technische universität
dortmund

fakultät für informatik
informatik 12

Evaluation and Validation

Peter Marwedel
TU Dortmund, Informatik 12
Germany

(2010年 12 月 05 日)
Subset of slides selected for EECS 222C.

These slides use Microsoft clip arts.
Microsoft copyright restrictions apply.

Graphics: © Alexandra Nolte, Gesine Marwedel, 2003

Validation and Evaluation

Definition: *Validation* is the process of checking whether or not a certain (possibly partial) design is appropriate for its purpose, meets all constraints and will perform as expected (yes/no decision).

Definition: Validation with mathematical rigor is called *(formal) verification*.

Definition: *Evaluation* is the process of computing quantitative information of some key characteristics of a certain (possibly partial) design.

technische universität
dortmund

fakultät für
informatik

© p. marwedel,
informatik 12, 2010

- 2 -

1

## Evaluation of designs according to multiple objectives

Different design objectives/criteria are relevant:
- Average performance
- Worst case performance
- Energy/power consumption
- Thermal behavior
- Reliability
- Electromagnetic compatibility
- Numeric precision
- Testability
- Cost
- Weight, robustness, usability, extendibility, security, safety, environmental friendliness

---

Performance evaluation

- **Estimated performance values:**
  Difficult to generate sufficiently precise estimates;
  Balance between run-time and precision

  π x
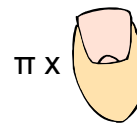
- **Accurate performance values:**
  As precise as the input data is.

We need to compute **average** and **worst case** execution times

Worst/best case execution times (1)

Def.: The **worst case execution time** (WCET) is an **upper bound** on the execution times of tasks.
The term is not ideal, since a program requiring the WCET for its execution does not have to exist (WCET is a **bound**).

Def.: The **best case execution time** (BCET) is a lower **bound** on the execution times of tasks.
The term is not ideal, since a program running at the BCET for its execution does not have to exist (BCET is a **bound**).

Slide added for EECS 222C from 2006/7 slides by P. Marwedel.

Worst/best case execution times
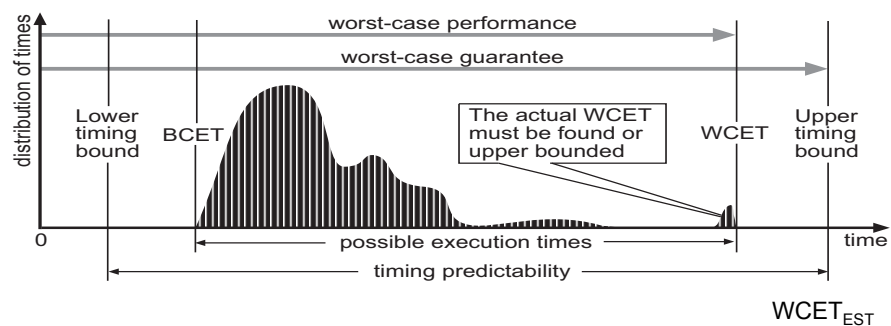
Requirements on WCET estimates:
- *Safeness:* $WCET \leq WCET_{EST}$!
- *Tightness:* $WCET_{EST} - WCET \rightarrow$ minimal

Worst case execution times (2)

Complexity:
- in the general case: undecidable if a bound exists.
- for restricted programs: simple for "old" architectures, very complex for new architectures with pipelines, caches, interrupts, virtual memory, etc.

Approaches:
- for hardware: requires detailed timing behavior
- for software: requires availability of machine programs; complex analysis (see, e.g., www.absint.de)

4