CAN FD
CAN with Flexible Data-rate

Automotive Electronics
Robert Bosch GmbH, Reutlingen
CAN FD - CAN with Flexible Data-rate

- Sensors & Embedded Control
- MM / Driver Assistance

Data-rate [bit/s]
- 100M
- 10M
- 1M
- 100k
- 10k

Implementation costs per node

CAN
- CAN FD

BroadR Reach Broadcom
MCST 150

Automotive Electronics

AE/EIN | 10/10/2012 | © Robert Bosch GmbH 2012. All rights reserved, also regarding any disposal, exploitation, reproduction, editing, distribution, as well as in the event of applications for industrial property rights.
Motivation

- Increasing demand for bandwidth in automotive communication
- Close gap between CAN (max. 1 MBit/s) and FlexRay (10 MBit/s)
- Time-triggered communication not flexible enough
- High effort for migration to FlexRay / Ethernet
  - Hardware costs
  - Software changes

→ Make CAN faster!
CAN FD - CAN with Flexible Data-rate

Speeding up CAN

Unchanged

- CAN arbitration
- CAN acknowledge mechanism

New

- switch to higher bit rate for transmission of
  - Data Length Code
  - Data Field
  - Frame CRC
- data fields with more than eight bytes possible
  - configured by unused DLC codes “1001” to “1111”
  - 12, 16, 20, 24, 32, 48, 64 bytes
- new CRC polynomials for longer data fields, HD=6
  - 17 bit: up to 16 byte data fields, 21 bit: up to 64 byte data fields
CAN FD - CAN with Flexible Data-rate

Speeding up CAN

- Based on existing CAN
  - well known technology, minimized risk
  - **changes limited to HW**: protocol controller
    - for bit rates up 1 Mbit/s standard CAN transceivers usable
  - **no changes to SW**: with 8 bytes data field (legacy SW fully compatible)
    - even higher data rate possible by data fields >8 bytes and SW change

- Costs similar to CAN
Average CAN FD Bit Rate

Frame ID: 11 bit, Bit Rate Arbitration: 1Mbit

- 64 bytes payload
- 48 bytes payload
- 32 bytes payload
- 16 bytes payload
- 8 bytes payload

Average Bit Rate (Mbit/s) vs. Bit Rate Data Phase (Mbit/s)
Net CAN FD Bit Rate

Frame ID: 11 bit, Bit Rate Arbitration: 1Mbit

- 64 bytes payload
- 48 bytes payload
- 32 bytes payload
- 16 bytes payload
- 8 bytes payload
CAN FD - CAN with Flexible Data-rate

CAN FD with Bit Rates below 1 MBit/s

→ Application:
→ Long bus line limits bit rate for arbitration
→ Arbitrate with e.g. 125 kbit/s
→ Transmission of data field with e.g 500 kbit/s

No requalification or redesign of transceivers necessary!

→ Option:
→ Transmission of long frames with up 64 bytes payload
→ Increase of net data rate
→ Handling of large data packets w/o segmentation
Synchronous Switching of Bit Rate

Example: CAN_0 wins arbitration at reserved/SRR bit

- Phase-shift before synchronization: \(1 \leftrightarrow 2\) \(-350\) ns
- Phase-shift after synchronization: \(3 \rightarrow 4\) \(+433\) ns
- Phase-shift at beginning of Data Phase: \(5 \rightarrow 6\) \(+433\) ns
CAN FD - CAN with Flexible Data-rate

CAN FD Standard Frame

- CAN FD Arbitration Phase
  - length: 30 bit times*
  - data rate: max. 1 MBit/s
- CAN FD Data Phase
  - length: 86 bit times* (8 data bytes)
  - data rate: > 1 MBit/s
- Remote Frames always in CAN Format
  - RTR bit replaced by reserved bit r1
  - r1 takes part in CAN arbitration
  - reserved for protocol expansion

* bit stuffing not considered

EDL – Extended Data Length
Substitutes first reserved bit in standard frames
EDL = recessive indicates CAN FD frame format (new DLC-coding and CRC)
EDL = dominant indicates standard CAN frame format

r1, r0 – reserved bits
Transmitted dominant, reserved for future protocol variants

BRS – Bit Rate Switch
BRS = recessive: switch to alternate bit rate
BRS = dominant: do not switch bit rate

ESI – Error State Indicator
ESI = recessive: transmitting node is error passive
ESI = dominant: transmitting node is error active

* 17 bit CRC for data fields with up to 16 bytes
CAN FD - CAN with Flexible Data-rate

CAN FD Extended Frame

<table>
<thead>
<tr>
<th>Arbitration Field</th>
<th>Control Field</th>
<th>Data Field</th>
<th>CRC Field</th>
<th>ACK</th>
<th>EOF</th>
<th>Int.</th>
<th>Bus Idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>11 bit Identifier</td>
<td>18 bit Identifier Extension</td>
<td>4 bit DLC</td>
<td>0-64 bytes</td>
<td>21* bit CRC</td>
<td>1 1</td>
<td>7</td>
<td>3</td>
</tr>
</tbody>
</table>

- CAN FD Arbitration Phase
  - length: 49 bit times*
  - data rate: max. 1 MBit/s
- CAN FD Data Phase
  - length: 86 bit times* (8 data bytes)
  - data rate: > 1 MBit/s
- Remote Frames always in CAN Format
  - RTR bit replaced by reserved bit r1
  - r1 takes part in CAN arbitration
  - reserved for protocol expansion

* 17 bit CRC for data fields with up to 16 bytes

** EDL – Extended Data Length **
Substitutes first reserved bit in standard frames
EDL = recessive indicates CAN FD frame format (new DLC-coding and CRC)
EDL = dominant indicates standard CAN frame format

** r1, r0 – reserved bits **
Transmitted dominant, reserved for future protocol variants

** BRS – Bit Rate Switch **
BRS = recessive: switch to alternate bit rate
BRS = dominant: do not switch bit rate

** ESI – Error State Indicator **
ESI = recessive: transmitting node is error passive
ESI = dominant: transmitting node is error active

* bit stuffing not considered
Structure of CAN FD Nodes

- **Physical Layer**
  - unchanged for CAN FD
  - same CAN transceivers

- **CAN FD Protocol Controller**
  - two sets of timing configuration

- **BTL and BRP control bit timing**
  - BTL and BRP switch between two sets

- **BSP controls frame (de)coding**
  - BSP defines Arbitration and Data Phase

- **CAN Message Handling**
  - shift register as (de)serializer
  - BSP does not limit data field length

### Diagram Description
- **CAN Bus**
- **Transceiver**
- **System Clock**
- **Bit Timing Logic**
- **Bit Stream Processor**
- **Shift Register**
- **CAN Message**
- **Physical Layer**
- **Protocol Controller**
- **Bit Time Configuration (Arbitration Phase & Data Phase)**
- **Baud Rate Prescaler**
- **Scaled Clock (tq)**
- **Tx-Rx-Loop-Delay**
- **Sync Mode**
  - **Tx Bit**
  - **Sampled Bit**
  - **Sample Point**
  - **Data Phase**
  - **CAN_H**
  - **CAN_L**
  - **CAN_Tx**
  - **CAN_Rx**
- **CAN_Rx**
- **CAN_Tx**
Example: (Bit rate in Data Phase) = 4 · (Bit rate in Arbitration Phase)

- Different length of time quanta, no Prop_Seg in Data Phase
- BRS bit with timing of Arbitration Phase until Sample Point
- CRC Delimiter with timing of Data Phase until Sample Point
CAN FD - CAN with Flexible Data-rate

CAN FD Demonstrator

<table>
<thead>
<tr>
<th>CAN FD Communication</th>
<th>CAN FD Network</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arbitration Phase</td>
<td>Bus Line</td>
</tr>
<tr>
<td>1 Mbit/s</td>
<td>42 m</td>
</tr>
<tr>
<td>Data Phase</td>
<td>CAN FD nodes</td>
</tr>
<tr>
<td>12 Mbit/s</td>
<td>7</td>
</tr>
<tr>
<td>Data Field</td>
<td>CAN Transceiver</td>
</tr>
<tr>
<td>64 Byte</td>
<td>NXP TJA 1040</td>
</tr>
</tbody>
</table>
Transceiver Delay and Bus Line Delay

CAN FD Communication
- Arbitration Phase: 1 Mbit/s
- Data Phase: 15 Mbit/s
- Bus Line: 42 m

Delays in the CAN FD Network measured at edge from BRS to ESI
- Transceiver Loop Delay: 126 ns
- Bus Line Delay: 163 ns
Transceiver Delay Compensation

- Transceiver Loop Delay is measured for each frame at the falling edge of bit EDL
  - Delay compensation independent of transceiver characteristics

- Transceiver delay measured in system clock periods
- Configurable offset added to adjust Secondary Sample Point SSP inside bit time
  - SSP position rounded down to next integer number of time quanta $t_q$
- Delayed transmit data compared against received data at SSP
  - check for bit errors
# Oscillator Tolerance – Rules

## Rules Arbitration Phase

<table>
<thead>
<tr>
<th>Rule</th>
<th>Description</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule 1</td>
<td>Resynchronization</td>
<td>$df &lt; \frac{sjw_A}{2 \cdot 10bt_A}$</td>
</tr>
<tr>
<td>Rule 2</td>
<td>Sampling of Bit after Error Flag</td>
<td>$df &lt; \frac{\min(pb1_A, pb2_A)}{2 \cdot [13bt_A - pb2_A]}$</td>
</tr>
</tbody>
</table>

## Rules Data Phase (when Bit Rate is switched)

<table>
<thead>
<tr>
<th>Rule</th>
<th>Description</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule 3</td>
<td>Resynchronization</td>
<td>$df &lt; \frac{sjw_D}{2 \cdot 10bt_D}$</td>
</tr>
<tr>
<td>Rule 4</td>
<td>Sampling of Bit after Error Flag</td>
<td>$df &lt; \frac{\min(pb1_A, pb2_A)}{2 \cdot \left[(6bt_D - pb2_D) \cdot \frac{BRP_D}{BRP_A} + 7bt_A\right]}$</td>
</tr>
<tr>
<td>Rule 5</td>
<td>Bit Rate Switch</td>
<td>$df &lt; \frac{sjw_D \left(\frac{BRP_A}{BRP_D} - 1\right)}{2 \cdot \left[2bt_A - pb2_A\right] \cdot \frac{BRP_A}{BRP_D} + pb2_D + 4bt_D}$</td>
</tr>
</tbody>
</table>
**CAN FD - CAN with Flexible Data-rate**

**Lab Validation – Exemplary Results**

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ISO Bus Topology</td>
<td>2x120Ω</td>
<td>42m</td>
<td>15,0 Mbit/s</td>
<td><img src="image1.png" alt="Graph" /></td>
</tr>
<tr>
<td>Passive Star</td>
<td>1x 60Ω</td>
<td>16m</td>
<td>3,5 Mbit/s</td>
<td><img src="image2.png" alt="Graph" /></td>
</tr>
<tr>
<td>Passive Star</td>
<td>2x120Ω</td>
<td>16m</td>
<td>7,5 Mbit/s</td>
<td><img src="image3.png" alt="Graph" /></td>
</tr>
</tbody>
</table>

Term. resistor CAN FD node

*CAN Transceiver: NXP TJA 1040
Conclusion

- Gradual introduction of CAN FD nodes into CAN networks possible
- Bosch CAN IPs currently upgraded to CAN FD (M_CAN, C_CAN)
  - will be integrated into μCs, first samples E2012
  - upgrade of Bosch VHDL Reference CAN Model to CAN FD
  - integration on μCs of other major manufacturers planned
- Tool support for CAN FD started
  - Vector and ETAS integrate CAN FD into their tool chain
- Development of vehicle demonstrator with CAN FD network
  - Joint project with automotive and semiconductor companies started
- Check EMC issues
  - Emission, Susceptibility / Immunity
- Impact on different network topologies under investigation
  - line, star, position of termination, different transceivers
- Standardization as ISO 11898-7
- Assure support in Higher Layer Protocols (e.g. AUTOSAR, CANopen)
Backup
**CAN FD - CAN with Flexible Data-rate**

**Physical Layer**

- **CAN Physical Layer**
  - Transceiver loop delay $\text{CAN}_\text{Tx} \rightarrow \text{CAN}_\text{Rx}$: up to 240ns
  - delay on CAN bus line: $\sim 5\text{ns/m}$

- **CAN FD Arbitration Phase**: arbitrate with remote nodes
  - Limitation: $\text{delay}[A \rightarrow B] + \text{delay}[B \rightarrow A] < TSEG1^*$

- **CAN FD Data Phase**: transceiver delay compensation for bit monitoring
  - Limitation: filter characteristics of input comparator and bus topology
  - independent of length of CAN bus line

---

$^*$TSEG1 = Time Segment before Sample Point
Remote Frames and CAN FD

- There are no remote frames in CAN FD format
  - RTR bit is replaced by dominant reserved bit r1
  - The reserved bit r1 takes part in CAN bit arbitration
  - Bit r1 reserved for future protocol expansions, e.g. using r1 as additional identifier bit
  - Receivers ignore the actual value of bits r1, r0 in CAN FD frames

- CAN FD controllers are able to handle remote frames in standard CAN format
CAN FD - CAN with Flexible Data-rate

Lab Validation – Setup

Board Setup

Vector CANoe

USB

CAN Case XL

short cables

FPGA-Board

Oscilloscope

Network Setup

Vector CANoe

USB

CAN Case XL

Bus Topology

FPGA-Board 1

FPGA-Board 2

CAN Transceiver

CAN cable

120 Ohm

Sub-D Connector

Monitoring (CANoe)