# Deterministic and Flexible Communication for Real-Time Embedded Systems


Supervisor
prof. Riccardo Sisto
dott. Ivan Cibrario Bertolotti

Candidate
Tingting Hu

Department of Control and Computer Engineering (DAUIN)
Politecnico di Torino


2015/02/27

# Outline

- Introduction
  - Controller Area Network (CAN)
- Deterministic Communication
  - 8B9B, VHCC, ZSC
- Flexible Communication
  - IP over CAN
  - Modbus CAN
- Conclusion

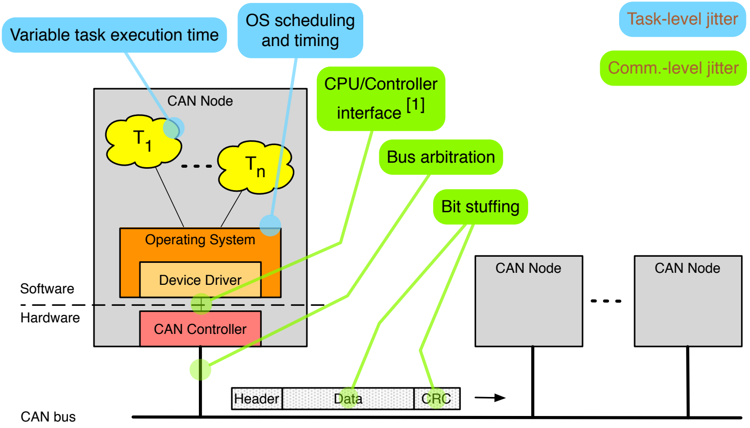
# Introduction

As embedded systems evolve from centralized to distributed architecture, communication becomes more and more important.

- Controller Area Network (CAN) is a real-time communication network
- It is the de facto standard in automotive and gained popularity in networked embedded control systems recently.
- Determinism is an important feature of real-time embedded systems:
  - Delay: time taken to complete a certain task
  - Jitter: variability in delay

Jitter impairs determinism, and worsens the real-time performance

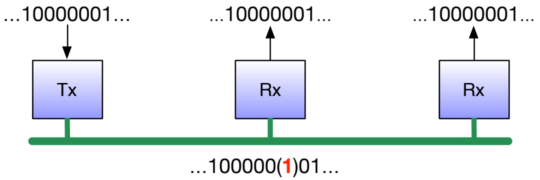
# Sources of jitter in CAN-based distributed system




[1] G. Cena, I. Cibrario Bertolotti, T. Hu, and A. Valenzano. "Performance Evaluation and Improvement of the CPU–CAN Controller Interface for Low-Jitter Communication". *Proc. IEEE ETFA*.

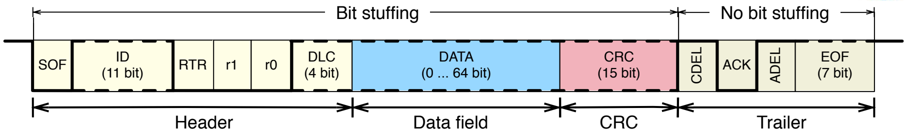
# Bit stuffing mechanism

At the physical layer, CAN relies on bit stuffing (BS) for receiver synchronization




- # of stuff bits depends not only on the frame length, but also on its content ⇒ Jitter in communication.
- For real-time systems with tight timing constraints, ~20% of system-wide jitter.
- BS interferes with the CRC-based error detection mechanism in CAN and jeopardizes data integrity severely.

# State of the art




- **Header**: fixed & known in advance ⇒ no communication jitter
- **Data field**: variable from message to message
- **CRC**: depends on both the header and the data field;
  calculated by hardware, at run time

- Existing approaches just prevent BS jitter in the data field, by either scrambling or encoding the payload in a way that less or no stuff bits will be added by the CAN controller during transmission.
- NO approaches available for the CRC.

# BS prevention mechanisms

ZSD and ZSC prevent BS from the data field and the CRC, respectively

- Zero Stuff-bits Data (ZSD)
  - Fixed-length payload encoding: 8B9B[2][3]
  - Variable-length payload encoding: VHCC
- Zero Stuff-bits CRC (ZSC)
- ZSD and ZSC are compatible with each other

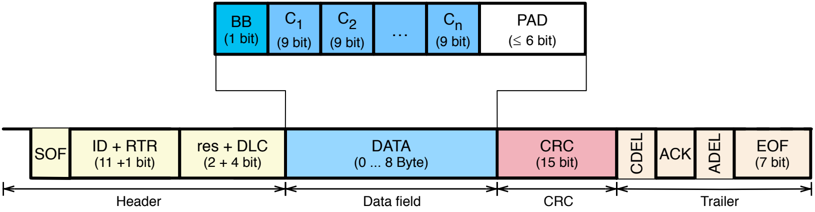
[2] G. Cena, I. Cibrario Bertolotti, T. Hu, and A. Valenzano. “Fixed-Length Payload Encoding for Low-Jitter Controller Area Network Communication”. *IEEE Trans. Ind. Inf.*

[3] G. Cena, I. Cibrario Bertolotti, T. Hu, and A. Valenzano. “Un codec a basso jitter per reti CAN”. *Automazione e strumentazione*.

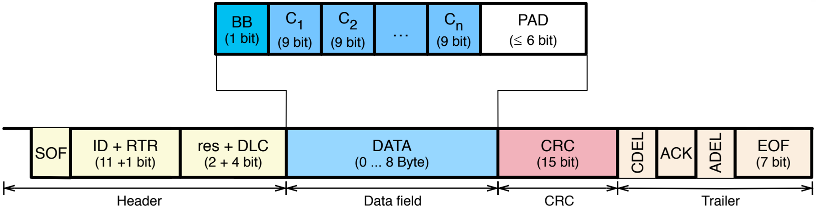
# 8B9B

Every byte of the original payload is translated into a distinct 9-bit codeword.

- Codebook property 1)
  - $\leq$ 4 consecutive bits at the same value within the 9-bit codeword
  - $\leq$ 2 consecutive bits at the same value at the beginning & end
- Padding is needed at the end of the data field
- Achieve better encoding efficiency than existing approaches.



# VHCC[4]




Instead of padding, pack sub-byte application information

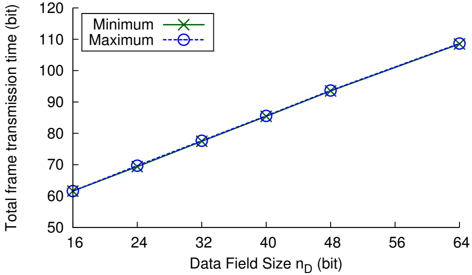
- Codebook property 2)
  - Nested codebooks: 8B9B codebook can also be used for any N bit to N+1 bit encoding, N $\in$ [1, 8].
  - Nested codebooks still preserve property 1)

[4] G. Cena, I. Cibrario Bertolotti, T. Hu, and A. Valenzano. "On a family of run length limited, block decodable codes to prevent payload-induced jitter in Controller Area Networks". *Computer Standards & Interfaces*.

# Zero Stuff-bits CRC (ZSC)

Instead of padding, use it to prevent BS in the CRC.

- Exploiting only 3 bits at the end of the data field, it is always possible to tune the CRC calculation to a value that is BS free
- independent from the payload content and the encoding scheme.
- ZSD + ZSC leads to deterministic communication[5]:




[5] G. Cena, I. Cibrario Bertolotti, T. Hu, and A. Valenzano. "A Mechanism to Prevent Stuff Bits in CAN for Achieving Jitterless Communication". *IEEE Trans. Ind. Inf.*

# Main achievements

- Theoretical/experimental results show that 8B9B and VHCC achieve better computational and communication efficiency[6]
- Highly optimized and portable codec was developed for dissimilar embedded platforms.
- Residual error probability decreases by about two orders of magnitude[7]
- ZSC: an Italian patent[8] application was submitted and a European extension is in progress

[6] G. Cena, I. Cibrario Bertolotti, T. Hu, and A. Valenzano. “Performance Comparison of Mechanisms to Reduce Bit Stuffing Jitters in Controller Area Networks”. *Proc. IEEE ETFA*.

[7] G. Cena, I. Cibrario Bertolotti, T. Hu, and A. Valenzano. “Effect of jitter-reducing encoders on CAN error detection mechanisms”. *Proc. IEEE WFCS*.

[8] G. Cena, I. Cibrario Bertolotti, T. Hu, and A. Valenzano. “Limitazione del bit stuffing in una trama di comunicazione di un segnale elettronico”. Italian patent application.

# Flexible communication

Application-level protocols for CAN are mainly automotive oriented.

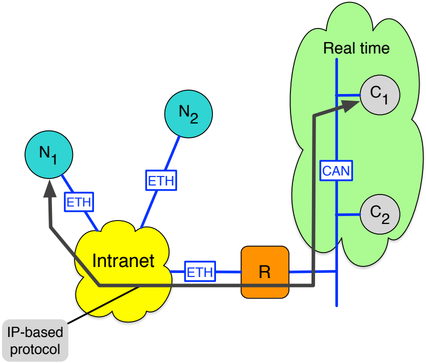
- SAE J1939 and ISO 11783 for communication/diagnostics among in-vehicle components.
- CANopen and DeviceNet for industrial automation
- ARINC 825 for local subsystem communication in civil aviation

Extend the flexibility of CAN by broadening the set of high-level protocols supported on it

- General-purpose protocol (IP)
- Special purpose protocol (Modbus)

# General-purpose protocol support

Internet Protocol (IP): the most widespread protocol, enormous software available




Goal:

- Integrate CAN (at the field level) into Intranet
- Support non real-time activities like remote configuration, firmware update ...

Existing approaches:

- Protocol translation

IP over CAN: permits IP datagrams to be transmitted on CAN

# IP over CAN[9]

When integrating different subsystems, coexistence among them is always the main concern

- Interference to real-time performance (due to IP traffic):
  - IP messages can be modeled as real-time messages with the lowest priority
  - Worst-case jitter is bounded.
- Non real-time performance is comparable to what can be achieved on a pure CAN link (in absence of RT traffic):

| $d$ (B) | 1 | 4 | 16 | 64 | 256 | 1024 |
|---|---|---|---|---|---|---|
| $r_{C\to C}$ (kB/s) | 0.78 | 2.91 | 8.90 | 18.02 | 24.76 | 26.08 |
| $r_{C\to E}$ (kB/s) | 0.65 | 2.49 | 7.97 | 17.80 | 24.07 | 26.43 |
| $r_{E\to C}$ (kB/s) | 25.18 | 26.12 | 26.36 | 26.50 | 26.46 | 26.44 |

Table : Mean data transfer rate $r$ for 10 MB data vs data chunk size $d$.

[9] G. Cena, I. Cibrario Bertolotti, T. Hu, and A. Valenzano. "Seamless Integration of CAN in Intranets". *Computer Standards & Interfaces*.

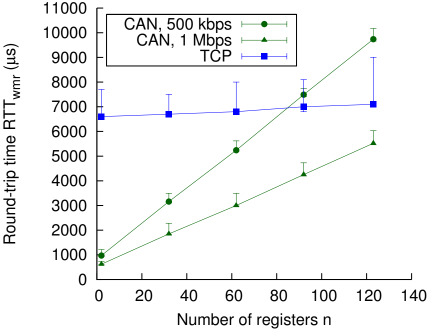
# Special purpose protocol support

## Modbus

An application-level real-time communication protocol, which is commonly used in building automation.

- Existing support for Modbus:
  - RS485: obsolete, extremely slow (19200 bps)
  - Ethernet: popular, extra cabling and intermediate devices
- CAN is a good compromise in terms of both link speed (1 Mb/s) and cost (bus architecture)
- Modbus CAN: enables Modbus traffic to be transmitted on the CAN bus

# Modbus CAN[10]




- Modbus CAN (mean RTT and jitter) outperforms Modbus TCP on a 100 Mb/s Ethernet, with a 1 Mb/s CAN bus.
- Break-even point (for mean RTT) at $n \simeq 80 \ldots 100$ registers when using a 500 kb/s CAN bus.

A compromise between link speed and software processing overhead

[10] G. Cena, I. Cibrario Bertolotti, T. Hu, and A. Valenzano. "Design, verification, and performance of a MODBUS-CAN adaptation layer". *Proc. IEEE WFCS*.

# Conclusion

- Research work was carried out focusing on the determinism and flexibility of CAN communication
- BS jitter is completely prevented all over the frame by ZSD + ZSC
- The application scenario of CAN was largely broadened by the design and implementation of IP over CAN and Modbus CAN
- Modbus CAN is adopted by industry for local subsystem communication and it paves the way of CAN in building automation.

# List of Publications (1/5)

## 2014


[1] I. Cibrario Bertolotti and T. Hu,
*Modular Design of an Open-Source, Networked Embedded System*,
*Computer Standards & Interfaces*, vol. 37, pp. 41–52, Jan. 2015. Available online since June 2014.

[2] G. Cena, I. Cibrario Bertolotti, T. Hu, and A. Valenzano,
*A Mechanism to Prevent Stuff Bits in CAN for Achieving Jitterless Communication*,
*IEEE Transactions on Industrial Informatics*, vol. 11, no. 1, pp. 83-93, Feb. 2015. Available online since Nov. 2014.

[3] T. Hu and I. Cibrario Bertolotti,
*Model Checking*,
in *Cary R. Spitzer, Thomas Ferrell, and Uma Ferrell (eds.), Digital Avionics Handbook*, 3rd edition, ISBN 9781439868614. CRC Press, Taylor and Francis, USA, chap. 42, pp. 42.1-42.20, 2014.

[4] A. Ballarino et al.,
*System-Level Performance of an Automation Solution Based on Industry Standards*,
*Proc. 19th IEEE Conference on Emerging Technologies and Factory Automation (ETFA)*, pp. 1–6, Sep. 2014.

# List of Publications (2/5)


[5] G. Cena, I. Cibrario Bertolotti, T. Hu, and A. Valenzano,
*Effect of Jitter-Reducing Encoders on CAN Error Detection Mechanisms*,
*Proc. 10th IEEE International Workshop on Factory Communication Systems (WFCS)*,
pp. 1–10, May 2014.

[6] G. Cena, I. Cibrario Bertolotti, T. Hu, and A. Valenzano,
*Design, Verification, and Performance of a Modbus–CAN Adaptation Layer*,
*Proc. 10th IEEE International Workshop on Factory Communication Systems (WFCS)*,
pp. 1–10, May 2014.

[7] G. Cena, I. Cibrario Bertolotti, T. Hu, and A. Valenzano,
*Un codec a basso jitter per reti CAN*,
*Automazione & strumentazione*, vol. 62, no. 3, pp. 76–79, Apr. 2014.

[8] G. Cena, I. Cibrario Bertolotti, T. Hu, and A. Valenzano,
*Limitazione del bit stuffing in una trama di comunicazione di un segnale elettronico*,
Italian patent application no. TO2014A000132, Feb. 2014. European extension in progress.

# List of Publications (3/5)


## 2013

[9] G. Cena, I. Cibrario Bertolotti, T. Hu, and A. Valenzano,
*Fixed-Length Payload Encoding for Low-Jitter Controller Area Network Communication*,
*IEEE Transactions on Industrial Informatics*, vol. 9, no. 4, pp. 2155–2164, Nov. 2013.

[10] I. Cibrario Bertolotti, L. Durante, T. Hu, and A. Valenzano,
*A Model for the Analysis of Security Policies in Industrial Networks*,
*Proc. 1st International Symposium for Industrial Control System & SCADA Cyber Security Research*, pp. 66–77, Sep. 2013.

[11] G. Cena, I. Cibrario Bertolotti, T. Hu, and A. Valenzano,
*Software-Based Assessment of the Synchronization and Error Handling Behavior of a Real CAN Controller*,
*Proc. 18th IEEE Conference on Emerging Technologies and Factory Automation (ETFA)*, pp. 1–9, Sep. 2013.

[12] G. Cena, I. Cibrario Bertolotti, T. Hu, and A. Valenzano,
*On a Family of Run Length Limited, Block Decodable Codes to Prevent Payload-Induced Jitter in Controller Area Networks*,
*Computer Standards & Interfaces*, vol. 35, no. 5, pp. 536–548, Sep. 2013.

# List of Publications (4/5)

## 2012

[13] I. Cibrario Bertolotti, L. Durante, T. Hu, and A. Valenzano,
*A Unified Class Model for Checking Security Policies in ICT Infrastructures*,
*Proc. 1st IEEE AESS European Conference on Satellite Telecommunications (ESTEL)*,
pp. 1–6, Oct. 2012.

[14] G. Cena, I. Cibrario Bertolotti, T. Hu, and A. Valenzano,
*Performance evaluation and improvement of the CPU–CAN controller interface for low-jitter communication*,
*Proc. 17th IEEE Conference on Emerging Technologies and Factory Automation (ETFA)*,
pp. 1–8, Sep. 2012.

[15] G. Cena, I. Cibrario Bertolotti, T. Hu, and A. Valenzano,
*Performance comparison of mechanisms to reduce bit stuffing jitters in controller area networks*,
*Proc. 17th IEEE Conference on Emerging Technologies and Factory Automation (ETFA)*,
pp. 1–8, Sep. 2012.

[16] G. Cena, I. Cibrario Bertolotti, and T. Hu,
*Formal Verification of a Distributed Master Election Protocol*,
*Proc. 9th IEEE International Workshop on Factory Communication Systems (WFCS)*,
pp. 245–254, May 2012.

# List of Publications (5/5)

## 2011


[17] I. Cibrario Bertolotti and T. Hu,
*Analisi di una pila protocollare open source per sistemi in tempo reale a basso costo*,
*Atti Congresso Nazionale AICA*, pp. 1–10, Nov. 2011.

[18] I. Cibrario Bertolotti and T. Hu,
*Real-Time Performance of an Open-Source Protocol Stack for Low-Cost, Embedded Systems*,
*Proc. 16th IEEE Conference on Emerging Technologies and Factory Automation (ETFA)*, pp. 1–8, Sep. 2011.

THANK YOU FOR YOUR ATTENTION