Vektorisierung

Seminar: Effiziente Programmierung

Richard Reiß

Arbeitsbereich Wissenschaftliches Rechnen
Fachbereich Informatik
Fakultät für Mathematik, Informatik und Naturwissenschaften
Universität Hamburg

2021-26-01

Gliederung (Agenda)

Data Alignment

- Vektorisierung ist eine Form von Data Alignment

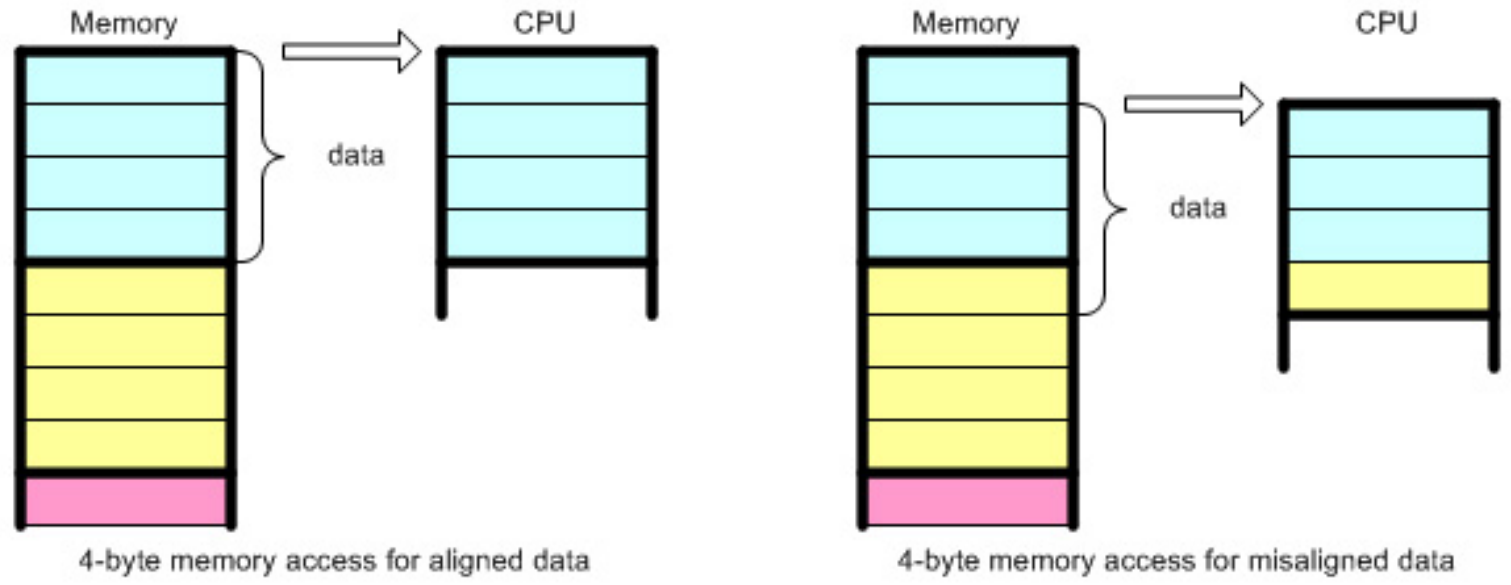

Abbildung: Unterschied zwischen dem Zugriff auf aligned und nicht aligned Daten [1]

Data Alignment Fortsetzung

- Zugriff auf nicht aligned Daten bzgl. Performanz ineffizient

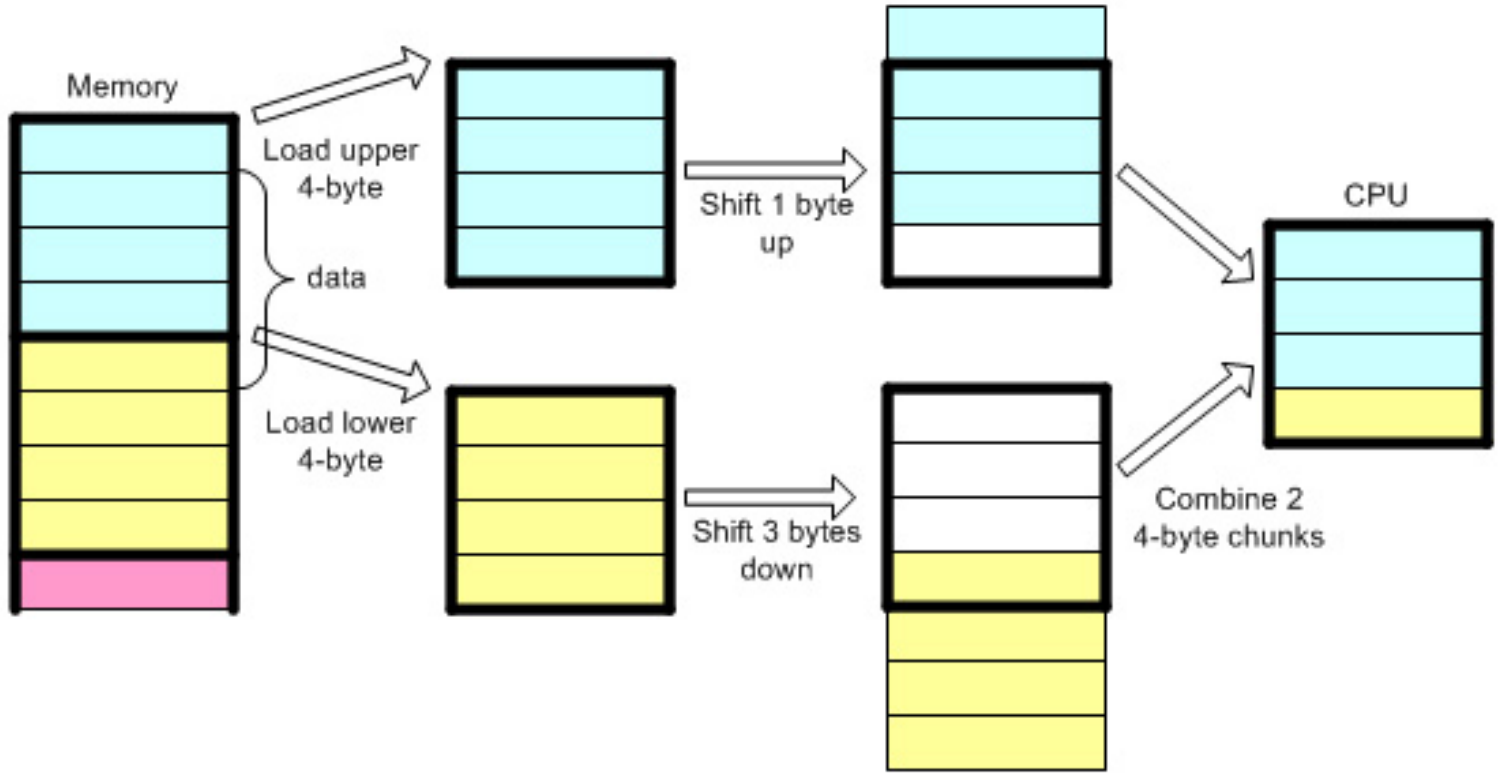

Abbildung: Ablauf eines Zugriffs auf nicht aligned Daten[1]

Vektorprozessor

- Ist Teil von modernen CPUs[2]
- Kann in einer Anweisung mehrere Daten bearbeiten[2]
- Wenn Daten optimal aligned, optimale Performanz[2]
- Für Performanz-Gewinn müssen Daten aligned sein[2]

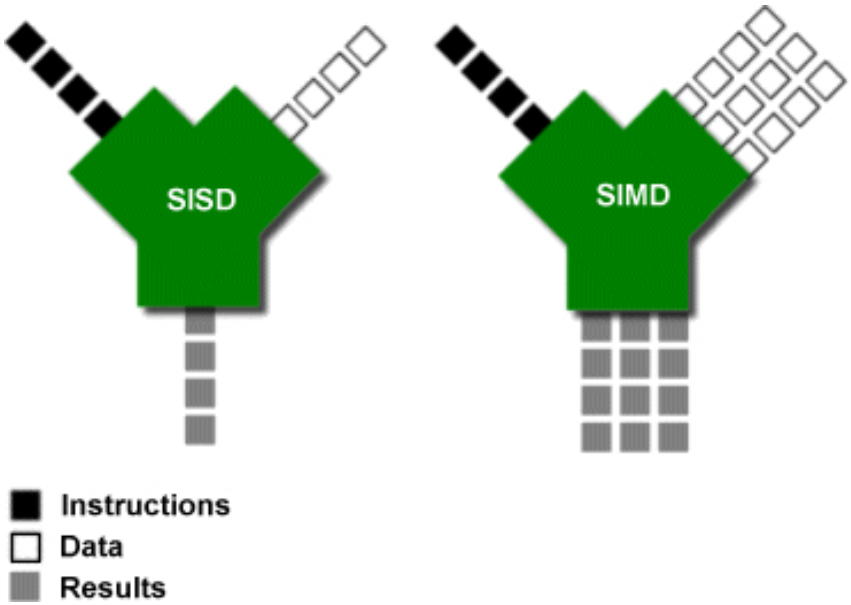

Abbildung: Arbeitsweise von Skalarprozessor(SISD) und Vektorprozessor(SIMD) [3]

Was sind Intrinsics?

- Hardware spezifische Befehlssätze
- effizienter als "manuell“ geschriebener Code
- erlauben Assembler ähnliche Befehle in höheren Sprachen[5]

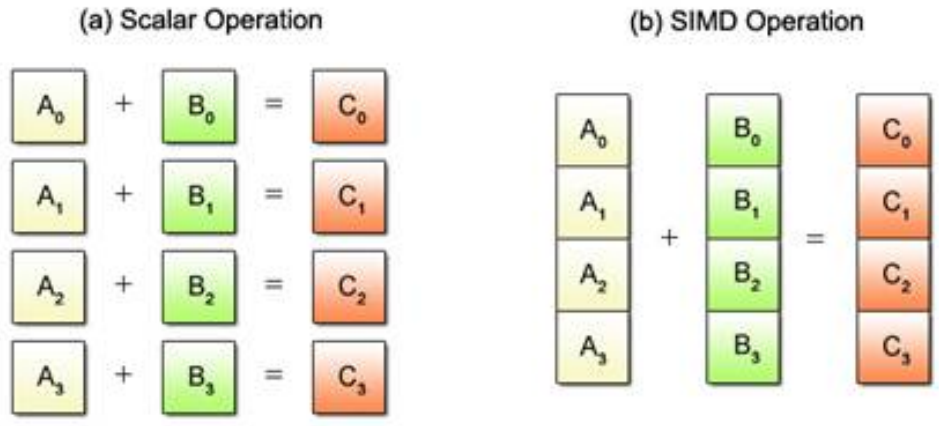

Abbildung: Für SIMD Instructions geeignete Berechnung[4]

Was sind Intrinsics?

- ungeeignet für das Bearbeiten von mehreren Daten in unterschiedlicher Form[4]

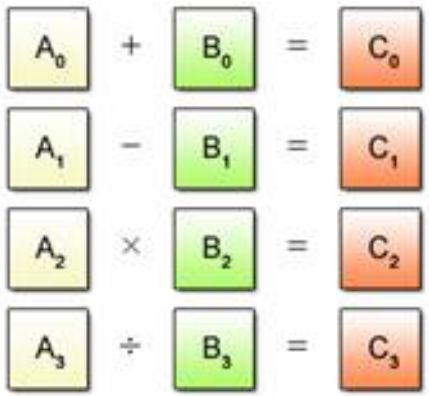

Abbildung: Für SIMD Instructions ungeeignete Berechnung[4]

Intel SIMD Intrinsics

- SSE (Streaming SIMD Extensions) 16 Byte Register
- AVX 1,2 (Advanced Vector Extensions) 32 Byte Register
- AVX-512 64 Byte Register

Befehlsstruktur in C++

- ___m128 _mm_add_ps (___m128 a, ___m128 b)
  - Addition von zwei float Vektoren(SSE)
- ___m256i _mm256_setr_epi64x (___int64 e3, ___int64 e2, ___int64 e1, ___int64 e0)
  - Beschreibt einen 32 Byte Vektor mit 8 Byte ints(AVX)
- Quelle: [5]

Code Beispiele

- (scalar und vektorisierter Code zur Berechnung $\Sigma_{x=1}^{2^{15}} x$)

Vor- und Nachteile von SIMD

- (+)effiziente Performanz
  - Intrinsics
  - optimale Nutzung der Register
- (-)Hardware Abhängigkeit (Portabilität)
- (-)schwierige Implementation
- (-)Wenn viele unterschiedliche Berechnungen auf einzelne Daten gebraucht werden, kein großer Performanz Gewinn
- ((-)Speichereffizienz)

Zukunftsaussicht

- Automatic vectorization
  - Machine Learning[6]
- Quelle[6]: KEVIN STOCK, LOUIS-NOEL POUCHET, and P. SADAYAPPAN,“Using Machine Learning to Improve Automatic Vectorization“

Zusammenfassung

- Vektorisierung ordnet Daten vektorprozessorkonform an
- Vektorprozessoren (SIMD)
- SIMD Intrinsics sind die effizientesten Befehle für Vektorisierte Daten
- Intrinsics sind hardwarespezifisch

Literatur I

[1] Song, Ho Ahn, "Data Alignment" *songho.ca*/2011/http://www.songho.ca/misc/alignment/dataalign.html.

[2] Roshni Y, "Vector Processor" *electronicsdesk.com*/https://electronicsdesk.com/vector-processor.html.

[3] https://arstechnica.com/features/2000/03/simd/

[4] http://ftp.cvut.cz/kernel/people/geoff/cell/ps3-linux-docs/CellProgrammingTutorial/BasicsOfSIMDProgramming.html

[5] Intel, "Intel Intrinsics Guide" *software.intel.com*/https://software.intel.com/sites/landingpage/IntrinsicsGuide/