jian-jia.chen [☺] tu-dortmund.de
lea.schoenberger [☺] tu-dortmund.de

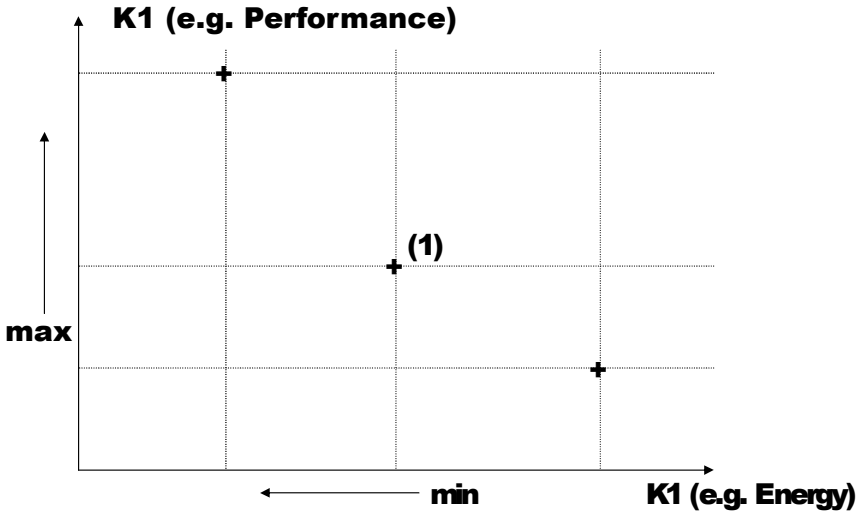
Exercise for the lecture
Embedded Systems
Wintersemester 17/18

Written Exercise Sheet 6

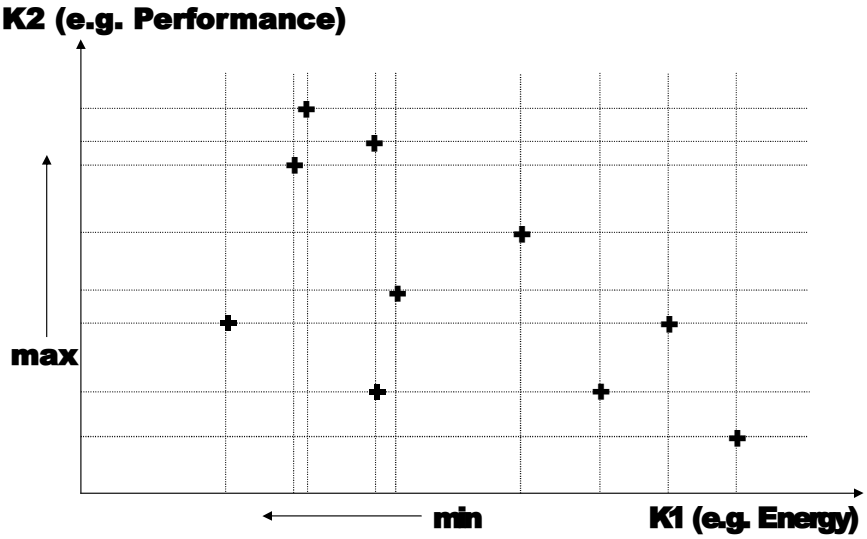
Hints: These assignments will be discussed at E23 OH14, from 10:15 am - 11:45 am on 16. Jan. 2018. You are not obligated to turn in the solutions.

1 Pareto Optimizations

- The following diagram reflects the evaluation of designs with respect to multiple criteria. We assume that we would like to maximize one criterion and minimize the other one. Please indicate the region which is dominated by design (1) (the region in which designs are “inferior” to design (1)). Also, indicate the region in which designs would dominate design (1) (the region in which designs are “superior” to design (1)).



- Suppose that design points marked “+” in the following diagram are given. Add the Pareto front to the diagram!



2 Hardware-Software Partitioning

Assume that a task can be either completely executed on an FPGA or on a microprocessor. Further assume that all tasks are independent. If a task τ_i is executed on the FPGA, it requires B_i configurable logical blocks (CLB) and has an execution time of F_i. If the task is executed on a microprocessor, its execution time is C_i. The tasks that are allocated to the microprocessor must be executed sequentially, whereas those allocated to the FPGA can be executed in parallel.

Which is the minimal size of the FPGA with respect to the number of configurable logical blocks (CLB) such that all tasks $\{\tau_1, \tau_2, \ldots, \tau_n\}$ can be finished previous to their deadlines D_i for τ_i with $x_i F_i \leq D_i$, assuming that all tasks are released and ready for execution at $t = 0$?

Formulate this hardware-software partitioning problem as an Integer Linear Program (ILP) and explain it. You are not obligated to solve the ILP.

3 Multiprocessor Partitioned Scheduling

Suppose that we are given M identical (homogeneous) processors. Each task τ_i is a periodic task with utilization U_i and period T_i. The utilization of a task τ_i is independent from which processor it is assigned to be executed. Please formulate an integer linear program to *partition* the given task set $\{\tau_1, \tau_2, \ldots, \tau_n\}$ so that a task τ_i can only and has to be assigned on one dedicated processor and the tasks can be guaranteed to meet their deadlines under *rate monotonic* scheduling. The given tasks are assumed to be independent.

Hint: You can use the Liu and Layland bound.

4 Compositional WCET

Mr. Smart has derived the WCET $4ms$ of a program on a (deterministic) platform at 1GHz CPU frequency. The system somehow is too powerful and can be slowed down to run at 500MHz CPU frequency (the rest of the platform remains the same) to reduce the power consumption. Mr. Smart concludes that the WCET under this new setting is 8ms for this program. Is that a safe estimation? Is the estimation too pessimistic? If it is pessimistic, how can he improve the estimation? If it is not pessimistic, why not?

5 May Cache and Must Cache

Suppose that we have an LRU-cache with associativity 4.

- Is it possible to have less than 4 (different) entries in the Must Cache (Abstract Interpretation)?
- Is it possible to have more than 4 (different) entries in the Must Cache (Abstract Interpretation)?
- Is it possible to have less than 4 (different) entries in the May Cache (Abstract Interpretation)?
- Is it possible to have more than 4 (different) entries in the May Cache (Abstract Interpretation)?

Explain your answer.

6 Cache Analysis

Consider the following program:

```
read a;
read b;
read a;
if (a>b) {
  read c;
  read d;
} else {
  read e;
  read f;
}
read x;
read a;
```

(a) Perform May Cache and Must Cache analyses on this program, assuming an LRU (least-recently used) replacement policy with associativity 4 that is empty at the start of the program. Is it possible to determine whether the last access to “a” results in a cache hit or a cache miss?

(b) We now assume that the cache uses an FIFO (first-in-first-out) replacement policy instead. Could a May Cache and Must Cache analyse determine whether the last access to “a” results in a cache hit or a cache miss if the cache is empty at the start of the program?

7 WCET for Systems with Scratchpad Memory

Explain the differences of WCET analysis when considering scratch pad memory (SPM) and Cache memory.