



**MicroSys**

**Hard- und Softwareaspekte zur Optimierung  
von embedded Systemdesigns  
CPU Plattformvergleich zur Systemleistung**

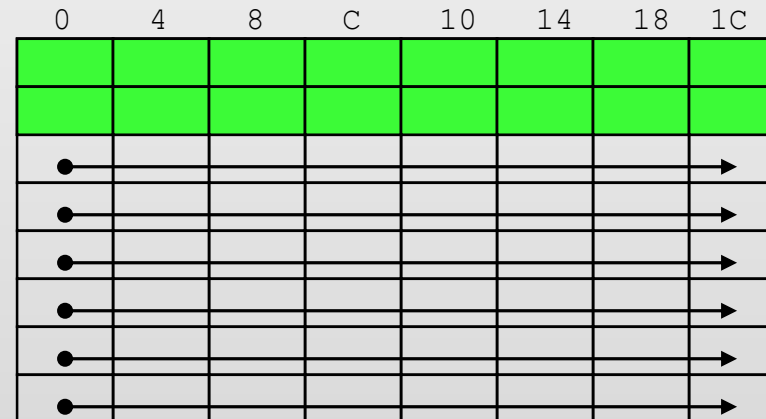
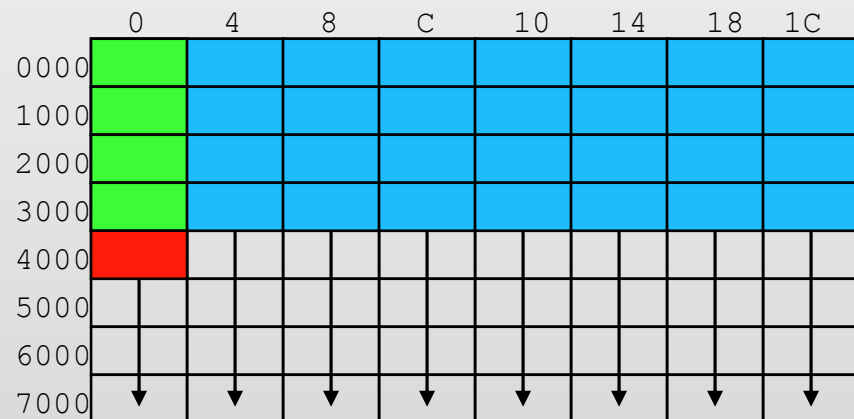
**ECC 2017**

**Kei Thomsen**

# C-Code Beispiel für „gute“ und „schlechte“ Programmierung

```
unsigned long array[1024][1024];
for (y = 0; y < 1024; y++)
    for (x = 0; x < 1024; x++)
        b += array[x][y];
```

```
unsigned long array[1024][1024];
for (y = 0; y < 1024; y++)
    for (x = 0; x < 1024; x++)
        b += array[y][x];
```



- Cache Line Fill
- Loaded 32 Bit Data
- Load 32 Bit Stall

# Ergebnis des Beispiels auf ARM, PowerPC und X86

CPU	MHz	Vertikal msec.	Vertikal MB/sec.	Horizontal msec.	Horizontal MB/sec	Faktor Vert./Hor.
ARM Cortex-A9 NXP i.MX6	800	19782	20	1209	338	<b>16</b>
ARM Cortex-A9 Xilinx ZYNQ	666	20077	20	1399	292	<b>14</b>
PowerPC QorIQ P2020	1200	41415	9	1351	303	<b>31</b>
PowerPC QorIQ T1042	1200	49304	8	1179	347	<b>42</b>
Vortex86-DX	800	15676	26	5246	78	<b>3</b>
Intel i7	2100	1480	270	276	1080	<b>5</b>

Tabelle1: 100x die C-Schleifen = 100x 4MB lesen und aufaddieren

# Ursache der Unterschiede

- Cache Zeilen Länge 32 oder 64 Byte
  - Burst Mode und Speicherbreite (16 / 32 / 64 / 128bit)
  - Größe des L1 Cache
    - Typisch 32KB
  - Memory Management Unit (MMU)
    - PowerPC: 512 Einträge a 4KB
    - Cortex A9: L1/L2 Tabelle mit 4 und 64KB Einträgen
- Bei guter Programmierung ist der Unterschied zwischen den CPUs eher klein.
- Bei schlechter Programmierung ist der Unterschied zwischen den CPUs sehr groß

# Zusammenfassung

- Performance... ja ein schnelle CPU hilft, aber der Human Factor ist nicht zu unterschätzen.
- Ein gesundes Verständnis der Technologie hilft beim Optimieren von Programmen.
- Nein, das ist nicht Sache vom Compiler! Der Compiler darf hier keine eigene Optimierung vornehmen!
  
- **Fragen?**
- Gerne auch am Stand von Elma/MicroSys