Towards a Risk-Assessment Based Approach for Refactoring of Legacy Components

Introduction

ASML is the world's leading provider of lithography systems for the semiconductor industry, manufacturing complex machines that are critical to the production of integrated circuits or chips.

Figure 1 Subsystems of the EUV Lithography Machine

The Production Control software cluster contains the high level supervisory controllers of the lithography system. Production Control is responsible for the high-level coordination of the subsystems that are realizing, among others, the material flow (wafer and reticle flow), and measurement / exposure processes of the lithography system.

Figure 2 Production Control supervisory controllers

Problem Description

Over the years the Production Control codebase has evolved to support new generations of lithography systems. For some software components, the continuous modifications have led to a suboptimal, hard-to-maintain implementation. In addition, the update of requirements and design documentation is often behind product development and the regression test set is often incomplete. This has led to so-called legacy components that are difficult to maintain and extend. Any change to such a component brings with it considerable risk of introducing undesired side effects. In order to support the next generation of lithography systems, critical legacy components undergo a process called refactoring. During the refactoring process, a new component is developed that is expected to behave the same as the legacy
component while reducing maintenance effort / risk at the same time. However, refactoring in itself poses significant risks as implicit requirements are hidden in the implementation of the legacy component.

Assignment

The objective of this assignment is to find ways of assessing the risks of refactoring by performing some form of equivalence checking between the legacy and refactored versions of a component, i.e. checking that the external observable behavior is identical. In case the behavior is not identical a counterexample is generated. Such counterexample should be presented in a form that helps the engineer to understand the difference in behavior. Based on this information the engineer can modify the legacy or refactored component and initiate a new equivalence check. This process enables the identification of hidden requirements during development and therefore reduces the risk of issues at the customer.

The large and complex state behavior of the Production Control components poses a challenge as full coverage of the behavior might not be possible. An important part of the assignment is to propose metrics, a sort of “certainty” factor, that indicate to what extent the legacy and refactored components are identical. Existing (partial) regression tests, code coverage measurements run in parallel with model learning could be used to determine such a factor.

The refactored component will be implemented using the ASD:Suite from Verum. As a result, the refactored component will consist out of an interface and implementation model. Formal verification is used to ensure that the implementing model is correct and complete with respect to the interface model. As a result, the interface model contains all the external observable behavior of the refactored component.

Case Description

Within the Production Control cluster there are several components, varying in size and complexity, which need to be refactored. One of these components is responsible for coordinating the position exchange of chucks (usually called chuck exchange or chuck swap) during wafer processing. Chucks are used to hold the wafer in place while the wafer is being processed. Within lithography systems there are two basic processing steps. First the wafer is measured after which the measurement data is used to expose the wafer. To increase throughput, Twinscan lithography systems use two chucks that work in parallel to measure and expose wafers at the same time, i.e. while one chuck is at the measure position the other chuck is at the expose position.

The ChuckSwap component is responsible for the coordination of actions that move the chuck at the measure position to the expose position and at the same time move the chuck at the expose position to the measure position. This software component is critical as it connects the two basic processing steps. A failure in this component will result in considerable loss in throughput or in the worst case, will prevent the machine from processing any wafers.

Refactoring of the ChuckSwap component is foreseen to improve its maintainability and extensibility. To reduce the risk of such a refactoring step we want to verify that the external observable behavior did not change after refactoring.

Provided by ASML

- Legacy component
• Refactored version of the legacy component
• Help/support from a domain expert on component requirements and expected component behavior
• Workspace at ASML Veldhoven

**Expected deliverables:**
• Models of the legacy and refactored component
• The ability to perform equivalence checking on these models
• Process: Guidelines/way of working/method for other legacy components requiring refactoring
• Master Thesis