CiA® 601 Draft Standard

Node and system design

Part 3: System design recommendation

Version: 1.0.0
06 September 2019

HISTORY

Date	Changes
2018-07-24	*Publication of Version 1.0.0* as draft standard proposal
2019-09-06	*Publication of Version 1.0.0* as draft standard - No change in content

General information on licensing and patents

CAN in Automation (CiA) calls attention to the possibility that some of the elements of this CiA specification may be subject of patent rights. CiA shall not be responsible for identifying any or all such patent rights.

Because this specification is licensed free of charge, there is no warranty for this specification, to the extent permitted by applicable law. Except when otherwise stated in writing the copyright holder and/or other parties provide this specification “as is” without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The entire risk as to the correctness and completeness of the specification is with you. Should this specification prove failures, you assume the cost of all necessary servicing, repair or correction.

Trademarks

CANopen® and CiA® are registered community trademarks of CAN in Automation. The use is restricted for CiA members or owners of CANopen® vendor ID. More detailed terms for the use are available from CiA.

CAN in Automation e. V.
Kontumazgarten 3
DE - 90429 Nuremberg, Germany
Tel.: +49-911-928819-0
Fax: +49-911-928819-79
Url: www.can-cia.org
Email: headquarters@can-cia.org

CONTENTS

1 Scope

This set of documents recommends the usage of CAN FD hardware implementations. It consist of the following parts:

- Part 1: Physical interface implementation
- Part 2: CAN controller interface specification
- Part 3: System design recommendation
- Part 4: Signal improvement

This part provides recommendations for CAN FD network design, especially, bit-timing setting rules and guidelines are given.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

/ISO11898-1/ ISO 11898-1:2015, Road vehicles – Controller area network – Part 1: Data link layer and physical signaling

/ISO11898-2/ ISO 11898-2:2016, Road vehicles – Controller area network – Part 2: High-speed medium access unit

/CiA601-1/ CiA 601-1 version 2.0.0, CAN FD Node and system design – Part 1: Physical interface implementation

/CiA601-2/ CiA 601-2 version 1.0.0, CAN FD Node and system design – Part 2: CAN controller interface specification

/CiA601-4/ CiA 601-4 version 2.0.0 (under development), CAN FD Node and system design – Part 4: Signal improvement

3 Terms and definitions

For the purpose of this document, the following terms and definitions and those given in /ISO11898-1/, /ISO11898-2/, /CiA601-1/, and /CiA601-2/ apply.

3.1
arbitration phase
phase in which the nominal bit time is used

3.2
CAN controller
hardware representation of a CAN node compliant with /ISO11898-1/, which is implemented on-chip the host controller or as stand-alone controller

3.3
data bit rate
number of bits per time during data phase, independent of the bit encoding/decoding

3.4
data bit time
duration of one bit in data phase

3.5
data phase
phase in which the data bit time is used

3.6
nominal bit rate
number of bits per time during arbitration phase, independent of the bit encoding/decoding

3.7
nominal bit time
duration of one bit in arbitration phase

3.8
system
network system comprising in minimum two CAN FD nodes, a wiring harness, and optional physical layer components such as connectors, termination resistors, etc.

3.9
transceiver
implementation compliant to /ISO11898-2/ comprising one or more physical media attachments

3.10
topology
all physical layer components (e.g. cabling, CMC, termination resistors, etc.), which are connected to the CAN_H or CAN_L pins of the transceiver, except the transceiver itself

4 Symbols and abbreviated terms

For the purpose of this document, the following symbols and abbreviated terms and those given in /ISO11898-1/, /ISO11898-2/, /CiA601-1/, and /CiA601-2/ apply.

A1	bit asymmetry 1
A2	bit asymmetry 2
BRP	bit rate prescaler
CAN	controller area network
CAN_H	CAN high signal
CAN_L	CAN low signal
CMC	common-mode choke
CRC	Cyclic Redundancy Check
EMC	electromagnetic compatibility
ESD	electro static discharge
SM1	safety margin 1
SM2	safety margin 2
SP	Sample Point
SSP	Secondary Sample Point
tq	time quantum or time quanta
mtq	minimum time quantum or minimum time quanta
PCB	printed circuit board
PM1	phase margin 1 of receiving node
PM1TX	phase margin 1 of transmitting node
PM2	phase margin 2 of receiving node
PM2TX	phase margin 2 of transmitting node

PVC polyvinyl chloride
Recom Recommendation
TD transmitter delay
TDC transmitter delay compensation

5 Variable definitions

This clause defines the variables used in this document.

The following subscripts are used:

- A: denotes variables of arbitration phase, e.g. tq_A
- D: denotes variables of data phase, e.g. tq_D

The following bit-timing configuration properties are dimensionless:

- BRP_D, BRP_A bit rate prescaler

The following bit-timing configuration properties are given in multiples of nanoseconds:

- SJW_D, SJW_A synchronization jump width
- $PS1_D$, $PS1_A$ phase segment 1
- $PS2_D$, $PS2_A$ phase segment 2
- BT_D, BT_A bit time duration
- tq_D, tq_A time quantum (plural: time quanta)
- *mtq* minimum time quantum (plural: minimum time quanta)
- *SSP offset* secondary sample point offset

The equation (1) specifies the time quantum for the data phase and the equation (2) for the arbitration phase.

$$tq_D = \frac{BRP_D}{f_{nom}} \quad (1)$$

$$tq_A = \frac{BRP_A}{f_{nom}} = tq_D \cdot \frac{BRP_A}{BRP_D} \quad (2)$$

The following bit-timing configuration properties are given in multiples of tq:

- sjw_D, sjw_A (re-) synchronization jump width
- $ps1_D$, $ps1_A$ phase segment 1
- $ps2_D$, $ps2_A$ phase segment 2
- bt_D, bt_A bit time duration

The equation (3) specifies the data bit time, and the equation (4) specifies the nominal bit time.

$$BT_D = bt_D \cdot tq_D \quad (3)$$

$$BT_A = bt_A \cdot tq_A \quad (4)$$

The following variables are CAN clock frequency related variables:

- f_{nom} nominal frequency of CAN clock
- f_{CAN} actual frequency of the CAN clock
- df maximum CAN clock tolerance
- df_{used} specified tolerance of the present CAN clock

6 CAN clock

6.1 Recommended frequencies

For CAN nodes that make use of bit rate switching the following CAN clock frequencies are suggested for the purpose to facilitate good interoperability:

- 20 MHz
- 40 MHz
- 80 MHz

NOTE It is recommended that according to Recom3 in 7.4 a node uses the highest available clock frequency for CAN communication because this increases robustness. Therefore consider the highest intended CAN FD data bit rate when choosing from these recommended frequencies.

6.2 CAN clock tolerance

6.2.1 Background

To calculate the maximum CAN clock tolerance (*df*), which is tolerated by the CAN protocol, the setup shown in Figure 1 is used.

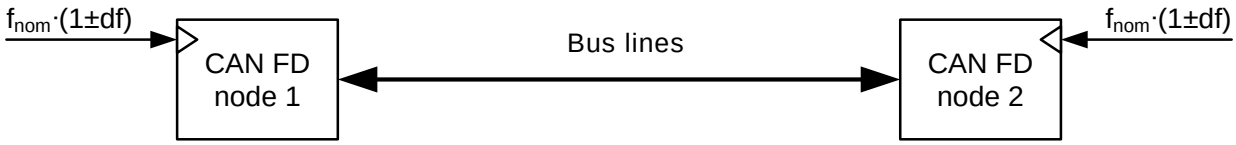

Figure 1 – Setup used for clock tolerance calculation

Just two CAN nodes are connected to a network. In both nodes the CAN controller uses a CAN clock with a frequency of f_{CAN}, while f_{CAN} is in the range

$$f_{nom} \cdot (1 - df) \leq f_{CAN} \leq f_{nom} \cdot (1 + df) \quad (5)$$

df is the maximum relative tolerance of the CAN clock around its nominal frequency. The maximum difference between the two CAN clocks is

$$f_{nom} \cdot (1 + df) - f_{nom} \cdot (1 - df) = f_{nom} \cdot 2df \quad (6)$$

For the calculation of *df*, an ideal system is assumed, in which the tolerance of the CAN clock is the only source of error. This means, that a system without any physical layer effects is assumed.

NOTE Unfortunately, the expressions “CAN clock tolerance” and “oscillator tolerance” are both used in the CAN community to express the same: the tolerance of the CAN clock used by a CAN controller. Also /ISO11898-1/ talks about “tolerance range of the oscillator frequencies”. However, the expression “CAN clock tolerance” is much more precise and therefore is used in this document. The CAN clock is typically derived from some other clocks e.g. from the clock provided by a ceramic resonator or a quartz oscillator. Consequently, in most systems the actual present “CAN clock tolerance” is worse than the “oscillator tolerance”.

6.2.2 Conditions from /ISO11898-1/ and CAN clock tolerance calculation

In /ISO11898-1/ the five conditions (see (7) to (11)) are defined for the actual tolerance of the CAN clock (df_{used}). These conditions allow calculating the theoretical bounds of the CAN clock tolerance as a function of the CAN bit time configurations. The conditions assume also the ideal system, in which there is no physical layer effect at all.

The conditions are derived based on worst-case scenarios, which rarely happen. This means during normal operation (i.e. non worst-case scenario) the system often can tolerate a clock difference that is higher than the calculated one. For details of the worst-case scenarios see bibliography [1].

When the following conditions are met, reliable CAN FD communication is possible from a bit-timing perspective:

- Conditions 1, 2, 3, 4, and 5 apply for CAN FD with bit rate switching (i.e. with higher data bit rate).
- Conditions 1 and 2 apply for CAN FD without bit rate switching.

The five conditions are given in /ISO11898-1/ 11.3.2.5 (3) to (7).

Condition 1: Resynchronization

$$df < \frac{sjw_A}{2 \cdot 10 \cdot bt_A} \tag{7}$$

Condition 2: Sampling of bit after error flag

$$df < \frac{\min\left(ps1_A, ps2_A\right)}{2 \cdot \left[13 \cdot bt_A - ps2_A\right]} \tag{8}$$

Condition 3: Resynchronization

$$df < \frac{sjw_D}{2 \cdot 10 \cdot bt_D} \tag{9}$$

Condition 4: Sampling of bit after error flag

$$df < \frac{\min\left(ps1_A, ps2_A\right)}{2 \cdot \left[\left(6 \cdot bt_D - ps2_D\right) \cdot \frac{BRP_D}{BRP_A} + 7 \cdot bt_A\right]} \tag{10}$$

Condition 5: Switching from arbitration phase to data phase

$$df < \frac{sjw_D - \max\left(0;\ \frac{BRP_A}{BRP_D} - 1\right)}{2 \cdot \left[\left(2 \cdot bt_A - ps2_A\right)\frac{BRP_A}{BRP_D} + ps2_D + 4 \cdot bt_D\right]} \tag{11}$$

The bit-timing parameters directly influence what CAN clock deviations are tolerated in a network. Consequently, careful selection of these parameters is essential to enable best communication performance.

To calculate *df* for a given bit-timing configuration, a result for each of the five conditions (7) to (11) need to be calculated. The *df* is equal to the smallest of the five results, as it fulfills all five conditions.

The “CAN FD bit-timing configuration and evaluation tool” spreadsheet provided with this document helps performing the calculation.

NOTE Condition 5 (see (11)) can deliver a negative value. This means that the bit-timing configuration combination of arbitration phase and data phase is not operational, because the potential initial phase error when switching to the data phase is too large.

6.2.3 Tolerance recommendation

For a reliable CAN FD operation, the tolerance df_{used} of the present CAN clock should be lower than the maximum tolerance *df* defined according to the conditions in /ISO11898-1/ (see 6.2.2):

$$df_{used} \leq df \quad (12)$$

For automotive use cases of CAN FD with bit rate switching, the actual tolerance df_{used} of the CAN clock is recommended to be:

$$df_{used} \leq \pm 0{,}3\ \% \quad (13)$$

The value df_{used} includes the initial tolerance, lifetime aging, and temperature drift. It does not include the jitter of individual clock edges. This means practically, that it is recommended to use a CAN clock with a lower tolerance than maximally tolerated by /ISO11898-1/. This is done to increase the margin for physical layer effects, which cannot be improved by device or system design.

6.3 Data bit rates

Table 1 lists all configurable data bit rates when the *BRP* is set to 1, and covers the configuration range of the bit-timing parameters recommended by /CiA601-2/. The timing parameters are the parameters written into the configuration register(s) to configure the CAN bit-timing.

The timing parameter ranges recommended by /CiA601-2/ enable data bit times with a length of 4 tq to 49 tq. The table lists the bit rates for the CAN clock frequencies recommended in 6.1.

NOTE The time segments' minimum configuration ranges specified in /ISO11898-1/ are subsets of the CiA recommendation and enables bit times with a length of 4 tq to 25 tq.

For a better overview Figure 2 shows the content of Table 1 as a diagram. At each data bit rate the SP can be positioned in a wide range.

Figure 2 visualizes this by showing for each number of "tq per bit time" the minimum (SP_{min}) and maximum (SP_{max}) SP position that is configurable. In the following two cases SP_{min} and SP_{max} are equal, i.e. only one SP position is possible.

- Case 1: All timing parameters (see /CiA601-2/) have the minimum possible value.
- Case 2: All timing parameters (see /CiA601-2/) have the maximum possible value.

Table 1 – Possible data bit rates

tq per bit time	CAN clock (tq in ns)						Data phase SP position [%]	
	20 MHz (50 ns)		40 MHz (25 ns)		80 MHz (12,5 ns)			
	Bit rate (fraction) [Mbit/s]	Bit rate (decimal) [Mbit/s]	Bit rate (fraction) [Mbit/s]	Bit rate (decimal) [Mbit/s]	Bit rate (fraction) [Mbit/s]	Bit rate (decimal) [Mbit/s]	Minimum	Maximum
4	1000/200	5,00	1000/100	10,00	10000/500	20,00	75	75
5	1000/250	4,00	1000/125	8,00	10000/625	16,00	60	80
6	1000/300	3,33	1000/150	6,67	10000/750	13,33	50	83
7	1000/350	2,86	1000/175	5,71	10000/875	11,43	43	86
8	1000/400	2,50	1000/200	5,00	10000/1000	10,00	38	88
9	1000/450	2,22	1000/225	4,44	10000/1125	8,89	33	89
10	1000/500	2,00	1000/250	4,00	10000/1250	8,00	30	90
11	1000/550	1,82	1000/275	3,64	10000/1375	7,27	27	91
12	1000/600	1,67	1000/300	3,33	10000/1500	6,67	25	92
13	1000/650	1,54	1000/325	3,08	10000/1625	6,15	23	92

tq per bit time	CAN clock (tq in ns)						Data phase SP position [%]	
	20 MHz (50 ns)		40 MHz (25 ns)		80 MHz (12,5 ns)			
	Bit rate (fraction) [Mbit/s]	Bit rate (decimal) [Mbit/s]	Bit rate (fraction) [Mbit/s]	Bit rate (decimal) [Mbit/s]	Bit rate (fraction) [Mbit/s]	Bit rate (decimal) [Mbit/s]	Minimum	Maximum
14	1000/700	1,43	1000/350	2,86	10000/1750	5,71	21	93
15	1000/750	1,33	1000/375	2,67	10000/1875	5,33	20	93
16	1000/800	1,25	1000/400	2,50	10000/2000	5,00	19	94
17	1000/850	1,18	1000/425	2,35	10000/2125	4,71	18	94
18	1000/900	1,11	1000/450	2,22	10000/2250	4,44	17	94
19	1000/950	1,05	1000/475	2,11	10000/2375	4,21	16	95
20	1000/1000	1,00	1000/500	2,00	10000/2500	4,00	20	95
21	1000/1050	0,95	1000/525	1,90	10000/2625	3,81	24	95
22	1000/1100	0,91	1000/550	1,82	10000/2750	3,64	27	95
23	1000/1150	0,87	1000/575	1,74	10000/2875	3,48	30	96
24	1000/1200	0,83	1000/600	1,67	10000/3000	3,33	33	96
25	1000/1250	0,80	1000/625	1,60	10000/3125	3,20	36	96
26	1000/1300	0,77	1000/650	1,54	10000/3250	3,08	38	96
27	1000/1350	0,74	1000/675	1,48	10000/3375	2,96	41	96
28	1000/1400	0,71	1000/700	1,43	10000/3500	2,86	43	96
29	1000/1450	0,69	1000/725	1,38	10000/3625	2,76	45	97
30	1000/1500	0,67	1000/750	1,33	10000/3750	2,67	47	97
31	1000/1550	0,65	1000/775	1,29	10000/3875	2,58	48	97
32	1000/1600	0,63	1000/800	1,25	10000/4000	2,50	50	97
33	1000/1650	0,61	1000/825	1,21	10000/4125	2,42	52	97
34	1000/1700	0,59	1000/850	1,18	10000/4250	2,35	53	97
35	1000/1750	0,57	1000/875	1,14	10000/4375	2,29	54	94
36	1000/1800	0,56	1000/900	1,11	10000/4500	2,22	56	92
37	1000/1850	0,54	1000/925	1,08	10000/4625	2,16	57	89
38	1000/1900	0,53	1000/950	1,05	10000/4750	2,11	58	87
39	1000/1950	0,51	1000/975	1,03	10000/4875	2,05	59	85
40	1000/2000	0,50	1000/1000	1,00	10000/5000	2,00	60	83
41	1000/2050	0,49	1000/1025	0,98	10000/5125	1,95	61	80
42	1000/2100	0,48	1000/1050	0,95	10000/5250	1,90	62	79
43	1000/2150	0,47	1000/1075	0,93	10000/5375	1,86	63	77
44	1000/2200	0,45	1000/1100	0,91	10000/5500	1,82	64	75
45	1000/2250	0,44	1000/1125	0,89	10000/5625	1,78	64	73
46	1000/2300	0,43	1000/1150	0,87	10000/5750	1,74	65	72
47	1000/2350	0,43	1000/1175	0,85	10000/5875	1,70	66	70
48	1000/2400	0,42	1000/1200	0,83	10000/6000	1,67	67	69
49	1000/2450	0,41	1000/1225	0,82	10000/6125	1,63	67	67

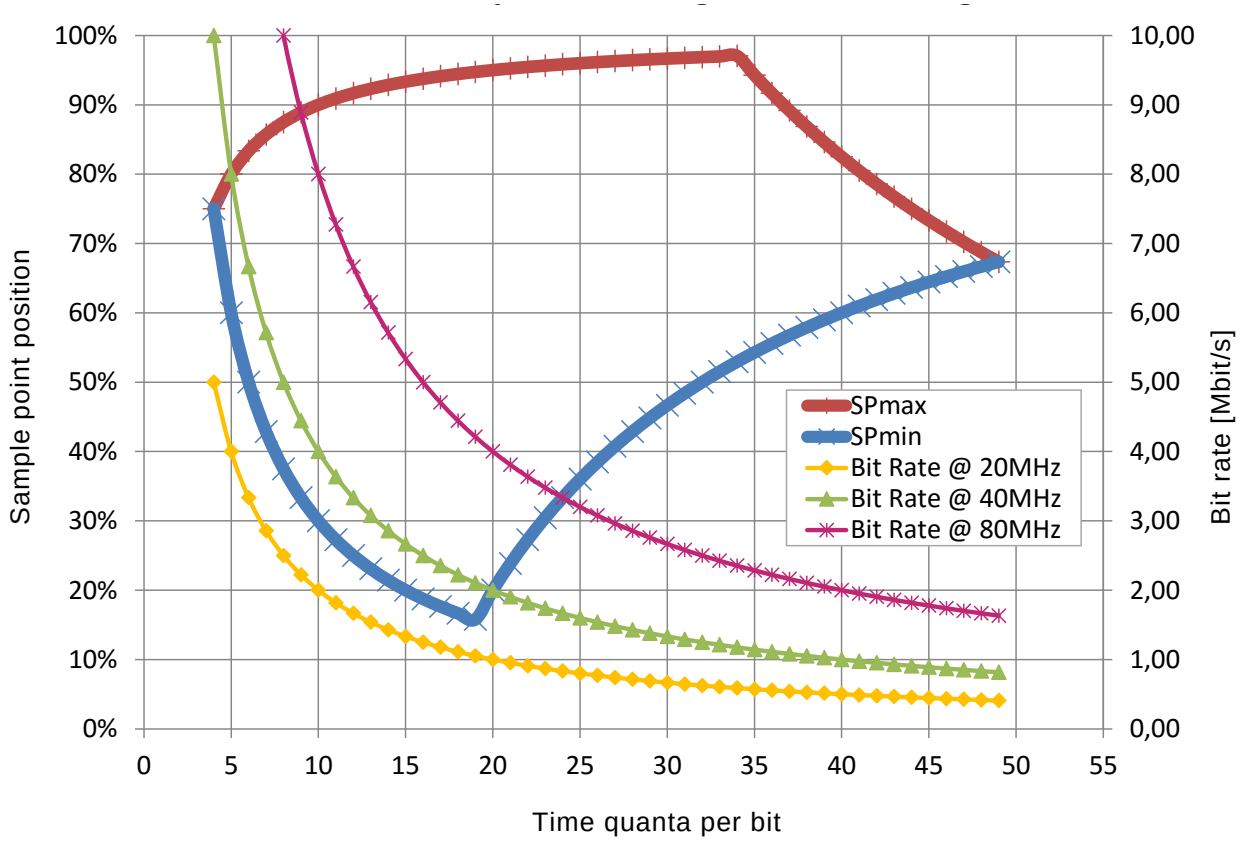

Figure 2 – Possible data bit rates and SP positions (min/max); this figure visualizes Table 1

7 Bit-timing configuration

7.1 Overview

This clause focuses on the CAN bit-timing configuration. Two uses cases exist.

- CAN FD without bit rate switching

  In this use case, the nominal bit-timing is used during the whole CAN data frame, like in Classical CAN. The arbitration phase is defined in /ISO11898-1/ as the part of the CAN frame in which the nominal bit-timing is used.

- CAN FD with bit rate switching

  In this use case two bit-timings are used in a CAN data frame: the nominal bit-timing for the arbitration phase and the data bit-timing for the data phase.

The CAN bit-timing configuration has two aspects. The system designer should consider both for an optimal result. These aspects are relevant for both use cases.

- Parameter choice

  The nominal and the data bit-timing configuration influence each other. Consequently, the large number of required bit-timing parameters should be chosen carefully to achieve an optimal result. Necessary recommendations for the bit-timing configuration are given in 7.4.

- SP positioning

  SP positioning means optimizing the robustness of the system by adapting it to the used physical layer properties.

  - Find optimal arbitration phase SP.
  - Find optimal data phase SP.
  - Find optimal SSP (only available, if TDC is used in the data phase).

7.2 Bit-timing configuration guideline

7.2.1 Nominal bit-timing configuration

To configure the nominal bit-timing, the following steps need to be done:

- Choose an initial nominal bit-timing configuration and consider the recommendations for the parameter choice in 7.4.
- Adapt the position of the arbitration phase SP as described in 7.6.

7.2.2 Data bit-timing configuration

To configure the data bit-timing, the following steps need to be done:

- Choose an initial data bit-timing configuration and consider the recommendations for the parameter choice in 7.4. If necessary adapt the nominal bit-timing configuration parameters. This initial data bit-timing configuration should be optimized later. For all data bit rates a data phase SP of 70 % is a recommended starting point.
- Determine the asymmetries in the system as described in 7.7.2.
- Evaluate, if the initial data bit-timing configuration is functional in the system setup and then optimize the data phase SP position. 7.7.5 describes both evaluation and SP optimization. The spreadsheet provided with this document helps to perform the evaluation and optimization.

- In case higher bit rate or improved robustness is required, the bit asymmetries in the system needs to be reduced and the PMs need to be optimized. These goals are achievable with the hints given in 7.7.5.2 and 7.7.7.

7.3 Relationships and dependencies

The following list of relationships and dependencies provides background information regarding the CAN bit-timing configuration:

- With each resynchronization, a receiving CAN node can correct a phase error of sjw_D in the data phase and sjw_A in the arbitration phase.
- The larger the ratio sjw_D/BT_D, the larger the resulting CAN clock tolerance in the data phase. The same holds also for the arbitration phase with sjw_A/BT_A.
- The absolute number of resynchronizations per unit of time increases towards higher bit rates. However, the absolute value of sjw_D or sjw_A decreases proportionally with the bit time. In other words, a higher bit rate leads to more, but smaller resynchronizations.
- A CAN node performs the bit rate switching at the SP of the BRS bit and CRC delimiter bit.
- All three available SPs are independent of each other: arbitration phase SP, data phase SP, and data phase SSP. They can be chosen independently.
- “Propagation time segment” of the bit time
    - Arbitration phase: The propagation time segment is mainly used to account for the worst-case round trip time between two CAN nodes. This includes any delays between two CAN controllers. Further, this segment also accounts for the time of a bit in which the bit signal cannot be considered as stable, e.g. due to ringing.
    - Data phase: In the data phase, there is no need to account for delay times. This segment is used to account for the time of a bit in which the bit signal cannot be considered as stable, e.g. due to ringing.

7.4 CAN bit-timing parameter choice

Many bit-timing parameters are required to configure the nominal and data bit-timing. To achieve a reliable CAN communication the dependencies between the bit-timing parameters should be considered.

The recommendations given in this clause cover only the parameter choice aspect. These recommendations reflect the current state of the art. Each recommendation is followed by an explanation and reasoning. Table 2 shows which recommendation is relevant for which use case.

Table 2 – Mapping between recommendations and use cases

Use case	Relevant recommendations
Classical CAN	Recom1, Recom2
CAN FD without bit rate switching	Recom1, Recom2
CAN FD with bit rate switching	All recommendations

- Recom1: Choose BRP_A and BRP_D as low as possible.
    - A lower *BRP* leads to shorter *tq*.
    - Shorter *tq* allows configuring a bit time with a higher resolution.

  - Advantages of a bit-timing configuration with high resolution are following.
    - The SP can be placed more accurately and therewith closer to the optimal position.
    - The size of the sync segment is reduced. This facilitates that receiving nodes synchronize more accurately to the sender, which is beneficial for robustness.

- Recom2: Choose sjw_A as large as possible.
  - The maximum possible value is $sjw_A = \min(ps1_A, ps2_A)$.
  - During arbitration phase, a large sjw_A allows a CAN node to resynchronize quickly on the leading transmitting node.
  - If the CAN controller does not allow configuring the maximum possible sjw_A because the value range for sjw_A is limited in the CAN controller, then consider increasing BRP_A. This allows configuring a larger sjw_A, unless this can impact phase error tolerance in an unwanted way, see Recom4.

- Recom3: Choose the highest available CAN clock frequency.
  - Recommendations for the CAN clock frequency are provided in 6.1.
  - A higher CAN clock frequency enables a shorter *tq*.
  - The reasoning why shorter *tq* should be preferred is given in Recom5.

- Recom4: Set $BRP_A = BRP_D$ (when not possible set $BRP_A \leq BRP_D$).
  - Condition 5 (see equation (11)) covers the bit rate switching and is usually the most significant.
  - $BRP_A = BRP_D$ leads to $tq_A = tq_D$. This prevents that during bit rate switching inside the frame an existing quantization error can transform into a phase error. The numerator of condition 5 also reflects this property.
  - $BRP_A \leq BRP_D$ leads to potential transformation of a quantization error to a phase error during bit rate switching in the CRC delimiter bit. As the bit rate switching from data to arbitration phase is not critical, the introduced phase error is typically tolerable.

- Recom5: Configure all CAN nodes to have the same arbitration phase SP and the same data phase SP.
  - The simplest way to achieve this is to use the identical bit-timing configuration in all CAN nodes. Due to potentially different CAN clock frequency this is not always possible. CAN clock frequencies for optimal interoperability are recommended in 6.1. The bit-timing configuration in systems with different CAN clock frequencies is provided in 7.8.
  - SPs in the arbitration phase may be different from SPs in the data phase without negatively influencing robustness, see 7.6.
  - Different SPs in the CAN nodes reduce robustness. There are following reasons.
    - Different SPs lead to different lengths of the BRS bits and CRC delimiter bits in the different nodes.
    - Different SPs lead to a phase error introduced by the bit rate switching.
  - The SSP may be different in the CAN nodes, without influencing robustness.

- Recom6: Choose sjw_D as large as possible.
  - The maximum possible value is $sjw_D = \min(ps1_D, ps2_D)$.
  - This configuration maximizes the CAN clock tolerance *df* (see 6.2).
- Recom7: Enable TDC for data bit rates ≥ 1 Mbit/s.
  - How to configure TDC is explained in 7.5, i.e. the SSP position.
  - Follow the BRP_D requirements defined in /ISO11898-1/ (see 7.5.1). The value of BRP_D is defined as one or two, when the TDC feature is turned on.

7.5 Transmitter delay compensation configuration

7.5.1 Overview

Transmitter delay compensation (TDC) is a feature for CAN FD with bit rate switching. TDC is recommended for data bit rates equal to or higher than 1 Mbit/s (see /CiA601-1/).

The TDC mechanism defines an SSP. /CiA601-1/ introduces TDC functionalities in detail. Figure 3 shows the definition of the SSP position. /ISO11898-1/ defines the position of the SSP as the distance from the start of the transmitted bit time.

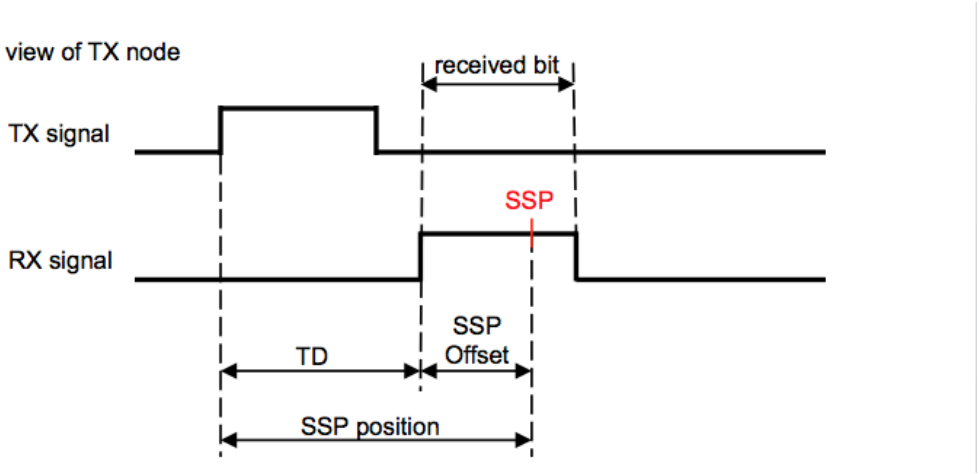

Figure 3 – Definition of the SSP position

/ISO11898-1/ allows the SSP position to be set in two ways.

- Set the SSP to a position derived from a measurement of the actual transmitter delay (TD). This means the CAN controller measures the TD.
- Set the SSP to a fixed position.
  This means the user configures a fixed TD.

/ISO11898-2/ defines a maximum transceiver loop delay of 255 ns, but does not define the minimum loop delay. This means the loop delay of a transceiver can be in a range of 0 ns to 255 ns, even if the practical loop delay range is smaller. Additionally, the transceiver loop delay is temperature-dependent and changes its value during operation. The transceiver loop delay is a part of TD and causes large TD variations. The PCB and host controller have a smaller contribution to the TD variation, as their delays are less temperature dependent.

Due to the large TD variation, the SSP configuration with TD measurement is recommended. This leads to a TD measurement in every transmitted frame, i.e. it adapts to TD variations.

When the TDC feature is turned on, the value of BRP_D is defined as one or two as in /ISO11898-1/. /ISO11898-1/ introduces this restriction to simplify the implementation of the CAN controllers.

7.5.2 SSP configuration – CAN controller measures TD

The SSP offset defines the SSP position relative to the beginning of the received bit. Use the SSP offset to position the SSP inside the received bit.

The SSP should be placed into the middle of the stable area of the received bit. This requires determining the asymmetries in the loopback case (node is transmitting and sampling own bits). The signal integrity at the transmitting node can be even worse than in any other distant node. As last step the phase margins PM1TX and PM2TX have to be used to check the SSP positioning (see 7.7.5.2).

A simpler way to position the SSP is to set

$$SSP = SP - \frac{1\ mtq}{mtq\ per\ bit\ time} = SP - \frac{1\ mtq}{BRP{\cdot}tq\ per\ bit\ time} \quad (14)$$

This means the SSP is 1 mtq earlier inside the received bit compared to the data phase SP inside the transmitted bit. For example, if the data phase SP is at 70 %, and a bit consists of 20 mtq, then configure the SSP to be at 70 % - 1 mtq / 20 mtq = 70 % – 5 % = 65 % in the received bit. To achieve this SSP position the SSP Offset is configured to 13 mtq.

Clause 7.7.5.2 explains why it is safe to use the simplified SSP positioning.

7.5.3 SSP configuration – user configures a fixed TD

Configuring the SSP with a fixed TD is not recommended, because of the large existing TD variations. See 7.5.1 for details.

7.6 Positioning of the arbitration phase SP

The constraints for the positioning of the arbitration phase SP are identical in Classical CAN and CAN FD.

The optimal positions of the arbitration phase SP and the data phase SP are different. The main reason therefore is that the nominal bit-timing has to respect additionally the longest round trip delay between two nodes. This is necessary for a reliable operation of the CAN arbitration mechanism.

The following list contains recommendations for the positioning of the arbitration phase SP:

- The SP should be late enough to consider a sufficiently long propagation segment. The propagation segment accounts for the longest round trip delay between two nodes and also for a potentially present ringing.
- The typically used SP position at 500 kbit/s is 80 %. This can support a line topology with 30 m between the two most distant nodes, if no ringing is present.
- The higher the bit rate the later the SP. The reason therefore is that propagation segment takes a larger portion of the bit time, i.e. it shifts the SP position towards the end of the bit.
- If the SP jitters (e.g. caused by clock jitter or EMC), it is not advisable to configure a SP position larger than 80 % of the bit time.

The bibliography documents [2] and [3] discuss the positioning of the arbitration phase SP.

7.7 Positioning of the data phase SP

7.7.1 Definitions

This clause introduces variables and terms that are used to find the optimal data phase SP.

- Bit asymmetry 1 (A1)

  This is the worst lengthening of a dominant bit in a given setup. This is equal to the worst-

case shortening of a recessive bit. A1 is the sum of the absolute values of the asymmetries of the individual sources:

$$A1 = A1_{transceiver} + A1_{topology} + A1_{galvanic\_isolation} + \cdots \quad (15)$$

The A1 is normally given in nanosecond (ns). Absolute values are used to avoid that one source compensates another. This is a worst-case approach, because a specific compensation cannot work under all circumstances.

- Bit asymmetry 2 (A2)

  This is the worst shortening of a dominant bit in a given setup. This is equal to the worst-case lengthening of a recessive bit. A2 is the sum of the absolute values of the asymmetries of the different sources:

$$A2 = A2_{transceiver} + A2_{topology} + A2_{galvanic\_isolation} + \cdots \quad (16)$$

  The A2 is normally given in nanosecond (ns). Absolute values are used for the same reason as in A1.

- $A1_{transceiver}$ and $A2_{transceiver}$

  $A1_{transceiver}$ is the worst-case lengthening of a dominant bit introduced by the transceivers of the transmitting node and the receiving node together, while $A2_{transceiver}$ is the worst-case shortening respectively. $A1_{transceiver}$ and $A2_{transceiver}$ can be calculated from the asymmetries specified in /ISO11898-2/. /CiA601-1/ shows how to do this. They are worst-case parameters and include the jitter introduced by the transceiver.

- $A1_{topology}$ and $A2_{topology}$

  The term topology refers to all physical layer components (e.g. cabling, CMC, termination resistors, etc.), which are connected to the CAN_H or CAN_L pins of the transceiver, except the transceiver itself. The $A1_{topology}$ and $A2_{topology}$ asymmetry parameters comprise the asymmetries caused by the topology.

7.7.2 Determine asymmetries for a given setup

A1 and A2 are calculated using the equations (15) respectively (16). Therefore, the asymmetry contribution of all sources between a transmitting CAN controller and a receiving CAN controller, e.g. asymmetries introduced by the host controller pins, by a galvanic isolation, by the transceiver, and etc. should be considered.

The spreadsheet provided with this document helps to calculate A1 and A2.

The most asymmetry values are given in data sheets of the corresponding physical layer components.

It is necessary to know, that $A1_{topology}$ and $A2_{topology}$ is different for every communication relationship. This means in a setup with n CAN nodes there are n^2 values for $A1_{topology}$ and n^2 values for $A2_{topology}$. To represent the worst-case, use the maximum $A1_{topology}$ and maximum $A2_{topology}$ value to calculate A1 and A2 respectively.

A reliable way to determine the maximum $A1_{topology}$ and maximum $A2_{topology}$ is to perform a measurement or simulation. 7.7.3 describes the recommended way to do this.

7.7.3 Asymmetry of topology – determine value

This clause shows one exemplary way how to determine the asymmetry introduced by the used topology ($A1_{topology}$ and $A2_{topology}$) by measurement. A similar approach is possible by simulation. The following steps are required.

1st step: Measure the characteristics of the physical layer at room temperature.

- Set up the system in the lab as shown in Figure 4.
- Use the type of transceivers that is used in the final system.
- Send at least one frame with each node. The frame should contain different bit patterns in the data phase, including the bit pattern for PM1 and PM2 (see 7.7.4).
- Measure and record the RX signal and the CAN_H and CAN_L signals at all n nodes. This leads to n^2 measurements as we have n^2 communication relationships. This includes the signals of the transmitting node itself.

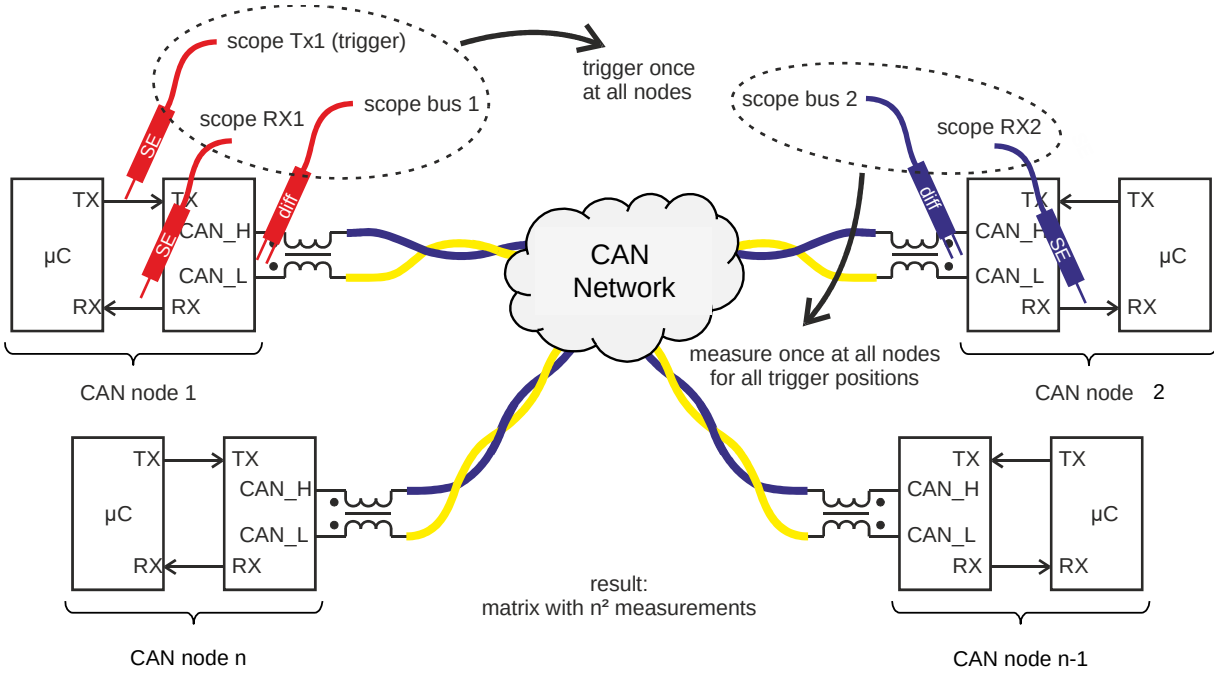

Figure 4 – Lab setup for the evaluation of the asymmetry introduced by the topology

2nd step**:** Determine the maximum bit asymmetries (A1 and A2) from the measurement of 1st step.

- Determine the maximum bit asymmetries (A1 and A2) in two ways.
  - RX based: Use the RX signal to calculate the bit asymmetries.
  - CAN_H or CAN_L based: Use the CAN_H and CAN_L signals and derive the RX signal from it by using worst-case thresholds from /ISO11898-2/. This mimics a worst-case transceiver. Use this RX signal to calculate the bit asymmetries.
- Choose for each communication relationship the worse asymmetry result from the two evaluations (RX based and CAN_H or CAN_L based).
- Plot separately the diagrams for A1 and A2. Figure 5 shows an example of the diagram for A1. In this example there are seven CAN nodes in the CAN network: TN1, TN2, TN3, TN4, TN5, E1, E2. The maximum A1 of 286 ns occurs when node E2 sends a frame and monitors its own bits.

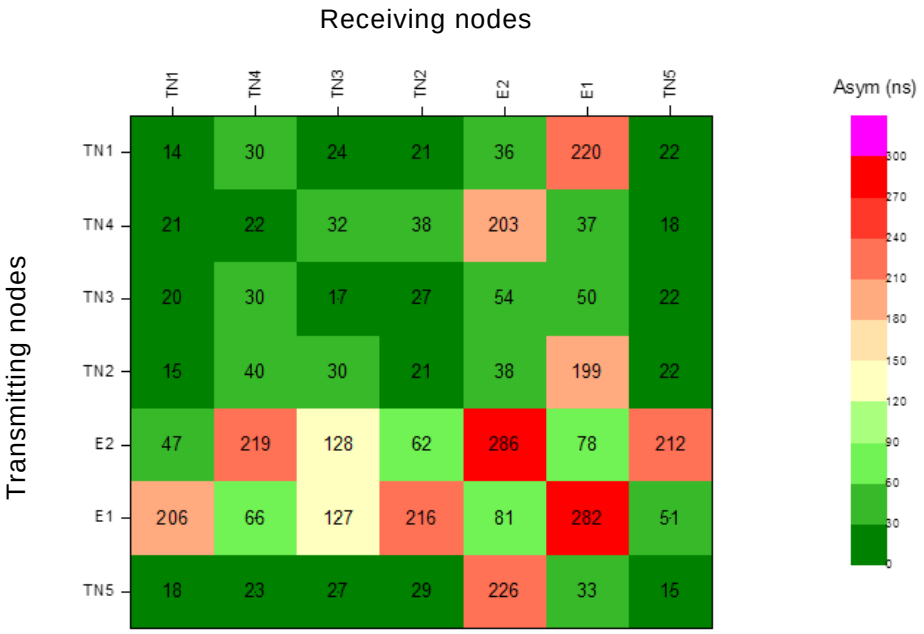

Figure 5 – Example of the diagram for A1

3rd step: Measure the characteristics of the physical layer without topology impacts.

- Build a setup with just two CAN nodes.
- Use the shortest possible CAN line and terminate the network adequately. This means the transceivers are now back-to-back.
- Perform the same measurement as in the 1st step. This means measure and record the RX signals of the two transceivers.

4th step: Determine the maximum bit asymmetries from 3rd step ($A1_{without\ topology}$ and $A2_{without\ topology}$).

- Determine the maximum bit asymmetries analogue to the 2nd step.

5th step: Calculate the maximum asymmetries of the topology ($A1_{topology}$ and $A2_{topology}$).

- The maximum asymmetry of the topology corresponds to the difference between the asymmetry of the whole physical layer (2nd step) and the asymmetry of the physical layer without topology (4th step).
- Calculate $A1_{topology}$ and $A2_{topology}$:

$$A1_{topology} = A1 - A1_{without\_topology} \quad (17)$$

$$A2_{topology} = A2 - A2_{without\_topology} \quad (18)$$

6th step: Repeat all previous steps at multiple temperatures in the targeted temperature range.

- This step is necessary to find the worst-case asymmetry in the desired temperature range.

7.7.4 Phase margin – tolerance for bit asymmetries

7.7.4.1 General

The PM is the allowed shift of a bit edge towards the SP of the bit, at a given tolerance of the CAN clock frequency (df_{used}). In other words, this is the edge shift caused by physical layer effects that is tolerated by the CAN protocol. For more details about the PMs see the bibliography [1].

During frame transmission the receiving nodes and transmitting nodes are sampling the bits from the CAN bus. Further, the edge between two bits can be too early or too late. This leads to four different PMs:

- PM1: Phase margin 1 of receiving node
- PM2: Phase margin 2 of receiving node
- PM1TX: Phase margin 1 of transmitting node
- PM2TX: Phase margin 2 of transmitting node

PM1 and PM2 help to evaluate whether the chosen data phase SP in the receiving nodes enables a robust communication. The PM1TX and PM2TX help to evaluate whether the chosen SSP in the transmitting node enables a robust communication.

In all four cases, the worst-case bit sequence, i.e. that leads to the lowest PMs, is when the transmitting node sends five dominant bits followed by one recessive stuff bit (see /CiA601-1/). This is the longest possible sequence of dominant bits followed by a recessive bit inside a frame. Current transceiver designs cause the largest bit asymmetry at this bit sequence, i.e. the recessive bit is typically shorter than its nominal value. Further effects additionally raise the asymmetry: e.g. asymmetric rise and fall times, network topology, EMC jitter, etc.

7.7.4.2 Phase margin 1 of receiving node

Figure 6 illustrates the worst-case scenario for PM1.

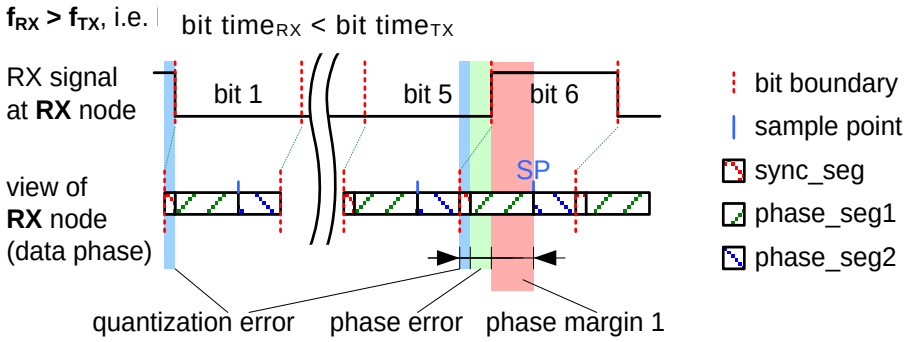

Figure 6 – Worst-case bit sequence for PM1

The following equation calculates the PM1 given in absolute time.

$$PM1 = \frac{6 \cdot BT_D - PS2_D - tq_D}{(1+df_{used})} - \frac{5 \cdot BT_D}{(1-df_{used})} \quad (19)$$

The parameters of this equation can be also expressed in tq instead of absolute time.

$$PM1 = \left(\frac{6 \cdot bt_D - ps2_D - 1}{(1+df_{used})} - \frac{5 \cdot bt_D}{(1-df_{used})}\right) \cdot BRP_D \cdot T_{CAN\_clock} \quad (20)$$

7.7.4.3 Phase margin 2 of receiving node

Figure 7 illustrates the worst-case scenario for PM2.

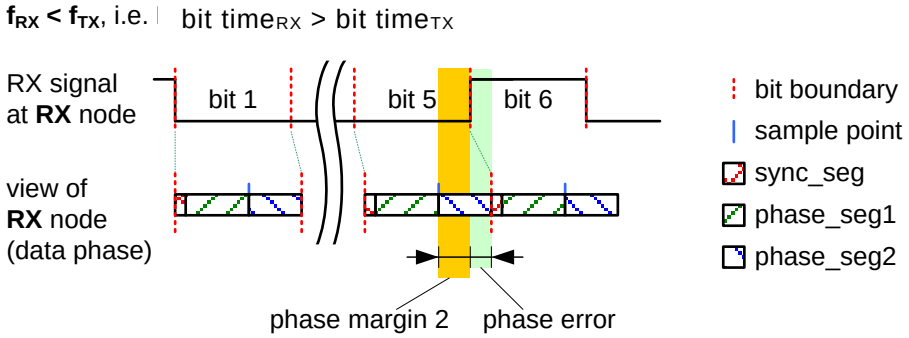

Figure 7 – Worst-case bit sequence for PM2

The following equation calculates the PM2 in absolute time.

$$PM2 = \frac{5 \cdot BT_D}{(1+df_{used})} - \frac{5 \cdot BT_D - PS2_D}{(1-df_{used})} \quad (21)$$

The parameters of this equation can be also expressed in tq instead of absolute time.

$$PM2 = \left(\frac{5 \cdot bt_D}{(1+df_{used})} - \frac{5 \cdot bt_D - ps2_D}{(1-df_{used})}\right) \cdot BRP_D \cdot T_{CAN\_clock} \quad (22)$$

7.7.4.4 Phase margin 1 of transmitting node

For the calculation of PM1TX it is assumed, that the TDC feature is enabled.

The following background knowledge is necessary to understand the worst-case scenario.

- The transmitting node can make a quantization error of 1 mtq, when measuring TD. mtq is the minimum tq length, which is equal to one CAN clock period long. This means the measured TD is in the range: x < TD < x + 1mtq, while x is the real TD.
- The actual sampling of the data bit at the SSP causes no additional quantization error, because the transmitting node just evaluates the RX signal at a specific point in time.
- The CAN clock tolerance can be neglected, as the same node is transmitting and sampling.

The worst-case for PM1TX occurs when TD contains no quantization error. Figure 8 illustrates the worst-case scenario for PM1TX.

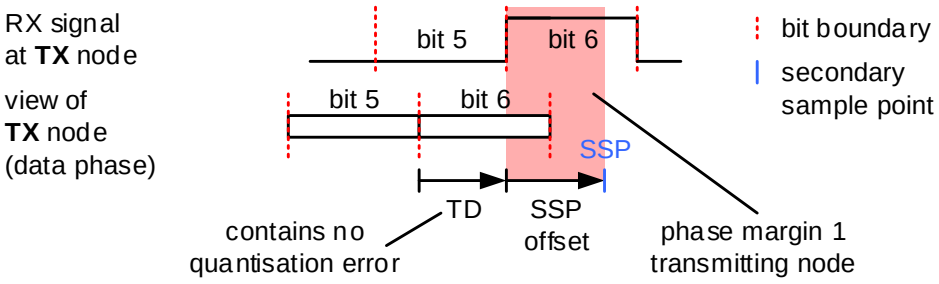

Figure 8 – Worst-case bit sequence for PM1TX

The following equation calculates PM1TX. SSPoffset is an integer number of mtq.

$$PM1TX = SSP\,offset \quad (23)$$

7.7.4.5 Phase margin 2 of transmitting node

For the calculation of PM2TX it is assumed, that the TDC feature is enabled.

The background knowledge that is necessary to understand the worst-case scenario is the same as for PM1TX.

The worst-case for PM2TX occurs when TD contains the maximum quantization error of 1 mtq. Figure 9 illustrates the worst-case scenario for PM1TX.

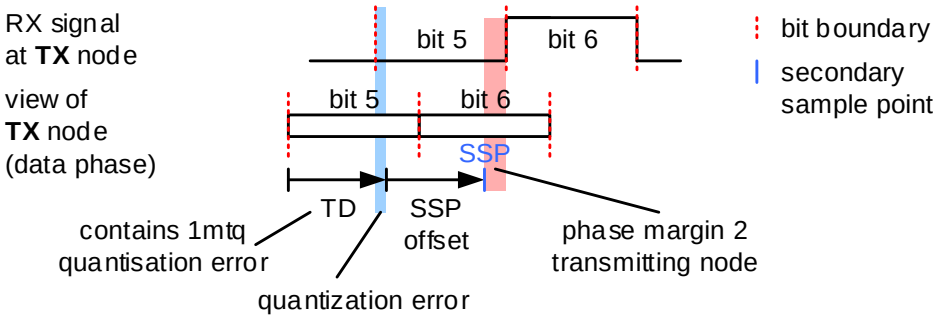

Figure 9 – Worst-case bit sequence for PM2TX

The following equation calculates PM2TX. SSP offset is an integer number of mtq.

$$PM2TX = BT_D - SSP\ offset - 1\ mtq \tag{24}$$

7.7.4.6 Example

This clause shows the typical values of the PMs for given bit rates. Based on the exemplary bit-timing configurations from Table 3 the PMs are calculated with equations given in 7.7.4.2 and 7.7.4.3. The example focuses only on the PM of the receiving node, because this is more critical than the PM of the transmitting node.

Table 3 – Exemplary bit-timing configurations for a CAN clock of 40 MHz

Data bit rate [Mbit/s]	SP [%]	BRP_D	tq_D per bit time	Propagation segment [tq_D]	$ps1_D$ [tq_D]	$ps2_D$ [tq_D]	sjw_D [tq_D]
1,0	75,0	1	40	19	10	10	10
2,0	70,0	1	20	7	6	6	6
4,0	70,0	1	10	3	3	3	3
5,0	75,0	1	8	3	2	2	2

Figure 10 and Figure 11 show the PM1 and PM2 values resulting from Table 3. The Figures show a set of curves, while each curve represents a specific CAN clock tolerance (df_{used}). Additionally it shows the configured data phase SP.

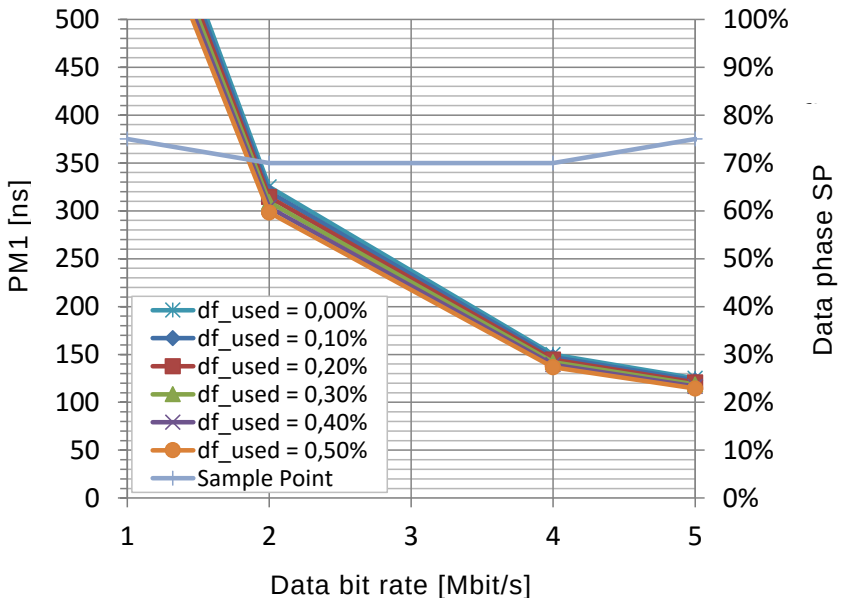

Figure 10 – PM1 for the exemplary bit-timing configurations in Table 3

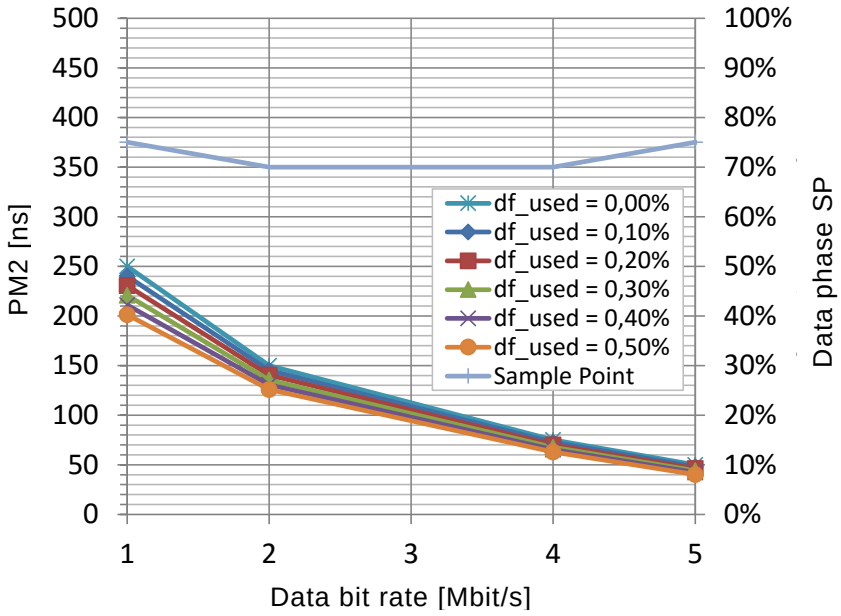

Figure 11 – PM2 for the exemplary bit-timing configurations in Table 3

Figure 10 and Figure 11 visualize most significant properties of PM1 and PM2.

- Property 1: PM1 and PM2 decrease quickly towards higher bit rates.
- Property 2: PM1 and PM2 depend on the SP position. PM1 increases and PM2 decreases when the SP position is shifted towards the end of the bit. If the SP position is shifted towards the beginning of the bit PM1 decreases and PM2 increases.

Example: In Figure 10, PM1 at 4 Mbit/s and 5 Mbit/s differ only by about 20 ns, because PM1 at 5 Mbit/s is improved by the late SP position.

- Property 3: The tolerance (df_{used}) of the CAN clock has a small impact on PM1 and PM2.

  Example: Figure 10 shows the PM1 for different df_{used} in the range df_{used} = ± 0 % up to df_{used} = ± 0,5 %. In the set of curves, each curve represents PM1 for one specific df_{used}. The PM1 curves lie very close to each other what means that df_{used} has a small impact on PM1. The same holds for Figure 11 with PM2. The reason is that the worst-case bit sequence for PM1 has only 6 bit. During these bits the receiving node accumulates only a small phase error.

7.7.5 Evaluation of the robustness

7.7.5.1 Receiving node

This clause introduces how to evaluate whether a given setup is functional under all conditions with a given bit-timing configuration. It focuses on the receiving node with the chosen data phase SP and therefore uses for evaluation PM1 and PM2.

The evaluation consists of the following steps.

- Determine A1 and A2 for the setup.
- Calculate PM1 and PM2 from the bit-timing configuration and the actual tolerance of the CAN clock (df_{used}).
- Calculate SM1 and SM2. The safety margin (SM) is the difference between the allowed asymmetry given by PM1 and PM2, and the actually present asymmetry given by A1 and A2.

$$SM1 = PM1 - A1 \tag{25}$$

$$SM2 = PM2 - A2 \tag{26}$$

- Check if the system is functional.
  - For a functional system it should be fulfilled that SM1 > 0 and that SM2 > 0.
  - The SMs should be balanced: SM1 ≈ SM2.

The spreadsheet provided with this document helps to perform the evaluation. The following hints help to optimize the result:

- Check if SMs values are large enough.

  A system is functional, if SM1 > 0 and SM2 > 0. Nevertheless, it is recommended to design a system in a way that SM1 and SM2 are not too close to zero. This has the following advantages.
  - A SM of e.g. SM1 = SM2 > 20 ns allows for future extensions of the CAN Network.
  - A SM larger than zero provides the tolerance for the unknown or unexpected effects that cause bit asymmetry.
- Balance SMs values.

  It is recommended to balance the SMs, i.e. it should be SM1 ≈ SM2. EMC jitter, which can occur in the system, uses up SM1 and SM2 equally. They can be balanced in the following way.
  - Shift the SP position by changing the data bit-timing to balance SM1 and SM2.

- Try to use shorter tq in the data phase. This allows positioning the SP in finer steps.

- Increase SMs values.
  - Check if balancing SM1 and SM2 can make the setup functional. As the SP position can only be changed in discrete steps, this cannot always work.
  - Reduce A1 and A2 to make the setup functional (see 7.7.5.2).
  - Increase PM1 and PM2 (see 7.7.7).

7.7.5.2 Transmitting node

This clause introduces how to evaluate whether a given setup is functional under all conditions with a given bit-timing configuration. It focuses on the transmitting node with the chosen SSP and therefore uses for evaluation PM1TX and PM2TX.

The evaluation is analog to that of the receiving node and therefore not described further.

This evaluation can be skipped when the SSP is configured to equation (14), which is referred to as simplified SSP positioning in 7.5.2.

The basic idea behind using the simplified SSP positioning is to use only PM1 and PM2 to check the robustness of a CAN bit-timing configuration. This is possible, because PM1 and PM2 are stricter than PM1TX and PM2TX for the chosen SSP position.

A detailed proof for the simplified SSP positioning is given in the following.

It is assumed in first step that the SSP is at the same relative position in the bit as the SP, what means SSP = SP.

The comparison of the worst-case scenarios of PM1 and PM1TX in Figure 6 and Figure 8 respectively indicates: PM1TX = PM1 + phase error + 1 tq. The conclusion is that when SSP = SP, then PM1TX > PM1, while the difference is more than 1 mtq.

The comparison of the worst-case scenarios of PM2 and PM2TX in Figure 7 and Figure 9 respectively indicates: PM2TX + 1 mtq = PM2 + phase error. This means:

- PM2 is reduced by the clock tolerance (phase error), but not by the quantization error.
- PM2TX is not reduced by the clock tolerance (phase error), but by the quantization error.

At a df_{used} = 0,3 % and 2 Mbit/s in the worst-case scenario of PM2 a phase error of 5 × 500 ns × (0,3 % × 2) = 15,0 ns can occur. At the same time with a CAN clock with 40 MHz the quantization error can be up to 1 mtq = 25 ns. The conclusion is that if “SSP = SP”, then PM2TX < PM2, but the difference is less than 1 mtq.

When the SSP is configured according the simplified SSP positioning rule in equation (14), then PM1TX decreases by 1 mtq, and PM2TX increases by 1 mtq, namely PM1TX > PM1 and PM2TX > PM2. From this follows, that if the simplified SSP positioning according equation (14) is used, it is sufficient to evaluate the robustness based on the data phase SP.

7.7.6 Optimization hints for the CAN physical layer design

7.7.6.1 General

This clause gives hints how to optimize the bit asymmetry introduced by the CAN physical layer components. These components include everything that is between the TX pin of the transmitting CAN controller and the RX pin of the receiving CAN controller.

7.7.6.2 Transceiver

Transceivers add much asymmetry. The following hints help to reduce the asymmetry.

- Transceivers that conform to the 5 Mbit/s recessive bit with specification provide a reduced asymmetric delay compared to 2 Mbit/s transceivers.
- Use a transceiver with additional signal improvement functionalities (see /CiA601-4/) to eliminate potential ringing.

7.7.6.3 Topologies

Topologies typically add much asymmetry. The following hints help to reduce the asymmetry.

- Use PVC-free cable insulation. For details see bibliography document [4].
- Reduce the total network length.
- Avoid long not terminated stubs, which are branches from the well-terminated CAN lines; use stubs in the range of centimeters instead of meters.
- Reduce the number of stubs per star point. The more stubs are connected to one star point, the higher the reflection factor gets.
- In case a star point with many branches is required, avoid identical stub lengths per star point.
    - Identical stub lengths cause a resonance effect regarding signal ringing.
    - Differences in a couple of tens of centimeter-range or ideally meter-range are recommended.
- In case multiple-star points are required, keep a significant distance between the two star points.
    - Use meter-range instead of centimeter-range between star points.
    - Short distance between two star points behaves similar like one big star point with all stubs combined.
    - See rule “reduce the number of stubs per star point”.
- Consider a high-impedance termination of not terminated stubs.
    - /ISO11898-2/ allows for a minimum R_L of 50 Ω (extended range down to 45 Ω).
    - The parallel circuit of all resistors including the input resistance of the used transceiver should be taken into account.
    - High-impedance terminations can stay in the lower single-digit kΩ-range and depend on the maximum number of connected nodes.
- Optimize the low-impedance termination (resistor position and resistor value). Another option is to increase the low-impedance termination resistance (e.g. 124 Ω instead of 120 Ω of /ISO11898-2/) to compensate for the high-impedance terminations in systems with many nodes.
- Limit the number of CAN nodes.
- Use a linear topology, terminated at both ends.
    - A long linear topology with many connected nodes between both terminated ends can causes as well signal integrity problems.

  - See rule “Limit the number of CAN nodes”.

- Wire cross-section: Increase it to approximately 0,35 mm^2 for the CAN_H wire and the CAN_L wire. Do not use different wire cross-sections for CAN_H and CAN_L.

7.7.6.4 Device design recommendations

The following list provides device design recommendations:

- Keep the CAN bus pin capacitance of the device towards GND as low as possible. Recommendation is staying below 50 pF per CAN wire to GND. Contributing elements towards this capacitance are:
  - Protection elements like ESD diodes, varistors or dedicated populated capacitors;
  - Parasitic capacitance of connectors;
  - Wiring tracks on the printed circuit boards;
  - Parasitic capacitance of the CMC;
  - Parasitic capacitance of the transceiver bus pins.
- CAN_H and CAN_L PCB tracks from connector to transceiver should be of equal distance and parallel.
- Keep the TXD and RXD PCB tracks between host controller and transceiver short.
- Configure the host controller TXD output pin with push-pull behavior: a pull-up or pull-down resistor behavior can cause additional asymmetries and propagation delays.
- Avoid any serial components like logical gates or resistors within the TXD and RXD connection lines between host controller and transceiver. In case galvanic isolation is required, take care of the potential additional asymmetry and select components accordingly.
- Use a CAN clock source with lower clock jitter.
- Avoid galvanic isolation, or use a galvanic isolation solution that adds only a small asymmetry.

7.7.7 Optimization hints for PMs

This clause gives hints how to optimize PMs. As consequence this increases the SMs of the CAN network.

- Optimize the bit-timing configuration (see 7.4) by reducing the tq length. This increases PM1 by reducing the quantization error.
- Use a CAN clock with lower tolerance (df_{used}). The maximum CAN clock tolerance calculated in 6.2 can remain as is. This improves PM1 and PM2.

7.8 CAN nodes with different CAN clock frequencies

This clause describes the possibilities to configure the bit-timing for systems with CAN nodes with different CAN clock frequencies. It is assumed that the CAN nodes support at least one CAN clock frequency recommended in 6.1.

- Possibility 1: Use the same tq length in all CAN nodes.
  - Therefore, increase the *BRP* in the CAN nodes with the faster CAN clock.

- Advantage: In this case, the identical bit-timing parameters (except the *BRP* value) can be used in all CAN nodes.

- Possibility 2: Choose SP positions that are configurable with all CAN clocks.
  - First, define the bit-timing configurations for the lowest CAN clock frequency. Recommendations for this step are provided in 7.4.
  - Second, define bit-timing configurations for the higher CAN clock frequencies. Make sure, that the bit rates and the SP positions are equal to the bit-timing configuration with lowest CAN clock. The SSP positions can be different in all nodes.
  - Advantage: In this case, the CAN nodes with the higher CAN clocks frequency benefit from the higher frequency. As these nodes use shorter tq they have a lower quantization error, what leads to a larger PM1 (see 7.7.4.2).
  - Table 4 shows an example for this case.

Table 4 – Exemplary bit-timing configuration for CAN nodes with different CAN clock frequencies according possibility 2

Parameter	Node A	Node B	Node C
CAN clock frequency	20 MHz	40 MHz	80 MHz
Bit rate	2 Mbit/s	2 Mbit/s	2 Mbit/s
Arbitration phase SP position	80 %	80 %	80 %
Data phase SP position	70 %	70 %	70 %
SSP position	70 %	70 %	70 %
tq per data bit time	10	20	40
Maximum quantization error per data bit time	50 ns	25 ns	12,5 ns

Both possibilities are recommended. Possibility 1 is simpler, as all nodes use the identical bit-timing configuration (except the *BRP* parameter). Possibility 2 uses other bit-timing configurations for the nodes with a higher CAN clock frequency and therewith increases slightly the robustness of these nodes.

8 Checklist

Table 5 provides references for CAN FD related terms; it does not claim completeness. For a system design each term should be checked and evaluated on its importance for the desired application.

Table 5 – CAN FD related terms

Topic	Term	Reference
CAN clock	CAN clock frequency recommendation	6.1 in /CiA601-3/
	CAN clock tolerance: explanation of the cases	See bibliography [1]
	CAN clock tolerance: calculation	6.2 in /CiA601-3/
Bit-timing configuration	Possible data bit rates	6.3 in /CiA601-3/
	Arbitration phase SP position	7.6 in /CiA601-3/
	Data phase SP position	7.7 in /CiA601-3/
	SSP position (only in data phase)	7.5.2 in /CiA601-3/
	TDC: definition	/CiA601-1/, /ISO11898-1/
	TDC: configuration	7.5 in /CiA601-3/
Robustness evaluation for the data phase	PM: background	See bibliography [1]
	PM: calculation	7.7.4 in /CiA601-3/
Transceiver	Transceiver parameters	/CiA601-1/
	Galvanic isolation	/CiA601-1/
	Bit asymmetry (of recessive and dominant bits)	/CiA601-1/
Topology	Determine asymmetry introduced by topology	See bibliography [4]
	Optimize asymmetry introduced by topology	7.7.6 in /CiA601-3/

Bibliography

The following documents provide useful information to design CAN (FD) networks:

[1] Arthur Mutter: "Robustness of a CAN FD Bus System – About Oscillator Tolerance and Edge Deviations", 14th international CAN Conference (iCC 2013), Paris, France, 2013

[2] Klaus Dietmayer: "Berechnung des Bit-timings bei CAN Bus Systemen", Teil 1 und Teil 2, Elektronik: 21/1997, Elektronik: 22/1997

[3] Florian Hartwich: "The Configuration of the CAN Bit-timing", 6th international CAN Conference (iCC1999), Turin, Italy, 1999

[4] Marc Schreiner: "CAN FD system design", 15th international CAN Conference (iCC 2015), Vienna, Austria, 2015

