

How Mask Data Error Rate Maintained Below 0.1 Percent While Volume Increased 2+ Fold – Through Automation

Susie Ross, Tin Ko, Jianli Fu, Mats Fredriksson, John Qiang Li, Hongxiao Shao

Skyworks Solutions, Inc.,
2427 W. Hillcrest Drive,
Newbury Park, CA 91320,
hongxiao.shao@skyworksinc.com

Keywords: Mask, Reticle, Automation, Reticle, EDA

Abstract

Reticle assemblies and release is one of the last steps in IC design data release to a mask vendor. In addition to IC data, the reticle contains PCM patterns, alignment markers, CD, etc. After a reticle design is completed, the correction factors (c-factor) for respective layers need to be applied to the data for the layers before they can be fractured for mask creation. A reticle design is about aggregation of design and foundry data and output masks, as well as the data necessary for foundry hardware tool operation. This paper describes specific automation solutions in two steps in the reticle design process, alignment marker data generation, and final reticle approval process. Automation, including the solutions discussed in this paper, enabled data release with 2+ fold volume increase while maintaining the error rate at <0.1 percent.

INTRODUCTION

Wafers are processed based on masks. Masks are created based on wafer designs with arrayed reticles, with layer specific correction factors (c-factor). A reticle is the basic unit containing IC data, process control monitor (PCM) structures, critical dimension (CD) structures, wafer level and layer to layer relative alignment markers, mask identification labels, etc.

There are many different approaches companies choose to handle specific steps during the process. For example, a wafer design and a reticle design with array instructions are equivalent in terms of mask making. Data to mask vendors can be conducted in the form of fractured data or raw gds data with c-factors provided. IC data can undergo optical correction for nano-scale process nodes (usually not necessary for GaAs processes).

In today's business environment, efficiency and cost are two of the most important considerations in daily operations and strategic planning. The processes of interest in this paper are GaAs based. The mask cost is <10% of that of an advanced deep-submicron technology node. From a total efficiency and cost stand point, the increased reticle throughput without an increase of error rate is the most effective way of achieving efficiency and cost goal.

Figure 1 shows the normalized mask set released for an extended period, which shows a multiple fold volume

increase. This paper describes two of the automation solutions deployed that enabled us to support such a dramatic volume increase without the increase of the error rate and with a reduced human intervention. These two solutions are 1). alignment mark data from reticle design to hardware tools automation, and 2). reticle approval automation.

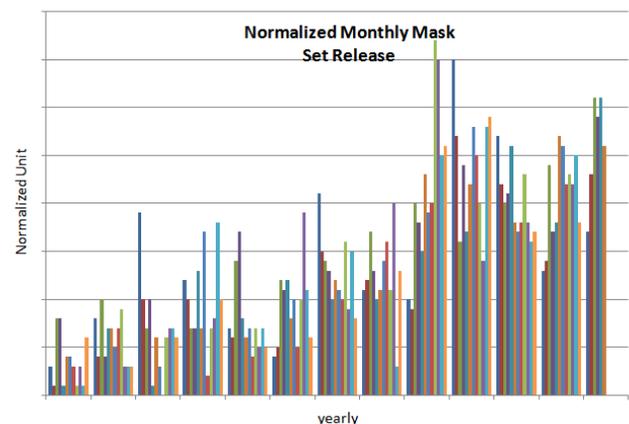


Figure 1. Normalized Mask Set Release for an Extended Period

GENERAL MASK GENERATION PROCESS DESCRIPTION

Figure 2 below shows the general flow for a mask generation from requesting to release.

In a mature product company, an engineering run typically starts with an engineering requirement with a selected technology. Then it goes through a request system, which would generate further requests for masks funding requisition, IC design workspace creation, PCM and marker cell identification, etc.

When funding is secured and IC data ready, a reticle is designed according to specified IC quantities and process technology specific PCM and marker cells. The reticle is ready to be reviewed by all stake-holders. Once approved, the reticle data can either be fractured according to the c-factor and the fractured data sent to the mask-making vendor, or the reticle/wafer data itself, along with c-factor information, is sent to the vendor, for making the mask plates.

Along with the reticle design release, the data required by hardware tools are generated.

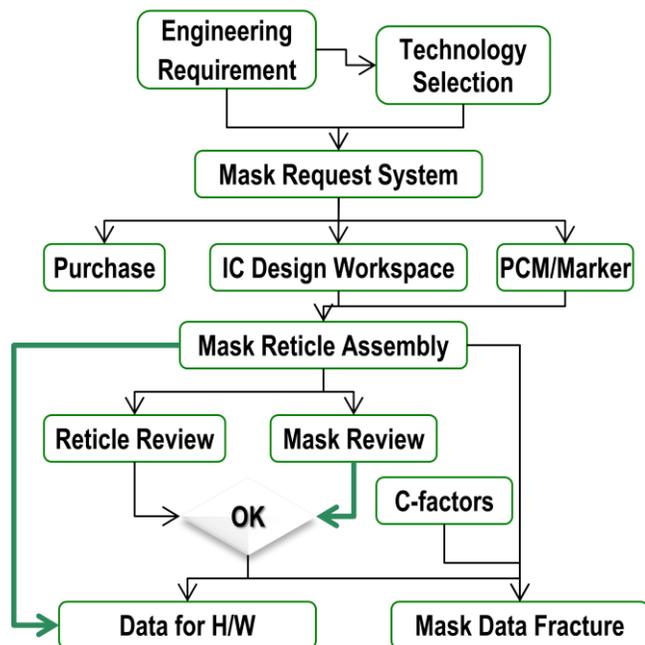


Figure 2. Mask Generation Process

PROBLEM DESCRIPTION – RETICLE REVIEW

The reticle data typically resides in an engineering computing system (Linux/Unix) created by commercial electronic design automation (EDA) tools. The process technology specific information on which the reticle design is based usually sits somewhere other than the engineering computing system (a PC-based solution, e.g.). The reticle design specification, such as IC quantities, is normally created with a PC application as well.

As a result, a review process involves going through multiple computer systems (Linux/Unix and PC), many different database solutions (EDA, Notes, SAP, etc.), and many different interfaces (WEB, Microsoft Office applications, etc.) It was time consuming, error-prone, and inconvenient.

Multiple systems, multiple databases, and multiple interface-based review processes was partly responsible for the low throughput suffered we previously suffered.

PROBLEM DESCRIPTION – H/W REQUIRED INFORMATION

Information for mask placement, mask alignment with respect to fixed objects (absolute), and against geometries on prior layers (relative) are contained in and defined by the alignment marker cells placed in a reticle design according to specifications. That information is required to be read out from the reticle design and be entered into the appropriate hardware tools. The information of pattern density of a given layer is also required to adjust the light exposure during the photo steps.

A traditional way of readout is manual with brute force, goes through the reticle design hierarchy in an EDA solution, reads the coordinates of the mark cells, records

them in Microsoft Word files, and emails them to the hardware engineers, who then make manual entry of the coordinates into the hardware tools.

The previous approach of getting the pattern density for a given layer is to record density during data-fracture, with the c-factor, in a Microsoft Word document, and email it to the fab engineer who then enters the data into the machine by hand.

The manual data readout is time-consuming (many 10s of minutes), and error-prone during the entire data generation-to-data entry process. The worst part is that the review and approval process, which includes the verification of the readout data, requires the repeat of the same process as before -- a complete waste of effort.

SOLUTION – RETICLE REVIEW

The fundamental problem for reticle review deficiency was due to multi-system, multi-database, multi-interface, etc. The solution is to eliminate the multiple sources, at least at the user level.

Figure 3 below describes a philosophy which enables an efficient and effective review.

1. Make all PC-based data available on the engineering system, the same as the one reticle designs locate;
2. Create software tools on Linux to convert all reference data in diverse databases to a common data format;
3. On-demand query of the reference data in common data format based on the review task assigned;
4. Review and compare.

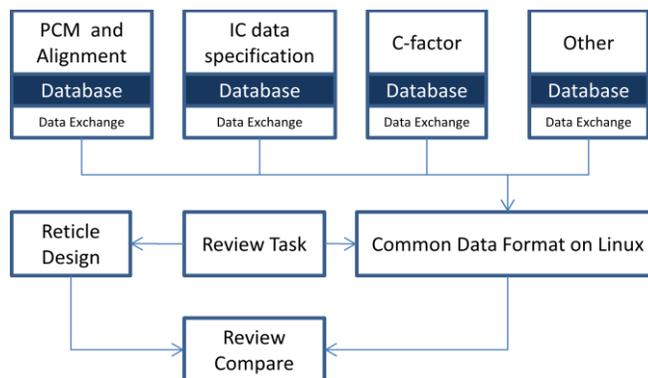


Figure 3. An Efficient Mask Review Approach

The approach described here eliminates about 90 percent of review choir – going through diverse systems, databases, interfaces, etc. – of getting the reference information required for the reticle review. The review of a reticle is still interactive and Web-based approval/rejection is manual.

There are values in interactively reviewing the reticle design for various purposes. We choose to keep the review interactive.

SOLUTION – H/W REQUIRED INFORMATION

The manual approach for obtaining hardware required information is necessarily time-consuming and error-prone. The worst part is lack of systematical data verification.

We notice that any reasonably mature electronic design automation (EDA) solution has its database query functions that can be leveraged to obtain defining information for all instances/geometries along reticle design hierarchy. Besides, almost all modern hardware tool supports automation with its firmware API. Figure 4 illustrates the basic approach for a reticle design and the hardware interface solution.

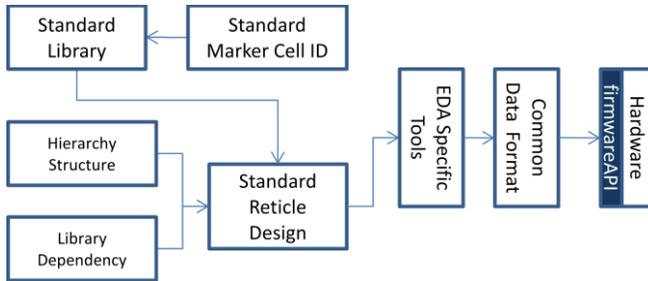


Figure 4. A Reticle Design and Hardware Interface Solution

The reticle design to hardware interface solution consists of two parts, the front side of obtaining data from a reticle design in a chosen EDA tool and converting the data into a common data format, and the back end of moving the data in common data format to a given hardware tool through its firmware application program interface (API).

The EDA tool specific query functions are necessary but not sufficient to acquire marker cell information with a viable and systematic solution. The standardization of the marker cells and the libraries that contain the marker cells, and the standardization of the reticle design hierarchy and the libraries that contain the blocks used in the reticle design hierarchy, allow the use of the query functions to systematically query the information of the specific named geometries in a cell and instance information of a specific cell in a specific library. The EDA tool query API and standardization of marker cell libraries and reticle design hierarchy make the front-side solution a reality.

Figure 4 covers the acquisition of the marker information. Figure 5 describes the means to obtain the pattern density for each masking layer.



Figure 5 Flow for Pattern Density for Each Masking Layer

Correction factors are the values to grow or shrink the geometries in a given masking plate so the geometries on the wafer will be the same as drawn in an EDA layout tool. Thus the pattern density important for photo exposure is the one on the mask plate, not the ones drawn in a reticle design. C-factor needs to be applied in pattern-density calculation. This

can be accomplished by sizing the geometries in the layout within an EDA layout tool or during design rule check.

The backend solution of programmably uploading data from a reticle design in common data format to a hardware tool is a computer-science exercise. The real challenge is that the H/W firmware is normally implemented on the PC, making it difficult to implement a complete solution on Linux, which is the preference of the authors.

The reticle design to hardware automation solution enabled the reduction of the entire process from many 10 minutes to one single button click. It also facilitated the validation of the data in a systematic way, making it part of our standard review process.

SUMMARY

We presented approaches to automate the reticle review process and reticle design to hardware interface solutions. These two solutions, along with many other similar ones, are part of our design automation and product life cycle (PLM) automation effort. Automation does not just make product generation process efficient, it also makes the change control and tracking possible and effective. It has been the automation that enabled us to maintain and reduce mask error rate with 2+ fold volume increase with same amount of resources.

