

Christian Kirbach

SSL-Beschleunigung mit modernen 3D-Grafikkarten

Bibliografische Information der Deutschen Nationalbibliothek:

Bibliografische Information der Deutschen Nationalbibliothek: Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de/> abrufbar.

Dieses Werk sowie alle darin enthaltenen einzelnen Beiträge und Abbildungen sind urheberrechtlich geschützt. Jede Verwertung, die nicht ausdrücklich vom Urheberrechtsschutz zugelassen ist, bedarf der vorherigen Zustimmung des Verlanges. Das gilt insbesondere für Vervielfältigungen, Bearbeitungen, Übersetzungen, Mikroverfilmungen, Auswertungen durch Datenbanken und für die Einspeicherung und Verarbeitung in elektronische Systeme. Alle Rechte, auch die des auszugsweisen Nachdrucks, der fotomechanischen Wiedergabe (einschließlich Mikrokopie) sowie der Auswertung durch Datenbanken oder ähnliche Einrichtungen, vorbehalten.

Copyright © 2005 Diplomica Verlag GmbH
ISBN: 9783836613101

Christian Kirbach

SSL-Beschleunigung mit modernen 3D-Grafikkarten

Christian Kirbach

SSL-Beschleunigung mit modernen 3D-Grafikkarten

Christian Kirbach
SSL-Beschleunigung mit modernen 3D-Grafikkarten

ISBN: 978-3-8366-1310-1

Druck Diplomica® Verlag GmbH, Hamburg, 2008

Zugl. Universität Siegen, Siegen, Deutschland, Diplomarbeit, 2005

Dieses Werk ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere die der Übersetzung, des Nachdrucks, des Vortrags, der Entnahme von Abbildungen und Tabellen, der Funksendung, der Mikroverfilmung oder der Vervielfältigung auf anderen Wegen und der Speicherung in Datenverarbeitungsanlagen, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Eine Vervielfältigung dieses Werkes oder von Teilen dieses Werkes ist auch im Einzelfall nur in den Grenzen der gesetzlichen Bestimmungen des Urheberrechtsgesetzes der Bundesrepublik Deutschland in der jeweils geltenden Fassung zulässig. Sie ist grundsätzlich vergütungspflichtig. Zuwiderhandlungen unterliegen den Strafbestimmungen des Urheberrechtes.

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk berechtigt auch ohne besondere Kennzeichnung nicht zu der Annahme, dass solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und daher von jedermann benutzt werden dürften.

Die Informationen in diesem Werk wurden mit Sorgfalt erarbeitet. Dennoch können Fehler nicht vollständig ausgeschlossen werden, und die Diplomarbeiten Agentur, die Autoren oder Übersetzer übernehmen keine juristische Verantwortung oder irgendeine Haftung für evtl. verbliebene fehlerhafte Angaben und deren Folgen.

© Diplomica Verlag GmbH
<http://www.diplom.de>, Hamburg 2008
Printed in Germany

Danksagung

Ich widme diese Seite allen Personen, die zur Entstehung dieser Diplomarbeit in unersetzbarer Weise beigetragen haben.

An erster Stelle richtet sich mein Dank an meine Eltern für ihre durchgehende Unterstützung und Befürwortung meines Studiums. Ich danke für die wissenschaftliche Betreuung Univ.-Prof. Dr. Christoph Ruland und Dipl.-Ing. Luigi Lo Iacono vom Lehrstuhl für digitale Kommunikationssysteme der Universität Siegen. Ebenso bedanke ich mich für die freundliche Zusammenarbeit und Unterstützung bei Dipl.-Inf. Nicolas Cuntz vom Lehrstuhl für Computergrafik für die ständige Hilfestellung bei Fragen zur Implementierung und seinen Anregungen zu neuen Lösungsansätzen.

Mein besonderer Dank gilt auch Dipl.-Inf. Dominik Göddeke vom Institut für Angewandte Mathematik und Numerik der Universität Dortmund. Er hat mir freundlicherweise Zugang zu seinem noch unveröffentlichten Tutorial gegeben.

Mein Dank gilt allen Korrekturlesern und -leserinnen: Dipl.-Ing. Andre Schäfer, Katharina Kirbach, Gunnar Spickermann, Hendrik Payer, Stefan Siebel und Dirk Sommer.

Zusammenfassung

Datenaustausch über das Internet gewinnt zunehmend an Bedeutung. Viele Geschäftsbereiche verlassen sich schon heute auf verschlüsselte Kommunikation, wie z.B. Online Banking und Online Shopping. Dabei erfordert insbesondere die Verwendung asymmetrischer kryptographischer Verfahren zur Absicherung der Kommunikation viel Rechenleistung. Aus diesem Grund setzen die Betreiber von Internet-Servern spezielle Hardware-Beschleunigerkarten ein. Sie übernehmen die aufwendigen Berechnungen und entlasten so die CPU der Server. Ein Nachteil dieser Krypto-Hardware sind hohe Anschaffungskosten. Der wachsende Bedarf nach Sicherheit in Informations- und Kommunikationssystemen beflügelt die Suche nach Alternativen, um die enorme Rechenlast zu bewältigen.

Moderne 3D-Grafikkarten sind in den letzten Jahren besonders leistungsstark geworden. Ihr jährliches Leistungswachstum übertrifft sogar das der verbreiteten Mikroprozessoren. Ein großer Verkaufsumsatz hat sie zu günstiger Massenware werden lassen. Zusätzlich macht sie eine steigende Programmierbarkeit sehr flexibel. Die moderne Wissenschaft nutzt Grafikprozessoren schon heute zur Berechnung von Finite Elemente Simulationen und hochauflösenden Computertomographien in Echtzeit. Neue Anwendungsbereiche werden ständig erschlossen. Im Rahmen dieser Diplomarbeit wird die Möglichkeit analysiert und bewertet, moderne 3D-Grafikkarten für die Berechnungen asymmetrischer kryptographischer Operationen einzusetzen. Insbesondere wird der weit verbreitete RSA-Algorithmus untersucht, welcher auf Exponentiationen von sehr großen Ganzzahlen und Modulo-Operationen basiert. Eine geeignete Grafikkartenbibliothek wie das Plattform übergreifende OpenGL ist erforderlich, um alle Operationen zu steuern. Sie wird eingesetzt, um Daten von und zur Grafikkarte zu transferieren und die erforderlichen Rechenvorgänge des Grafikprozessors anzustoßen. Die programmierbaren Einheiten des Prozessors werden mit Hilfe der Programmiersprache Cg programmiert, ein Derivat der Hochsprache C, das speziell auf die Programmierung von Grafikprozessoren angepasst wurde. Hardware übergreifende Cg-Profile stellen sicher, dass die Programme auf einer Vielzahl von Grafikkarten eingesetzt werden können.

Zum Schluss wird die Geschwindigkeit dieser Implementierung mit der von gängigen RSA-Implementierungen verglichen, die ausschließlich Berechnungen auf der CPU durchführen.

Abstract

Secure internet communications are constantly becoming more important. Lots of businesses rely on encrypted transactions, e.g. online banking and online shopping. In particular asymmetric SSL encryption requires lots of computational power. Dedicated hardware accelerator cards are usually deployed at large server sites. They perform expensive calculations efficiently in favour of the server's CPU. One major drawback of cryptographic hardware is their high price tag. As demand for security increases new possibilities are embraced to handle the workload.

3D graphics cards have become increasingly powerful mainstream products in recent years. Their power growth rate even exceeds general purpose microprocessors. They are in mass production and cheap. Additionally improving programming capabilities have turned them into flexible and versatile devices. Modern science already computes finite element simulations and high-resolution real-time computer tomography on graphics processors. New applications continue to emerge.

This thesis analyses and evaluates the possibility of harnessing the power of modern 3D graphics cards for asymmetric cryptography. In particular the popular RSA algorithm is examined. It involves computing the power of large integer numbers and modulus operations. A suitable graphics library like the cross-platform OpenGL is required for controlling all necessary operations. It is used to perform data transfers to and from the graphics card and initiate calculations done by GPU programs. These shader programs will be written in the Cg programming language, a derivative of C adjusted for GPU programming. Vendor independent Cg profiles ensure that programs can be compiled for a variety of cross-vendor GPUs.

Finally the performance will be evaluated and compared to state-of-the-art RSA implementations that run on CPUs.

Inhaltsverzeichnis

1	Einleitung	1
2	Moderne 3D-Grafikkarten	4
2.1	Einbindung in die Computerarchitektur	5
2.2	Die Render-Pipeline	6
2.3	Der Vertex-Shader	7
2.4	Der Fragment-Shader	8
3	Programmierung von 3D-Grafikkarten	10
3.1	Zusammenspiel beteiligter Komponenten	11
3.2	OpenGL	12
3.3	Direct3D	13
3.4	Cg - C for Graphics	14
3.5	Fragment-Programme	16
4	Modulare Langzahl-Arithmetik	19
4.1	Integer-Multiplikation	20
4.2	Parallelisierung der Integer-Multiplikation	21
4.2.1	Parallelisierung mit eindimensionalen Gittern	21
4.2.2	Parallelisierung mit zweidimensionalen Gittern	26
4.3	Modulo-Berechnung	31
4.4	Modulare Exponentiation	33
4.4.1	Square-and-Multiply Algorithmus	33
4.4.2	Links-nach-rechts Methode	36
4.4.3	Rechts-nach-links Methode	37
4.4.4	Sliding Window Exponentiation	37
5	Implementierung	39
5.1	Anforderungen und Rahmenbedingungen	40
5.2	Systemarchitektur	42
5.3	Realisierung	44
5.3.1	Datenformate	45
5.3.2	Frame Buffer Objects	46
5.3.3	Erzeugung von Texturen	46
5.3.4	Transfer von Texturdaten	47
5.3.5	Berechnung der Partialprodukte	49

5.3.6	Bildung der Spaltensummen	50
5.3.7	Verarbeitung der Überträge	54
5.4	Funktionsübersicht der Implementierung in C	59
5.5	Tests und Debugging	62
6	Integration in OpenSSL	64
6.1	API für OpenSSL	64
6.2	Änderungen an OpenSSL	67
6.2.1	Anpassungen am Quellcode	67
6.2.2	Änderungen an Makefiles	68
7	Evaluierung	70
7.1	Vergleich mit CPU-Lösung	70
7.2	Zeitbedarf wichtiger Operationen	73
7.3	Ansatzpunkte für Verbesserungen	75
8	Zusammenfassung und Ausblick	77
	Literaturverzeichnis	79
A	Installation der Software	82
A.1	Systemvoraussetzungen	82
A.2	Installationsanweisungen	82
B	Hilfs- und Testanwendungen	84
B.1	Java-Applikation zum Vortrag	84
B.2	Messfunktionen	86
B.3	Testfunktionen	89
C	Messergebnisse	91
C.1	Zeitnahme mit GPU-Implementierung	91
C.2	Wichtige Teilberechnungen der GPU-Implementierung	91
C.3	Zeitmessungen von CPU-Lösungen	93
D	Die Software zur Diplomarbeit	95