

# Entwicklung einer Entwurfsumgebung für parallele datenverarbeitende Hardwarearchitekturen

Jones Yudi Mori Alves da Silva

## Kurzfassung

Smart Cameras sind Kameras welche ein Bild nicht nur aufnehmen, sondern auch komplexe Bildverarbeitung (Image Processing - IP) und Computer-Vision-Anwendungen (CV) auf diesem ausführen. Diese Art von Anwendungen unterliegen grötenteils Echtzeitanforderungen. Echtzeitanforderungen implizieren, dass ein vorgeschriebenes Zeitverhalten eingehalten werden muss. Ist dies nicht der Fall, wird dies als Fehler bewertet. Darüber hinaus werden in IP/CV Anwendungen große Datenmengen in vielen einzelnen Optionen verarbeitet. Die daraus resultierenden Anforderungen sind bis heute schwer in eingebetteten Systemen zu realisieren und weitem die typischen Anforderungen an eingebettete Systeme, wie beispielsweise geringe Herstellungskosten und Leistungsaufnahme, Temperaturmanagement sowie hohe Fehlertoleranz. Seit einigen Jahren gibt es im Bereich der Computer-Architekturen einen Trend hin zu Mehrkernsystemen. Die VLSI-Technologie erlaubt bereits heute heterogene und mehrkernige System-on-Chips (SoCs), wodurch Anwendungen von spezialisierten und effizienteren Verarbeitungsressourcen profitieren können. In der Domäne der Smart Cameras ist eine der größten Herausforderungen / eines der größten Probleme, der Datentransfer vom Pixel-Sensor zur Datenverarbeitungseinheit. Aus diesem Grund existieren verschiedene Arbeiten welche das Konzept der sensornahen Bildverarbeitung untersuchen. Sensornahe Bildverarbeitung zielt darauf ab, die Datenaufnahme und Datenverarbeitung so nah wie möglich zusammenzubringen, und so die Effizienz der Datenübertragung zu steigern. Eine Erweiterung dieses Konzepts ist unter dem Namen Focal-Plane Image Processing (FPIP) bekannt. FPIP nutzt die parallele Datenaufnahme und den Transport zu mehreren Datenverarbeitungseinheiten. Dieses Konzept ermöglicht die Untersuchung hochgradiger Parallelisierung und dadurch eine Steigerung der System-Effizienz. In dieser Arbeit stellen wir unsere Analyse von zukünftigen eingebetteten Many-Core Systemen für die parallele Datenverarbeitung mit Fokus auf IP- und CV-Anwendungen vor. Die Arbeit kann in drei Hauptteile gegliedert werden, welche sich hinsichtlich der Ansätze und Abstraktionsschicht unterscheiden. Der erste vorgestellte Ansatz verfügt über einen hohen Abstraktionsgrad aus der Sicht eines Anwendungsentwicklers. Auf dieser Abstraktionsschicht definieren wir die Programmiersicht. Diese beinhaltet, wie Anwendungen entwickelt und analysiert werden, um die benötigten Informationen für die Entwicklung einer effizienten Systemarchitektur zu ermöglichen. Hierzu wurde eine Methodik für das effiziente Profiling von Anwendungen und der effizienten Exploration der Systemarchitektur entwickelt, welche auf Task-Graphen, High-Level-Synthese und Graph-Clustering basiert. Für die Umsetzung dieser Methodik wurde ein Task-Graph-Simulator auf Basis des SystemC/Transaction-Level-Modeling (TLM) implementiert. Die im Rahmen dieser Arbeit entstandenen Ergebnisse ermöglichen eine Reduktion des Entwicklungsraums. Die übrigen Implementierungsansätze wurden im Folgenden auf einem tieferen Abstraktionslevel untersucht. Der zweite vorgestellte Ansatz untersucht die verbleibende Menge der Implementierungsansätze im Entwicklungsraum auf einer tieferen Abstraktionsschicht. Dazu wurde ein weiterer SystemC/TLM Simulator für die Kommunikation zwischen den Datenverarbeitungseinheiten entwickelt. Der Simulator wurde in ein Werkzeug integriert, welches die Abschätzung der Leistungsaufnahme, Chip-Fläche sowie des zeitlichen Verhaltens ermöglicht. Die in diesem Abschnitt erstellten Ergebnisse liefern detailliertere Informationen über die Leistungsfähigkeit der untersuchten Architekturen. Der dritte vorgestellte Ansatz führt die Analyse auf einer weiteren, tieferen Abstraktionsschicht weiter. Dieser Ansatz basiert auf einer Modellierung auf dem Register-Transfer-Level (RTL) mit einer zyklusakkuraten Simulation in VHDL sowie Syntheseergebnissen der Systemarchitekturen. Darauf aufbauend wurde eine Basisarchitektur definiert und untersucht, deren Effizienz mit dem Stand der Technik verglichen wurde. Mehrere neuartige Konzepte werden vorgestellt und diskutiert. Unter Verwendung der gezogenen Schlüsse wurde die Basisarchitektur verbessert und verschiedene Architekturkonzepte mit anwendungsspezifischen Optimierungen entwickelt.