Technische Universität München
Chair of Scientific Computing
Dr. Ralf-Peter Mundani
Dr. Andreas Paul

Parallel Programming and HPC
Exercise Sheet 1: Performance Measurement

3th May 2011

1 Memory Access

a) The following loop

```
for i = 0 to N do
    a[i] = ( 1 + s ) · b[i] + t · c[i] · d[i] + e[i]
od
```

should be executed on arbitrary processor architectures. How many (machine) cycles are necessary in order to execute one full iteration if

I. a scalar processor is used which in each cycle is able to **load two words** *or* to **store one word** as well as to execute **one multiplication** *or* **one addition**,

II. a superscalar processor (MULT & ADD) is used which is able to **load** *or* to **store one word per cycle**,

III. a superscalar processor (MULT & ADD) is used which is able to **load** *or* to **store two words** per cycle?

You can assume that all scalar quantities are kept in registers during the entire program execution, write accesses (to memory) and arithmetic operations are done in parallel, and read accesses (from memory) might be executed in advance.

b) How many cycles are necessary for the following vector triad?

```
a[i] = b[i] · c[i] + d[i]
```

2 Speedup and Parallel Efficiency

The speedup S for solving a problem on p processing elements compared to one processing element can be calculated with Amdahl's law

$$S = \frac{p}{\sigma \cdot p + (1 - \sigma)}$$

where σ denotes the percentage of serial work to be done. The efficiency E of this parallel approach can be calculated with $E = \frac{S}{p}$.

Consider some parallel program with a percentage of serial work of 4%. Calculate the maximum amount p of processing elements, thus, the parallel efficiency E is at least 90%.