



MICROSNIP

## Автомобильный стандарт LIN

### и микроконтроллеры

### для его реализации

#### Введение

Передача данных — популярная тема для производителей электронной техники. Межконтинентальные корпоративные сети, Интернет, спутниковая телефония... Огромное количество информации, дорогое оборудование, высокая скорость передачи данных... А как быть, если нужно построить небольшую приборную локальную сеть с низкой скоростью передачи данных и предельно низкой ценой, обеспечить при этом системе гибкость и простоту? Наш рассказ пойдет как раз о таких системах. Существенная доля рынка таких устройств — автомобильная электроника. Электроприводы зеркал, электролюк, климат-контроль, АБС, навигационная система, электронное управление впрыском топлива — ... Совсем недавно таких приборов просто не существовало. Теперь в этом списке значится не один десяток устройств. А для того чтобы скоординировать работу различных автомобильных подсистем, сделать автомобиль более комфортным — для этого необходимо обеспечить возможность обмена данными между различными электронными приборами в автомобиле.

Электроника все больше и больше входит в нашу повседневную жизнь. Там,

где раньше было немислимо дорого применить хотя бы транзистор, сейчас применяются микроконтроллеры.

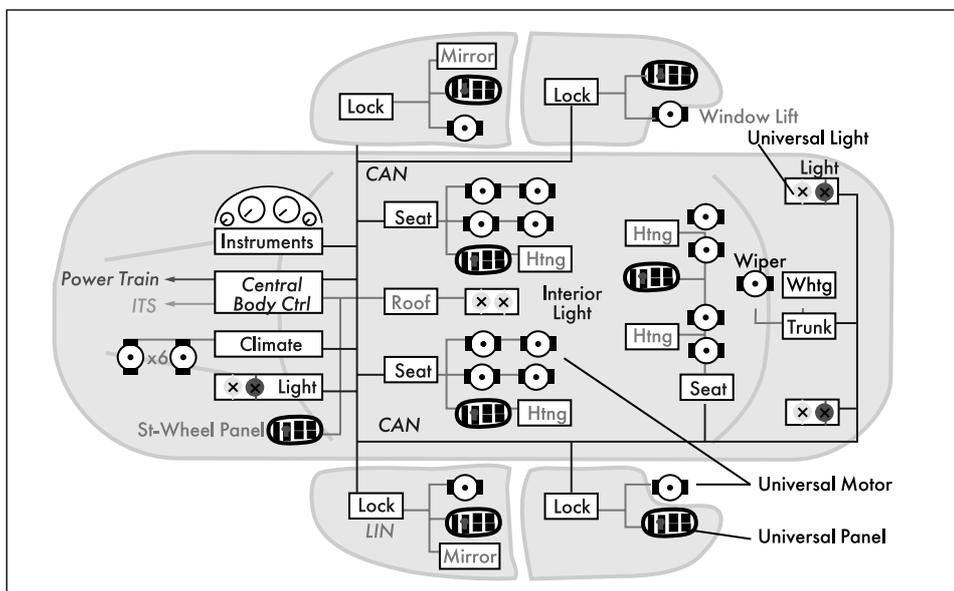
#### Протоколы CAN и LIN: особенности и различия

В современном автомобиле электроника выполняет бесчисленное множество функций. Все их можно условно разделить на две части: первое — это обеспечение надежного функционирования основных узлов автомобиля (например, электронное управление двигателем) и обеспечение безопасности (АБС, подушки безопасности и т. д.). Ко второй половине можно отнести различные электронные системы управления, служащие для обеспечения комфорта пассажиров и даже для их развлечения. В первом случае нужен высоконадежный, достаточно скоростной канал связи, во втором — простой и дешевый. Кроме того, оба эти протокола должны быть стандартными, что упростит производителям различной автомобильной электроники делать унифицированные модули, пригодные для использования в различных автомобилях разных производителей. В качестве первого де-факто выступает скоростной промышленный высоконадежный протокол CAN. Он спроектирован таким образом, чтобы обеспе-

чить надежную передачу данных от одного узла другому при любых обстоятельствах. В качестве второго до недавнего времени никаких стандартов не было, и каждый производитель был вынужден придумывать свои собственные системы. И вот совсем недавно в таком качестве утвержден стандарт LIN.

Технические требования протокола LIN (Local Interconnection Network) разработаны консорциумом европейских автопроизводителей и других известных компаний, включая Audi AG, BMW AG, Daimler Chrysler AG, Motorola Inc., Volcano Communications Technologies AB, Volkswagen AG, и VolvoCar Corporation. Протокол LIN предназначен для создания дешевых локальных сетей для обмена данными на коротких расстояниях. Он служит для передачи входных воздействий, состояний переключателей на панелях управления и т. д., а также ответных действий различных устройств, соединенных в одну систему через LIN, происходящих в так называемом «человеческом» временном диапазоне (порядка сотен миллисекунд). Основные задачи, возлагаемые на LIN консорциумом европейских автомобильных производителей, — объединение различных автомобильных подсистем и узлов (таких как дверные замки, стеклоочистители, стеклоподъемники, управление магнитолой и климат-контролем, электролюк и т. д.) в единую электронную систему.

LIN-протокол утвержден Европейским Автомобильным Консорциумом как дешевое дополнение к сверхнадежному протоколу CAN. На рисунке видно, что LIN и CAN дополняют друг друга и позволяют объединить все электронные автомобильные приборы в единую многофункциональную бортовую сеть. Причем область применения CAN — участки, где требуется сверхнадежность, скорость, область же применения — LIN-объединение дешевых узлов, работающих с малыми скоростями передачи информации на коротких дистанциях, сохраняя при этом универсальность, многофункциональность, а также простоту разработки и отладки. Стан-



дарт LIN включает технические требования на протокол передачи, а также на среду передачи данных. Как последовательный протокол связи LIN эффективно поддерживает управление электронными узлами в автомобильных системах с шиной класса «А» (двухнаправленный полудуплексный), что подразумевает наличие в системе одного главного (Master) и нескольких подчиненных (slave) узлов.

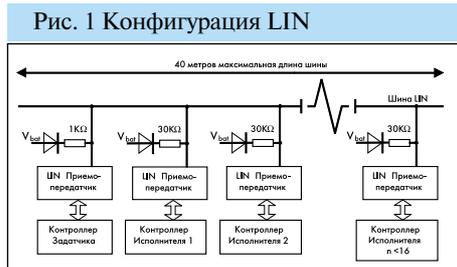
**LIN – характеристики и организация**

Протокол LIN поддерживает двухнаправленную передачу данных по одному проводу длиной до 40 метров, используя недорогой микроконтроллер с генератором на RC-цепочке, без использования кварцевого резонатора. Основная идеология — как можно больше задач переложить на программное обеспечение с целью уменьшения стоимости конструкции. Контроллеры автоматически проводят самосинхронизацию при каждой посылке данных.

В основу LIN положена концепция «single-master/multi-slave», дешевое исполнение, основанное на обычных последовательных интерфейсах UART/SCI, как программная, так и аппаратная возможность реализации, самосинхронизирующаяся система, работающая от RC-генератора и не требующая кварцевого резонатора для Slave- устройств, гарантированное время ожидания для передаваемого сигнала. Дешевое однопроводное исполнение и скорость до 20 кбит/сек. Возможен перевод шины в режим микропотребления «SLEEP», когда шина выключается с целью уменьшения потребляемого тока, но любой узел на шине при необходимости может включить ее вновь. Основное отличие протокола LIN от шины CAN заключается в том, что концепция LIN — это система связи с очень низкой стоимостью за счет снижения эффективности. Технические требования линейного приемопередатчика удовлетворяют стандартам ISO 9141.

Структура шины представляет собой нечто среднее между I<sup>2</sup>C™ и RS-232. Шина подтягивается вверх к источнику питания через резистор в каждом узле и вниз через открытый коллекторный переход приемопередатчика, как в I<sup>2</sup>C™. Но вместо стробирующей линии, каждый передаваемый байт обрамляется стартовым и стоповым битами и передается асинхронно, как в RS-232.

На рис. 1 показана типовая конфигурация шины LIN. Для обмена данными используется один сигнальный провод, в каждом узле подтянутый к источнику питания через резистор. В качестве вы-



ходного каскада используется транзистор с открытым коллектором. Активным состоянием является низкий уровень на шине данных, в это состояние ее может перевести любой узел, пошлав в шину низкий уровень. В пассивном состоянии на шине напряжение питания Vbat (9...18 В), означая, что все узлы на шине в неактивном состоянии. Рабочее напряжение питания находится в пределах 9...18 В, но все узлы должны выдерживать перегрузки и сохранять работоспособность при увеличении напряжения на шине вплоть до 40 В. Обычно микроконтроллер в каждом узле подключен к шине через приемопередатчик, который и обеспечивает защиту от перегрузок. Это позволяет использовать обычный микроконтроллер с напряжением питания 5 В, в то время как сама шина работает на больших напряжениях.

Шина в каждом узле подтягивается к напряжению питания (Vbat). Для устройства-здатчика (master) значение терминального резистора составляет 1 кОм, для устройств-исполнителей (slave) — 20...47 кОм. Максимальная длина шины составляет 40 метров.

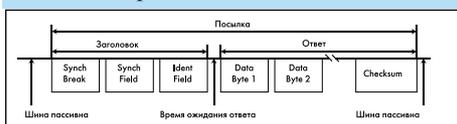
Каждый передаваемый байт обрамлен стартовым и стоповым битами, как показано на рис. 2. Передача начинается с младшего бита. Стартовый бит равен нулю, а стоповый — пассивному состоянию шины (единице).

Рис. 2. Формат передачи данных



Все управление шиной осуществляет задатчик (master). Он посылает в шину запрос с адресом интересующего его исполнителя, а затем осуществляет с ним обмен данными. Исполнители (slave) лишь передают или принимают данные по запросу задатчика. Передача сообщения (рис. 3) начинается задатчиком с посылки сигнала «Synch Break», которое

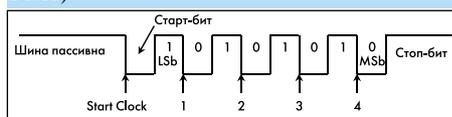
Рис. 3. Формат посылки



представляет собой 13 последовательно идущих нулей и сообщает всем исполнителям, что начался цикл обмена; затем идет поле синхронизации (Synch Field) и поле идентификации (Ident Field). Поле синхронизации передается задатчиком в начале каждого сообщения, и все исполнители должны принять это сообщение и подстроить частоту своего собственного приемопередатчика. Второй байт каждой посылки — поле идентификации (адреса), в котором сообщается, с каким исполнителем начинается обмен данными в этой посылке и сколько байт будет содержаться в ответе исполнителя. Только этот исполнитель имеет право передать данные задатчику. Но как только этот ответ появляется на шине, любой другой исполнитель также может принять эти данные. Таким образом, для того чтобы передать данные от одного исполнителя другому, совершенно необязательно пересылать их непосредственно через задатчика.

Протокол LIN подразумевает использование RC-цепочки в качестве задающего генератора микроконтроллеров исполнителей. Поэтому каждое сообщение содержит поле синхронизации и каждый исполнитель обязан подстроить по этому полю частоту своего приемопередатчика. Для того, чтобы определить время передачи одного бита, необходимо засечь время четырех периодов стартовой посылки, разделить на 8 и округлить (рис. 4).

Рис. 4. Байт синхронизации (Synch Field)



В идентификационном поле сообщается информация о том, что же, собственно, последует дальше. Поле идентификации (рис. 5) разделено на три части: четыре бита (0–3) содержат адрес исполнителя, с которым будет производиться обмен информацией, два бита (4–5) указывают количество передаваемых байт и последние два бита (6–7) используются для контроля четности. Четыре бита адреса могут выбирать одного из 16-ти исполнителей, каждый исполнитель может отвечать 2-мя, 4-мя,

Рис. 5. Поле идентификации

P1	P0	ID5	ID4	ID3	ID2	ID1	ID0	ID5	ID4	Date Bytes
P0: Бит четности ID0 ⊕ ID1 ⊕ ID2 ⊕ ID4		ID5		ID4		Date Bytes		0	0	2
P1: Бит четности ID1 ⊕ ID3 ⊕ ID4 ⊕ ID5		ID5		ID4		Date Bytes		0	1	2
ID0-3: Адрес устройства		ID5		ID4		Date Bytes		1	0	4
ID4-5: Длина посылки		ID5		ID4		Date Bytes		1	1	8

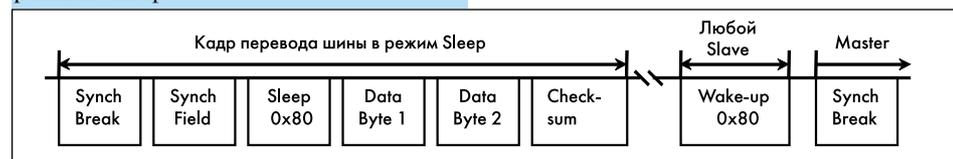
или 8-ю байтами, таким образом получаем 64 типа различных сообщений на шине. Спецификация LIN не устанавливает каких-либо жестких рамок на передаваемую информацию (за исключением команды «Sleep»), оставляя свободу творчества для программистов.

Задатчик может послать команду всем исполнителям перейти в микромощный режим («Sleep»), выставив в поле идентификации байт 0x80 (рис. 6). Исполнители, приняв его, освобождают шину и переходят в «спящий» режим с выходом из него по изменению состояния на шине. Любой исполнитель может активизировать шину, передав байт 0x80. После этого все узлы ожидают дальнейших опросов задатчика в обычном режиме.

#### Программная реализация

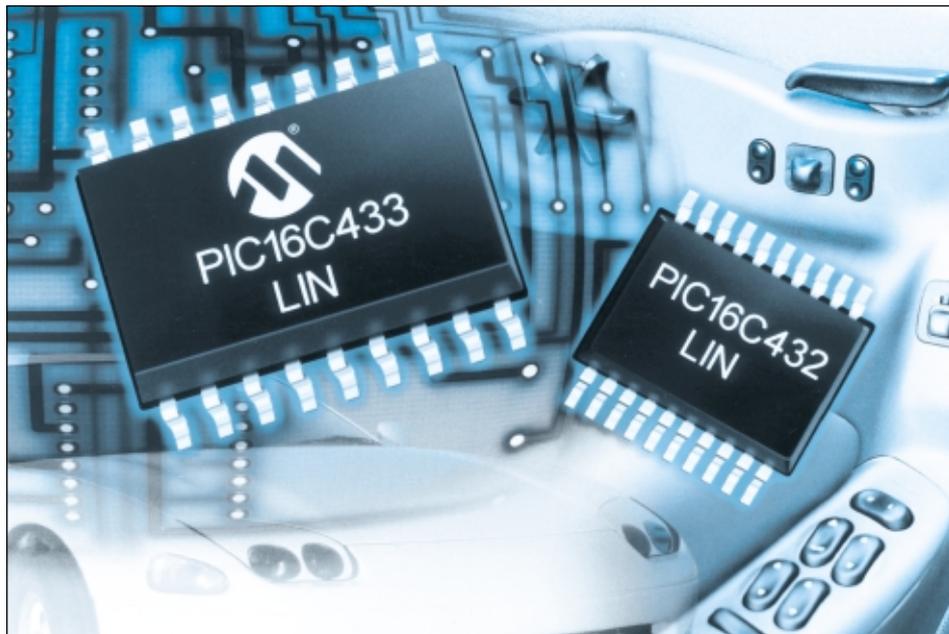
Протокол LIN можно организовать программно на любом микроконтроллере, выпускаемом фирмой Microchip (рис. 7). Очень удобно для этих целей применять малогабаритные и дешевые PIC12C508 и PIC16C505. Практически все реализуется программно, необходимо лишь соединить микроконтроллер с шиной через приемопередатчик на дискретных элементах. На сайте фирмы [www.microchip.com](http://www.microchip.com) находится пример такой конструкции и приведен исходный текст программы микроконтроллера (Application Note AN729).

Рис. 6. Включение и выключение режима «Sleep»



#### Аппаратная реализация

Для удобства проектирования встроенных систем управления для автомобильных применений, Microchip представила семейство из двух микроконтроллеров PIC16C432 и PIC16C433 с аппаратно-встроенным приемопередатчиком автомобильного протокола обмена данными LIN. Эти микроконтроллеры содержат на кристалле аппаратный приемопередатчик, и его не придется создавать на отдельных элементах. При этом снижается себестоимость и экономится место на печатной плате изделия. Специально спроектированные для автомобильных применений, эти микроконтроллеры имеют небольшой корпус, и призваны объединить различные автомобильные подсистемы и модули в единую, надежную и многофункциональную бортовую систему автомобиля.



При классической архитектуре PIC16C432/433 имеет 2 К×14 бит слов однократно-программируемой программной памяти, 128 байт оперативной памяти данных. Имея на одном кристалле микроконтроллер и приемопередатчик LIN в корпусе с 18 и 20 выводами, можно до предела сократить количество внешних навесных деталей, повысив при этом надежность устройства в целом. А наличие четырехканальных 8-битных АЦП позволяет обрабатывать аналоговые сигналы.

схемное программирование (ICSP™), power-on reset (POR), power-up timer (PWRT), oscillator start-up timer (OST), режим пониженного энергопотребления SLEEP, возможность выбора типа задающего генератора, сторожевой таймер (WDT) с отдельным RC-генератором для повышения надежности, PIC16C432 также имеет функцию brown-out reset (BOR).

#### Средства разработки и отладки

Фирма Microchip предлагает полный программно-аппаратный комплект для разработки систем на базе микроконтроллеров и протокола LIN, призванный уменьшить время, необходимое для разработки, повысить интенсивность труда и таким образом снизить затраты на разработку в целом и сократить время выхода готового изделия на рынок. Все программное обеспечение создается при помощи бесплатной среды MPLAB™. Так же предлагается стартовый комплект, в который входит демонстрационная плата с узлами-исполнителями и устройство-здатчик, которое может обмениваться данными с исполнителями и выдавать принимаемую информацию через последовательный порт RS-232 на персональный компьютер. Состоит из девяти различных плат, соединяющихся с двумя отдельными панелями. В комплект входят:

- 1 плата управления;
- 1 плата Master устройства LIN Bus;
- 3 привода устройств LIN Bus;
- 2 макетные платы для устройств LINBus;
- 1 привод управления креслом;
- 1 плата декодера системы доступа.

В комплект так же входит все необходимое программное обеспечение, в том числе исходные тексты программ для устройств Master/Slave на ассемблере.

Рис. 7. Пример реализации Slave-устройства

