

AN754 - Понимание битовой синхронизации модуля CAN фирмы Microchip

Введение	1
Время передачи бита	1
Номинальное время передачи бита	2
Сегмент синхронизации	2
Сегмент распространения	2
Сегмент фазы 1 и сегмент фазы 2.....	2
Точка выборки.....	2
Время обработки информации (IPT).....	3
Ширина перехода синхронизации (SJW).....	3
Квант времени	3
Регистры управления битовой синхронизацией.....	3
Программирование сегментов синхронизации	4
Синхронизация времени передачи бита	5
Допуск генератора.....	5
Задержка распространения.....	5
Синхронизация	6
Аппаратная синхронизация	6
Синхронизация с восстановлением тактовых интервалов.....	6
Правила синхронизации.....	7
Подведение итогов	8
Вычисление допуска генератора для SJW	8
Пример 1. Расчет минимального значения SJW	8
Настройка бита	9
Пример 2. Максимальный допуск генератора	9
Пример 3. Максимальная скорость передачи	10
Заключение	11
Термины и определения	11

Введение

Протокол CAN (*Controller Area Network – контроллерная локальная сеть*) представляет собой асинхронную последовательную шину с битовым кодированием NRZ (*Non-Return to Zero – без возвращения к нулю*), разработанную для быстрой и надежной связи в жестких условиях, таких как автомобильные и промышленные приложения. Протокол CAN позволяет пользователю программировать скорость передачи, точку выборки бита и сколько раз осуществлять выборку бита. Благодаря этим свойствам сеть можно оптимизировать под конкретное приложение.

Существует определённая зависимость между параметрами битовой синхронизации, задержками распространения в физической среде и допусками генераторов для всей системы. В данном документе осуществляется исследование этой зависимости, поскольку она относится к модулю CAN фирмы Microchip и помогает в оптимизации битовой синхронизации для заданных физических атрибутов системы.

Время передачи бита

Время передачи бита CAN (*CAN bit time*) составлено из нескольких не перекрывающихся сегментов. Каждый из этих сегментов состоит из целочисленных отрезков времени, называемых квантами времени (*Time Quanta - TQ*). О них будет рассказано ниже.

Номинальная скорость передачи (*Nominal Bit Rate - NBR*) (f_{bit}) определена в спецификации CAN как число бит в секунду, передаваемых идеальным передатчиком без восстановления тактовых интервалов, и может быть описана уравнением:

$$NBR = f_{bit} = \frac{1}{t_{bit}} \text{ бит/с, где}$$

t_{bit} - номинальное время передачи бита (см. ниже)

Номинальное время передачи бита

Номинальное время передачи бита (*Nominal Bit Time* - **NBT**) (t_{bit}) составлено из нескольких не перекрывающихся сегментов (**Рис. 1**), поэтому NBT представляет собой сумму следующих сегментов:

$$NBT = t_{bit} = t_{SyncSeg} + t_{PropSeg} + t_{PS1} + t_{PS2}, \text{ где}$$

$t_{SyncSeg}$ - сегмент синхронизации

$t_{PropSeg}$ - сегмент распространения

t_{PS1} - сегмент фазы 1

t_{PS2} - сегмент фазы 2

Связанные с NBT точка выборки (*Sample Point*), ширина перехода синхронизации (*Synchronization Jump Width* - **SJW**) и время обработки информации (*Information Processing Time* - **IPT**) будут пояснены ниже.

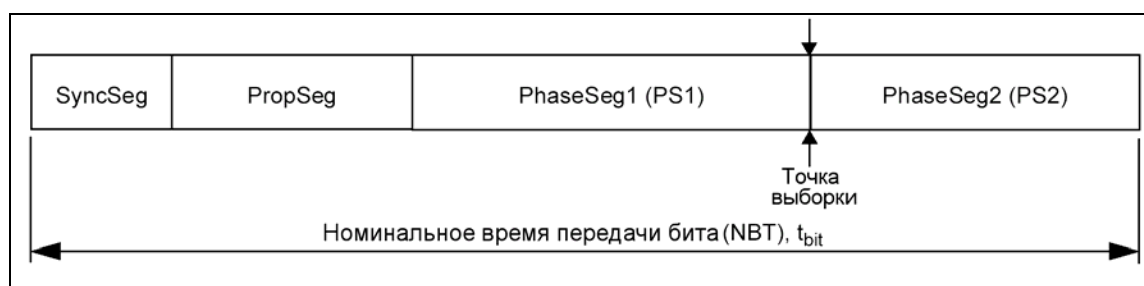


Рис. 1. Сегменты времени распространения бита

Сегмент синхронизации

Сегмент синхронизации (*SyncSeg*) - первый сегмент в NBT, используется для синхронизации узлов на шине. Появление фронтов бита ожидается в пределах SyncSeg. Этот сегмент имеет фиксированную длину в 1TQ.

Сегмент распространения

Сегмент распространения (*PropSeg*) предназначен для компенсации физических задержек между узлами. Задержка распространения определяется как время распространения сигнала на линии шины от передатчика к приемнику и обратно, включая задержки, связанные с драйвером шины. Длина PropSeg программируется от 1TQ до 8TQ.

Сегмент фазы 1 и сегмент фазы 2

Два сегмента фазы, PS1 и PS2 используются, чтобы компенсировать искажения фазы фронта на шине. При синхронизации с восстановлением тактовых интервалов может быть удлинен PS1, либо может быть укорочен PS2. PS1 программируется от 1TQ до 8TQ, а PS2 - от 2TQ до 8TQ.

Точка выборки

Точка выборки - точка во времени распространения бита, в которой читается и интерпретируется логический уровень. Точка выборки расположена в конце сегмента фазы 1 (PS1). Исключение из этого правила составляет ситуация, когда режим выборки настроен на осуществление выборки три раза на бит. В этом случае выборка по-прежнему осуществляется в конце PS1, однако, две дополнительные выборки принимаются в интервалах по 1/2TQ перед концом PS1 и значение бита определяется по значению, полученному в большинстве случаев (два или три раза).

Время обработки информации (IPT)

Время обработки информации (IPT) - время, требующееся логике, чтобы определить уровень бита. Для CAN-модуля Microchip IPT начинается в точке выборки, уровень бита измеряется за один TQ и устанавливается в течение второго TQ (то есть IPT занимает 2TQ). Поскольку сегмент фазы 2 также начинается в точке выборки и является последним сегментом во времени передачи бита, то требуется, чтобы минимальная длина PS2 была не меньше, чем IPT.

Поэтому:

$$PS2_{min} = IPT = 2TQ.$$

Ширина перехода синхронизации (SJW)

Ширина перехода синхронизации (SJW) корректирует время бита по мере необходимости на 1TQ - 4TQ (в зависимости от настройки), чтобы поддержать синхронизацию с передаваемым сообщением. Подробнее о синхронизации будет рассказано ниже.

Квант времени

Каждый сегмент, составляющий время передачи бита, состоит из целочисленных отрезков времени, называемых квантами времени (*Time Quanta* - TQ). Длина каждого кванта времени основывается на периоде генератора (T_{OSC}). Основа TQ равна двойному периоду генератора. **Рис. 2** показывает, как период бита получается из T_{OSC} и TQ. Длина TQ равна периоду времени TQ (T_{BRPCLK}), который задается с помощью программируемого делителя, называемого делителем скорости передачи (*Baud Rate Prescaler* - BRP). Это показано в следующем уравнении:

$$TQ = 2 \cdot BRP \cdot T_{OSC} = \frac{2 \cdot BRP}{F_{OSC}},$$

где BRP эквивалентно настройке, показанной на **Рис. 3**.

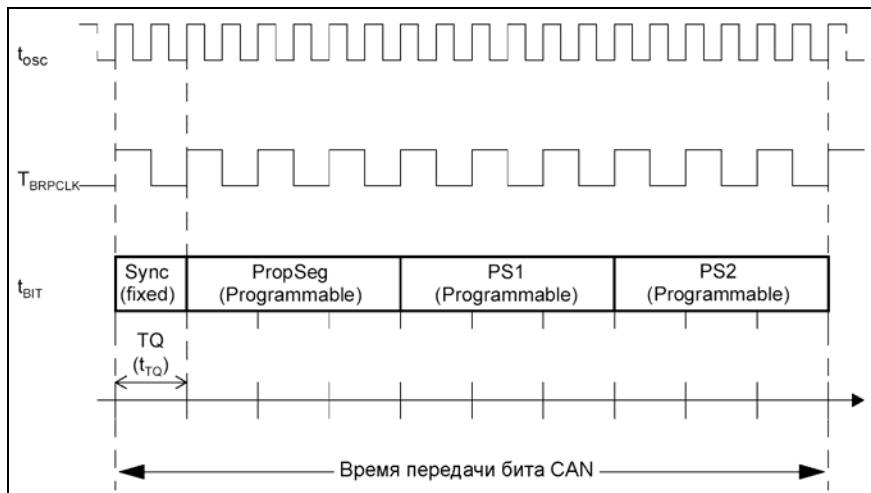


Рис. 2. TQ и период бита

Регистры управления битовой синхронизацией

Регистры управления битовой синхронизацией *CNF* – это три регистра, которые настраивают время передачи бита CAN. **Рис. 3** подробно описывает функции регистров CNF.

С помощью настройки длины TQ и числа TQ в каждом сегменте, и номинальное время передачи бита, и точку выборки можно легко настроить по желанию.

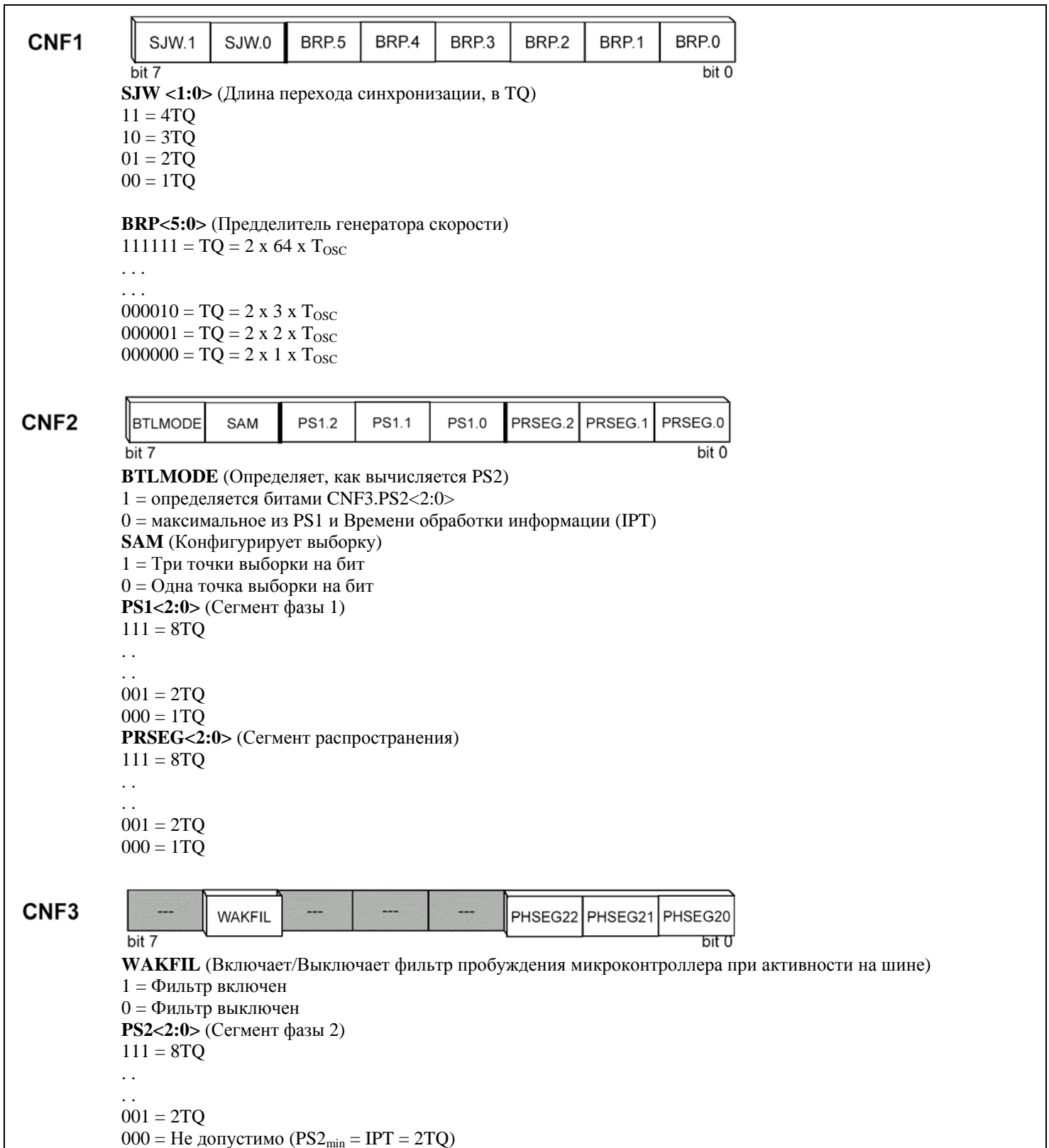


Рис.3. Регистры управления битовой синхронизацией CAN (регистры CNF MCP2510)

Программирование сегментов синхронизации

Существует несколько требований для программирования сегментов битовой синхронизации CAN:

1. PropSeg + PS1 ≥ PS2
2. PropSeg + PS1 ≥ t_{PROP}
3. PS2 > SJW

Синхронизация времени передачи бита

Все узлы на шине CAN должны иметь одинаковую номинальную скорость передачи (NBR). Шум, сдвиг фазы и уход частоты создают ситуации, когда номинальная скорость передачи не равна фактической скорости передачи в реальной системе. Поэтому узлы должны иметь метод достижения и поддержания синхронизации с сообщениями шины.

Допуск генератора

Битовая синхронизация для каждого узла системы CAN получается из опорной частоты (F_{osc}) узла. Это создает ситуацию, когда сдвиг фазы и уход частоты произойдет между узлами из-за не идеальных допусков генератора между узлами.

Спецификация CAN указывает, что в наихудшем случае допуск генератора составляет 1.58% и соответствует только низкой скорости передачи (125 Кбит/с или менее). В данном примере применения допуски генератора подробно не рассматриваются.

Задержка распространения

Протокол CAN имеет заданные рецессивное (логическая 1) и доминантное (логический 0) состояния для осуществления поразрядной схемы неразрушающего арбитража. Именно на эту методологию больше всего воздействуют задержки распространения. Каждый узел, включенный в арбитраж, должен быть способен произвести выборку уровня бита внутри одного и того же времени передачи бита. Например, если два узла на противоположных концах шины начинают передавать их сообщения в одно и то же время, они должны произвести арбитраж для управления шиной. Этот арбитраж эффективен, только если оба узла способны выполнять выборку в течение одного и того же времени передачи бита. На **Рис. 4** показана однонаправленная задержка распространения между двумя узлами. Критические задержки распространения (вне точки выборки) приведут к ошибочному арбитражу. Это подразумевает, что длина шины ограничена заданной скоростью передачи данных CAN.

Если предположить, что все узлы в системе имеют сходные задержки компонентов, то задержка распространения в CAN-системах рассчитывается следующим образом:

$$t_{PROP} = 2 \cdot (t_{BUS} + t_{CMP} + t_{DRV}), \text{ где}$$

t_{BUS} - время прохождения сигнала туда и обратно в физической среде шины.

t_{CMP} - входная задержка компаратора.

t_{DRV} - выходная задержка драйвера.

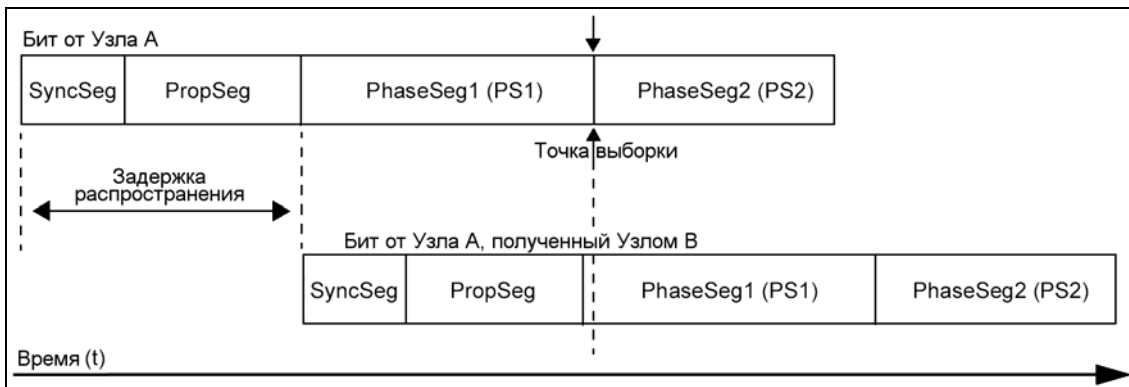


Рис. 4. Однонаправленная задержка распространения

Синхронизация

Все узлы на одной шине CAN должны иметь одинаковое номинальное время передачи бита (NBT). Кодирование NRZ не встраивает время в сообщение. Приёмники должны синхронизироваться по потоку передаваемых данных, чтобы обеспечить правильное декодирование сообщения.

Существует два метода достижения и поддержания синхронизации.

Аппаратная синхронизация

Аппаратная синхронизация происходит только по первому переходу от рецессивного к доминантному (логический переход от 1 к 0) в течение простоя шины, который указывает начало кадра (*Start-of-Frame - SOF*). Аппаратная синхронизация вызывает сброс счетчика битовой синхронизации в SyncSeg, заставляя фронт располагаться в пределах SyncSeg. В этой точке все приёмники будут синхронизированы с передатчиками.

Аппаратная синхронизация происходит только один раз в течение сообщения. Кроме того, синхронизация с восстановлением тактовых интервалов не может происходить в течение того же самого времени передачи бита (SOF), в течение которого происходит аппаратная синхронизация.

Синхронизация с восстановлением тактовых интервалов

Синхронизация с восстановлением тактовых интервалов осуществляется, чтобы поддержать начальную синхронизацию, которая была установлена аппаратной синхронизацией. Без восстановления тактовых интервалов принимающие узлы могут выйти из синхронизации из-за ухода частоты между ними.

Синхронизация с восстановлением тактовых интервалов осуществляется функциями устройства цифровой фазовой автоподстройки (*Digital Phase Lock Loop - DPLL*), которое сравнивает фактическую позицию перехода рецессивного состояния в доминантное на шине с позицией ожидаемого перехода (внутри SyncSeg) и при необходимости корректирует время передачи бита.

Искажение фазы бита задаётся позицией фронта в отношении к SyncSeg, измеренной в TQ, и определяется следующим образом:

$e = 0$ - переход находится внутри SyncSeg.

$e > 0$ - переход находится перед точкой выборки (к PS1 прибавляется TQ).

$e < 0$ - переход находится после точки выборки предыдущего бита (из PS2 вычитается TQ).

На **Рис. 5** показано, как искажения фазы (кроме нулевого) заставляют время передачи бита удлиняться или укорачиваться. Для медленного передатчика искажение фазы положительно, поэтому сегмент фазы 1 удлиняется. Для быстрого передатчика искажение фазы отрицательно, поэтому сегмент фазы 2 укорачивается.

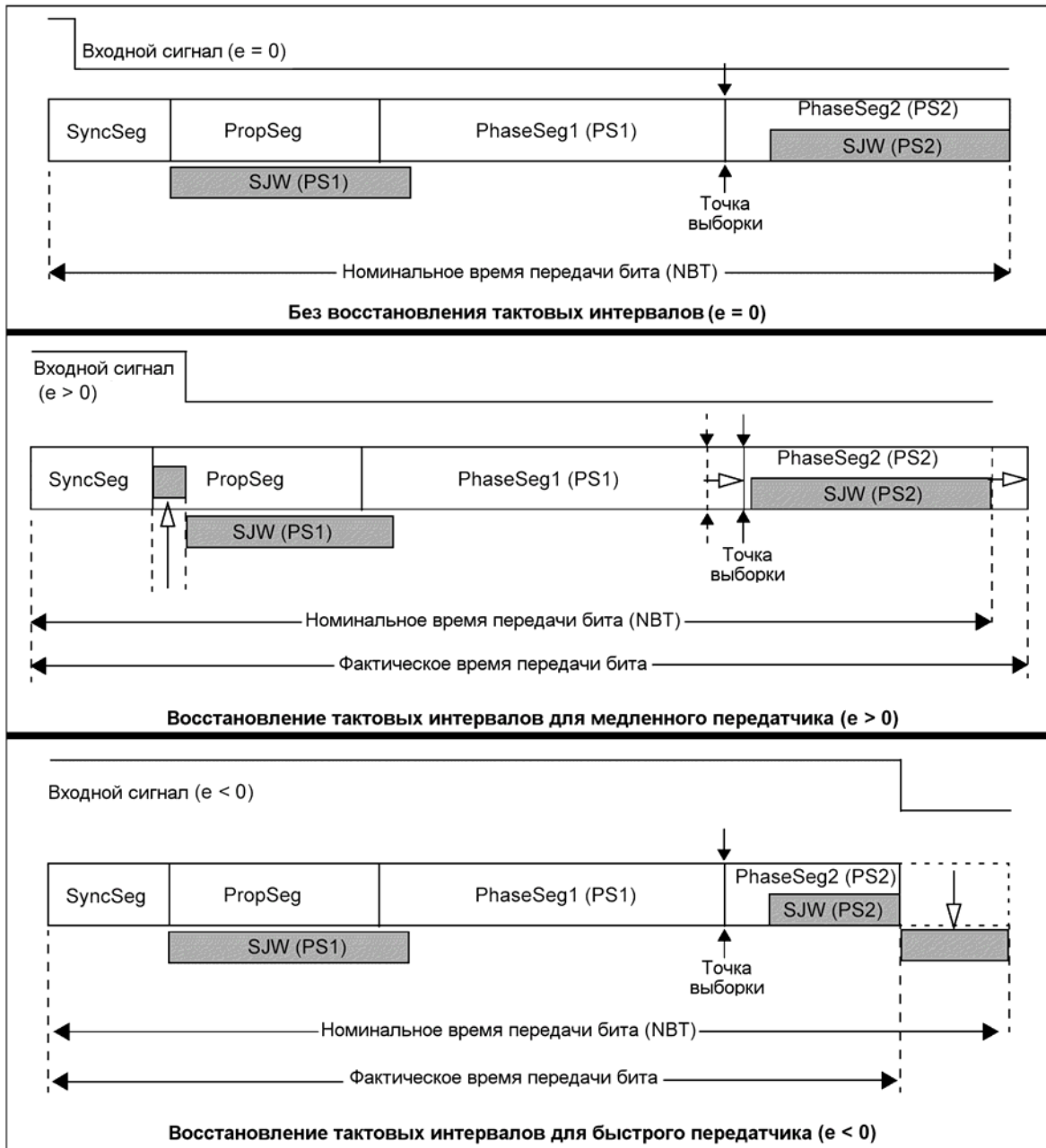


Рис. 5. Синхронизация времени передачи бита

Правила синхронизации

1. Только переход из рецессивного состояния в доминантное будет использоваться для синхронизации.
2. Разрешается только одна синхронизация в пределах одного времени передачи бита.
3. Переход будет использоваться для синхронизации, только если значение предыдущей точки выборки отличается от значения шины сразу же после перехода.
4. Передающий узел не будет выполнять восстановление тактовых интервалов при положительном искажении фазы ($e > 0$). Это подразумевает, что передатчик не будет выполнять восстановление тактовых интервалов из-за задержки распространения им же переданного сообщения. Приёмники будут выполнять синхронизацию как обычно.
5. Если абсолютная величина фазового искажения больше, чем ширина фазового перехода (SJW), то соответствующий сегмент фазы (PS1 или PS2) будет откорректирован до величины, равной SJW.

Подведение итогов

Как указывалось выше, CAN протокол осуществляет схему поразрядного неразрушающего арбитража, которая позволяет узлам осуществлять арбитраж для управления шиной. Поэтому всем узлам необходимо осуществлять выборку битов в пределах одного и того же времени передачи бита. Зависимость между задержкой распространения и допуском генератора влияет на скорость передачи данных и длину шины. В **Таблице 1** показаны некоторые обычно принимаемые длины шины и соответствующие скорости передачи.

Этот пример не рассматривает все подробности настройки времени передачи бита для всех случаев, однако рассматриваются некоторые основные методологии конфигурирования времени передачи бита CAN.

Таблица 1. Скорость передачи данных и длина шины

Скорость передачи данных (Кбит/с)	Длина шины (м)
1000	30
500	100
250	250
125	500
62.5	1000

Вычисление допуска генератора для SJW

Правило битового наполнения (битстаффинга) (*bit stuffing* – *битовое наполнение*) гарантирует, что не более пяти одинаковых битов будут переданы последовательно в течение одного кадра сообщения. Исключение составляет конец сообщения, состоящий из десяти рецессивных битов: один разделитель АСК, семь битов конца кадра (*end-of-frame* - *EOF*) и три бита межкадрового интервала.

Восстановление тактовых интервалов может происходить только на переходе рецессивный-к-доминантному. Это означает, что из-за битстаффинга между моментами восстановления тактовых интервалов может быть максимум десять битов (**Рис. 6**).

Для определения минимального SJW можно использовать допуск генератора между самым медленным и самым быстрым узлами. Пусть узел А - медленный (самое длинное время передачи бита), а узел В - быстрый (самое короткое время передачи бита). Тогда:

$$10 \cdot t_{bit(A)} \leq 10 \cdot t_{bit(B)} + t_{SJW(B)}, \text{ где}$$

$t_{bit(n)}$ - время передачи бита узла n

$t_{SJW(n)}$ - SJW узла n

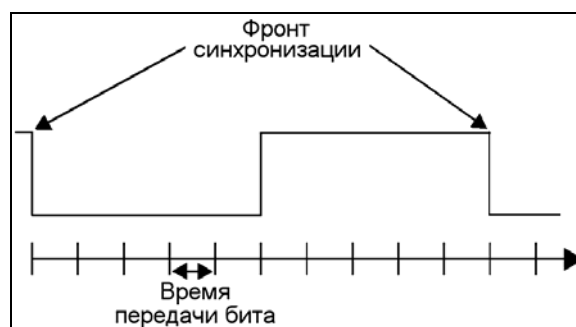


Рис. 6. Максимальное время между фронтами синхронизации

Пример 1. Расчет минимального значения SJW

Дано:

Номинальное время передачи бита (NBT) = 1 мкс

Допуск генератора = 1.25%

Примечание: количество TQ на бит = 8.

Найти: минимальное значение SJW.

Решение:

$$t_{bit(A)} = 1 + 1 \cdot 1.25\% = 1 + 0.0125 = 1.0125 \text{ мкс}$$

$$t_{bit(B)} = 1 - 1 \cdot 1.25\% = 1 - 0.0125 = 0.9875 \text{ мкс}$$

$$TQ_{(A)} = 1.0125/8 = 0.1265625 \text{ мкс} = 126.5625 \text{ нс}$$

$$TQ_{(B)} = 0.9875/8 = 0.1234375 \text{ мкс} = 123.4375 \text{ нс}$$

Используя уравнение, приведенное выше:

$$t_{SJW(B)} \geq 10 \cdot t_{bit(A)} - 10 \cdot t_{bit(B)} = 0.250 \text{ мкс} = 250 \text{ нс}$$

$$\text{Количество } TQ_{SJW} \geq t_{SJW(B)}/TQ_{(B)} = 250/123.4375 = 2.025$$

$$\text{Количество } TQ_{SJW} = 3$$

Кроме того, можно использовать следующее уравнение, чтобы поддержать синхронизацию в течение нормальной работы шины:

$$SJW > (2\Delta f)(10NBT) .$$

Решение для допуска генератора (Δf):

$$\Delta f < SJW / 20NBT .$$

Настройка бита

Вообще, чем длиннее шина, тем меньше максимальная скорость передачи данных из-за задержек в линии. Увеличение допусков генератора между узлами может сильно увеличивать эту зависимость.

Разработчики CAN-системы при создании сети должны учитывать эту зависимость. Следующие примеры демонстрируют настройку битовой синхронизации для достижения максимального допуска генератора или максимальной скорости передачи бита.

Пример 2. Максимальный допуск генератора

Максимальный допуск генератора для максимальной скорости передачи данных достигается, когда сегменты фазы 1 и 2 равны максимальной ширине перехода синхронизации (4TQ). К тому же сегмент распространения минимален, указывая на короткую шину и быстрый приёмопередатчик.

Как указывалось выше, задержка распространения - двойная сумма задержек шины, электрической схемы приёмника (компаратора) и драйвера:

$$t_{PROP} = 2 \cdot (t_{BUS} + t_{CMP} + t_{DRV}) .$$

Дано:

$$\text{Длина шины} = 50 \text{ м}$$

$$\text{Задержка шины} = 5.5 \text{ нс/м}$$

$$t_{BUS} = 50 \cdot 5.5 = 275 \text{ нс}$$

$$t_{CMP} = 40 \text{ нс}$$

$$t_{DRV} = 60 \text{ нс}$$

$$t_{PROP} = 2 \cdot (t_{BUS} + t_{CMP} + t_{DRV}) = 2 \cdot (275 + 40 + 60) = 750 \text{ нс}$$

Поскольку сегмент распространения используется для компенсации задержек распространения и должен быть установлен минимум в 1TQ, предполагаемый квант времени = $t_{PROP} = 750 \text{ нс}$.

Рис. 7 показывает настройку битовой синхронизации для максимального допуска генератора.

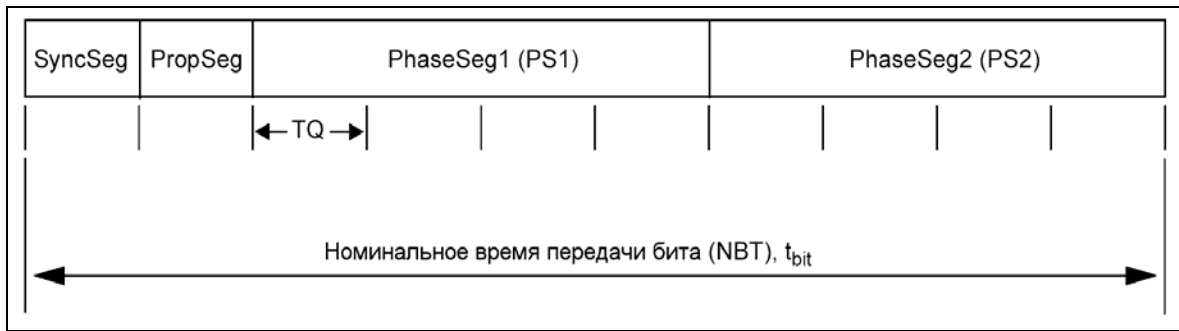


Рис. 7. Битовая синхронизация для максимального допуска генератора

Здесь:

$$TQ = t_{PROP} = 750 \text{ нс}$$

$$t_{bit} = 10TQ = 7.5 \text{ мкс} \Rightarrow NBR = 1 / 7.5 \text{ мкс} = 133.3 \text{ Кбит/с}$$

$$SyncSeg = 1TQ$$

$$PropSeg = t_{PROP} = 1TQ$$

$$PS1 = SJW_{MAX} = 4TQ$$

$$PS2 = SJW_{MAX} = 4TQ$$

$$SJW_{MAX} = 0.4NBT = 4TQ$$

Пример 3. Максимальная скорость передачи

Предыдущий пример показал, что для данной длины шины максимальная скорость передачи данных обратно пропорциональна допуску генератора (когда допуск генератора увеличивается, скорость передачи данных уменьшается). Для достижения максимальной скорости передачи данных для данной длины шины акцент делается на настройку времени передачи бита для задержек распространения (то есть установка PropSeg в максимум). Допуск генератора должен быть минимизирован.

Даны те же самые задержки, что и в предыдущем примере:

$$t_{BUS} = 50 \cdot 5.5 = 275 \text{ нс}$$

$$t_{CMP} = 40 \text{ нс}$$

$$t_{DRV} = 60 \text{ нс}$$

$$t_{PROP} = 2 \cdot (t_{BUS} + t_{CMP} + t_{DRV}) = 2 \cdot (275 + 40 + 60) = 750 \text{ нс}$$

Поскольку допуск генератора минимален, сегменты фазы и SJW могут быть установлены в минимум.

Пусть полное время передачи бита равно $10TQ$, тогда PropSeg может быть установлен в $6TQ$, который установит $TQ = 125 \text{ нс}$. **Рис. 8** показывает настройку битовой синхронизации для максимальной скорости передачи бита.

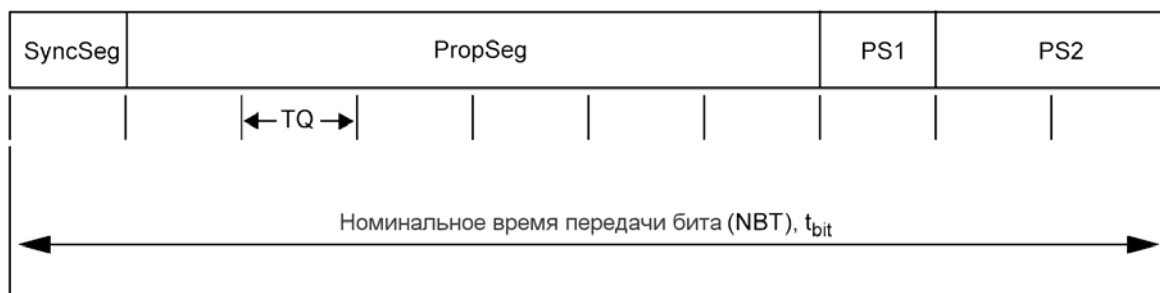


Рис.8. Битовая синхронизация для максимальной скорости передачи бита

Здесь:

$$TQ = t_{PROP} / 6 = 125 \text{ нс}$$

$$t_{bit} = 10TQ = 1.25 \text{ мкс} \Rightarrow NBR = 1 / 1.25 \text{ мкс} = 800 \text{ Кбит/с}$$

$$SyncSeg = 1TQ$$

$$PropSeg = t_{PROP} = 6TQ$$

$$PS1 = 1TQ$$

$$PS2 = 2TQ$$

$$SJW_{MIN} = 1TQ$$

Заключение

Настройка битовой синхронизации CAN - не произвольный процесс. Разработчик системы должен знать о факторах, влияющих на битовую синхронизацию, и компенсировать их, чтобы получить оптимальную производительность в сети. Например, если разрабатываемая система использует генераторы с максимальным допуском, максимальная длина шины сокращается. Аналогично, если требуется максимальная длина шины, допуски генератора необходимо минимизировать. Также нужно учитывать скорость передачи данных CAN, так как она является третьей переменной, которая определяет максимум длины шины и минимум допусков генератора.

Этот пример применения поможет системным инженерам в проектировании контроллерной локальной сети для оптимальной производительности, основываясь на требованиях системы.

Термины и определения

Доминантный бит (*Dominant bit*) - логический 0, аннулирует рецессивный бит во время арбитража.

Рецессивный бит (*Recessive bit*) - логическая 1.

Узел CAN - точка в сети, где подключена CAN-связь.

Номинальное время передачи бита (*Nominal Bit Time - NBT*) - число битов в секунду, передаваемых идеальным передатчиком.

Задержка распространения (*Propagation Delay*) - время прохождения сигнала туда и обратно по физической среде шины.

Аппаратная синхронизация (*Hard Synchronization*) - сбрасывает битовые счётчики принимающих узлов. Происходит только в начале кадра (*Start Of Frame - SOF*).

Синхронизация с восстановлением тактовых интервалов (*Resynchronization*) - поддерживает синхронизацию коррекцией битов в случае необходимости.

Время обработки информации (*Information Processing Time - IPT*) - время, требующееся для определения уровня бита. Начинается в точке выборки.

Начало кадра (*Start Of Frame - SOF*) - первый доминантный бит в течение простоя шины. Указывает на начало кадра.

Точка выборки (*Sample Point*) - позиция в пределах бита, где осуществляется определение логического уровня.