

Übungsblatt 6

(10 Punkte)

Besprechung ab Dienstag, 6. Juni 2017

6.1 OpenCL - Eigenständiges Erweitern (9 Punkte)

Führen Sie folgende Änderungen am "Hello_World_OpenCL" Programm, mit welchem Sie sich bereits letzte Woche ausführlich beschäftigt haben, durch:

- a. Erweitern Sie das "Hello_World_OpenCL" Programm so, dass statt zwei Inputwerten und einem Outputwert drei Inputwerte und zwei Outputwerte erzeugt werden. Im folgenden benannt als W1, W2, W3, O1 und O2.
- b. Ändern Sie das Programm nun so, dass anstatt mit cl_int mit cl_float gerechnet wird.
Füllen Sie alle Inputwerte auch direkt mit zufälligen reellen Zahlen anstatt einer Durchnummerierung.
- c. Programmieren Sie das Programm so um, dass zwei unterschiedliche Kernel nacheinander ausgeführt werden:
 - (a) Der erste Kernel soll folgendes berechnen: $(W1+W2+W3)*(W1-W2-W3)$ und dies in O1 speichern
 - (b) Der zweite Kernel soll die Nachkommastellen der Zahl mit der Zahl vor dem Komma für jeden der drei Inputs multiplizieren und die Summe dieser Ergebnisse in O2 speichern. Für 337,4284 wäre das z.B. $337*0,4284=144,3708$.

6.2 Bonusaufgabe - Filter (1 Punkt)

Führen Sie folgende Änderungen am "Hello_World_OpenCL" Programm, mit welchem Sie sich bereits letzte Woche ausführlich beschäftigt haben, durch.

- Legen Sie ein `cl_int` Array an.
- Füllen Sie das Array mit Zufallswerten zwischen 0 und 255.
- Erstellen Sie einen Kernel, welcher jedes Feld neu berechnet. Dabei soll jeder Wert so angepasst werden, dass er 50% näher am Durchschnittswert der umliegenden Felder ist. Umliegende Felder wären bei Feld 1 zum Beispiel die Felder 2,4 und 5, wovon dann der Durchschnitt berechnet werden muss. Danach benötigt man noch die Differenz zwischen Ursprungswert und dem Durchschnitt, welche halbiert werden muss (50%) und dann auf den Ursprungswert addiert wird, um den neuen Wert zu erhalten. (siehe Beispiel)
- Denken Sie darüber nach, was Sie gerade gebaut haben und was für einen Anwendungszweck ein solches Verfahren hat.

Hinweis: Achten Sie darauf, dass Sie immer bei `cl_int` bleiben, also niemals Nachkommastellen erzeugen.

Beispiel:

1	2	3
4	5	6
7	8	9

In diesem Fall wären die neuen Werte für die Felder:

$$\begin{aligned}
 \text{Feld 1: } & 1 + (((2+4+5)/3-1)/2) & = 2 \\
 \text{Feld 2: } & 2 + (((1+3+4+5+6)/5-2)/2) & = 2 \\
 \text{Feld 3: } & 3 + (((2+5+6)/3-3)/2) & = 3 \\
 \text{Feld 4: } & 4 + (((1+2+5+7+8)/5-4)/2) & = 4 \\
 \text{Feld 5: } & 5 + (((1+2+3+4+6+7+8+9)/8-5)/2) & = 5 \\
 \text{Feld 6: } & 6 + (((2+3+5+8+9)/5-6)/2) & = 6 \\
 \text{Feld 7: } & 7 + (((4+5+8)/3-7)/2) & = 6 \\
 \text{Feld 8: } & 8 + (((4+5+6+7+9)/5-8)/2) & = 7 \\
 \text{Feld 9: } & 9 + (((5+6+8)/3-9)/2) & = 8
 \end{aligned}$$

Allgemeine Hinweise: Die Übungstermine und weitere Informationen finden Sie unter <http://ls12-www.cs.tu-dortmund.de/daes/de/lehre/lehrveranstaltungen/sommersemester-2017/rechnerarchitektur-ss17.html>. Die Übungsblätter werden in den Übungen bearbeitet und besprochen. Eine Abgabe vorher ist also nicht nötig, dennoch sollte man sich im Voraus mit den Übungen vertraut machen. Für die Teilnahme an der Klausur nach BPO 2013 / der Fachprüfung nach DPO 2001 ist der Übungsschein *nicht* erforderlich.