HOCHSCHULE COBURG

Hochschule für angewandte Wissenschaften Coburg

Fakultät Elektrotechnik und Informatik

Studiengang: Informatik

Bachelorarbeit

# Entwicklung einer hardwarebeschleunigten Berechnung der Mandelbrotmenge auf einem FPGA

von

Daniel Kirchner

Matrikelnummer: 02219415

Abgabe der Arbeit: 15.07.2019

Betreut durch: Prof. Oliver Engel, Hochschule Coburg

**Zusammenfassung**

Im Rahmen dieser Bachelorarbeit wurde eine hardwarebeschleunigte Visualisierung der Mandelbrotmenge auf einem FPGA realisiert. Hierfür werden diverse mathematische und designtechnische Performanceoptimierungen vorgestellt, welche dann in einparalleles FPGA-Design implementiert wurden. Weiterhin sollen einige Eigenschaften und Besonderheiten der Mandelbrotmenge und von Fraktalen im Allgemeinen aufgezeigt werden.
Das Projekt wurde für das Zybo Zynq-7000 Trainer Board entwickelt, welches über einen VGA-Output die Repräsentation des Fraktals in Form eines 800x600@60Hz Videosignals ausgibt. Zur optimalen Ausnutzung der auf diesem Board gegebenen Ressourcen (DSPs, BRAM) wurde die *Vivado Design Suite* mit dem integrierten IP-Katalog verwendet.

# Inhaltsverzeichnis

# Abbildungsverzeichnis

# Codebeispielverzeichnis

## Abkürzungsverzeichnis

| | |
|---|---|
| JAX-RS | Java API for RESTful Web Services |

# 1 Einleitung

## 1.1 Motivation

Die stets wachsende Zahl von Komponenten die auf einem mikroelektronischen Bauteil pro Zeiteinheit untergebracht wird, ist ein Phänomen, welcbes Gordon Moore schon im Jahr 1965 aufgefallen ist [2]. Die populäre, nach ihm benannte Beobachtung, dass die Anzahl der Transistoren pro integriertem Schaltkreis exponentiell mit der Zeit ansteigt ist allgemein als das Mooresche Gesetz bekannt.

Diese Gesetzlichkeit machte es möglich den stets wachsenden Leistungsanforderungen an moderne Hardware gerecht zu werden, indem immer mehr (und komplexere) identische Allzweck-Prozessoren (Kerne) pro CPU verbaut wurden.

Dieses Vorgehen kann jedoch nicht unbegrenzt lange betrieben werden, da die heute verwendeten MOS-Transistoren sich rapide ihren physikalischen Grenzen annähern. Ein besserer Umgang mit dem stetig steigenden Bedarf an Rechenleistung ist die Entwicklung von spezialisierter Hardware, welche zwar nur ein kleines Aufgabenspektrum abdeckt, dies jedoch mit hoher Performanz und Energieeffizienz tut.

Ein Beispiel hierfür ist die moderne Grafikkarte (GPU), welche dem Prozessor Darstellungsberechnungen abnimmt, wodurch dieser mehr Zeit hat, andere Aufgaben zu übernehmen. Die Grafikkarte führt diese Aufgaben mit enorm hohem Durchsatz und niedrigen Berechnungszeiten durch, welche ein herkömmlicher Prozessor alleine nicht erreichen könnte.

Auch andere Hardwarekomponenten, wie die Netzwerkkarte, kryptographische Beschleuniger, oder Soundkarten sind in fast allen Computersystemen verbaut und entlasten den Hauptprozessor. Man spricht auch von heterogenen Computersystemen.

Die im Rahmen dieser Arbeit vorgestellte Hardware soll ein Beispiel für eine derartige heterogene Komponente sein. Auf einem Field Programmable Gate Array (FPGA,) soll eine performante und energieffiziente Visualisierung der sogenannten Mandelbrotmenge realisiert werden. Diese Problemstellung ist auch durch einen ordinären Prozessor lösbar, lastet diesen jedoch enorm aus und ist somit auch sehr energieineffizient.

## 1.2 Aufgabenstellung

Die Aufgabe dieses Projektes ist es, eine komplett in Hardware stattfindende Berechnung der Mandelbrotmenge durchzuführen und die Ergebnisse über eine VGA-Schnittstelle darzustellen

(ähnlich Bild)

Weiterhin soll die Hardware durch externe Peripherie konfigurierbar hinsichtlich der angestellten Berechnungen sein. So soll etwa aktuell abgebildete Bereich der Mandelbrotmenge oder auch die Farbgebung der Darstellung im laufenden Betrieb geändert werden können.

Hierfür wurde das FPGA-Trainer Board *Zybo Zynq-7000* zur Verfügung gestellt, welches in genauer vorgestellt wird.

# 2 Technische Grundlagen

Zum besseren Verständnis des Gesamtprojektes sollen in diesem Kapitel einige technische Konzepte erläutert werden.

## 2.1 FPGAs

Ein *Field Programmable Gate Array* (kurz FPGA) ist ein Schaltkreis, welcher mit Hilfe von Hardwarebeschreibungssprachen konfiguriert werden kann, um beliebig komplexe logische Schaltungen zu realisieren.
Das Grundelement eines solchen Bausteines bilden die sogenannten *Lookup Tables* (kurz LUTs), welche zu einem beliebigen n-bit Input ein 1-bit Output Signal produzieren. Die zugrundeliegende Logiktabelle einer LUT ist hierbei frei programmierbar.

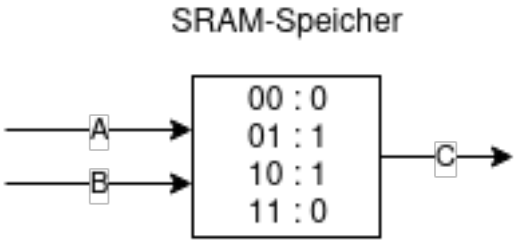


Abbildung 1: Beispielhafter Aufbau einer 2-Input LUT

Eine LUT, welche die Operation $C = A \oplus B$ implementiert ist in Abbildung 1 zu sehen. In dieser wird in einem SRAM-Speicher für jede Inputkombination ein Outputwert hinterlegt, wodurch jede 2-Bit Funktion abgebildet werden kann. In Verbindung mit einem Flipflop [1] bildet eine LUT dann einen sogenannten Logikblock (s. Abbildung 2). [1]

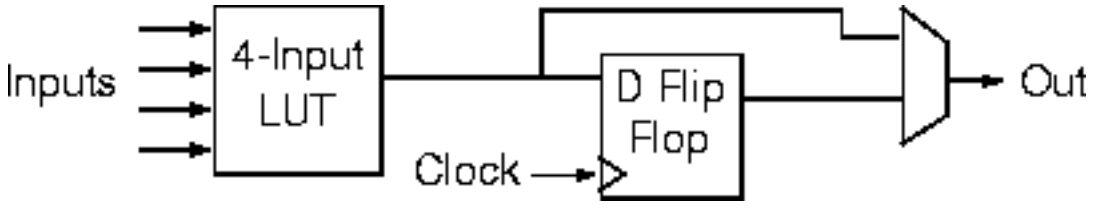


Abbildung 2: Logikblock, aus [1]

Ein FPGA verbindet nun durch ebenfalls konfigurierbare Bussysteme viele solcher Logikblöcke um komplexere Schaltkreise abzubilden.

[1] Ein Flipflop ist ein Speicherelement, welches einen einzigen Bit Daten halten kann.

Weiterhin verfügen FPGA-Boards oft noch über ergänzende Hardwarekomponenten, von denen die im Falle dieses Projektes vorhandenen im Folgenden gezeigt werden sollen.

**DSP** Ein *Digitaler Signalprozessor* (DSP) ist ein fest integrierter Baustein, welcher durch Multiplizierer und Akkumulatoren binäre Algorithmen beschleunigt. So übernimmt dieser etwa grundlegende mathematische Operationen, was dazu führt, dass diese flächen- und energiesparender durchgeführt werden, als bei der Verwendung von LUTs [3, S. 52]. Typische Anwendungsgebiete dieser Bausteine sind Fließkommamultiplikationen, Schnelle Fourier-Transformationen oder einfache Zähler [3, S. 14]. In diesem Projekt wurden die DSPs hauptsächlich aufgrund ihrer 25x18 Bit Multiplizierer verwendet, welche kaskadiert werden können um beliebig Breite Multiplikationen durchzuführen.

**Block RAM** FPGAs verfügen meist über *Block RAM* (BRAM), welcher zur Speicherung binärer Daten dient. Dieser Speicher ist lese-/schreibsynchron, wodurch Inkonsistenzen beim Speicherprozess ausgeschlossen sind [3, S. 11]. Um auf diese Speicherblöcke Zugriff zu erlangen muss der von Xilinx mitgelieferte Baustein ”Block RAM Generator”verwendet werden.

**IO-Komponenten** Zur Kommunikation mit der Außenwelt verfügen FPGAs über diverse Schnittstellen wie z.B. Knöpfe, Schalter, aber auch komplexere Anbindungen wie etwa VGA (s. hierzu Abschnitt 2.2) oder PMOD-Anschlüsse. Diese sind so angebunden, dass ihre Signale direkt in Logikschaltungen von LUTs integriert werden können.

## 2.2 VGA-Interface

# 3 Theoretische Grundlagen

## 3.1 Fraktale

## 3.2 Die Mandelbrotmenge

# 4 Test und Umsetzung

## 4.1 Gesamtsystem

## 4.2 Komponentenbeschreibung

### 4.2.1 Mandelbrot-Core

### 4.2.2 Mandelbrot-Koordinator

### 4.2.3 Dual Port Block RAM

### 4.2.4 Lookup Tables

## 4.3 Peripherie und funktionale Beschreibung

# 5 Optimierungen

## 5.1 Algebraische Optimierung

## 5.2 Designoptimierungen

# Literaturverzeichnis

[1] FPGA Architecture for the Challenge. `http://www.eecg.toronto.edu/~vaughn/challenge/fpga_arch.html`. Zugriff am: 12.06.2019.

[2] Gordon E Moore et al. Cramming more components onto integrated circuits, 1965.

[3] Xilinx, Inc. *7 Series DSP48E1 Slice*, März 2018. v1.10.