

## CAN Gets Even Better

### Ways to Transition from Classic CAN to the Improved CAN FD

Robert Bosch introduced the new CAN protocol CAN FD (CAN with Flexible Data rate) in March 2012. Key new features of this protocol are that it extends the useful data length from eight to 64 bytes and offers significantly higher data transmission rates. Based on its performance data, CAN FD is positioned between High Speed CAN (1 Mbit/s) and FlexRay (10Mbit/s). It is ideal for filling the performance gap between these bus systems at low cost. This article explains, from the perspective of a tool supplier, what needs to be modified and the effects of such changes on CAN FD development and simulation. These modifications are made on various levels: from the hardware level to potential data formats and various communication layers.

CAN FD gives the automotive industry a new foundation for better networking solutions in areas where bottlenecks have been occurring for years now due to growth in data traffic in vehicle electronics. This situation has resulted in compromises and makeshift approaches such as subdividing a network into multiple buses, which drives costs higher. However, switching from CAN to the higher performance FlexRay is associated with high investment costs. For the majority of developers a migration from the event-driven CAN to the time-triggered FlexRay also requires enormous changes to familiar ways of working and thinking.

#### Evolution Instead of Revolution

For around 20 years now, CAN has been the dominant bus system in the automotive industry, and so a broad base of developer know-how already exists in this area. This exper-

tise can still be applied to CAN FD projects, since CAN FD preserves existing CAN concepts such as bus arbitration, message acknowledgment, event-driven control, etc. CAN FD can be used very flexibly. Applications can benefit from either the higher bit rate, longer useful data length or both. When somewhat higher transmission rates are needed due to long bus lines, e.g. in some truck applications, the useful data length of up to 64 bytes helps to achieve noticeably higher data throughput.

Although CAN FD shares many of the same types of functionality as CAN, improved protocol extensions and adaptations to hardware and software are still required. Among other things, CAN FD introduces three new bits to the control field: the EDL (Extended Data Length), BRS (Bit Rate Switch) and ESI (Error State Indicator) bits. The EDL bit distinguishes frames in CAN FD format from those in clas-

sic CAN format: the EDL bit is recessive (High level) for CAN FD frames and dominant (Low level) for CAN frames. Similarly, a recessive BRS bit switches to the higher bit rate when sending the data field. The ESI bit is used to identify an error state of a CAN FD node. In addition, another four bits form the Data Length Code (DLC) that covers the extended useful data length; its possible values are 12, 16, 20, 24, 32, 48 and 64 bytes (**Figure 1**).

### Challenge for Tool Suppliers

For a supplier of software development tools, the challenge in introducing a new system such as CAN FD is to make suitable tools available as soon as the customer group begins to develop its first CAN FD ECUs. Integrating CAN FD in an existing test and simulation environment requires making rapid modifications to the various hardware levels up to the application level. An associated database is also needed to describe the communication network. In this context, the first question that should be asked is what is the best way to model CAN FD? Is a PDU abstraction necessary, considering the useful data length of up to 64 bytes? PDUs (Protocol Data Unit) are not only commonly used with FlexRay; they are also supported by the AUTOSAR System Description. They can be used for such purposes as additional CRC checks, or they can serve as Tx triggers.

Essentially, it makes sense to preserve as many of the practice-proven aspects of the CAN world as possible in modeling CAN FD. For example, it is possible to quickly migrate existing CAN test and simulation systems if CAN FD is not treated as a new bus system, but instead as an extension of CAN. Then the necessary configuration changes to tools are relatively minor. It is even possible to continue to use test scripts and databases. Then, OEMs and suppliers would be able to rapidly implement initial projects if they can initially limit CAN FD to eight data bytes. It is still possible to use DBC databases; they are widely used in CAN applications and represent the quasi industrial standard. In

addition, DBC can already handle useful data that is 64 bytes in length for protocols such as J1939.

### Flexible Hardware by FPGAs and Interchangeable Transceivers

The structure of the software tools – and their primary uses of analysis, testing, logging and simulation – reveal that CAN FD adaptations are required on nearly all layers of the automotive protocol stack. On the physical level, the tools almost always utilize a network interface for bus access. Network interfaces can be implemented in various ways, based on either FPGAs or standard controller chips. The latter represent a cost-effective solution, but they are strictly limited to a specific set of functions, and they must first be made available for CAN FD. The more flexible and quickly available alternative – at least in the initial CAN FD phase – is to use interfaces with FPGAs. This technology not only fulfils the necessary real-time requirements, it also enables driver updates to provide new functions such as 64-byte support or bug fixes. The question of whether CAN FD needs new transceivers depends on the specific use case. In vehicle networks, it is expected that a transmission rate between two and four Mbit/s will be implemented. For flashing ECUs, however, the rate should be as high as possible; eight Mbit/s or more are conceivable here, depending on the CAN transceiver that is used. Configurable hardware with piggyback transceivers, i.e. small interchangeable plug-on boards (**Figure 2**), offers a flexible approach that will support future transceivers.

On the communications level, developers often need analysis tools that depict the frames and enable access to specific data of the frame. Here, it is absolutely essential to correctly interpret the new CAN FD bits EDL, BRS and ESI; the same applies to the extended useful data length (**Figure 3**). It is easy to make the necessary modifications by modeling frames as events sent out by the tool. That is, it is sufficient to add the the CAN FD bits to existing CAN events and

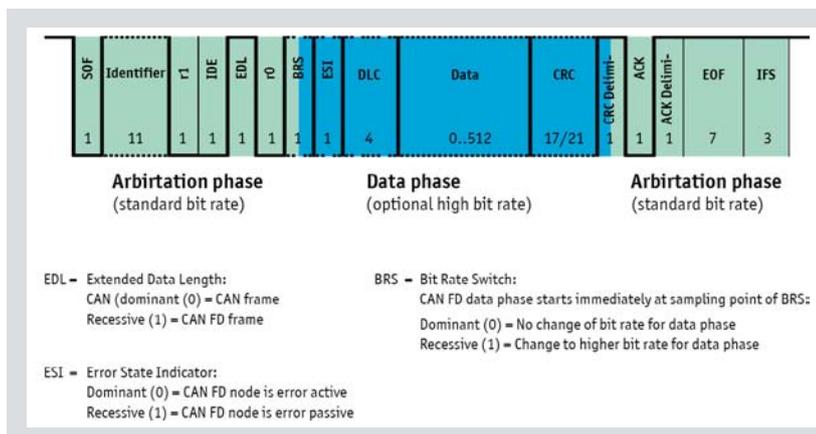


Figure 1: CAN FD frame



Figure 2: CAN FD bus interfaces with piggyback transceivers

extend the useful data length to the longer length. Only analysis windows that display the messages need to distinguish between CAN frames and CAN FD frames.

**CAN FD on Higher Layers**

On higher layers, the focus is on diagnostics, network management (NM), the transport protocol (TP), interaction layer (IL) and Tx models. While applications on the transport layer interpret even more frames, the application layer generally no longer works with frames, but with signals instead. Network management, the interaction layer and Tx models are all dependent on specific automotive OEM requirements. The extended useful data length of CAN FD affects both the interaction layer and the transport layer, and so it requires modifications. The advantage of modular tools with suitable interfaces is that not only are the OEM-specific features easy to implement or interchange, but the extensions required for CAN FD are easy to implement as well. On the application level, developers are in a good starting situation if they have designed their applications so that they are independent of a specific protocol

and they are systematically based on signals. They can continue to use their simulation models, test scripts and graphic user interfaces nearly unmodified.

**The Cards Have Been Reshuffled: CAN (FD), LIN, FlexRay**

Of course, the process of introducing a new bus system has some effects on current network protocols. Hardly any changes are expected in the convenience area of LIN, since CAN FD does not offer any advantages there. It must be realized, however, that it is impossible to implement simple TP Routing (known as raw TP) with useful data greater than eight bytes by simply copying the data of a CAN frame to a LIN frame. The situation is different with FlexRay, since CAN FD is a more economical alternative that is always the preferred solution when event-driven applications are being used. The use of FlexRay, on the other hand, is more limited to real-time critical and deterministic applications. Some migrations will be made in existing CAN systems. Networks with high bus loads of about 50 percent are candidates for CAN FD. The higher bandwidth of CAN FD avoids splitting networks with high bus loads, and it is then

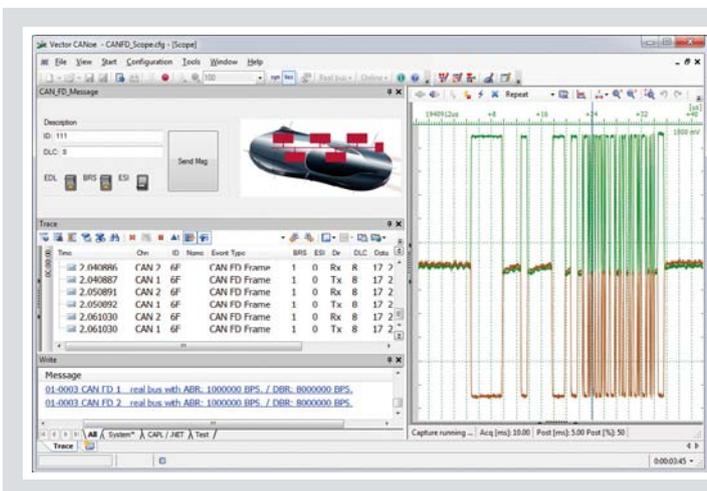


Figure 3: CAN FD simulation and test tool

possible to track multiple networks back to a single one. This saves on costs by not requiring as many gateways, and a beneficial side effect is that it eliminates undesirable gateway latencies.

Currently, there is no consensus on whether there will be mixed CAN/CAN FD networks; this depends primarily on decisions by the vehicle manufacturer. Certain actions would definitely need to be taken to implement mixed operation of CAN and CAN FD. For example, in FD operation CAN nodes need to be passively disconnected, because they would otherwise detect a form error due to the EDL bit and would send an Error Frame.

### Migration from CAN to CAN FD

An advanced analysis, test and simulation tool can offer valuable assistance in nearly all situations related to the development of CAN FD ECUs and networks. The possibilities range from analyzing an individual network node to fully simulating entire networks. Such a tool would give developers the ability to predict future bus loads and make optimal decisions based on reliable simulation data, e.g. to determine whether or not it is necessary to make a split into several networks. The tool is used to perform remaining bus simulations over the course of development, which substitute for a missing subnetwork and its components. Simulated nodes are then be replaced by real ECUs in a step by step manner. The simulation can also serve as a gateway, and it can connect to a new CAN FD network with a classic CAN branch. It is not necessary to make any modifications to existing CAN ECUs here.

### Conclusion

With flexible FPGA-based network interfaces and high-performance analysis, test and simulation tools, automotive OEMs and suppliers have all of the components they need for successful CAN FD developments. The described system supports different database formats and enables a variety of abstractions on the signal and PDU level. Open interfaces make it possible to implement OEM-specific properties. Network simulations and remaining bus simulations ensure optimal test conditions in all situations that occur during the development process.

**Translation of a German publication in Elektronik automotive, issue 04/2013**

### Image rights:

Vector Informatik GmbH

### Literature:

CAN FD specification V1.0, Robert Bosch GmbH

### Link:

Vector CAN Solutions: [www.vector.com/can\\_fd](http://www.vector.com/can_fd)



**Peter Decker**

has been employed at Vector Informatik GmbH in Stuttgart since 2002 and is currently Product Manager in the area of Multi-Bus Development for CANoe/CANalyzer.