

Schnell und flexibel

Field Programmable Gate Arrays (FPGAs) erobern die industrielle Bildverarbeitung

Field Programmable Gate Arrays sind seit langem unverzichtbare Werkzeuge im Umfeld der Electronic Design Automation (EDA). Sie werden meistens zum Rapid Prototyping, der Entwicklung, Emulation und Verifikation von komplexen digitalen Systemen herangezogen. Die aufgrund der speziellen Systemarchitektur hochparallele Arbeitsweise dieser Chips prädestiniert sie geradezu für solche Aufgaben.

Für die meisten Aufgabenstellungen in der industriellen Bildverarbeitung werden ebenfalls parallele statt sequenzielle Prozesse benötigt; nichts lag deshalb näher, als die Rechenleistung der FPGAs auch in typischen Machine Vision Applikationen einzusetzen. Bislang aber ist die hardwarenahe Programmierung ein Stolperstein für eine größere Akzeptanz dieser Technologie. Silicon Software, Pionier FPGA-basierter Framegrabber, präsentiert mit einer C++-basierten, objektorientierten Hochsprache für FPGA Hardware sowie mit dem graphischen Werkzeug VisualApplets Studio ein komplettes Entwicklungspaket, mit dem Bildvorverarbeitung ohne aufwendige Programmierung möglich ist.

What can I do with my frame grabber's IQ ?

Seit ein paar Jahren kursiert der Begriff IIPB („intelligent image processing board“) als Abgrenzung zu den einfachen Framegrabbern in der Machine Vision Branche. Die Intelligenz bezieht sich auf das Vorhandensein eines Vision Prozessors, der zusätzliche integrierte Bildvorverarbeitungsfunktionen ermöglicht, die für Projekte eingesetzt werden können, meist aber ungenutzt brachliegen. Weitergehende Bildverarbeitungen werden in der Regel auf Software ausgelagert. Bei steigenden Anforderungen ist allerdings die Software, trotz der Leistungsfähigkeit moderner PCs, auf eine Vorverarbeitung angewiesen, um eine Echtzeitfähigkeit gewährleisten zu können. Was liegt näher, als sich die vorhandene Rechenkapazität auf dem Framegrabber als „customizable intelligence“ zunutze zu machen.

Auf aktuellen Framegrabbern konkurrieren CPUs, Multimedia-Chips, DSPs und FPGAs um die Gunst der Käufer. Jede Prozessorarchitektur hat ihre spezifischen Vor- und Nachteile und somit auch prädestinierte Einsatzmöglichkeiten. Voraussetzung ist die Möglichkeit, überhaupt auf den Prozessor zugreifen zu können. Dieses liegt in der jeweiligen Produktpolitik, ob die Funktionalität nur durch den Hersteller selbst oder aber auch durch den Kunden erweitert werden kann. Eine weitere Hürde kann die Auseinandersetzung mit einer hardwarenahen Programmierung sein. Oft fehlt in Projekten die notwendige Einarbeitungszeit oder erfahrene Hardwareprogrammierer.

Eine revolutionäre Art der Verbindung zwischen Hard- und Software

Zwei wichtige Eigenschaften liegen den FPGAs zugrunde. Ihre Rechenleistung ist auf die Verarbeitung von großen Datenmengen, die parallel anliegen ausgerichtet und ihre Hardwarestruktur ist flexibel. Dieses bedeutet, dass die Prozessorfunktionalität durch Software veränderbar ist. Durch diese moderne Technologie der reprogrammierbaren FPGA-Prozessoren lassen sich hohe Verarbeitungsgeschwindigkeiten von Hardware mit der flexiblen Anpassung durch Software kombinieren. Für den Kunden als auch für den Hersteller ergeben sich dadurch gleichermaßen signifikante zeitliche und finanzielle Vorteile. Änderungen der Hardware können in Stunden statt in Monaten umgesetzt werden und individuelle Anpassungen der Hardwarelösung können auch noch nachträglich implementiert werden.

microEnable – ein reprogrammierbarer Framegrabber

Das Konzept der microEnable Produktlinie beruht auf der Kombination einer zeitgemäßen Hardwarebasis zwischen Kamera und Computer mit einem leistungsfähigen und flexiblen FPGA-Prozessor. Zwei CameraLink Kameras bis 85 MHz können angeschlossen werden, 96 MByte dienen als on-board Zwischenspeicher und ein 64bit/66 MHz PCI Interface als Schnittstellen zum Host-PC. Über ein graphisches Auswahlmenü wird der Framegrabber initialisiert. Durch das Laden eines Hardware-App-



Abb. 1: Xilinx Spartan-IIe FPGA, Quelle Xilinx



Abb. 2: microEnable-III Framegrabber



Abb. 3: VisualApplets Studio Oberfläche



Abb. 4: Kundenspezifische Applets unter microDisplay

lets wird der FPGA in Sekundenbruchteilen umprogrammiert und erhält die optimierte Hardwarestruktur für die Aufgaben eines Zeilen- oder Flächenkamera-Framegrabbers. Spezielle Auslesereihenfolgen der Kamerasensoren können ebenso berücksichtigt werden, wie auch zusätzliche Shading-Korrekturen oder Bayerfilter-Konvertierungen in Echtzeit. Abhängig von der Leistungsfähigkeit des verwendeten FPGAs sind weitere, zusätzliche Bildvorverarbeitungen oder die Konfiguration individueller Sensoren möglich.

Drag and Drop – mit Graphik wird's noch einfacher

Oft sind es nur kleinere Änderungen an der Framegrabber-Funktionalität, die einen entscheidenden Projektvorteil bringen können. Sei es durch die Implementierung zusätzlicher Filter oder die Möglichkeit den Systemaufbau nachträglich zu ändern. Dabei soll jedoch keine zeit- und kostenaufwendige Abhängigkeit vom Hersteller oder einem FPGA-Programmierservice entstehen. Aus diesen Gründen wurde das VisualApplets Studio entwickelt. VisualApplets Studio ist ein graphisch zu bedienendes Programm mit intuitiver Bedienoberfläche. Es ermöglicht einem Laien Anwendungen für microEnable nach einem Baukastenprinzip zusammen zu

stellen, ohne über hardwarenahes FPGA-Know-How verfügen zu müssen. In VisualApplets Studio werden vorgefertigte Bildverarbeitungsmodulare zu Verfügung gestellt, die in Minutenschnelle zu fertigen Applikationen zusammengestellt werden können. Für Anwender, die spezielle Funktionsmodule benötigen, kann VisualApplets Studio durch individuelle Module erweitert werden, die durch eine C++ basierte Hardwaresprache erstellt werden. VisualApplets Studio kontrolliert dabei die verfügbaren Hardware-Ressourcen und den Datenfluss mit möglichen Nebenläufigkeiten.

Das Programm gibt als Ergebnis ein Hardware-Applet zur Einbindung über das SDK und zur Verwendung in dem graphischen Konfigurationsprogramm microDisplay aus.

Die „klassische“ Art der Programmierung von FPGAs

Häufigste verwendete Basis zur Programmierung der FPGA Hardware ist VHDL. Das Synonym VHDL steht für VHSIC Hardware Definition Language (VHSIC = Very High Speed Integrated Circuit). Mit dieser Hardware-Beschreibungssprache wird ein funktionales Software-Modell erstellt. Eine Schwierigkeit für Software-Entwickler, die eine sequenzielle Programmierung mit Unterfunktionen gewohnt sind, ist das Umdenken bei der Formulierung eines Algorithmus. Alle Funktionen, die ein Algorithmus beinhaltet sind in der Regel gleichzeitig implementiert und werden über einen Datenfluss nur gesteuert. Bei Nichtbeachten dieser Problematik kann es zu Nebenläufigkeit der Hardwarefunktionen bei der Analyse des Codeverhaltens kommen. Ein großes Akzeptanzproblem ist somit nicht die Einarbeitung in die Syntax von VHDL, sondern vielmehr die veränderte Herangehensweise und Denkweise der Codeerzeugung und Algorithmusformulierung.

Ein zweites Akzeptanzproblem ist das, aus der Sichtweise eines Softwareprogrammierers zu niedrige Abstraktionsniveau von VHDL. Beispiele hierfür sind das Problem langer Codierungszeiten, eine fehlende Seiteneffektfreiheit, die langwierige Fehlersuche, die fehlende Hardwareunabhängigkeit auf Prozessor- und Systemebene sowie die schlechte Wiederverwertbarkeit von Programmiercode.

Das Ausgangsproblem, einen Prozessor mit immenser Rechenkapazität aber veralteten Softwarewerkzeugen programmieren zu müssen, war der Anlass für Silicon Software, eine objektorientierte Hochsprache auf C++-Basis

zu realisieren mit dem Ziel, der FPGA-Technologie eine substanziell größere Verbreitung im Bildverarbeitungsumfeld zu sichern.

Das Programmierwerkzeug wird intern bei Silicon Software seit 5 Jahren zur Projekt- und Applikationsentwicklung eingesetzt.

C++-Objektorientierung auch für Gate Arrays

Das Coding-Werkzeug ist ein moderner Programmieransatz, der durch die Verwendung der Sprache C++ sowohl für Hard- als auch für Softwareprogrammierer einfach anzuwenden ist. Ein einfaches und intuitives Bibliothekskonzept bildet die Basis, das lediglich einen C++-Compiler und die Place&Route Werkzeuge des FPGA-Herstellers voraussetzt.

Durch die Verwendung der Standardprogrammiersprache C++ zur Codeentwicklung, ergeben gegenüber VHDL sich eine Reihe von Vorteilen. Hardwaremodule können überladen werden und so an die Bedürfnisse des Anwenders ideal angepasst werden. Beliebige Abstraktionslevels können realisiert werden, indem verschiedene Ablaufmodelle implementiert werden. Simulation und Synthese sind aus einem Guss, so dass keine separaten Werkzeuge verwendet werden müssen.

Wichtige Kriterien bei dem Entwurf des Programmierwerkzeuges war die Auslegung auf höchste Verarbeitungsperformance der Hardware und extrem kurze Design-Cyclen der Software, die Möglichkeit das Programmierwerkzeug an beliebige Hardwareumgebungen anpassen zu können und eine nahezu von der Hardware unabhängige Beschreibung von Anwendungen zu ermöglichen.

Prototypen der Softwareprodukte werden auf der Vision 2003 präsentiert. Die microEnable Produktlinie wird dort ebenfalls bei Stemmer Imaging präsentiert. VisualApplets Studio und das Programmierwerkzeug sind ab Q2/2004 verfügbar.

DIE AUTOREN

Dr. Ralf Lay,
Geschäftsführer
SILICONSOFTWARE

Dr. Helmut Schwarz
Sales Manager,
Vertriebsbüro München

Tel. 0621/789507 0
info@silicon-software.de,
schwarz@silicon-software.de,
www.silicon-software.de

