

ARTIST2 Workshop on Requirements for Flexible Scheduling in Complex Embedded Systems

Real-time networks perspective

Prepared by: Paulo Pedreiras, Ricardo Marau, Luís Almeida

DET – IEETA
Universidade de Aveiro
Aveiro-Portugal



Background

Nowadays, current **complex embedded systems** are **distributed** (DES)

- ✓ Cars, planes, industrial machinery ...

There is also a trend to **increase integration** among subsystems as a way to

- ✓ **Improve efficiency** in using systems resources
- ✓ **Reduce** number of active **components** and **costs**
- ✓ **Manage complexity**

Leading to a **strong impact of the network** on the global system properties:

- ✓ Composability, timeliness, flexibility, dependability...

Approaches

Safety concerns have typically led to **static** approaches in the design of DES

- ✓ **Fault-tolerance** mechanisms become **simpler**
- ✓ Proliferation of **static Time-Triggered** architectures using **TDMA** with pre-allocated slots
(**TTP, TT-CAN, FlexRay, SAFEbus, SwiftNet**)

However, static approaches:

- ✓ Tend to be **inefficient** in the use of system resources
- ✓ Do not easily accommodate **changes** in the **operational environment** or **system configuration**

Moreover

There is a growing interest in using DES in **dynamic operational scenarios**:

- ✓ Systems with **variable number of users**, either humans or not (traffic control, radar...)
- ✓ Systems that operate in **changing physical environments** (robots, cars...)
- ✓ Systems that can **self-reconfigure dynamically** to cope with hazardous events or evolving functionality (cars, planes, ...)

QoS adaptation, graceful degradation, survivability

Network requirements

Dynamic (flexible) management of bandwidth while guaranteeing both real-time and safety constraints.

- ✓ Act upon **periodic communication**, e.g. related to **control information** (potentially bandwidth consuming)
- ✓ **Adapt transmission rates** according to **effective needs**
- ✓ Explore subsystems that **operate occasionally**
- ✓ Explore **variable sampling/tx rates** according to the current system **control stability state**

But

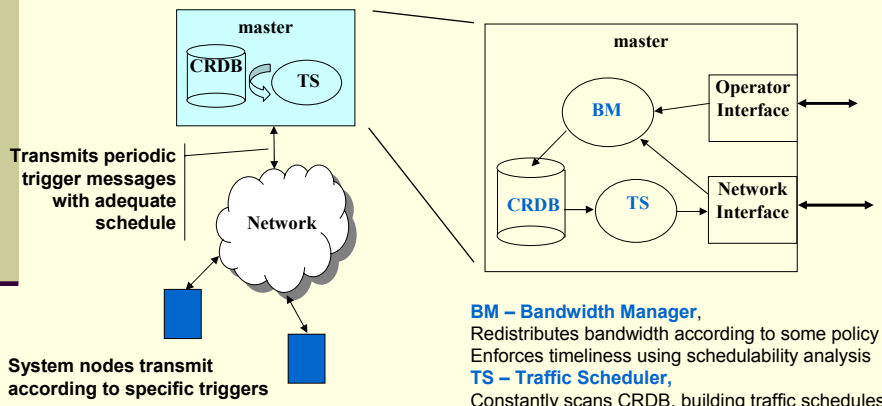
- ✓ How to implement such level of **flexibility** without jeopardizing **timeliness** and **safety**?

Paris (Massy), 16 June 2006

5

Possible architecture

Master-slave paradigm, for flexibility control



Paris (Massy), 16 June 2006

6

Implementation

This architecture is the basis of the
FTT (Flexible Time-Triggered) architecture

Three protocols have already been developed
according to this architecture

- ✓ **FTT-CAN** and **FTT-Ethernet/ FTT-SE**
 - ✓ **Efficient master/multi- slave implementation**
 - ✓ **Efficient combination of sync(TT)/asnc(ET) traffic**