CAN FD: Measuring and reprogramming

The complexity of the CAN FD technology is equivalent to the regular CAN network but it offers a significantly increased bandwidth. It is therefore an alternative to Flexray or Ethernet networks.

Author

Armin Happel

Erik Sparrer

Peter Decker

Vector Informatik GmbH
Ingersheimer Str. 24
DE-70499 Stuttgart
Tel.: +49-711-80670-0
Fax: +49-711-80670-111

Links
www.vector.com

Literature:
CAN-FD specification V1.0, Robert Bosch GmbH

CAN with a flexible data rate (CAN FD) is a technological evolution of the CAN network. It provides more bandwidth than CAN with less complexity than Flexray. The network specialists at Vector investigated two typical applications, measurement of ECU internal signals via XCP and ECU reprogramming, using the CAN FD system.

ECU measurement with XCP on CAN FD

In ECU development, the measurement and calibration of multiple signals and parameters for open and closed loop control algorithms represents an important calibration use case. ECU developers prefer to use the XCP (Universal Measurement and Calibration Protocol) measurement and calibration protocol that has been standardized by ASAM e.V. In the current protocol version 1.2, CAN FD is introduced as a new XCP transport layer. XCP enables the utilization of measurement and calibration tools such as Vector's CANape (Figure 1) to modify parameters during real-time operation and measure the altered behavior of the ECU. Considering a CAN system the bandwidth of the transmission medium may quickly become exhausted, depending on the number of signals to be monitored. XCP on CAN FD significantly extends the capabilities with up to 64 bytes of payload and data rates of at least 5 Mbit/s in the data phase.

XCP on CAN FD data throughput

To estimate the maximum available data throughput of XCP over CAN respective to CAN FD, the frame size versus the available payload within a frame has been investigated for a measurement of multiple ECU signals. The data throughput calculations are based on the assumption of 100 % bus load. The actual size of the frame fields for CAN and CAN FD are shown in Table 1 and Table 2. However, frame sizes cannot be predicted precisely for either CAN or CAN FD. To assure synchronization of bus nodes to signal edges, in dependence of its content an a-priori unknown amount of additional stuff bits is in- ▷

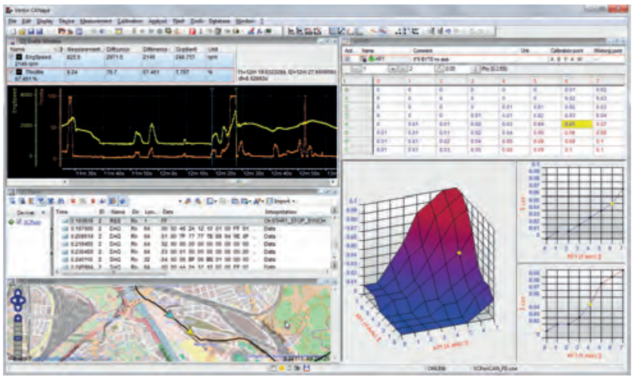

Figure 1: Measurement over XCP on CAN FD with CANape

serted into the frame. To model the stuff bit dependent frame size variation, a best and worst case scenario has been analyzed.

The results of data throughput calculations are graphically represented as a sector (Figure 2, Table 3), where a frame may reside in dependent of its actual contents. To verify the theoretical calculation, a realistic measurement reflecting a practical measurement use case was processed based on a simulation environment. At the laboratory setup – which consists of CANape measurement and calibration software, suitable interface hardware and a PC-based ECU emulation – the time of flight between the in- and output of the CAN/CAN FD driver was measured in both directions. The experimentally measured values greatly meet the mathematical prediction (Figure 2, Table 5) and hence verify the modeling of the available data throughput. Comparing the acquired measurement data needed for a transmission using CAN versus CAN FD, the data throughput of CAN FD has been found to be increased by factor 1.3 up to 5.4 depending on the system's configuration (Table 4).

Above its already improved bandwidth, XCP over CAN FD possesses further potential for improvement. Due to the equivalent physical communication basis of CAN and CAN FD, it is likely that the communication paths of existing ECU software will still be limited to an eight-byte data transmission after migrating to CAN FD. In this case XCP can only benefit from the higher data transmission rate but cannot utilize the full 64 bytes of payload available in CAN FD frames. To optimize the data transmission rate, the payload of small XCP packets could be concatenated as a large CAN FD frame (Figure 3). Vector is currently working on a proposal that enables packet concatenation for XCP over CAN FD in a future XCP specification.

Flash programming

(Re-) programming of flash memory is the second use case in which significant improvements are expected through the utilization of ▷

Table 1: Structure of a CAN frame

Name	Size [Bit]
Start of Frame	1
Arbitration Field	12
Control Field	6
Data Field	≤ 64
CRC Field	15
Acknowledge Field	2
End of Frame	10

Table 2: Structure of a CAN FD frame

Name	Size [Bit]
Start of Frame	1
Arbitration Field	12
Control Filed (1st part)	4
Control Field (2nd part)	5
Data Field*	≤ 512
CRC Field*	18 / 22
Ackknowledge Field	2
End of Frame	10

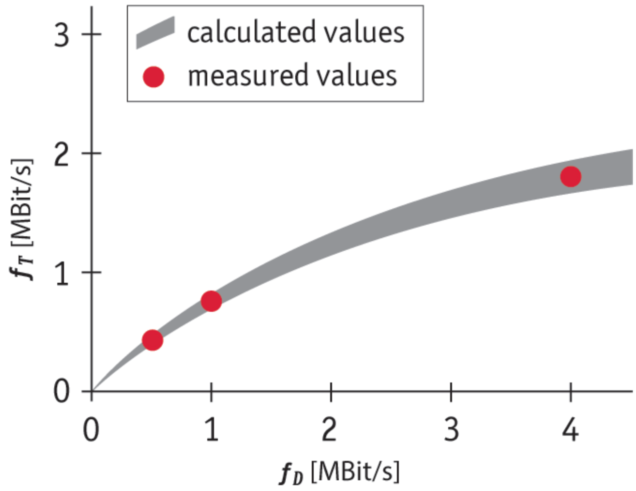

Figure 2: Measured and calculated CAN FD data throughput in ECU measurement

fast network protocols. In the three flash phases “delete”, “download/program” and “verify”, the download time is a key factor in conventional CAN systems, that can be accelerated by faster bus systems such as Flexray, Ethernet and CAN FD.

Regardless of the transmission protocol, it makes sense to use additional optimization strategies for downloading, such as data compression and pipelined programming. Although compression by an LZSS (Lempel-Ziv-Storer-Szymanski) algorithm reduces the volume of data to be transmitted, its efficiency is highly dependent on the data structure, and data extraction in the ECU generates additional CPU load that need to be taken into account. Pipelined programming, on the other hand, represents a type of parallelization: while a data segment is still being written in the ECU, transmission of the next segment is already started. Therefore, the potential performance gain from this method is the greatest when programming times are shorter than data transmission times.

Table 3: Calculated data throughputs of data measurement with XCP on CAN FD (fA=500 kbit/s)

f_D [kBit/s]	f_T [kBit/s] Best Case	f_T [kBit/s] Worst Case
500	407	341
1000	753	635
4000	2130	1825

Table 4: Comparison of measured data throughputs of data measurement with XCP on CAN and CAN FD

	Measured CAN	Measured CAN FD	Factor
Min.	294 kBit/s	380 kBit/s	1.3
Max.	318 kBit/s	1712 kBit/s	5.4

Table 5: Measured data throughputs of a data measurement with XCP on CAN FD (fA=500 kbit/s).

f_D [kBit/s]	f_T [kBit/s]
500	401 ± 21
1000	724 ± 46
4000	1884 ± 172

Flexray offers a transmission rate of 10 Mbit/s, but it is not fully available for (re-) programming. In the periodic communication sequence of the time-triggered protocol, all PDUs (Protocol Data Unit) are predefined in fixed slots. If many slots are reserved for diagnostic service requests such as for download, this reduces bandwidth for the useful data. Realistic configurations provide for 4 PDUs to 8 PDUs with 42 bytes to 255 bytes each per cycle for diagnose services. Vector engineers have measured download rates of 40 to 60 kB/s when pipelined programming is used.

Ethernet with Diagnostics over IP (DoIP) per ISO 13400-2 is also well-suited for fast reprogramming of ECUs. In testing 100 Mbit Ethernet and a typical microcontroller with a pure flash write rate of 180 kB/s, results were largely a function of the buffer size of the Transfer-Data service. A 16 KiB buffer enables throughput of around 150 kB/s, which is already near the limit of the flash memory used in the test.

Reprogramming via CAN FD

Since semiconductor manufacturers do not offer any microcontrollers that provide CAN FD support yet, network specialists at Vector used a microcontroller in which the CAN FD controller was implemented in an FPGA for their CAN FD measurements. The software stack on the board consists of a standard Vector UDS bootloader. The ISO 15765-2 transport layer and CAN driver were extended for support of CAN FD. To permit a quick test setup process for download testing, the CANoe simulation and testing tool was used, because the tool already offers CAN FD support. This software uses an external DLL which provides the flash programming procedure and transport layer functions. In the future, the Vector vFlash flash tool will become available for CAN FD.

With the transport layer that is used, the theoretically attainable transmission rate in flashing over CAN FD is 270 kB/s to 370 kB/s at 4 Mbit/s in the CAN FD data phase. However, real measured values lie well below this (Figure 4). Surprisingly, the compression and pipelining optimization strategies were counterproductive for CAN FD in the test environment that was used. The reason is that, in the laboratory setup used, the programming time for the internal flash memory became the limiting factor in the flashing process. So this made optimizations to the download phase ineffective. However, further tests with more powerful CPUs are needed to arrive at more general conclusions about data throughput and the effectiveness of optimizations. A key finding of the measurements is that CAN FD delivers a significantly higher data throughput than CAN (Figure 4), and the effort required for migration is negligible.

Summary and outlook

Overall, it is still difficult to arrive at an objective comparison of the serial bus systems CAN FD, Flexray and Ethernet due to their different microcontrollers and constraints, but certain tendencies can be clearly discerned. In the case of Flexray, high download ▷

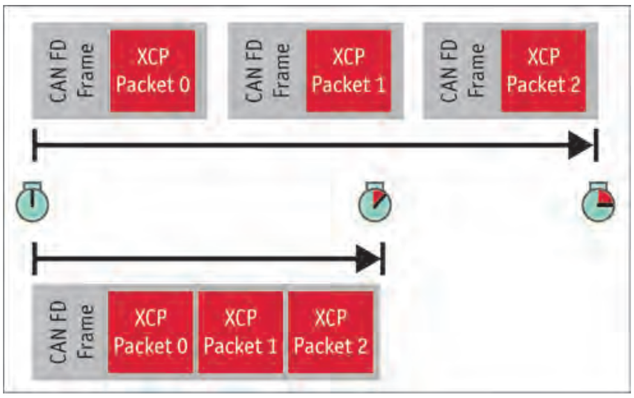

Figure 3: Faster data transmission by multiple XCP packets combined in one CAN FD frame

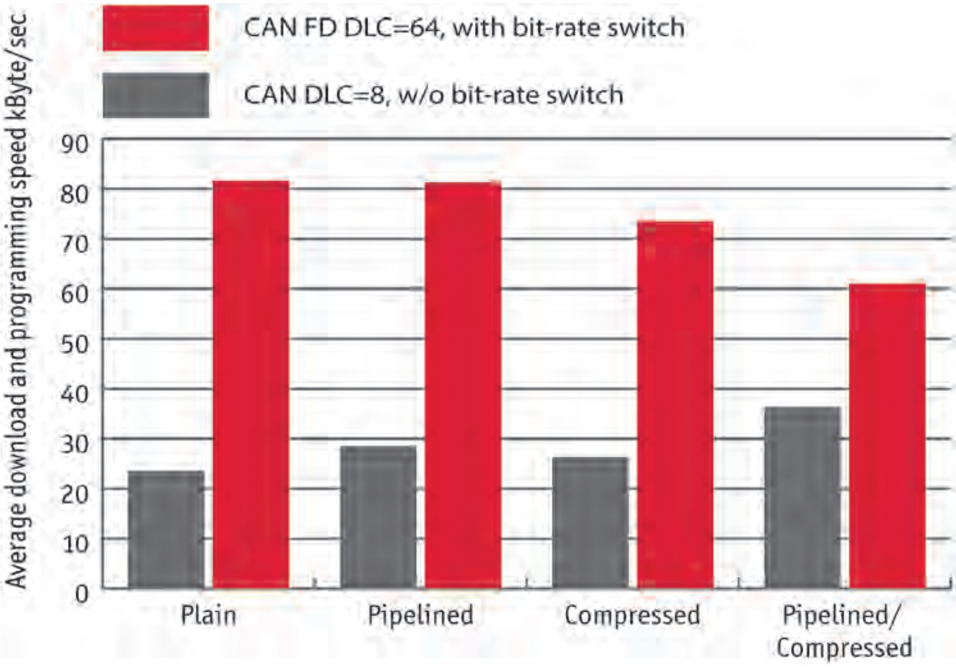

Figure 4: Comparison of download and programming times with CAN and CAN FD

speeds and high performance for the real time data payload are not both achievable at the same time. 100 Mbit Ethernet delivers the fastest transmission rates, but it requires complex software configuration, and its hardware costs are higher than for CAN FD. CAN FD appears to be the most balanced solution, it offers high data rates and the potential for further improvement at moderate costs. In addition, it is relatively easy to migrate to the improved CAN, because of the close similarities between CAN and CAN FD. Both protocols are based on the same physical layer, and this enables reuse of transceivers, wiring and bus topologies. Since the communication principle has not changed either, existing know-how can still be applied. The modifications to affected software layers in calibration and reprogramming that need to be made are relatively minor.

CAN FD enables significant throughput gains in both measurement and reprogramming of ECUs. In (re-) programming, this shifts the bottleneck more to the flash memory. Further development to shorten the memory access times of the MCUs that are used promise additional performance gains. Efforts by Vector to extend the XCP specifications to include packet concatenation with CAN FD also offer the potential for increasing performance of the new protocol that is still untapped. ◀