A STUDY IN DYNAMIC NEURAL CONTROL OF SEMICONDUCTOR FABRICATION PROCESSES

***JILL CARD**
Digital Equipment Corporation

ABSTRACT:
This paper describes a generic dynamic control system designed for use in semiconductor fabrication process control. The controller is designed for any batch silicon wafer process that is run on equipment having a high number of variables that are under operator control. These controlled variables include both equipment state variables such as power, temperature, etc. and the repair, replacement, or maintenance of equipment parts, which cause parameter drift of the machine over time. The controller consists of three principal components: 1) an automatically updating database, 2) a neural network prediction model for the prediction of process quality based on both equipment state variables and parts usage, and 3) an optimization algorithm designed to determine the optimal change of controllable inputs that yield a reduced operation cost, in-control solution. The optimizer suggests a set of least cost and least effort alternatives for the equipment engineer or operator. The controller is a PC driven software solution that resides outside the equipment and does not mandate implementation of recommendations in order to function correctly. The neural model base continues to learn and improve over time. An example of the dynamic process control tool performance is presented retrospectively for a plasma etch system. In this study, the neural networks exhibited overall accuracy to within 20% of the observed values of .986, .938, and .87 for the output quality variables of Etch Rate, Standard Deviation, and Selectivity, respectively, based on a total sample size of 148 records. The control unit was able to accurately detect the need for parts replacements and wet clean operations in 34 of 40 operations. The controller suggested chamber state variable changes which either improved performance of the output quality variables or adjusted the input variable to a lower cost level without impairment of output quality.

INTRODUCTION

Run-to-run control of semiconductor fabrication batch processes, such as plasma etch, photolithography, chemical vapor deposition, etc. is determined by maintenance of targeted values of post process metrology variables. These variables designed to measure the quality of the process are extremely sensitive to changes in the process chamber conditions such as pressures, temperatures, gas mixtures, etc. Additionally, the aging of process tool parts, times since last tool cleaning, and times since calibrations all impact the quality of the process performance through longer term parameter drifting. The run-to-run control of these processes is difficult due to the high number of variables that are available for operator manipulation. The number of continuous chamber variables can number more than 20. The added complexity due to 20 or more variously aging replaceable/repairable/calibrated part types makes the overall problem of run-to-run control beyond the capacity of a single operator's "intuition". We introduce a generic software tool designed to maintain process control for semiconductor processes that are principally effected by the above variable types. The tool consists of routines to construct an automatically updating

Currently at IBEX Process Technology, Inc, West Newbury, MA; jill.card@neumath.com

database, a neural network model for the prediction of quality metrics given input variables of both continuous and time-dependent type, and an operator solution optimization routine designed to both minimize cost and overall human effort. We introduce results of the implementation of this tool for a plasma etch process within Digital Semiconductor.

The difficult problems of run-to-run or real-time control of semiconductor fabrication tool controls are well addressed in the literature [1-6]. Current approaches to tool control include dynamical systems studies using traditional techniques of closed loop kinetics modeling[2, 7-9] and a static implementation using a predictive model followed by an optimization feedback for run-to-run correction capability[10-13]. We offer a software implementation based on the latter approach using a cascade correlation neural network for the critical prediction algorithm. While other such neural network approaches to quality metric prediction for semiconductor tools have been reported [3, 14-16], we offer a predictive model that accounts for both continuously modulated variables as well as time-dependent maintenance variables, such as parts cleaning, and replacement. This mixed model was developed to serve as a real time aid for the tool operator and engineer, and incorporates all the actions that are considered for maintenance of process control on a daily basis. In addition to the training and implementation of a neural network predictive model, a heuristic optimization algorithm has been developed to address production floor issues of time and cost. We minimize the total effort required to maintain the process while minimizing cost to reflect the operator's hesitation to adhere to a regime (software driven or not) that requires excessive numbers of variables to be adjusted or parts replaced at any single time.

DYNAMIC CONTROLLER SOFTWARE COMPONENTS

The Dynamic Controller software consists of three primary analysis sections: 1) automatic database update and quality data assurance, 2) univariate and multivariate sensitivity analyses based on the neural prediction model, and 3) optimization analysis, providing a cost ordered set of corrective action solutions. The benefits of this program for the semiconductor fabrication process are substantial. The automatically updating database component makes possible the run-to-run control structure and offers the engineer a fully integrated database that incorporates the process variables, the output quality variables, and the ages of all resident parts within the machine as a single record. Incorporation of the Dynamic Controller will yield more efficient repair actions and corrective action strategies than operator "intuition" alone. There is potential for a more effective parts replacement strategy than is currently in place by either a premature preventive maintenance replacement, or a delayed replacement at the time of part total wearout. Finally, the requirement to monitor via costly monitor quality wafers the output of the fabrication process can be reduced as the neural network model becomes more accurate over time in its projections of process quality. We describe the fundamental elements of each of the Dynamic Controller components, as well as their intended use by both process tool operators and engineers, below.

Automatically Updating Database

The automatically updating database merges information coming from 3 distinct information databases: 1. the input chamber information recorded by a tool interface. These variables represent the pressures, temperatures, gas flows, positions, etc. of critical variables at the time of the process tool action. 2. the output quality variables. These variables are measured by hand using precision metrology tools. Due to the expense in the metrology performance, these quality checks occur only as frequently as are needed to maintain process control during the production runs. For our study, the output quality variables were etch rate, standard deviation of the etch rate across the wafer, and oxide selectivity. 3. the maintenance log of all changes in process tool parts, calibrations, tool cleans, etc. This database includes a reference total tool operation minutes recorded. In so doing, the exact ages of the parts and times between calibrations and cleans is computed for each data record.

An added feature available in the Dynamic Controller software utility is the immediate display of all recorded variable values outside of "safe" tool operating limits. The engineer evaluates these potential invalid data records and either accepts as valid or rejects them prior to any further analysis. All data exceptions are listed in easily viewed format listings for engineer review and record. The merged database is created in MSAccess format and available for further use outside of the Dynamic Controller applications.

The merged database serves as the training, test, and validation set used in neural network model development. As the database grows with subsequent monitor wafer records, the neural model is replaced with a subsequent model based on previous records plus any new events occurring since previous model training. The rate of the retraining of networks can be as often as every record, at set time intervals, or upon indication that the model no longer is an accurate fit to current data. The anticipation is that the model becomes more accurate as time goes on, increasing the precision of the analytical suggestions.

Sensitivity Analysis

The Sensitivity Analyses are based on the utility of the neural network prediction model in being able to evaluate the change in output quality variables as a consequence of varying any of the input variables from the current nominal fabrication tool state. The analyses are displayed in univariate form, i.e. showing the change in output variable states as each input variable is varied across its operating range individually. This analysis consists of a single operating state when considering the maintenance activity variables. The state of repair or cleaning is analyzed with respect to its impact on the output quality variables. A multivariate analysis is also available, showing the outcome of the output quality variables in the face of multiple, simultaneous input variable changes from nominal.

The learning algorithm used for the network trained in the dynamic control study is called the cascade-correlation architecture, introduced by Scott Fahlman [17] in 1991. The learning begins with no hidden units present: input nodes are directly connected to output units. These weights are trained via a backpropagation algorithm until an asymptote is reached on the error reduction. Hidden units are added one at a time (or in vector candidate groups) and trained to maximize the correlation between the hidden unit's output and the residual error at the output of the current training vector. This is achieved by inputting a hidden node, which gets weight connections from all input nodes and from all pre-existing hidden nodes. The output from the new hidden node is not connected to the network. A training vector is presented to the network and the following value, S, is maximized, for:

$$S = \sum_{O} \left| \sum_{P} (V_P - \overline{V})(E_{P,O} - \overline{E}_O) \right|, \tag{1}$$

where V is the candidate unit value, E = the error vector of observed to predicted output values, P = the number of input training vectors, and 0 = number of nodes of the output vector. The value S represents the correlation between V and the output error observed. In this notation, Vp is the output candidate unit value for the *pth* of P input vectors. $\overline{V}$ is the mean candidate unit value over all P input vectors. Similarly, Ep,o is the error computed for the *oth* of O output nodes for the *pth* input vector, and E_o is the average error value for the *oth* output node across the set of P input vectors. S is maximized with respect to the adaptive weights, w_i, as follows:

$$\partial S / \partial w_i = \sum_{P,O} \sigma_O (E_{P,O} - \overline{E_O}) f'_P I_{i,P} \qquad (2)$$

where σ_O is the sign of the correlation between the candidate's value and output *O, f'p* is the derivative for pattern P of the activation function with respect to the sum of its inputs, and $I_{i,P}$ is the input the candidate unit receives from unit *i* for pattern P. A gradient ascent is performed to maximize S. Training proceeds with training patterns until no further improvement in S is attained. The node is then added to the hidden layer, its input weights frozen, and its output weights trained via backpropagation. Additional hidden nodes are added one at a time with the output vector of the previous hidden nodes cascaded through weights to subsequent units. In this way, the most current hidden candidate's weight structure is being trained to reduce the current residual error not explained by the previous hidden nodes.

Optimization Analysis

The Optimization Analysis available within the Dynamic Controller incorporates the knowledge that the objectives of the program include maintaining process control of the output variables at minimum overall cost and at least effort on the part of the operator or process engineer. Overall cost includes the cost of operating the etch tool given all input variable settings and the corresponding neural network output variable predictions. Each variable value is cost weighted by the amount of deviation from the variable process target value. The solution space to maintenance of output variable control is not generally unique, with many solutions available at total cost lower than the current observed (nominal) settings of the process tool. The Controller software capitalizes on this non-uniqueness by displaying, in either graphical, or spreadsheet format, the listing of solutions which meet process quality control objectives with total cost lower than current nominal

operating cost. The displays are given in terms of actual variable settings or more easily scanned relative differences from nominal. The following is a summary of an heuristic optimization algorithm which reduces cost and engineering effort while maintaining process control for the fabrication tool. Detailed description of the algorithm can be found in Card., et al[1].

We begin by defining **z**, a vector of size $m + p$, where m = number of input variables and p = the number of output variables for the etch tool model. We define the set $\Phi = \{z^j \in \Re^{m+p} : j \leq s \in I$; an s vector set$\}$ which represents all input variable combinations and their resulting predicted output variables comprising $\mathbf{z}^j$ that are likely candidates for reducing tool operation cost over the current (nominal) cost. All z^j are evaluated using $f(z)$, the objective function of overall operating cost for tool variable settings **z**.

$$f(\mathbf{z}^j) = \sum_{i=1}^{m} g_i(z_i^{\,j}) + K\left(\sum_{i=m+1}^{m+p} g_i(z_i^{\,j})\right) \qquad (3)$$

Where z_i = the *ith* input variable value of **z** for $i \leq m$ or the *(i-m)th* output variable for $m < i \leq m + p$,

g_i = the cost function associated with the *ith* variable, z_i,

K = a weighting coefficient $\geq p$ to reflect added emphasis on process control.

The optimization algorithm is a discrete approximator which starts with the transform of the continuous range of every input variable (continuous or maintenance) into a discrete set of values. Each of the continuous input variables are assigned discrete variable values graded from the lowest acceptable to highest acceptable operating value.

All replacement variables, defined as RF minutes since the time of last part replacement/cleaning, can take on only 2 discrete values: RF time = 0 (representing a replacement activity), or the current recorded RF minutes since last part replacement/cleaning.

We refer to a nominal or base vector as the current observed vector of variable values, z_0. We seek to alter the input variable values of $\mathbf{z}_o$ to $\mathbf{z}_{opt}$ s.t. $f(\mathbf{z}_{opt})$ is minimum over all $\mathbf{z}_j \in$ ☜.

The set of potential candidate vectors to be optimized is constructed as follows. We vary one input variable from its nominal value, holding all other input variables at the nominal values assigned in $\mathbf{z}_0$. The output variables, z_i for $i = m+1,\ m+2, \ldots\ m+p$, are the associated predicted neural network output values corresponding to the altered candidate input vector. If one or more output variable approaches its targeted value, exceeding some minimum threshold level, then the level of the altered input variable will be included for subsequent optimization analysis. We repeat the above assessment of varying a single variable from nominal over all discrete values for the chosen input variable and over all input variables.

A set of candidate variables and variable levels is constructed which contains a total of $n_1, n_2, \ldots, n_m$ variable values of significant effect for the total of m input variables. We construct the set of $\mathbf{s}$ vectors, $\mathbf{z}^j \in$ ☜ by creating $(\mathbf{s} = \prod_{i=1}^{m} n_i)$ input vectors representing the fully crossed levels and variables satisfying the above output variable threshold criteria. All other input variables within each vector not altered by the cross are set to the nominal value found within $\mathbf{z}_o$. The output variables of each $\mathbf{z}^j$ are determined by evaluating the neural network outputs based on the constructed input vector.

The vector $\mathbf{z}_{opt}$ is the vector for which $f(z^j)$ is minimum for all $\mathbf{z}^j \in$ ☜ . By including for optimization only value levels of input variables that have a beneficial univariate effect on output quality variables, we assume that multivariate input variable combinations including variables not exhibiting a large main effect will not have large cost reduction effects. This subsetting of possible variables to be altered effectively imposes a "least effort" solution by minimizing the number of variables to be altered simultaneously.

PLASMA ETCH EXAMPLE

Neural Network Modeling

The following example shows results of a study performed at Digital Semiconductor using an 8" plasma etch wafer process tool. The output quality variables to be predicted via neural modeling were the etch rate, the standard deviation of the etch across the wafer, and the selectivity of the etch for oxide versus photoresist material. These post etch quality variables are predicted by 3 independent neural networks, with input vector variables defined in Table 1. All networks were trained, tested and validated using a total sample of size 148 records ranging over 18 months of monitor wafer data collection. The records were randomly assigned to the three subgroups with a final distribution of samples being 96 Train, 30 Test, and 22 Validation. Table 2 contains the statistics of interest for the three networks, including: input layer and hidden cascade layer node number, root mean square values for the train, test, and validation samples, and accuracy measures. Figures la-c through 3a-c show the observed vs. projected output plots for the three output variables across the sample sets. Overall, the 3 prediction models indicate a close fit between the observed vs. predicted, with the noted exception of a single observation at number 9 of the Validation sample. The observed drop in output variables for that etch record was due to an incorrect gap setting on the etch tool that occurred following a maintenance operation. This type of

error is considered to be catastrophic in nature, rather than due to parameter drift. The control system of this study was not designed to capture such events. Consequently, the lack of fit is not seen as lack of confidence in the neural fit for the quality variable predictions.

Optimization Analysis

The optimization algorithm analysis was performed on the 148 data records. The intent is to show that for the model described, at the level of prediction accuracy exhibited, how well do output suggestions from the optimization algorithm match what are determined to be a sound and cost effective courses of action under a wide variety of process conditions. A total of 20 test cases were chosen from the total study sample. These cases include 10 records that involved a corrective action performance in actual fabrication operations. The remaining 10 cases were the records immediately following each of the corrective action events. In this way, we hoped to ascertain whether die controller recognized situations requiring corrective action as well as no action.

Table 3 lists the combined results of the controller suggestions for part replacement actions compared to the actual known fixes for the 10 records just preceding a corrective action. We tabulate the effectiveness based on total numbers of parts actually replaced versus suggested replacements. Among the 26 parts replaced, 19 were due to parameter drift or wear during the 10 events. Only 2/19 (10.5%) were definite misses, with up to 4 needed replacements (21%) being missed by the controller suggestions.

Results from the remaining 10 events that represent the monitor wafer etch runs immediately following a repair or replacement action are summarized in Table 4. The 15 correct responses by the optimizer include no request for change or replacement in parts which had just been replaced correctly since the preceding etch run. An additional response categorized by the heading, "still requesting part replacement", indicates the situation where a clamp attach was properly requested by the controller, but not replaced on the fabrication floor until several monitor runs later. In total there were 2/18 (11 %) incorrect responses from among the responses either requesting a part change or sensing a part that had been changed.

We conclude that for the replacement of parts, the controller is consistent in approximately 90% accurate sensing of both the need to replace parts as well as not to replace parts following an effective repair action.

The results of the continuous variable optimization analysis are listed in Table 5. The Table summarizes the significant optimizer information as follows: the nominal continuous variable is listed and the average change shown as a delta value for two subgroups of cases: those cases where the delta was an increase over nominal and those cases where it was negative. The number of cases out of the 20 test cases is also recorded beside the delta value. The most significant entry on Table 5 is the recommendation for an average increase of Argon Gas flow from 301 ccm to 308.76 ccm in 18 out of 20 test cases. The increase of the Ar gas flow will increase the Etch Rate, which is routinely operating slightly below its targeted value. There were 12 cases suggesting an Upper Electrode Setting of 39.60 ^{O}C, (4 cases suggesting increase from 39 ^{O}C ; 8 cases suggesting decrease from 40 ^{O}C). This suggests a general conclusion to change the process target setpoint to a lower cost of operation setting. The final continuous variable entry that is highlighted on the table is the recommended average decrease in RF Forward power from 1402 watts to 1396.46 watts in 10 of the cases. This should have the impact to lower Etch Rate and Standard Deviation somewhat, but also is movement in the direction of the 1400 watt operating target setpoint.

We conclude that for the continuous input variables, the controller attempts to both 1) minimize deviation of input variables from their targeted values when output variables aren't effected significantly, and 2) adjust input variables to maintain output variables at or near their targeted values.

DISCUSSION

The Dynamic Controller software package described in this paper was designed to offer the operator or engineer of a semiconductor fabrication process tool a run-to-run feedback regarding optimal maintenance actions and process recipe setpoint changes required to maintain output process quality control, while minimizing costs and effort. The software, as described offers both engineering "what-if" sensitivity analysis capability as well as optimization solution analyses.

We show the viability of the software tool in minimizing costs while maintaining process control for the plasma etch process for 8" wafer technology. The benefits to equipment and process engineering are 1) efficient corrective actions through accurate maintenance strategies, 2) cost effective parts replacements strategies based on exhibited fitness for use, 3) reduced need for quality monitoring wafer checks, given faith in the accuracy of the neural network model, 4) corrective action at the onset of process control problems without the process having to wander beyond acceptable limits, and 5) a tool which makes suggestions, rather than mandates. A mistake in the controller's assessment need never override an engineer's better judgement.

The current speed of analysis processing is sufficient for use as a batch mode run-to-run controller. Current worst case sensitivity and optimization analyses can be performed in under 2 minutes on a 233MHz PC, with the large majority of observed cases requiring process times of less than 30 seconds.

Continued development of commercially available software is being performed by NeuMath, Inc.. Product beta test is planned at both IBM and Lucent Technologies manufacturing facilities.

ACKNOWLEDGEMENTS

Many thanks to Bill Ziminsky and Mark Naimo for their assistance in the programming and design of this project.

REFERENCES

1. G. Ming, X.Zhe, and S. Youxian, "Design of model predictive controller based on neural systems," Control and Computers, vol. 25 - 2, pp. 29-36. 1997.

2. G. Lu, M. Bora, L.L. Tedder, and G.W. Rubloff, "Integrated dynamic simulation of rapid thermal chemical vapor deposition of polysilicon " IEEE Transactions on Semiconductor Manufacturing, vol. 11-1, pp.63-74, 1998.

3. S. Bushman, T.F. Edgar, and I Trachtenberg, "Modeling of plasma etch systems using ordinary least squares, recurrent neural network, and projection to latent structure models," Journal of the Electrochemical Society, vol.144-.4, pp. 1379-89, 1997.

4. N.S. Alvi, "Manufacturing process improvements using advanced control methodologies", in proceedings of 1995 Japan International Electronic Manufacturing Technology Symposium 1995, pp. 276-282.

5. R. Telfeyan, J. Moyne, A. Hurwitz, and J. Taylor, "Demonstration of a process-independent run-to-run controller", in Proceedings of the Symposium on Process Control, Diagnostics, and Modeling in Semiconductor Manufacturing 1995, pp. 48-60.

6. X.A. Wang and R.L. Mahajan, "Artificial neural network model-based run to run process controller", in Proceedings of the Symposium on Process Control, Diagnostics, and Modeling in Semiconductor Manufacturing, 1995, pp.32-47.

7. P. Spence, C. Schaper, and A. Kermani, "Concurrent design of an RTP chamber and advanced control system", Modeling and Simulation of Thin-Film Processing, pp. 347-58.

8. E. Zafiriou, H.W. Chiou, and R.A. Adomaitis,. "Nonlinear Model based run-to-run control for rapid thermal processing with unmeasured variable estimation", in Proceedings of the Symposium on Process Control, Diagnostics, and Modeling in Semiconductor Manufacturing, 1995, pp. 18-31.

9. S. Belikov and B. Friedland, "Closed-Loop adaptive control for rapid thermal processing," in Proceedings of the 34th Conference on Decision and Control, 1998, vol. 3, pp. 2476-81.

10. C.D. Himmel, T.S. Kim, A. Krauss, E.W. Kamen, and G.S. May, "Real-time predictivecontrol of semiconductor manufacturing processes using neural networks", in Proceedings of the 1995 American Control Conference, 1995, vol. 2, pp. 1240-4.

11. J.P Card, D.L. Sniderman, and C. Klimasauskas , "Dynamic Neural Control for a Plasma Etch Process," IEEE Transactions on Neural Networks, vol. 8-4, pp. 883-901, 1997.

12. J.A. Stefani, S. Poarch, S. Saxena, and P.K. Mozumder, "Advanced process control of a
CVD tungsten reactor", IEEE Transactions on Semiconductor Manufacturing, vol. 9-3, pp. 366-83, 1996.

13. J.R. Moyne, A.N. Chaudhry, and R. Telfeyan, "Adaptive Extensions to a multibranch run-to-run controller for plasma etching," Journal of Vacuum Science and Technology A, vol. 13-3-2, pp. 1787-91, 1995.

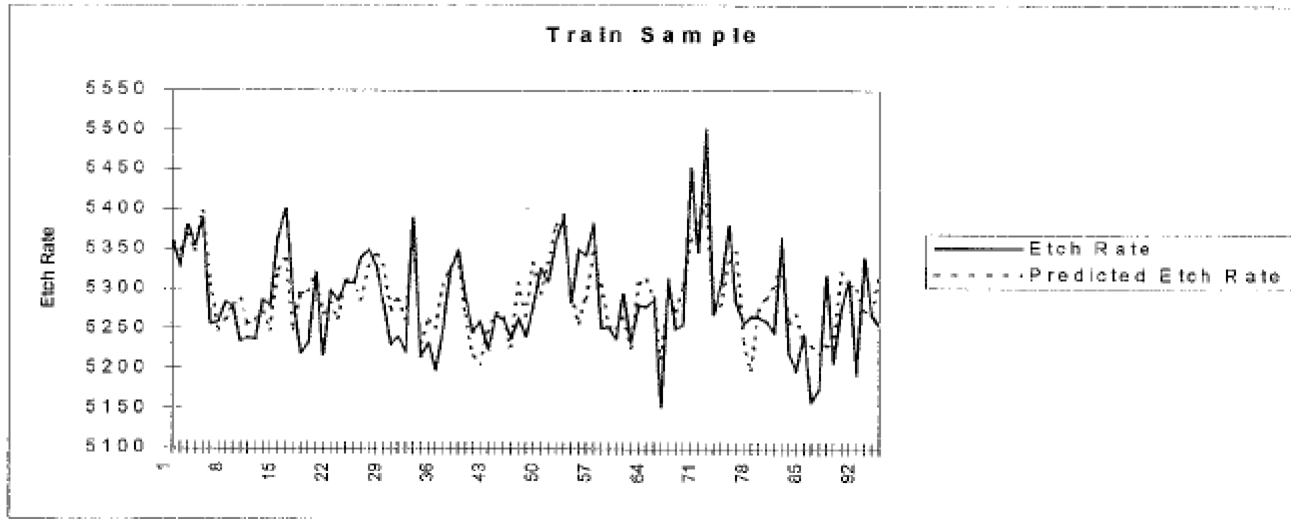
14. T.S. Kim and G.S. May, "Modeling of via formation by photosensitive dielectric materials for MCM applications," in Proceedings of 5th International Conference on Properties and Applications of Dielectric Materials, 1997, vol. 2, pp. 930-3.

15. J. Si, Y.L. Tseng, M. Clayton, S. Felker, B. Yoo, J. Martinez, J. Durham, Kim Dang, "Real time plasma etch process modeling by neural networks", in IEEE 6th International Conference on Emerging Technologies and Factory Automation Proceedings, 1997, pp. 347-52.

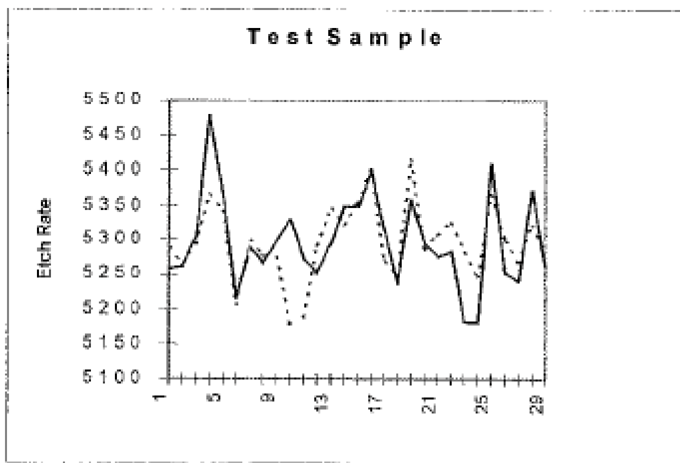
16. A.S. Kelkar, R.L. Mahajan, and R.L. Sani, "Real-time physiconeural solutions for MOCVD", Transactions of the ASMA Journal of Heat Transfer, vol. 118-4, 814-21,1996.

17. S.E. Fahlman and C. Lebiere, (1991). "The Cascade-Correlation Learning Architecture,"
Carnegie Mellon Univ., Pittsburah, PA, Tech. Rep. CMU-CS-90-100, 1991.

(a)



(b)



(c)

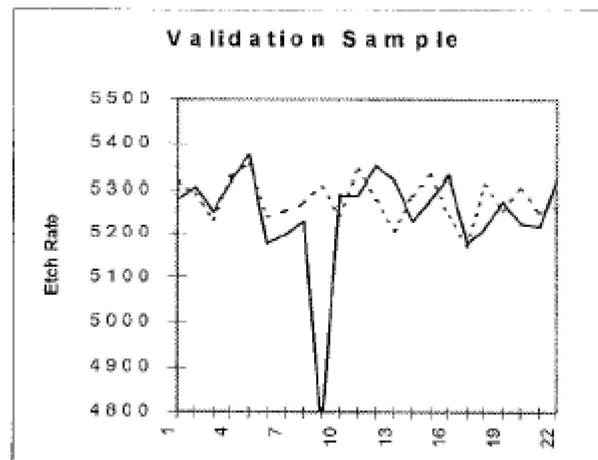
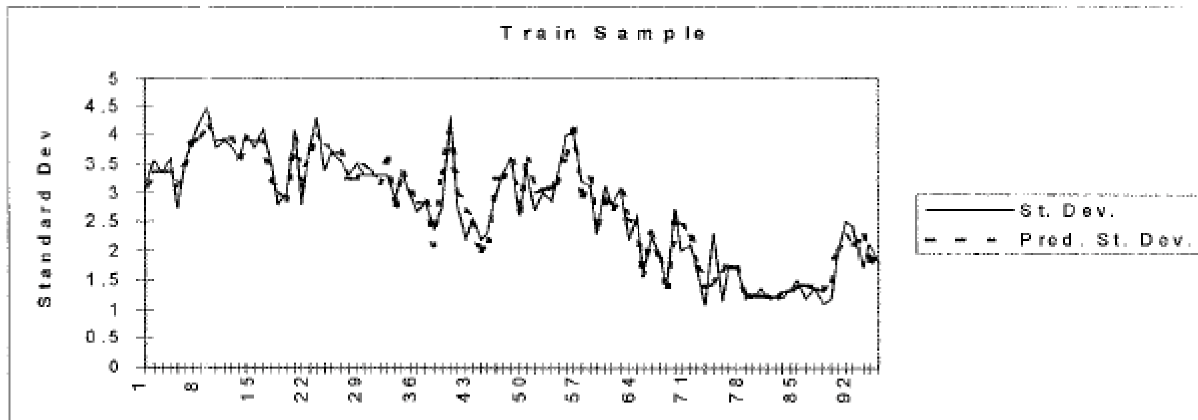
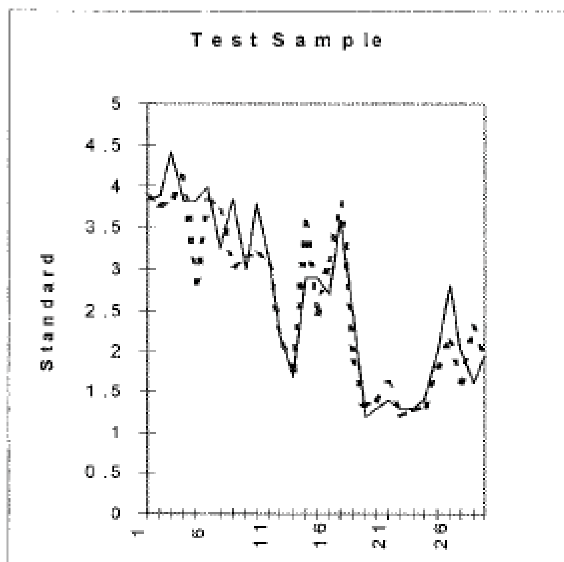


Figure 1: Neural network fit for Etch Rate on (a) Train, (b) Test, and (c) Validation samples.

(a)



(b)



(c)

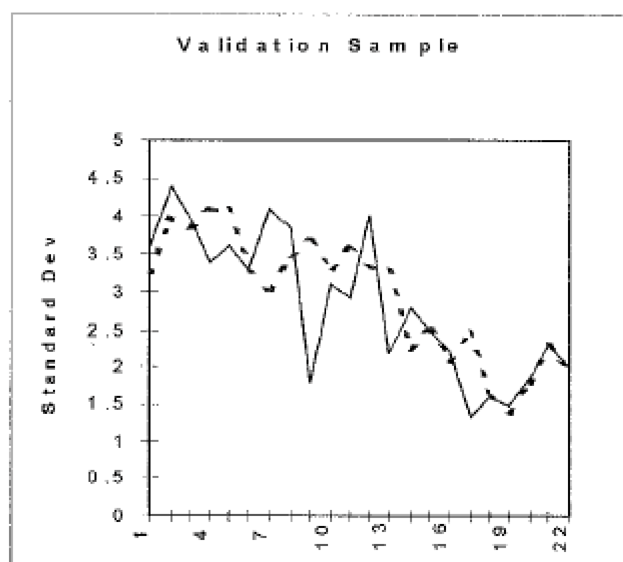


Figure 2: Neural network fit for Standard Deviation on (a) Train, (b) Test, and (c) Validation samples.