



APPROVED FOR PUBLIC RELEASE
DISTRIBUTION UNLIMITED

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

VLSI PUBLICATIONS

AD-A204 317

VLSI Memo No. 88-485
November 1988

EQUIPMENT MODEL FOR THE LOW PRESSURE CHEMICAL VAPOR DEPOSITION OF POLYSILICON

George Henry Prueger

DTIC
ELECTE
FEB 06 1989
S H

Abstract

An equipment model has been developed for the low pressure chemical vapor deposition (LPCVD) of polycrystalline silicon in a horizontal tube furnace. The model predicts the wafer-to-wafer deposition rate down the length of the tube. Inputs to the model include: silane flow rates from three injectors, injector locations, locations of and temperatures at three thermocouples, operating pressure, the number of wafers, wafer diameter, the location of the wafer load, and other physical dimensions of the furnace such as tube length and inner diameter. The model is intended to aid the process engineer in the operation of equipment, including the selection of optimum process parameters and process control based on measured deposition thicknesses. The model is also flexible enough to aid in the design of new equipment.

The one dimensional finite difference model encompasses the convective and diffusive fluxes of silane and hydrogen in the annular space between the wafer load and tube walls. The reaction of silane is modeled with full account taken of the generation and transport of hydrogen. Kinetic and injection parameters in the model were calibrated using a series of nine statistically designed experiments which varied four parameters over three levels. The model accurately predicts the axial deposition profile over the full range of experimentation and demonstrates good extrapolation beyond the range of experimental calibration. The model was used to predict a set of process parameters that would result in the least variation of deposition rate down the tube. The predicted parameters agree well with experimentally determined optimum conditions.

Acknowledgements

Submitted to the Department of Mechanical Engineering, MIT, March 1988, in partial fulfillment of the requirements for the degree of Master of Science in Mechanical Engineering. This work was sponsored in part by the Defense Advanced Research Projects Agency under contract number N00014-85-K-0213 and the Microelectronics and Computer Technology Corporation.

Author Information

Prueger - Current Address: Rocketdyne Division, Rockwell International Corporation,
Mail Code AC51, 6633 Canoga, Canoga Park, CA 91303. (818) 710-3359.

Copyright© 1988 MIT. Memos in this series are for use inside MIT and are not considered to be published merely by virtue of appearing in this series. This copy is for private circulation only and may not be further copied or distributed, except for government purposes, if the paper acknowledges U. S. Government sponsorship. References to this work should be either to the published version, if any, or in the form "private communication." For information about the ideas expressed herein, contact the author directly. For information about this series, contact Microsystems Research Center, Room 39-321, MIT, Cambridge, MA 02139; (617) 253-8138.

EQUIPMENT MODEL FOR THE LOW PRESSURE CHEMICAL VAPOR
DEPOSITION OF POLYSILICON

by

GEORGE HENRY PRUEGER

B.S. Mech. Eng., Tulane University
(1986)

SUBMITTED TO THE DEPARTMENT OF
MECHANICAL ENGINEERING
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE IN MECHANICAL ENGINEERING

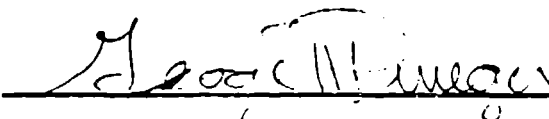
at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

March, 1988

© Massachusetts Institute of Technology 1988

Signature of Author



Department of Mechanical Engineering
March, 1988

Certified by



Emanuel Sachs
Assistant Professor, Mechanical Engineering
Thesis Advisor

Accepted by



Ain A. Sonin
Chairman, Graduate Committee

EQUIPMENT MODEL FOR THE LOW PRESSURE CHEMICAL VAPOR DEPOSITION OF POLYSILICON

by

George Henry Prueger

Submitted to the Department of Mechanical Engineering
on March 28, 1988 in partial fulfillment of the
requirements for the Degree of Master of Science in
Mechanical Engineering

ABSTRACT

An equipment model has been developed for the low pressure chemical vapor deposition (LPCVD) of polycrystalline silicon in a horizontal tube furnace. The model predicts the wafer-to-wafer deposition rate down the length of the tube. Inputs to the model include: silane flow rates from three injectors, injector locations, locations of and temperatures at three thermocouples, operating pressure, the number of wafers, wafer diameter, the location of the wafer load, and other physical dimensions of the furnace such as tube length, and inner diameter. The model is intended to aid the process engineer in the operation of equipment, including the selection of optimum process parameters and process control based on measured deposition thicknesses. The model is also flexible enough to aid in the design of new equipment.

The one dimensional finite difference model encompasses the convective and diffusive fluxes of silane and hydrogen in the annular space between the wafer load and tube walls. The reaction of silane is modeled, with full account taken of the generation and transport of hydrogen. Kinetic and injection parameters in the model were calibrated using a series of nine statistically designed experiments which varied four parameters over three levels. The model accurately predicts the axial deposition profile over the full range of experimentation and demonstrates good extrapolation beyond the range of experimental calibration. The model was used to predict a set of process parameters that would result in the least variation of deposition rate down the tube. The predicted parameters agree well with experimentally determined optimum conditions.

Thesis Supervisor: Dr. Emanuel Sachs

Title: Assistant Professor of Mechanical Engineering

ACKNOWLEDGEMENTS

I would like to thank those people who made this project possible. My research advisor, Dr. Emanuel Sachs, who guided this work, provided intense discussion, and whose enthusiasm about the project made the long hours worthwhile. Roberto Guerrieri, who undertook development of the mathematical and numerical model representations. BTU Engineering Corporation, with special thanks to Mr. Arthur Waugh, Mr. Morris Simson, and Mr. Steve Lai for providing advice, guidance, and access to experimental facilities.

Special thanks go to the Defense Advanced Research Projects Agency under contract N0014-85-K-0213 as well as the Microelectronics and Computer Technology Corporation for support of this project.

Accession For	
NTIS GRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
Dist	
A-1	Special



BIOGRAPHY

George H. Prueger was born in Talara, Peru in 1964. He received a Bachelor's Degree in Mechanical Engineering from Tulane University in May of 1986. Since that time he has been enrolled at the Massachusetts Institute of Technology as a candidate for the Master's of Science Degree in Mechanical Engineering. While at MIT, he has been involved as a Research Assistant with the Computer Aided Fabrication (CAF) Group developing a computer model for the LPCVD of polysilicon.

Contents

Abstract	2
Contents	5
1 INTRODUCTION	7
1.1 Motivation and Background	7
1.2 Related Work	8
2 NEED FOR EQUIPMENT MODELS	10
2.1 Definition and Configuration	10
2.2 Uses for Equipment Models	11
2.3 Construction of Equipment Models	12
2.4 Goal of the Current Work	13
3 MODEL DEVELOPMENT	14
3.1 Modeling Approach	14
3.2 Process Physics	14
3.2.1 Injected Gas Dynamics	15
3.2.2 Axial Mass Transport	15
3.2.3 Radial Mass Transport	16
3.2.4 Heat Transfer	16
3.2.5 Reaction Kinetics	17
3.3 Model Physics	18
3.4 Mathematical Formulation	20
3.5 Numerical Technique	24
3.6 Adjustable Parameters	24
4 EXPERIMENTATION	25
4.1 Motivation	25

4.2	Experiment Design	25
4.3	Procedure	28
4.4	Experimental Process Optimization	29
5	CALIBRATION OF CONSTANTS	33
5.1	Parameter Fit to L_0 Array	33
5.2	Choice of Injector Model	34
5.3	Calibration Check	34
6	MODEL RESULTS	37
7	CONCLUSIONS	45
	References	47
A	APPENDIX A - Numerical Technique	49
A.1	The total velocity equation	50
A.2	The convection-diffusion equation	51
B	APPENDIX B - Polysilicon Model Code	54
B.1	Main Code	55
B.2	bernoulli.f	73
B.3	dgtsl.f	77
C	Appendix C - Measurement Data Sheets	79

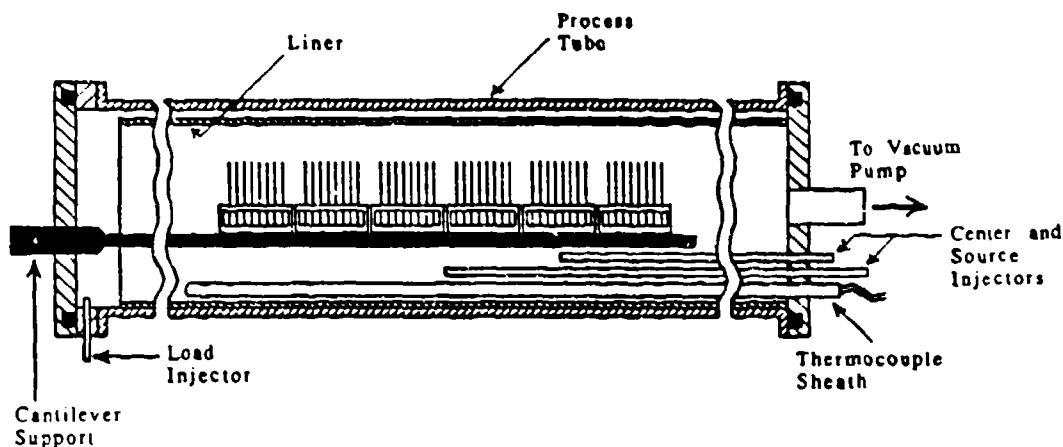


Figure 1: Schematic of LPCVD reactor.

1 INTRODUCTION

1.1 Motivation and Background

Low pressure chemical vapor deposition (LPCVD) of polysilicon in tubular hot-wall reactors has become a wide spread process in the manufacture of semiconductor devices. The proliferations of this process over previous atmospheric cold-wall pressure CVD is due to the ability to process larger quantities of wafers with better uniformities across wafers and wafer-to-wafer uniformities without a significant loss (5 to 10 times less) in growth rates. The low pressures (0.2-1 torr) increase the diffusion coefficient by three orders of a magnitude making the process reaction rate limited. A potential problem is the accumulation of silane at the reactor walls causing particulates and undue stress on the process tube. This problem can be avoided by adding a liner inside the process tube which can be easily removed and replaced for periodic cleaning.

The BTU Engineering/Bruce System 7351C horizontal, hot-wall furnace is a typical commercial reactor used for the LPCVD of polysilicon, schematically shown in Figure 1. The process area consists of a quartz process tube surrounded by a three zone heating coil with a quartz liner inside the process tube. The wafer load is inserted and removed on a silicon carbide cantilever attached to the front door of the reactor. The wafer load rests in quartz boats and is situated concentrically

in the tube so that the wafers are perpendicular to the main gas flow. The vacuum is pulled from the source¹ end of the tube. Gas (pure silane, SiH_4) can be injected through three injectors - fixed load end - and two moveable injectors - which are usually situated in the center and source zones. The temperature of the zoned heater is controlled during deposition by three thermocouples which are situated in a quartz sheath inside the process tube and are individually located near the beginning of the wafer load, in the center, and near the end of the wafer load. These parameters are necessary inputs to an equipment model.

At present, processes are designed by empirical rules. Current process specifications require across wafer uniformities on the order of $\pm 2\%$ and across wafer load uniformities of $\pm 5\%$. Equipment specifications claim that across wafer uniformities of $\pm 1\%$ and across wafer load uniformities of $\pm 3\%$ can be produced. Obtaining the equipment parameter settings to achieve these process specifications is an expensive and time consuming chore based on trial and error experimentation inside a small operating regime in which the equipment and process is well behaved. Investigation of operating regimes outside the empirical "envelope" is a prohibitively expensive task. As device specifications become ever more stringent, process specifications will be more difficult to meet. Equipment models are the most cost effective method to achieve the necessary goals of equipment operators and equipment designers.

1.2 Related Work

Many studies have been conducted to advance the physical understanding of the LPCVD of polysilicon. Hitchman [1] derived a basic linear model of the LPCVD reaction kinetics. Van Den Brekel [2] and Claassen [3] investigated the LPCVD reaction chemistry for polysilicon and developed an understanding for the main effects

¹Throughout this paper the furnace will be referred to in three zones: the load zone, front area of the furnace where the wafers are inserted and removed; the source zone, which is the area at the rear of the tube, and the center zone.

of the reaction chemistry as supported by experimental observations. Middleman [4] developed a numerical model for the mass transport in an annular LPCVD reactor. He showed the effect of including diffusion along with the gas convection and that the flow in the annular region was insufficient to create significant circulating flows between the wafers. The above studies were invaluable to the physical understanding of the LPCVD process.

The most complete equipment models developed have been based on two very different premises: numerical solutions of the system physical representation; and expert systems mainly based on empirical rules. A numerical model for the LPCVD of polysilicon was developed by Jensen [5,6]. This model embodies the most advanced reaction kinetic model derived from the studies of Van Den Brekel [2] and Claassen [3] with a gas flow model dependent on the system geometry. The model predicts the axial deposition profile for polysilicon deposition for a ramped temperature processes in which there is no gas injection other than at the load end of the furnace. It is important to note that, although the model is able to predict the axial profiles for which it was tested, the model and data were not run with the same equipment parameter settings. The expert system approach most recently developed at Berkeley [7] for the LPCVD of doped polysilicon. The expert system partitions the deposition process goals into six modules for determination of the resistivity, thickness, uniformity, grain size, film stress, and a support module. Each individually search a data base of empirical rules for the correct equipment settings to meet the process specifications. The expert system is limited to operation in a small operating window in which it has empirical knowledge and would benefit from an inclusion of equipment noise parameters such as temperature fluctuations or gas flow variations for uniformity analysis.

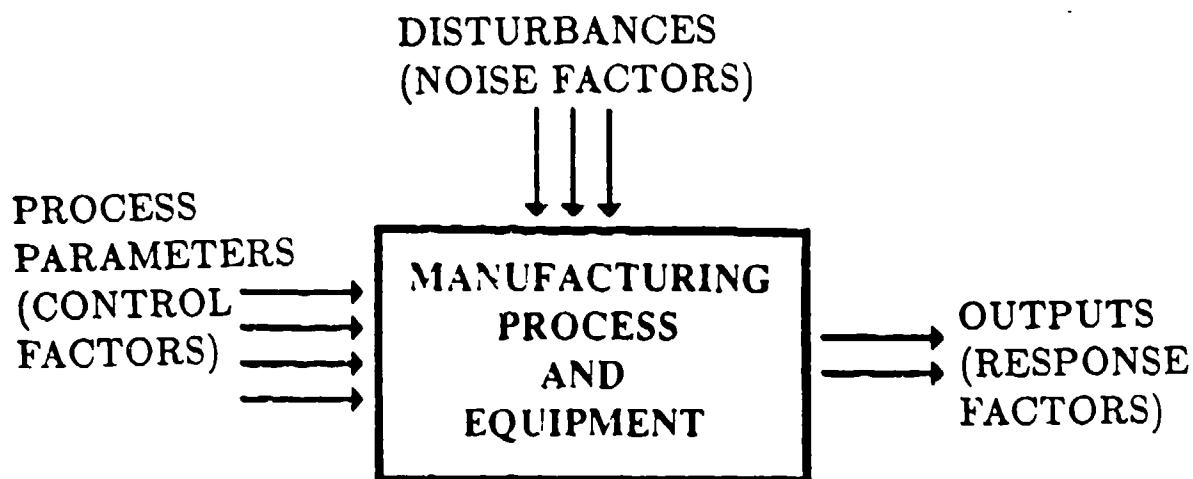


Figure 2: Generic representation of an equipment model.

2 NEED FOR EQUIPMENT MODELS

2.1 Definition and Configuration

In the most general sense, an equipment model is a body of knowledge which provides predictions about the outputs from a unit manufacturing process, given information about the inputs to the process. Figure 2 illustrates a generic equipment model with outputs and two classes of inputs, the process parameters and the disturbances. The process parameters are those parameters that we exercise direct control over, for example, temperature, pressure, gas flow rates. The disturbances are those inputs to the process which are subject to unintended and undesired variations. In some cases, the magnitude of the disturbances can be monitored, while in other cases they cannot. Examples of disturbances include variations in the properties of incoming material and variations in the process parameters themselves. The alternate labels presented in parentheses in Figure 2, (control factors, noise factors, response factors) are the corresponding terminology used by Taguchi [8] to describe manufacturing processes.

At a minimum, a competent equipment model must provide information about

the outputs given information about the process parameters. A more useful model, will also include predictions about the effect of disturbances. It is vital that the model provide information about the variation of the outputs and the dependence of this variation on the process parameters and the disturbances. The variations might include across the wafer variation, wafer to wafer variation, and batch to batch variation. A competent model might address all three classes of variation, or might focus on the most important class as indicated by experience. Accurate predictions about the process mean are in general less important than accurate predictions about variation, since in most processes the mean can be adjusted to its target value without a substantial defect on variation. An example of such an adjustment would be the length of time in an LPVCD deposition. Occasional exceptions to this generalization will require that the model accurately predict the mean as well.

2.2 Uses for Equipment Models

Equipment models can be used to aid in the operation of existing equipment, or in the design of new equipment. The two classes of uses, will require models of somewhat different construction.

In the area of operations, equipment models can be used to optimize the operation of a process. Typically, the model would be used to find a set of process parameters which result in the least variation of the outputs. Such optimization procedures might be used in lieu of experimental optimization, but would more typically be used to focus attention on the most suitable range of operation with final adjustments performed experimentally. If the equipment model includes predictions of the effective disturbances, the optimization procedure using the model can include minimization of the effect of the disturbances. This is an especially valuable approach to optimization, since including disturbance factors in an experimental program would involve an enormous number of experimental points.

Equipment models can also be used for on-line quality control or process control.

Having selected the operation point in the optimization procedure, the model can now serve to guide adjustments made locally around the operating point. For example, as an LPCVD tube drifts with build up from successive runs, the model can be used to predict the changes in gas flow rates needed to bring the results back as close as possible to the target.

A model which captures the effect of internal parameters such as geometric dimensions, and choice of materials, can be used as a simulation tool for the design of new equipment. Such a model can substantially reduce the development time for new equipment.

2.3 Construction of Equipment Models

Equipment models may be constructed by two distinctly different approaches: physically based mechanistic modeling and statistical modeling. Each approach has its distinct advantages. Physically based models have the advantage of broad applicability, good extrapolation beyond the range of experimental verification, and good prediction of process sensitivities. Statistical modeling has the advantages of ease of application and good absolute accuracy within the range of measurement.

Physically based models may be either closed form or numerical (finite element methods, boundary element methods) in nature. Statistical models are most effectively developed using techniques of statistical experimental design, such as Box "factorial experimental design and response surfaces" [9] and Taguchi "orthogonal array" [8]. The unifying feature of designed experiments is that all the parameters of interest are varied simultaneously, in contrast to the more conventional one variable at a time experimental techniques. In this way, the total experimental range is explored with a minimum number of experiments.

2.4 Goal of the Current Work

In current practice, the two approaches to model construction discussed above have been followed independently. The broad goal of this work is to fuse the two methods and gain the benefits of both.

In the current work, we have constructed a physically based model using finite difference numerical methods. The model has four adjustable coefficients embedded in it which represent areas of uncertainty about the physics. These coefficients have been calibrated using a series of statistically designed experiments in order to assure wide range of application of the model with the minimum number of experiments.

3 MODEL DEVELOPMENT

3.1 Modeling Approach

The model consists of a one-dimensional representation of the LPCVD reactor for polysilicon. The inputs to the model include: the physical dimensions of the process area (i.e. liner diameter, process tube length, injector diameter); diameter of the process wafers; position of the wafer load; flow rates from the three injectors; positions of two of the injectors; temperatures and positions for each of the profile thermocouples. The axial temperature profile is determined by a linear interpolation of the temperatures for each of the thermocouple sites. The axial flow is modeled with a convection-diffusion representations incorporating area changes due wafer load and a laminar, plug flow velocity distribution.

The model predicts the axial deposition profile of the polysilicon. This solution is calculated using a Newton-Raphson method on the center-difference numerical representation of the one-dimensional system.

The following sections describe the analysis behind the assumptions in the model and the manner in which it was constructed.

3.2 Process Physics

The process physics can be likened to a coupling between a number of physical mechanisms:

- introduction of gas into the furnace
- mass transport in the axial flow direction
- mass transport between the wafers
- heat transfer
- chemical reaction at the hot surfaces

3.2.1 Injected Gas Dynamics

The injector exit gas flow is a highly complex problem. Due to the small injector diameter (~ 4 mm) the velocities from the injectors are on the order of $0.5 \times \text{Mach } 1$. It has also been shown (by a Poiseuille flow approximation in the injector) that there is a large pressure drop at the injector exit. This indicates that there is an extreme expansion wave at the exit of the injector [10]. From analysis of the Reynolds number for the exit jet, the flow was shown to be turbulent ($\text{Re} \sim 37$). The physical understanding of these systems is limited.

3.2.2 Axial Mass Transport

The relative importance of the convective flux to the diffusive flux is captured by the Peclet number, $\text{Pe}_a \equiv \frac{Vd}{D}$. Where V is the estimated average gas velocity in the in the annular region ($V \sim 200$ cm/sec), d is a relative length on the order of the furnace length ($d \sim 200$ cm), and D is the diffusion coefficient at the operating temperature and pressure ($D \sim 6000$ cm²/sec). Since this analysis shows the Peclet number on be of the order of one, both the convective and diffusive fluxes must be represented in the model.

An indication of the flow regime can be determined by the the Reynolds number, $\text{Re} \equiv \frac{\rho V d}{\mu}$. Where V is the estimated average gas velocity in the annular region ($V \sim 200$ cm/sec), ρ is the gas density ($\rho \sim 2 \times 10^{-4}$ kg/m³), d is a relative length on the order of the furnace length ($d \sim 200$ cm), and μ is the gas viscosity ($\mu \sim 3 \times 10^{-5}$ kg/m sec). The Reynolds number was found to be on the order of one. This indicates that a laminar low approximation can be made.

An analysis was conducted to estimate the effect of the volumetric increase of gas due to the chemical reactions at the hot surfaces. The analysis consisted of a hybrid Poiseuille flow approximation in the annular region accounting for possible slip flow conditions at the boundaries [11]. The analysis showed that there is a negligible pressure drop down the length of the furnace. The pressure drop was on

the order of 0.1% of the total pressure (~ 250 mtorr). This means that a constant pressure assumption in the tube is valid.

3.2.3 Radial Mass Transport

The Peclet number can be used to indicate the relative importance of the convective and diffusive terms in the radial direction by changing the relative order of the length to that of the liner diameter (~ 21 cm) and estimating a radial velocity between the wafers. Due to the symmetry of the wafer loading in the furnace any convective flow between the wafers would be very small. From this analysis it can be seen that the Peclet number is much less than one ($Pe_r \ll 1$), indicating that convective transport is negligible as compared to the diffusive transport.

Another method of analysis of the radial mass transport effect can be done through the Sherwood number, $Sh \equiv \frac{kd}{D}$. Where k is the mass transport coefficient of the reaction from the Arrhenius reaction dependence ($k \sim 8 \times 10^{-8}$ cm/sec), d is a relative length on the order of the tube diameter ($d \sim 20$ cm), and D is the diffusion coefficient ($D \sim 6000$ cm²/sec). The Sherwood number is also much less than one ($Sh \ll 1$). This indicates that growth rate is reaction rate limited, or that the diffusion time for the gases between the wafers is much smaller than the reaction time at the wafer surface, indicating that the radial gas concentration can be considered uniform.

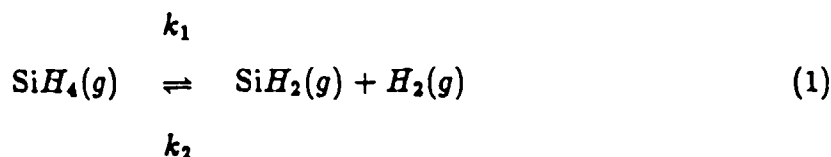
3.2.4 Heat Transfer

An order of magnitude analysis for the relative importance of radiative to convective heat transfer can be captured for small temperature variations as: $\frac{\sigma T^3 d}{8k}$. Where σ is the Stefan-Boltzmann's constant (5.67×10^{-8} W/m² °K⁴), T is the operating temperature ($T \sim 898^\circ\text{K}$), d is a relative length on the order of the tube diameter ($d \sim .214$ m), and k is the thermal conductivity of the gas ($k \sim 2 \times 10^{-2}$ W/m K). The analysis show that $\frac{\sigma T^3 d}{8k} \sim 42$ indicating that radiation is the dominant mode

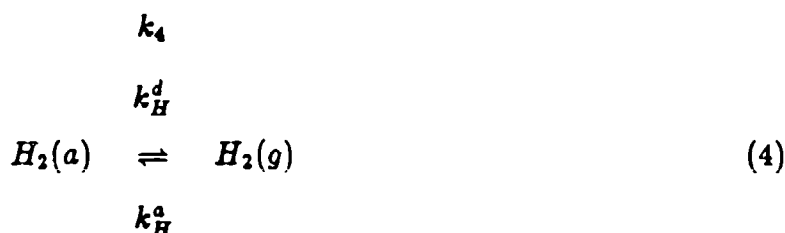
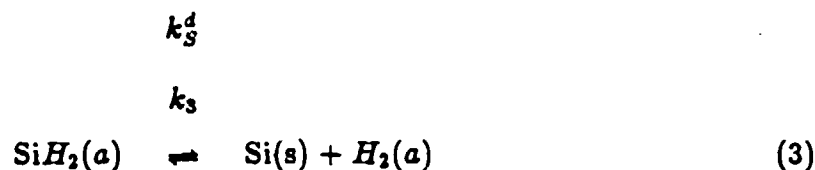
of heat transfer. This allows for an isothermal assumption in the radial direction.

3.2.5 Reaction Kinetics

The reaction kinetics involved in the pyrolysis of silicon from pure silane has been studied [2,3] and the general consensus has been that the silane, SiH_4 , breaks down in the gas phase as:



The SiH_2 then adsorbs on the hot surfaces, silicon is deposited, and the by-product, hydrogen, desorbs:



The reaction equation for the deposition of silicon from silane describing the above chemical reaction equations and substantiated by experimental evidence was formulated by Roenigk and Jensen [6] as:

$$\mathcal{R} = \frac{k_1 X_{\text{SiH}_4} C_{\text{tot}}}{1 + K_s X_{\text{SiH}_4} C_{\text{tot}} + K_h \sqrt{C_{\text{tot}} (1 - X_{\text{SiH}_4})}} \left[\frac{\text{molSi}}{\text{m}^2 \text{sec}} \right] \quad (5)$$

Where X_{SiH_4} is the molar fraction of silane, $(1 - X_{\text{SiH}_4})$ represents the molar fraction of hydrogen, C_{tot} is the total concentration of the gases, K_s and K_h are the equilibrium coefficients (adsorption to desorption) of silane and hydrogen and indicate

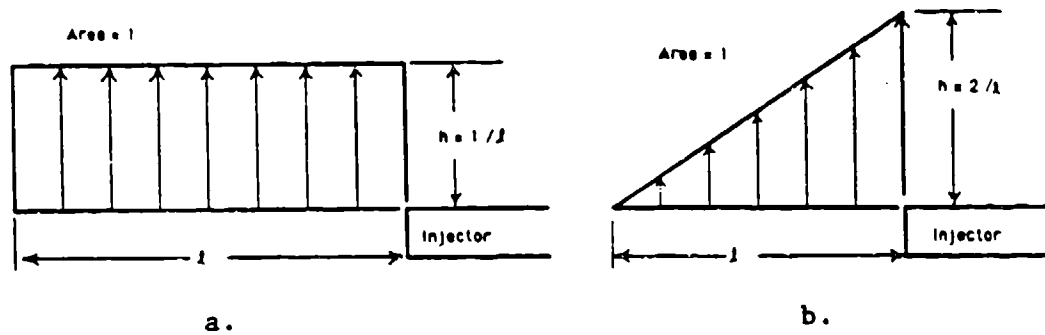


Figure 3: Representation of the source and center injector empirical formulations:
a. Flat model. b. Ramp model.

the effect of silane and hydrogen concentrations on the silicon deposition, and k_1 is an Arrhenius reaction rate dependence represented as:

$$k_1 = k_0 \exp \left(\frac{\Delta E_A}{RT} \right) \quad (6)$$

Where k_0 is the Arrhenius reaction constant, ΔE_A is the activation energy, R is the universal gas constant, and T is the temperature. This is the major contributor to the effect of temperature on the reaction.

3.3 Model Physics

The model captures the dominant characteristics of the afore mentioned physical mechanisms and the physical representation of the system.

Gas is injected through three injectors. The load injector, which is directed in the flow direction, is modeled by a constant concentration distribution across the tube cross-sectional area. This is possible because there are no wafers in the load region and the gas will diffuse rapidly in the furnace. The two counter-flow injectors, center and source, are modeled by incorporating as much physical knowledge of the flow pattern as possible. Two empirical flow models were proposed: a flat injector model, Figure 3a, and a ramp model, Figure 3b. The ramp model is more physically representative because it incorporates the higher concentration of the gas at the

injector exit which would result form an expansion wave. The flat injector model gives an average approximation of the occuring exit jet phenomena. Choice of the best injector spray model will be discussed in Section 5.2.

Due to the complexity of the flow physics the lenght of the injector spray was left as an adjustable parameter:

$$l = \text{con} \Delta Q_{\text{injected}} \quad (7)$$

Where Q_{injected} is the flow rate of the gas from the injector, con is an adjustable parameter, Δ is the wafer spacing, and l represents the effective distance of the gas spray. A linear relationship between the spray length and the flow rate was chosen since the exact physics was too complex to determine the proportionality and a linear relationship is more stable than other possible relationships.

Incorporation of the injector model into the LPCVD model was done by integrating a silane generation term, $\mathcal{F}$, over a transversal section of the tube to give an empirical, one-dimensional injector function:

$$F(z) = \int_{L_t} \mathcal{F} dL_t \quad (8)$$

where L_t is the perimeter of the tube. This gives a one-dimensional embodiment of the gas injection per unit length of tube area.

The axial mass transport in the furnace was modeled by a convection-diffusion flow representation with a laminar, plug flow velocity distribution. The basis for this representation arises from the analysis of the axial Peclet number ($Pe_a \sim 1$) and the Reynolds number ($Re \sim 1$).

The radial mass transport is neglected ($Pe_r \ll 1$ and $Sh \ll 1$). This allows a one-dimensional numerical analysis. Meaning that the gas concentration at each axial position in the furnace has a constant radial gas concentration.

Since the heat transfer is radiation dominated, as previously shown, an isothermal assumption can be made in the radial direction. This decouples the temper-

ature from the model equations since only the axial temperature will effect the diffusion coefficients of the gases and the gas densities.

The reaction model, Equation 5, is included as a silane depletion term in the silane flux equation. The model is combined with a reaction area per unit length term which includes the depletion mass from the wafer load and the hot-wall surfaces. This reaction area per unit length term was found by integrating the areas of the wafer surface and the cylindrical wall surface:

$$R(z) = \frac{2}{\Delta} \left(\int_{A_w} \mathcal{R} dA_w + \int_{A_t} \mathcal{R} dA_t \right) \quad (9)$$

where A_w is the area associated with the surface of the wafer and A_t is the area associated with a cylinder having a length $\Delta/2$, where Δ is the wafer spacing, and having a radius equal to the radius of the tube. The evaluation of the integrals give:

$$R = (r_w^2 + r_t \Delta) \frac{2\pi}{\Delta} \mathcal{R} \quad (10)$$

which can be written in a compact form as:

$$R = L_r \mathcal{R} \quad (11)$$

where L_r is the effective reaction length defined as $\frac{2\pi}{\Delta} (r_w^2 + r_t \Delta)$. This is the one-dimensional depletion term used in the model.

3.4 Mathematical Formulation

The model was formulated based on the convective and diffusive fluxes of the silane and hydrogen gases and the stoichiometry of the chemical reaction:

$$\begin{aligned} \nabla (\bar{N}_{SiH_4}) &= 0 \\ \nabla (\bar{N}_{H_2}) &= 0 \end{aligned} \quad (12)$$

Where $\bar{N}_{SiH_4}$ and $\bar{N}_{H_2}$ are the hydrogen and silane molar flows. Proper boundary conditions are coupled to the above equations.

Assuming that z is the coordinate along the tube and $\hat{n}$ is an inward normal vector to the generic surface, the boundary conditions for silane are:

- at the wafer surface

$$\vec{N}_{SiH_4} \cdot \hat{n} = -\mathcal{R} \quad (13)$$

- at the hot wall:

$$\vec{N}_{SiH_4} \cdot \hat{n} = -\mathcal{R} + \mathcal{F}(\bar{z}) \quad (14)$$

where the term $\mathcal{R}$ is as formulated in Equation 5 and $\mathcal{F}(\bar{z})$ takes empirically into account that silane is injected into the furnace as discussed in Section 3.2.1.

- at the load zone:

$$\vec{N}_{SiH_4} \cdot \hat{n} = \tilde{C}_0 \vec{V}_0 \cdot \hat{n} \quad (15)$$

where V_0 is the gas velocity of the gas injected by the load injector and $\tilde{C}_0$ is the molar density of the injected gas;

- and assuming that at the end of the tube there is no chemical reaction:

$$\frac{\partial C_{SiH_4}}{\partial z} = 0 \quad (16)$$

Similarly, the boundary conditions for hydrogen are:

- at the wafer surface and at the hot wall:

$$\vec{N}_{H_2} \cdot \hat{n} = 2\mathcal{R} \quad (17)$$

where the $2\mathcal{R}$ results from the stoichiometry of the chemical reaction in which for the every silicon molecule deposited two hydrogen molecules are generated;

- at the load zone:

$$\vec{N}_{H_2} \cdot \hat{n} = 0 \quad (18)$$

- and assuming that at the end of the tube there is no chemical reaction:

$$\frac{\partial C_{H_2}}{\partial z} = 0 \quad (19)$$

Next, the fluxes in Equation 12 can be combined as : $\vec{N}_{tot} = \vec{N}_{SiH_4} + \vec{N}_{H_2}$, giving:

$$\nabla (\vec{N}_{tot}) = 0 \quad (20)$$

With the boundary conditions defined by the linear combination of the afore mentioned boundary conditions.

Assuming now that:

$$\vec{N}_{SiH_4} = -C_{tot} D \vec{\nabla} X_{SiH_4} + C_{tot} \left(\frac{\vec{N}_{tot}}{C_{tot}} \right) X_{SiH_4} \quad (21)$$

and

$$\vec{N}_{H_2} = -C_{tot} D \vec{\nabla} X_{H_2} + C_{tot} \left(\frac{\vec{N}_{tot}}{C_{tot}} \right) X_{H_2} \quad (22)$$

where $\vec{N}_{tot}/C_{tot} = \vec{V}$ is the molar average velocity of the gas and $C_{tot} = C_{SiH_4} + C_{H_2}$. The model, coupled with the boundary conditions previously introduced, can be stated as:

$$\begin{aligned} \nabla (-C_{tot} D \vec{\nabla} X_{SiH_4} + C_{tot} \vec{V} X_{SiH_4}) &= 0 \\ \nabla (-C_{tot} D \vec{\nabla} X_{H_2} + C_{tot} \vec{V} X_{H_2}) &= 0 \end{aligned} \quad (23)$$

This model consists of a system of elliptic partial differential equations which can be solved numerically.

Equation 23 can be reformulated to take advantage of more efficient numerical techniques. Since:

$$\nabla (C_{tot} \vec{V}) = 0 \quad (24)$$

and with the proper boundary conditions:

$$\nabla (-C_{tot} D \vec{\nabla} (X_{SiH_4} + X_{H_2}) + C_{tot} \vec{V} (X_{SiH_4} + X_{H_2})) = \nabla (C_{tot} \vec{V}) \quad (25)$$

It follows that the solution can be obtained in terms of one of the two concentrations and the molar average velocity. The problem can thus be formulated as:

$$\nabla (C_{tot} \vec{V}) = 0 \quad (26)$$

and

$$\nabla \left(-C_{tot} D \vec{\nabla} X_{SiH_4} + C_{tot} \vec{V} X_{SiH_4} \right) = 0 \quad (27)$$

From the assumptions found in Section 3.2 the equations can be simplified to one-dimensional form in the axial direction. Since, in the one-dimensional approximation, the gas flow is uniform across any radial area, Equations 27 and 26 can be stated as:

$$-\frac{d}{dz} \left(A_{tw} D C_{tot} \frac{dX_{SiH_4}}{dz} + A_{tw} C_{tot} \frac{d\psi}{dz} X_{SiH_4} \right) = -(R - F(z)) \quad (28)$$

and

$$-\frac{d}{dz} \left(A_{tw} C_{tot} \frac{d\psi}{dz} \right) = (R + F(z)) \quad (29)$$

where A_{tw} is the cross sectional flow area defined according to the position in the tube and

$$-\frac{d\psi}{dz} = V \quad (30)$$

is a potential flow approximation of the molar average velocity used for numerical efficiency.

The boundary conditions associated with Equation 29 can be stated as:

$$-A_{tw} \big|_{z=0} C_{tot}(z=0) \left(\frac{d\psi}{dz} \right)_{z=0} = Q_0 \quad (31)$$

and

$$\psi(L) = 0 \quad (32)$$

The first condition states that at the load zone the total gas velocity is determined by the amount of silane introduced and the second condition sets the reference value for the potential. The boundary conditions associated with Equation 28 are:

$$-\left(A_{tw} C_{tot} D \frac{dX_{SiH_4}}{dz} + A_{tw} C_{tot} \frac{d\psi}{dz} X_{SiH_4} \right)_{z=0} = Q_0 \quad (33)$$

and

$$\left(\frac{dX_{SiH_4}}{dz} \right)_{z=L} = 0 \quad (34)$$

Which again state that the silane flow at the load zone is determined by the amount of silane introduced through the load injector and that at the very end of the furnace the silane concentration is constant. In these equations L is the length of the tube and Q_0 is the quantity of SiH_4 introduced at the front of the tube for each time unit.

3.5 Numerical Technique

The discretization of the equations consisted of a center difference method over a one dimensional grid which consisted of $N+1$ grid points having coordinates z_i , $i = 1, N+1$. Equations 28 and 29 were decoupled and then linearized using the Newton-Raphson method, Appendix A. The resulting discretized equations were arranged as tri-diagonal matrices and solved using a standard solving package. Included in Appendix B is the code of the polysilicon model. The model is written in C with two Fortran subroutines.

3.6 Adjustable Parameters

There are four adjustable parameters in the model which will be fit to experimental data collected from a designed experiment.

Three of the constants are in the reaction model depicted in Equation 5. They are k_1 , K_s , and K_A . These parameters are physical constants which arise from the chemical reaction for the deposition of polysilicon. The Arrhenius type dependency of the reaction rate is found in k_1 , where k_0 from Equation 6 is to be extracted from experimental data. The effects of the silane and hydrogen adsorption and desorption on the growth rate are quantified by K_s and K_A , respectively.

The fourth constant is in the empirical injector function for the spray length, Equation 7.

Experiment Number	Pressure (micorr)	Q_{load} (% of total)	Q_{center} (% of total)	X_{source} (% of tube length from center)
1	200	20	26.7	9
2	200	30	36.7	12
3	200	40	46.7	15
4	250	20	36.7	15
5	250	30	46.7	9
6	250	40	26.7	12
7	350	20	46.7	12
8	350	30	26.7	15
9	350	40	36.7	9

Figure 4: Taguchi L_9 array.

4 EXPERIMENTATION

4.1 Motivation

Careful consideration must be given to the design of a set of experiments. The purpose of the experiments must be clearly defined so that useful results are obtained. The experiments in this work were necessary to calibrate and test the numerical equipment model. Therefore, the design of the experiments had to investigate a large operating space so that the model would be as accurate as possible. A second consideration was the expense of each experiment. It was thus necessary to design a set of experiments which would take into account both of the above considerations.

4.2 Experiment Design

A set of experiments was designed to calibrate and test the accuracy of the model. The Taguchi L_9 orthogonal array, developed by Genechi Taguchi [8], was chosen for this experimental design, Figure 4. This design allowed the investigation of a

large operating space, with four equipment parameters at three levels, in only nine experiments. The limitations of this design were that the four parameters would have to be independent of each other and that factor interactions could not be studied from the results. These are not inhibiting limitations. Independent factors can always be found or interacting factors can be combined into one independent factor [12].

The choice of equipment parameters to be used as experiment factors was also important. The model was developed to predict the axial deposition profile. To verify the model, it was thus necessary to choose equipment parameters which had a dominant effect on the axial deposition profile.

There are many equipment parameters which effect the axial growth profile. These include:

1. pressure variations
2. flow rates from each of the three injectors
3. temperature variations
4. off center of the cantilever or liner alignment, which would create a static pressure differential across the wafers promoting convective flow between the wafers
5. amount of prior deposition in the furnace
6. prior condition of the wafers
7. wafer thickness variations, causing temperature non-uniformities within the wafers
8. thermocouple aging
9. wafer spacing

10. exterior conditions, including humidity and temperature

Before the equipment parameters to be used as experiment factors were chosen it was necessary to determine the effect of equipment noises on the deposition profile. A set of parameter settings was found which gave a relatively flat profile. These settings were the base line equipment settings. The furnace was run with these settings for a number of runs which spanned approximately 100,000 Å of deposition on the furnace walls. After a clean of the equipment, the base line was run again. This set of experiments indicated that the repeatability of the equipment was not greatly effected by equipment noises. Figures of the measured data include error bands for the experimental growth rates as determined from these experiments.

From these experiments, the four parameters considered to be the most dominant as to affecting the axial growth rate and allowed ease of control by the operator were: (1) pressure, (2) temperature, (3) gas flow rates from the three injectors, and (4) positions of the two moveable injectors.

This gave seven parameters which could be varied. The L_9 design only allowed four. To quantify the the Arrhenius and equilibrium coefficients in the reaction model, Equation 5, the pressure was a necessary parameter. The second consideration was that the parameters had to be independent of each other and the physical limitations of the equipment had to be kept in mind. It was necessary to choose a total flow rate which would allow the desired variation in the pressure. Based on experience of the BTU Engineering staff, a flow rate of 150 sccm was chosen as the total flow. This meant that the three injector flow rates would be coupled. The load and center injector flow rates were chosen as parameters, thereby fixing the source injector flow rate. The fourth parameter chosen was the source injector position. Since the position of this injector effects the axial profile uniformity more than the center injector position. The factors thus chosen were:

- pressure

- load injector flow rate (Q_{load})
- center injector flow rate (Q_{center})
- position of source injector (X_{source})

Factor levels for the experiment factors were chosen based on the base line settings. The high and low values for the levels were determined, based on the prior experience of the BTU Engineering staff, to give a large representation of the operating space for the furnace.

An independent set of experiments was conducted using flow through the load injector only. These experiments were conducted to investigate the parameter in the reaction model with the maximum amount of concentration variation. These experiments are discussed in Section 5.3.

4.3 Procedure

The experiments were conducted in a commercial BTU Engineering/Bruce Systems 7351C horizontal, hot-wall reactor with three zone temperature control. The equipment consisted of a 230/240 mm x 88.5 inch quartz process tube fitted with a 214/220 mm x 79 inch quartz liner. The furnace had a three zone heater with a 32 inch flat zone. The total wafer load consisted of 150, 6 inch wafers, in six, 6 inch, 50 wafer quartz boats with 3/32 inches center to center spacing. Only 25 wafers were inserted per boat giving a wafer spacing of 3/16 inches center to center. The first and last boats were dummy boats. The production load thus consisted of 100 wafers in the center four boats.

The recipe used to run the furnace was the recipe which BTU recommends for furnace operation when running a flat polysilicon process. The experiments were run by varying parameters in the recipe according to the L_9 experiment design structure. The deposition time in each experiment was 75 min. Thirteen test wafers

were inserted at locations 20, 26, 35, 45, 55, 65, 75, 85, 95, 105, 115, 124, and 130 in the wafer load of 150 wafers (wafers 20 and 130 where in the dummy load).

These wafers were then measured with a nanospec and an ellipsometer. Measurements were made at the top, bottom, center, left, and right of the wafers. The average of these measurements was the growth attributed to each of the wafers.

4.4 Experimental Process Optimization

The experiment factors from the L_9 design were optimized according to Taguchi's signal-to-noise ratio (SN) criteria for "nominal is best."

$$SN = 10 \log \left(\frac{\text{mean}^2}{\text{variance}} \right) \quad (35)$$

The SN is calculated by determining the mean of the axial deposition profile and calculating the variance of the measured data from this mean. The "nominal is best" SN is used because the optimum deposition profile would be flat with the least variance about the mean. This criteria for the SN gives a measure of the relative flatness of the profile. The assumption is that the mean can be scaled to the desired value. In this case, the mean can be scaled by time to obtain the desired amount of total deposition.

Table 1 shows the calculated signal-to-noise ratios for the L_9 . Figure 5 and Figure 6 indicate the profile differences between the best factor level set (Experiment 2) and the worst factor level set (Experiment 9).

Choice of the optimum factor level settings was determined by averaging the SN for the experiments in which the particular factor level was used. For example, the SN for experiments 1, 2, and 3 were averaged to obtain the relative SN attributed to factor 1 level 1. The optimum settings were the factor levels with the largest SN's. Table 2 gives the optimum parameter levels determined by the analysis and Figure 7 shows the corresponding deposition profile using these settings. This optimum profile is more axially uniform than the previous L_9 experiments and the base line,

Runs	Parameters				S/N
	1	2	3	4	
1	1	1	1	1	23.01
2	1	2	2	2	27.55
3	1	3	3	3	18.61
4	2	1	2	3	21.91
5	2	2	3	1	20.89
6	2	3	1	2	23.24
7	3	1	3	2	24.23
8	3	2	1	3	24.79
9	3	3	2	1	17.30

Table 1: Signal-to-noise ratio results for L_9 design.

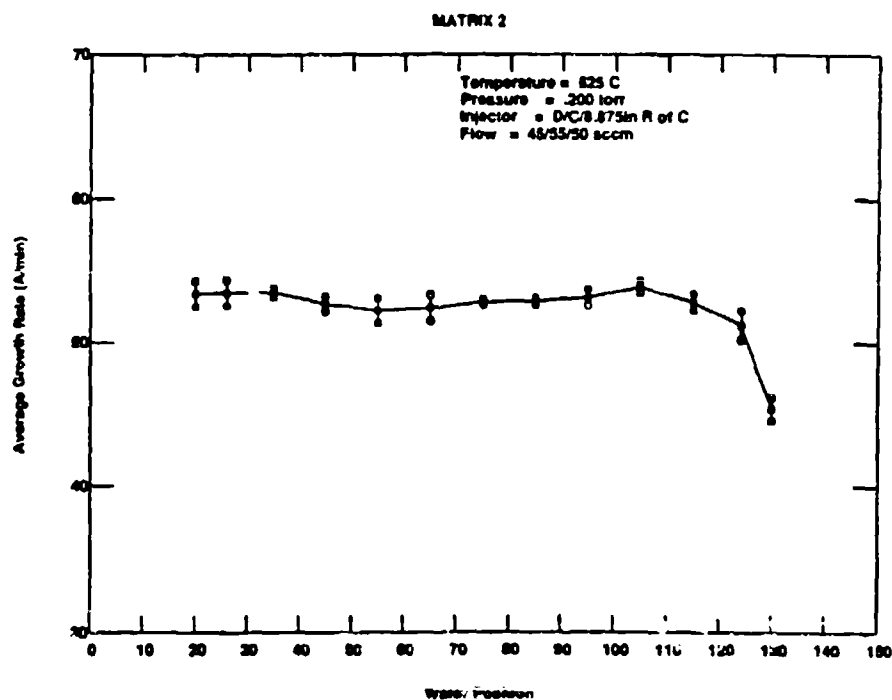


Figure 5: Deposition profile with largest SN for L_9 array.

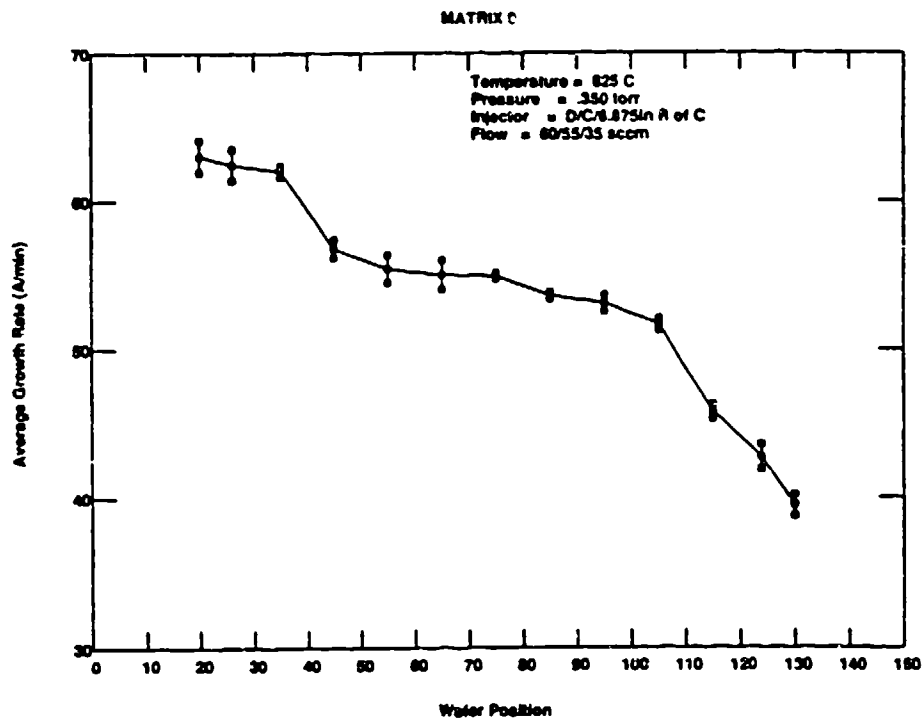


Figure 6: Deposition profile with Lowest SN for L_0 array.

Average Parameter S/N				
Level	(1) Pressure	(2) Q_L	(3) Q_C	(4) X_S
1	23.05	23.05	23.68	20.40
2	22.02	24.41	22.25	25.01
3	22.11	19.72	21.24	21.77

Table 2: Optimum factor levels.

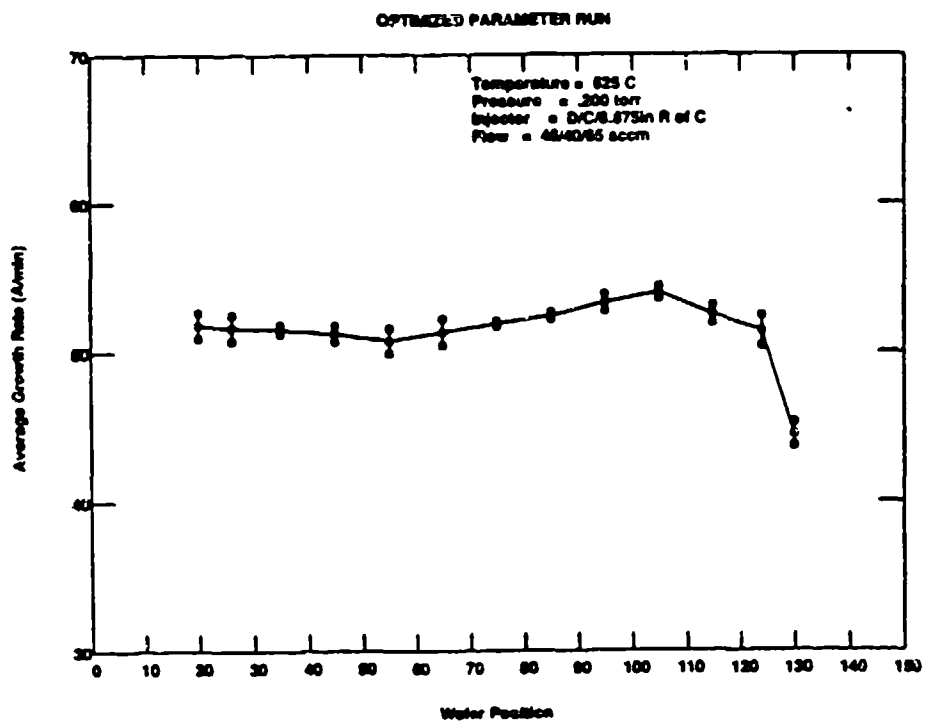


Figure 7: Deposition profile for optimum factor levels.

thus indicating that the equipment was optimized.

5 CALIBRATION OF CONSTANTS

5.1 Parameter Fit to L_0 Array

The kinetic constants in the reaction rate model, Equation 5, and the injector constant, Equation 7, were fit using a design optimization package called OPTDES developed at Brigham Young University [13]. This package used a non-linear reduced gradient method to optimize a least square objective function. The objective function included all nine experiments by summing the squared differences between the 13 measured data values and the predicted values at the corresponding wafer locations for each individual experiment in the L_0 and then summing the values for all nine experiments.

The constants obtained for the reaction model were:

$$k_1 = 1.202 \times 10^{10} \exp(-18,500/T) \text{ mol/m}^2/\text{s/atm}$$

$$K_s = 0.386 \times 10^5 \text{ atm}^{-1}$$

$$K_h = 1.904 \times 10^4 \text{ atm}^{-1/2}$$

The kinetic constants found from this regression are substantially different from those obtained by Roenigk and Jensen [6]:

$$k_1 = (1.6 \pm 0.4) \times 10^9 \exp(-18,500/T) \text{ mol/m}^2/\text{s/atm}$$

$$K_s = (0.7 \pm 0.1) \times 10^5 \text{ atm}^{-1}$$

$$K_h = (0.6 \pm 0.3) \times 10^2 \text{ atm}^{-1/2}$$

Our kinetic constants indicate that hydrogen has a much greater effect in inhibiting the reaction rate; K_h is three orders of a magnitude larger. The Arrhenius constant in k_1 is larger by an order of magnitude due to the large increase of K_h . We also found that K_s has relatively little effect on the predicted profiles in the range of $0.01 \times 10^5 \text{ atm}^{-1}$ to $1.0 \times 10^5 \text{ atm}^{-1}$ since the denominator is dominated by the effect of hydrogen. The activation energy in k_1 , Equation 6, was not fit to the data, but was determined by other investigators [14].

The ramp injector gave slightly better maximization of the objective function criteria. The injector fitting constant of Equation 7 was found to be 5.165. To give an idea of what this means, for an injector flow rate of 60 std cm³/min the modeled injected gas length is approximately 3cm.

5.2 Choice of Injector Model

The choice of the model to represent the gas spray dynamics from the two moveable injectors was done based on two criteria: a qualitative analysis of the injection dynamics; and maximization of the least square objective function.

After the the model parameters were fit to the L₉ data it was found that the injector constant for the flat injector function was approximately twice that of the ramp function: 9.626 (flat) v. 5.165 (ramp). This is consistant with the idea that the flat model represented an average approximation of the ramp model. The ramp function gave a slightly larger objective function result, indicating that the variance from the measured data was less for this function.

Since qualitatively the ramp function is more representative of the expected gas dynamics and the objective function criteria was maximized for the ramp function, the ramp function was chosen as injector model.

5.3 Calibration Check

To check the use of the L₉ experiment set for calibration of the kinetic paramters a "mini-experiment" matrix was designed to capture the main effects of the kinetic parameters in Equation 5. The design consisted of three experinents in which the pressure and temperature were varied. In these experiments, the total flow was introduced through the load injector only. This was done in order to maximize the effect of the silane depletion and hydrogen generation on the deposition profile. The

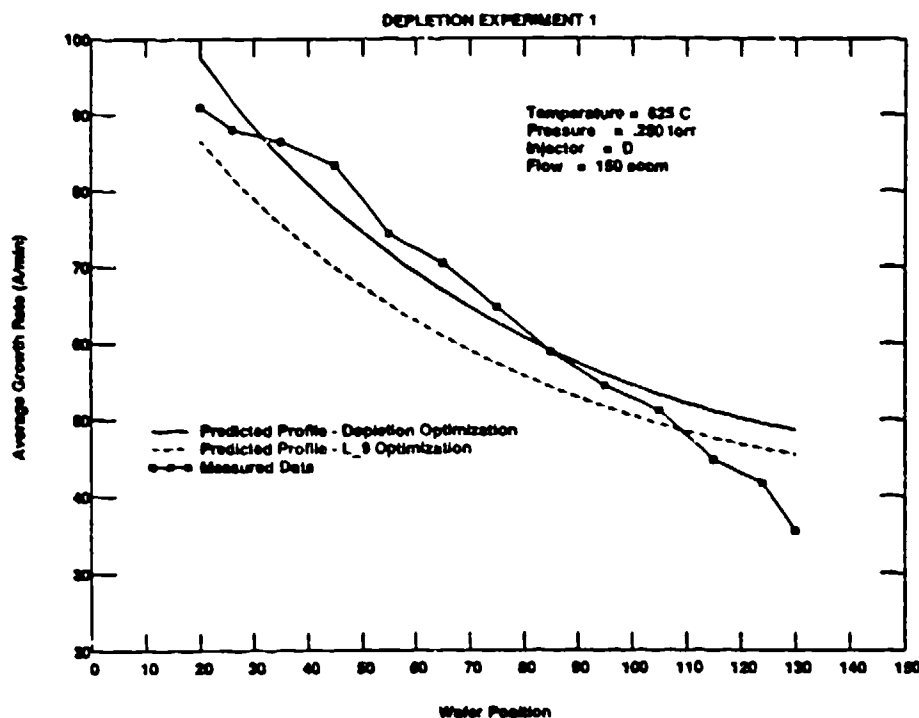


Figure 8: Second parameter fit of reaction model - Experiment 1.

constants found in the parameter fit to this data are:

$$k_1 = 1.412 \times 10^{10} \exp(-18,500/T) \text{ mol/m}^2/\text{s}/\text{atm}$$

$$K_s = 0.368 \times 10^6 \text{ atm}^{-1}$$

$$K_A = 1.814 \times 10^4 \text{ atm}^{-1/2}$$

These constants are very close to those obtained by the parameter fit to the L_9 design. The difference in the predicted profiles can be seen in Figures 8 - 10, where the predicted profiles with the respective constant fits are plotted versus the measured data for each of the experiments from three experiment design. This small variation in the two predicted profiles indicates that the L_9 design was adequate to fit the reaction kinetic parameters. From Figures 9 and 10, which were run at the same pressure, but different temperatures, it can be seen that the activation energy used is representative of the system.

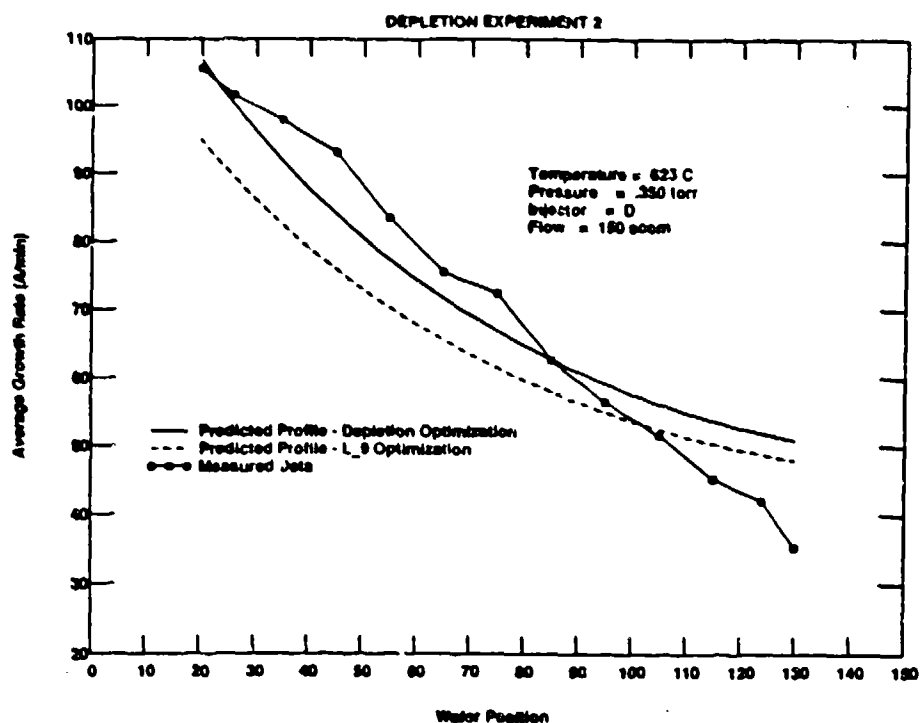


Figure 9: Second parameter fit of reaction model - Experiment 2.

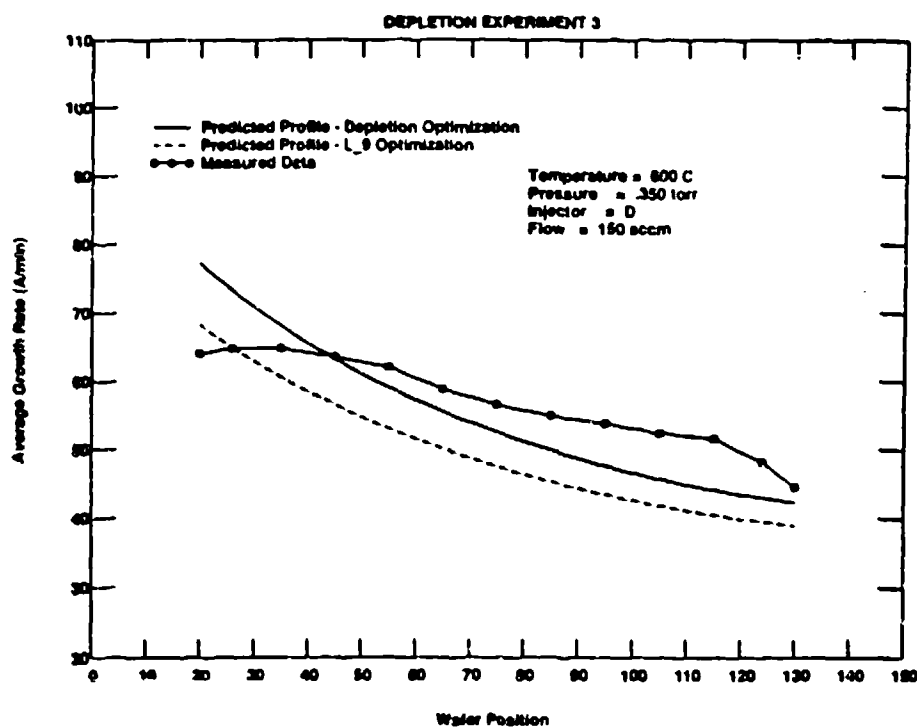


Figure 10: Second parameter fit of reaction model - Experiment 3.

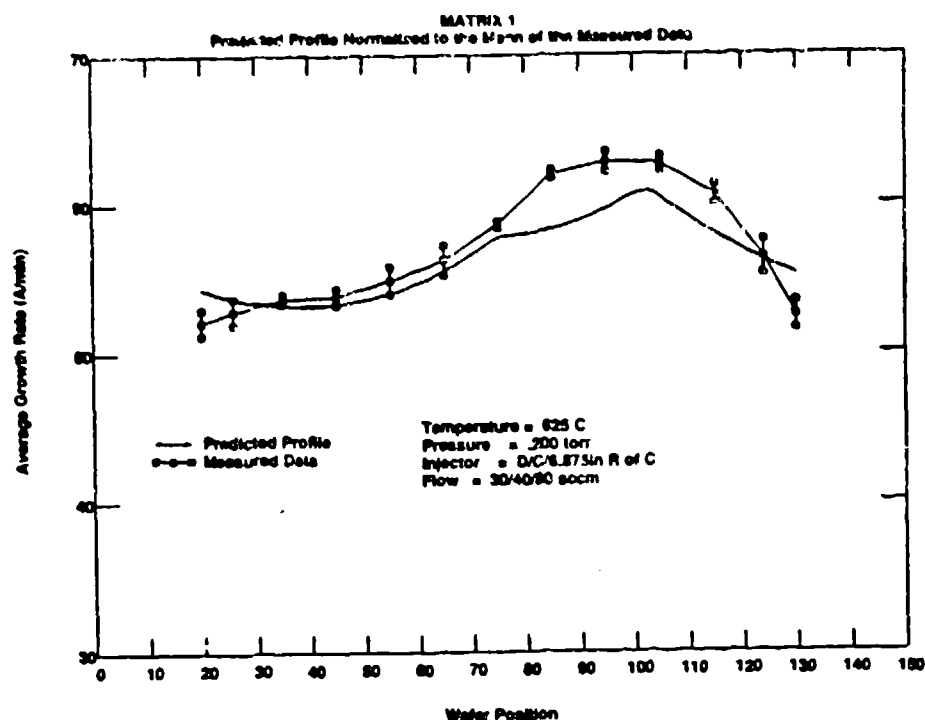


Figure 11: Experiment 1.

6 MODEL RESULTS

The predicted deposition profiles closely resemble the measured profiles. Figures 11 - 20 give the predicted deposition profiles for each of the L_0 experiments and the optimized parameter run according to the previous Taguchi analysis with the measured data. It is important to note that in each simulation, the model was run with the same parameter settings as the equipment.

The primary purpose of our model is to predict the thickness variation down the length of the tube. Given a combination of process parameters (flow rates, injector positions, pressure, temperature, etc.) a calibration experiment can be run to determine the amount of time needed to determine a target thickness. The predicted profiles are therefore normalized to the means of the respective runs. Normalization was achieved by shifting the entire predicted curve by a multiplicative constant equal to the ratio of the measured profile mean to the predicted profile

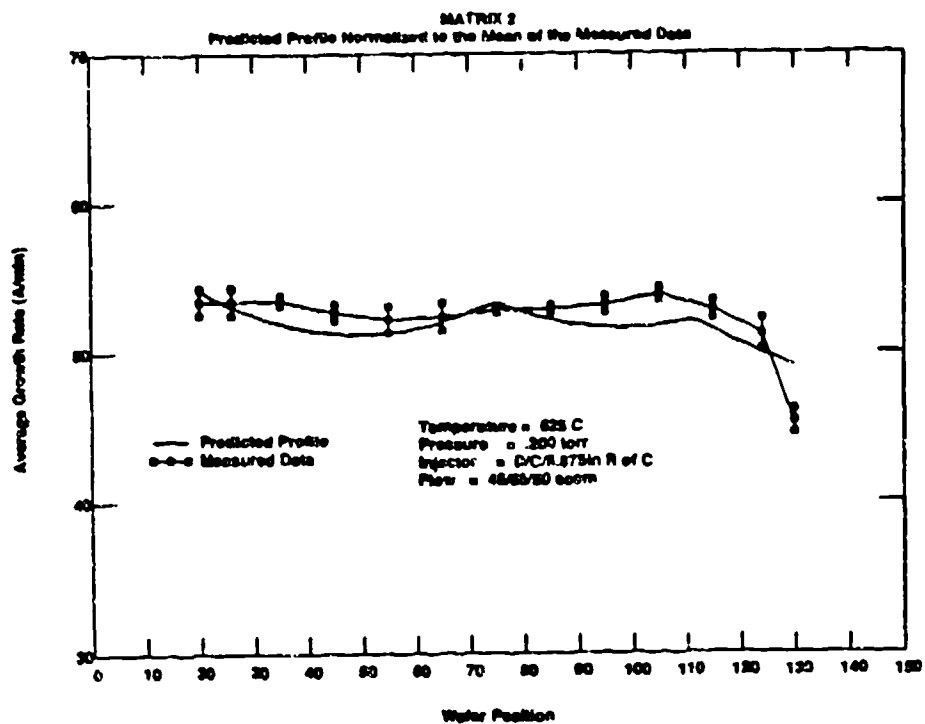


Figure 12: Experiment 2.

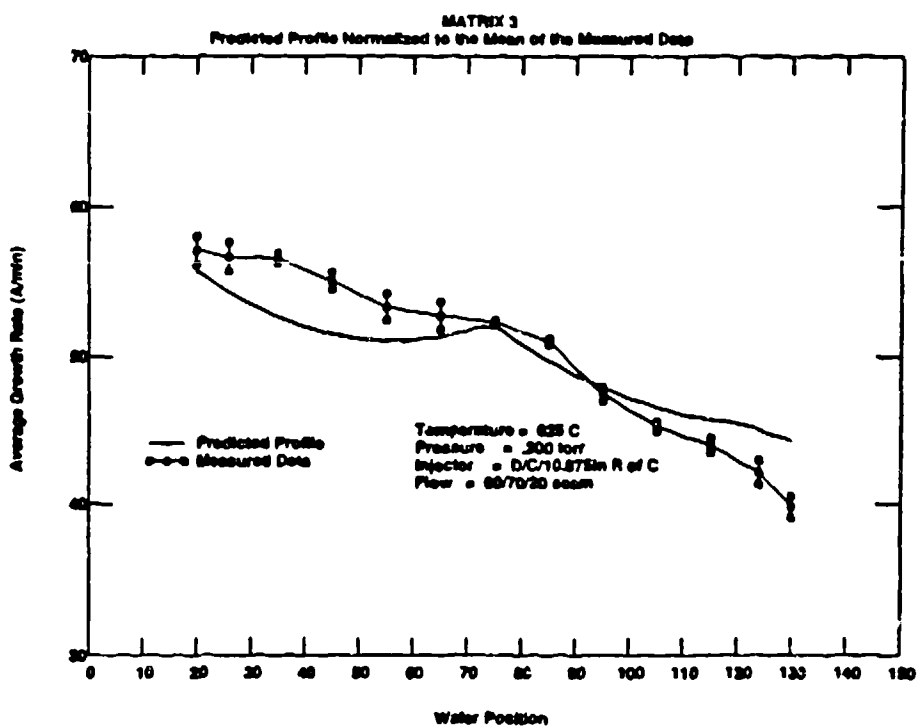


Figure 13: Experiment 3.

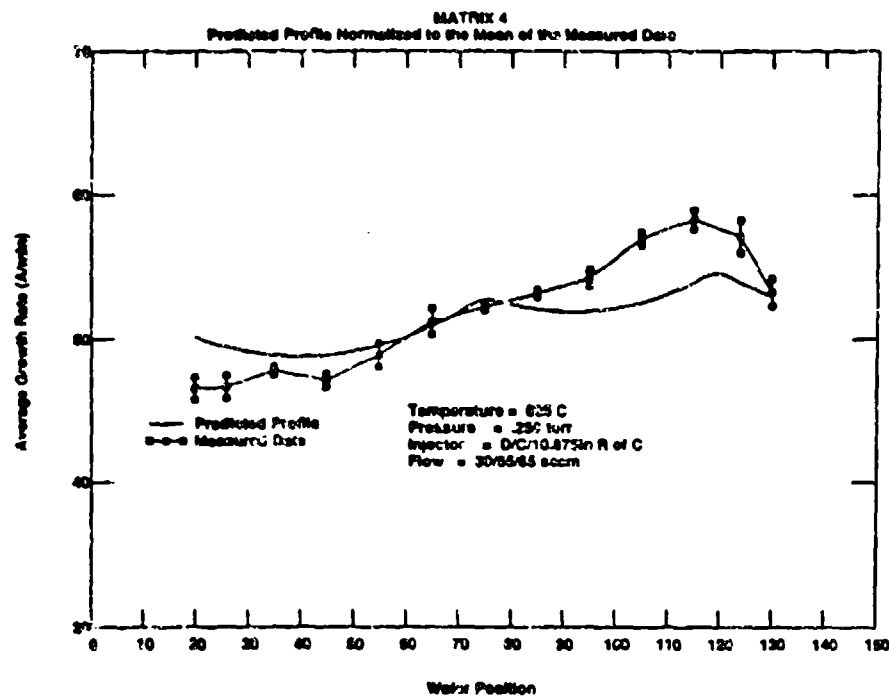


Figure 14: Experiment 4.

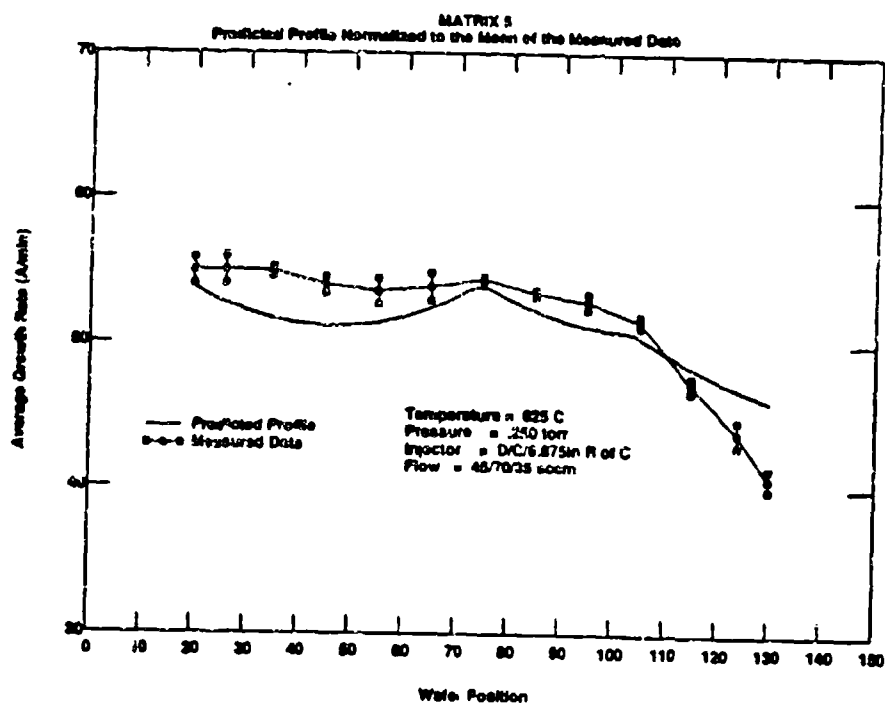


Figure 15: Experiment 5.

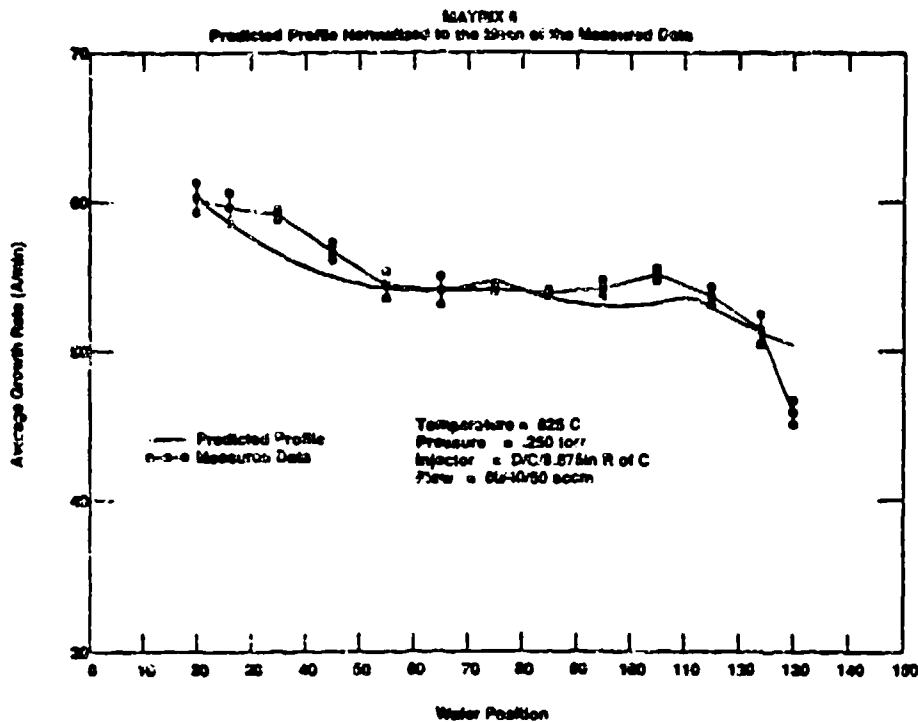


Figure 16: Experiment 6.

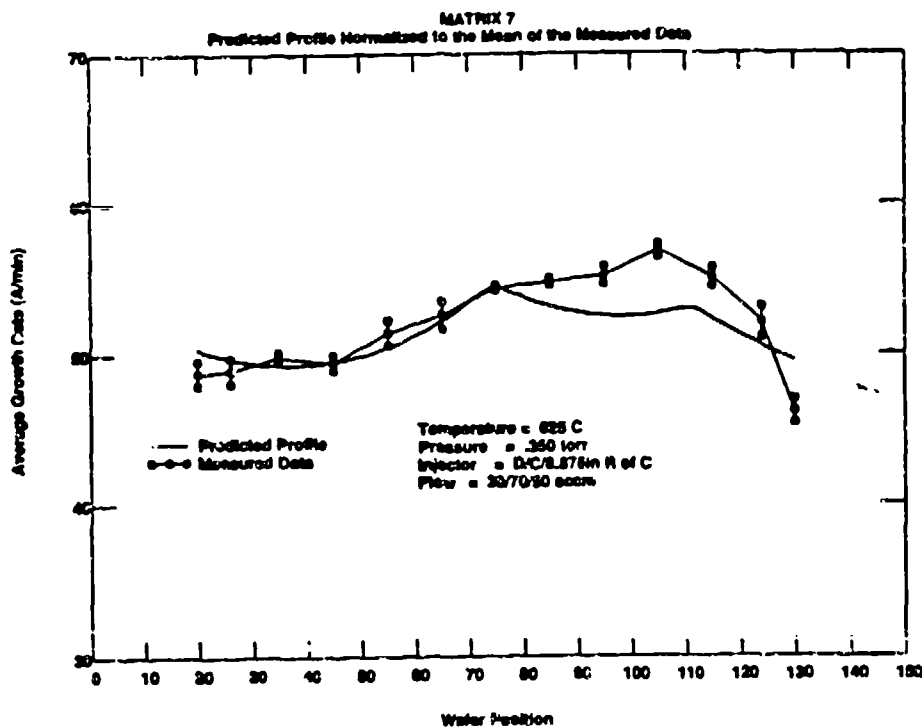


Figure 17: Experiment 7.

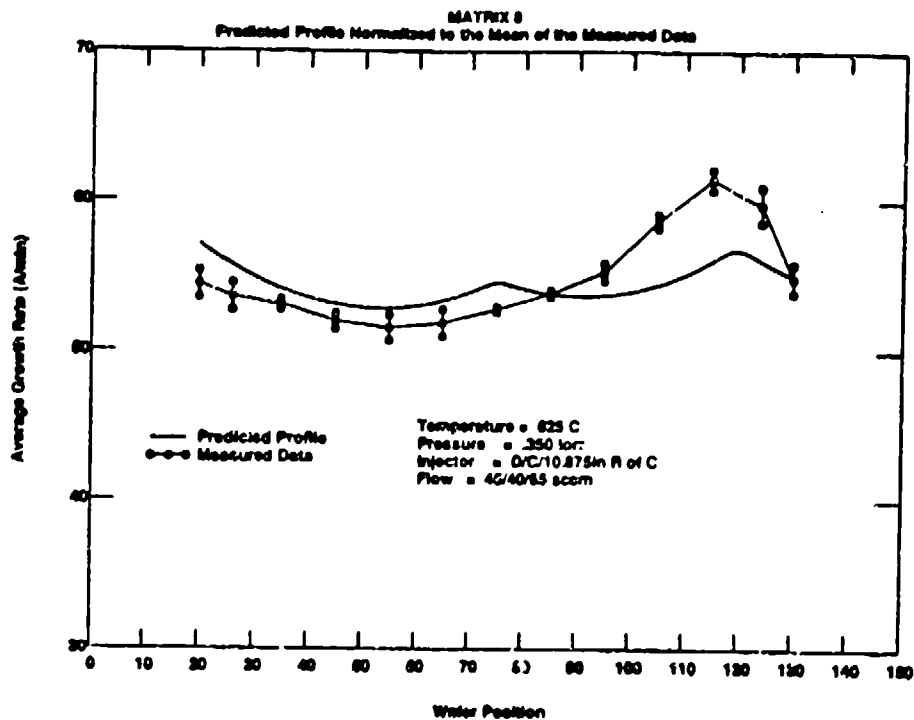


Figure 18: Experiment 8.

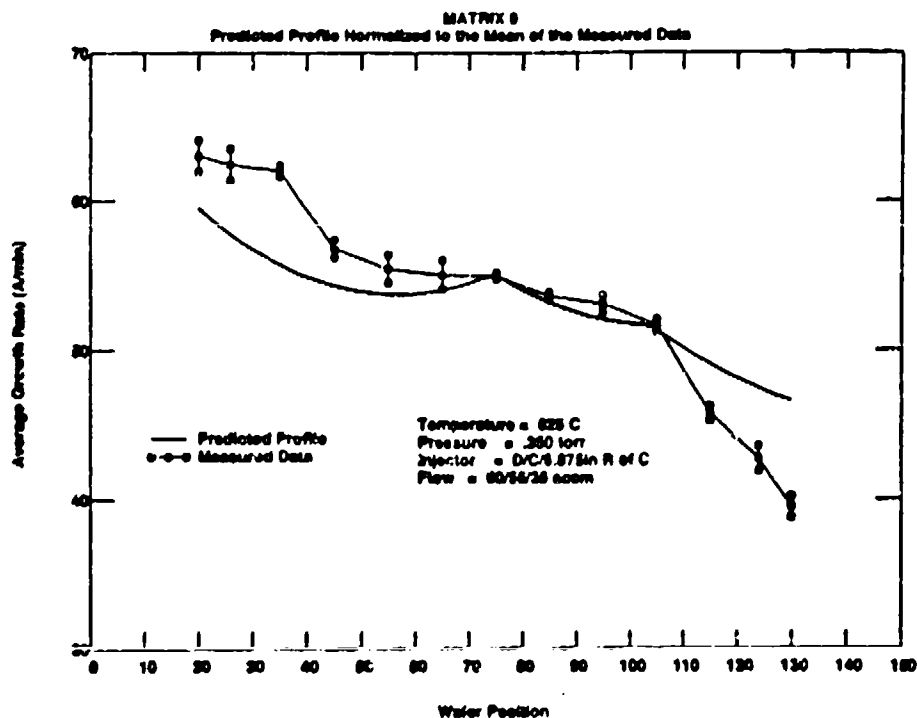


Figure 19: Experiment 9.

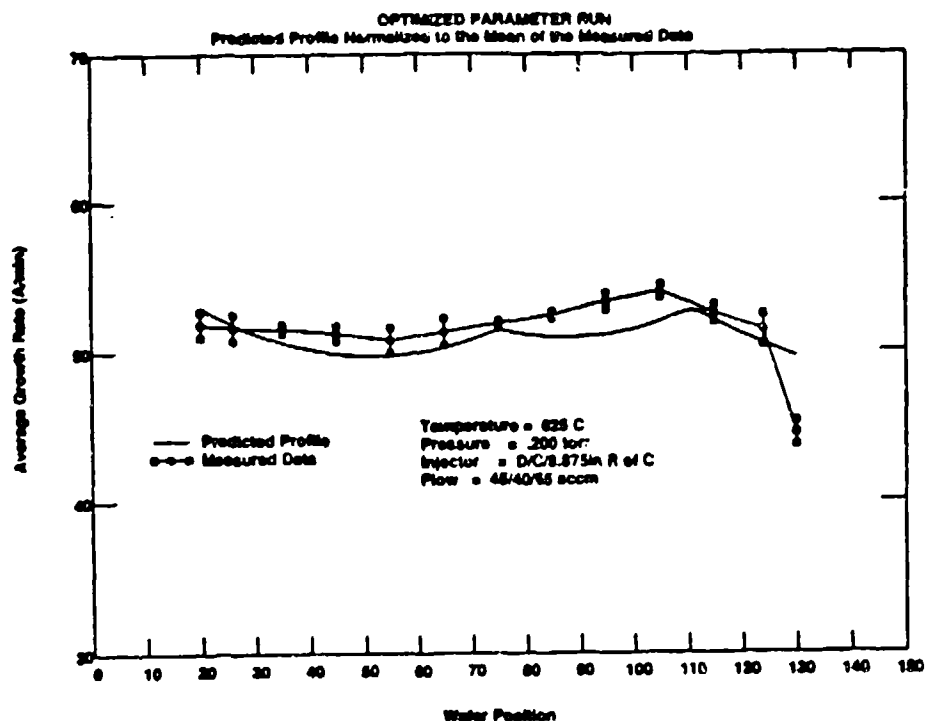


Figure 20: Optimized factor levels.

mean for each profile. In the worst case, Experiment 1, the mean of the predicted profile was 10% lower than the mean of the experimental data. The average error in the prediction of the mean deposition rate was 5%. No systematic explanation was found for the deviation in the mean growth rates.

The peaks in the predicted profiles are due to the approximation of the injector gas spray. It should be noted that the predicted profiles give an accurate representation of the measured profiles except at positions localized above the injector exit positions.

As a measure of the accuracy of the model to predict the variations in the equipment parameters, the signal-to-noise ratios, according to Equation 35, of the measured profiles were plotted against the SN of the predicted deposition profiles, Figure 21. A perfect correspondence would have resulted in a line with a slope of one and a y-intercept of zero. The results (slope = 1.05, y-intercept = -7.1) indicate that the model accurately predicts the effect of variations in the equipment

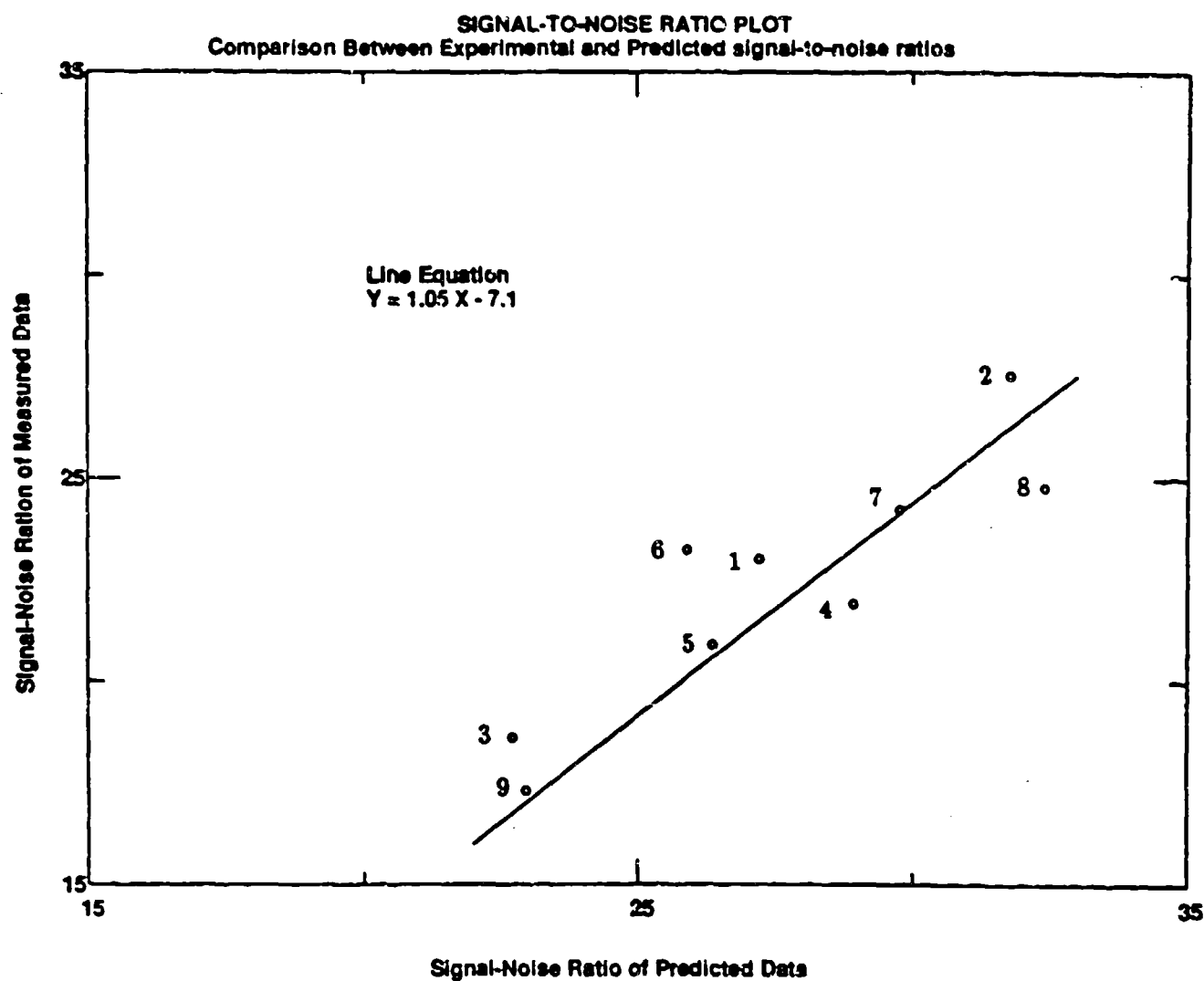


Figure 21: Signal-to-noise ratio measure. The numbers indicate the experiment number from the L_0 .

parameters on the axial deposition profile. The slope shows that the measured data and predicted data correlate very well since they follow the same trend. The y-intercept is a measure of a constant offset of the predicted data from the measured data. Meaning, in this case, that the model predicts higher SN for each of the experiments than those found from the measured data. This plot shows that the optimum operating point, if determined from the experimental data, would be Experiment 2, but if it was to be determined from the model predictions, Experiment 8 would be chosen. These values are very close, there is not much deviation from either of these choices with respect to the profile flatness. The experiment optimized by the Taguchi SN analysis is closest to Experiment 2.

7 CONCLUSIONS

A model has been developed which predicts the wafer-to-wafer deposition rate of polysilicon down the length of a horizontal tube furnace. Inputs to the model include: silane flow rates from three injectors, two injector locations, locations of and temperatures at three thermocouples, operating pressure, the number of wafers, wafer diameter, the location of the wafer load, and other physical dimensions of the furnace such as tube length and liner diameter. The model construction consists of a one dimensional finite difference numerical representation of the convective and diffusive fluxes of silane and hydrogen. Silane is injected and hydrogen is generated by reaction at the wafers and tube wall. The silane injection and mixing was modeled with a ramp function which is an approximate model incorporating a qualitative understanding of the injection phenomena.

Parameters in the reaction kinetics model and injector function were fit to a set of nine statistically designed experiments which varied four parameters, two injector flow rates, one injector position, and pressure, over three levels. These parameters were fit to all nine experiments in the design to give the best possible fit to all the data. The reaction kinetic parameters were independently fit to a second set of three experiments in which flow was admitted through only the load injector. The experiments were designed to explore a different region of operation, spanning a wide range of silane and hydrogen concentrations. The close correspondence between the reaction parameters fit to each set of experiments demonstrated that the model extrapolates well to regions beyond the scope of the initial experimental space.

An extension of the model, which included equipment disturbances or noises, was used to optimize the process by finding the settings at which the process was most robust; that is, the settings which gave the flattest profile over the full range of the noises. The settings predicted by the model correspond very closely to an experimentally determined robust operating point as determined by a Taguchi signal-

to-noise ratio analysis of the statistically designed experimental space.

For the process engineer, the model is well suited for process optimization. Equipment and process disturbances can also be included to determine the most robust operating point, as has been demonstrated. Furthermore, the model can be extended for use in on-line quality control. Having optimized the operating point by process optimization, the operator can use the model to correct the equipment settings, based on product measurements, to maintain the required deposition specifications.

The equipment designer will find the model useful for testing new ideas in equipment design. Minor extensions can be made to the model, such as adding injectors, to determine the benefits of these changes without the cost of materials and experimentation.

References

- [1] M. L. Hitchman, J. Kane, and A. E. Widmer, "Polysilicon Growth Kinetics in a Low Pressure Chemical Vapour Deposition Reactor," *Thin Solid Films*, vol. 59, pp. 231-247, 1979.
- [2] C. H. J. V. D. Brekel and L. J. M. Bollen, "Low Pressure Deposition of Polysilicon Silicon From Silane," *Journal of Crystal Growth*, vol. 54, pp. 310-322, 1981.
- [3] W. A. P. Claassen *et al.*, "The Deposition of Silicon from Silane in a Low-Pressure Hot-Wall System," *Journal of Crystal Growth*, vol. 57, pp. 259-266, 1982.
- [4] S. Middleman and A. Yeckel, "A Model of the Effects of Diffusion and Convection on the Rate and Uniformity of Deposition in a CVD Reactor," *Journal of the Electrochemical Society*, vol. 133, no. 9, pp. 1051-1056, 1986.
- [5] K. F. Jensen and D. B. Graves, "Model and Analysis of Low Pressure CVD Reactors," *Journal of Electrochemical Society*, vol. 130, no. 9, pp. 1950-1957, 1983.
- [6] K. F. Roenigk and K. F. Jensen, "Analysis of Multicomponent LPCVD Processes," *Journal of the Electrochemical Society*, vol. 132, no. 2, pp. 448-454, 1985.
- [7] K. Lin, *An Expert System For Polysilicon Recipe Generation*. Master's thesis, University of California, Berkeley, July 1987.
- [8] G. Taguchi, *Introduction to Quality Engineering*. Tokyo: Asian Productivity Organization, 1986.
- [9] G. E. P. Box, W. G. Hunter, and J. S. Hunter, *Statistics for Experimentors*. John Wiley & Sons, 1978.

- [10] G. Birkhoff and E. H. Zarantonello, *Jets, Wakes and Cavities*. New York: Academic Press, Inc., 1957.
- [11] G. P. Brown, A. DiNardo, *et al.*, "The Flow of Gases in Pipes at Low Pressures," *Journal of Applied Physics*, vol. 17, pp. 802-813, October 1946.
- [12] M. S. Phadke, R. N. Kacker, D. V. Speeney, and M. J. Grieco, "Off-Line Quality Control in Integrated Circuit Fabrication Using Experimental Design," *The Bell System Technical Journal*, vol. 62, no. 5, pp. 1273-1309, 1983.
- [13] A. Parkinson, R. Balling, and J. Free, "OPTDES: A Software System for Optimal Engineering Design," in *Proceedings of the American Society of Mechanical Engineers Conference on Computers and Engineering*, (Las Vegas, Nevada), August 1986.
- [14] W. A. Bryant, "The Kinetics of the Deposition of Silicon by Silane Pyrolysis at Low Temperatures and Atmospheric Pressure," *Thin Solid Films*, vol. 60, pp. 19-25, 1979.

A APPENDIX A - Numerical Technique

In this Appendix, the numerical techniques for the solution of Equations 29 and 28 are presented and discussed. It is assumed that the equations are discretized over a one-dimensional grid, which consists of $N + 1$ grid points having coordinates $z_i, i = 1, N + 1$. The intervals defined by two grid nodes i and $i + 1$ are referred to by using the standard notation.

A.1 The total velocity equation

Equation 29 is a non-linear Poisson's equation which can be solved using first-order finite-difference techniques. In order to explicit the non-linearity of this equation we can reformulate Equation 28 and 29 as follows:

- as a first step it is possible to observe that if $N_{SiH_4} \ll C_{tot} D \frac{dX_{SiH_4}}{dz}, \frac{d\psi}{dz} C_{tot} X_{SiH_4}$ then we have that $X_{SiH_4} \simeq \exp\left(\frac{\psi}{D}\right)$
- Generalizing the previous observation, in general we can change the variable X_{SiH_4} into a new variable ϕ defined as:

$$\phi \doteq \psi + D \log X_{SiH_4} \quad (36)$$

which gives

$$X_{SiH_4} = e^{\frac{\phi - \psi}{D}} \quad (37)$$

- Using this formulation, the recombination term can be expressed as a function of the potential ψ and the non-linear problem solved by using the Newton-Raphson method.

To do this we have to compute the derivative of the recombination term with respect to the potential ψ :

$$\frac{dR}{d\psi} = -\frac{R}{D} \left(1 - \frac{C_{tot} X_{SiH_4} (k_2 - k_3 (C_{tot} (1 - X_{SiH_4}))^{-1/2} / 2)}{1 + k_2 C_{tot} X_{SiH_4} + k_3 \sqrt{C_{tot} (1 - X_{SiH_4})}} \right) \quad (38)$$

The Newton-Raphson method can be expressed as a succession of linear problems. Assuming that $\psi = \psi^0 + \Delta\psi$ we get:

$$-\frac{d}{dz} \left(A_{tw} C_{tot} \frac{d\Delta\psi}{dz} \right) - \frac{dR}{d\psi} |_{\psi^0} \Delta\psi = \frac{d}{dz} \left(A_{tw} C_{tot} \frac{d\psi^0}{dz} \right) + (R |_{\psi^0} + F(z)) \quad (39)$$

- The discretization of the previous equation gives for $i = 2, N - 1$:

$$A_i C_i \left(\frac{2}{s_{i+1} - s_{i-1}} \right) \left(\frac{\Delta \psi_i - \Delta \psi_{i-1}}{s_i - s_{i-1}} \right) - A_{i+1} C_{i+1} \left(\frac{2}{s_{i+1} - s_{i-1}} \right) \left(\frac{\Delta \psi_{i+1} - \Delta \psi_i}{s_{i+1} - s_i} \right) - \left(\frac{dR_i}{d\psi} \right) |_{\psi^0} \Delta \psi_i = -A_i C_i \left(\frac{2}{s_{i+1} - s_{i-1}} \right) \left(\frac{\psi_i^0 - \psi_{i-1}^0}{s_i - s_{i-1}} \right) + A_{i+1} C_{i+1} \left(\frac{2}{s_{i+1} - s_{i-1}} \right) \left(\frac{\psi_{i+1}^0 - \psi_i^0}{s_{i+1} - s_i} \right) + (R_i |_{\psi^0} + F) \quad (40)$$

- The boundary conditions can be taken into account as:

- Boundary condition at the load zone for $i = 1$:

$$-A_{i+1} C_{i+1} \left(\frac{2}{s_{i+1} - s_i} \right) \left(\frac{\Delta \psi_{i+1} - \Delta \psi_i}{s_{i+1} - s_i} \right) - \left(\frac{dR_i}{d\psi} \right) |_{\psi^0} \Delta \psi_i = A_{i+1} C_{i+1} \left(\frac{2}{s_{i+1} - s_i} \right) \left(\frac{\psi_{i+1}^0 - \psi_i^0}{s_{i+1} - s_i} \right) + (R_i |_{\psi^0} + F) + Q_0 \left(\frac{2}{s_{i+1} - s_i} \right) \quad (41)$$

- Boundary condition at the end of the tube. The determination of the potential at node $i = N$ has to be done using the following equation:

$$A_i C_i \left(\frac{2}{s_{i+1} - s_{i-1}} \right) \left(\frac{\Delta \psi_i - \Delta \psi_{i-1}}{s_i - s_{i-1}} \right) - A_{i+1} C_{i+1} \left(\frac{2}{s_{i+1} - s_{i-1}} \right) \left(\frac{-\Delta \psi_i}{s_{i+1} - s_i} \right) - \left(\frac{dR_i}{d\psi} \right) |_{\psi^0} \Delta \psi_i = -A_i C_i \left(\frac{2}{s_{i+1} - s_{i-1}} \right) \left(\frac{\psi_i^0 - \psi_{i-1}^0}{s_i - s_{i-1}} \right) + A_{i+1} C_{i+1} \left(\frac{2}{s_{i+1} - s_{i-1}} \right) \left(\frac{-\psi_i^0}{s_{i+1} - s_i} \right) + (R_i |_{\psi^0} + F) \quad (42)$$

A.2 The convection-diffusion equation

The discretization of the total velocity equation can be derived in two different ways: either by assuming that the potential is linear over each element or assuming that the velocity is constant over each element. While the two approaches are equivalent in the total velocity equation, the discretization of the convection-diffusion equation is different in these two cases. More technically, the assumption about the field can be stated in a framework related to the mixed finite-element techniques. This approach is more useful for the convection-diffusion equation since it allows the imposition of the conservation of the masses in a very straightforward way.

Assuming that N_{SiH_4} is constant over each element, we have:

$$C_{tot} D \frac{dX_{SiH_4}}{dz} + C_{tot} X_{SiH_4} \frac{d\psi}{dz} = N \quad (43)$$

Multiplying each side of the previous equation by $\exp(\psi/D)$ we obtain:

$$N e^{\frac{\psi}{D}} = \left(C_{tot} D \frac{dX_{SiH_4}}{dz} + C_{tot} X_{SiH_4} \frac{d\psi}{dz} \right) e^{\frac{\psi}{D}} = C_{tot} D \frac{d(X_{SiH_4} e^{\frac{\psi}{D}})}{dz} \quad (44)$$

Integrating both sides of the previous equation and taking into account that ψ is linear over each interval, we have:

$$N \int_{z_i}^{z_{i+1}} e^{\frac{\psi}{D}} \left(\psi_i + \frac{\psi_{i+1} - \psi_i}{z_{i+1} - z_i} (z - z_i) \right) dz = \int_{z_i}^{z_{i+1}} C_{tot} D \frac{d(X_{SiH_4} e^{\frac{\psi}{D}})}{dz} dz \quad (45)$$

Evaluating the previous integrals, we have:

$$N D \frac{z_{i+1} - z_i}{\psi_{i+1} - \psi_i} e^{\frac{\psi}{D}} \Big|_{z_i}^{z_{i+1}} = C_{tot} D (X_{SiH_4} e^{\frac{\psi}{D}}) \Big|_{z_i}^{z_{i+1}} \quad (46)$$

Rewriting the previous expression and using for clarity X for X_{SiH_4} , we have that the flux over each element can be expressed as:

$$N = \frac{C_{tot}}{z_{i+1} - z_i} \left(X_{i+1} \frac{\psi_i - \psi_{i+1}}{e^{\frac{\psi_i - \psi_{i+1}}{D}} - 1} - X_i \frac{\psi_{i+1} - \psi_i}{e^{\frac{\psi_{i+1} - \psi_i}{D}} - 1} \right) \quad (47)$$

Defining

$$B_{i,i+1} = \frac{\psi_i - \psi_{i+1}}{e^{\frac{\psi_i - \psi_{i+1}}{D}} - 1} \quad (48)$$

we obtain that the flux going from node $i + 1$ to node i can be expressed as:

$$N = \frac{C_{tot}}{z_{i+1} - z_i} (B_{i,i+1} X_{i+1} - B_{i+1,i} X_i) \quad (49)$$

The determination of the concentration C_i can be obtained by imposing that the flux entering into the "cell" surrounding node i is equal to the recombination-generation term. In this way we have:

$$- \left[\left(\frac{2A_{i+1}C_{i+1}}{z_{i+1} - z_{i-1}} \right) \frac{B_{i,i+1}X_{i+1} - B_{i+1,i}X_i}{z_{i+1} - z_i} - \left(\frac{2A_iC_i}{z_{i+1} - z_{i-1}} \right) \frac{B_{i-1,i}X_i - B_{i,i-1}X_{i-1}}{z_i - z_{i-1}} \right] = - (R_i - F(x_i)) \quad (50)$$

Since the previous equation is non-linear we can solve it iteratively by using a the

Newton method. Defining $X = X^0 + \Delta X$, we have:

$$\begin{aligned}
 & - \left[\left(\frac{2A_{i+1}C_{i+1}}{s_{i+1}-s_{i-1}} \right) \frac{B_{i,i+1}\Delta X_{i+1} - B_{i+1,i}\Delta X_i}{s_{i+1}-s_i} - \left(\frac{2A_iC_i}{s_{i+1}-s_{i-1}} \right) \frac{B_{i-1,i}\Delta X_i - B_{i,i-1}\Delta X_{i-1}}{s_i-s_{i-1}} \right] + \\
 & \left(\frac{dR_i}{dX} \right) |_{X_0} \Delta X_i = \\
 & + \left[\left(\frac{2A_{i+1}C_{i+1}}{s_{i+1}-s_{i-1}} \right) \frac{B_{i,i+1}X_{i+1}^0 - B_{i+1,i}X_i^0}{s_{i+1}-s_i} - \left(\frac{2A_iC_i}{s_{i+1}-s_{i-1}} \right) \frac{B_{i-1,i}X_i^0 - B_{i,i-1}X_{i-1}^0}{s_i-s_{i-1}} \right] \\
 & - (R_i - F(z_i))
 \end{aligned} \tag{51}$$

The boundary condition can be taken into account as follows:

- at the load zone for $i = 1$:

$$\begin{aligned}
 & - \left(\frac{2A_{i+1}C_{i+1}}{s_{i+1}-s_i} \right) \frac{B_{i,i+1}\Delta X_{i+1} - B_{i+1,i}\Delta X_i}{s_{i+1}-s_i} + \left(\frac{dR_i}{dX} \right) |_{X_0} \Delta X_i = \\
 & \left(\frac{2A_{i+1}C_{i+1}}{s_{i+1}-s_i} \right) \frac{B_{i,i+1}X_{i+1}^0 - B_{i+1,i}X_i^0}{s_{i+1}-s_i} - (R_i - F(z_i)) + \left(\frac{2Q_0}{s_{i+1}-s_i} \right)
 \end{aligned} \tag{52}$$

- at end of the tube for $i = N$ we have that the transport is due to convective effects only. This implies that the concentration at nodes N and $N + 1$ is the same and more precisely that the equation which determines the concentration at node N is:

$$\begin{aligned}
 & - \left[\left(\frac{2A_{i+1}C_{i+1}}{s_{i+1}-s_{i-1}} \right) \frac{\Delta X_i(-\psi_i)}{s_{i+1}-s_i} - \left(\frac{2A_iC_i}{s_{i+1}-s_{i-1}} \right) \frac{B_{i-1,i}\Delta X_i - B_{i,i-1}\Delta X_{i-1}}{s_i-s_{i-1}} \right] + \\
 & \left(\frac{dR_i}{dX} \right) |_{X_0} \Delta X_i = \\
 & \left[- \left(\frac{2A_{i+1}C_{i+1}}{s_{i+1}-s_{i-1}} \right) \frac{X_i^0\psi_i}{s_{i+1}-s_i} - \left(\frac{2A_iC_i}{s_{i+1}-s_{i-1}} \right) \frac{B_{i-1,i}X_i^0 - B_{i,i-1}X_{i-1}^0}{s_i-s_{i-1}} \right] - (R_i - F(z_i))
 \end{aligned} \tag{53}$$

B APPENDIX B - Polysilicon Model Code

In this Appendix the code for the polysilicon model is presented. The structure of the code is depicted in Figure B.1. The main part of the code, including the input routines, initial guess, empirical setup of the injectors, and the setup of the tri-diagonal matrix is written in C. Two subroutines used to solve a parameter in the convection-diffusion equations, `bernoulli.f` - written by Roberto Guerrieri, and a tri-diagonal matrix solver, `dgtls.f` - obtained from LINPACK, are written in Fortran.

The code presented also has a L_0 inner array adapted to the main code. This allowed for the optimization of the model parameters to all nine of the experiment runs.

B.1 Main Code

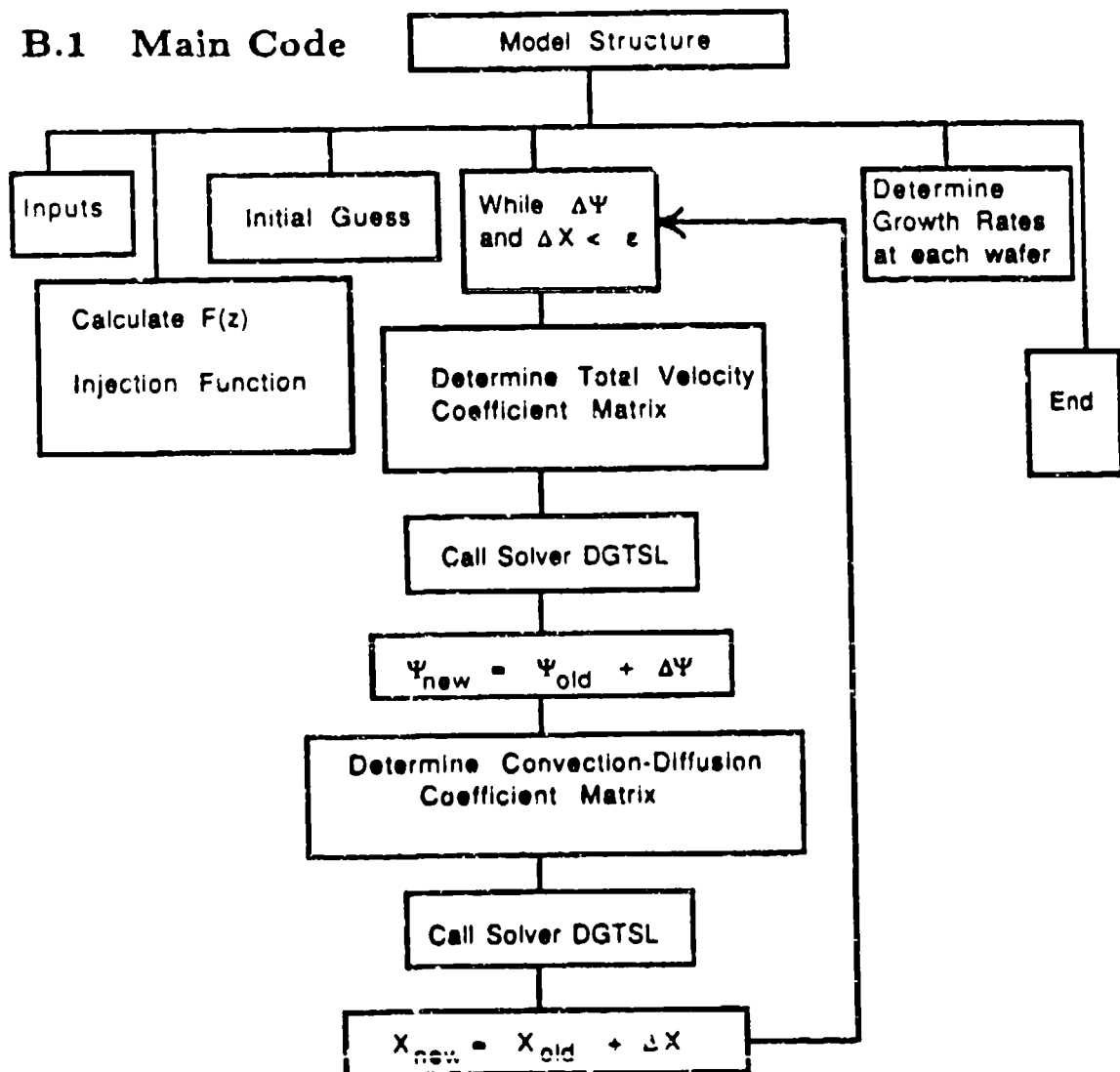


Figure B.1: Code Structure

```

/* subroutine for OPTDES */

#include <stdio.h>
#include <math.h>

/* define the global constants */

#define PI      4*atan(1.0)
#define sq(x)   x*x
#define cu(x)   x*x*x
#define trac(x) x - floor(x)
#define GC      8.206e-5 /* atm*m^3/gm mole*K */

/* This program is the core of a simulator for the deposition of */
/* polysilicon in an annular diffusion furnace */

main(argc, argv)
    int argc;
    char *argv[];
{
    /* define variables to be used */

    FILE *sp, *fclose();
    double ro[301], ql, qc, qs, t[301], p, pci, psi, pfirstw, li, lid;
    double fl, wdia, idia, D[301], kl[301], Ks[301], Kh[301], C_tot[301];
    double F[301], delw, delz, waferp, aw, a', cat, sat, Lr[301], Atw[301];
    double X[301], si[301], del_si[301], del_X[301], max_del_si, max_del_X;
    double lpt, tl, cpt, tc, spt, ts;
    double kinetics(), injection(), init_guess(), velocity(), conv_diff();
    double file_out(), dgtsl_(), dberl_();
    double kk0, ks0, kh0, Ea, con;
    double tp[4], tq1[4], tqc[4], tpsi[4], sncount;
    char input[100];
    int N, nw, n, wafer[150], center_inject_elem, source_inject_elem, i, outer;
    int count, l19;
    static int l9[10][5] = {
        {0, 0, 0, 0, 0},
        {0, 1, 1, 1, 1},
        {0, 1, 2, 2, 2},
        {0, 1, 3, 3, 3},
        {0, 2, 1, 2, 3},
        {0, 2, 2, 3, 1},
        {0, 2, 3, 1, 2},
        {0, 3, 1, 3, 2},
    }
}

```

```

    {0, 3, 2, 1, 3},
    {0, 3, 3, 2, 1},
};

sp = fopen("sn_out.dat", "w");
fclose(sp);

/* number of elements in tube N */
N = 299;

/* input the necessary parameters */

/* printf("Input the wafer diameter (mm): "); */
gets(input);
wdia = atof(input) / 1000.0; /* meters */

/* printf("Input the liner inner diameter (mm): "); */
gets(input);
lid = atof(input) / 1000.0; /* meters */

/* printf("Input the total length of the furnace (cm): "); */
gets(input);
fl = atof(input) / 100.0; /* meters */

delz = fl / (N + 1); /* distance between nodes */

/* printf("Input the position of the load thermocouple from the load end (cm): "); */
gets(input);
lpt = atof(input) / 100.0; /* meters */

/* printf("Input the load control temperature (C): "); */
gets(input);
tl = atof(input) + 273.15; /* degrees Kelvin */

/* printf("Input the position of the center thermocouple from the load end (cm): "); */
gets(input);
cpt = atof(input) / 100.0; /* meters */

/* printf("Input the center control temperature (C): "); */
gets(input);
tc = atof(input) + 273.15; /* degrees Kelvin */

/* printf("Input the position of the source thermocouple from the load end (cm): "); */
gets(input);
spt = atof(input) / 100.0; /* meters */

/* printf("Input the source control temperature (C): "); */
gets(input);
ts = atof(input) + 273.15; /* degrees Kelvin */

/* determine the t[i] from a -300 C/m temperature loss slope */
if (rint(lpt/delz) == 0) {
    lpt = lpt + delz;
}
for (i = 1; i <= rint(lpt/delz); i++) {
    if (rint(lpt/delz) == 1) {
        t[i] = tl;
    }
    else {
        t[i] = tl + ((-300.*lpt)/(rint(lpt/delz) - 1))*(rint(lpt/delz) - i);
    }
}

/* temperature between tl and tc */

```

```

for (i=rint(lpt/delz); i<=rint(cpt/delz); i++) {
    t[i] = t1 + ((tc-t1)/(rint(cpt/delz)-rint(lpt/delz)))*(i - rint(lpt/delz));
}

/* temperature between tc and ts */
for (i=rint(cpt/delz); i<=rint(spt/delz); i++) {
    t[i] = tc + ((ts-tc)/(rint(spt/delz)-rint(cpt/delz)))*(i - rint(cpt/delz));
}

/* temperature between ts and the source end at -80 C/m */
for (i=rint(spt/delz); i<=(N+1); i++) {
    if (rint(spt/delz) == (N+1)) {
        t[i] = ts;
    }
    else {
        t[i] = ts + (-80*(i-spt)/(N+1-rint(spt/delz)))*(i - rint(spt/delz));
    }
}

/* ----- end temperature profiling ----- */

/* printf("Input the total process pressure (torr): "); */
gets(input);
p = atof(input)/760.0; /* atm */

/* calculate the silane density at pressure and temperature */
for (i=1; i<=(N+1); i++) {
    ro[i] = p / (GC * t[i]); /* gm mole / m^3 */
}

/* printf("Input the inner diameter of the injector (mm): "); */
gets(input);
idia = atof(input) / 1000.0; /* meters */

/* printf("Input the center injector position from load end (cm): "); */
gets(input);
pci = atof(input) / 100.0; /* meters */

/* printf("Input the source injector position from load end (cm): "); */
gets(input);
psi = atof(input) / 100.0; /* meters */

/* printf("Input the number of wafers in the load: "); */
gets(input);
nw = atoi(input); /* no units */

/* printf("Input the position of the first wafer from the load end (cm): "); */
gets(input);
pfirstw = atof(input) / 100.0; /* meters */

/* printf("Input the length of the wafer load including boats (cm): "); */
gets(input);
l1 = atof(input) / 100.0; /* meters */

/* printf("Input the load injector flow rate (sccm): "); */
gets(input);
ql = atof(input)*(1.0/p)*(t[1]/(25+273.15))*(1.0/(cu(100.0)))
    *(1.0/60.0)*ro[1];
/* gm mole / sec */

/* printf("Input the center injector flow rate (sccm): "); */

```

```

gets(input);
i = rint(pci/delz);
qc = atof(input)*(1.0/p)*(t[i]/(25+273.15))*(1.0/(cu(100.0)))
                                     *(1.0/60.0)*ro[i];
                                     /* gm mole / sec */

/* printf("Input the source injector flow rate (sccm): "); */
gets(input);
i = rint(psi/delz);
qs = atof(input)*(1.0/p)*(t[i]/(25+273.15))*(1.0/(cu(100.0)))
                                     *(1.0/60.0)*ro[i];
                                     /* gm mole / sec */

/* printf("input k0: "); */
gets(input);
kk0 = atof(input) * 1e9;

/* printf("input ks: "); */
gets(input);
ks0 = atof(input) * 1e5;

/* printf("input kh: "); */
gets(input);
kh0 = atof(input) * 1e2;

/* printf("input Ea: "); */
gets(input);
Ea = atof(input);

/* printf("input data file flag: "); */
gets(input);
outer = atoi(input);

/* printf("input con: "); */
gets(input);
con = atof(input);

/* that is the end of the input */

/* set taguchi parameters */

tp[1] = .200;
tp[2] = .250;
tp[3] = .350;

tql[1] = 30.0;
tql[2] = 45.0;
tql[3] = 60.0;

tqc[1] = 40.0;
tqc[2] = 55.0;
tqc[3] = 70.0;

tpsi[1] = 113.3475;
tpsi[2] = 118.4275;
tpsi[3] = 123.5075;

/* begin taguchi 19 array */
for (l19 = 1; l19 <= 9; l19++) {
    /* set the correct run parameters */
    outer = l19;
    p = tp[l19][l19][1]/760.0;

```

```

for (i=1; i<=(N+1); i++) {
    ro[i] = p / (GC * t[i]);      /* gm mole / m^3 */
}

ql = tql[19[119][2]]*(1.0/p)*(t[1]/(25+273.15))*(1.0/(cu(100.0)))
    *(1.0/60.0)*ro[1];
    /* gm mole / sec */

i = rint(pci/delz);
qc = tqc[19[119][3]]*(1.0/p)*(t[i]/(25+273.15))*(1.0/(cu(100.0)))
    *(1.0/60.0)*ro[i];
    /* gm mole / sec */

psi = tpsi[19[119][4]]/100.0;
i = rint(psi/delz);
qs = (150.0-tql[19[119][2]]-tqc[19[119][3]])*(1.0/p)*
    (t[i]/(25+273.15))*(1.0/(cu(100.0)))
    *(1.0/60.0)*ro[i];
    /* gm mole / sec */

/* calculate necessary areas and reaction length, all units in meters */

delw = ll/ nw;      /* artificial wafer spacing */

aw = PI * sq(wdia / 2.0); /* wafer area */
ai = PI * sq(idia / 2.0); /* inject area */
cat = PI * sq(lid / 2.0); /* cross area of tube */
sat = PI * (lid) * fl; /* surface area of tube */

for (i=1; i<=(N+1); i++) {
    if ( i < floor(pfirstw/delz) || i > floor((pfirstw+ll)/delz) ) {
        Lr[i] = (2.0*PI/delw)*(lid/2.0)*delw; /* reaction length outside
        wafer region */
    }
    else
        Lr[i] = (2.0*PI/delw)*(sq(wdia/2.0)+(lid/2.0)*delw); /* reaction length
        in wafer region */
}

/* calculate the Atw for each annular area */

for (i=1; i<=(N+1); i++) {
    if ( i < floor(pfirstw/delz) || i > floor((pfirstw+ll)/delz) ) {
        Atw[i] = cat; /* out of wafer region */
    }
    else
        Atw[i] = cat - aw; /* in wafer region */
}

/* end of area and reaction length calculations */

/* calculate the positions, from the load end, injectors */

/* injector positions by element */
center_inject_elem = rint( pci / delz);
source_inject_elem = rint( psi / delz);

/* end of positioning */

/* Calculate the constants for the runs */
kinet[Cs(N, p, t, D, kl, Ks, Kh, C_tot, kk0, ks0, kh0, Ea);

```

```

/* Call function for the empirical injection F(z) */
injection(F, ai, delw, qc, qs, center_inject_elem, source_inject_elem,
          N, delz, Atw, p, t, con);

/* Call function for the initial guesses */
init_guess(ql, qc, qs, Lr, ll, N, si, X, F, delz, Atw, C_tot,
           kl, Ks, Kh, D);

/* Main body of program including the loop for the determination of the
   values X and si */

max_del_X = 200.;
max_del_si = 200.;

while (max_del_si > 0.0001*si[N] || max_del_X > 0.0001*C_tot[1]) {
    velocity(N, si, ql, Atw, X, F, delz, Lr, del_si, kl, Ks, Kh,
             D, C_tot);

    conv_diff(N, si, ql, Atw, X, F, delz, Lr, del_X, kl, Ks, Kh,
              D, C_tot);

    max_del_X = fabs(del_X[1]);
    max_del_si = fabs(del_si[1]);

    for (i=2; i<=(N+1); i++) {
        if (fabs(del_si[i]) > max_del_si) {
            max_del_si = fabs(del_si[i]);
        }
        if (fabs(del_X[i]) > max_del_X) {
            max_del_X = fabs(del_X[i]);
        }
    }
}

file_out(N, C_tot, X, kl, Ks, Kh, delz, delw, pfirstw, nw, outer);

/*   printf(" THATS ALL FOLKS \n\n"); */

/* exit(2); */

} /* 19 loop */

sp = fopen("sn_out.dat", "r");
sncount = 0.0;
for (ll9 = 1; ll9<=9; ll9++) {
    fscanf(sp, "%10s", input);
    sncount = sncount + atof(input);
}
printf("sn = %f\n", sncount);

} /* end bracket */

/* in this subroutine the values for the reaction equation and
   diffusion constant */

double kinetics(n, p, t, D, kl, Ks, Kh, C_tot, kk0, ks0, kh0, Ea)
int n;
double p, t[301], D[301], kl[301], Ks[301], Kh[301], C_tot[301];
double kk0, ks0, kh0, Ea;
{
    /* p : total system pressure. */

```

```

/* t : system temperature. */
double tR, k0, Va, Vb, Ma, Mb, V, M, con, con1;
int i;

/* Calculate the total concentration */

/* From Jensen's paper: Kh = (0.6 +/- 0.3) x 10^2 atm^-1/2 and
   Ks = (0.7 +/- 0.1) x 10^5 atm^-1. The constant of the Arrhenius
   reaction equation is (1.6 +/- 0.4) x 10^9 mol/m^2/s/atm. */

/* Also place the terms in correct units with concentrations */
for (i=1; i<=(n+1); i++) {
    con = GC * t[i];

    k0 = kk0*con; /* units mol*m/gm mole*sec */
    k1[i] = k0 * exp((-Ea)/((1.987e-3)*t[i]));
    Ks[i] = ks0*con; /* units m^3/gm mole */
    Kh[i] = kh0*sqrt(con); /* units m^3/gm mole */
}

/* these are the values and calculation for the diffusion equation */
Va = 14.3; /* Atomic volume hydrogen */
Vb = 46.8; /* Atomic Volume silane */
Ma = 2.016; /* Molecular weight of hydrogen */
Mb = 32.1175; /* Molecular weight of silane */

V = sq((pow(Va, (double) (1./3.)) + pow(Vb, (double) (1./3.))));
M = sqrt( (1.0 / Ma) + (1.0 / Mb) );

for (i=1; i<=(n+1); i++) {
    tR = ((t[i]-273.15)*9.0/5.0)+32 + 460.0;
    D[i] = .0069 * (pow(tR, (double) (3./2.)) / p) * (M / V);

    /* Convert D [ft^2/ hr] to [m^2/sec] */
    con1 = (1.0/3600.0) * (0.0929);
    D[i] = D[i] * con1;
}

/* initialize the C_tot array */
for (i=1; i<=(n+1); i++) {
    C_tot[i] = p / (GC * t[i]);
}
/* end of routine */
}

double injection(f1, a11, delw1, qcl, qsl, center1, sourcel,
                n, delz, Atw, p, t, con)
int n, center1, sourcel;
double f1[301], a11, delw1, qcl, qsl, delz;
double Atw[301], p, t[301], con;
{
    /* Define local variables */

```



```

double lic, lis, conl, fs[301], fc[301];
int bnc, bns, errorc, errors, il, i2, i3;

/* constant function spread over distance of spray */

/* estimate injection length based on 60 sccm covers 20 wafers */
/* at 625 C and .250 torr */

conl = (con * delwl) / (60.0*(760.0/.250)*((625+273.15)/(25+273.15))*
                      (1.0/(cu(100.0)))*(1.0/60.0)*
                      ((.250/760)/(GC*(625+273.15))) / ail);
lic = conl * (qcl / ail);
lis = conl * (qsl / ail);

/* spread injection over injection length (lic and lis) */

bnc = centerl - floor(lic / delz);
bns = sourcel - floor(lis / delz);

if ( bnc < 1 ) {
    errorc = (1 - bnc) + 1;
    bnc = 1;
}
else
    errorc = 1;

if ( bns < 1 ) {
    errors = (1 - bns) + 1;
    bns = 1;
}
else
    errors = 1;

/* find injection for center injector */

for (il = 1; il<=(n+1); il++) {
    fc[il] = 0.0;
    fs[il] = 0.0;
}

i2 = 1;
for (i1 = bnc; i1<=centerl; i1++) {
    fc[i1] = (((2./(centerl-bnc+1))*qcl)/(centerl - bnc + 2))* i2;
    i2 = i2 + 1;
}

/* find injection for source injector */

i1 = 1;
for (i2 = bns; i2<=sourcel; i2++) {
    fs[i2] = (((2./(sourcel-bns+1))*qsl)/(sourcel - bns + 2)) * i1;
    i1 = i1 + 1;
}

/* find injector amounts for all positions of the furnace */

for (i3 = 1; i3<=(n+1); i3++) {
    fl[i3] = fc[i3] + fs[i3];
}

/* relocate the injection sites from elements to nodes */

for (i3 = 1; i3<=(n+1); i3++) {
    fl[i3] = fl[i3] / delz;
}

```

```

}

/* This function defines the initial guess for the Newton-Raphson
   solution of the model */

double init_guess(ql, qc, qs, Lr, ll, n, si, X, F, delz, Atw, C_tot,
                  kl, Ks, Kh, D)
{
    int n;
    double ql, qc, qs, Lr[301], ll, si[301], X[301], F[301], delz, Atw[301];
    double C_tot[301], kl[301], Ks[301], Kh[301], D[301];

    /* local variables */
    int i;
    double X_guess, V[301], react_rate, R[301], Q[301];

    /* First calculate the constant X_silane */
    /* fix X[] */
    for (i=1; i<=(n+1); i++) {
        X[i] = (1./2.);
    }

    X_guess = (1./2.);

    /* calculate the velocity */

    /* The constant reaction rate */
    for (i=1; i<=(n+1); i++) {
        react_rate = (kl[i]*X_guess*C_tot[i]) / ((1+Ks[i]*X_guess*C_tot[i]
                                                    +Kh[i]*sqrt(C_tot[i]*(1. - X_guess))));
        R[i] = Lr[i]*react_rate;
    }
    /* based on conservation of flow since the density is constant */

    Q[1] = ql + F[1]*delz;
    V[1] = Q[1] / (Atw[1]*C_tot[1]);

    for (i=2; i<=n; i++) {
        Q[i] = Q[i-1] + (R[i] + F[i])*delz;
        V[i] = Q[i] / (Atw[i]*C_tot[i]);
    }

    /* calculate si[] */
    si[n+1] = 0.0;
    for (i=n; i>=1; i--) {
        si[i] = si[i+1] + V[i]*delz;
    }
}

/* This program evaluates the Velocity equation for
   the deposition of polysilicon in an annular diffusion furnace*/

/*cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc*/

/* Starting the function velocity() */

double velocity(n, si, ql, Atw, X, F, delz, Lr, del_si, kl, Ks, Kh,

```

```

        D, C_tot)
int n;
double delz, si[301], q1, Atw[301], X[301], F[301], Lr[301];
double dal_si[301], k1[301], Ks[301], Kh[301], D[301], C_tot[301];

/* n: #of nodes;
   si[]: 'old' value of the potential, which is actually the
         'guessed' si[] for the first iteration. On return of
         'velocity()' si[] carries the new values of potential.
   X[]: is the previously established concentration profile,
         in the first iteration it will be the initial guess
         values.
   F[]: Injection array where F[i] is the injected flow
         into the ith element. Note ith element is between
         nodes i and (i+1). */

(
/* Declaration of the local variables */
double si_old[301], si_new[301], X_old[301];
double R[301], dR_by_dsi[301], react_rate, force[301], con1, con2, con3, con4;
double lower[301], upper[301], diag[301];
double dgtsl_();
int i, info, npl;

/*cccccccccccccccccccccccccccccccccccccccccccccccccccccccc*/
/* Assign si[] to si_old[] for later use */

for(i=1; i<=(n+1); i++) {
    si_old[i] = si[i];
    X_old[i] = X[i];
}

/*cccccccccccccccccccccccccccccccccccccccccccccccccccccccc*/
/* dR_by_dsi = dR/dsi at si_old */
/* react_rate is the reaction rate */

for(i=1; i<=(n+1); i++) {
    react_rate = (k1[i]*C_tot[i]*X_old[i]) / (1. + Ks[i]*C_tot[i]*X_old[i] +
                                                Kh[i]*sqrt(C_tot[i]*(1.-X_old[i])));
    R[i] = Lr[i]*react_rate;

    con1 = (C_tot[i]*X_old[i])*(Ks[i]-(Kh[i]*pow(C_tot[i]*(1.-X_old[i]),
                                                (double) -0.5))/2.);
    con2 = (1.0 + Ks[i]*C_tot[i]*X[i]+Kh[i]*sqrt(C_tot[i]*(1. - X_old[i])));
    dR_by_dsi[i] = -1.*Lr[i]*(react_rate / D[i])*(1.0 - con1/con2);
}

/*cccccccccccccccccccccccccccccccccccccccccccccccccccccccc*/
/* now let's solve the potential equation */
/* Defining the coefficient matrix */

/* for i = 1 */

lower[1] = 0.0;
upper[1] = -1.0*(2.0*Atw[2]*C_tot[2]/delz)*(1.0/delz);
diag[1] = ((2.0*Atw[2]*C_tot[2]/delz)*(1.0/delz) - (dR_by_dsi[1]));

/* for i = 2 to n-1 */

for(i=2; i<=(n-1); i++){
    lower[i] = -1.0*(Atw[i]*C_tot[i]/dalz)*(1.0/delz);

```

```

upper[i] = -1.0*(Atw[i+1]*C_tot[i+1]/delz)*(1.0/delz);
diag[i] = ((Atw[i]*C_tot[i]/(sq(delz))) + (Atw[i+1]*C_tot[i+1]/(sq(delz))) -
           dR_by_dsi[i]);
}

/* for i = n */

lower[n] = -1.0*(Atw[n]*C_tot[n]/delz)*(1.0/delz);
upper[n] = 0.0;
diag[n] = ((Atw[n]*C_tot[n]/(sq(delz))) + (Atw[n+1]*C_tot[n+1]/(sq(delz))) -
           dR_by_dsi[n]);

/* Defining the forcing function vector */

/* for i = 1 */

con1 = 2.0*Atw[2]*C_tot[2]/delz;
con2 = 0.0;
con3 = R[1] + F[1];
con4 = 1.0 / delz;
force[1] = con1*con4*(si_old[2] - si_old[1]) + con3 + 2*con4*q1;

/* for i = 2 to n-1 */

for(i=2;i<=(n-1);i++) {
con1 = 1.0 / delz;
con2 = (Atw[i]*C_tot[i]/delz);
con3 = R[i] + F[i];
con4 = (Atw[i+1]*C_tot[i+1]/delz);
force[i] = -1.0*con2*con1*(si_old[i] - si_old[i-1]) +
           (1.0*con1*con4)*(si_old[i+1] - si_old[i]) + con3;
}

/* for i = n */

con1 = (1.0/delz);
con2 = (Atw[n]*C_tot[n]/delz);
con3 = R[n] + F[n];
con4 = (Atw[n+1]*C_tot[n+1]/delz);
force[n] = -1.*con2*con1*(si_old[n] - si_old[n-1]) - con4*con1*(si_old[n])
           + con3;

/* Now the LINPACK subroutine is called */

for (i=1;i<=n;i++) {
lower[i-1] = lower[i];
diag[i-1] = diag[i];
upper[i-1] = upper[i];
force[i-1] = force[i];
}

lower[n] = 0.0;
diag[n] = 0.0;
upper[n] = 0.0;
force[n] = 0.0;

npl = n;

dgtsl_(&npl, lower, diag, upper, force, &info);

/* check if solution found */

if ( info != 0 ) {
printf("\n Matrix is singular, info = %d\n",info);
exit(3);
}

```

```

/* Assign the returned values of DGTSL to del_si
   and find si(i) at the new time step */

for(i=1;i<=n;i++){
del_si[i] = force[i-1];
si_new[i] = si_old[i] + del_si[i];
si[i] = si_new[i];
/* printf("si[%d] %e X[%d] %e\n",i,si[i],i,X[i]); */
}
si[n+1] = 0.0;

/*cccccccccccccccccccccccccccccccccccccccccccccccc*/
}

/* This program evaluates the Convection-Diffusion equation for
   the deposition of polysilicon in an annular diffusion furnace*/

/*cccccccccccccccccccccccccccccccccccccccccccccccc*/
/* Latest version as of DEC.17th 1987. */

/* Starting the function conv_diff() */
/* We need to specify the values 'si' for this function */

double conv_diff(n, si, ql, Atw, X, F, delz, Lr, del_X, kl, Ks,
                 Kh, D, C_tot)
int n;
double delz, si[301], ql, Atw[301], X[301], F[301], Lr[301];
double del_X[301], kl[301], Ks[301], Kh[301], D[301], C_tot[301];

/* n: #of elements; # of nodes is (n+1) going from 0 to n:
   si[]: potential calculated in the function 'velocity' for
        the current time step;
   X[]: 'old' concentration of Silane, which is actually the
        'guessed' X[] for the first iteration. On return of
        'conv_diff()', X[] carries the 'new' values of Silane
        concentration.
   F[]: Injection array where F[i] is the injected flow
        into the ith element. Note ith element is between
        nodes i and (i+1). */

{

/* Declaration of the local variables */
double X_old[301], X_new[301], X_tot[301], si_old[301];
double Bi_ip1[301], Bip1_i[301], Bi_im1[301], Bim1_i[301];
double R[301],dR_by_dX[301],react_rate, force[301];
double con.con1,con2,con3,con4;
double lower[301], upper[301], diag[301];
double dgtsl_(), dberl_(), bp, bn;
int i, info;

/*cccccccccccccccccccccccccccccccccccccccccccccccc*/
/* Assign X[] to X_old[] for later use */

for(i=1;i<=(n+1);i++){
X_old[i] = X[i];
si_old[i] = si[i];
}

```

```

/*cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc*/
/* dR_by_dX = dR/dX */
/* react_rate is the reaction rate */

for(i=1; i<=(n+1); i++) {
react_rate = (k1[i]*C_tot[i]*X_old[i])/(1. + Ks[i]*C_tot[i]*X_old[i] +
Kh[i]*sqrt(C_tot[i]*(1.-X_old[i])));
R[i] = Lr[i]*react_rate;

con1 = (Ks[i]*C_tot[i])-(Kh[i]*0.5*pow(C_tot[i]*(1.-X_old[i]), (double) -0.5));
con2 = 1. + Ks[i]*C_tot[i]*X_old[i] + Kh[i]*sqrt(C_tot[i]*(1.-X_old[i]));

dR_by_dX[i] = Lr[i]*react_rate*((1./X_old[i]) - (con1/con2));
}

/*cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc*/
/* Let us evaluate the vectors B[] where
Bi_ip1[i] = B[i,i+1], etc. */

/* @ node i = 1, Qo is introduced */

con = (si[1] - si[2])/ D[1];
dberl_(&con, &bp, &bn);
Bi_ip1[1] = bp * D[1];
Bip1_i[1] = bn * D[1];

/* @ node i = n+1, si[n+1] = 0.0 */

con = (si[n] - si[n+1])/ D[n];
dberl_(&con, &bp, &bn);
Bi_im1[n] = bp * D[n];
Bim1_i[n] = bn * D[n];

/* For i = 2, (n-1), the B[]'s are */

for(i=2; i<=(n-1); i++){

con = (si[i] - si[i+1])/ D[i];
dberl_(&con, &bp, &bn);
Bi_ip1[i] = bp * D[i];
Bip1_i[i] = bn * D[i];

con = (si[i] - si[i-1])/ D[i];
dberl_(&con, &bp, &bn);
Bi_im1[i] = bp * D[i];
Bim1_i[i] = bn * D[i];
}

/*cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc*/
/* now let's solve the convection-diffusion equation */
/* Defining the coefficient matrix */

lower[1] = 0.0;
upper[1] = -(2.*Atw[2]*C_tot[2]/delz)*((Bi_ip1[1])/delz);
diag[1] = (2.*Atw[2]*C_tot[2]/delz)*((Bip1_i[1])/delz)+dR_by_dX[1];

for(i=2; i<=(n-1); i++){
lower[i] = -(Atw[i]*C_tot[i]/delz)*(Bi_im1[i])/delz;
upper[i] = -(Atw[i+1]*C_tot[i+1]/delz)*(Bi_ip1[i])/delz;
diag[i] = ((Atw[i+1]*C_tot[i+1]/delz)*(Bip1_i[i]/delz)+(Atw[i]*C_tot[i]/delz)*
(Bim1_i[i]/delz) + dR_by_dX[i]);
}

```

```

lower[n] = -(Atw[n]*C_tot[n]/delz)*(Bi_lml[n])/delz;
upper[n] = 0.0;
diag[n] = ((Atw[n+1]*C_tot[n+1]/delz)*(si_old[n]/delz) +
           (Atw[n]*C_tot[n]/delz)*(Biml_i[n]/delz) + dR_by_dX[n]);

/* Defining the forcing function vector */

force[1] = (2.*Atw[2]*C_tot[2]/delz)*((Bi_lpl[1]*X_old[2]-Bipl_i[1]*X_old[1])/
                                         delz) - (R[1] - F[1]) + (2.*q1/delz);

for(i=2;i<=(n-1);i++){

force[i] = ((Atw[i+1]*C_tot[i+1]/delz)
            *((Bi_lpl[i]*X_old[i+1]-Bipl_i[i]*X_old[i])/delz)
            - (Atw[i]*C_tot[i]/delz)
            *((Biml_i[i]*X_old[i]-Bi_lml[i]*X_old[i-1])/delz) - (R[i]-F[i]));

}

force[n] = -(Atw[n+1]*C_tot[n+1]/delz)*(X_old[n]*si_old[n]/delz) -
            (Atw[n]*C_tot[n]/delz)*((Biml_i[n]*X_old[n]-Bi_lml[n]*X_old[n-1])/delz)
            - (R[n] - F[n]);

for(i=1;i<=n;i++){
    lower[i-1] = lower[i];
    diag[i-1] = diag[i];
    upper[i-1] = upper[i];
    force[i-1] = force[i];
}

lower[n] = 0.0;
diag[n] = 0.0;
upper[n] = 0.0;
force[n] = 0.0;

/* Now the LINPACK subroutine is called */

dgtsl_(&n, lower, diag, upper, force, &info);

if ( info != 0 ) {
    printf("\n Matrix is singular, info = %d\n",info);
    exit(4);
}

/* Assign the returned values of DGTSL to del_C
   and find C(i) at the new time step */

for(i=1;i<=n;i++){
    del_X[i] = force[i-1];
    X_new[i] = X_old[i] + del_X[i];
    X[i] = X_new[i];
    /* printf("si[%d] %e X[%d] %e\n",i, si[i],i, X[i]); */
}

X[n+1] = X[n];

/*cccccccccccccccccccccccccccccccccccccccccccccccccccccccc*/
}

```

```

/* This subroutine finds the correct wafer growth rates and pumps them
   to the file growth.dat */

double file_out(n, C_tot, X, kl, Ks, Kh, delz, delw, pfirstw, nw, outer)
  int n, nw, outer;
  double C_tot[301], X[301], kl[301], Ks[301], Kh[301], delw, delz;
  double pfirstw;

{
  /* local variables */

  FILE *sp, *spl;
  int i, j;
  double react_rate, wafer, X_prime, C_tot_prime, kl_prime, Ks_prime;
  double Kh_prime, least();

  sp = fopen("opt_growth.dat", "w");
  spl = fopen("opt_con.dat", "w");

  fprintf(spl, ".col node X\n");

  for (i=1; i<=n+1; i++) {
    fprintf(spl, "%d \t %e\n", i, X[i]);
  }

  wafer = pfirstw;
  j = 1;

  while (j <= nw) {
    i = floor(wafer/delz);
    if ( i == 0 ) {
      i = 1;
    }

    X_prime = ((X[i+1] - X[i])/1.0)*(frac(wafer/delz)) + X[i];
    C_tot_prime = ((C_tot[i+1] - C_tot[i])/delz)*(frac(wafer/delz)) + C_tot[i];
    kl_prime = ((kl[i+1] - kl[i])/1.0)*(frac(wafer/delz)) + kl[i];
    Ks_prime = ((Ks[i+1] - Ks[i])/1.0)*(frac(wafer/delz)) + Ks[i];
    Kh_prime = ((Kh[i+1] - Kh[i])/1.0)*(frac(wafer/delz)) + Kh[i];

    react_rate = (kl_prime*C_tot_prime*X_prime)/
      (1. + Ks_prime*C_tot_prime*X_prime + Kh_prime*
       sqrt(C_tot_prime*(1.-X_prime)));

    react_rate = react_rate*28.986*(1./2.328)*(1./(cu(100)))*60.*1e10;
      /* A/min */

    if ( j == 20 ) {
      fprintf(sp, "%f\n", react_rate);
    }

    if ( j == 26 ) {
      fprintf(sp, "%f\n", react_rate);
    }

    if ( j == 35 ) {
      fprintf(sp, "%f\n", react_rate);
    }

    if ( j == 45 ) {

```



```

    fprintf(sp, "%f\n", react_rate);
}

if ( j == 55 ) {
    fprintf(sp, "%f\n", react_rate);
}

if ( j == 65 ) {
    fprintf(sp, "%f\n", react_rate);
}

if ( j == 75 ) {
    fprintf(sp, "%f\n", react_rate);
}

if ( j == 85 ) {
    fprintf(sp, "%f\n", react_rate);
}

if ( j == 95 ) {
    fprintf(sp, "%f\n", react_rate);
}

if ( j == 105 ) {
    fprintf(sp, "%f\n", react_rate);
}

if ( j == 115 ) {
    fprintf(sp, "%f\n", react_rate);
}

if ( j == 124 ) {
    fprintf(sp, "%f\n", react_rate);
}

if ( j == 130 ) {
    fprintf(sp, "%f\n", react_rate);
}

j = j + 1;

wafer = wafer + delw;

}

fclose(sp);
fclose(spl);

least(outer);

/* end */
}

/* variance solver */

double least(outer)
    int outer;

{
    FILE *sp1, *sp2, *sp3, *fclose();
    double c, am, ap, v, m[15], p[15], sn, vp, vm;
    int i;
    char input[100];

    if ( outer == 1 ) {

```

```

    spl = fopen("1.dat", "r");
}
if ( outer == 2 ) {
    spl = fopen("2.dat", "r");
}
if ( outer == 3 ) {
    spl = fopen("3.dat", "r");
}
if ( outer == 4 ) {
    spl = fopen("4.dat", "r");
}
if ( outer == 5 ) {
    spl = fopen("5.dat", "r");
}
if ( outer == 6 ) {
    spl = fopen("6.dat", "r");
}
if ( outer == 7 ) {
    spl = fopen("7.dat", "r");
}
if ( outer == 8 ) {
    spl = fopen("8.dat", "r");
}
if ( outer == 9 ) {
    spl = fopen("9.dat", "r");
}

sp2 = fopen("opt_growth.dat", "r");
sp3 = fopen("sn_out.dat", "a");

/* input the measured values */
for (i=0; i<=12; i++) {
    fscanf(spl, "%10s", input);
    m[i] = atof(input);
}

/* input the predicted values */
for (i=0; i<=12; i++) {
    fscanf(sp2, "%10s", input);
    p[i] = atof(input);
}

/* solve for average */

am = 0.0;
ap = 0.0;

/* input the measured values */
for (i=1; i<=11; i++) {
    am = am + m[i];
    ap = ap + p[i];
}
am = am / 11.0;
ap = ap / 11.0;

/* solve for the variance */
v = 0.0;
for (i=1; i<=11; i++) {
    v = v + pow((p[i]-m[i]), (double) 2.0);
}
v = v / 11.0;

/* solve for SN */
sn = -10 * log10(v);

fprintf(sp3, "%f\n", sn);

fclose(sp2);
fclose(spl);
fclose(sp3);
}

```

B.2 bernoulli.f

```

C-----
C-----
C-----
SUBROUTINE dber(X,BP,DBP,BN,DBN)
C
C-----
C
C      THIS ROUTINE EVALUATES THE BERNOULLI FUNCTION
C      AND ITS DERIVATIVE FOR THE ARGUMENTS X AND -X
C-----
C
C      INPUT VARIABLE
C
C      X          - ARGUMENT OF THE BERNOULLI FUNCTION
C
C      OUTPUT VARIABLES:
C
C      BP          - BERNOULLI FUNCTION OF X
C      BN          - BERNOULLI FUNCTION OF -X
C      DBP         - DERIVATIVE OF BERNOULLI FUNCTION OF X
C      DBN         - DERIVATIVE OF BERNOULLI FUNCTION OF -X
C
C      DATA:
C
C      XLIM        - 10.**(-ND)
C      EPS         - 10.**(-NDIGIT)
C      ND          - NUMBER OF ADMITTED ERRONEOUS DIGITS
C      NDIGIT      - NUMBER OF SIGNIFICANT DIGITS
C-----
C
C      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C      DATA XLIM,EPS /0.01, 1.D-15/
C-----
C
C      Executable code.
C-----
C
C      AX=DABS(X)
C
C      COMPUTE THE ASYMPTOTIC VALUES OF THE BERNOULLI FUNCTION
C
C      IF (AX.GT.80.) THEN
C        IF (X.GT.0.) THEN
C          BP=0.
C          DBP=0.
C          BN=X
C          DBN=-1.
C          RETURN
C        ELSE
C          BP=-X
C          DBP=-1.
C          BN=0.
C          DBN=0.
C          RETURN
C        END IF
C      END IF
C
C      COMPUTE THE BERNOULLI FUNCTION IN THE INTERMEDIATE RANGE
C
C      IF (AX.GT.XLIM) THEN
C        ARG=DEXP(X)

```



```

C      BP      - Bernoulli function of X
C      BN      - Bernoulli function of -X
C
C      DATA:
C
C      XLIM     = 10.**(-ND)
C      EPS      = 10.**(-NDIGIT)
C      ND       = number of admitted erroneous digits
C      NDIGIT   = number of significant digits
C-----
C      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C      DATA XLIM, EPS/0.01D0, 1.D-15/
C-----
C      Executable code.
C-----
C
C      AX=DABS(X)
C
C      Compute the asymptotic values of the Bernoulli function.
C
C      IF (AX.GT.80.D0) THEN
C        IF (X.GT.0.D0) THEN
C          BP=0.D0
C          BN=X
C          RETURN
C        ELSE
C          BP=-X
C          BN=0.D0
C          RETURN
C        END IF
C      END IF
C
C      Compute the Bernoulli function in the intermediate range.
C
C      IF (AX.GT.XLIM) THEN
C        ARG =DEXP(X)
C        ARG1=ARG-1.D0
C        BP  =X/ARG1
C        ARG =1.D0/ARG
C        ARG1=ARG-1.D0
C        BN  =-X/ARG1
C      ELSE
C
C        Evaluate the Bernoulli function for small values of the argument.
C
C        I=1
C        FP=1.D0
C        FN=1.D0
C        DF=1.D0
C        GP=0.5D0
C        GN=0.5D0
C        DG=0.5D0
C        SIGN=1.D0
C
C        Evaluate the terms of the series.
C
C      10  I=I+1
C          AI=DFLOAT(I)
C          SIGN=-SIGN
C          DF=DF*X/AI
C          FP=FP+DF

```

```
FN=FN+DF*SIGN
IF (DABS (DF) .GT.EPS) GO TO 10
BP=1.D0/FP
BN=1.D0/FN
END IF
RETURN
END
```

B.3 dgtsl.f

```

subroutine dgtsl(n,c,d,e,b,info)
integer n,info
double precision c(301),d(301),e(301),b(301)

c
c
c dgtsl given a general tridiagonal matrix and a right hand
c side will find the solution.
c
c on entry
c
c   n      integer
c          is the order of the tridiagonal matrix.
c
c   c      double precision(n)
c          is the subdiagonal of the tridiagonal matrix.
c          c(2) through c(n) should contain the subdiagonal.
c          on output c is destroyed.
c
c   d      double precision(n)
c          is the diagonal of the tridiagonal matrix.
c          on output d is destroyed.
c
c   e      double precision(n)
c          is the superdiagonal of the tridiagonal matrix.
c          e(1) through e(n-1) should contain the superdiagonal.
c          on output e is destroyed.
c
c   b      double precision(n)
c          is the right hand side vector.
c
c on return
c
c   b      is the solution vector.
c
c   info   integer
c          = 0 normal value.
c          = k if the k-th element of the diagonal becomes
c              exactly zero. the subroutine returns when
c              this is detected.
c
c linpack. this version dated 08/14/78 .
c jack dongarra, argonne national laboratory.
c
c no externals
c fortran dabs
c
c internal variables
c
c integer k,kb,kpl,nml,nm2
c double precision t
c begin block permitting ...exits to 100
c
c   info = 0
c   c(1) = d(1)
c   nml = n - 1
c   if (nml .lt. 1) go to 40
c       d(1) = e(1)
c       e(1) = 0.0d0
c       e(n) = 0.0d0
c
c   do 30 k = 1, nml
c       kpl = k + 1
c
c       find the largest of the two rows
c
c       if (dabs(c(kpl)) .lt. dabs(c(k))) go to 10
c

```

```

c          interchange row
c
c          t = c(kpl)
c          c(kpl) = c(k)
c          c(k) = t
c          t = d(kpl)
c          d(kpl) = d(k)
c          d(k) = t
c          t = e(kpl)
c          e(kpl) = e(k)
c          e(k) = t
c          t = b(kpl)
c          b(kpl) = b(k)
c          b(k) = t
10      continue
c
c          zero elements
c
c          if (c(k) .ne. 0.0d0) go to 20
c          info = k
c          .....exit
c          go to 100
20      continue
c          t = -c(kpl)/c(k)
c          c(kpl) = d(kpl) + t*d(k)
c          d(kpl) = e(kpl) + t*e(k)
c          e(kpl) = 0.0d0
c          b(kpl) = b(kpl) + t*b(k)
30      continue
40      continue
c          if (c(n) .ne. 0.0d0) go to 50
c          info = n
c          go to 90
50      continue
c
c          back solve
c
c          nm2 = n - 2
c          b(n) = b(n)/c(n)
c          if (n .eq. 1) go to 80
c          b(nml) = (b(nml) - d(nml)*b(n))/c(nml)
c          if (nm2 .lt. 1) go to 70
c          do 60 kb = 1, nm2
c              k = nm2 - kb + 1
c              b(k) = (b(k) - d(k)*b(k+1) - e(k)*b(k+2))/c(k)
60      continue
70      continue
80      continue
90      continue
100     continue
c
c          return
c          end

```


C Appendix C - Measurement Data Sheets

Matrix1.e

process date: 9/14/87
process temperature: 625 C
process pressure : .200 torr
silane flows: 30/40/80 sccm
injector positions: D/C/6.875in R of C

wafer id	top	center	bottom	left	right	ave	unif
20	3908.00	3919.00	3889.00	3945.00	3864.00	3905.00	0.78
26	3911.00	4000.00	3982.00	3927.00	3990.00	3962.00	1.01
35	4046.00	4003.00	4005.00	4048.00	4012.00	4022.80	0.56
45	4062.00	4013.00	4022.00	4053.00	4027.00	4035.40	0.52
55	4134.00	4079.00	4118.00	4127.00	4125.00	4116.60	0.53
65	4216.00	4183.00	4237.00	4219.00	4223.00	4215.60	0.47
75	4400.00	4279.00	4480.00	4411.00	4400.00	4394.00	1.65
85	4628.00	4634.00	4693.00	4653.00	4630.00	4647.60	0.59
95	4658.00	4679.00	4844.00	4695.00	4668.00	4708.80	1.63
105	4648.00	4668.00	4821.00	4703.00	4650.00	4698.00	1.54
115	4488.00	4478.00	4646.00	4621.00	4471.00	4540.90	1.88
124	4205.00	4217.00	4271.00	4257.00	4177.00	4225.40	0.91
130	3926.00	3929.00	3961.00	3952.00	3908.00	3935.20	0.54

ave unif : 0.97
center ave: 4237.000 +/- 6.76
ave ave : 4262.092 +/- 7.07

deposition time: 75.000 min.

wafer id	average growth rate A/min
20	52.07
26	52.83
35	53.64
45	53.81
55	54.89
65	56.21
75	58.59
85	61.97
95	62.78
105	62.64
115	60.54
124	56.34
130	52.47

matrix2.e
 process date: 9/16/87
 process temperature: 625 C
 process pressure : .200 torr
 silane flows: 45/55/50 sccm
 injector positions: D/C/80875in R of C

wafer id	top	center	bottom	left	right	ave	unif
20	4023.00	4019.00	3984.00	4053.00	3950.00	4005.80	0.99
26	3966.00	4051.00	4023.00	3975.00	4028.00	4008.60	0.91
35	4040.00	3996.00	3990.00	4034.00	4002.00	4012.40	0.57
45	3978.00	3928.00	3936.00	3968.00	3951.00	3952.20	0.53
55	3928.00	3898.00	3921.00	3914.00	3924.00	3917.00	0.30
65	3920.00	3907.00	3969.00	3913.00	3910.00	3929.80	0.64
75	3934.00	3934.00	4037.00	3934.00	3967.00	3961.20	1.13
85	3952.00	3937.00	4008.00	3967.00	3971.00	3967.00	0.67
95	3966.00	3961.00	4036.00	3986.00	3986.00	3987.00	0.74
105	3969.00	3998.00	4226.00	4018.00	3993.00	4040.80	2.60
115	3907.00	3929.00	4087.00	3950.00	3933.00	3961.20	1.82
124	3819.00	3835.00	3886.00	3851.00	3815.00	3842.00	0.73
130	3389.00	3396.00	3437.00	3413.00	3371.00	3401.20	0.74

ave unif : 0.95
 center ave: 3906.846 +/- 4.19
 ave ave : 3922.015 +/- 4.19

deposition time: 75.000 min
 average
 wafer growth rate
 id A/min

20	53.41
26	53.45
35	53.50
45	52.70
55	52.23
65	52.40
75	52.82
85	52.89
95	53.26
105	53.88
115	52.82
124	51.23
130	45.35

matrix3.e
 process date: 9/16/87
 process temperature: 625 C
 process pressure : .200 torr
 silane flows: 60/70/20 sccm
 injector positions: D/C/10.875in R of C

wafer id	top	center	bottom	left	right	ave	unif
20	4386.00	4277.00	4248.00	4288.00	4207.00	4282.00	1.55
26	4206.00	4279.00	4264.00	4222.00	4272.00	4246.60	0.77
35	4264.00	4231.00	4223.00	4263.00	4232.00	4247.60	0.46
45	4159.00	4096.00	4104.00	4160.00	4126.00	4129.00	0.72
55	4004.00	3970.00	4011.00	3997.00	4011.00	3998.60	0.43
65	3939.00	3927.00	4008.00	3937.00	3970.00	3956.20	0.84
75	3889.00	3899.00	4014.00	3885.00	3924.00	3922.20	1.36
85	3826.00	3817.00	3844.00	3828.00	3829.00	3828.80	0.25
95	3621.00	3470.00	3489.00	3621.00	3609.00	3562.00	2.13
105	3397.00	3352.00	3427.00	3426.00	3375.00	3395.40	0.96
115	3269.00	3258.00	3385.00	3326.00	3254.00	3298.40	1.71
124	3128.00	3138.00	3230.00	3199.00	3109.00	3160.80	1.62
130	3010.00	3013.00	3029.00	3022.00	2862.00	2987.20	2.36

ave unif : 1.17
 center ave: 3748.231 +/- 11.96
 ave ave : 3770.077 +/- 11.73

deposition time: 75.000 min

wafer id	average growth rate A/min
20	57.08
26	56.65
35	56.57
45	55.05
55	53.31
65	52.75
75	52.30
85	51.05
95	47.49
105	45.27
115	43.98
124	42.14
130	39.83

matrix4.e
 process date: 9/18/87
 process temperature: 625 C
 process pressure : .250 torr
 silane flows: 30/55/65 sccm
 injector positions: D/C/10.875in R of C

wafer id	top	center	bottom	left	right	ave	unif
20	3480.00	3470.00	3453.00	3620.00	3432.00	3491.00	2.13
26	3453.00	3488.00	3487.00	3473.00	3602.00	3500.60	1.67
35	3648.00	3478.00	3577.00	3642.00	3593.00	3587.60	1.91
45	3635.00	3465.00	3475.00	3631.00	3484.00	3538.00	2.46
55	3673.00	3625.00	3685.00	3675.00	3687.00	3669.00	0.69
65	3831.00	3824.00	3874.00	3837.00	3845.00	3842.20	0.50
75	3886.00	3883.00	4011.00	3889.00	3906.00	3915.00	1.39
85	3981.00	3946.00	4021.00	3999.00	3990.00	3987.40	0.69
95	4058.00	4026.00	4112.00	4088.00	4056.00	4068.00	0.81
105	4211.00	4217.00	4456.00	4272.00	4209.00	4273.00	2.47
115	4241.00	4271.00	4672.00	4425.00	4258.00	4373.40	4.17
124	4143.00	4222.00	4648.00	4260.00	4148.00	4284.20	4.89
130	3948.00	3983.00	4100.00	4002.00	3945.00	3995.60	1.58

ave unif : 1.95
 center ave: 3838.308 +/- 7.96
 ave ave : 3886.538 +/- 8.02

deposition time: 75.000 min
 average
 wafer growth rate
 id A/min

20	46.55
26	46.67
35	47.83
45	47.17
55	48.92
65	51.7
75	52.20
85	53.17
95	54.24
105	56.97
115	58.31
124	57.12
130	53.27

matrix5.e
 process date: 9/16/87
 process temperature: 625 C
 process pressure : .250 torr
 silane flows: 45/70/35 sccm
 injector positions: D/C/6.875in R of C

wafer id	top	center	bottom	left	right	ave	unif
20	4147.00	4134.00	4087.00	4179.00	4042.00	4117.80	1.31
26	4060.00	4171.00	4129.00	4089.00	4161.00	4122.00	1.14
35	4159.00	4087.00	4084.00	4156.00	4096.00	4116.40	0.92
45	4077.00	4017.00	4032.00	4065.00	4051.00	4048.40	0.60
55	4016.00	3981.00	4037.00	4008.00	4034.00	4015.20	0.56
65	4007.00	4000.00	4120.00	4013.00	4057.00	4039.40	1.24
75	4016.00	4035.00	4222.00	4022.00	4089.00	4076.80	2.11
85	3974.00	3971.00	4088.00	4000.00	4012.00	4009.00	1.18
95	3921.00	3929.00	4067.00	3957.00	3954.00	3965.60	1.48
105	3836.00	3839.00	3946.00	3858.00	3835.00	3862.80	1.23
115	3482.00	3464.00	3648.00	3620.00	3481.00	3539.00	2.47
124	3267.00	3260.00	3327.00	3315.00	3256.00	3285.00	1.02
130	3052.00	3043.00	3067.00	3056.00	3040.00	3051.60	0.35

ave unif : 1.20
 center ave: 3840.846 +/- 9.24
 ave ave : 3865.308 +/- 9.02

deposition time: 75.000 min
 average
 wafer growth rate
 id A/min

20	54.90
26	54.96
35	54.89
45	53.98
55	53.54
65	53.86
75	54.76
85	53.45
95	52.87
105	51.50
115	47.19
124	43.80
130	40.69

matrix6.e
 process date: 9/21/87
 process temperature: 625 C
 process pressure : .250 torr
 silane flows: 60/40/50 sccm
 injector positions: D/C/8.875in R of C

wafer id	top	center	bottom	left	right	ave	unif
20	4550.00	4547.00	4494.00	4574.00	4444.00	4521.80	1.16
26	4400.00	4529.00	4497.00	4428.00	4500.00	4470.80	1.21
35	4486.00	4404.00	4393.00	4470.00	4424.00	4435.40	0.92
45	4307.00	4213.00	4215.00	4268.00	4249.00	4250.40	0.92
55	4106.00	4051.00	4085.00	4091.00	4104.00	4087.40	0.54
65	4056.00	4017.00	4080.00	4046.00	4063.00	4052.40	0.58
75	4044.00	4025.00	4128.00	4041.00	4062.00	4060.00	0.99
85	4037.00	4007.00	4074.00	4044.00	4042.00	4040.80	0.59
95	4043.00	4025.00	4125.00	4058.00	4054.00	4061.00	0.94
105	4032.00	4073.00	4418.00	4079.00	4064.00	4133.20	3.88
115	3953.00	3989.00	4201.00	4005.00	3988.00	4027.20	2.46
124	3819.00	3844.00	3922.00	3860.00	3830.00	3855.00	1.05
130	3420.00	3427.00	3479.00	3445.00	3425.00	3439.20	0.70

ave unif : 1.23
 center ave: 4088.538 +/- 7.23
 ave ave : 4110.354 +/- 6.89

deposition time: 75.000 min
 average
 wafer growth rate
 id A/min

20	60.29
26	59.61
35	59.14
45	56.67
55	54.50
65	54.03
75	54.13
85	53.88
95	54.15
105	55.11
115	53.70
124	51.40
130	45.86

matrix7.e
 process date: 9/21/87
 process temperature: 625 C
 process pressure : .350 torr
 silane flows: 30/70/50 sccm
 injector positions: D/C/8.875in R of C

wafer id	top	center	bottom	left	right	ave	unif
20	3686.00	3661.00	3631.00	8710.00	3614.00	3660.40	1.07
26	3624.00	3693.00	3684.00	3655.00	3713.00	3673.80	0.95
35	3822.00	3667.00	3681.00	3820.00	3712.00	3740.40	2.01
45	3736.00	3659.00	3701.00	3733.00	3721.00	3710.00	0.85
55	3846.00	3836.00	3883.00	3854.00	3875.00	3858.80	0.51
65	3901.00	3903.00	4042.00	3906.00	3982.00	3946.80	1.60
75	3996.00	4024.00	4274.00	4016.00	4095.00	4081.00	2.80
85	4059.00	4057.00	4216.00	4095.00	4129.00	4111.20	1.60
95	4092.00	4090.00	4242.00	4136.00	4153.00	4142.60	1.50
105	4103.00	4170.00	4655.00	4210.00	4190.00	4265.60	5.19
115	4010.00	4063.00	4437.00	4087.00	4068.00	4133.00	4.17
124	3864.00	3883.00	4015.00	3906.00	3877.00	3909.00	1.57
130	3449.00	3449.00	3484.00	3471.00	3461.00	3462.80	0.43

ave unif : 1.86
 center ave: 3858.077 +/- 5.66
 ave ave : 3899.646 +/- 6.15

deposition time: 75.000 min
 average
 wafer growth rate
 id A/min

20	48.81
26	48.98
35	49.87
45	49.47
55	51.45
65	52.62
75	54.41
85	54.82
95	55.23
105	56.87
115	55.11
124	52.12
130	46.17

matrix8.e
 process date: 9/23/87
 process temperature: 625 C
 process pressure : .350 torr
 silane flows: 45/40/65 sccm
 injector positions: D/C/10.875in R of C

wafer id	top	center	bottom	left	right	ave	unif
20	4105.00	4093.00	4057.00	4135.00	4031.00	4064.20	1.00
26	3978.00	4051.00	4031.00	4000.00	4044.00	4020.80	0.77
35	4013.00	3955.00	3955.00	4013.00	3976.00	3982.40	0.73
45	3921.00	3876.00	3881.00	3918.00	3901.00	3899.40	0.53
55	3872.00	3845.00	3869.00	3872.00	3873.00	3866.20	0.31
65	3876.00	3861.00	3925.00	3896.00	3896.00	3890.80	0.62
75	3933.00	3916.00	4042.00	3940.00	3961.00	3958.40	1.25
85	4032.00	3996.00	4075.00	4050.00	4036.00	4037.80	0.71
95	4132.00	4097.00	4195.00	4184.00	4127.00	4147.00	0.99
105	4286.00	4408.00	4414.00	4492.00	4396.00	4399.20	1.68
115	4440.00	4624.00	4855.00	4641.00	4476.00	4607.20	3.57
124	4260.00	4462.00	4784.00	4620.00	4282.00	4481.60	4.99
130	4045.00	4096.00	4247.00	4134.00	4064.00	4117.20	1.94

ave unif : 1.47
 center ave: 4098.462 +/- 6.06
 ave ave : 4114.785 +/- 5.76

deposition time: 75.000 min
 average
 wafer growth rate
 id A/min

20	54.46
26	53.61
35	53.10
45	51.99
55	51.55
65	51.88
75	52.78
85	53.84
95	55.29
105	58.66
115	61.43
124	59.75
130	54.90

matrix9.e
 process date: 9/23/87
 process temperature: 625 C
 process pressure : .350 torr
 silane flows: 60/55/35 sccm
 injector positions: D/C/6.875in R of C

wafer id	top	center	bottom	left	right	ave	unif
20	4742.00	4734.00	4701.00	4768.00	4692.00	4727.40	0.66
26	4665.00	4699.00	4684.00	4676.00	4700.00	4684.80	0.32
35	4670.00	4637.00	4637.00	4667.00	4651.00	4652.40	0.34
45	4273.00	4238.00	4246.00	4273.00	4267.00	4259.40	0.38
55	4174.00	4097.00	4170.00	4167.00	4181.00	4157.80	0.83
65	4109.00	4060.00	4189.00	4121.00	4158.00	4127.40	1.19
75	4063.00	4059.00	4239.00	4088.00	4160.00	4121.80	1.87
85	3989.00	3967.00	4121.00	4019.00	4012.00	4021.60	1.47
95	3926.00	3927.00	4124.00	3988.00	3942.00	3981.40	2.10
105	3842.00	3843.00	3968.00	3873.00	3850.00	3875.20	1.38
115	3415.00	3406.00	3467.00	3452.00	3417.00	3431.40	0.77
124	3189.00	3170.00	3246.00	3232.00	3181.00	3203.60	1.04
130	3033.00	2668.00	3037.00	3036.00	3032.00	2961.20	5.54

ave unif : 1.38
 center ave: 3961.923 +/- 15.17
 ave ave : 4015.800 +/- 13.63

deposition time: 75.000 min

wafer id	average growth rate A/min
20	63.03
26	62.46
35	62.03
45	56.79
55	55.44
65	55.03
75	54.96
85	53.62
95	53.09
105	51.67
115	45.75
124	42.71
130	39.48

optimized parameter run
 process date: 12/4/87
 process temperature: 625 C
 process pressure : .200 torr
 silane flows: 45/40/65 sccm
 injector positions: D/C/8.875in R of C

wafer id	top	center	bottom	left	right	ave	unif
20	3905.00	3887.00	3651.00	3976.00	3824.00	3888.60	1.49
26	3872.00	3874.00	3862.00	3828.00	3930.00	3873.20	0.95
35	3885.00	3844.00	3849.00	3895.00	3855.00	3865.60	0.59
45	3861.00	3826.00	3833.00	3856.00	3845.00	3844.20	0.39
55	3835.00	3700.00	3830.00	3842.00	3837.00	3808.80	1.60
65	3852.00	3827.00	3872.00	3852.00	3856.00	3851.80	0.42
75	3883.00	3869.00	3940.00	3884.00	3895.00	3894.20	0.70
85	3920.00	3899.00	3968.00	3962.00	3920.00	3933.80	0.76
95	3973.00	3960.00	4086.00	4035.00	3960.00	4002.80	1.40
105	3956.00	4010.00	4270.00	4065.00	3968.00	4053.80	3.16
115	3889.00	3894.00	4041.00	3985.00	3892.00	3940.20	1.76
124	3879.00	3822.00	3854.00	3850.00	3880.00	3857.00	0.62
130	3325.00	3312.00	3366.00	3400.00	3285.00	3337.60	1.36

ave unif : 1.17
 center ave: 3824.923 +/- 4.47
 ave ave : 3857.815 +/- 4.41

deposition time: 75.000 min
 average
 wafer growth rate
 id A/min

20	51.85
26	51.64
35	51.54
45	51.26
55	50.78
65	51.36
75	51.92
85	52.45
95	53.37
105	54.05
115	52.54
124	51.43
130	44.50