



Minimizing System Modification in an Incremental Design Approach

Pop, Paul; Eles, Petru; Pop, Traian; Peng, Zebo

Published in:

Ninth International Symposium on Hardware/Software Codesign. CODES 2001 (IEEE Cat. No.01TH8571)

Link to article, DOI:

[10.1109/HSC.2001.924672](https://doi.org/10.1109/HSC.2001.924672)

Publication date:

2001

Document Version

Publisher's PDF, also known as Version of record

[Link back to DTU Orbit](#)

Citation (APA):

Pop, P., Eles, P., Pop, T., & Peng, Z. (2001). Minimizing System Modification in an Incremental Design Approach. In *Ninth International Symposium on Hardware/Software Codesign. CODES 2001 (IEEE Cat. No.01TH8571)* (pp. 183-188). <https://doi.org/10.1109/HSC.2001.924672>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

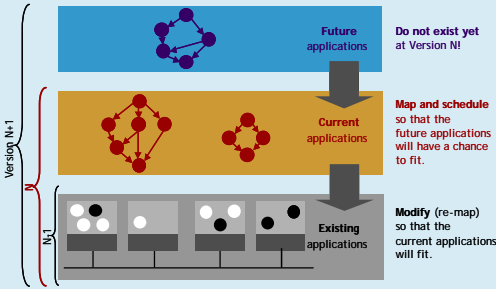
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Minimizing System Modification in an Incremental Design Approach

Paul Pop, Petru Eles, Traian Pop, Zebo Peng

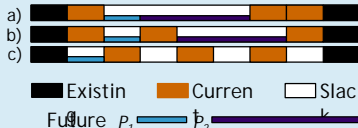
Department of Computer and Information Science, Linköpings universitet, Sweden

Incremental Design

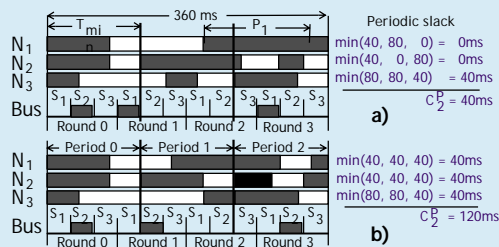


- Start from an already existing system with applications
- Implement new functionality on this system Mapping and Scheduling
- To reduce design and testing time: As few as possible modifications of the existing applications
- After the new functionality has been implemented: It should be easy to add functionality in the future

Design Criteria



- First criterion: C_1^P, C_1^m**
How well the resulted slack sizes accommodate a future application



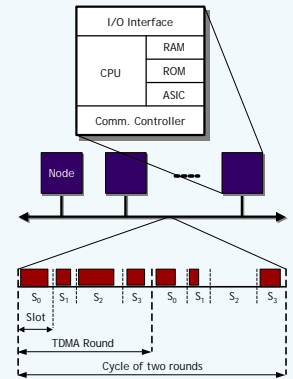
- Second criterion: C_2^P, C_2^m**
How well the slack is distributed in time to accommodate T_{min} , t_{need} and b_{need}

Paul Pop, Petru Eles, Traian Pop, Zebo Peng:

An approach to Incremental Design of Distributed Embedded Systems, Design Automation Conference, 2001

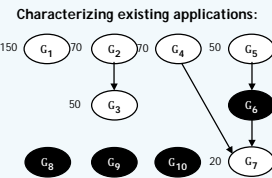
Summary

- Mapping and scheduling of distributed embedded systems for hard-real time applications
 - Static cyclic scheduling of processes and messages,
 - Bus access scheme: time-division multiple-access.
- Incremental design process
 - Already existing system,
 - Implement new functionality,
 - a) Existing system modified as little as possible,
 - b) new functionality can be easily added to the system.
- Mapping strategy
 - a) Subset selection to minimize modification time,
 - b) Two design criteria, objective function.



Problem Formulation

- Input**
 - A set of *existing* applications.
 - A *current* application to be mapped.
 - The system architecture.
- Output**
 - A **mapping and scheduling of the *current* application**, so that the incremental design requirements are satisfied.
- Requirements**
 - a) constraints of the *current* application are satisfied and minimal modifications are performed to the *existing* applications.
 - b) new *future* applications can be mapped on the resulted system.



- Characterizing future applications:**
 - Typical WCETs
 - Typical message sizes
 - Smallest expected period T_{min}
 - Expected necessary processor time t_{need}
 - Expected necessary bandwidth b_{need}

Mapping Strategy

- Initial mapping and scheduling
- Requirement a) **Subset selection problem**
Select that subset Ω of existing applications so that the current application fits and the modification cost $R(\Omega)$ is minimized:

$$R(\Omega) = \sum_{i \in \Omega} R_i$$

Three approaches to the subset selection problem

- Exhaustive Search (ES)
- Ad-Hoc Solution (AH)
- Subset Selection Heuristic (SH)

- Requirement b) Objective function minimization:

$$C = w_1^P (C_1^P) + w_1^m (C_1^m) + w_2^P \max(0, t_{need} - C_2^P) + w_2^m \max(0, b_{need} - C_2^m)$$

