Bus Access Optimization for Distributed Embedded Systems Based on Schedulability Analysis

Paul Pop, Petru Eles, Zebo Peng
Department of Computer and Information Science
Linköping University
Sweden
Outline

- Motivation
- System Architecture
- Schedulability Analysis
- Communication Synthesis
- Experimental Results
- Conclusions
Motivation and Characteristics

- Embedded System Design.
  Scheduling, Communication, Bus Access.

- Characteristics:
  - Heterogeneous system architecture.
  - Fixed priority preemptive scheduling for processes.
  - Communications using the time-triggered protocol (TPP).

Contributions and Message

Contributions:
- Proposed a schedulability analysis for distributed hard-real time systems that use the time-triggered protocol.
- Developed optimization strategies for the communication synthesis problem.

Message:
- By optimizing the bus access scheme the “degree of schedulability” of the system can be significantly improved.
Event-Triggered vs. Time-Triggered

- **Event-triggered**: activation of processes and transmission of messages is done at the occurrence of significant events.
- **Time-triggered**: activation of processes and transmission of messages is done at predefined points in time.

<table>
<thead>
<tr>
<th></th>
<th>Processes</th>
<th>Messages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Event-triggered</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Time-triggered</td>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>
Hardware Architecture

- Hard real-time distributed systems.
- Nodes interconnected by a broadcast communication channel.
- Nodes consisting of: TTP controller, CPU, RAM, ROM, I/O interface, (maybe) ASIC.
- Communication between nodes is based on the time-triggered protocol.

Bus access scheme: time-division multiple-access (TDMA).

Schedule table located in each TTP controller: message descriptor list (MEDL).
Software Architecture

- Real-Time Kernel running on the CPU in each node.
- The worst case administrative overheads are known.
- Fixed priority preemptive scheduling.
- Tick scheduler in each kernel.

![Diagram of Software Architecture]

- CPUN0
- TTP Controller
- CPU
- N1
- RTK
- m0
- m1
- m2
- m3
- S0
- S1
- m0
- m1
- m2
- event-triggered
- time-triggered

Bus Access Optimization for Distributed Embedded Systems Based on Schedulability Analysis
Paul Pop, Petru Eles, Zebo Peng
March 30, 2000
Problem Formulation

Input
- An application modelled as a set of processes.
- Each process has an execution time, a period, a deadline, and a priority.
- The system architecture and mapping of processes to nodes are given.
- Each message has a known size.

Output
- A schedulability analysis (response time analysis) for hard real-time systems that use the time-triggered protocol for communications.
- The MEDL for the TTP controllers so that the process set is schedulable on an as cheap (slow) as possible processor set.
Scheduling of Messages over TTP

- Messages are dynamically produced by the processes.
- Frames are statically determined by the MEDL.

1. Single message per frame, allocated statically:
   Static Single Message Allocation (SM)

2. Several messages per frame, allocated statically:
   Static Multiple Message Allocation (MM)

3. Several messages per frame, allocated dynamically:
   Dynamic Message Allocation (DM)

4. Several messages per frame, split into packets, allocated dynamically:
   Dynamic Packets Allocation (DP)
Schedulability Analysis

- **Starting point**: schedulability analysis for distributed hard real-time systems with communication based on a simple TDMA protocol.
  

- Schedulability test: \( \text{response time } r_i \leq \text{deadline } D_i \) for each process.

- The **response time** \( r_i \) depends on the **communication delay** between sending and receiving a message.

- The **communication delay** is calculated differently for each of the four approaches to message scheduling over TTP (SM, MM, DM, DP):
Schedulability Analysis (Continued)

- The **communication delay** for a message \( m \) depends on:

1. Static Single Message Allocation (SM): \( T_{m_{\text{max}}} \)
2. Static Multiple Message Allocation (MM): \( T_{m_{\text{max}}} \)

Optimizing Buss Access (SM and DM)

Process P2 misses its deadline!

Swapping m1 with m2: all processes meet their deadlines.

Putting m1 and m2 in the same slot: all processes meet their deadline, the communication delays are reduced.
Optimization Strategy

- The synthesis of the MEDL is performed off-line: optimization process.

- Comparison of the four messages scheduling approaches: fair only for near-optimal results.

- Optimization strategies based on Greedy Approaches and Simulated Annealing.

- Cost function: degree of schedulability.
Experimental Results

Average percentage deviation from the best among the four message scheduling approaches:
Experimental Results (Continued)

The quality of the greedy optimization heuristics:

![Average percentage deviations (in %)](chart)

- **OptimizeSM**
- **OptimizeMM**
- **OptimizeDM**
- **OptimizeDP**
Conclusions

- Static priority preemptive process scheduling. Communications based on a time-triggered protocol.

- Four different message scheduling policies over TTP:
  - analysis of the communication delays and
  - optimization strategies for the buss access scheme.

- Approaches compared using extensive experiments:
  - guidelines for designers.

- Optimizing the buss access scheme the “degree of schedulability” of the system can be significantly improved.