Übung 4
Computational Physics III

Matthias Plock (552335) Paul Ledwon (561764)

30. Mai 2018

Inhaltsverzeichnis

1 Aufgabe 1

1.1 (Aufgabenteil i) Skizzierung der Reduktion

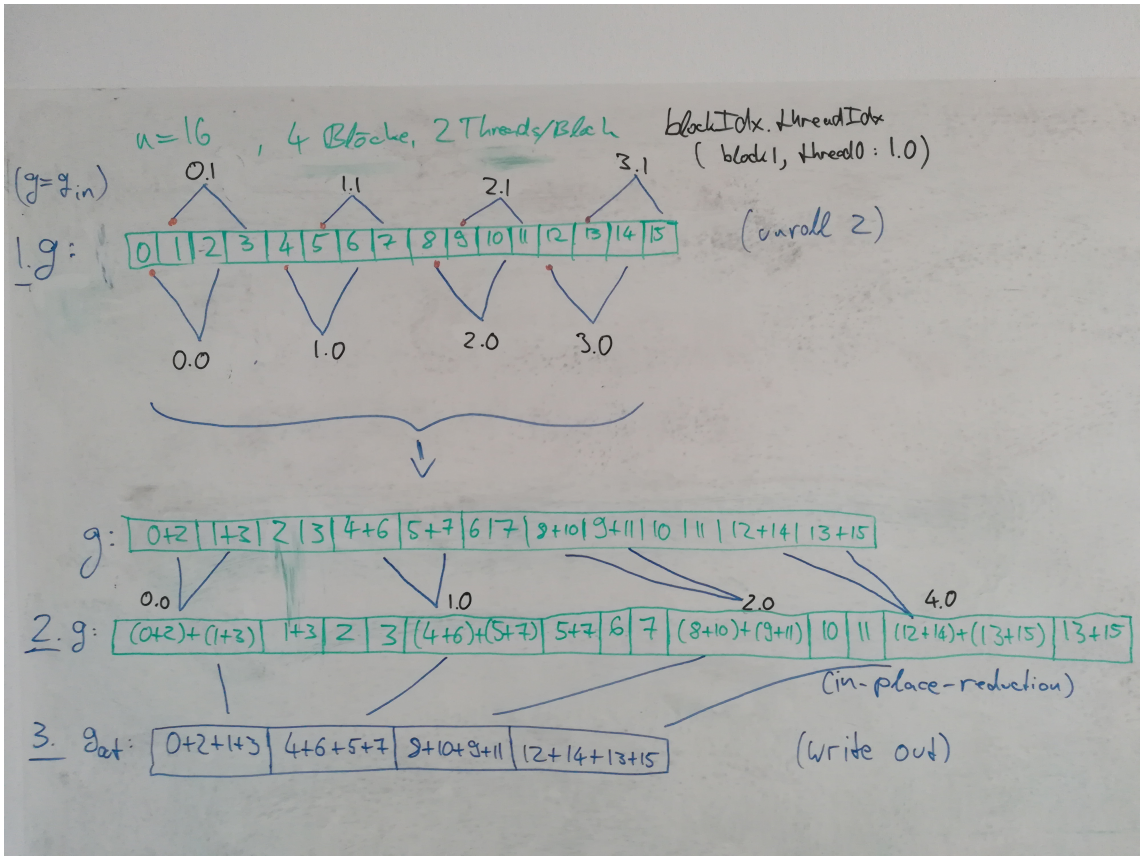
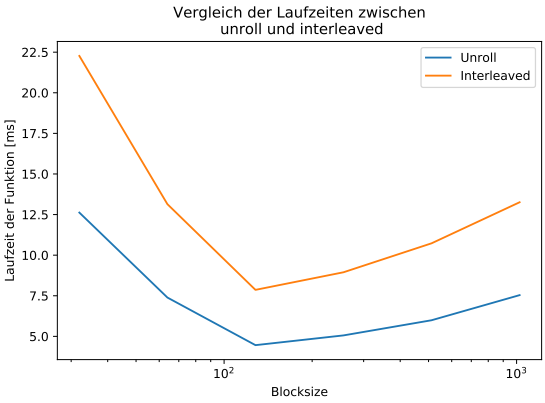
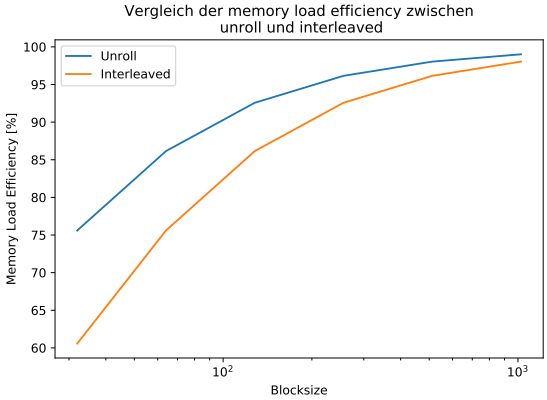


Abbildung 1: Skizzierung der Reduktion von einem Array mit 16 Elementen

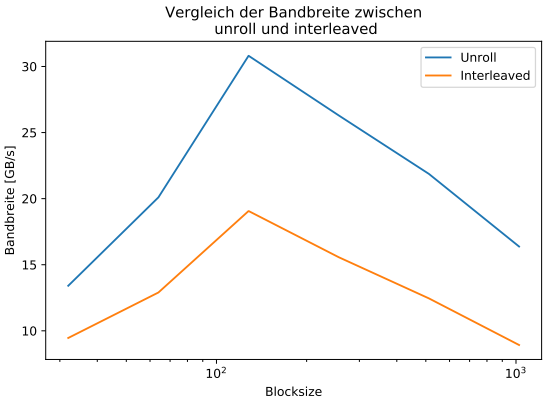
1.2 (Aufgabenteil ii, iii, iv) Messung der Laufzeiten, Global Memory Load Efficiency, Global Load Throughput und Achieved Occupancy



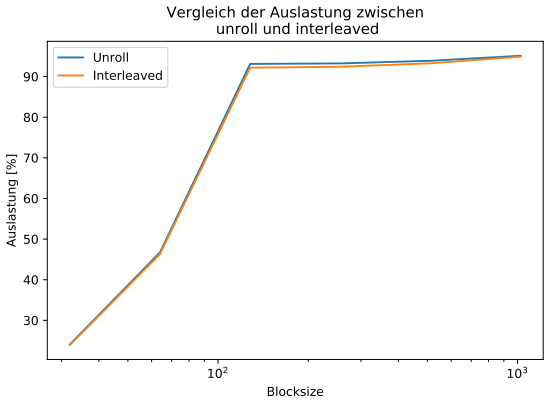
(a) Laufzeiten in Abhaengigkeit der gewaehlten Blocksize.



(b) Memory Load Efficiency in Abhaengigkeit der gewaehlten Blocksize.



(c) Bandbreitennutzung in Abhaengigkeit der gewaehlten Blocksize.



(d) Achieved Occupancy in Abhaengigkeit der gewaehlten Blocksize.

Abbildung 2: Unterschiedliche Parameter in Abhaengigkeit der Blocksize. Man sieht, dass die Laufzeit und die Bandbreitennutzung bei einer Blocksize von 128 (d.h. 4 warps pro Block) am optimalsten ist. Ab diesem Wert ist auch die Auslastung im Bereich von ueber 90%.

In Abb. 2a) sind die Laufzeiten fuer den interleavedReduce-Algorithmus und den UnrollReduce-Algorithmus in Abhaengigkeit dargestellt. Den größten Einfluß auf die Laufzeit scheint die "Global Load Throughput", also die Bandbreite, zu haben. Da die auf den Threads ausgeführten Berechnungen numerisch nicht sehr aufwändig sind, wird ein Großteil der Zeit benoetigt, um die Daten zwischen Host und Device zu kopieren. Fuer die Blocksize von 128, bei der die Laufzeit minimal ist, ist auch die Bandbreite am größten. Eine höhere Auslastung und Memory Load Efficiency hat auch einen positiven auf die Laufzeit, bei größeren Blocksizes sind sowohl Auslastung und Memory Load Efficiency höher.

1.3 (Aufgabenteil v, vi)
Optimaler Wert fuer die Anzahl der Elemente q, die vor Reduktion summiert werden

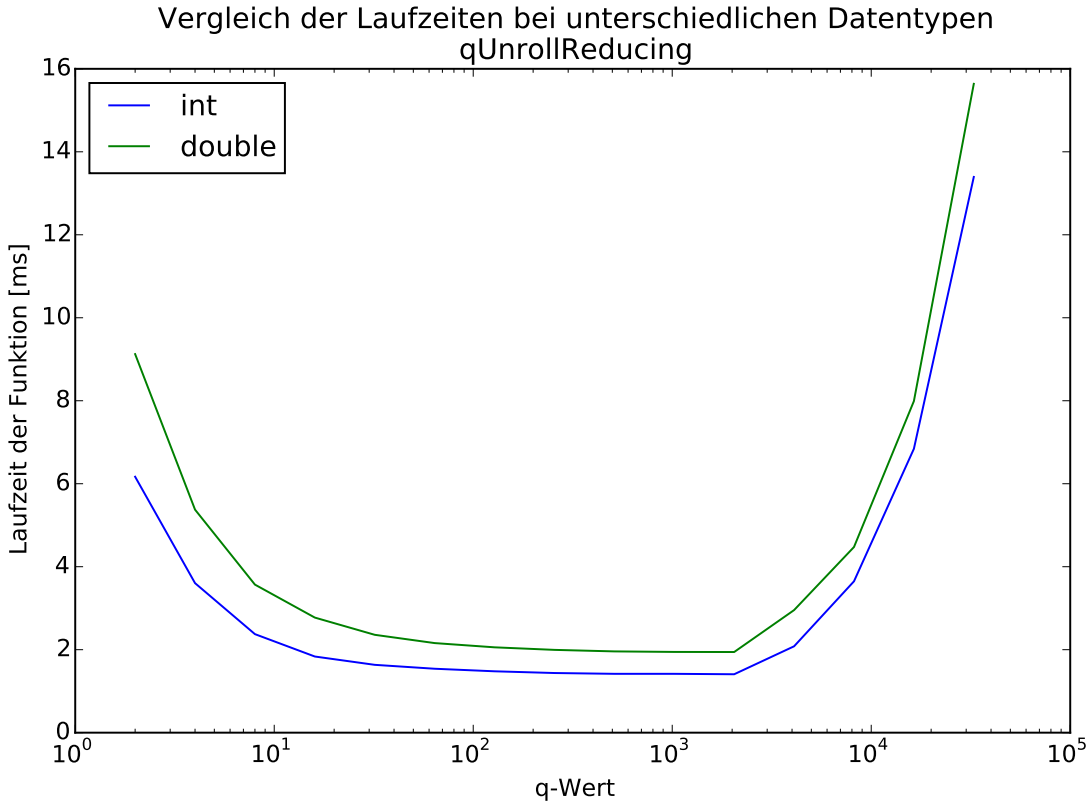


Abbildung 3: Laufzeiten der Methoden qReduceUnrolling und qReduceUnrollingDouble fuer verschiedene Werte von q. Vor der Reduktion werden q Werte pro Thread addiert. Das resultiert mitunter in einer kuerzeren Ausfuehrungszeit. Das Minimum bei den Datentypen single sowie double liegt bei q = 2048. Die Laufzeiten fuer den Datentyp double sind etwa 30% groesser als fuer den Datentyp int. Das koennte daran liegen, dass auf der Grafikkarte nicht so viele Register fuer den Datentyp double vorhanden sind.