Bridging Formal Models
An Engineering Perspective

Software engineers create informal models that describe the cooperation of hardware and software components. These models make system design more manageable and are used to generate code, but do not guarantee the absence of errors.

To obtain error free designs, engineers can create formal models that provide ways to mathematically show the absence of unintentional behavior. Creating these models is an arduous task. This thesis shows how these models can be derived from informal specifications, can be extracted from software code, and can be obtained from other specification languages. Additionally, this thesis demonstrates how to communicate verification results to engineers.

Because these models are constructed by hand they are likely to contain delicate flaws. This thesis describes a mathematical approach to prevent these flaws. The approach constructs formal models that use the operational semantics of the input language. We clarify the required steps, and demonstrate its applicability by showing the subtle differences found in a specification language.