

DISCO – DIStributed Embeddable Systems for COntrl Applications: Project Overview

J.A. Fonseca, A. Mota, P. Fonseca, L. Almeida, E. Martins



**Departamento de Electrónica e Telecomunicações
IEETA – Instituto Engenharia Electrónica e Telemática de Aveiro
Universidade de Aveiro
Aveiro - PORTUGAL**

WTR'2001 - Workshop de Tempo Real, Florianópolis, Brasil

22 May 2001



OBJECTIVES

- **To achieve simpler and more flexible solutions for distributed embedded systems (DESs).**
- **To adapt DESs to control applications.**
- **To improve flexibility and adaptability in order to reduce operation and maintenance costs.**
- **To maximize the utilization of system resources.**



Main Project Topics

- **Flexible control requirements**
- **Scheduling and dispatching techniques**
 - **Protocols**
 - **Hardware based solutions**
- **Global systems management**



Target Systems

- **Distributed Embedded Systems**
- **Fieldbus as the connection between nodes**
- **Low processing power microprocessors or microcontrollers as the nodes CPUs**
- **Special emphasis on CAN (Controller Area Network)**
- **Emphasis on hard real-time applications (cars, mobile robots, machine tools, ...)**



Flexible Control Requirements (1)

- **QoS-based analysis of control requirements.**
- **Establishment of operational ranges for control parameters (sampling period, sampling and actuation jitter, ...).**
- **Evaluation of the impact of network induced sampling jitter in system identification techniques.**
- **Distribution of control tasks:**
 - Partition of control operations in sub-tasks.
 - Synchronization of sub-tasks.



Flexible Control Requirements (2)

First work on the topic:

- **Influence in system identification performance of jitter induced by the specific MAC of CAN.**

Conditions:

- **Message delays under traffic load based on the PSA benchmark.**

Some of the results:

- **Jitter taken into consideration as a fractional dead time leads to a much better model identification** (compared with when it is not considered).
- **Improvements: 2.3 to 7700** (square errors between the actual system and the identified model);
 - **some immunity to high transmission loads** (independently of the messages priority).



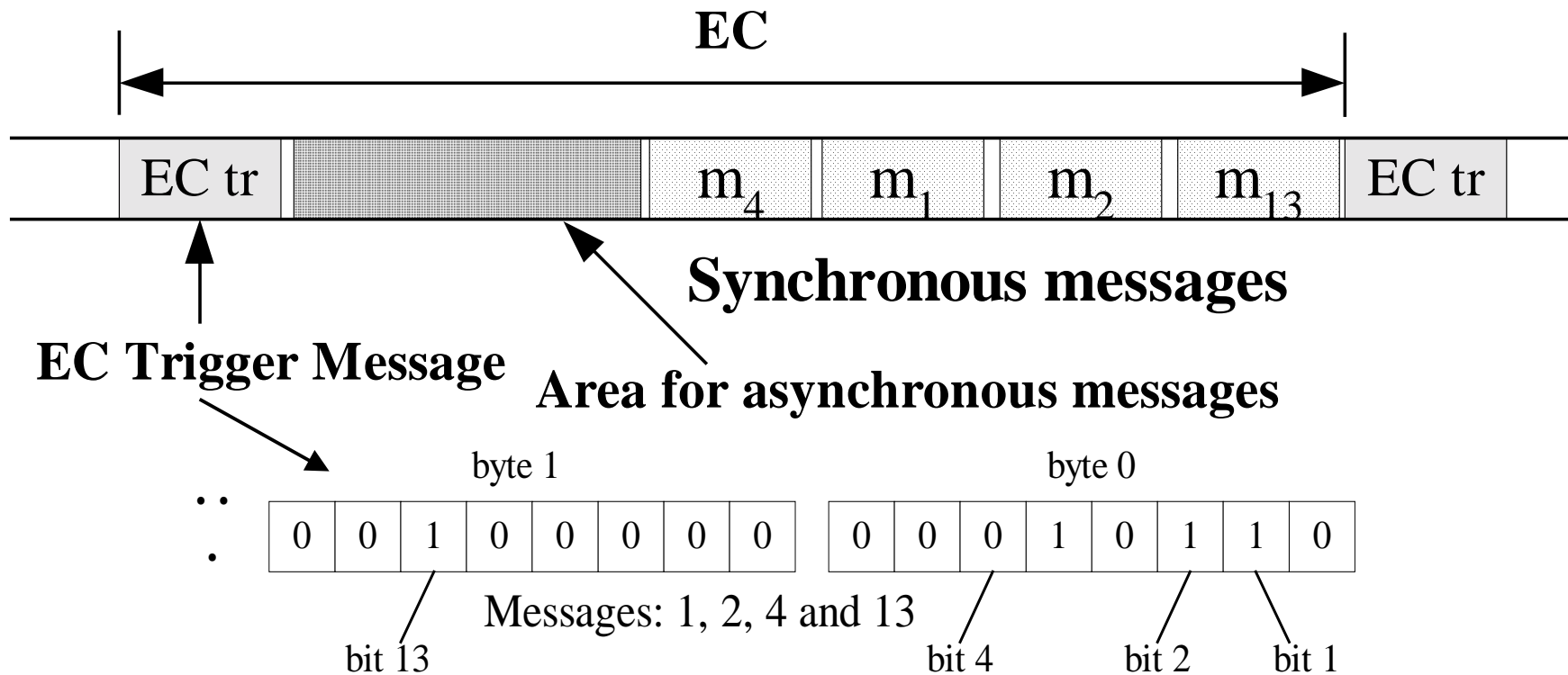
Scheduling and dispatching techniques (1)

- **The planning scheduler (1997)**
 - Flexibility without losing real-time behavior
- **FTT-CAN (1998/9 - ...)**
 - Flexible Time-Triggered communication for CAN
 - Autonomously controlled communication system
 - Allows on-line reconfiguration under guaranteed system timeliness
 - Supports both synchronous and asynchronous messages
 - Synchronous messages guaranteed to meet deadlines by on-line schedulability analysis.
 - Asynchronous messages transmitted according to a best-effort approach.



Scheduling and dispatching techniques (2)

FTT-CAN





Scheduling and dispatching techniques (3)

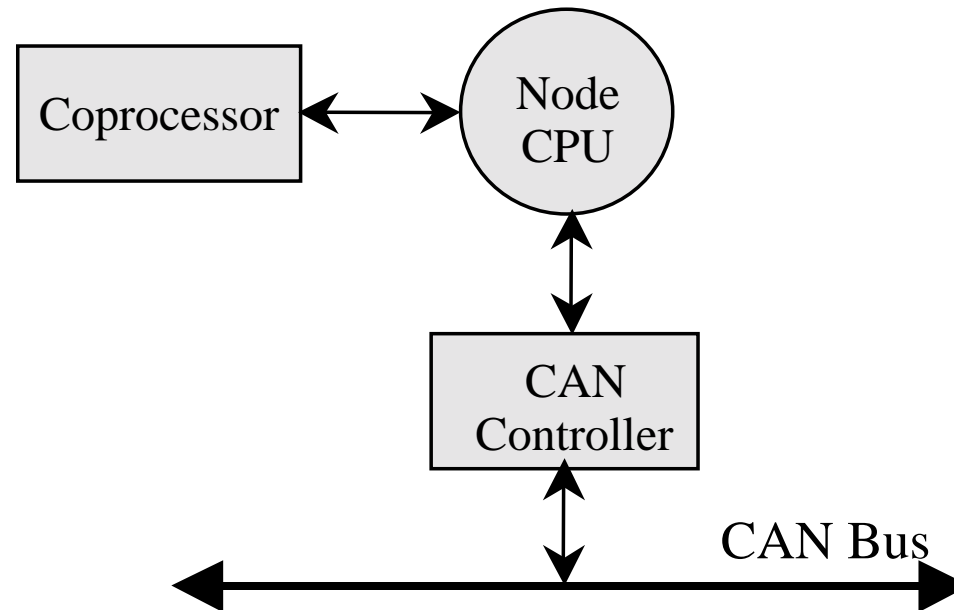
- **Hardware-based solutions**

- Custom processors based on FPGAs (Field-Programmable Gate Arrays) to implement scheduling algorithms.
- Improvement of scheduling execution time.
- Changes needed in just one of the system nodes.
- **First approach: adapted to the planning scheduler and FTT-CAN.**
- **First prototype (February 2001):**
 - Xilinx XC4010 + 8051 μ C.
 - Set of 8 periodic variables, fixed priority.
 - Plan with 16 time units (Ecs).
 - Worst case execution time of 63 μ s (@ 12MHz), against 200msec for a 8051 μ C.



Scheduling and dispatching techniques (4)

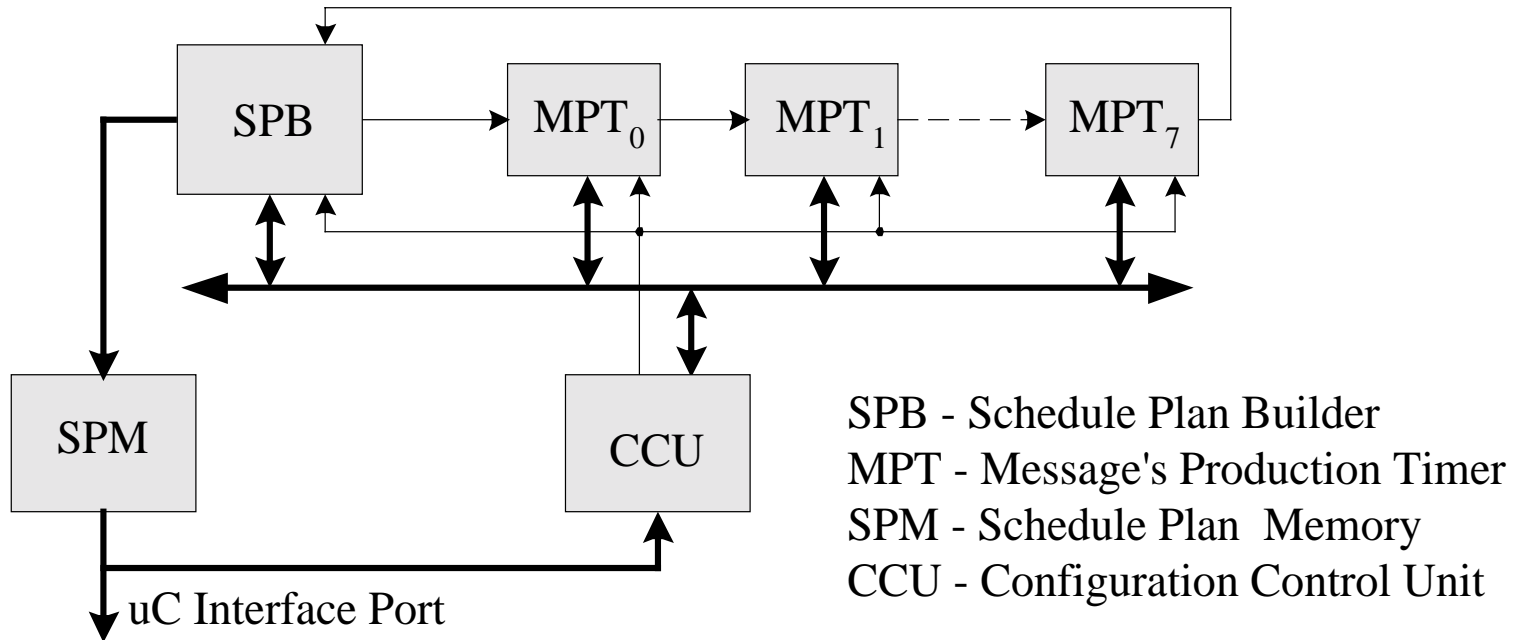
- **PSCoP – Planning Scheduler Co-processor**





Scheduling and dispatching techniques (5)

- **PSCoP – Planning Scheduler Co-processor**





Summary of DISCO project current results

- **Reduction of the influence of network-induced jitter in control systems.**
- **Validation of FTT-CAN (time-triggered protocol for CAN):**
 - **Hard-real time embedded distributed systems**
 - **Adequate flexibility and responsiveness for on-line change of system parameters.**
- **Hardware co-processors for scheduling:**
 - **Quasi-dynamic scheduling (automatic change of parameters).**
 - **Integration of other operations: analysis, error recovery.**



Work-in-progress

- **Performance evaluation of different control algorithms (under various jitter and period conditions).**
- **Specification of parameter ranges (different QoS from the control point of view).**
- **Dependability issues in FTT-CAN.**
 - **Error contention and recovery.**
- **New versions of co-processors**
 - **Including schedulability analysis**
 - **Multiple scheduling policies**
- **Global management mechanisms**
 - **Global admission control of (dynamic) system requirements.**
 - **Network-centric perspective.**
 - **Joint schedulability analysis of tasks in nodes and fieldbus messages.**



Expected final results

- **More flexible and simpler distributed systems for embedded applications.**
- **Based in off-the-shelf components and common fieldbus networks.**
- **Improved reactivity.**
- **Simplification of the global system management.**



Conclusions

- **Automation and control systems can benefit from the improvement of distributed embedded systems.**
- **It seems that a QoS-based operation can coexist with hard real-time guarantees.**
- **Adequate dynamic behavior can be obtained with simple protocols and available off-the-shelf programmable circuits.**
- **Consequences:**
 - **Improvement in composability**
(also expected in dependability)
 - **Simplification of the global system management (setup, maintenance, upgrade, “plug-and-play”).**