevents them from showing up in the encoding. The two misclassifications for the normal begin with a "1" followed by "2s" instead of beginning with a "2". The type 3 misclassification has a small negative spike. Lastly, the system basically cannot distinguish if there is a positive or negative spike to the normal template, although it is able to detect a significant slope change in the otherwise monotonically increasing waveform. Figure 8.7 can explain this ambiguity. Depending on how we interpret the different curve regions, we might come up with a positive or a negative spike for the same curve.

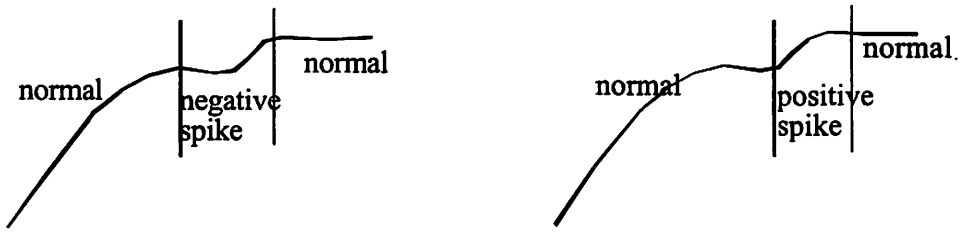


Figure 8.7. Two possible interpretations of the shape of the same curve.

Spike Evaluator

One apparent way to resolve this structural ambiguity is to add quantitative measuring ability to the classifier, in order to find out the sign and magnitude of the spikes. Similar schemes have been implemented for ECG waveform analysis. For example, for more accurate ECG waveform

classification, Koski, et al. [3] compute the amplitude and duration of candidate P wave and T wave. Based on these numerical attributes, the wave in question is designated as a noisy waves, a P wave or a T wave. Here, a spike evaluator is proposed to measure the magnitude and sign of spikes (Figure 8.8). We first take the smoothed signal, centered, and standardized by its standard deviation, and we then subtract a reference signal. On the residual plot, the maximum peak value represents the value of the spike. In our study, we put a threshold of 0.3, which means that if the spike is less than 0.3 times the standard deviation of the signal, we consider the process to be normal. Using this criterion, 60 examples of type 1, and 29 of type 2 are classified as faulty. The improved results are shown in Table 8.2. Notice that a small spike added to a signal is a very common phenomenon. It should not be a surprise that out of the ten type 3 signals, one has a small spike. If we construct a spike evaluator for fault type 3, the one classifying error due to the small negative spike added to the signal would be corrected as well.

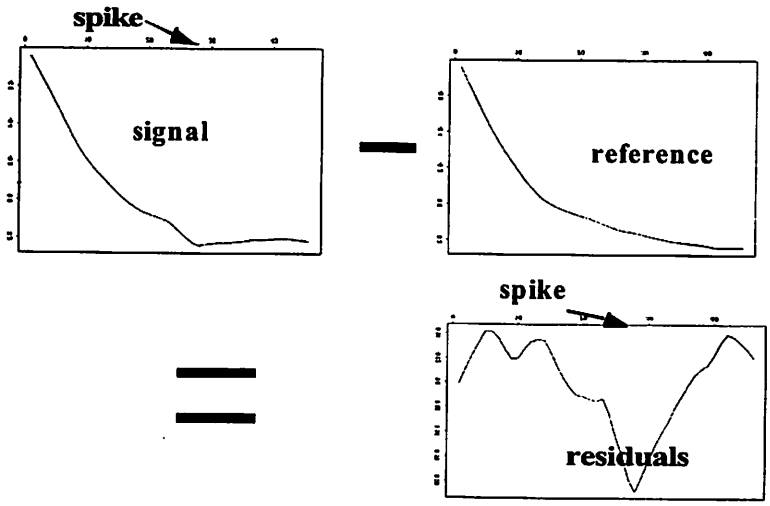


Figure 8.8. The way to measure the spike magnitude in the classifier.

Type	Normal	1	2	3	4	unknown
Correct	1312	60	29	9	2	3
Miss	2	0	0	1	0	--

Table 8.2. Improved waveform classification with the spike evaluator.

8.3 Analysis of the "High Speed" Data

High speed data is acquired during the plasma ignition stage, before any etching occurs. This stage is the transition between the pre-etch and the main etch. The sample rate is 100 samples per second, instead of one or two samples per second, as was the case when monitoring during the entire etching period. The high sampling rate is needed to capture the detail of the transition waveform, and this is where the term "high speed" comes from.

Our assumption about the "high speed" waveform is that each waveform corresponds to an operating condition. The goal of the analysis is to describe the shape of a waveform and thus determine its operating condition. There are two designed parameters for the operating conditions, namely, "tune" and "load." They can be assigned to different experimental levels, such as "high," "medium-high," "baseline," "medium-low," and "low."

Let us examine some waveforms. Figure 8.9 shows two baseline waveforms and two medium-low tune and load waveforms. For the baseline waveforms, the region between the first and second spikes might be somewhat different; otherwise, the two wave-forms will have very similar structures. For the medium low tune and load waveforms, the region after the big positive peak can be quite different. Also, we can infer from inspection that the negative peak can be sometimes narrower (as in the first waveform) and sometimes wider (as in the second waveform). A human brain can effortlessly analyze the waveforms and come up the above observations. We will build our automated analysis system with these observations in mind.

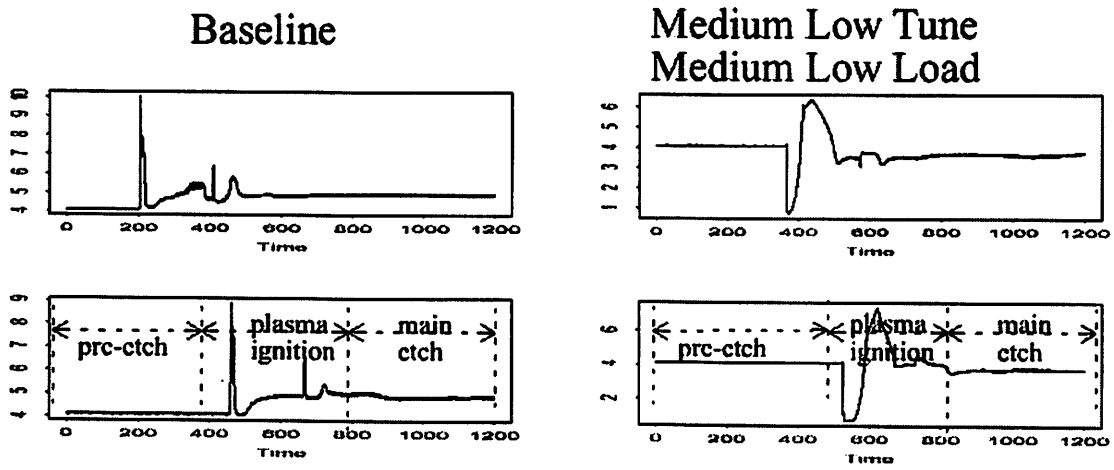


Figure 8.9. Two types of TCP line impedance waveforms for two different operation conditions.

Syntactic analysis is partly science and partly art. For accurate classification, the importance of engineering judgment cannot be overemphasized. This means that the rules encoded in the system are going to be highly specific to the nature of the data. The "high speed" data waveforms are much more complicated than the main-etch waveforms we analyzed previously, so we cannot use the analytic scheme for the main-etch waveforms. Using line segments as the primitive elements would make the classifier extremely complicated. Also, using slope attributes alone would not adequately describe the "high speed" waveforms.

Horowitz proposes a syntactic algorithm for detecting peaks in ECG signals [6]. Belforte uses a peak-coding table look-up method to analyze ECG signals [7]. After taking the first derivative on the raw ECG data, the waveform is parsed into peaks. Based on the amplitude and duration of a peak, a letter code is assigned to it. Trahanias and Skordalakis suggest using peak and segment as two types of primitives, and one can build a hierarchy for a waveform from the primitives in a bottom-up fashion [8][9]. However, the use of a "peak" as a primitive can be troublesome. Notice that if a positive peak is followed by a negative peak, the two peaks will share a common arm in the middle. That is, a lower-level element is being shared by two higher-level elements; this will complicate the syntactic structure description. Also, it may be difficult to define the duration and amplitude of a peak if the left and right

arms of a peak are uneven. Nevertheless, we believe that the recognition of complicated waveforms can be done in a fairly straightforward way, as discussed next.

A new scheme is proposed for recognizing "high speed" waveforms. Three types of primitives are used: UP (monotonically increasing), FLAT (approximately constant), and DOWN (monotonically decreasing). Each primitive consists of small straight line segments. For our data, the line segments with slope between -0.1 and 0.1 unit per data point are considered FLAT; less than -0.1, DOWN; greater than 0.1, UP. See Figure 8.10 for drawing of the primitives.

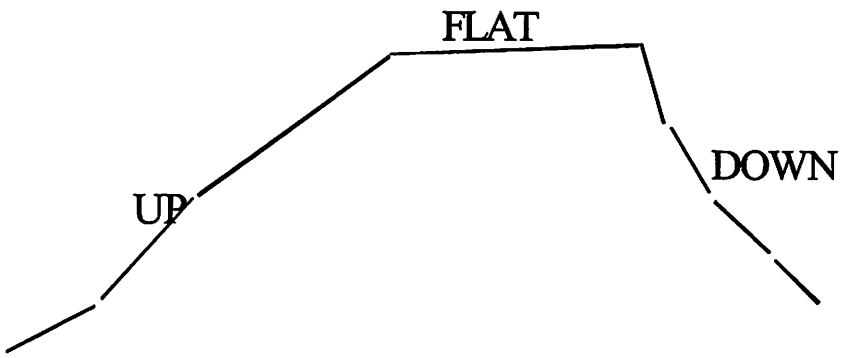


Figure 8.10. Illustration of three types of primitives.

Then we perform the nosy segment processing techniques in Chapter 7 on the waveform. Since the waveforms are much more complicated, and we do not have distinct names such as stable etches, endpoint transitions for the primitives. We decide to encode the waveform differently.

Three attributes are used to describe each primitive, in the form of {S, D, A}, where,

S is the slope code, which can be -1 (DOWN), 0 (FLAT), and 1 (UP);

D is the duration code which can be 0, 1, and 2, in order of length. If duration of a primitive is less than 10, D=0; between 10 and 30, D=1; greater than 30, D=2;

A is the amplitude code which can be 0, 1, and 2, in order of magnitude. If amplitude of a primitive is less than 0.4, A=0; between 0.4 and 2, D=1; greater than 2, D=2.

The criteria for quantization can be assigned based on the process engineer's experience with the signal. One should try to make the number of primitives corresponding to each quantized value roughly the same. This will make the task of building the classifier easier. Consider the case where we use a very strict criterion on the FLAT primitive, in which case only line segments with slope very close to zero will be assigned slope code of 0. Then the number of FLAT primitives will be very small, and it simply defeats the purpose of having a FLAT attribute; since the FLAT attribute were to be left largely unused, we might as well just two attributes, UP and DOWN.

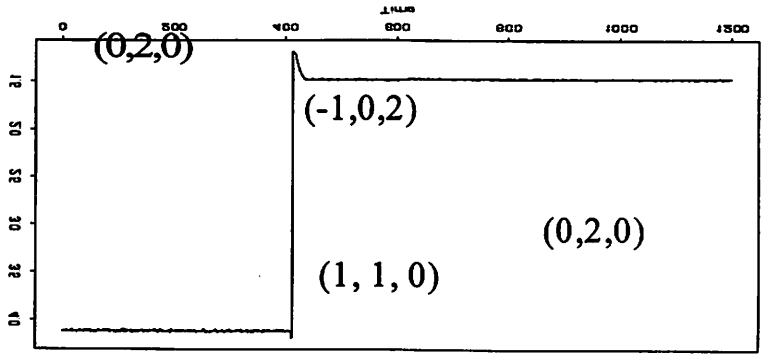


Figure 8.11. An encoding example. This is the low-tune and high-load waveform.

For the above low-tune and high-load waveform, the list of the numerical values for the primitives is

(0, 412, 0.02) (-1, 1, 2.99) (1, 23, 0.35) (0, 763, 0.01), which can be coded as,

(0, 2, 0) (-1, 0, 2) (1, 1, 0) (0, 2, 0).

The syntactic rules have to be created to take into account the error tolerances used in extracting the primitives. In training the classifier, one should be careful with primitives close to the boundary value. If there is a reason to believe that the corresponding primitive of the subsequent waveform may take

on either of the two encoding values which share a common boundary, we should use the logical OR ("|") operator on the two values, so that both values will be accepted.

Consider the third primitive of the above waveform. Its amplitude is 0.35, which is fairly close to the boundary value of 0.4. We should make the classifier accept both 0 and 1 for the amplitude attribute. The classifier for the waveform can be,

(0,2,0)(-1, 0, 2) (1, 1, 0|1) (0, 2, 0).

Indeed, engineering intuition is of great help in building the back to the observation on the LHext waveform. The basic idea is to write the regular expression based on the common region. Anything attached to the common region will be acceptable.

{Common} {Anything}

Anything = -2 | -1 | 0 | 1 | 2 | , | (|)

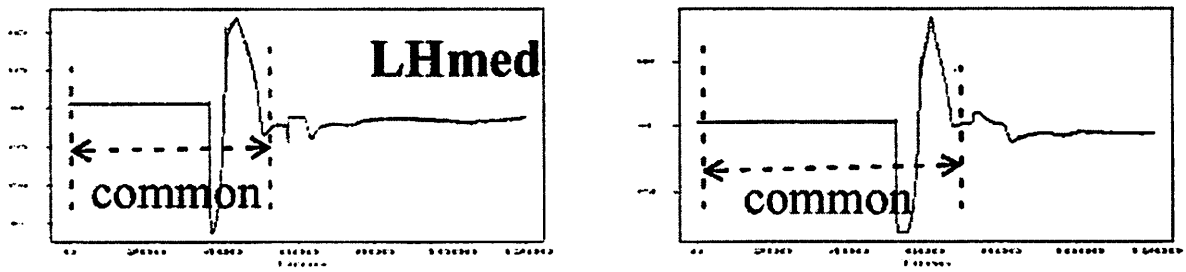


Figure 8.12. Highlight the common region in two waveforms collected at the same operating conditions.

In this case, the common region is a big FLAT segment followed by a negative peak and a positive peak. Notice that the top portion of the negative peak might be relatively flat. Therefore, after

segmentation, a small FLAT primitive corresponding to the top of the negative peak might exist. With this in mind, the common region can be coded as follows:

top_flat = (0,0|1,0)

common= (0,2,0)(-1,0,2){top_flat}?(1,2,2)(-1,0,1)(1,0|1,0|1)(-1,2,2)

Please see Appendix A for the flex code of the classifier.

Figure 8.13 shows the basic schematic of the LAM 9400 plasma etcher, which is a transformer-coupled plasma (TCP) system. The inductive planar coils at the top of the chamber are wound from near the center to the outer radius of the chamber. Plasma is created by applying RF power to the inductive coil. Another RF power source is applied to the substrate for ion-bombardment of the wafer. There is one matching network for each RF source. The upper one is a capacitive network, consisting of two variable capacitors, the tune vane capacitor and the load capacitor. The lower one is a L-type network; the variable circuit elements are the tune vane capacitor and the load coil (see Figure 8.14). A matching network tries to match the impedance it "sees," as to maximize the power transfer from the RF source to the plasma. During the matching operation, we can acquire a list of signals from each network. Some useful signals for fault detection and diagnosis are listed in Table 8.3. For this work, we analyze TCP line impedance waveforms for classifying machine operating condition. For this classification purpose, it is sufficient to analyze just one signal. Multiple-signal analysis is still under investigation.

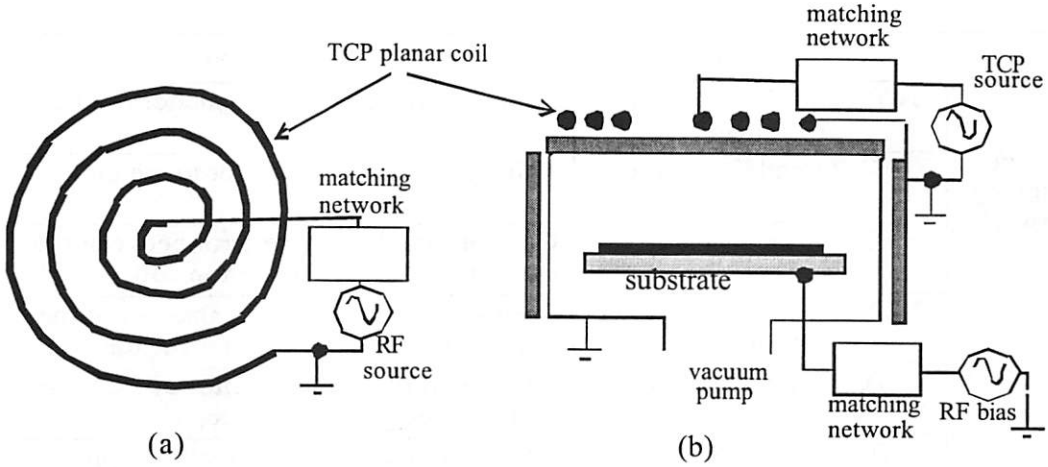


Figure 8.13. a) Top view of the inductive planar coil. b) The side-view illustration of a TCP system.[24]

	Position	Description
Upper Matching Network	TCP Tune Vane Capacitor Command	Value for the tune vane capacitor to match
	TCP Load Capacitor Command	Value for the load capacitor to match
	TCP Phase Control	Control signal of phase error between the current and voltage at the top coil
	TCP Tune Vane Capacitor Position	Position of the tune vane capacitor of the upper matching network for the top coil
	TCP Load Capacitor Position	Position of the load capacitor of the upper matching network for the top coil
	TCP Line Impedance	Apparent input impedance of the upper matching network
Lower Matching Network	RF Tune Vane Capacitor Control	Control signal for the tune vane capacitor of the lower matching network
	RF load coil Control	Control signal for the load coil of the lower matching network
	RF Tune Vane Capacitor Position	Position of the tune vane capacitor of the lower matching network
	RF Load Coil Position	Position of the load coil of the lower matching network
	RF power	Power transferring to the substrate
	RF Line Impedance	Apparent input impedance of the lower-matching network
	RF voltage	Substrate bias with respect to ground

Table 8.3. Real-Time Signals Collected for the Lam TCP 9400.

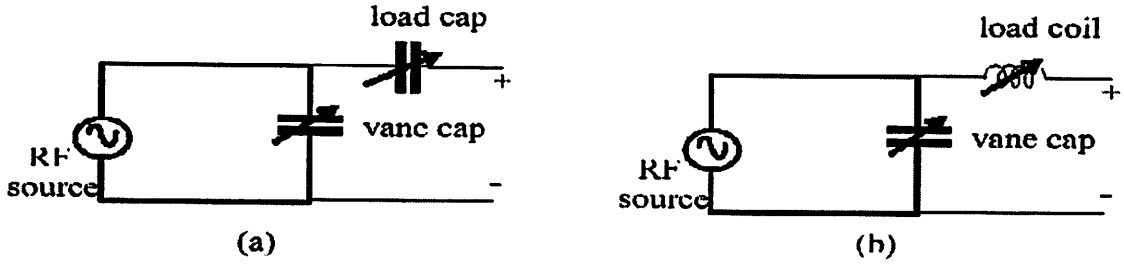


Figure 8.14. a) A capacitive matching network. b) An L-type matching network.

The two designed parameters for the operating conditions, "tune" and "load," are the pre-specified values for the two variable capacitors of the upper matching network to follow. They each can have one of the experimental designed levels of "high," "medium high," "baseline," "medium low," and "low."

Each parameter is on a standardized scale, shown in Figure 8.15. "H" and "L" stand for high and low, respectively; "ext" and "med" stand for extreme and medium respectively. "HLext" means that the operating condition of extremely high tune and extremely low load. There are nine operating conditions: Baseline, HHext, LLext, HLext, LHext, HHmed, LLmed, HLmed, LHmed.

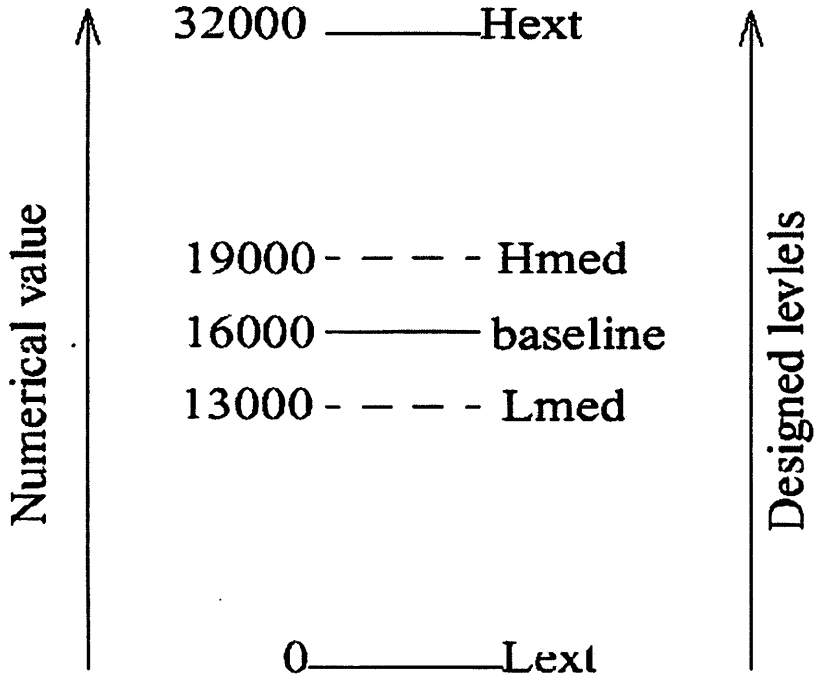


Figure 8.15. The designed-level description of the parameters tune and load.

The results are summarized in Table 8.4. The bold italic wafer numbers signify the misclassified cases. The baseline miss has to do with the fact that a routine spike is significantly weaker in the other

signals, so that the second and third peaks of the line impedance signal disappear. the LLmed miss has to do with a high spike occurring in the common region, so that the recognizable pattern is greatly "damaged." Notice that if the high spike occurs far away from the common region (first waveform of Figure 8.16), the common region will not be altered, and thus classifying error will not occur.

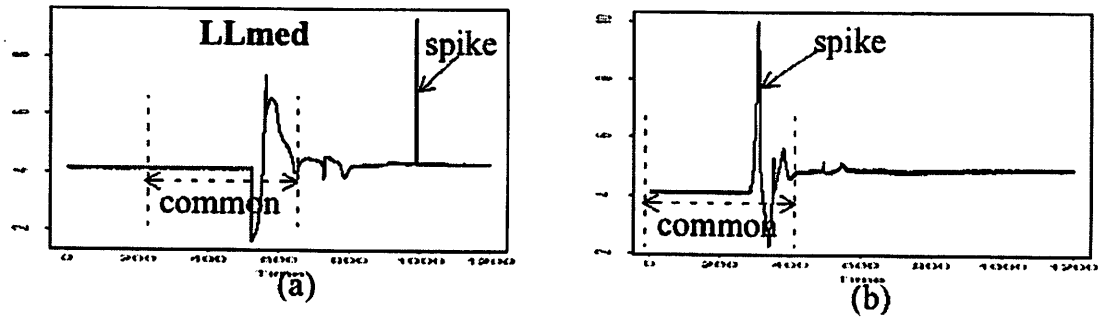


Figure 8.16. The high-spike effect on the waveforms. (a) The spike occurs far away from the common region. (b) The spike occurs right at the common region.

Finally, for the LHmed wafers, #26 and #28 waveforms are similar to LLmed ones (see Figure 8.12 and Figure 8.16). #27 waveform is similar to LHext ones (Figure 8.11). This means that #27 waveform is totally different from those of #26 and #28. As a matter of fact, the similarity that confused the classifying task is so great that even a human expert will not be able to make a distinction. This implies that probably any pattern recognition scheme will not tell those confounded waveforms apart. Therefore, the author will not consider this as a classification error.

Type	Wafer number	Comment
Baseline	1,2,3,*16*	missing spike
HHext	4,5,6	
LLext	7,8,9	
HLext	10,11,12	
LHext	13,14,15	
HHmed	17,18,19	
LLmed	*20*,21,22	extra spike
HLmed	23,24,25	
LHmed	26,27,28	confused, with LHx, LLm.

Table 8.4. Result summary. The italic wafer numbers signify misclassification.

Chapter 9

LOW FREQUENCY ANALYSIS FOR PLASMA ETCH DIAGNOSIS

9.1 Introduction

In previous chapters we discussed the use of three different sources of signals for plasma etch diagnostics. In this chapter we address the use of low frequency signals. For low frequency analysis, we increase the sampling rate from 1~2 Hz to 10 kHz, in order to acquire machine signals coming out of the etching chamber, such as power, pressure, impedance, load and tune capacitor position, etc. The National Instrument A/D converter is, with maximum sampling rate of 300 kHz. The low frequency patterns that we observe may come from different sources.

1) The machine consists of many sub-parts that are to operate at different frequencies. Many mechanical and electrical parts operate at low frequencies in the range of a few hundred Hz up to a few KHz.

2) Harmonics of high frequency, such as the ones generated by the RF sources and optical emission, mix with each other and generate low frequency products.

3) Lieberman, et al found out that under certain settings of chamber pressure and power, plasma discharges of SF_6 and Ar/SF_6 exhibit oscillating behavior in charged partide density, electron temperature and plasma potential.

4) Praburam and Goree [21] observed that when the plasma chamber under operation is dusty, there is a void of ionization wave moving back and forth in the chamber at frequency of orders of 10 Hz.

9.2 Literature survey

The literature provides some evidence that plasma absorbs energy of signals at relatively low frequencies, and can be treated as a filter for low frequency signals. Bongdira etc. [20] and Henion etc.

[19] did some iron nitriding experiments. They varied the frequency of the input pulse to the plasma chamber, from 0 Hz to 1 KHz, and the parameters of the resulting nitriding samples were considerably different (Table 9.1). Also, Lebeau [18] measured ion cyclotron resonance heating (ICRH) power absorption in a plasma subjected to different modulation frequencies (50~300 Hz). The power absorption increased with frequency (Figure 9.1). Thus, from these experiments, we see that the plasma chamber can absorb power from low frequency signals effectively and selectively for different frequencies. We can reasonably expect that if we change the composition of the plasma by altering the chamber settings, the filtering properties will change accordingly. This change of filtering properties has been confirmed in the preliminary work.

Parameter	Frequency (Hz)						
	50	100	150	200	250	300	500
Vickers hardness (HV 1)	500	730	375	265	330	330	260
$\gamma'(111)$ diffraction peak intensity (a.u.)	92	100	18	14	19	19	7
$\gamma'(200)$ diffraction peak intensity (a.u.)	52	77	12	9	11	11	5

Table 9.1. Metallurgical parameters of a nitriding sample as a function of plasma frequency [20].

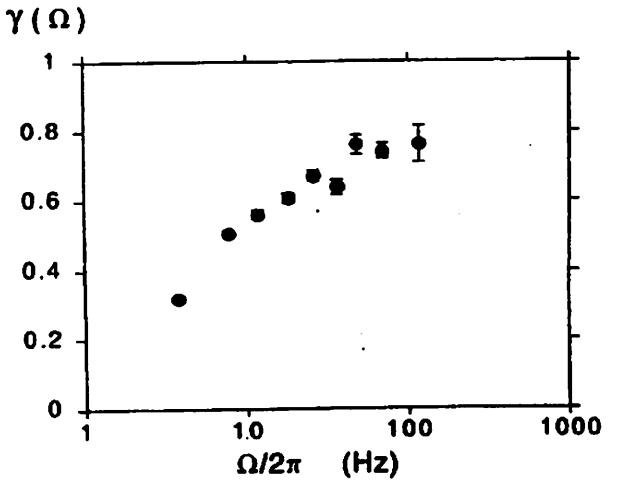


Figure 9.1. Fraction of the RF power confined with a confinement time longer than the modulation period [18].

Therefore, by performing frequency transformation on the output signals of low frequency, we should be able to monitor the chamber and wafer states. Designed experiments can be used to identify the spectral features that behave consistently from wafer to wafer, and ignore those that behave inconsistently.

There are many other examples where spectral analysis has exploited low-frequency resonance of various systems:

Scholtz etc. [11] used low frequency noise to characterize semiconductor devices. They plotted noise vs. temperature at different low frequencies. Jevtic [13] modeled the relationship between low-frequency noise and imperfection of the device.

Fritsch etc. [12] used a low-frequency micromechanical resonant vibration sensor for wear monitoring of mechanical tools, such as drills and mills (Figure 9.2). Seifert etc. [15] used very low frequency (10-6 Hz to 10 Hz) to detect and classify aging phenomena of composite insulating materials.

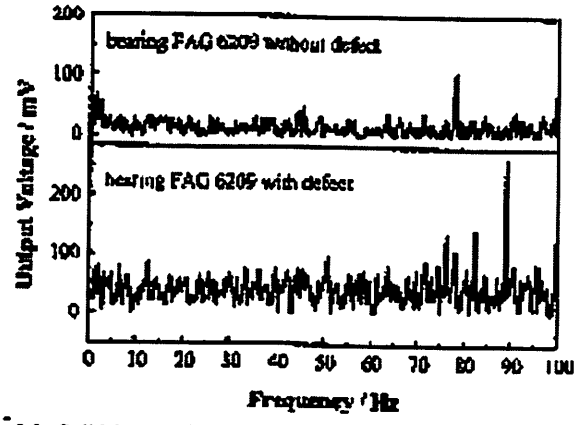


Figure 9.2. Solid-borne vibration spectrum measured with a piezoelectric acceleration senor, for the cases of a bearing being without defect and with defect [12].

To avoid the need for expensive high rating transformers, Hilder etc. [24] and Kruger etc. [16] used test supplies of 0.01 Hz to test cables or circuits, and tried to predict the 50 Hz behavior of the system.

The above examples in the literature suggest that we may be able to use empirical low frequency analysis to perform plasma etch fault detection, equipment and wafer state modeling..

9.3 Preliminary experiments

A few preliminary experiments with the chamber state were conducted to explore the low frequency behaviors in respond to varying machine parameters, the time drifting phenomenon, and its relationship with the OES spectrum. The parameters involved are HBr flow rate, chamber pressure, top and bottom RF power. The center points can be found in Table 6.1 in Chapter 6.

Figure 9.3 and 9.4 show the 410 Hz peak from the RF top impedance signal and the RF bottom impedance signal respectively. The peak from the RF top impedance signal goes down with increasing top power, whereas the one from the bottom signal goes down with increasing bottom power. Also, to study the time effect on the peaks, two time slots were scheduled about 20 days apart, before and after Christmas 1999. The peaks drift down for both top and bottom cases.

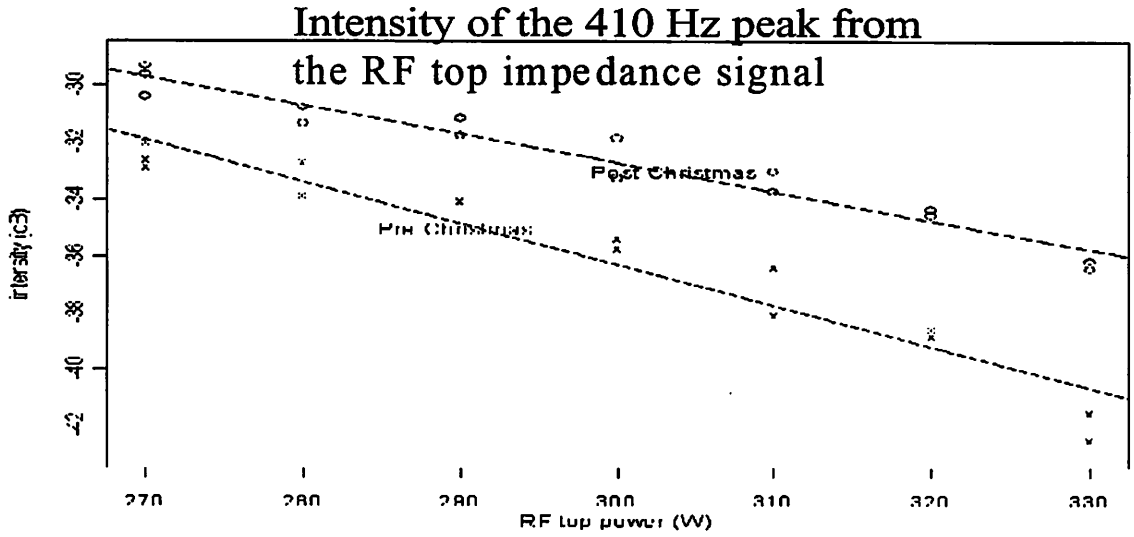


Figure 9.3. Varying RF top power -10% to +10%.

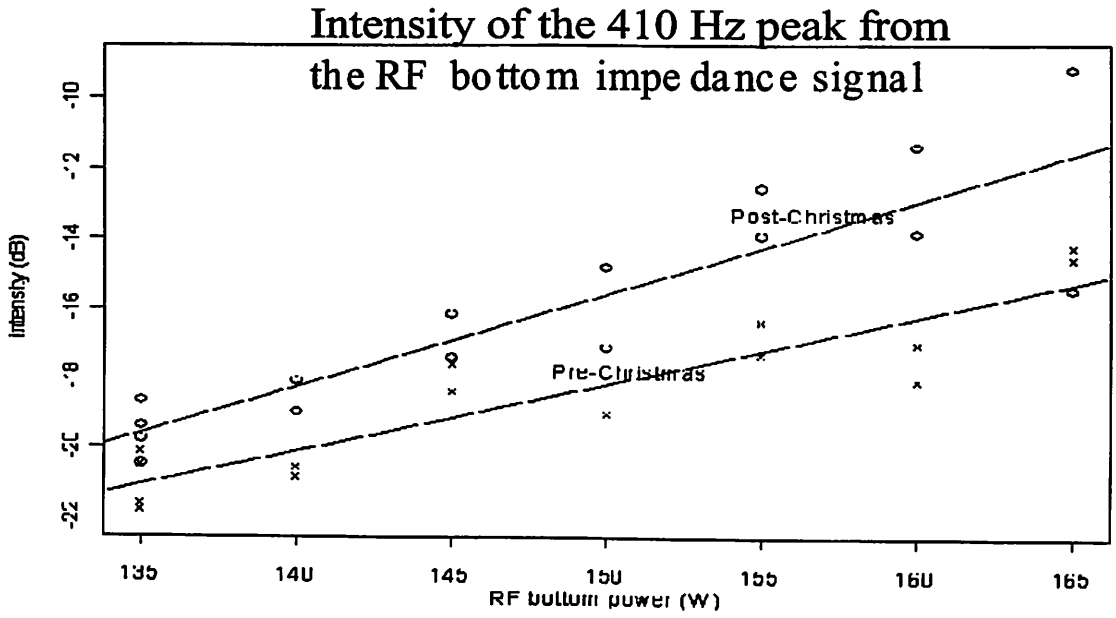


Figure 9.4. Varying RF bottom power -10% to +10%.

When varying the HBr setting, bandwidth transition of harmonics can be observed on the spectrum; the drifting behavior due to time of the bandwidth transition can be seen as well, as shown in Figure 9.5, 9.6 for the December 2000, and January 2001 experiments, respectively. Notice, the plot label specxxx.yy, where xxx is the wafer run number, yy is the signal number. "10" is the load capacitor position signal, and 13 is the tune capacitor position signal. Wafer numbers 183, 185, 187, 189, 191, 193, and 195, correspond to HBr setting of 135, 140, 145, 150, 155, 160, 165 sccm, respectively. An even wafer number was set at the same HBr flow rate of the odd wafer number immediately before it. For instance, for wafer 184, the HBr setting is 135 sccm, which is the same as for wafer 183. Notice that the center point plot is not shown in order to conserve space. It looks similar to the one with the lower settings. For the experiment before Christmas, the bandwidth transition point occurs between 160 and 165 sccm. After Christmas, the transition occurs right after the center point.

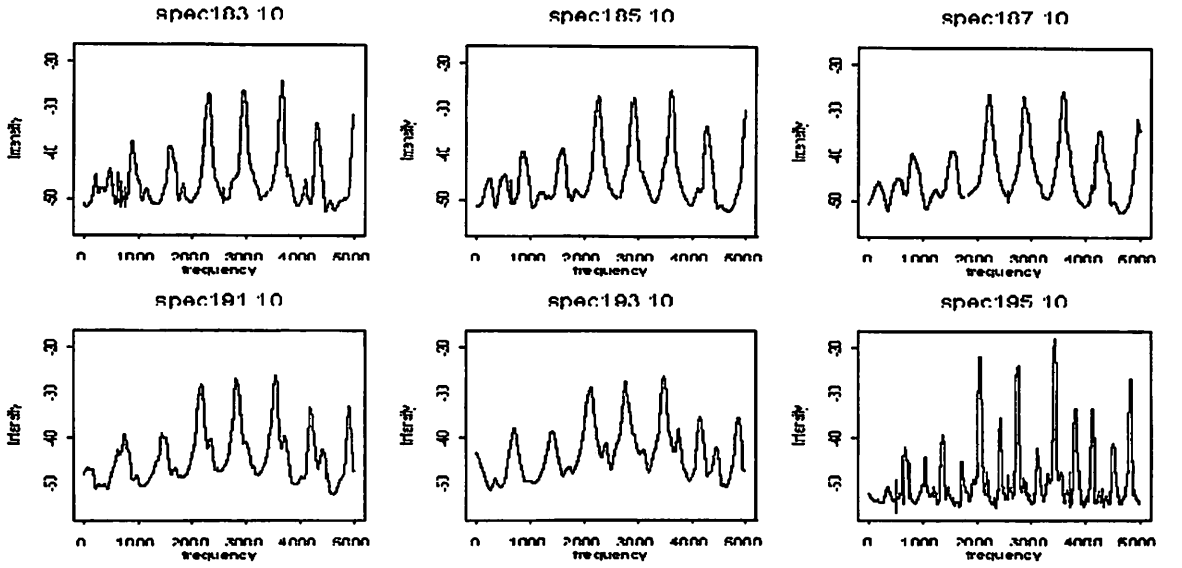


Figure 9.5. Frequency plot of the signal for tune capacitance position, pre-Christmas HBr -10% to +10%.

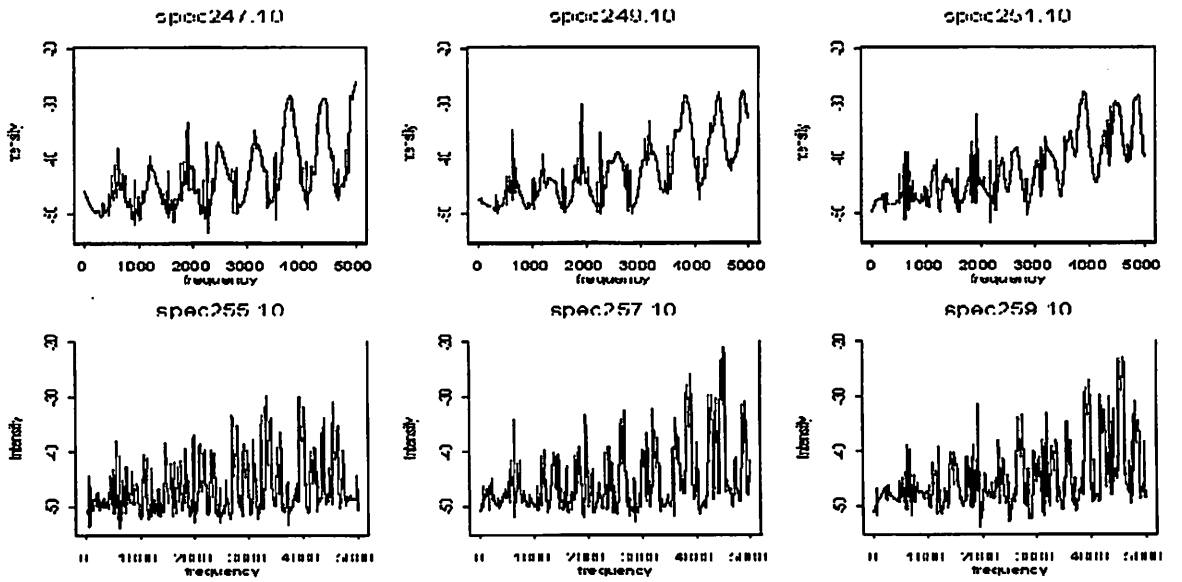


Figure 9.6. Frequency plot of the signal for tune capacitance position, post-Christmas HBr -10% to +10%.

Figure 9.7 shows an OES plot with varying HBr setting. The OES data was acquired from the etching system simultaneously in addition to the low frequency data. There are seven settings and the wafer group size for each one is three. The OES signal intensity appears to be in linear relationship with the varying HBr setting, and does not seem to have an relationship with the low frequency harmonics bandwidth. Another experiment was performed by varying chamber pressure, with nine set points and utilizing two wafers (Figure 9.8). We can see in Figure 9.9 that the OES intensity stays constant for the first three setting points, then a transition point follows, and the OES intensity starts to decrease linearly with increasing chamber pressure. This transition point coincides with the low frequency bandwidth transition point.

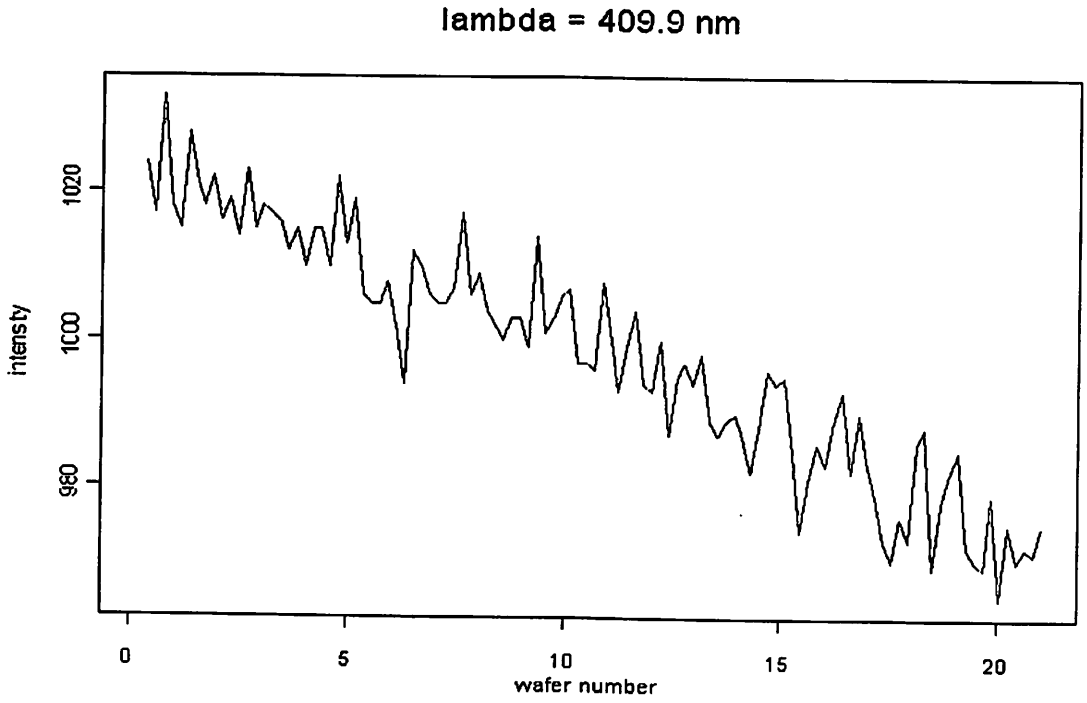


Figure 9.7. An OES peak intensity plot for varying HBr 125 to 175 ccms.

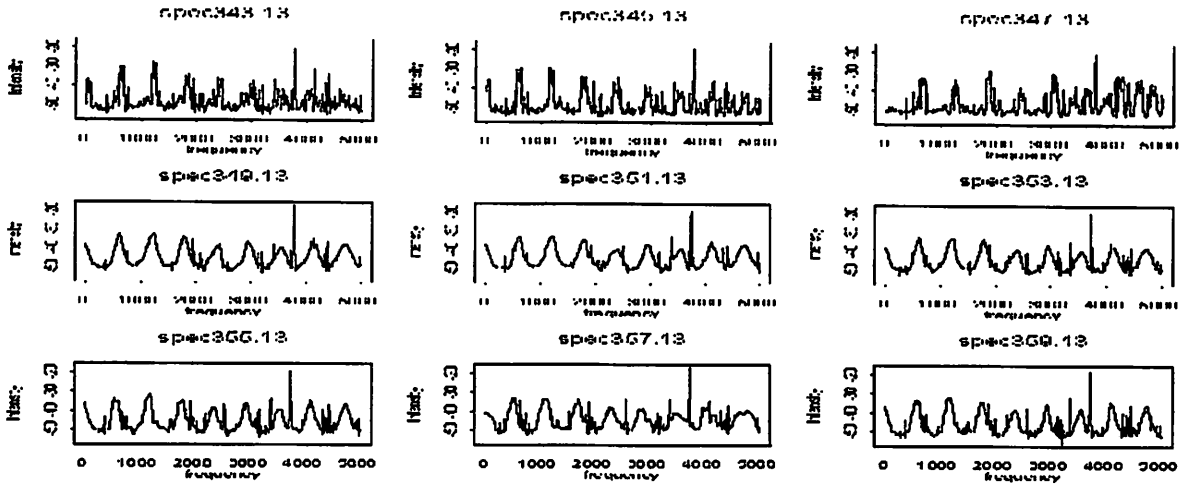


Figure 9.8. Frequency plots of the signal for tune capacitance position, varying pressure 10 to 14 mtorr.

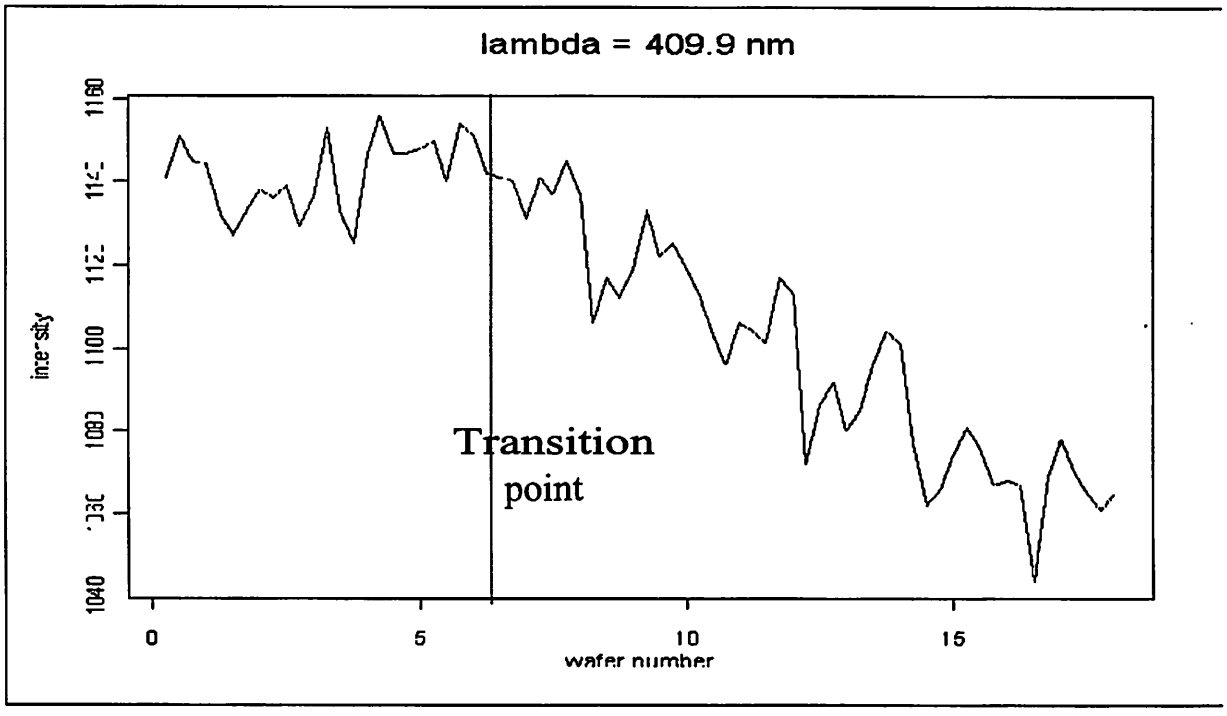


Figure 9.9. An OES peak intensity plot for varying pressure 10 to 14 mtorr.

Additional experiments can be performed in a manufacturing environment. An examination of the low frequency spectrum can be made before and after a preventive maintenance procedure, and before and after the replacement of the RF generator. In this way, we can determine if low frequency analysis provide diagnostic information for maintenance purpose.

On the RF power signal spectrum, it is found that some peaks gradually shift their frequency from wafer to wafer. As shown Figure 9.10, peak 1 stays very much on the same frequency. Peak 2 shifts gradually to the right. Peak 3 and 4 are initially close together, and then gradually drift apart. The physical explanation of the frequency shift is subject to future investigation. Some researchers also came across similar phenomena. Brodskii etc. [22] observed the same dynamic behavior (Figure 9.11), but could not explain it. It might have to do with the transient behavior of the chamber during the first few minutes after the plasma ignition (Figure 9.12), as observed by Roth etc. [23]. These works were not able to explain the transient effect, although they eliminated a few causes. These are some phenomena which make plasma diagnosis challenging.

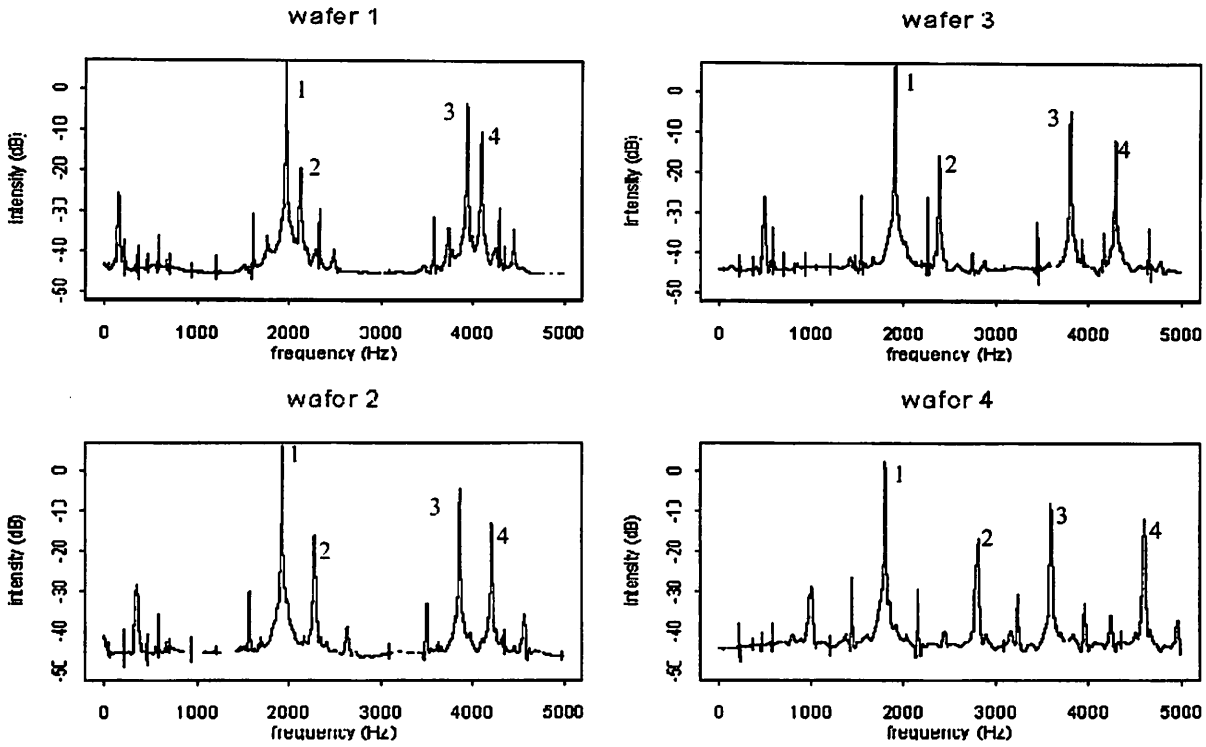


Figure 9.10. Illustration of the frequency shift.

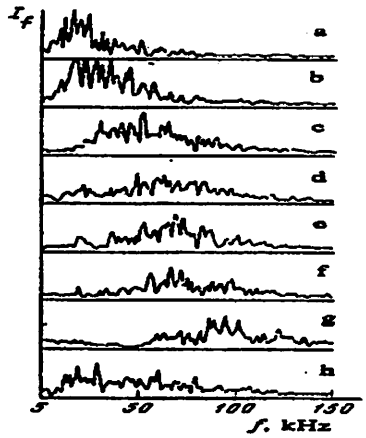


Figure 9.11. Spectral dynamics of the ohmic-heating signal: a) t=150 ms; b) 250; c) 300; d) 400; e) 500; f) 700; g) 850; h) 950 ms [22].

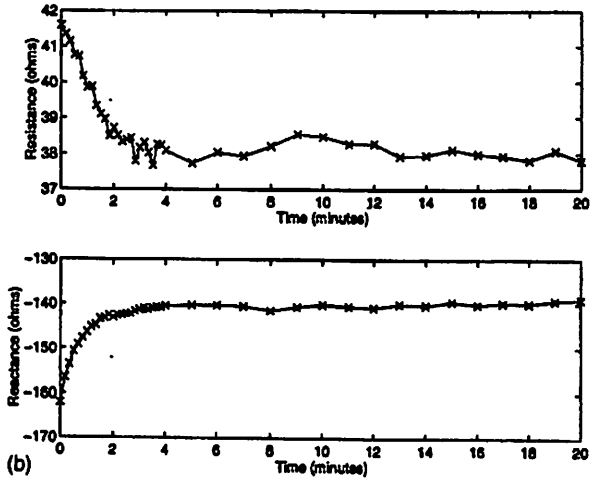


Figure 9.12. Transient values of the plasma impedance after plasma ignition. The plot is showing the impedance of etching a single wafer.

9.4 Proposed technique for analyzing the LF spectra

The analyzing scheme will be similar to the previous diagnostic examples. Figure 9.13 shows an overall design of the system. After turning the raw data into a smooth spectrum by wavelet transform, we build a baseline curve and peak primitives from it. We then encode the primitives. Finally, attribute grammar is used to determine the fault category of the raw input.

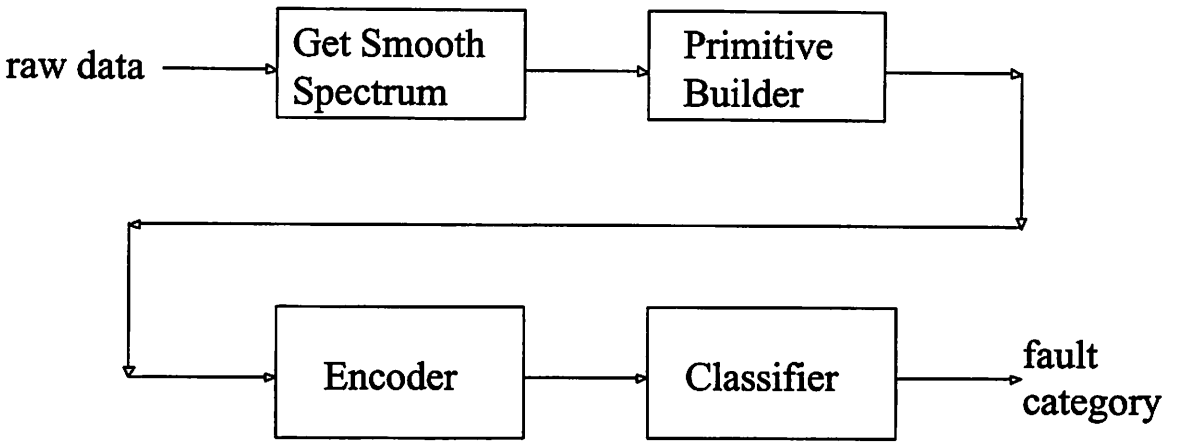


Figure 9.13. A schematic of the diagnostic system for low frequency analysis.

The periodogram of a noisy signal is typically very noisy; and the variance of the periodogram at a particular frequency may be as large as the power of the frequency component itself. Also, this variance does not decrease with increase of data sample size [9]. Gao's spectral wavelet denoising technique [10] may be used to smooth the periodogram. This technique is computationally efficient; it can estimate a nonsmooth spectrum at a near-optimal rate. This method preserves the sharpness of the peaks, while making a smooth estimate on the baseline.

There are two categories of wavelet functions: *father* wavelet ϕ (t), which is used to describe the smooth and low-frequency parts of the signal f(t); *mother* wavelet ψ (t), which is used to capture the detailed and high-frequency parts of the signal f(t). There are many wavelet functions to choose from. The wavelet pair we chose for smoothing the spectrum is called "S8". With scaling index j, and translation index k, they will appear as the following,

$\phi_{j,k}(t)=2^{-j/2}\,\phi\,(2^{j}t - k)$ and

$\psi_{j,k}(t)=2^{j/2}\,\psi\,(2^{j}t - k)$.

When we try to describe the details of the signal, we usually want to have multi-resolution, i.e., more than one wavelet level number for computing the coefficients of the mother wavelets.

$$d_{j,k} = \int \psi_{j,k}(t) f(t) dt \ , \ j = 1,2,..., \ J$$

where j is the wavelet level number, the smaller numbers indicate the coefficients for the finer details of the signal. Notice the J is the maximum wavelet level number, which is a user defined parameter.

When computing the smooth part of the signal, one level of the father wavelet is needed

$$s_{J,k} = \int \phi_{J,k}(t) f(t) dt$$

The denoising procedure is illustrated in Figure 9.14 and outlined below. Notice that in the illustration we set the maximum wavelet level number J to 6.

1) Perform Fast Fourier Transform (FFT) on the raw signal to obtain the log-periodogram.

2) Apply a Discrete Wavelet Transform (DWT) with multiple levels to the log-periodogram.

3) Apply a special threshold rule to the mother wavelet coefficients, according the formula:

$\lambda_j = \max(\pi (\log_e(n)/3)^{-5}, \log_e(2n) 2^{(j-1)/4})$

where j is the wavelet level number, n is the length of the raw data. Any wavelet coefficient smaller than the threshold is shrunk to zero.

4) Apply the Inverse Discrete Wavelet Transform (IDWT) to the remaining coefficients to get the smooth log-periodogram.

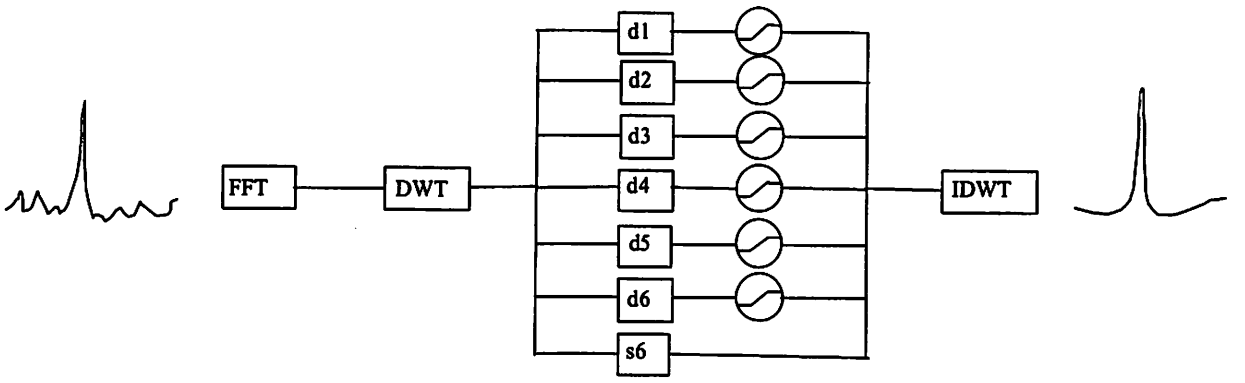


Figure 9.14. Illustration of Gao's spectral wavelet denoising method [32.].

After obtaining a clean spectrum, structural analysis may be performed. Syntactic analysis with attribute grammar [7,8] will be utilized. Two types of primitives, baseline curve and peak, will be used (Figure 9.15). A peak is a sharp spike. The baseline consists of relatively flat and smooth curves with peaks among them. The baseline curve and peak consist of lower-level primitives, the line segments, which can be obtained by a piecewise linear approximation technique on the log-periodogram, as presented in Chapter 7. Based on their repeatable behavior, peaks will be selected for various diagnostic purposes. For both baseline curves and peaks, we will monitor the numerical attributes of amplitude, bandwidth, and power (area under the curve).

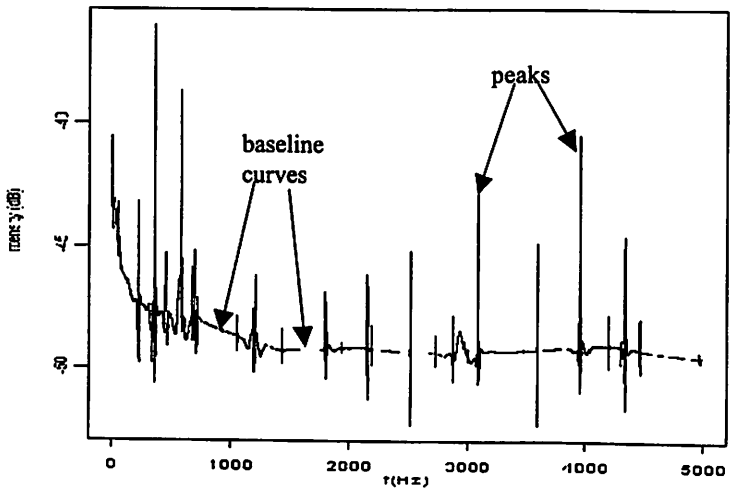


Figure 9.15. Demonstration of baseline curves and peaks.

For a baseline curve, we encode it as: (*BC*, *slope code*, *length code*), where *BC* stands for "baseline curve;" *slope code* can be *up* (1), *flat* (0), and *down* (-1); *length code* can be *short* (0), *medium* (1), and *long* (2). For a peak, we encode it as: (*P*, *amplitude code*, *bandwidth code*), where *P* stands for "peak;" *amplitude code* can be *small* (0), *medium* (1), and *large* (2); *bandwidth code* can be *narrow* (0), *medium* (1), and *wide* (2).

The following illustrates the encoding of two hypothetical examples of two peaks with a baseline in between.

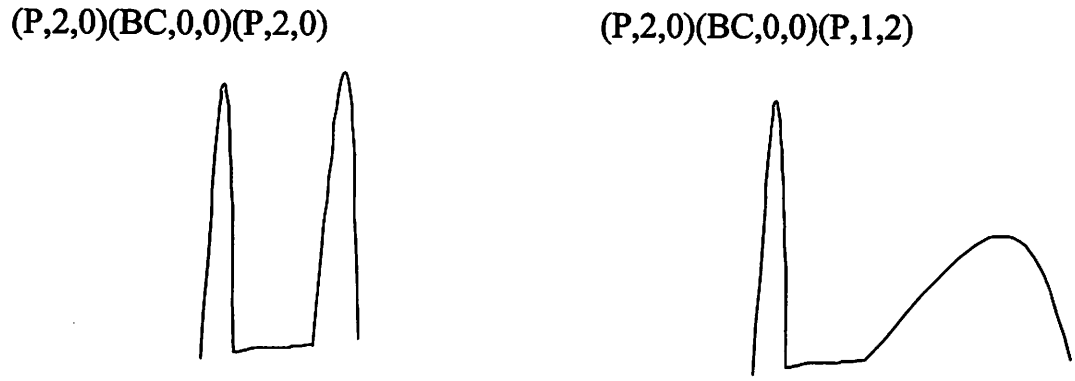


Figure 9.16. On the left: the encoding for a large narrow peak, a flat and short baseline curve, and another large narrow peak. On the right: the encoding for a large narrow peak, a flat and short baseline curve, followed by a wide peak with medium amplitude.

Qualitatively, we will use regular expressions (see Chapter 8 for a discussion on them) to classify the spectrum. For instance, (P, 2, 0) (BC, 0, 0)?(P, 2, 0|1|2) describes two large peaks with or without a baseline curve in-between; the bandwidth of the later peak can be arbitrary. This expression will accept the example on the left in Figure 9.16, and reject the one on the right.

Quantitative attributes will be used systematically to classify the spectra in finer detail. Table 9.2 illustrates the use of attribute grammar. The left column is the qualitative systematic description of the spectrum. The right column specifies the calculation and manipulation of various quantitative attribute of the spectrum. In order to convert the prototype of the rules into executable code, a parser generator such as *yacc* or *bison* will be needed.

Syntax Rules	Attribute Rules
Spe=<LF_spe><HF_spe>	
LF_spe=(BC,-1,2)(P,1,1|2)...	*LF_spe.BC_power=sum(BC_j.power) *If (LF_spe.BC_power>threshold) alarm("cleanness")|
HF_spe=(P,0|1,0)(BC,1,1)(P,0,1)...	*If (P_j.BW>threshold) alarm("gas")

Table 9.2. An illustration of attribute grammar.

Chapter 10

CONCLUSION AND FINAL REMARKS

10.1 Work Summary

We have set up a plasma diagnostic system with three sources of signals, OES, RF, and machine signals. CF_2 OES lines 275 nm and 321 nm are found to be better than any other signals for poly-etch endpoint detection. In addition, excellent statistical models for wafer state prediction are obtained by linear stepwise regression on all available signals. A data exploration system, based on syntactical analysis, is developed for efficiently browsing of the data archive, allowing users unprecedented flexibility in examining the data both qualitatively and quantitatively. Two case studies of syntactic analysis for diagnostics are presented. Finally, the use of low frequency signals for plasma diagnostics in investigated. The syntactic method for analyzing the signals is proposed.

10.2 Remarks on Syntactic Analysis

The most promising technique proposed in this thesis is syntactic analysis. The syntactic method is shown to be robust and accurate for fault detection and diagnosis in plasma etching. For the successful operation of this system, the expertise of the process engineer plays a key role. The system complements the process engineer's expertise in interpreting the etching signals, therefore, parameters of the system must be trained to suit the engineer's needs.

At a glance, syntactic analysis is quite similar to the encoding and decoding techniques in digital signal processing (DSP). In DSP, the engineer first defines a number of logical values, and assigns a voltage or frequency level for each logical value. The data is presented with a stream of logical values, encoded into physical signal levels (voltage or frequency), and transmitted over noisy channels. The receiver will try to ignore the noise in the received signal, and try to match it to a predefined logical value. In syntactic analysis, we define a number of fault categories based on our experience. For diagnosing a plasma etching signal, we would ignore effects such as machine aging, preventive maintenance,

chamber memory, small spikes, and so on, and try to match the signal to a predefined fault category. The similarities between syntactic analysis and DSP are highlighted in Figure 10.1.

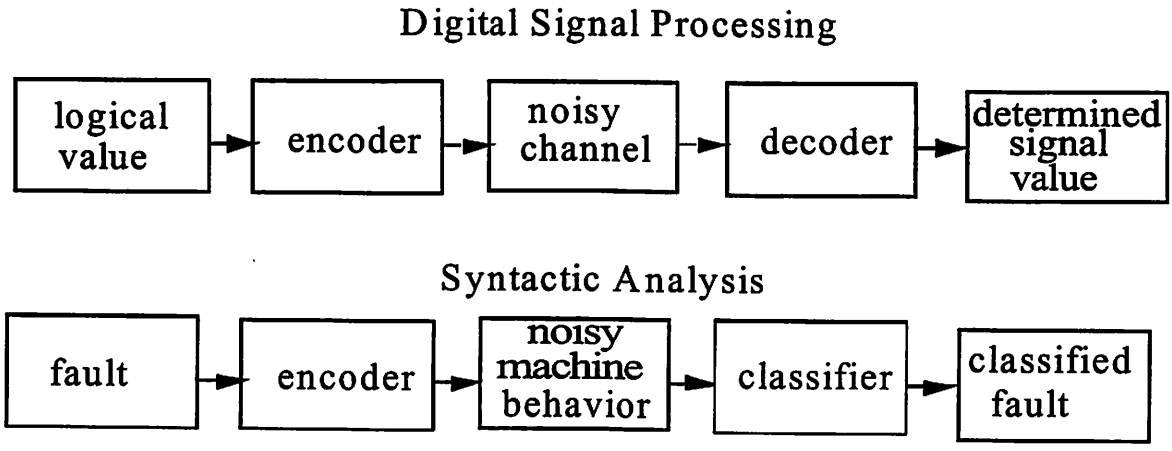


Figure 10.1. Comparison of the overall architectures of DSP and syntactic systems.

The syntactic techniques for solving the classification problems in this thesis may appear *ad hoc*. The reader might wonder if there is any general syntactic method for all the patterns, using the same set of primitives, such that one can develop a syntactic system systematically. While some general schemes (such as You and Fu's) have been proposed, in our experience, this is not desirable, since they might lead to overly complicated grammars, and thus induce higher diagnostic error rates.

In the literature, uses of syntactic analysis to recognize objects tend to be pattern dependent. Many researcher use context-free grammar. For a different pattern, a different set of segmentation primitives must be used; a different grammar must be specified; also, different attribute information, such as segment length, time duration, and amplitude, may need to be considered (this is done usually by using attributed grammar, described briefly in 6.3). Similarly, for the plasma main etch signal pattern, line segments are used as the segmentation primitives; a regular grammar (a subset of context-free grammar) is specified, and the spike magnitude is the attribute considered.

You and Fu [22] propose a general 2-D shape recognition method, in which curve segments are used as primitives. Each curve has four attributes: direction, length, total angle change, and the degree of

symmetry. Also, the angle between two adjacent curve segments is considered. While this method can describe a 2-D pattern in detail, it may complicate the task of classifying plasma ignition waveforms. Obviously, in You and Fu's scheme, if curve segments are used as primitives, the grammar for classification will be extremely complicated. However, if we use monotonic segments (UP, FLAT, DOWN) as the primitives, with the qualitative attribute of amplitude and duration, the classifier's grammar will be very simple.

10.3 Future Directions

In order to fully test the value of the diagnostic system, there is no substitute for incorporating it into a manufacturing environment from our research environment. Since the diagnostic system is non-intrusive, the set-up disturbance to manufacturing will be minimal.

In Chapter 5, we have discussed the use of two OES signals for endpoints detection. The long-term robustness still needs to be tested in a manufacturing environment. Also, we need to develop rigorous syntactic diagnostic models for both equipment and wafer states, making them applicable to one maintenance cycle or longer.

Since the real-time data waveforms of plasma etch drift constantly due to machine aging, the waveform is significantly different between the beginning and the end of a maintenance cycle. Since the real-time etch waveform reflects the actual etching behavior of the machine, it would be very helpful if we can capture the amount of drift of a plasma etch signal, such that preventive maintenance can be scheduled according to how much the shape of the waveform has changed. Attributed grammar can be used to achieve this. There are two parts to attribute grammar: the qualitative part and the quantitative part. The qualitative part focuses on the rough structural description of the waveform. Loosely speaking, it is the grand human impression on the signal, which we mainly use throughout this thesis for classification purpose. The quantitative part is the numerical measurement of the wave-form attributes, for instance, the amplitude and duration of a peak, the distance between peaks, etc.

Lastly, we should incorporate low frequency signals into the diagnostic system, in addition to the other three existing sources of signals. The author believes that the low frequency signals can provide

valuable diagnostic information about specific parts of the machine, in addition to plasma stability, equipment state and wafer state.

References

[1] Bruha, I.; Madhavan, G. P. "Use of attributed grammars for pattern recognition of evoked potentials," IEEE Transactions on Systems, Man., and Cybernetics. Nov./ Dec. 1988, vol. 18, (no. 6): 1046-9.

[2] Horowitz, S.L. "A syntactic algorithm for peak detection in waveforms with applications to cardiography," Communications of the ACM, May 1975, vol.18, (no.5):281-5.

[3] Koski, A.; Juhola, M.; Meriste, M. "Syntactic recognition of ECG signals by attributed finite automata," Pattern Recognition, Dec. 1995, vol.28, (no.12):1927-40.

[4] Papakonstantinou, G.; Skordalakis, E.; Gritzali, F. "An attribute grammar for QRS detection," Pattern Recognition, 1986, vol.19, (no.4):297-303.

[5] Udupa, J.K.; Murthy, I. S.N. "Syntactic approach to ECG rhythm analysis," IEEE Transactions on Biomedical Engineering, July 1980, vol.BME-27, (no.7):370-5.

[6] Horowitz, S.L. "A syntactic algorithm for peak detection in waveforms with applications to cardiography," Communications of the ACM, May 1975, vol.18, (no.5):281-5.

[7] Belforte, G.; de Mori, R.; Ferraris, F. "A contribution to the automatic processing of electrocardiograms using syntactic methods," IEEE Transactions on Biomedical Engineering, March 1979, vol.BME-26, (no.3):125-36.

[8] Trahanias, P.; Skordalakis, E. "Bottom-up approach to the ECG pattern-recognition problem," Medical & Biological Engineering & Computing, May 1989, vol.27, (no.3):221-9.

[9] Trahanias, P.; Skordalakis, E. "Syntactic pattern recognition of the ECG," IEEE Transactions on Pattern Analysis and Machine Intelligence, July 1990, vol.12, (no.7):648-57.

[10] Skordalakis, E. "Syntactic ECG processing: a review," Pattern Recognition, 1986, vol.19, (no.4):305-13.

[11] Antonacopoulos, A.; Economou, A. "A structural approach for smoothing noisy peak-shaped analytical signals," Chemometrics and Intelligent Laboratory Systems, 6 July 1998, vol.41, (no.1):31-42.

[12] Skordalakis, E. "Recognition of noisy peaks in ECG waveforms," Computers and Biomedical Research, June 1984, vol.17, (no.3):208-21.

[13] Cleveland, W. S. "Robust locally weighted regression and smoothing scatterplots," Journal of the American Statistical Association, Dec. 1979, vol 74, (no. 368): 829-836.

[14] Lee, S.; Spanos, C. "Prediction of wafer state after plasma processing using real-time tool data," IEEE Trans. on Semiconductor Manufacturing, Aug 1995, Vol 8, (no 2): 252-61.

[15] Lee, S. "Semiconductor Equipment Analysis and Wafer State Prediction System Using Real-time Data," PhD. dissertation, December 1994.

[16] Huang, Herbert W. "Adaptive and Predictive Modeling for Real-Time Statistical Process Control," M.S. thesis, 1996.

[17] Brillinger, D. Time Series: Data Analysis and Theory, Expanded Edition, Holden-Day, San Francisco, California, 1981.

[18] Brillinger, D. "The Digital Rainbow: Some History and Applications of Numerical Spectrum Analysis," The Canadian Journal of Statistics, 1993, Vol. 21, (no. 1):1-19

[19] Rompelman, O et al. "Heart Rate Variability in Relation to Psychological Factors," Ergonomics, 1980, Vol. 23, (no. 12):1101-1115

[20] Fu, K. S. Syntactic Pattern Recognition and Applications, Prentice-Hall, Englewood Cliffs, New Jersey, 1982.

[21] You, K. C. and Fu, K. S., "A Syntactic Approach to Shape Recognition Using Attributed Grammars," IEEE Trans. Syst. Man Cybern. SMC-9, June 1979:334-45

[22] Nadler, M.; Smith, E. P. Pattern Recognition Engineering, Wiley, New York, 1993.

[23] Lieberman, M. and Lichtenberg, A., Principles of Plasma Discharges and Materials Processing, John Wiley and Sons, New York, 1994.

[24] Chen, R. "OES-base Sensing for Plasma Processing in IC Manufacturing," PhD. dissertation, December 1997.

[25] SIA Semiconductor Industry Association, The National Technology Roadmap for Semiconductors, 1997 Ed.

[26] White, D.A.; Boning, D.; Butler, S.W.; Barna, G.G. "Spatial characterization of wafer state using principal component analysis of optical emission spectra in plasma etch," IEEE Transactions on Semiconductor Manufacturing, Feb. 1997, vol.10, (no.1):52-61.

[27] White, D.A.; Goodlin, B.E.; Gower, A.E.; Boning, D.S.; Chen, H.; Sawin, H.H.; Dalton, T.J. Low open-area endpoint detection using a PCA-based T/sup 2/ statistic and Q statistic on optical emission spectroscopy measurements. IEEE Transactions on Semiconductor Manufacturing, vol.13, (no.2), IEEE, May 2000. p.193-207.

[28] Baker, M.D.; Himmel, C.D.; May, G.S. "In-situ prediction of reactive ion etch endpoint using neural networks," IEEE Transactions on Components, Packaging, and Manufacturing Technology, Part A, vol.18, (no.3), Sept. 1995. p.478-83.

[29] Yue, H.H.; Qin, S.J.; Wiseman, J.; Toprac, A. Plasma etching endpoint detection using multiple wavelengths for small open-area wafers. Journal of Vacuum Science & Technology A (Vacuum, Surfaces, and Films), vol.19, (no.1), AIP for American Vacuum Soc, Jan. 2001. p.66-75.

[30] Yue, H.H.; Qin, S.J.; Markle, R.J.; Nauert, C.; Gatto, M. Fault detection of plasma etchers using optical emission spectra. IEEE Transactions on Semiconductor Manufacturing, vol.13, (no.3), IEEE, Aug. 2000. p.374-85.

[31] Zhao, Dong Wu. "A syntactic methold for analyzing plasma etching signals," M.S. thesis, EECS UCB, July 1999.

[32] Gao, H.Y. "Wavelet Estimation of Spectral Densities in Time Series Analysis," PhD thesis, Statistics UCB, 1993.

[33] Scholz, F.; Hwang, J.M.; Schroder, D.K. "Low frequency noise and DLTS as semiconductor device characterization tools." Solid-State Electronics, vol.31, (no.2), Feb. 1988. p.205-17.

[34] Fritsch, H.; Lucklum, R.; Iwert, T.; Hauptmann, P.; Scholz, D.; Peiner, E.; Schlachetzki, A. "A low-frequency micromechanical resonant vibration sensor for wear monitoring," Sensors and Actuators A (Physical), vol.A62, (no.1-3), Elsevier, July 1997. p.616-20.

[35] Jevtic, M. "Low-frequency noise diagnostic of microelectronic devices," 1995 20th International Conference on Microelectronics. Proceedings. New York, NY, USA: IEEE, 1995. p.219-24.

[36] Gertner, A.G.; Ratnieks, P.N.; Pukinskis, A.P. "Low-frequency signal spectrum analyzer for testing and diagnostics of tape transport mechanism," Automatic Control and Computer Sciences, 1992, vol.26, (no.6):39-46.

[37] Seifert, J.M.; Stietzel, U.; Karner, H.C. "The ageing of composite insulating materials-new possibilities to detect and to classify ageing phenomena with dielectric diagnostic tools," Conference Record of the 1998 IEEE International Symposium on Electrical Insulation, vol 2, New York, NY, USA: IEEE, 1998. p.373-7 vol.2.

[38] Kruger, M.; Feurstein, R.; Filz, A. "New very low frequency methods for testing extruded cables," Conference Record of the 1990 IEEE International Symposium on Electrical Insulation, New York, NY, USA: IEEE, 1990. p.286-9.

[39] Fischer, B.; Kramer, M. "Study of lower-hybrid wave propagation in the presence of low-frequency fluctuations," Journal of Plasma Physics, vol.48, pt.3, Dec. 1992. p.435-52.

[40] Lebeau, D.; Koch, R.; Messiaen, A.M.; Vandenplas, P.E. "ICRH power absorption measurements from analysis of RF sinusoidal low-frequency modulation experiments in TEXTOR," Plasma Physics and Controlled Fusion, vol.37, (no.10), Oct. 1995. p.1141-68.

[41] Henrion, G.; Hugon, R.; Fabry, M.; Scherentz, V. "Reactivity of a DC-pulsed plasma: plasma diagnostics and nitrided sample analysis," Surface and Coatings Technology, vol.97, (no.1-3), Elsevier, Dec. 1997. p.729-33.

[42] Bougdira, J.; Henrion, G.; Fabry, M.; Remy, M.; Cussenot, J.R. "Low frequency d.c. pulsed plasma for iron nitriding," Materials Science & Engineering A, vol.A139, 1 July 1991. p.15-19.

[43] Praburam, G.; Goree, I. "Experimental observation of very low-frequency macroscopic modes in a dusty plasma," Physics of Plasmas, vol.3, (no.4), AIP, April 1996. p.1212-19.

[44] Brodskii, Yu.Ya.; Moldavskii, Yu.E.; Suvorov, E.V.; Fedoseev, L.I.; Chistyakov, V.V. "Measurement of low-frequency plasma turbulence in the T-10 tokamak," Soviet Journal of Plasma Physics, Feb. 1992, vol.18,, (no.2):80-2.

[45] Roth, W.C.; Carlile, R.N.; O'Hanlon, J.F. "Electrical characterization of a processing plasma chamber," Journal of Vacuum Science & Technology A (Vacuum, Surfaces, and Films), vol.15, (no.6), AIP for American Vacuum Soc, Nov.-Dec. 1997. p.2930-7.

[46] Hilder, D.A.; Black, I.A.; Gray, V.N. "The application of ramp and low frequency a.c. voltages to discharge detection," Conference on diagnostic testing of high voltage power apparatus in service, London, UK: IEE, 1973. p.14-19.

[47] Lieberman, M.A.; Lichtenberg, A.J.; Marakhtanov, A.M. "Instabilities in low-pressure inductive discharges with attaching gases," Applied Physics Letters, vol.75, (no.23), AIP, 6 Dec. 1999. p.3617-19.

[48] Weisberg, Sanford, Applied linear regression, 2nd ed., Wiley, New York, 1985.

Appendix A

The Classifer for the High Speed Data

```
/*encoding.c*/
#include <stdio.h>
#include <string.h>
int GetDurCode(int dur){
  if (dur <10)
    return 0;
  else if(dur >= 10 && dur <=30)
    return 1;
  else if(dur > 30)
    return 2;
  printf("Error, wrong duration sign %d\n", dur);
  exit(1);
}

int GetAmpCode(float amp){
  if (amp <0.4)
    return 0;
  else if(amp >= 0.4 && amp <=2)
    return 1;
  else if(amp > 2)
    return 2;
  printf("Error, wrong amp sign %d\n", amp);
  exit(1);
}

main(){
  int idx, dur, slopecode, durcode, ampcode;
  float amp;
  FILE *inf, *outf;
  inf=fopen("forhist.dat.txt", "r");
  outf=fopen("code.list", "w");
  while(fscanf(inf, "%d %d     %f     %d",&idx, &dur, &amp, &slopecode)!=EOF){
    if(idx==1) fprintf(outf, "\n");
    fprintf(outf, "(%d,%d,%d)",slopecode,
              GetDurCode(dur),GetAmpCode(amp));
  }
```

```
  fprintf(outf, "\n");
}

/*classfier.l*/

%{
/* need this for the call to atof() below */
#include <math.h>
        char *fileName;
%}
SYM     "("|")"|","|0|1|2|"-1"
DIGIT   [0-9]
ID      [a-z][a-z0-9]*
ONEORTWO 1|2
AI "("0,2,0")("1,0,2")("-1,0,2")("1,0,1")("-1,1,2")("0,1,0")"
AII  {SYM}
AIII "("1,0,(1|2)")("-1,(0|1),(1|2)")("0,(1|2),0")"
AIV  "("1,(0|1),(0|1)")("-1,(0|1),(0|1)")("0,2,0")"
BI "("0,2,0")("-1,0,1")("1,2,2")("0,1,0")("-1,1,2")("1,1,2")("-1,1,2")"
BII "("1,0,1")("-1,0,1")("1,1,1")("-1,(0|1),(0|1)")"
BIII1 "("0,0,0")("1,(0|1),0")("0,2,(0|1)")"
BIII2 "("0,1,0")("-1,0,0")("0,2,0")"
BIII {BIII1}|{BIII2}

Cop_pos_spike "("1,0,1")("-1,1,1")"
Cop_flat "("0,(0|1),0")"
Cop_down "("-1,(0|1),(0|1)")"
Ccommon "("-1,2,2")"{Cop_flat}?"("1,2,2")("0,1,0")("-
1,1,2")"{Cop_pos_spike}?"("1,1,2")("-1,(0|1),2")"
D "("0,2,0")("-1,0,2")("1,1,0")("0,2,(0|1)")"
Fcommon "("1,1,2")("-1,0,2")("1,1,2")("-1,0,1")("1,0,2")("-1,1,2")"
Hcommon "("0,2,0")("-1,0,2")"{Cop_flat}?"("1,2,2")("-
1,0,1")("1,(0|1),(0|1)")"{Cop_flat}?"("-1,2,2")"
Icommon "("0,2,0")("1,0,2")("-1,1,2")("0,2,(0|1)")"
%%
{AI}{AII}{AIII}{AIV}$     {
            printf( "Type A matched: %s %s\n", yytext,
                    fileName);
            }
{BI}{BII}{BIII}$     {
            printf( "Type B matched: %s %s\n", yytext,
                    fileName);
            }
{SYM}*{Ccommon}{SYM}*$     {
```

```
            printf( "Type C matched: %s %s\n", yytext,
                    fileName);
            }
 SYM}*{Fcommon}{SYM}*$      {
            printf( "Type F matched: %s %s\n", yytext,
                    fileName);
            }
{Hcommon}{SYM}*$      {
            printf( "Type H matched: %s %s\n", yytext,
                    fileName);
            }
{Icommon}{SYM}*$      {
            printf( "Type I matched: %s %s\n", yytext,
                    fileName);
            }
{D}$      {
            printf( "Type D matched: %s %s\n", yytext,
                    fileName);
            }
%%

main( argc, argv )
int argc;
char **argv;
{
++argv, --argc;  /*("-2")+("-1")*0*$      { skip over program name */
if ( argc > 0 ){
yyin = fopen( argv[0], "r" );
fileName=argv[0];
}
else{
fileName="Standard input";
yyin = stdin;
}
yylex();
}
```