

Objectives:

- Create a generic “recipe book” for embedded network safety cases
- Apply to specially designed safety-critical embedded networks
- Apply to application-specific messaging over off-the-shelf networks



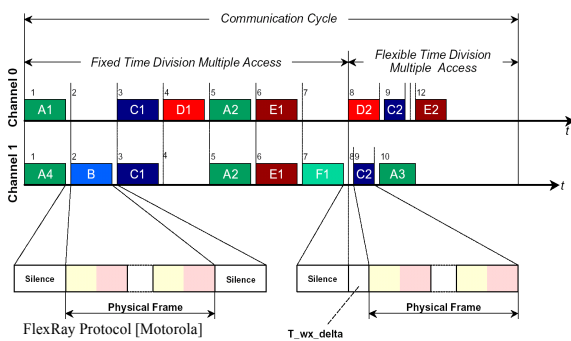
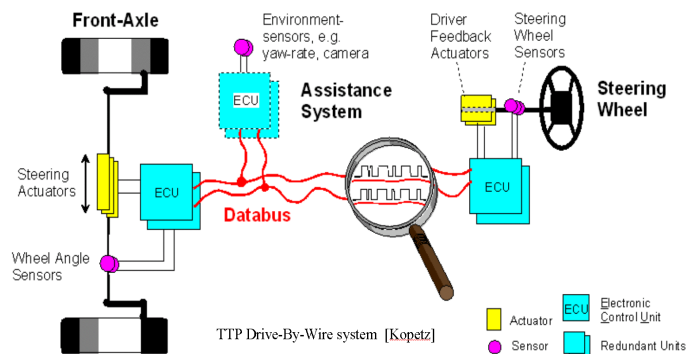
Challenges:



- Lightweight mechanisms for safe operation (FlexRay)
- Use of commodity networks (Ethernet; WiFi) in critical applications

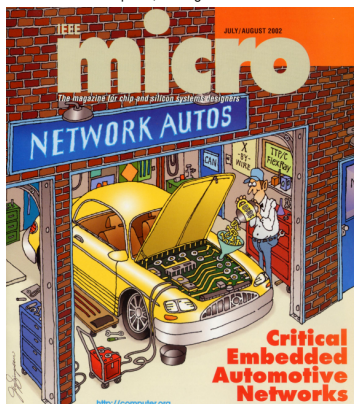
Strategy:

- Efficient group membership
 - New paradigm: Group by period, not physical node
 - Cost of guarantees (availability, bandwidth)
 - Appropriate fault identification and tolerance (e.g., transient vs. permanent)
- End-to-end approach to sources of errors
 - Data values & real time delays both matter
 - Network interface
 - Network transmission errors
 - Intermediate network routing/forwarding errors



- Apply to protocols of interest
 - FlexRay (Bosch; GM)
 - Flexiblok rail safety system (Bombardier Transportation)
 - TCN (rail application reference protocol)
 - TTP (automotive application reference protocol)
 - CAN (automotive application non-critical reference protocol)
- Create generic approach to understanding protocols
 - Mechanisms to provide basic safety building blocks (should be in hardware/firmware)
 - Policies to manage mechanisms (should be in software)

Guest Editor: P. Koopman, Carnegie Mellon



Expected Outcomes:



- Methodology for creating protocol safety cases
 - End-to-end data transmission errors – assumes correct design
 - Complements formal methods approaches that check design correctness
- Tradeoffs: cost/performance points with quantifiable safety levels
- Answers to pressing protocol questions:
 - What policies and approaches are required to achieve safe FlexRay operation?
 - Provide an independent check to Flexiblok safety case

