Lithography overlay controller formulation

Christopher A. Bode [*a], Anthony J. Toprac [a], Richard D. Edwards [a], and Thomas F. Edgar [b]

[a] Advanced Micro Devices, 5204 E. Ben White Blvd, MS 568 Austin, TX 78741
[b] The University of Texas at Austin, Austin, TX 78712-1062

ABSTRACT

Lithography overlay refers to the measurement of the alignment of successive patterns within the manufacture of semiconductor devices. Control of overlay has become of great importance in semiconductor manufacturing, as the tolerance for overlay error is continually shrinking in order to manufacture next-generation semiconductor products. Run-to-run control has become an attractive solution to many control problems within the industry, including overlay. The term run-to-run control refers to any automated procedure whereby recipe settings are updated between successive process runs in order to keep the process under control. The following discussion will present the formulation of such a controller by examining control of overlay. A brief introduction of overlay will be given, highlighting the control challenge overlay presents. A data management methodology that groups like processes together in order to improve controllability, referred to as control threads, will then be presented. Finally, a discussion of Linear Model Predictive Control will show its utility in feedback run-to-run control.

Keywords: Lithography overlay, run-to-run control, control threads, Linear Model Predictive Control

1. INTRODUCTION

Lithography overlay remains one of the more important challenges facing the semiconductor manufacturing industry. In a recent National Technology Roadmap for Semiconductors (NTRS), overlay control is listed as one of the five greatest challenges facing the lithography process.[1.] This challenge is due in large part to the necessary reduction in the overall overlay tolerance for 0.18 μm technology, which has been reduced to an estimated 65 nm (mean + 3σ) in the site measurement error. With this decrease in overlay error tolerance comes an increased need for more advanced control of the lithography process.

Overlay is the metric used to measure the alignment of successive patterned layers within the manufacture of semiconductor devices. At each point on the surface of a wafer, overlay (O) is defined as the difference between the position of the topmost layer (P_1) and the position of the preceding patterned layer (P_2).

$$O = P_1 - P_2 \qquad (1)$$

In practice, overlay is comprised of a number of parameters that describe the alignment of the patterns. Common examples of such include x- and y-translation, magnification, wafer and reticle rotation, x- and y-scale, and orthogonality.[2.]

Measurement of overlay error involves the use of special target structures within the lithography patterns. Such targets can take a number of forms, with one common example being a box-in-box structure. The outer box of such a target resides within the preceding pattern, while the inner box is part of the resist pattern of the current layer. The position of the inner box to the outer box of these targets generates a measurement of site x- and y-translation error. The set of these measurements, along with the location of each site, is fed to an overlay model such as that shown in Equation 2. [3.] Least-squares minimization techniques are then used to model the lot-averaged overlay parameters from the site measurements.

$$\begin{aligned} O_{X,x} &= T_X + E_X X - R_X Y + m_x x - r_x y + \rho_{X,x} \\ O_{Y,y} &= T_Y + E_Y Y + R_Y X + m_y y + r_y x + \rho_{Y,y} \end{aligned} \qquad (2)$$

* Correspondence: Email: christopher.bode@amd.com; Telephone: (512) 602-4379; Fax: (512) 602-5795

This model generates an estimate for mean wafer translations T_x and T_y, wafer scaling errors E_x and E_y, wafer rotation errors R_x and R_y, magnification parameters m_x and m_y, and reticle rotation r_x and r_y. The overlay model is fitted to each set of overlay measurements by minimizing the residual terms of the equations, $\rho_{X,x}$ and $\rho_{Y,y}$.

For each of the overlay parameters that can be calculated from the above model, there exists within the exposure tool recipe an adjustable parameter responsible for its control. Each of these variables can be assumed independent from the others, such that one adjustable recipe parameter will affect only its designated overlay error parameter. Typically, a unity gain between the recipe and error parameters exists so that a unit change in the recipe parameter can be assumed to make a unit change in the resulting error parameter. Equation 3 is a representation of this relationship.

$$y_i = u_i + c_i \tag{3}$$

The model shows the relationship between the recipe parameter u_i and its corresponding error parameter y_i with an intercept term c_i. The inclusion of the intercept term is necessary because a non-zero recipe setting is often required to produce no overlay error.

Process control of overlay is facilitated through this relationship between the recipe settings and the overlay error parameters. When one or more of the overlay error terms deviates from target, the corresponding manipulated variables within the recipe are adjusted to bring the process back to target. The challenge within such a method centers around estimating the value of the model intercept. It is this parameter that quantifies the control "state" of the process, and largely determines alignment performance. With an accurate measurement or estimation of the state, generation of the requisite recipe settings is straightforward.

Although it is easy to calculate the set of overlay states for a particular lot, it is quite challenging to predict overlay states based upon historical operation of a masking operation. A multitude of sources of variation exist which will affect the overlay state of the tool and the lot to be masked. Tools are subject to drift and abrupt changes in the state due to maintenance events performed on the tools. The lots themselves will have unique contributions to overlay state based on previous processing. For a given masking operation, the noise induced by these sources of error are significant. Figure 1 shows the overlay translation state for a particular masking operation, normalized to the control limits of that process. Even if the process was perfectly centered, without additional management a significant portion of the line would have error near or outside the control limits due to the noise present within the data.

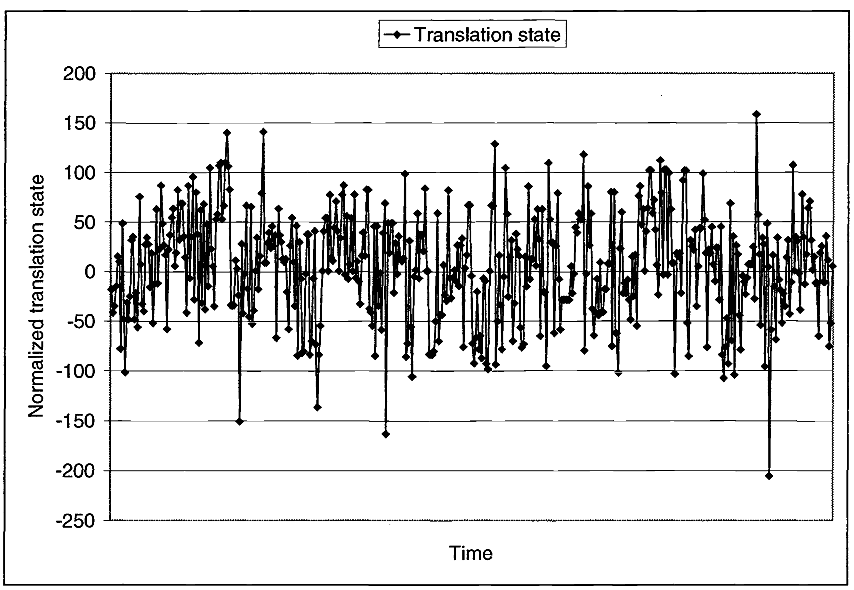

Figure 1: Illustration of noise within a single masking process

The nature of lithography overlay metrology creates one of the more challenging control problems within the industry. The performance of the masking process, at least in terms of overlay, is highly dependent upon the performance of the previous masking operation. This dependence is due to the fact that alignment is characterized by the position of the current layer as well as the previous layer. The preceding layer, in much the same manner, depended upon the layer that came before it. In a very real sense, the alignment of one layer is subject to the overlay performance of each layer that has been masked up to that point within the manufacturing process.

The foremost challenge within the control of overlay is this lack of an independent measurement of the performance of a given stepper. Overlay metrology is dependent upon the performance of at least two masking operations, with no additional reference to determine each operation's contribution to overlay process variation. Increased variation in one masking operation is as likely to be able to be attributed to the current operation as it is to the previous operation. This variation makes it very challenging to identify the overlay control state, and therefore the optimal control settings. Control efforts designed to manage overlay must then be cognizant of the interdependencies between each of the layers.

2. CONTROL THREADS

In the control of semiconductor manufacturing, the prevailing mindset it to treat each process run as a discrete time step within a continuous process. This is to say that while process performance may vary significantly over a long time period, the performance from lot-to-lot is assumed to be rather repeatable. This assumption is supported by the widespread use of Statistical Process Control (SPC), which aims to ensure that a process remains in control by assuring that it behaves in a repeatable manner over a short time horizon. While in many cases this assumption may be a valid, it is predicated on the belief that any lot within the manufacturing line will end up with the same results given equivalent processing.

For some processes, especially lithography in terms of overlay, this assumption is poor. Even granting that a tool will have perfect repeatability, the resultant overlay performance signal may still display significant variation due to that measurement's dependency on the position of the previous layer. This dependence will manifest itself through the incoming alignment state of the wafer lots. With overlay metrology dependent upon a referential measurement, the incoming state of the wafer contributes to the variation of the process as much as the process itself. Therefore, in addition to maintaining the process itself to ensure repeatable performance, one must also ensure that the incoming state is taken into account when recipe settings are chosen.

Two methods exist to account for the incoming state. One method involves measuring the incoming state of the lot in order to predict its impact on process performance. A typical example of such would be to measure the incoming film thickness to a Chemical Mechanical Planarization (CMP) process that requires a specified final film thickness. The CMP recipe may be adjusted according to the incoming thickness in order to compensate for variation from the deposition process that created the film. Such a method requires an accurate measurement of the incoming state, and an equation which incorporates this state into the control model. One additional constraint is that every lot must be pre-measured in order to compensate for incoming variation through feed-forward metrology.

These requirements run contrary to the nature of overlay. While any overlay measurement made at the previous masking operation could be construed as an feed-forward measurement, the nature of the measurement prevents this from providing useful information. Each metrology event is not a measurement of the position of a single pattern, but its relative position to another pattern. In order to be able to determine the location of a single pattern, one must be able to precisely define the location of the other pattern involved in the measurement. This information is not made available through conventional overlay metrology, so the use of such measurements as feed-forward information is infeasible. In addition to this, overlay metrology is quite time-consuming, making it expensive to measure each lot.

The second method available is to ensure that each lot to be run through a particular process has the same incoming state. With such a method, the actual state is less important than assuring that each lot within a group has the same state. By collecting such lots together, one can be confident that any lot within the group will have the same resultant process metrics. This grouping of like states enables one to determine the best recipe settings for the group based on initial data, and then apply those setting to the rest of the lots. Each group, or control thread, would be segregated from the rest of the line based upon the set of criteria that determines the incoming state.

2.1. Thread criteria

Control threads are defined by a set of criteria that determine the state of a process as well as the lot to be controlled. The ability to segregate a manufacturing line into control threads depends upon the ability to determine which process factors affect the state of the tool or incoming control state of a particular lot. For the most part, these factors will involve discrete context variables that apply to that lot. The particular entity used to process the lot, the tool type, the layer at which the lot will be run, and other such context variables are typical examples of such criteria. In addition, past context variables may also be applied in such cases where they make a contribution to the control state. In the CMP example discussed above, the tool that created the film to be polished will largely determine the pre-polish film thickness, so it may be useful to include the tool ID into the definition of the control thread.

The inherent danger involving the use of threads is the so-called "curse of dimensionality." Each criterion used to define a control thread will divide the line by the number of values that criteria can take. With each additional parameter in the thread definition, the control threads are again fragmented as many times as there are possible values of the new criterion. While it is true that the states within a given thread will have more similar values, there will be fewer lots assigned to any individual thread. Estimation of the control state of each thread would then be based on fewer data points, which would lead to degraded control. Care must be taken to not over-specify the control threads to the point where there is insufficient data to support them. A balance must be struck between the variation of the state within a thread and the number of threads created from the criteria.

2.2. Overlay threads

Definition of the thread criteria for any controller begins with finding those context variables that determine the state for the controlled process. A large number of variables may contribute to the variation within a process, but only a subset of these variables can be used within the thread definition. The goal of the threads is to capture most of the variation within the state, which can be accomplished by using those variables that have the greatest impact. Successful implementation of control threads will be achieved when lots processed in a short time period will have roughly the same state, such that the state trace a continuous trajectory when plotted over time.

The following examples of thread criteria are taken from actual process data from AMD's state-of-the-art Fab 25 production facility. The control state plotted in each of the figures is that of the model intercept *c* calculated from the overlay model for each lot within the trends (see Eq. 1). The data has been normalized by the maximum state value within the complete set of data.

2.2.1. Layer

The particular layer to be masked is of importance to overlay. While some of the particular overlay variables have little to no sensitivity to the layer ID, variables such as scale show a strong correlation. Figure 2 shows the average scale state over a one-month period for a number of layers within the manufacturing process. As evident within the figure, there is a strong correlation between the state and the position within the line, especially toward the back end of the process. One possible explanation of this result is the effect of exposure energy on thermal wafer expansion. Any thermal expansion effects exhibited at one layer must be repeated at the next such that the pattern is expanded in the same manner. The effect becomes cumulative, as each individual expansion of the layer adds to the total expansion from its original dimensions. This effect must be taken into account by the threads such that each lot within a control thread requires the same expansion correction.[4.]

The effect of layer on the scale state becomes apparent when the variation across the line is compared to that within a single layer. The error bars around each point represent the three-sigma variation in the state for that layer. While the variation in some layers in on the order of the entire line, most layers have much less variation. When added to the thread criteria, the layer ID will shrink the distribution from the complete range to those shown by the error bars, which can result in a dramatic reduction in the apparent noise and improvement in state estimation.

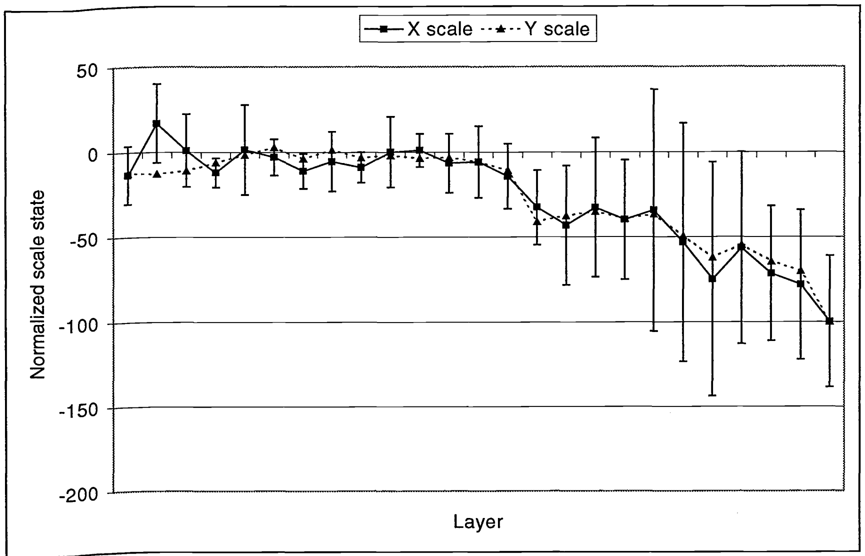

Figure 2: Correlation of scale state to process layer

2.2.2. Entity

The tool used to produce a particular pattern will also contribute to the overlay state of the masking process. Differences between tool types, as well as those between different tools within the same tool type, will contribute some amount of variation within the state. Aberrations within the stepper lens, physical differences between the stage alignment systems, and error or drift in tool matching will all affect the position of the intercept term of the overlay model. Such a tool dependency is illustrated in Figure 3, which shows two threads that differ only by the entity used to mask the lots. The translation overlay state is plotted as a function of time for both threads, showing a marked and sustained difference between the two threads.

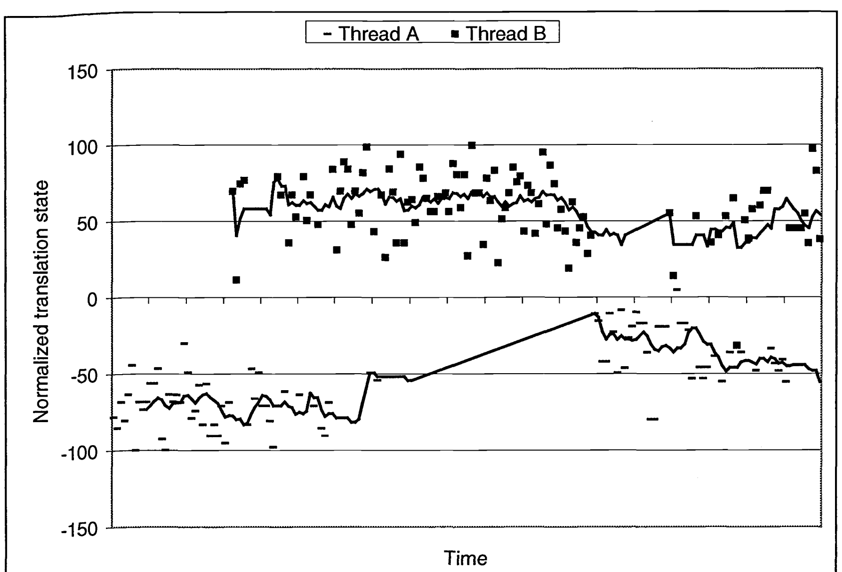

Figure 3: Dependence of translation state upon entity ID

The trends shown in the figure illustrate the utility of control threads. Combining the threads into a single set of data, the state could likely be found anywhere within the ±100% range of states shown in the figure. Split into two threads, however, the state of each thread varies over a much smaller range at any given point in time. The bimodal distribution of state values then takes on the form of a normal distribution about a single mean, allowing for much tighter control of the process line. Though the state drifts over time for each thread, the state remains relatively constant over a short time frame.

2.2.3. Previous entity

As the overlay measurement incorporates the properties of both the current and previous patterned layer, it is important to take into account the alignment properties of the tool used to mask the previous layer. The previous stepper will introduce some variation within the overlay state, in as much as any stepper will have a unique intercept to the overlay model. The same can be said for each stepper at each layer used to process a lot since it entered the manufacturing line, but it would be infeasible to include each of these entities into the thread definition. Without absolute dedication of each operation to a particular stepper, the line would be fragmented at each masking layer. Division of the line to such an extent would severely inhibit the effectiveness of the control thread methodology, as the threads would suffer from increased data poverty as additional steppers were added to the thread definition.

Figure 4 illustrates the correlation between the previous entity and magnification state. Note that the only thread criterion that differs between the two threads is previous entity, meaning that the difference in state is not due to the entity used to process the current layer.

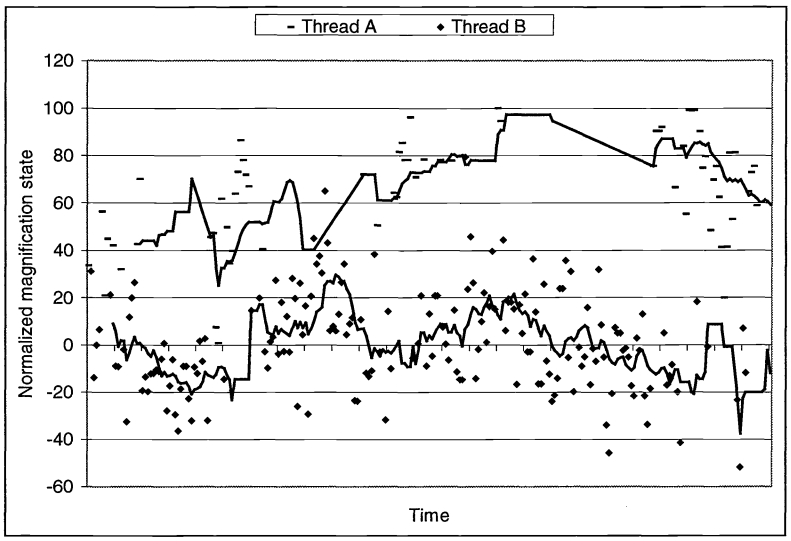

Figure 4: Dependence of magnification state upon previous entity

It is interesting to note that the two threads follow the same trend throughout the time frame shown. The drift within a single control thread is most likely due to variation in the stepper at the current layer, while the actual value of the control state is due to the incoming lot state as a function of the previous stepper.

3. FEEDBACK CONTROL

Once the thread criteria are defined and applied to the manufacturing line, the data should exhibit a relatively constant control state with a normal distribution. Such a trend suggests that the major sources of deterministic variation within the process have been removed. The limit of control thread utility would be complete elimination of deterministic variation, leaving only white noise inherent to the masking process. However, this solution alone does not complete the control methodology required to maintain the process centered on target. As evident in the examples of overlay threads shown in the previous section, even a properly specified control thread is still subject to drift. While the control state may be constant over a short time frame, the state will invariably change over a longer time period.

Additional controls must be put into place to monitor the process performance in order to ensure that the process remains on target. Statistical Process Control is common method that may be employed to ensure a thread remains in control. As is assumed in the application of SPC, the performance of an individual control thread should remain relatively constant over a short time frame. As the thread drifts away from the target, one or more or the SPC control rules would be violated. This would signal that the thread recipe requires an update, at which time a qualified individual could update the recipe to bring the thread back under control. While this is sufficient to ensure a thread remains in control, it is highly dependent upon engineering support.

Run-to-run control, also known as Advanced Process Control (APC), has recently become an attractive alternative to this more conventional control method. APC utilizes a feedback control model that relates the measured variables to the recipe parameters. The control model allows the necessary changes to the recipe to be calculated automatically based upon the historical performance of the thread. This allows the updates to become automated, such that the recipe settings may be adjusted by the controller to continuously control the process to target. Many algorithms have been proposed and implemented within the industry. One such algorithm put into practice at AMD is Linear Model Predictive Control.

3.1. Linear Model Predictive Control

Linear Model Predictive Control (LMPC) refers to control algorithms that use a linear process model and a linear or quadratic open-loop objective function with linear constraints to compute the requisite manipulated variable settings over a future time horizon.[5.] LMPC provides numerous advantages over competing control methods, primarily due to its flexibility. The heart of the LMPC is a linear state-space model of the process to be controlled,

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k \\ y_k &= Cx_k \end{aligned} \tag{4}$$

This set of linear equations relates the state variables x_k to the manipulated variables u_k and the controlled variables y_k. The matrices *A, B, C* are configurable model constants that define the relationship between the variables.

LMPC utilizes this state-space model to control the process using the objective function

$$\min_{u^N} \sum_{j=0}^{\infty} (y_{k+j}^T Q y_{k+j} + u_{k+j}^T R u_{k+j} + \Delta u_{k+j}^T S \Delta u_{k+j}) \tag{5}$$

The matrices Q, *R*, and *S* are penalties on the output, input and rate of change of the input vectors respectively. Each is symmetric, positive and semidefinite, although *R* must be positive definite to guarantee controller stability over an infinite control horizon. Standard quadratic program functions are used to calculate a new input vector that will minimize the objective function over the desired control horizon.

State estimation is usually performed with a optimal linear observer, such as the Kalman filter. The model prediction error is used to update the state as follows:

$$\hat{x}_{k+1|k} = A\hat{x}_{k|k-1} + Bu_k + L(y_k - C\hat{x}_{k|k-1}) \tag{6}$$

The filter predicts the state vector at time *k+1* using state, output, and input observations through time *k*. Methods exist to predict the optimal linear observer gain, *L*, from the process model with known noise disturbance terms. The value of the gain can also be chosen to produce the desired controller performance from trial and error if the character of the process disturbances are not known explicitly.

The formulation provides a comparative advantage over traditional run-to-run control methods. The Exponentially Weighted Moving Average (EWMA) controller is a widely used feedback control algorithm within semiconductor manufacturing, employing a deadbeat control law derived from an inversion of the plant model.[6,7.] The response of the controller is largely determined by the amount of noise inherent in the process, in that the filtering action of the EWMA controller will determine the control response. Those processes that have significant levels of noise will require a high degree of filtering, which in turn generates a sluggish response to disturbances. The control and filtering action are decoupled in LMPC with the addition of penalty terms on the input vector. In fact, the EWMA method is a subset of LMPC in which the penalty matrices *R* and *S* are set to zero, the horizon is set to one, and the controller is operated with no constraints. The flexibility afforded by the objective function allows better tuning of the process to provide a more robust control solution.

3.2. Run-to-run Linear Model Predictive Control of overlay

LMPC control of overlay can be accomplished by defining a control model based upon the model of the alignment process. Development of the state-space model for LMPC requires a modification of the standard linear model to include an intercept term, p_k, that takes into account the non-zero intercept term in the process model. This can be accomplished by using the output disturbance model form of LMPC. A vector p_k is added to the model as a constant disturbance term in the output, as shown in Equation 7.

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k \\ p_{k+1} &= p_k \\ y_k &= Cx_k + p_k \end{aligned} \tag{7}$$

The disturbance vector and the state vector can be combined to form an augmented state vector, which allows the output disturbance model to follow the form of a standard linear state-space model. The time indices k are assumed to refer to the succession of wafer lots passed through a particular masking operation.

$$\begin{aligned} \begin{bmatrix} x_{k+1} \\ p_{k+1} \end{bmatrix} &= \begin{bmatrix} A & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} x_k \\ p_k \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u_k \\ y_k &= \begin{bmatrix} C & I \end{bmatrix} \begin{bmatrix} x_k \\ p_k \end{bmatrix} \end{aligned} \tag{8}$$

The filter equation must also be updated to include the augmented state vector. The following is the form chosen for overlay control.

$$\begin{bmatrix} \hat{x}_{k+1|k} \\ \hat{p}_{k+1|k} \end{bmatrix} = \begin{bmatrix} A & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \hat{x}_{k|k-1} \\ \hat{p}_{k|k-1} \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u_k + \begin{bmatrix} 0 \\ L \end{bmatrix} (y_k - C\hat{x}_{k|k-1} - \hat{p}_{k|k-1}) \tag{9}$$

This form gives open-loop estimation of the original state vector and linear filtering of the disturbance vector. The number of states that can be estimated from a single metrology event is necessarily less than or equal to the number of measurements from that metrology. As there are an equal number of states and measurements, only one of the states can be estimated at a time. The state in the overlay model is assumed to have no physical significance, and takes the value of the previous input at all k indices. This is achieved by setting A to zero, which is appropriate for overlay control in that successive runs have no correlation. All other matrices in the model equal the identity matrix, except for the configurable Kalman filter gain, L.

The parameters within the objective function are chosen to achieve the desired response of the control system. The input penalty term, R, is excluded from the overlay objective as it has no relevance to control of this process. The inputs have neither cost nor bounds within the region of control required to bring overlay to target, leaving Q and S as the two configurable parameters within the equation. Since relative weight, rather than absolute value, will determine the performance, some flexibility in choosing the values of Q and S exists. In general the lower the ratio of Q to S, the slower the controller will respond to variation within the process. While a lower ratio means that the controller will respond slower to real variation within the process, it also means that it will respond less to noise in the control state.

4. APPLICATION AND RESULTS

The control methodologies discussed have been applied to overlay control at AMD in both the Fab 25 and the Fab 30 manufacturing facilities. This was achieved using the Catalyst® control framework product from KLA-Tencor Control

Solutions. Each of the exposure and metrology tools within the lithography module were integrated with this software package to create a basis for communication between each of the tools and the overlay run-to-run controller.

Implementation of the controller is facilitated through communication with the fabrication facility's automation systems each time a lot is run on a tool within the module. When a lot is first assigned to an exposure tool, a call is made to the Catalyst® control framework. Included in this call is all of the context and recipe information associated with the run. Using this information, the overlay controller assigns the lot to a control thread. State information is retrieved from the historical performance of the thread, and is then passed to the LMPC control algorithm. The algorithm then calculates the set of overlay control settings for the run, and passes them back to the exposure tool for inclusion into the process recipe. All pertinent information about the control run is then store to a database internal to the Catalyst® framework for later reference.

In much the same manner, a call is made to the APC system each time a lot is measured for overlay metrology. The desired overlay measurements are passed into the framework, along with the context and recipe information associated with the metrology run. Information from the masking operation is retrieved from the Catalyst® database and is paired with the overlay results to complete the information about the overlay state of the control run. This information is then used to update the control thread to which the lot was assigned.

The overlay control project at AMD has produced a significant increase in alignment performance. The process capability, C_{pk}, has steadily increased over the life of the project. The run-to-run controller has reduced the average standard deviation of maximum site overlay error by 40% over the life of its implementation, which has increased the Cp_k by that same amount. The rework rate due to misalignment has also been significantly reduced, with a rate at least 50% lower than the nominal rate before the controller was implemented. Lots that are misaligned to the extent that they are outside the specified control limits are stripped and reprocessed. In reducing the number of lots that are reworked, the controller has also increased overall tool availability and decreased the average cycle time within the lithography module.

Apart from the quantifiable improvements in control results, several ancillary benefits have come from the automation of overlay control. As the system is completely automated, there is no longer a need for engineering support of overlay on normal masking operation. This automated control frees a significant amount of engineering resources that can be reapplied to support other needs within the module. In addition to this, test wafer utilization for the qualification of overlay has been eliminated. Apart from the monetary benefit, this elimination of test wafer runs increases tool uptime and decreases the amount of involvement necessary from the manufacturing floor. All of these improvements are significant, in that they allow the module to focus on other issues that may require attention.

5. CONCLUSIONS

Run-to-run control is a very useful tool in the management of semiconductor manufacturing processes. Control threads have been shown to be a useful tool in improving the controllability of a process. By grouping lots with similar control states into separate control entities, the control thread methodology can significantly reduce the amount of apparent process noise to which the process is subject. Feedback controllers such as Linear Model Predictive Control, when used in conjunction with control threads, can automate the control of a process to continuously keep it on target.

Overlay control efforts at AMD have significantly benefited from the use of run-to-run control. Increased process capability and a decreased rework rate are two significant, quantifiable improvements to overlay control. In addition to these, automated control has significantly improved resource allocation within the lithography module. Engineering and floor resources have been freed from overlay maintenance, allowing them to be reapplied to other issues. Tool uptime and cycle time have also been improved through run-to-run control.

REFERENCES

1. Semiconductor Industry Association National Technology Roadmap for Semiconductors, 1998 Update.
2. H. J. Levinson, *Lithography Process Control*, p.87, SPIE Optical Engineering Press, Bellingham, WA, 1999
3. M. A. van den Brink, C. G. M. del Mol, and R. A. George, "Matching Performance for Multiple Wafer Steppers using an Advanced Metrology Procedure," *Integrated Circuit Metrology, Inspection and Process Control II*, K. M. Monahan, editor, SPIE, **921**, pp. 180 -197.

4. T. Saito, S. Sakamoto, K. Okuma, H. Fukumoto, and Y. Okuda, "Mask Overlay Scaling Error Caused by Exposure Energy Using a Stepper," *Integrated Circuit Metrology, Inspection, and Process Control VII*, M. T. Postek, editor, SPIE, **1926**, pp. 440 - 449.
5. K. R. Muske and J. B. Rawlings, "Model Predictive Control with Linear Models," *AIChE Journal*, **39(2)**, February 1993, pp. 262 - 287.
6. T. F. Edgar, W. J. Campbell, and C. A. Bode, "Model-Based Control in Microelectronics Manufacturing," *Proceedings of IEEE Conference on Decision and Control*, p. 4185, Phoenix, AZ, December 1999.
7. E. D. Castillo and A. M. Hurwitz, "Run-to-Run Process Control: Literature Review and Extensions," *J. Qual. Tech.*, **29(2)**, April 1997, pp. 184 - 196.