Zusammenfassung

Sicherheitskritische elektronische Systeme werden immer komplexer und gleichzeitig immer allgegenwärtiger. Wie andere Ingenieurwissenschaften muss auch die Informatik gangbare Herangehensweisen anbieten, um den korrekten Entwurf solcher Systeme zu gewährleisten. Besonders wichtig innerhalb des Entwurfsprozesses ist die Phase der initialen *Spezifikation*, da ihre Ergebnisse Auswirkungen auf alle nachfolgenden Entwicklungsschritte haben. Die umfassendste und zuverlässigste Analyse von Systemspezifikationen kann durch die Verwendung formaler Methoden erreicht werden, die es durch den Einsatz automatischer Verifikationstechniken ermöglichen, einen mathematischen Beweis der Systemkorrektheit zu erhalten.

Im Bereich der formalen Spezifikation gibt es jedoch keine einzelne universell einsetzbare Notation, die gleichermaßen geeignet für alle Aspekte von Systemen wäre. Stattdessen werden *integrierte formale Methoden* erforscht, die unterschiedliche Spezifikationssprachen kombinieren, um ihre individuellen Stärken auszunutzen und gleichzeitig eine gemeinsame semantische Basis für die anschließende Verifikation aufrechtzuerhalten. Eine solche Notation ist die Spezifikationssprache *CSP-OZ-DC*, in der die Prozessalgebra *Communicating Sequential Processes* (CSP) zur Beschreibung von Verhaltensaspekten, die zustandsbasierte Notation *Object-Z* (OZ) zur Beschreibung von Datenaspekten und die Realzeit-Logik *Duration Calculus* (DC) zur Beschreibung von Realzeitaspekten von Systemen vereinigt sind.

Das hauptsächliche Hindernis für die erfolgreiche Anwendung automatischer Verifikationsmethoden ist jedoch das Problem der *Explosion des Zustandsraums*, also der exponentiellen Vergrößerung der Anzahl der zu analysierenden Systemzustände. Zahlreiche Techniken zur Bewältigung dieses Problems wurden bereits vorgeschlagen, unter ihnen die des *Slicing*, das seinen Ursprung im Gebiet der Programmanalyse hat, wo es zur Berechnung derjenigen Programmteile verwendet wird, die im Hinblick auf eine bestimmte Fragestellung relevant sind.

In der vorliegenden Dissertation wird eine Herangehensweise zum Slicing integrierter formaler Spezifikationen entwickelt, die maßgeschneidert für die reichhaltige syntaktische Struktur von CSP-OZ-DC-Spezifikationen ist, und die gleichzeitig im Rahmen ihrer Verifikation bezüglich Realzeitanforderungen einsetzbar ist. Der Slicing-Ansatz besteht im Wesentlichen aus drei Schritten: Erstens wird die Spezifikation im Hinblick auf verschiedene Typen von Abhängigkeiten zwischen ihren syntaktischen Elementen analysiert, wobei mehrere neue Abhängigkeitstypen wie Synchronisations- und Zeitabhängigkeit definiert werden. Zweitens wird die Gesamtheit dieser Abhängigkeiten genutzt, um diejenigen Teile der Spezifikation zu identifizieren, die relevant für die gegebene Verifikationseigenschaft sind. Drittens wird der Slice der Spezifikation berechnet, also eine reduzierte Version der vollen Spezifikation, in der alle Elemente ohne Einfluss auf die Verifikationseigenschaft entfernt sind.

Ein *Korrektheitsbeweis* zeigt, dass anstelle der ursprünglichen Spezifikation nun der Slice für eine Verifikation verwendet werden kann, ohne das Verifikationsergebnis zu verändern. Der Beweis basiert auf dem Begriff der *Projektion* zwischen einer Spezifikation und ihrem Slice. Es wird gezeigt, dass der entwickelte Slicing-Ansatz die Existenz einer solchen Projektionsbeziehung garantiert. Darauf aufbauend wird gezeigt, dass die jeweilige Logik zur Beschreibung von Verifikationseigenschaften stotter-invariant ist, das heißt, unter der Voraussetzung der Existenz einer Projektionsrelation kann sie nicht zwischen Slice und ursprünglicher Spezifikation unterscheiden, sodass das Verifikationsergebnis in beiden Fällen das gleiche sein wird.

Schließlich wird die Werkzeugunterstützung vorgestellt, die zur Entwicklung, zum Slicing und zur Verifikation von CSP-OZ-DC-Spezifikationen implementiert wurde, ergänzt durch mehrere Fallstudien und experimentelle Ergebnisse zur Evaluierung der Effektivität des Slicing-Ansatzes.