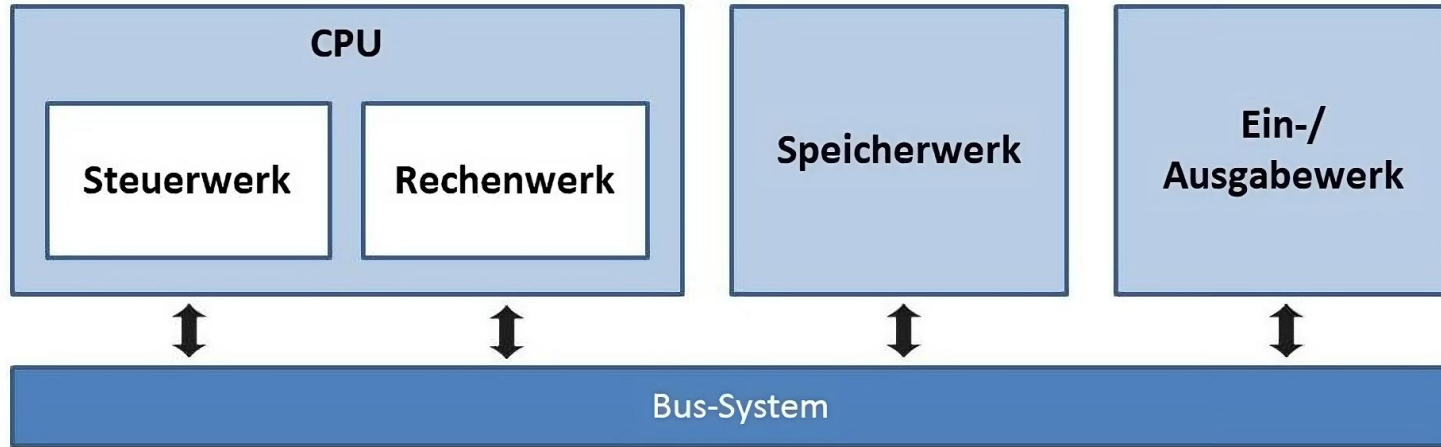


Translation Lookaside Buffer

Cache Simulation und Analyse

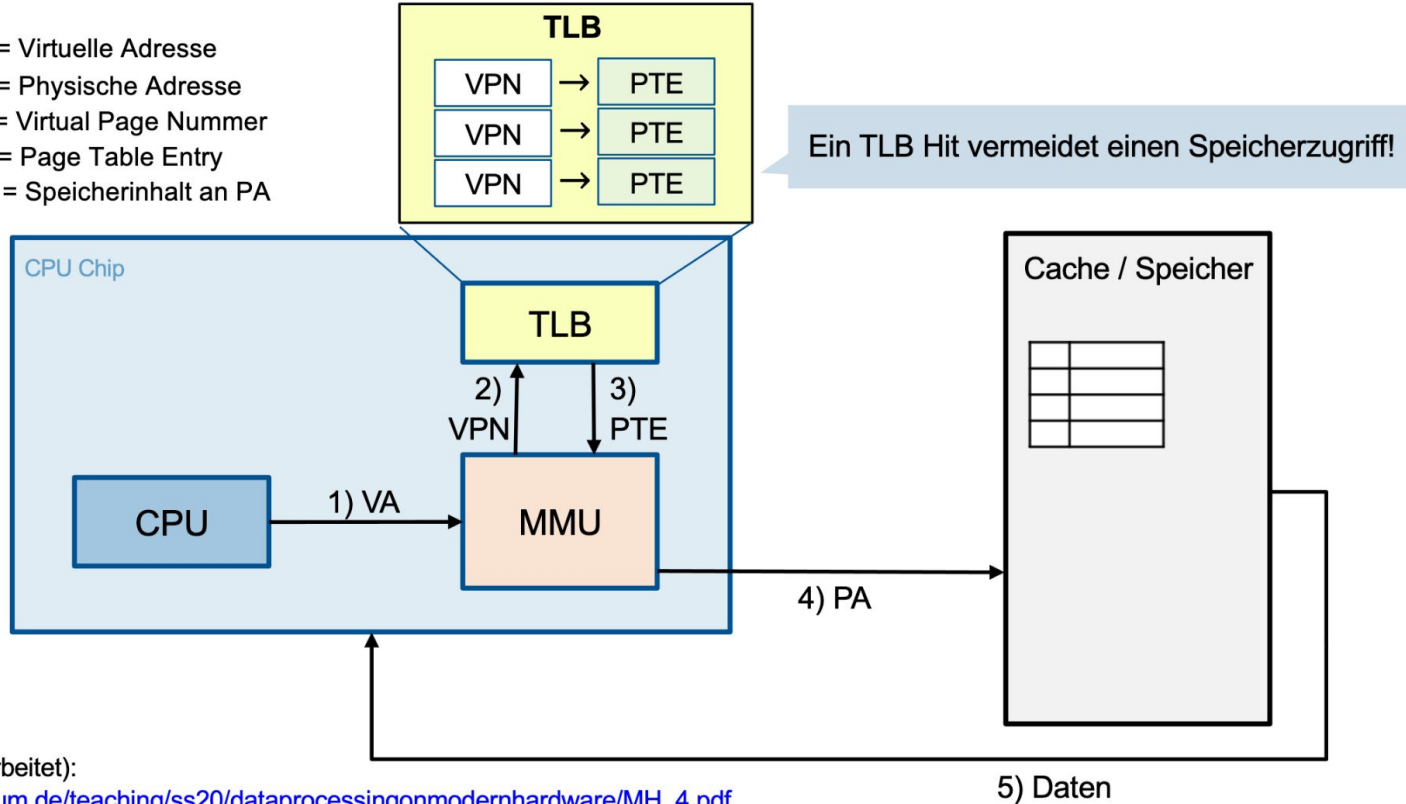
James Wagner, Simon Weckler, Kian Shirazi

Aufgabenstellung



Aufgabenstellung

VA = Virtuelle Adresse
PA = Physische Adresse
VPN = Virtual Page Nummer
PTE = Page Table Entry
Daten = Speicherinhalt an PA



Quelle (bearbeitet):

http://db.in.tum.de/teaching/ss20/dataprocessingonmodernhardware/MH_4.pdf

Adressen - Aufbau

Virtuelle Adresse:

Virtuelle Seitennummer	Seiten Offset
32 - b Bits	b Bits

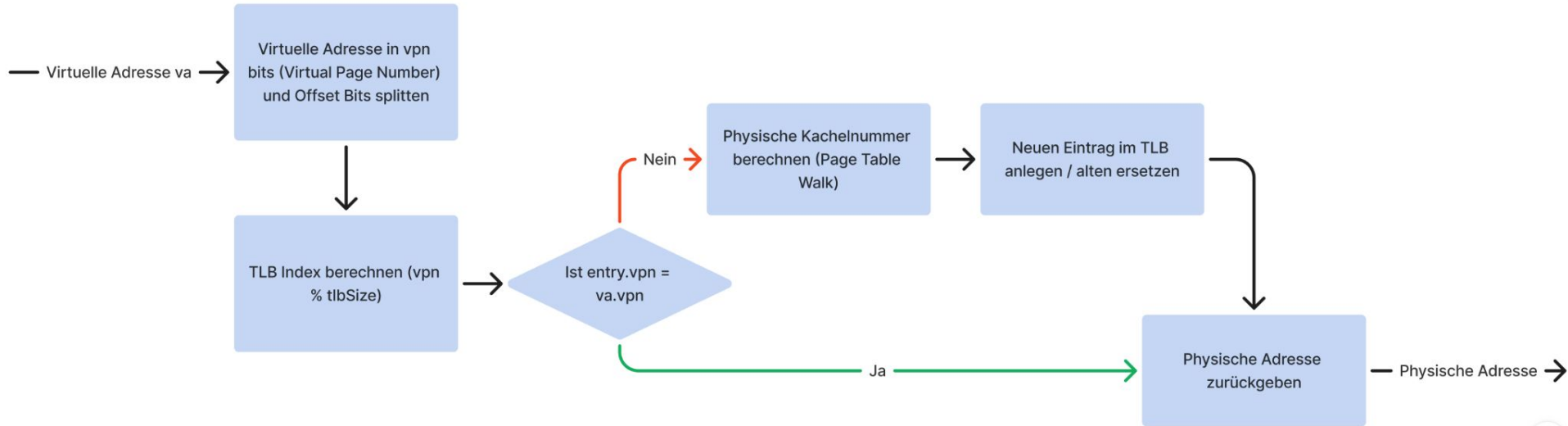
$$B = \text{Seitengröße} = \text{Kachelgröße}$$
$$b = \log_2(B)$$

Physische Adresse:

Physische Kachelnummer	Seiten Offset
32 - b Bits	b Bits

Translation Lookaside Buffer

Virtuelle Seitennummer	Physische Kachelnummer	Valid Bit
32 - b Bits	32 - b Bits	1 Bit



Beispiel - Summe über eine verkettete Liste

```
#include <stdio.h>

/* A Linked list node */
struct Node {
    int data;
    struct Node* next;
};

// function to find the sum of
// nodes of the given linked list
int sumOfNodes(struct Node* head)
{
    struct Node* ptr = head;
    int sum = 0;
    while (ptr != NULL) {

        sum += ptr->data;
        ptr = ptr->next;
    }

    return sum;
}
```

1. Initialisierung:
 - a. pointer auf aktuellen Knoten wird vom **Stack** gelesen
 - b. Summe wird auf dem **Stack** mit 0 initialisiert
2. Schleife:
 - a. Adresse auf Knoten wird vom **Stack** gelesen
 - b. Daten des Knoten werden vom **Heap** gelesen
 - c. Summe wird vom **Stack** gelesen
 - d. Summe + Daten des Knotens wird auf dem **Stack** gespeichert
 - e. Pointer auf nächsten Knoten wird vom **Heap** gelesen
 - f. Pointer auf nächsten Knoten wird auf den **Stack** geschrieben

Wie verhält sich eine Liste mit 1000 Knoten, verteilt auf einen Heap der Größe 64KiB?

Beispiel - Summe über eine verkettete Liste

Simulationsparameter:

TLB Latency	10 Zyklen
Memory Latency	60 Zyklen
Anzahl Requests	6006
TLB Größe	17 Einträge
Page Size	4096 B

Beispiel - Summe über eine verkettete Liste

Ergebnis:

Hits	5989
Misses	17
Ausführungszeit (Zyklen)	421440

$$\frac{64 \text{ KiB}}{4096 \text{ B}} = 16$$

Ausführungszeiten in Zyklen:

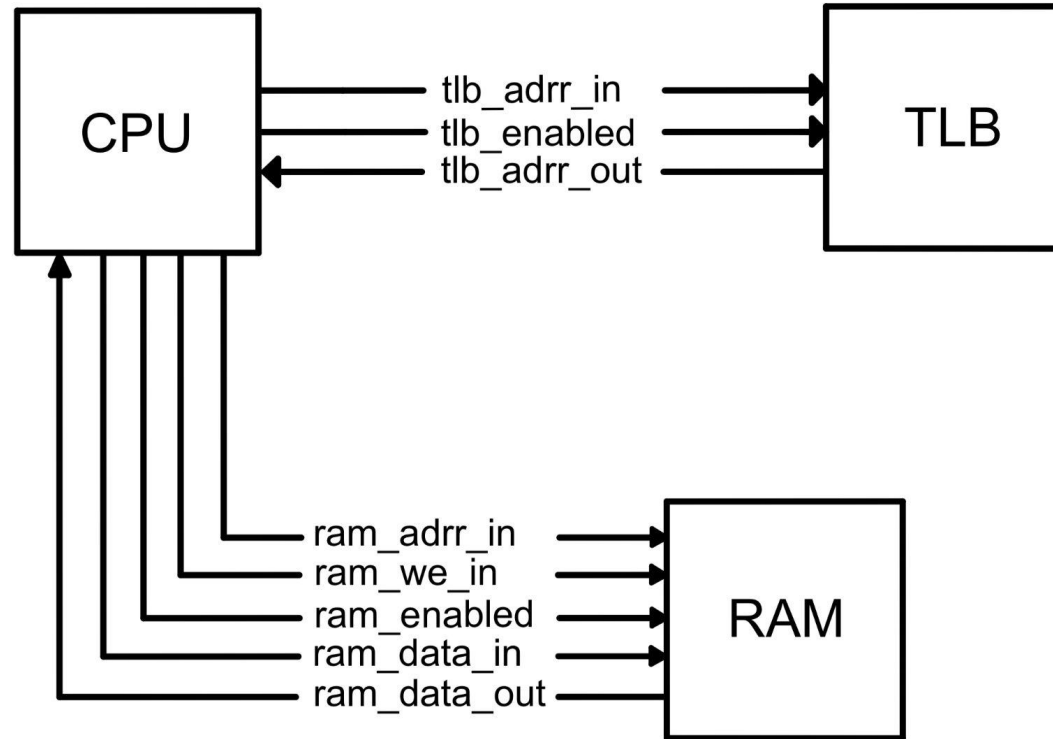
Maximale Ausführungszeit (ohne TLB)	$6006 \cdot 2 \cdot 60 = 720720$
Optimale Ausführungszeit (nur Hits)	$6006 \cdot 10 + 6006 \cdot 60 = 420420$
Unsere Simulation:	421440
Speedup:	$720720 / 421440 \approx 1.71$

Beispiel - Summe über eine verkettete Liste

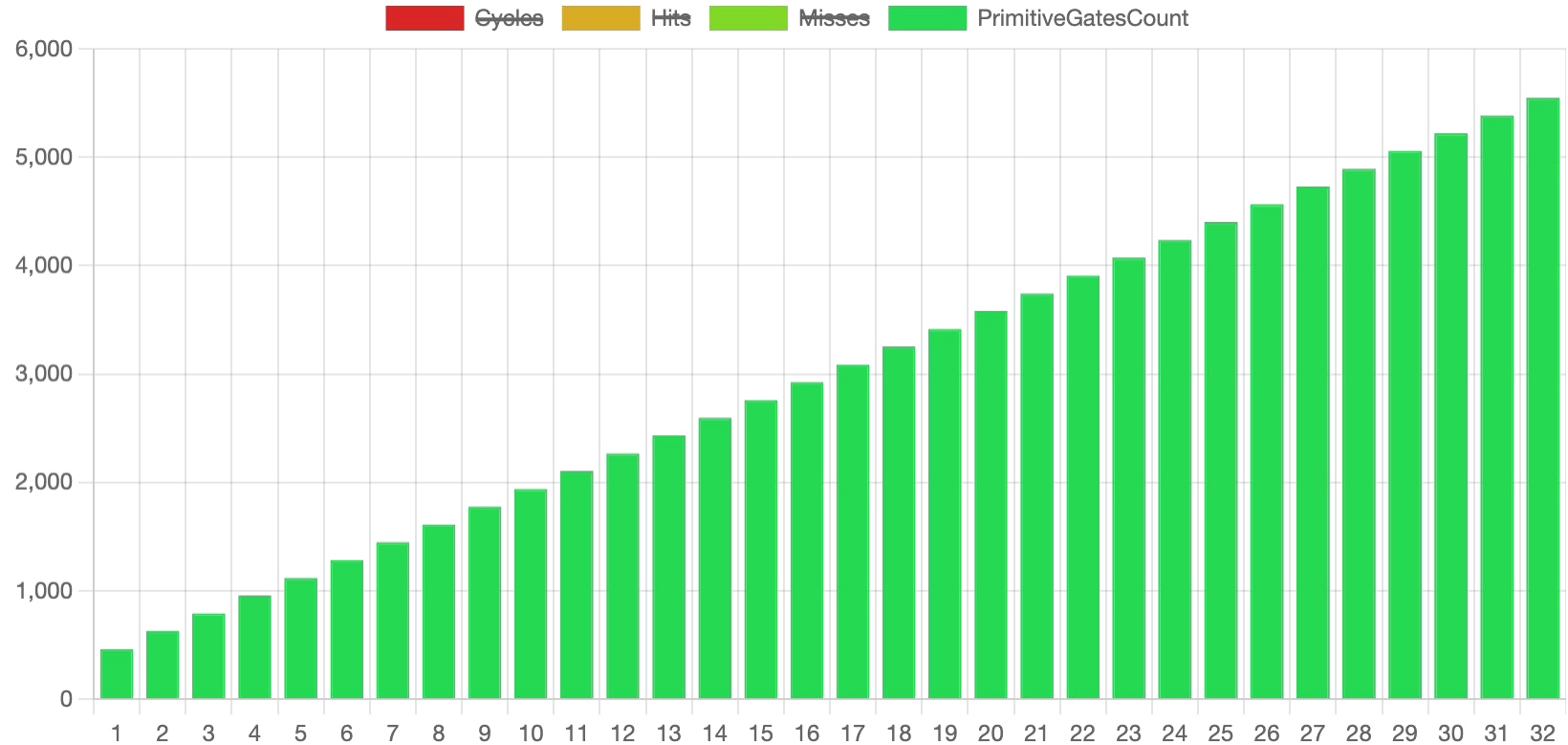
Ergebnis, wenn die Knoten auf einen Heap der Größe 256 MiB (65536 Seiten) gleichmäßig verteilt sind:

Hits	4820
Misses	1186
Ausführungszeit (Zyklen)	491940
Speedup	$720720 / 491940 = 1.46$

Schaltkreismodell



Schätzung der primitiven Gatter



Fazit

- Höhere Effizienz durch Beschränkung des TLBs auf Zweierpotenzen
- Zeitmanagement (Zyklen) durch SystemC simpel (+)
- Sehr hohes Abstraktionslevel → SystemC nicht wirklich hilfreich (-)
- Sequenzielle Abläufe umständlich (SystemC ist event-basiert) (-)
- Mit 60 Zyklen wurde eine relativ kleine Memory Latency für das Beispiel gewählt (-)

Danke für eure Aufmerksamkeit

A solid blue triangle is located in the bottom right corner of the slide, pointing towards the center.

Quellen

1 Computer Organization and Design

Patterson David A., Hennessy John L. - Computer Organization and Design: The Hardware / Software Interface (1998) - Seiten 589 - 591

2 Memory Hierarchy Reconfiguration for Energy and Performance

Balasubramonia R., Albonesi D. - Memory Hierarchy Reconfiguration for Energy and Performance in General-Purpose Processor Architectures (2000)
- Seite 10

3 Apple Silicon CPU Optimization Guide

Apple Silicon CPU Optimization Guide 3.0, 2024 - Seiten 117 - 119

4 Intel 64 and IA-32 Architectures Optimization Reference Manual

[Intel 64 and IA-32 Architectures Optimization Reference Manual, 2019](#) - Seiten 56, 79, 92 + 68, 23-6

Quellen

5 AMD Zen Reference

[Processor Programming Reference \(PPR\) for AMD Family 17h Model 01h, Revision B1 Processors, 2017](#) - Seiten 68 - 70

6 Anandtech DDR5 RAM Test

<https://www.anandtech.com/show/16143/insights-into-ddr5-subtimings-and-latencies>

7 DDR5 Latenzen

<https://www.computerbase.de/2022-06/ddr5-arbeitsspeicher-bandbreiten-latenzen/>