

DIGITAL SIGNAL PROCESSING USING CUDA

Digital Signal Processing using CUDA

Nico Wehmeier, Richard Pfeifer, Fabian Jung





Inhalt

Aufgabenstellung

berblick

Host Code

Verwaltung Devices

Levenberg Marquardt

Benchmark

Skalierbarkei

Danksagung



01 Aufgabenstellung

- Ausgangsitutation
 - Messgerte erzeugen Datenstrom
 - Datenstrom nicht kontinuierlich
 - Serielle Implementierung
- Anforderungen
 - Portierung auf GPU
 - Hoher Datendurchsatz
 - Skalierbar auf bis zu 4 GPUs/Node



berblick

- Host.
 - Eingabe Datei auslesen
 - Daten zwischenspeichern
- Verwaltung Devices
 - Daten aus Puffer lesen
 - Zu den Devices streamen
 - Kernel starten
 - Ausgabe schreiben
- Levenberg Marquardt
 - Parameter einer Nherungsfunktion bestimmen
 - Markante Stellen (Anfangs-, Endwert, Maximum) ermitteln und zurckgeben



Host Code



Verwaltung Devices

- Jedes Devices wird von eigeneme Thread verwaltet
- Asynchrone Aufrufe
 - Pipeline



Levenberg Marquardt (1)

- Eingabedaten
 - Samples
 - Compute Capability 1.x: ca. 800)
 - Compute Capability 2.0 oder hher: ca. 2500)
 - Interpolationsschritt
 - beliebige Dezimalzahl grer 0)
 - Interpolation durch Texture Memory)



Levenberg Marquardt (2)

- Verarbeitung
 - eine Ausgleichungsrechnung pro Block
 - Grund: ca. das 5-fache der Sampleanzahl an Shared Memory bentigt (bei 1000 Samples ca. 20 kB)
 - Zugriff auf Samples durch Texture Memory
 - Shared Memory gespart
 - schnelle Interpolation mglich
 - Vorgehensweise
 - Anfangs- und Endwert ermitteln
 - abhngig vom Schwellwert Bereich festlegen
 - fr den Bereich Nherungsfunktion ermitteln
 - Qualitt durch Residuen und Maximum der Funktion ermitteln



Levenberg Marquardt (3)

- Ausgabedaten
 - -3 Parameter einer quadratischen Funktion: $a * x^2 + b * x + c$
 - Anfangs- und Endwert
 - Maximum
 - durchschnittliche Abweichung
 - Status (Fehler, Erfolg, Abbruch)



Levenberg Marquardt (4)

- Ergebnis
- [TODO Bild 'Beispielergebnis' einfgen]



Benchmark



Skalierbarkeit



Danksagung