Journal of Information Processing Vol.21 No.1 46–52 (Jan. 2013)

[DOI: 10.2197/ipsjjip.21.46]

Regular Paper

Minimization of FlexRay Bus Bandwidth for Hard Real-Time Applications

Makoto Sugihara[1,a)] Akihito Iwanaga[1,b)]

Received: March 5, 2012, Accepted: September 10, 2012

Abstract: It is essential to cut down the fabrication cost especially in mass production. This paper presents a design methodology in which we reduce the operating frequency of a communication bus under hard real-time constraints so that we can cut down the cost of a communication mechanism of an in-vehicular embedded system. The reduction of the operating frequency contributes to choosing a slower and cheaper wire harness that constitutes an in-vehicular network system. We formalize a bus bandwidth minimization problem to optimize a payload size of a frame under hard real-time constraints on the assumption that each and every signal is uniquely mapped to its own time slot of the time division multiple access (TDMA) scheme. Our experimental results show that our methodology obtained an optimal payload size of a frame and an optimal operating frequency of a bus for several hypothetical automotive benchmarks. Our method achieved one-fifth of the typical bandwidth of a FlexRay bus, that is 10 Mbps, for the SAE benchmark signal set.

Keywords: FlexRay, TDMA, fabrication cost, design methodology

1. Introduction

The application of information technology (IT) to an automobile offers a driver various values such as high safety, high comfort, high convenience, and low energy consumption. The car industry will adopt a larger and more complex embedded system as a means to realizing a competitive automotive product of high function. System designers must now realize a large embedded system while they must lower the fabrication cost for it.

It is essential to cut down the fabrication cost especially in mass production. An automobile includes a large and complex embedded system that is generally expensive. Such an embedded system typically includes a communication mechanism through which an electronic control unit (ECU) communicates with one another under hard real-time constraints. This paper mainly focuses on cutting down the fabrication cost of a communication bus. A luxury car includes over 100 ECUs and 3 km of wire harness whose weight amounts to about 40 kg. An embedded system contains a large quantity of wire harness that affects the fabrication cost of the system.

There exist two major network standards: the controller area network (CAN) [1] and the FlexRay [3]. The CAN is the most popular communication network standard. The CAN, however, has difficulty in low determinacy with communication latency because the CAN arbitrates communication messages in a priority-based fashion. The CAN is also sensitive to the offset and jitter of a communication message. The CAN is incapable of realizing hard real-time applications such as x-by-wire systems from lack of determinacy in communication latency [2]. By contrast, the FlexRay network is another communication network standard which provides determinacy in communication latency as well as flexibility in using network bandwidth [3]. The FlexRay offers determinacy in communication latency with the time division multiple access (TDMA) scheme while it offers flexibility in bandwidth utilization with the flexible time division multiple access (FTDMA) scheme. The FlexRay requires to assign every communication message its own time slot(s) so that it can guarantee constant latency and throughput. System designers must schedule every communication message within static segments when designing their system in exchange for obtaining constant latency and throughput.

This paper proposes a design methodology which reduces the operating frequency of a FlexRay bus so that it can reduce the cost to fabricating a networked embedded system. This paper assumes a communication mechanism in which each and every communication signal is mapped to its own static slot. The reduction of network bandwidth generally contributes to cutting down the cost of the wire harness. To the best of our knowledge, our work is the first study to minimize the operating frequency of a communication bus by optimizing a network parameter for the TDMA scheme so that the fabrication cost is reduced. The FlexRay specification Version 2.1 Revision A clearly states as follows: *Currently the only bit rate defined for FlexRay is 10 Mbit/s. There is, however, the intent that the protocol will be expanded in the future to include bit rates lower than 10 Mbps* [3]. Design methodology for reducing the operating frequency of a bus will become beneficial especially after the FlexRay consortium permits designers to change the operating frequency of a bus.

The remainder of this paper is organized as follows: Section 2

[1] Kyushu University, Fukuoka 819–0395, Japan
[a)] sugihara@slrc.kyushu-u.ac.jp
[b)] iwanaga@soc.ait.kyushu-u.ac.jp

briefly reviews a FlexRay communications system protocol specification. Section 3 discusses and formulates a bandwidth minimization problem in which we assume that the size of a payload segment of a frame is a constant under hard real-time constraints. Section 4 regards the payload size as a variable and extends the bandwidth minimization problem shown in Section 3 for further bandwidth reduction. Section 5 discusses our experiments on bandwidth minimization. Section 6 compares our methodology with that of some of related work. A concluding remark is finally given in Section 7.

2. FlexRay

This section reviews the FlexRay communications system protocol specification [3]. Further detail should be referred to the FlexRay protocol specification [3].

2.1 Frame

A *frame* is a container that encapsulates transmitted data. A frame consists of three segments: *header segment*, *payload segment* and *trailer segment*. **Figure 1** details the FlexRay frame format. A sender node is to transmit a frame via the FlexRay bus such that the header segment appears first, the payload segment appears next, and the trailer segment appears last. The sender node is to transmit the fields in left to right order as shown in Fig. 1.

The FlexRay specification defines five sequences, a *transmission start sequence* (TSS), a *frame start sequence* (FSS), a *byte start sequence* (BSS), a *frame end sequence* (FES), and a *dynamic trailing sequence* (DTS), for encoding a frame. An encoded frame begins with a TSS and an FSS. A raw frame is split into bytes. A BSS is added to the front of each and every byte. An FES is added just after the last byte data of raw data. A DTS is also added after an FES of an encoded frame in a dynamic segment that will be explained in Section 2.2. **Figure 2** shows an encoded frame in a static segment that will be explained in Section 2.2. A TSS initiates connection setup through the FlexRay bus. A sender node generates a TSS that is composed of a continuous LOW for a period specified by a FlexRay parameter. An FSS appears after the TSS. An FSS compensates for a possible quantization error in the first BSS. An FSS is composed of a HIGH signal for a bit time. A BSS offers bit stream timing information to receiver nodes. The BSS shall be composed of both a HIGH signal for a bit time and a LOW signal for a bit time. Each byte of raw data shall be sent on the bus as an encoded byte sequence that consists of one BSS followed by eight bits of raw data. An FES marks the end of the last encoded byte of a frame. The FES shall be composed of both a LOW signal for a bit time and a HIGH signal for a bit time.

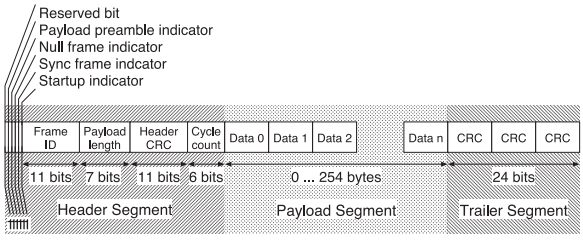


Fig. 1 Frame format in the FlexRay specification.

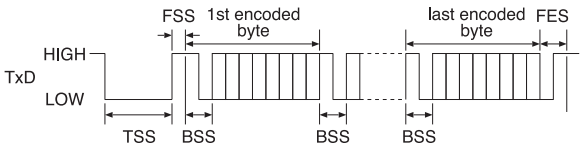


Fig. 2 Frame encoding in the FlexRay specification.

2.2 Timing Hierarchy

In the FlexRay specification, the *timing hierarchy* consists of four timing hierarchy levels: *communication cycle level*, *arbitration grid level*, *macrotick level*, and *microtick level*. **Figure 3** shows an overview of the timing hierarchy in the FlexRay specification.

A *communication cycle* is defined as one complete instance of the communication structure that is periodically repeated to comprise the media access method of the FlexRay communication system. A communication cycle consists of four different time segments: *static segment*, a *dynamic segment*, a *symbol window*, and a *network idle time* at the communication cycle level. The *static segment* is a period of time during which all frames are arbitrated and sent with the TDMA scheme. The TDMA scheme is generally a channel access method for shared channel networks. It temporally divides a channel into time slots each of which can be used only by a specified communication message. The FTDMA scheme is used to arbitrate transmissions in a priority-based fashion within a dynamic segment. A symbol window is a communication period during which the network is diagnosed with a symbol. A network idle time is a communication-free period and is utilized for adjusting time deviation. This paper focuses only on the static segment that is essential to realize hard real-time applications in automotive products.

The next lower level is the arbitration grid level. There exist two types of time grids: a *static slot* and a *minislot*. A static segment consists of one or more consecutive static slots. A dynamic segment consists of zero or more consecutive minislots.

The next lower level is the macrotick level. There exist only one time unit, that is a *macrotick*, at the macrotick level. A macrotick is the smallest granularity unit of the global time in the FlexRay communication system. A static segment or network idle time consists of one or more macroticks. A dynamic segment or symbol window consists of zero or more macroticks, which

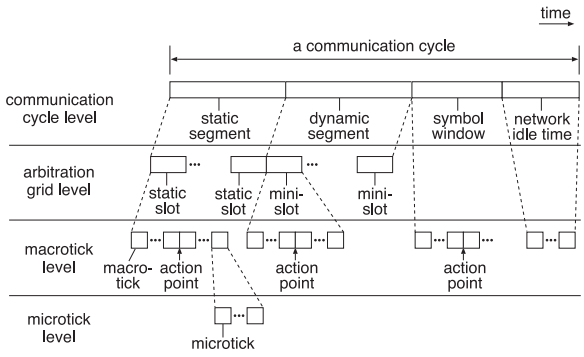


Fig. 3 Timing hierarchy within a communication cycle.

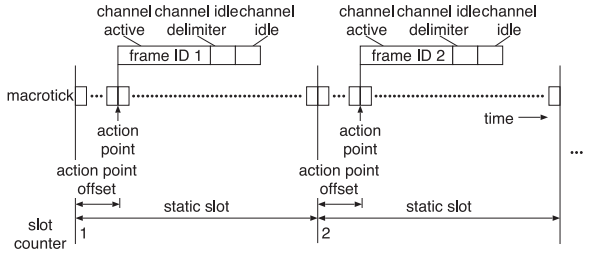


Fig. 4 Transmission timing within a static segment.

means that system designers may optionally use a dynamic segment and symbol window. An *action point* shown in Fig. 3 is the time at which a transmitter begins to transmit a FlexRay frame.

The microtick level is the lowest level of the timing hierarchy in the FlexRay specification. The macrotick level is defined by *microticks*. A microtick is a time unit of local time in each node.

2.3 Media Access Control

The FlexRay protocol specifies that media access control stands on a recurring communication cycle. As pointed out in Section 2.2, a communication cycle contains a static segment and a dynamic segment. The static segment offers the TDMA scheme. By contrast, the dynamic segment offers the FTDMA scheme.

We simply review the TDMA scheme for static segments. Each and every frame shall have its own frame identifier. The FlexRay network system has a slot counter whose value is initialized with zero and is incremented by one when the next time slot comes. A frame may be transmitted if the value of the slot counter and the frame identifier of the frame come to match. The value of the slot counter is reinitialized with zero after a static segment.

The FlexRay standard specifies that any static slot consists of an identical number of macroticks. This implies that the size of a frame must be a global constant within a static segment. **Figure 4** describes the above transmission mechanism within a static segment.

3. Bus Bandwidth Minimization

This section discusses the reduction of the operating frequency of a FlexRay communication bus in order to reduce the cost of fabricating an in-vehicular network. The reduction of the operating frequency of a FlexRay communication bus contributes to cost reduction as low operating frequency enables one to choose slow and cheap wire harness for producing a communication system. This section formalizes a bus bandwidth minimization in which we minimize the operating frequency of a communication bus under hard real-time constraints on the assumption that each and every signal is uniquely mapped to its own time slot of TDMA scheme.

3.1 Problem Formulation

A *communication signal* is defined as the behavioral characteristic of a node that sends a constant size of data periodically. An instance of a communication signal is defined as a *communication message*. A communication message requires to be divided into frames if the communication message is larger than a payload segment of a frame. We mainly focus on a TDMA network system in which sender node N requires to send S bit data every C time units and also requires to complete transmitting it within time D. We characterize the behavior of a sender node by a 4-tuple (N, C, D, S).

We now assume a TDMA system in which a set of signals $\mathbb{S} = \{s_1, s_2, \cdots, s_{N_{\text{sig}}}\}$ are sent via a static segment which consists of a set of static slots $\mathbb{T} = \{t_1, t_2, \cdots, t_{N_{\text{sig}}}\}$. We assume that each and every signal is uniquely mapped to its own time slot. Let the bandwidth be w bps for the TDMA network. The minimization of bandwidth w contributes to reducing the cost for network hardware. Signal s_i is characterized by a 4-tuple (N_i, C_i, D_i, S_i). We assume that signal s_i is always assigned to a single static slot t_i.

As shown in Fig. 1, a frame consists of a header segment of B_{h} bits, a payload segment of P bits, and a trailer segment of B_{t} bits. The size of a frame F in bits is

$$F = B_{\text{h}} + B_{\text{t}} + P. \tag{1}$$

The size of an encoded frame in bits is formulated as follows:

$$\begin{aligned} F_{\text{enc}} &= F + \left(TSS + FSS + \frac{F}{8} \cdot BSS + FES\right) \\ &= \left(1 + \frac{BSS}{8}\right) \cdot F + TSS + FSS + FES \\ &= \left(1 + \frac{BSS}{8}\right) \cdot P + O, \end{aligned} \tag{2}$$

$$O = \left(1 + \frac{BSS}{8}\right)(B_{\text{h}} + B_{\text{t}}) + TSS + FSS + FES, \tag{3}$$

where TSS, FSS, BSS, and FES are the numbers of bits for a TSS, an FSS, a BSS, and an FES respectively.

For simplicity we focus only on a static segment in calculating time for a communication cycle though a communication cycle may consist of four parts: a static segment, a dynamic segment, a symbol window, and a network idle time. We assume that a communication cycle consists of a static segment. Consequently, a communication cycle t_{cc} is equal to the time for all static slots and is formulated as follows.

$$t_{\text{cc}} = \frac{N_{\text{sig}} F_{\text{enc}}}{w}. \tag{4}$$

We send a communication message in the form of frames. **Figure 5** shows how a communication message is sent with frames. First of all, waiting time occurs before the first frame is sent with the corresponding static slot. Let the worst-case waiting time be equal to a communication cycle t_{cc} on the assumption that the time difference between the time at which a transmission request arrives and the time at which the first static slot begins is unknown. Time for sending frames follows the waiting time. The time for sending all frames is formulated as follows.

$$\left(\left\lceil \frac{S_i}{P} \right\rceil - 1\right) t_{\text{cc}} + \frac{F_{\text{enc}}}{w}. \tag{5}$$

The worst-case latency to transmit a communication message of signal i, l_i, is the summation of the worst-case waiting time and the time for sending all frames of a communication message as shown in the following equation.

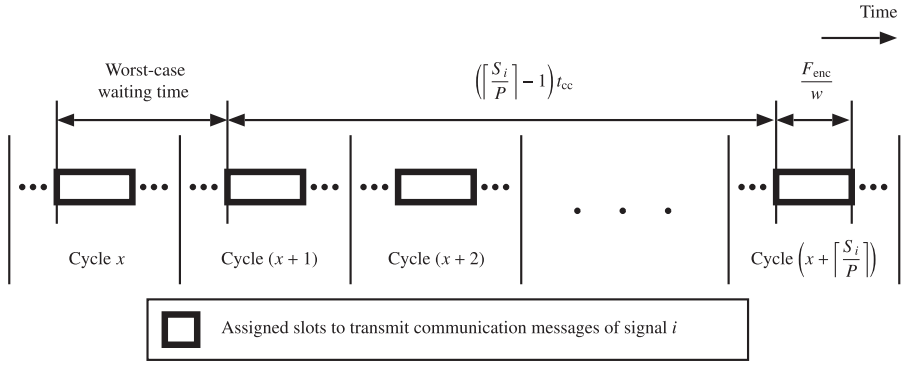


Fig. 5 A calculation model for transmission time of a communication message.

$$\begin{aligned} l_i &= t_{cc} + \left(\left\lceil \frac{S_i}{P} \right\rceil - 1\right) t_{cc} + \frac{F_{enc}}{w} \\ &= \left\lceil \frac{S_i}{P} \right\rceil t_{cc} + \frac{F_{enc}}{w} \\ &= F_{enc} \left(\left\lceil \frac{S_i}{P} \right\rceil N_{sig} + 1\right) \frac{1}{w}. \end{aligned} \tag{6}$$

The worst-case latency l_i must be less than or equal to D_i in order to satisfy the hard deadline constraint. From Eq. (6) the following constraint is introduced.

$$F_{enc} \left(\left\lceil \frac{S_i}{P} \right\rceil N_{sig} + 1\right) \frac{1}{w} \le D_i. \tag{7}$$

The mathematical model is finally built as follows.

Minimize the cost function w
subject to
(1) $F_{enc} \left(\left\lceil \frac{S_i}{P} \right\rceil N_{sig} + 1\right) \frac{1}{w} \le D_i$.
Variables
- w is a real variable.

3.2 Linearization

We assume that various bandwidths are available to use. A finite set of bandwidths $\mathbb{W} = \{W_1, W_2, \cdots, W_{N_{bw}}\}$ is given. We introduce the following binary variable x_j to indicate whether or not the bandwidth W_j is adopted.

$$x_j = \begin{cases} 1 & \text{if } W_j \text{ is the bandwidth of the bus,} \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

The bandwidth of a TDMA bus is now formulated as follows.

$$w = \sum_j W_j x_j. \tag{9}$$

A single bandwidth must be chosen for the TDMA bus and the following constraint is introduced.

$$\sum_j x_j = 1. \tag{10}$$

From Eqs. (9) and (10), the inverse of variable w is also formulated as follows.

$$\frac{1}{w} = \sum_j \frac{x_j}{W_j}. \tag{11}$$

From Eq. (11), Eq. (7) is transformed as follows.

$$F_{enc} \left(\left\lceil \frac{S_i}{P} \right\rceil N_{sig} + 1\right) \sum_j \frac{x_j}{W_j} \le D_i. \tag{12}$$

The integer linear programming (ILP) model is given as follows.

Minimize the cost function $w = \sum_j W_j x_j$
subject to
(1) $\sum_j x_j = 1$.
(2) $F_{enc} \left(\left\lceil \frac{S_i}{P} \right\rceil N_{sig} + 1\right) \sum_j \frac{x_j}{W_j} \le D_i$, $1 \le \forall i \le N_{sig}$.
Variable
- x_j is a binary variable, $1 \le \forall j \le N_{bw}$.

4. Payload Size Optimization

This section presents a bandwidth minimization problem in which an optimal payload size is sought out under hard real-time constraints so that the bandwidth of a FlexRay bus is minimized.

4.1 Problem Formulation

In the previous section we treated the payload size as a constant. In this section, we treat the payload size as a variable. We use notation p for the variable payload size for further bandwidth reduction of a FlexRay bus. The FlexRay standard clearly states that designers may use a payload size of an even number between 0 to 254 bytes [3]. It also states that all static slots consist of an identical number of macroticks. This indicates that the payload size is identical among all frames of communication messages. Payload size optimization probably reduces the bandwidth of a bus as it is expected to reduce the unused part of a payload segment. The size of a frame is now formulated in the same way for Eq. (1) as follows.

$$f = B_h + B_t + p. \tag{13}$$

The size of an encoded frame is now formulated in the same way for Eq. (2) as follows.

$$f_{enc} = \left(1 + \frac{BSS}{8}\right) \cdot p + O. \tag{14}$$

A communication cycle t_{cc} is formulated in the same way for Eq. (4) as follows.

$$t_{cc} = \frac{N_{sig}}{w} \left\{\left(1 + \frac{BSS}{8}\right) \cdot p + O\right\} \tag{15}$$

The worst-case latency to transmit a communication message of signal i, l_i' is formulated in the same way for Eq. (6) as follows.

$$l_i' = \left\{\left(1 + \frac{BSS}{8}\right) \cdot p + O\right\} \left(\left\lceil \frac{S_i}{p} \right\rceil N_{sig} + 1\right) \frac{1}{w} \tag{16}$$

A hard real-time deadline constraint is formulated in the same way for Eq. (7) as follows.

$$\left\{\left(1+\frac{BSS}{8}\right)\cdot p+O\right\}\left(\left\lceil\frac{S_i}{p}\right\rceil N_{\mathrm{sig}}+1\right)\frac{1}{w}\le D_i. \tag{17}$$

The mathematical programming model is given as follows.

Minimize the cost function w
subject to
(1) $\left\{\left(1+\frac{BSS}{8}\right)\cdot p+O\right\}\left(\left\lceil\frac{S_i}{p}\right\rceil N_{\mathrm{sig}}+1\right)\frac{1}{w}\le D_i$.
Variable
- w is a real variable.
- p is an integer variable.

Bound
- $p\in\{x|x=2y, 0\le y\le 127, y\in\mathbb{Z}\}$.

4.2 Linearization

We assume that a finite set of integers $\mathbb{P}=\{P_1,P_2,\cdots,P_{N_{\mathrm{pl}}}\}$ is given as available payload sizes in static segments. We introduce a new binary variable as follows.

$$y_k=\begin{cases}1 & \text{if the size of a payload segment is } P_k,\\ 0 & \text{otherwise.}\end{cases} \tag{18}$$

A single payload size must be chosen. Therefore, the following constraint is introduced.

$$\sum_k y_k=1. \tag{19}$$

The variable p is formulated using variable y_k as follows.

$$p=\sum_k P_k y_k. \tag{20}$$

From Eqs. (19) and (20) the inverse of variable p is formulated as follows.

$$\frac{1}{p}=\sum_k \frac{1}{P_k}y_k. \tag{21}$$

Nonlinear term $\left\lceil\frac{S_i}{p}\right\rceil$ is linearized using variables $y_1,\cdots,y_k$ as follows.

$$\left\lceil\frac{S_i}{p}\right\rceil=\sum_k\left\lceil\frac{S_i}{P_k}\right\rceil y_k. \tag{22}$$

From Eqs. (11), (20) and (22), constraint (17) is transformed as

$$\sum_j\frac{O}{W_j}x_j+\sum_{j,k}A_{i,j,k}x_j y_k\le D_i, \tag{23}$$

where

$$A_{i,j,k}=\frac{1}{W_j}\left\{ON_{\mathrm{sig}}\left\lceil\frac{S_i}{P_k}\right\rceil+P_k\left(\left\lceil\frac{S_i}{P_k}\right\rceil N_{\mathrm{sig}}+1\right)\left(1+\frac{BSS}{8}\right)\right\}. \tag{24}$$

We introduce a new binary variable $z_{j,k}$ to linearize $x_j y_k$ as follows.

$$z_{j,k}-x_j\le 0, \tag{25}$$
$$z_{j,k}-y_k\le 0, \tag{26}$$
$$z_{j,k}-x_j-y_k\ge -1. \tag{27}$$

The constraint (23) is finally transformed using the variable $z_{j,k}$ as follows.

$$\sum_j\frac{O}{W_j}x_j+\sum_{j,k}A_{i,j,k}z_{j,k}\le D_i \tag{28}$$

The ILP model is finally given as follows.

Minimize the cost function $w=\sum_j W_j x_j$
subject to
(1) $\sum_j x_j=1$.
(2) $\sum_k y_k=1$.
(3) $\sum_j\frac{O}{W_j}x_j+\sum_{j,k}A_{i,j,k}z_{j,k}\le D_i$.
(4) $A_{i,j,k}=\frac{1}{W_j}\left\{ON_{\mathrm{sig}}\left\lceil\frac{S_i}{P_k}\right\rceil+P_k\left(\left\lceil\frac{S_i}{P_k}\right\rceil N_{\mathrm{sig}}+1\right)\left(1+\frac{BSS}{8}\right)\right\}$.
(5) $z_{j,k}-x_j\le 0,\ 1\le\forall j\le N_{\mathrm{bw}},\ 1\le\forall k\le N_{\mathrm{pl}}$.
(6) $z_{j,k}-y_k\le 0,\ 1\le\forall j\le N_{\mathrm{bw}},\ 1\le\forall k\le N_{\mathrm{pl}}$.
(7) $z_{j,k}-x_j-y_k\ge -1,\ 1\le\forall j\le N_{\mathrm{bw}},\ 1\le\forall k\le N_{\mathrm{pl}}$.
Variable
- x_j is a binary variable.
- y_k is a binary variable.
- $z_{j,k}$ is a binary variable.

5. Experiment

This section quantitatively discusses the effectiveness of our bandwidth minimization approach.

5.1 Experimental Setup

We developed a program which produces an ILP model that is shown in the previous section. The file format of the ILP model is the LP format. We coded the program in the Perl language. We inputted the ILP model into the IBM ILOG CPLEX 12.3 [4] so that we obtained an optimal bus bandwidth and payload size. We run the CPLEX on a PC whose CPU was an Intel Core i7-X980 (3.33 GHz).

We adopted the SAE benchmark signal set which gave communication requirements in an automotive embedded system. Further detail in the SAE benchmark signal set should be referred to Kutlu's paper [5]. The SAE benchmark contains 53 types of signals. We hypothetically made 10 benchmark sets using the SAE benchmark as shown in **Table 1**. In the table, benchmark set b_n contains n copies of the SAE benchmark signal set.

We adopted Park's network parameters as shown in **Table 2** [6]. We assumed that there was no channel idle time just after a channel idle delimiter.

We assumed that one of the the even numbers of bytes between 0 and 254 was available to use as the size of a payload segment of a frame. We also assumed that the bandwidths, 1, 2, 3, 4, 5, 6, 7, 8, 9, and 10 MHz, were available to use as the bandwidth of a FlexRay bus.

Table 1 Benchmark signal sets.

Benchmark set name	# signals
$b_{1/4}$	13
$b_{2/4}$	26
$b_{3/4}$	39
b_1 (the SAE benchmark)	53
b_2	106
b_3	159
b_4	212
b_5	265
b_6	318
b_7	371

Table 2 Network parameters for experiment.

	Length
TSS	9 bits/frame
FSS	1 bit/frame
BSS	2 bits/frame byte
FES	2 bits/frame
Idle delimiter	11 bits/frame
Header w/o BSS	5 B/frame
Header w BSS	45 bits/frame
Trailer w/o BSS	3 B/frame
Trailer w BSS	27 bits/frame

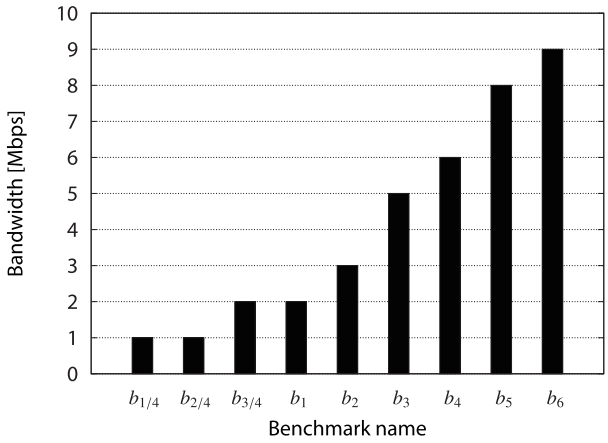


Fig. 6 Bandwidths for benchmark signal sets.

5.2 Experimental Results

We optimized a payload size for all benchmark signal sets given in Table 1. All optimization processes were finished within a second. It was infeasible to obtain a solution for benchmark b_7. Benchmark b_7 would require a bandwidth more than 10 Mbps on the assumption that each and every signal is uniquely mapped to its own time slot of the TDMA scheme. **Figure 6** shows the bandwidths obtained by solving ILP models. The more signals a signal set includes, the higher bandwidth it requires. Figure 6 means that the bandwidth required in an application varies depending on a signal set. For example, the required bandwidth is not 10 Mbps, which is a typical bandwidth of a FlexRay bus, but 2 Mbps for benchmark b_1! Our design approach achieved an 8 Mbps lower bandwidth for benchmark signal set b_1 than the typical bandwidth the FlexRay specification assumes. Our design methodology is very effective in reducing the operating frequency of a bus by building a FlexRay network of a small number of signals.

6. Related Work

Several design approaches have been proposed for FlexRay networking systems [6], [7], [8]. **Table 3** summarizes the related work. Table 3 shows whether or not each approach considers design objectives: bandwidth, the number of signals that meet deadline constraints, extensibility for developing future products, and the bandwidth utilization rate.

Park et al. focused on maximizing the number of signals that satisfy their own deadline constraint [6]. They optimized the size of a payload segment and the time for a communication cycle so that they maximized the number of signals whose deadline was met under the condition that the bandwidth was a constant. They mainly assumed a situation that all of the given deadlines cannot be satisfied with a communication bus. In fact, their approach is regarded as a soft real-time one as it does not necessarily guarantee all deadlines. Their method can seek a schedule for any bandwidth of a bus because low bandwidth simply results in a low number of satisfied signals. Their method is incapable of minimizing the bandwidth of a bus under hard real-time constraints because the reduction of bandwidth simply results in a compromised number of satisfied signals. It depends on an optimization result whether or not the deadline constraint of each signal is satisfied.

Table 3 Related work.

	Bandwidth (cost)	# signals meeting deadline constraints	Extensibility	Bandwidth utilization rate
Park et al.	No	Yes	No	No
Zeng et al.	No	No	Yes	No
Schmidt et al.	No	No	No	Yes
Our approach	Yes	No	No	No

Schmidt et al. focused on a bandwidth utilization rate on the assumption that the operating frequency of a bus is constant [7]. They maximized a bandwidth utilization rate so that they could transmit as much data as possible. Schmidt's idea is close to ours. Their approach, however, is different from ours in terms of regarding the bandwidth as a constant. While the bandwidth of a bus is treated as a constant, the fabrication cost of wire harness cannot be reduced.

Zeng et al. introduced a measurement, i.e. the *extensibility* of a communication mechanism that is defined as the number of free communication slots [8]. They insisted that maximizing the extensibility measurement alleviates a future task in developing a new car. We think that the design cost reduced by Zeng's approach is smaller than the fabrication cost reduced by our approach especially in mass production. Zeng's approach would be effective in a circumstance that the design cost is dominant. Their approach is probably effective in producing a small volume of cars.

7. Concluding Remark

We proposed a design methodology in which system designers minimize the operating frequency of a bus under hard real-time constraints on the assumption that each and every signal is uniquely mapped to its own static slot within a communication cycle. Our design methodology contributes to cost reduction because system designers can choose a slower and cheaper wire harness. Our experimental results showed that our optimization technique lowered the operating frequency of a bus for given benchmark signal sets. Compared with the typical bandwidth of the FlexRay, that is 10 Mbps, our design methodology achieved one-fifth of the typical bandwidth for the SAE benchmark. The fact is quite instructive to future revision in the next FlexRay specification standard that the reduction of an operating frequency of a bus helps system designers to reduce the cost of their system. The next FlexRay standard should specify that system designers are given a sufficient number of operating frequencies of a bus to reduce their fabrication cost. Our design methodology accelerates the FlexRay system to be applied to mass-produced cars whose fabrication cost must be competitive. Our future work includes

the extension of our design paradigm to a cycle-multiplexing mechanism that assumes that system designers may map a signal to one or more static slots within 64 communication cycles.

References

[1] CAN: Control Area Network, available from ⟨http://www.can.bosch.com/⟩.

[2] Cena, G. and Valenzano, A.: Achieving round-robin access in controller area networks, *IEEE Trans. Ind. Electron.*, Vol.49, No.6, pp.1202–1213 (Dec. 2002).

[3] FlexRay Consortium: FlexRay communications system protocol specification version 2.1 revision A (Dec. 2005).

[4] The IBM Corp.: The IBM ILOG CPLEX Optimization Studio V12.3 documentation, available from ⟨http://publib.boulder.ibm.com/infocenter/cosinfoc/v12r3/⟩.

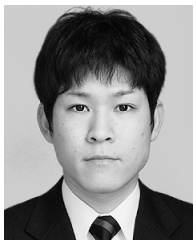
[5] Kutlu, A., Ekiz, H. and Powner, E.T.: Performance analysis of MAC protocols for wireless control area network, *Proc. International Symposium on Parallel Architectures, Algorithms, and Networks*, pp.494–499 (June 1996).

[6] Park, I. and Sunwoo, M.: FlexRay network parameter optimization method for automotive applications, *IEEE Trans. Industrial Electronics*, Vol.58, No.4, pp.1449–1459 (2011).

[7] Schmidt, K. and Schmidt, E.G.: Message scheduling for the FlexRay protocol: The static segment, *IEEE Trans. Vehicular Technology*, Vol.58, No.5, pp.2170–2179 (June 2009).

[8] Zeng, H., Matale, M.D., Ghosal, A. and Sangiovanni-Vincentelli, A.: Schedule optimization of time-triggered systems communicating over the FlexRay static segment, *IEEE Trans. Industrial Informatics*, Vol.7, No.1, pp.1–17 (Feb. 2011).

Makoto Sugihara was born in Tokyo, Japan in 1974. He received his B.E., M.E. and Ph.D. degrees in Computer Science and Communication Engineering from Kyushu University, Japan in 1996, 1998 and 2001, respectively. He joined the Fujitsu Laboratories, Japan, in 1998. From 2002 to 2003, he was a Visiting Researcher at Duke University, NC, USA. From 2003 to 2007, he was a Researcher at the ISIT, Fukuoka, Japan, and was also a Visiting Associate Professor at Kyushu University, Fukuoka, Japan. From 2007 to 2009, he was a Lecturer at Toyohashi University of Technology, Aichi, Japan. From 2009 to 2011, he was an Associate Professor at Toyohashi University of Technology, Aichi, Japan. He is currently an Associate Professor at the System LSI Research Center, Kyushu University, Fukuoka, Japan. His research interests include design methodologies for VLSI chips and embedded systems. He is a member of IEEE and IEEE Computer Society, IEICE, and IPSJ. He received the IPSJ 40th Anniversary Best Paper Award, the Hiroshi Ando Memorial Award, and the Funai Information Science Encouragement Award in 2001, 2008, and 2009 respectively.



Akihito Iwanaga was born in Yamaguchi, Japan in 1988. He received his B.E. degree in Computer Science and Electronic Engineering from the National Institution for Academic Degrees and University Evaluation, Japan in 2011. He is currently a student at the Department of Automotive Science, Graduate School of Integrated Frontier Science, Kyushu University. His research activities include designing a hard real-time system and an automotive network. He is a member of IPSJ and JSAE.