Vehicle Controller Area Network Response Time Analysis and Measurement Issues - to Reduce the Gap between Estimation and Measurement

2017-01-0018

Published 03/28/2017

Jeong Chan Kim
Hyundai Autron Company

Kai Richter
Symtavision GmbH

Myung Hyun Koo
Hyundai Autron Company

Matthias Hagner
Symtavision GmbH

Chung Hi Lee
Hyundai & Kia Corp.

CITATION: Kim, J., Richter, K., Koo, M., Hagner, M. et al., "Vehicle Controller Area Network Response Time Analysis and Measurement Issues - to Reduce the Gap between Estimation and Measurement," SAE Technical Paper 2017-01-0018, 2017, doi:10.4271/2017-01-0018.

Abstract

Along with the efforts to cope with the increase of functions which require higher communication bandwidth in vehicle networks using CAN-FD and vehicle Ethernet protocols, we have to deal with the problems of both the increased busload and more stringent response time requirement issues based on the current CAN systems. The widely used CAN busload limit guideline in the early design stage of vehicle network development is primarily intended for further frame extensions. However, when we cannot avoid exceeding the current busload design limit, we need to analyze in more detail the maximum frame response times and message delays, and we need good estimation and measurement techniques. There exist two methods for estimating the response time at the design phase, a mathematical worst-case analysis that provides upper bounds, and a probability based distributed response time simulation. While both provide valuable information at design phase, we cannot easily measure message response times using the established bus tracing techniques because those bus traffic traces only contain the reception times of each message. Determining the response time requires knowing also the point in time this message is generated within the control unit, which is usually not possible.

In this paper, we present an approach to approximate these intra-ECU message generation times in order to enable reasonable response time measurements. The approach uses a new frame-burst timing analysis that solely uses the standard bus trace information, in particular the reception times of frames. The improved method reduces the gap of estimation and measurement of timing behavior in a CAN network and enables analyzing the network timing efficiently at all phases of vehicle network developments.

Introduction

As system integrators coping with the new vehicle features of increasingly coordinated functions such as Advanced Driver Assistance Systems (ADAS), it is important to adapt timing analysis methods properly throughout the E/E architecture development phases. Although many researches are ongoing to implement the next generation of networking technologies beyond the classic CAN network vehicles, still there are many remained uncertainties to simulate and to measure the CAN frame response times practically. To tackle these topics in this paper, we describe the methods of response timing verification for vehicle network system developments and a measured maximum response time analysis method which is based on the burst period analysis with practical point of view.

When a vehicle network topology with messages is designed, the first acceptance criterion for the network message specification is the busload level. The maximum busload requirement has been used for many years at almost all OEMs with busload limits ranging from 30% to 60%. Since the requirement should be satisfied from the early design phase, if the current specification is exceeding the upper bound, various kinds of optimization methods need to be performed.

To reduce the busload, from changing the message properties (identifier, send type, period, etc.) to redesigning network topology to use subnet or multiple transceivers in a control unit, the total cost of the change can be expensive. And more, as the busload is increasing continuously, the precise analysis of busload effects to the frame latency is required.

Frame latency analysis methods can be broadly classified in deterministic timing analysis, probabilistic timing analysis and measurement based analysis. The most often used conventional timing analysis approach is the deterministic timing analysis which has been mentioned as the worst case response time (WCRT) analysis [3]. However due to lack of system knowledge in organizations, responsibility of legacy systems, and the unknown probability of occurrence of the analysis results, the calculated WCRT bound is hard to use in vehicle production phases [5]. Furthermore, most of the improved analysis methods are usually based on modeling of fine-grain software or hardware behaviors to reduce the gap of results with measured values. Since the scope of OEM information to do the analysis at the early phase of network development is usually about the communication network topology including CAN controller types and message specifications, the preferred method needs both to handle the control unit's internal behavior as black box model and to simulate the related key design parameters with bounds which are affecting overall timing performance.

For that purpose, we use probabilistic analysis which focuses on typical cases rather than worst cases only – we use SymTA/S which can actually do both. This distribution analysis provides timing statistics and a probability of deadline violations, although it does not guarantee that the worst case is ever found. The analysis is based on a holistic system simulation, where the interferences of each element to other elements are considered, based on the modeled system architecture and parameterization.

Few probabilistic *measurement* approaches exist that take into account the full details of the ECU internal timing and rely on extensive testing performed on the real system under analysis. However, determining reasonable engineering safety margin is extremely difficult, especially when the system may exhibit discontinuous changes in timing due to pathological cache access patterns or other unanticipated timing behavior [6]. Furthermore, since we are assuming the black box approach even in the measurement based analysis, the measured information on a running vehicle network is only the received message timing, in our case this is following the vector CANoe trace data ASCII format. Using the trace data, we want to analysis both the actual bus load and the maximum response time of frames.

The bus load is easily determined by both, the simulation and the measurement. However to the best of our knowledge, there is no available method to analysis the maximum response time of frames by using only the received frame events.

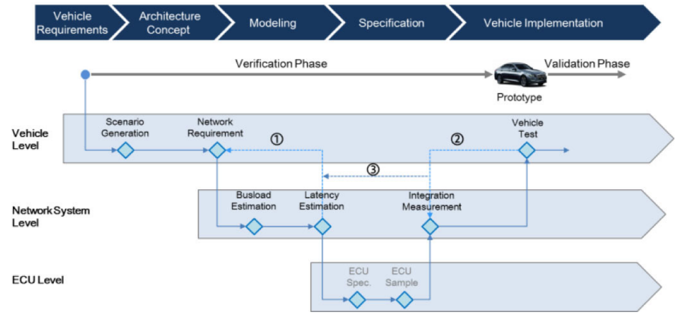

Figure 1. Network Timing Analysis Process Overview

In this paper, we describe a method to approximate CAN message's response times from measured bus traces. As indicated in the Figure 1, we will focus on the following issues.

1. How can we estimate as precisely as possible the busload and the maximum response times of the messages? And we want to further enhance the decision metrics and acceptance criteria for both busload and response times at the early development phase.
2. How can we measure the maximum frame response at the network integration phase only from the bus traces, without tracing ECU internal timing behavior?
3. How can we reduce the gap between estimated and measured results?

The contribution of this paper is to describe a method which can be used as guidelines for the above issues. The method does not require control unit level timing information, the maximum CAN frame response time is rather determined at the network level by using a new developed maximum burst-time analysis method.

Understanding the Busload and WCRT Analysis Accuracy

The purpose of busload analysis is to estimate the remained available communication bandwidth, usually in percent of the total available bandwidth. And it has been used for long times as the first design rule for the network signal specifications. In case of a standard CAN message which is shown in Figure 2, when the data field size is 8 bytes, the maximum protocol overhead size is total 71 bits which is containing both the constant part of 47 bits (for all head and tail bits excepting data field) and the dynamic part of 24 bits for maximum stuffed bits.

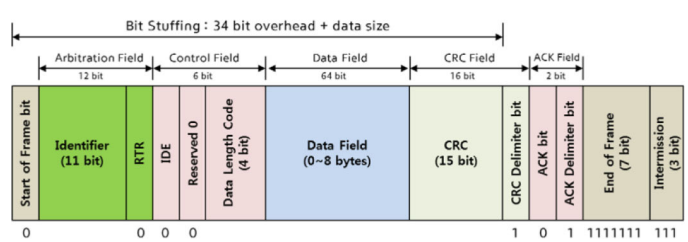

Figure 2. Standard CAN Frame Layout

The equation (1) is showing the network busload which is calculated by summing all the message utilizations in a network [3]. For each message, the utilization provides the ratio of the total transferred data size to the maximum transferrable data size during a period.

$$Busload = \sum\{(8S + 47 + BitStuffing) * \frac{\tau_{bit}}{T_m}\} \tag{1}$$

Where S is the number of data field bytes, T_m is the period of message m, τ_{bit} is the transmission time for a single bit, and the bit stuffing size is the floor integer value of (34+8S-1)/4.

For the busload simulation, assumptions must be made for the actual number of stuff bits that can lead to a difference to real measurements, and the actual number of bit stuffing is usually higher than that would be expected [7, 8]. So if we know the generated bit stuffing number for a given network topology by actual measurements, the accuracy of busload simulation can be increased. In our experiments, when we are simulating the maximum case of bit stuffing, the difference of simulation with measurement is about 5%. And for using the average case of bit stuffing (12 bits), the result is within 2% difference with measurements.

Regarding non-cyclic event messages, as opposed to periodic messages, we can observe that in our case, the most critical timing requirements are found in the chassis and powertrain networks. These mainly consist of periodic messages. So the busload analysis of non-cyclic event messages is not much effecting to the overall result in practical.

For the error frames, these can in fact influence the bus load and the response times as a result from message retransmission; basically the error frame occurrence is largely related with the fixed physical network topology. Since the electrical network properties of a network topology should be verified prior to the busload analysis phase, we can assume there is a little effect of error frame to busload analysis.

Although the described busload analysis method is providing the first measure to decide how many new messages can be added, the message response times are needed as a second criterion to decide how fast the message can respond within the given deadline (or cycle time). For the purpose, the WCRT analysis is used to know the upper bounds on frame response times while providing deterministic guarantees. It means that at least the WCRT analysis result should be less than the deadline.

About the limitations of the WCRT analysis, it is not able to capture the complexity of the real systems and the result is often pessimistic which can lead to unnecessarily conservative design choices [4]. The WCRT analysis assumes that several frames are trying to send simultaneously at once in control units, leading to longer queuing times in the transmit buffers and longer arbitration times due to bus congestion. Such worst-case message queuing times can become as large as up to 10 ~ 20 times of the pure transmission time. But while such timing situations may only occur in rare situations (or never) in reality, this kind of upper bound analysis is absolutely necessary as a second verification criteria of vehicle network development process.

Because of the relationship of the two kinds of design margins which are shown in Figure 3, there are cases when the busload design limit is satisfied but the response time design limit is not satisfied.

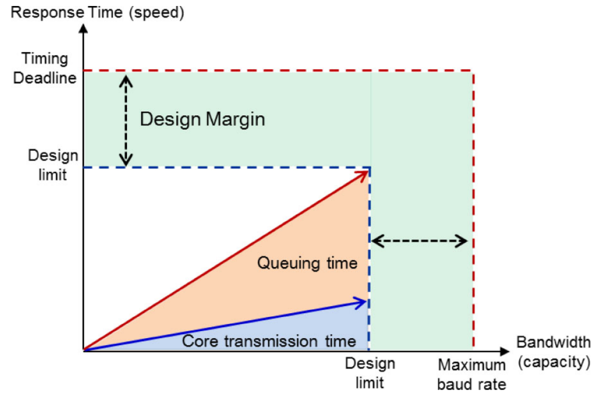

Figure 3. Busload and Response Time Design Margin

Probabilistic Simulation of Response Time

In addition to worst-case response time analysis which can be used as design decision criteria especially for time critical systems where missing deadlines can lead to system failures, the typical-case timing analysis for frames carrying not so time critical information can be applied in general. Also for the time-critical system analysis, this approach can be used to provide evidences for showing how close the typical-case timing is to the worst-case timing.

To find out the average fulfilling timing behavior, we use SymTA/S distribution analysis. This analysis simulates the timing behavior of the system (e.g., the bus) and provides response time statistics that can be observed in different kinds of diagrams (e.g., histograms).

In this approach, the analysis method is considering the ECU level differences of actual start timing of allocated frames which are assumed all frames are sent simultaneously for the worst case response time analysis. The key difference is that it will vary two major parameters: the clocks of the different ECUs and the transmission times. The rest will be used exactly in the same way.

In a bus system, the general timing behavior is not deterministic. Even if there are only periodic frames sent, the ECU clocks may drift and, hence, there may be a shift in the timing distance between the sending of two cyclic frames. Additionally, depending on the start-up of the entire system in the car and, hence, of the connected ECUs, the initial time difference of the sending of the frames may vary. Hence, a simulation in SymTA/S has to handle such changes and point out these different situations.

One simulation in SymTA/S means to run multiple “runs” after another. At each of these “runs”, the clocks and start-up order of the ECUs is changed. And so the sending out of the frames in comparison to the behavior of the other ECUs is changed, too. One example is depicted in Figure 4.

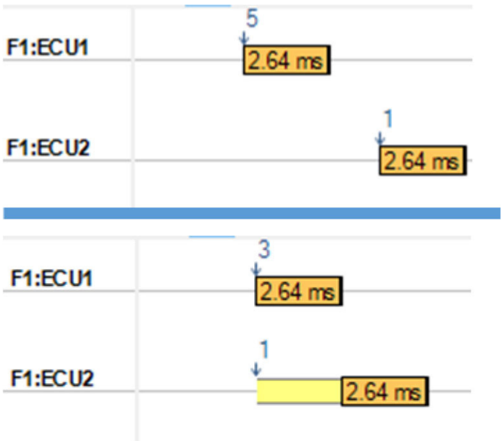

Figure 4. Clock Drift Example

The above figure is showing two different “runs”. In the first example, the startup behavior has led to a situation in which there is a difference between the sending of the two frames. Hence, there is no interference between both. In the second example, the startup behavior led to a situation in which two frames from two different ECUs are created at the same time. Hence, the one with the lower CAN ID is sent out first and the other one is delayed. This leads to a longer response time of the second frame. SymTA/S is encouraging all users to simulate as much “runs” as possible to cover many different situations and to get a statistical result about the timing behavior.

The second variation is the transmission times. This is only possible if there are varying frame sizes. If the frame has a constant size, it will always have the same transmission time. If it varies, SymTA/S will consider a transmission time based in the possible boundaries defined by the model.

Both properties are considered quite differently in a mathematical worst-case analysis which seeks for maximizing the simultaneous frame generation (within the boundaries of the CAN specification) and always consider the maximum frame size and hence transmission time.

If there are sporadic frames in the bus system, there is a third variation. In SymTA/S it is possible to define a probability of the occurrence of this frame, whereas in the worst case, it will always consider a sending of all frames.

To choose a value for the “runs”, SymTA/S is using the Uniform distribution as basis considering the defined boundaries.

Additionally, for the user to get a representative statistic, it can be defined how long such a “run” should be, and how many shall be simulated. Based on that, the simulations are observed and a timing statistics is generated.

In Figure 5, a histogram of the response times a frame is depicted. It shows the relationship between occurrences of response times in all simulations and the actual value of the response times. For example, Frame F35 has around 1,350 occurrences of the response time 0.265 ms, which is also the best response time that was found in the simulations. There are also some occurrences with larger response times.

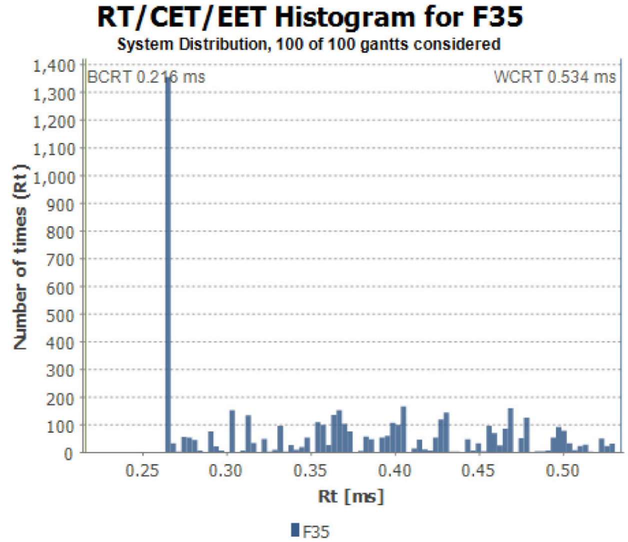

Figure 5. Probablistic Histogram of a Frame Response Time

In addition and for better orientation, Figure 5 also contains the best-case and worst-case response times obtained from the mathematical analysis. It also shows that some of the occurrences are already quite close to the worst case response time of 0.534 ms.

Measured Max Response Time Analysis/Trace Analysis

A key part of the timing analysis is to compare the simulated timing with the reality. In the past sections we described analysis that is based on a model of the reality. The model was created based on import information that is also used for the network configuration. Nevertheless, a comparison with the reality is necessary to increase the confidence in the analysis results.

Regarding the analysis of CAN buses and the measurement, one workflow is established for many years. It is possible to use Vector’s CANoe/CANalyzer tools to created traces that describe the real behavior on a bus in form of log information. That means, the tool will create a file that contains one lines with information when a certain frame is received. This information can be used to build a Gantt diagram of the behavior of the bus including information like the start sending of a frame and of course the receiving of a frame. It is also possible to calculate the average load of the bus or observe the load change during the measurement.

However, what is not part of the trace information is the actual activation or creation time of a frame. The difference is depicted in Figure 6.

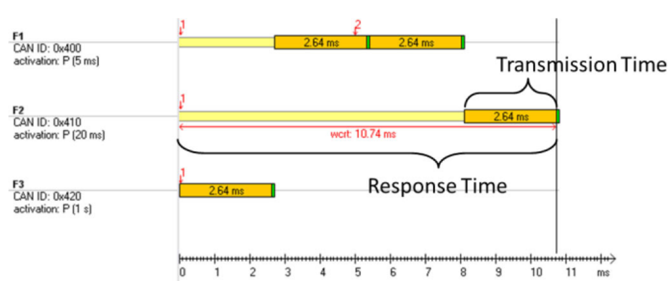

Figure 6. Frame Response Time

The response time is the time from activation of the frame until the complete transmission including delays due to both the higher priorities frames and a lower priority blocking frame. For the vehicle level traces measurement cases, logging the activation time of the frames is impossible, as it is happening inside of an ECU. Hence, as we can only find out the start of the sending and not the activation from the CANoe trace, we cannot find out the response time directly from the trace data.

To estimate the queuing start time in a frame by using the measured bus line data without the ECU's internal timing information, we developed a method that uses information about message burst times. In normal cases, when a queuing delay happens by higher priority frames, we see several consecutive messages on the bus / in the trace log, which is called as burst of messages. So by careful analysis for each such burst periods, we can find out which higher priority frames could possibly result in a queuing delay for other messages within the burst period. From this information, we can estimate the earliest point in time, at which any message could have been activated. And from that, we can approximate the response time. The proposed "measure max response time" method is summarized as follows.

Step1. Extraction of burst periods from trace data

- Core transmission time calculation using the transferred bits counts for all frames.
- If the two consecutive received frame time gap is same with the intermission bits (3 bits) transfer time, compose a burst period.

Step2. Frame queuing start time calculation within a burst period

- Find the prior arrived consecutive high priority frames.
- After finding the high priority frames, find a next consecutive lower priority frame if it exits.
- The queuing start time is estimated as the start transmission time of the last found frame.

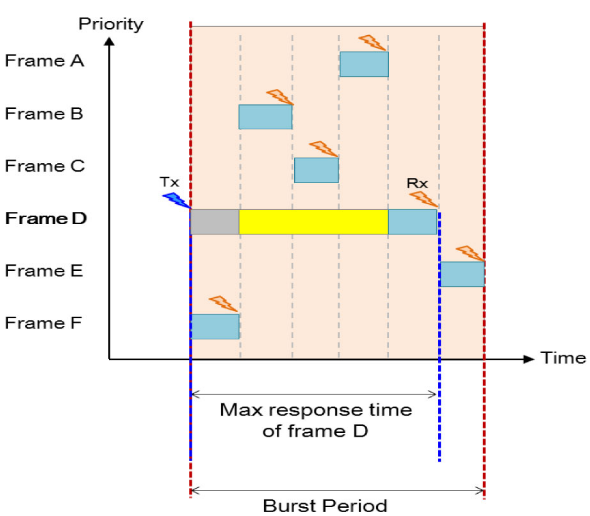

Figure 7. Measured Max Response Time Analysis within a Burst Period

In the figure 7, the measure max response time of frame D is depicted by summing its core transmission time (showed as light blue box), queuing delay times by the higher priority frame's transmission times (showed as yellow box) and a blocking time from a lower frame F (showed as grey box).

Effective Cycle Time and Jitter Analysis

In the last section, we explained how we can determine the worst-case response time by simulation and by vehicle-level measurements, and then compare both. As another important timing property, we want find out how much the response time is changing over time. For this we can use the effective cycle time analysis method. The effective cycle time is defined as the time distance of the consecutively received frame times. Especially, because the greater amount of frame response time delay is happening at the transmitting buffer inside a node, the effective cycle time analysis method is a practically effective method to understand the buffering behavior inside a node.

So it is good to use the effective cycle time analysis as a simple problem detection method in case of performance degradation situations. Figure 8 is showing an example of gateway output message delay problem which is detected by a cluster warning indicator. This problem can happen from time to time and it is usually not detected by diagnostic trouble code inside the gateway. We therefore want to use vehicle-level measurements as black box analysis by measuring the effective cycle times for both the destination bus messages and the input source bus messages of the gateway. Such multi-bus message analysis (in contrast to analyzing the internal scheduling behavior of the gateway) provides a possibility to determine gateway response times for routed frames. This way, we can check if there are unexpected message delays in the gateway.

Of course, the effective cycle time analysis method shall be used together with the previous mentioned response time analysis and measurement methods. And we think the basic benefit of using this method is to understand both the implemented average periods of cyclic messages and the maximum and minimum values in a short time by using the real measured data in the vehicle.

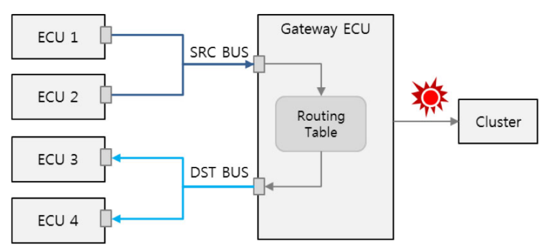

Figure 8. Gateway Output Delay Example

To know the minimum and maximum effective cycle times, the following equations are used by only using the received frame time information.

$$\text{Max Cycle} = \max_n(\mathit{receive\ time}_n - \mathit{receive\ time}_{n-1})$$

$$\text{Min Cycle} = \min_n(\mathit{receive\ time}_n - \mathit{receive\ time}_{n-1})$$

$$\text{Average Cycle} = \operatorname*{avg}_n(\mathit{receive\ time}_n - \mathit{receive\ time}_{n-1})$$

$$\text{Jitter} = \max(\text{Max Cycle} - \text{Period}, \text{Min Cycle} - \text{Period})$$

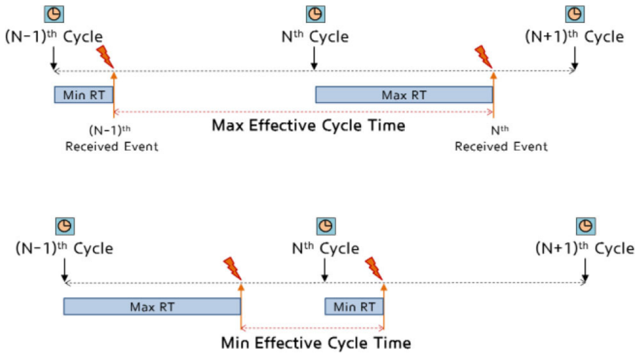

Figure 9. Measured Effective Cycle Times

Figure 10 is showing the effective cycle time of a 10 ms cyclic message over a period of time. We see that the effective cycle time is first between 8 and 12, which is acceptable, but the difference grows continuously to between 3 and 17, which indicates a significant distortion to the communication. Finally and all of a sudden, the frame distances resort to their original behavior with only very little deviations from the desired period of 10ms.

We call this deviation of the effective cycle time (between 3 and 17) from the nominal cycle time (10) the message jitter. For such jitters, we usually have an important constraint. They shall for instance not be higher than 70% of a message period. Even though there are acceptable cases for high jitters, this case is usually one of the potential problematic cases based on our experiences.

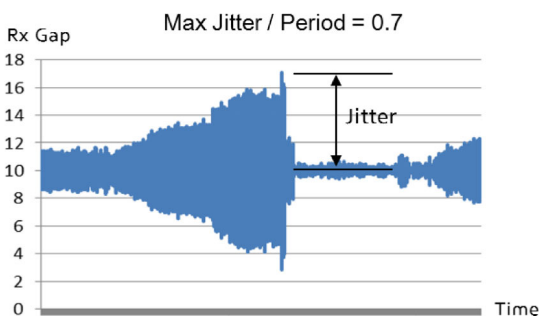

Figure 10. Maximum Response Time Jitter

Comparison

	Worst Case Analysis	Simulation Analysis	Trace Analysis
Input	Simple frame model (ID, Size, Period)	Full frame model(ID, Size, Period, ECU, Clock)	Measured frame arrival times
Quality of the analysis	Upper bound/guaranteed boundary	Statistical output/ histograms	Depend on vehicle test case
Response time analysis	Theoretical calculation from activation till complete transmission	Simulation to find an average behavior from activation till complete transmission	An approximated time from activation till complete transmission

This section should give a short summary about the three different presented analysis technologies. Each of them has a different value and the combination of them will give a full picture about the timing behavior of the bus system.

Case Study

By using the below message example which consisted of 42 cyclic frames allocated with 9 ECUs, the figure 11 is showing the 3 kinds of different response time analysis results.

ECU	Period (ms)	Message ID
ECU1	10	0x1d7, 0x1d5, 0x1d3, 0x1d1, 0x1d0, 0x1d2
	20	0x1d6
	200	0x1d4, 0x420
ECU2	10	0x138, 0x137
ECU3	200	0x372
ECU4	10	0x281, 0x278, 0x101, 0xfb
ECU5	10	0x31d, 0x2ea
	20	0x431, 0x392, 0x311
	100	0x436, 0x58b
	200	0x553, 0x588, 0x5be, 0x435
ECU6	10	0xf6, 0xf5, 0xf4, 0xf3, 0xf1, 0xf0
	200	0x4e7, 0x4b1, 0x47c, 0x410, 0x446
ECU7	10	0x131, 0x130
ECU8	20	0x370
ECU9	10	0x220

We performed all three analyses. The simulation we performed for a single simulation “run” and for 1000 simulation “runs”.

- The red line shows the WCRT results of the worst-case analysis. The mathematical worst-case analysis is assuming all the possible delays by higher priority frames, therefore the response times of lower priority messages are continuously increasing.
- The blue line shows the maximum response times from 1000 bus simulations. We see that they are below the worst-case times. This is what we expect because WCRT are conservative and always at least as large as the simulated results.
- The gray line shows the results of a single simulation run, representing one specific situation, while the blue line represents 1000 situations and is therefore much more reliable because it captures especially the clock drift effect (source of so many different situations) to a reasonable degree.
- The orange line finally shows the response times that we approximated from a measured trace using the burst timing analysis method described in this paper.

In particular, we see that the measured (approximated) response times of some messages are similar to the simulated response times of a single “run”, for other messages they are similar to the simulation result of 1000 “runs”. This observation also covers areas where the mathematical worst-case results are much larger than the simulation results even for 1000 “runs”.

Although such experiments do not provide mathematical evidence, for our practical network development task, we conclude that

a. real response times can be determined by our burst periods based approximation technique from bus traces, so we can understand the gap of response times between analysis and measured for a given vehicle test condition, and
b. distribution analysis methods can simulate realistic upper bounds of message response times and therefore can be used to make good decisions at network design stage without using the real vehicle.

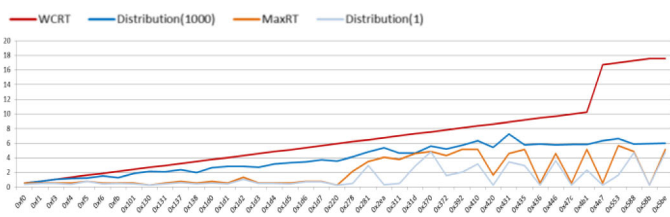

Figure 11. Comparison of Response Time Analysis Results

Summary

In this paper, we have identified the main challenging issues about CAN response time analysis, which is important to handle the increased bus load situation with further optimization approaches throughout the vehicle network development phases. To understand the gap of the unrealistically assumed worst-case response time analysis results with the actual response times in vehicles without using the detail information of network nodes, both the probabilistic simulation analysis method and a new maximum response time approximation method are shown. Also we advocate the usefulness of response time jitter analysis method to find out the level of instability for frame response times. Further work will encompass an analysis method for optimizing gateway performance.

References

1. Richter Kai, Ernst Rolf, “How OEMs and Suppliers can face the Network Integration Challenges”, Design Automation and Test in Europe, Pages 183–188, 2006, ISBN: 3-9810801-0-6.
2. Traub, M., Lauer, V., Becker, J., Jersak, M. et al., "Using Timing Analysis for Evaluating Communication Behavior and Network Topologies in an Early Design Phase of Automotive Electric/Electronic Architectures," SAE Technical Paper 2009-01-1379, 2009, doi:10.4271/2009-01-1379.
3. Davis, Robert I. Burns Alan, Bril Reinder J., Lukkien Johan J., “Controller Area Network (CAN) Schedulability Analysis: Refuted, Revisited and Revised”, Real-Time Systems, Jan. 2007, doi:10.1007/s11241-007-9012-7
4. Nicolas Navet, Shehnaz Louvart, Jose Villanueva, Sergio Campoy-Martinez, et al., "Timing verification of automotive communication architectures using quantile estimation", Embedded Real-Time Software and Systems, Mar 2014.
5. Quinton Sophie, Bone Torsten T., Hennig Julien, Neukirchner Moritz, et al., "Typical Worst Case Response-Time Analysis and its Use in Automotive Network Design", Design Automation Conference, 2014.
6. Cazorla Francisco J., Quiñones Eduardo, Vardanega Tullio, Cucu Liliana, et al., "PROARTIS: Probabilistically Analysable Real-Time Systems", ACM Transactions on Embedded Computing Systems, 2013.
7. Nolte Thomas, Hansson Hans, Norström Christer, Punnekkat Sasikumar, "Using bit-stuffing distributions in CAN analysis", IEEE Real-Time Embedded Systems Workshop, 2001.
8. Gunnarsson, D., Traub, M., and Pigorsch, C., "Timing Evaluation in E/E Architecture Design at BMW," *SAE Int. J. Passeng. Cars – Electron. Electr. Syst.* 7(1):178–188, 2014, doi:10.4271/2014-01-0317.

Contact Information

Jeong Chan Kim
JeongChan.Kim@Hyundai-Autron.com
Hyundai Autron Company
463-400 Seongnam Gyeonggi South Korea

The Engineering Meetings Board has approved this paper for publication. It has successfully completed SAE’s peer review process under the supervision of the session organizer. The process requires a minimum of three (3) reviews by industry experts.

ISSN 0148-7191

http://papers.sae.org/2017-01-0018