CSC9006 Real-Time Systems

Lecture: The Time-Triggered Architecture

Chao Wang

Department of Computer Science and Information Engineering

June 10th, 2021

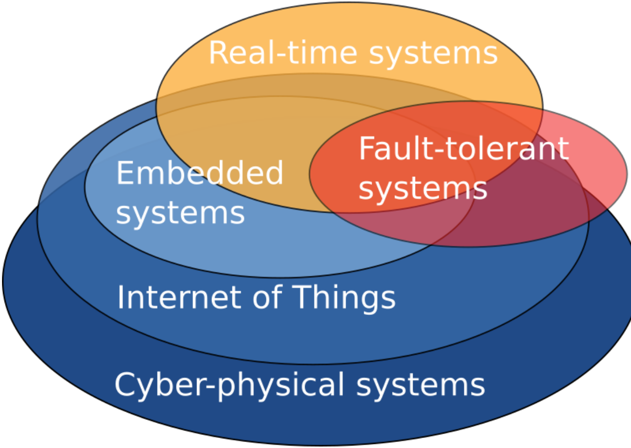
NATIONAL TAIWAN NORMAL UNIVERSITY

Lecture Outline

References

- Chapter 14 in the following textbook
  - Kopetz, Hermann. Real-Time Systems: Design Principles for Distributed Embedded Applications. Springer; 2nd ed. 2011 edition.
- Obermaisser, Roman & Kopetz, Hermann & Kuster, Sibylle & Huber, Bernhard & Salloum, Christian & Zafalon, Roberto & Auzanneau, Fabrice & Gherman, Valentin & Kronlof, Klaus & Waris, Heikki & Cristau, Gérard & Edelin, Gilbert & Millet, Philippe & Borth, Michael & Couvreuer, Chantal & Suri, Neeraj & Bokor, Peter & Dobre, Dan & Serafini, Marco & Hiller, Martin. (2009). GENESYS: A Candidate for an ARTEMIS Cross-Domain Reference Architecture for Embedded Systems.

The Big Picture (my view)



Introduction

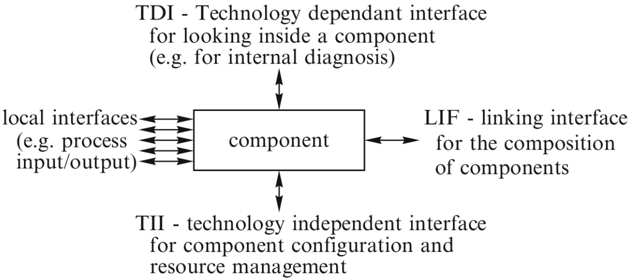
- The time-triggered architecture (TTA) for embedded systems
  - A framework that integrates many concepts covered in this course
- Objectives of TTA: dependable and robust
- Key principles of TTA
  - Complexity management
  - Recursive component concept
  - Coherent communication
- Services of TTA
  - Core system services
  - Optional system services
  - Application specific services

Complexity Management

- Simplicity is an explicit concern in architecturing embedded systems
- A consistent, global time base may greatly simplify the architecture:
  - consistent temporal order of all events
  - deterministic behavior
  - conflict-free time-controlled communication channels
  - detection of a message loss
  - latency measurement

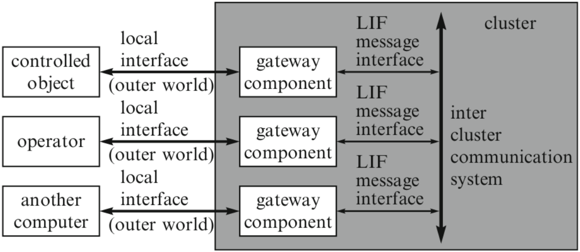
Component Orientation

- Components are primitive structure elements of TTA
- Each component is a fault-containment unit (FCU)
- One benefit of time-triggered communication: phase alignment
  - component-to-component latency can be bound
  - end-to-end latency of a transaction can be bound if ___



Component Orientation (cont.)

- A component can be expanded to a cluster
  - Cluster LIF (internal to the cluster) vs. external LIF
- a gateway component bridges components/clusters/environment
- Recursive component concept: change of viewpoints
  - as a cluster (the cluster LIF of the gateway component)
  - as a single component (the external LIF of the gateway component)



Coherent Communication

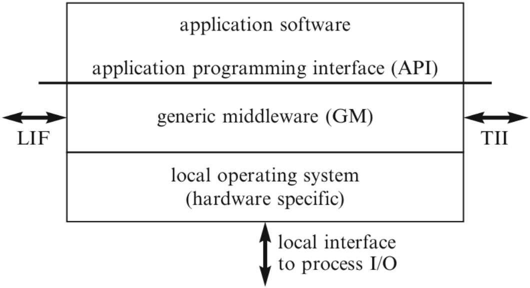
- The fate-sharing model
  - "... it is acceptable to lose the state information associated with an entity if, at the same time, the entity itself is lost." – David Clark, an architect of the DARPA net
- The only communicaction mechanism in TTA
  - Unidirectional BMTS: Basic Message Transport Service
- Other benefits of time-triggering (compared to event-triggering):
  - constant transport delay
  - a minimum jitter of one granule of the global time

How Do All These Relate to *Dependability*?

- A component is a fault-containment unit (FCU)
  - why?
- The BMTS performs multicasting, thus reducing the probe effects
- The BMTS has deterministic behaviors
- The recursive component concept helps in system evolution
- In addition, in TTA
  - components publish their ground state (g-state) periodically
  - global time is available

Services of TTA

- Component-based services
  - Core system services
  - Optional system services
  - Application specific services
- A generic middleware may include some of the core system services and of the optional system services



Core System Services

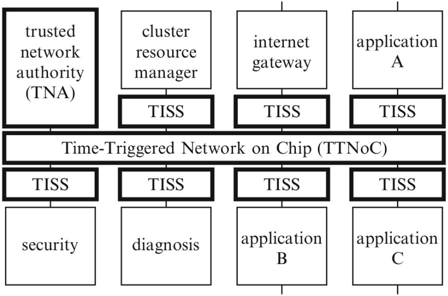
- The absolutely essential set of services
  - for the maintenance of the desired architecture properties
  - for the building of higher-level services
- Examples
  - Basic time service
  - Basic configuration service
  - Basic execution control service
  - basic communication service

Optional System Services

- A well-defined supportive functionality encapsuled in a self-contained system component
- Useful across many application domains
- Examples
    - Diagnostic services
    - Security services
    - Gateway services

Putting Everything Together: The Time-Triggered MPSoC

- MPSoC: Multiprocessor systems on chip
- TISS: Trusted interface subsystems

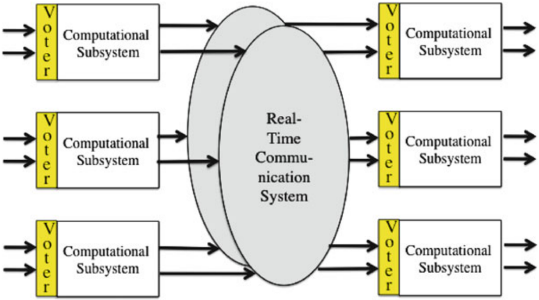


Dependability

- Section 1.4 in the textbook
- The notion of dependability covers the meta-functional attributes of a computer system that relate to the quality of service a system delivers to its users during an extended interval of time.
- Dependability is measured with respect to the following attributes:
  - reliability
  - safety
  - maintainability
  - availability
  - security

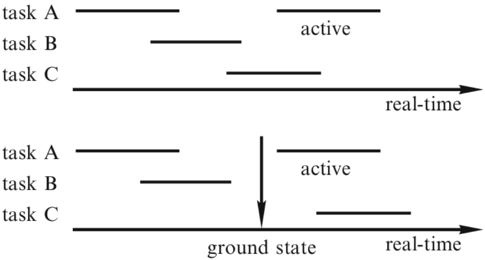
FCU vs. FTU

- Sections 6.4.1 and 6.4.2 in the textbook
- FCU: Fault-containment unit
- FTU: Fault-tolerant unit
  - FTU masks the failure of a single FCU inside the FTU



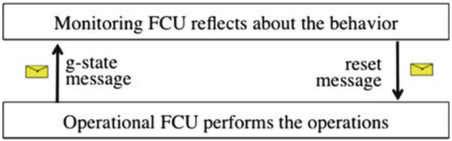
The Ground State (G-State)

- Sections 4.2.3 and 6.6 in the textbook
- Need for periodic reintegration instants
- G-state = the state at the reintegration instant
- Ground cycle = the interval between two reintegration instants



Robustness

- Section 6.5 in the textbook
- A system is said to be robust if the severity of the consequences of a fault is inversely proportional to the probability of fault occurrence, i.e., faults that are expected to occur frequently should have only a minor effect on the quality of service of the system.
- Design for robustness is concerned with the fast restoration of the normal system service after a fault has occurred.
- The two-channel approach for a robust system:



BMTS

- Basic message transport service, defined in Section 7.2 in the textbook
- The BMTS multicasts a message with a high reliability, a small delay, and minimal jitter; we say that BMTS is at the waistline of a communication model
- According to the temporal properties of a given BMTS, we may classify messages into three types:
  - event-triggered (also, review Section 4.3.3)
  - rate-constrained
  - time-triggered (also, review Section 4.3.4)