

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Владимирский государственный университет  
имени Александра Григорьевича и Николая Григорьевича Столетовых»

# БОРТОВЫЕ СЕТИ ТРАНСПОРТНЫХ СРЕДСТВ

Учебное пособие



Владимир 2019

УДК 629.3.05.(075.8)

ББК 39.33–04я73

Б83

**Авторы:** А. А. Кобзев, Ю. Е. Мишулин,  
В. А. Немонтов, А. О. Веселов

**Рецензенты:**

Доктор технических наук, профессор  
профессор кафедры тепловых двигателей и энергетических установок  
Владимирского государственного университета  
имени Александра Григорьевича и Николая Григорьевича Столетовых  
*С. Г. Драгомиров*

Кандидат технических наук, доцент  
начальник лаборатории **испытания электроприводов**  
ПАО «НИПТИЭМ»  
*Р. В. Родионов*

Печатается по решению редакционно-издательского совета ВлГУ

**Бортовые** сети транспортных средств : учеб. пособие /  
А. А. Кобзев [и др.] ; Владим. гос. ун-т им. А. Г. и Н. Г. Столе-  
товых. – Владимир : Изд-во ВлГУ, 2019. – 219 с. – **ISBN**

Разработано в соответствии с Федеральными Государственными образова-  
тельными стандартами высшего образования по направлениям подготовки 13.03.02  
«Электроэнергетика и электротехника», 15.03.06 «Мехатроника и робототехника»,  
15.04.06 «Мехатроника и робототехника». Изложены вопросы построения бортовых  
информационно-управляющих систем транспортных средств, предназначенных для  
автоматизации рабочих процессов управления, диагностики, сбора, обработки и  
отображения информации на транспортных средствах как мехатронных системах.

Подготовлено для студентов бакалавриата и магистратуры дневной формы  
обучения, а также может быть использовано при обучении студентов заочной фор-  
мы обучения по соответствующим курсам.

Табл. 27. Ил. 77. Библиогр.: 27 назв.

УДК 629.3.05.(075.8)

ББК 39.33–04я73

**ISBN**

© ВлГУ, 2019

## ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ .....	5
<b>1. СОСТАВ И СТРУКТУРА БОРТОВЫХ СЕТЕЙ МОБИЛЬНЫХ ТРАНСПОРТНЫХ СРЕДСТВ .....</b>	<b>8</b>
1.1. СТРУКТУРА БИУС.....	8
1.2. ОСНОВНЫЕ ФУНКЦИИ БИУС.....	9
1.3. ПРОТОКОЛЫ ОБМЕНА .....	10
1.4. ТИПЫ МУЛЬТИПЛЕКСНЫХ ШИН БОРТОВОГО ОБОРУДОВАНИЯ .....	14
1.5. АВТОМАТИЗИРОВАННОЕ РАБОЧЕЕ МЕСТО ВОДИТЕЛЯ МНОГООСНОГО КОЛЁСНОГО ШАССИ.....	21
КОНТРОЛЬНЫЕ ВОПРОСЫ К ГЛАВЕ 1.....	25
<b>2. СТРУКТУРНАЯ ОРГАНИЗАЦИЯ ВЫЧИСЛИТЕЛЬНЫХ СЕТЕЙ ПОДВИЖНЫХ ОБЪЕКТОВ .....</b>	<b>27</b>
2.1. СЕТЬ MODBUS.....	27
2.1.1. <i>Физический уровень MODBUS</i> .....	29
2.1.2. <i>Канальный уровень MODBUS</i> .....	30
2.1.3. <i>Прикладной уровень MODBUS</i> .....	36
2.1.4. <i>Передача в сети MODBUS</i> .....	37
КОНТРОЛЬНЫЕ ВОПРОСЫ К РАЗДЕЛУ 2.1.....	38
2.2. СЕТЬ PROFIBUS .....	39
2.2.1. <i>Физический уровень PROFIBUS</i> .....	40
2.2.2. <i>Канальный уровень PROFIBUS DP</i> .....	42
2.2.3. <i>Резервирование в PROFIBUS</i> .....	49
2.2.4. <i>Описание устройств сети PROFIBUS</i> .....	50
2.2.5. <i>Протоколы сети PROFIBUS</i> .....	51
2.2.6. <i>Программирование PROFIBUS-DP с помощью STEP 7</i> .....	55
КОНТРОЛЬНЫЕ ВОПРОСЫ К РАЗДЕЛУ 2.2.....	56
2.3. СЕТЬ CAN.....	57
2.3.1. <i>Физический уровень CAN</i> .....	58
2.3.2. <i>Передающая среда</i> .....	65
2.3.3. <i>Формат кадра в сети CAN</i> .....	79
2.3.4. <i>Аппаратная реализация сети CAN</i> .....	81
2.3.5. <i>Протоколы верхнего уровня HLP</i> .....	86

2.3.6. Сравнительная характеристика основных HLR протоколов.....	101
2.3.7. Элементная база для реализации сети CAN.....	107
КОНТРОЛЬНЫЕ ВОПРОСЫ К РАЗДЕЛУ 2.3.....	113
2.4. СЕТЬ LIN .....	114
2.4.1. Конфигурация шины LIN.....	117
2.4.2. Физический уровень интерфейса .....	120
2.4.3. Программная реализация .....	123
2.4.4. Аппаратная реализация .....	123
2.4.5. Элементная база для реализации сети LIN .....	125
КОНТРОЛЬНЫЕ ВОПРОСЫ К РАЗДЕЛУ 2.4.....	140
2.5. FLEXRAY.....	140
2.5.1. Основные характеристики FlexRay.....	146
КОНТРОЛЬНЫЕ ВОПРОСЫ К РАЗДЕЛУ 2.5.....	153
2.6. СЕТЬ ETHERCAT .....	153
КОНТРОЛЬНЫЕ ВОПРОСЫ К РАЗДЕЛУ 2.6.....	159
3. РЕКОМЕНДАЦИИ ПО СТРУКТУРНОЙ ОРГАНИЗАЦИИ ВЫЧИСЛИТЕЛЬНОЙ СЕТИ .....	160
ЗАКЛЮЧЕНИЕ.....	164
БИБЛИОГРАФИЧЕСКИЙ СПИСОК.....	165
ПРИЛОЖЕНИЯ .....	168

## ВВЕДЕНИЕ

В настоящее время техническая оснащенность автомобилей различными электронными системами значительно возросла. Последние достижения в области электроники и микропроцессоров способствовали повышению надежности, эргономичности и безопасности автомобилей. Классификация современных электронных систем автомобилей [25] показана на рисунке В-1.



Рисунок В-1 – Классификация электронных систем автомобилей

ЭСУА - электронные системы управления автомобилем;

ЭСУД - электронные системы управления двигателем;

СБСА - специализированные бортовые системы автомобиля;

ЛВС - локальные вычислительные сети.

Электронные системы управления автомобилем подразделяются на два типа: обеспечивающие безопасность при движении автомобиля и улучшающие управляемость и эргономичность автомобиля. Например, система курсовой устойчивости предотвращает увод автомобиля в неуправляемый занос, а автоматическая коробка перемены передач облегчает управление автомобилем.

Электронные системы управления двигателем включают в себя системы впрыска, зажигания и пуска двигателя и другие приборы электрооборудования автомобиля, обеспечивающие надежную работу

двигателя и его высокий КПД. Специализированные бортовые системы автомобиля включают в себя различные системы отображения информации, необходимой для водителя; системы, повышающие комфортабельность автомобиля; системы навигации; противоугонные устройства.

Эти системы в той или иной степени связаны друг с другом. Выходные сигналы некоторых датчиков могут использоваться несколькими электронными системами. Можно применять один компьютер для управления всеми автомобильными системами (но на текущий момент и в ближайшем будущем это экономически нецелесообразно). Начинает претворяться в жизнь другое техническое решение, когда контроллеры отдельных ЭБУ связываются друг к другом коммуникационной шиной для обмена данными. Датчики и исполнительные механизмы, подключенные к данной шине через специальные согласующие устройства, становятся доступными для всех ЭБУ. Это решение представляет собой локальную вычислительную сеть (ЛВС) на борту автомобиля.

Локальные вычислительные сети являются мультиплексными системами передачи информации. Они позволяют значительно уменьшить число проводов, контактов и звеньев связи на автомобиле.

Термин «мультиплексный» широко используется в автомобильной промышленности. Обычно его относят к последовательным каналам передачи данных между различными электронными устройствами автомобиля. Несколько проводов, по которым передаются управляющие сигналы, заменяются шиной для обмена данными. Уменьшение количества проводов в электропроводке автомобиля - одна из причин разработки мультиплексных систем. Другая причина - необходимость объединения в ЛВС контроллеров различных ЭБУ для эффективной работы и диагностики.

Объединение вычислительных устройств транспортного средства в единый программно-аппаратный комплекс часто называют бортовой информационно-управляющей системой (БИУС) [7].

Бортовая информационно-управляющая система (БИУС), предназначена для автоматизации рабочих процессов управления, диагностики, сбора, обработки и отображения информации на транспортных средствах.

Структурно современные БИУС во многом predetermined, и задача создания БИУС состоит в поиске такого варианта реализации, который, с одной стороны, выполняет функционал, возложенный на БИУС конкретного изделия, а с другой стороны, использует традиционно доминирующие в данной сфере технологии, доступную элементную базу, отраслевые стандарты и т. д.

Подход к рассмотрению структурной организации современных БИУС изложенный в данном учебном пособии отражает анализ и оценку существующий подходов в части таких параметров, как: иерархичность (одноуровневые и многоуровневые сети); количество узлов в сети (подсети) (минимальное и максимальное); архитектура сети; использование смешанных сетей; анализ протоколов верхнего и нижнего уровня. Рассмотрены наиболее распространенные вычислительные системы с позиции аппаратной, программной реализации, особенности протокола обмена и распределение на модуле по требованию быстродействия.

# 1. СОСТАВ И СТРУКТУРА БОРТОВЫХ СЕТЕЙ МОБИЛЬНЫХ ТРАНСПОРТНЫХ СРЕДСТВ

## 1.1. Структура БИУС

БИУС – вид автоматизированной системы управления, человеко-машинная система, предназначенная для автоматизации рабочих процессов управления, диагностики, сбора, обработки и отображения информации на транспортных средствах. БИУС облегчает управление и упрощает эксплуатацию сложных систем, выполняя за водителя часть функций.

Бортовая информационно-управляющая система (БИУС) транспортного средства (ТС) это комплекс автоматической оценки состояния автомобиля и его агрегатов, оценки параметров движения, для информирования водителя об этих состояниях и выработки управляющих воздействий на исполнительные механизмы автомобиля в процессе выполнения различных задач [4].

БИУС представляет собой распределённую вычислительную сеть для жёстких условий эксплуатации (широкий температурный диапазон, вибрация, удары и другие внешние воздействия), объединяющую центральный процессорный блок и периферийные контроллеры различных агрегатов автомобиля [1]: ДВС, АБС/ЛБС, управление КПП, управление климатом в кабине и предпусковым подогревателем, управление подкачкой колёс [17] и т. д. В качестве физической среды передачи информации в БИУС наиболее широко используется CAN [0, 13, 22] в виде дифференциальной витой пары. Это обусловлено фактическим доминированием в настоящее время данного стандарта промышленной сети в автомобильной технике, в том числе и военной [22, 19, 23, 8, 9]. Преимуществами CAN являются отлаженность стандарта, широкие возможности в выборе физической среды передачи данных, развитая аппаратная поддержка, в том числе с обработкой ошибок, высокая надёжность, поддержка режима реального времени, ориентирование на распределённые системы управления [0, 13].

Структурно современные БИУС во многом предопределены, и задача создания БИУС состоит в поиске такого варианта реализации, который, с одной стороны, выполняет функционал, возложенный на БИУС конкретного изделия, а с другой стороны, использует традиционно доминирующие в данной сфере технологии, доступную элементную базу, отраслевые стандарты и т. д.

## 1.2. Основные функции БИУС

БИУС осуществляет как информационные функции в отношении водителя, так и управляющие функции в отношении агрегатов и устройств автомобиля.

В соответствии с [16] основные функции БИУС следующие:

- управление источником энергии (двигателем внутреннего сгорания, газотурбинным двигателем, генератором, накопителем электрической энергии), преобразователями энергии, потребителями энергии;
- управление основными системами: рулевым управлением, подвеской, трансмиссией, тормозной;
- управление противоаварийными и вспомогательными системами: системой противостолкновения, освещением и световой сигнализацией, вентиляцией, климатической, навигационной, диагностической и др.;
- информационное обеспечение водителя.

### Примеры выпускаемых систем

БИУС для автомобилей разрабатываются в:

- ОАО «НКБ ВС», г. Таганрог [11];
- ЗАО «НТЦ «ЭЛИНС», г. Зеленоград [26];
- ООО «НТП «ДЕКА», г. Москва [10];
- ООО «НИЦ СК «Континент», г. Москва [1];
- УП «ИЦТ «Горизонт» (Беларусь) [3];
- ОАО «СКБ «Камертон» (Беларусь) [5].

В ОАО «НКБ ВС» (г. Таганрог) разработана электронная система общего управления, контроля и диагностики [24, 12], которую в ряде источников называют БИУС. Данная система предназначена для автоматизации общих процессов управления, контроля и диагностики на автомобилях семейств «Мустанг», «Волк», «Тигр», «Тайфун», специальных колёсных шасси и другой колесной техники, оснащённых агрегатами с электронным управлением.

В ООО «НТП «ДЕКА» (г. Москва) создана бортовая сеть нового поколения [10], которую по функциональному назначению и техническим решениям также можно отнести к БИУС. Сеть состоит из унифицированных модулей: блока органов управления, унифицированного функционального контроллера и дисплея. Взаимодействие между блоками осуществляется по шине CAN.

Унитарное предприятие «Институт цифрового телевидения «Горизонт» (Республика Беларусь, г. Минск) разработало БИУС, реализующую в основном информационные функции [3]. БИУС устанавливается на большегрузные автомобили и тягачи и в своём составе имеет блок сопряжения и управления, блок обработки информации, два видеомодуля на цветных жидкокристаллических панелях, телевизионную видеокамеру заднего/бокового обзора и акустическую систему.

### **1.3. Протоколы обмена**

Программная составляющая БИУС во многом определяется её аппаратной реализацией. В частности, реализация БИУС на тех или иных вычислительных устройствах определяет необходимость (или её отсутствие) использования операционной системы.

Принципиальным моментом является наличие требования работы БИУС в режиме реального времени, когда задержка в реакции системы на событие сверх определённого времени может привести к негативным последствиям [15]. На практике подавляющее большинство бортовых систем в части управляющих функций должны работать в режиме реального времени. В настоящее время существует множество операционных систем реального времени. Среди них есть и отечественные разработки [14], в том числе военного и космического назначения.

В случае использования микроконтроллерных вычислительных устройств появляется ряд преимуществ: отсутствует необходимость выбора и использования операционной системы, уменьшается время загрузки и готовности устройств к работе после подачи электропитания.

Для организации обмена информацией между блоками БИУС по CAN-сети используются следующие стандарты протоколов [17] высокого уровня (прикладной уровень): CAL, CANopen, DeviceNet, SDS, CANKingdom, SeleCAN, SAEJ1939 - и низкого уровня (физический уровень): ISO 11898, ISO 11783, SAEJ2284, ISO11992.

Наиболее распространённой системой классификации сетевых протоколов является теоретическая модель OSI (базовая эталонная модель взаимодействия открытых систем, англ. Open Systems Interconnection Basic Reference Model). Спецификация этой модели была окончательно принята в 1984 году Международной Организацией по

Стандартизации (ISO). В соответствии с моделью OSI протоколы делятся на 7 уровней, расположенных друг над другом, по своему назначению – от физического (формирование и распознавание электрических или других сигналов) до прикладного (API для передачи информации приложениями). Взаимодействие между уровнями может осуществляться, как вертикально, так и горизонтально. В горизонтальном взаимодействии программам требуется общий протокол для обмена данными. В вертикальном – посредством интерфейсов.

### **Прикладной уровень**

Прикладной уровень – уровень приложений (англ. Applicationlayer) обеспечивает взаимодействие сети и приложений пользователя, выходящих за рамки модели OSI. На этом уровне используются следующие протоколы: HTTP, gopher, Telnet, DNS, SMTP, SNMP, CMIP, FTP, TFTP, SSH, IRC, AIM, NFS, NNTP, NTP, SNTP, XMPP, FTAM, APPC, X.400, X.500, AFP, LDAP, SIP, ITMS, Modbus TCP, VACnet IP, IMAP, POP3, SMB, MFTP, BitTorrent, eD2k, PROFIBUS.

### **Представительский уровень**

Представительский уровень (англ. Presentationlayer) – уровень представления данных. На этом уровне может осуществляться преобразование протоколов и сжатие/распаковка или кодирование/декодирование данных, а также перенаправление запросов другому сетевому ресурсу, если они не могут быть обработаны локально. Запросы приложений, полученные с уровня приложений, он преобразует в формат для передачи по сети, а полученные из сети данные преобразует в формат, понятный приложениям. К этому уровню традиционно относят следующие протоколы: HTTP, ASN.1, XML-RPC, TDI, XDR, SNMP, FTP, Telnet, SMTP, NCP, AFP.

### **Сеансовый уровень**

Сеансовый уровень (англ. Sessionlayer) управляет созданием/завершением сеанса связи, обменом информацией, синхронизацией задач, определением права на передачу данных и поддержанием сеанса в периоды неактивности приложений. Синхронизация передачи обеспечивается помещением в поток данных контрольных точек, начиная с которых возобновляется процесс при нарушении взаимодействия. Используемые протоколы: ASP, ADSP, DLC, Named Pipes, NBT, NetBIOS, NWLink, Printer Access Protocol, Zone Information Protocol, SSL, TLS, SOCKS.

### **Транспортный уровень**

Транспортный уровень (англ. Transport layer) организует доставку данных без ошибок, потерь и дублирования в той последовательности, как они были переданы. Разделяет данные на фрагменты равной величины, объединяя короткие и разбивая длинные (размер фрагмента зависит от используемого протокола). Используемые протоколы: TCP, UDP, NetBEUI, AEP, ATP, IL, NBP, RTMP, SMB, SPX, SCTP, DCCP, RTP, TFTP.

### **Сетевой уровень**

Сетевой уровень (англ. Network layer) определяет пути передачи данных. Отвечает за трансляцию логических адресов и имён в физические, за определение кратчайших маршрутов, коммутацию и маршрутизацию, за отслеживание неполадок и заторов в сети. Используемые протоколы: IP, IPv6, ICMP, IGMP, IPX, NWLink, NetBEUI, DDP, IPSec, ARP, RARP, DHCP, BootP, SKIP, RIP.

### **Канальный уровень**

Канальный уровень (англ. Datalinklayer) предназначен для обеспечения взаимодействия сетей на физическом уровне. Полученные с физического уровня данные проверяет на ошибки, если нужно исправляет, упаковывает во фреймы, проверяет на целостность, и отправляет на сетевой уровень. Канальный уровень может взаимодействовать с одним или несколькими физическими уровнями. Спецификация IEEE 802 разделяет этот уровень на 2 подуровня - MAC (Media Access Control) регулирует доступ к разделяемой физической среде, LLC (Logical Link Control) обеспечивает обслуживание сетевого уровня. Используемые протоколы: STP, ARCnet, ATM, DTM, SLIP, SMDS, Ethernet, FDDI, Frame Relay, Local Talk, Tokenring, Star Lan, L2F, L2TP, PPTP, PPP, PPPoE, PROFIBUS.

### **Физический уровень**

Физический уровень (англ. Physical layer) предназначен непосредственно для передачи потока данных. Осуществляет передачу электрических или оптических сигналов в кабель или в радиоэфир и, соответственно, их приём и преобразование в биты данных в соответствии с методами кодирования цифровых сигналов. Используемые протоколы: RS-232, RS-422, RS-423, RS-449, RS-485, ITU-T, xDSL, ISDN, T1, E1, 10BASE-T, 10BASE2, 10BASE5, 100BASE-T, 1000BASE-T, 1000BASE-TX, 1000BASE-SX.

В мировой практике, среди сетей общего применения, наиболее широко распространен протокол HTTP (англ. Hyper Text Transfer Protocol – «протокол передачи гипертекста»). Относится к прикладному и представительскому уровням теоретической модели OSI. HTTP базируется на технологии «клиент-сервер».

Области промышленного применения: создание удаленных диспетчерских пунктов, Web-приложения для SCADA систем, программное обеспечение промышленных контроллеров, организация видеонаблюдения.

Для организации взаимодействия между элементами автоматизации в промышленных сетях передачи данных широко применяется коммуникационный протокол MODBUS. Существуют три основные реализации протокола MODBUS, две для передачи данных по последовательным линиям связи, как медным EIA/TIA-232-E (RS-232), EIA-422, EIA/TIA-485-A (RS-485), так и оптическим и радио: MODBUS RTU и MODBUS ASCII, и для передачи данных по сетям Ethernet поверх TCP/IP: MODBUS TCP. Максимальное количество сетевых узлов в сети MODBUS - 31. Протяженность линий связи и скорость передачи данных зависит от физической реализации интерфейса. Элементы сети MODBUS взаимодействуют, используя клиент-серверную модель, основанную на транзакциях, состоящих из запроса и ответа.

Области промышленного применения: организация связи датчиков и исполнительных механизмов с контроллером, связь контроллеров и управляющих компьютеров, связь с датчиками, контроллерами в SCADA системах.

В Европе широкое распространение получила открытая промышленная сеть PROFIBUS (PROcess FieLd BUS). Изначально, прототип этой сети был разработан компанией Siemens для своих промышленных контроллеров.

PROFIBUS объединяет технологические и функциональные особенности последовательной связи полевого уровня. Она позволяет объединять разрозненные устройства автоматизации в единую систему на уровне датчиков и приводов. Сеть PROFIBUS основывается на нескольких стандартах и протоколах, использует обмен данными между ведущим и ведомыми устройствами (протоколы DP и PA) или между несколькими ведущими устройствами (протоколы FDL и FMS).

Сеть PROFIBUS можно ассоциировать с тремя уровнями модели OSI: физический, канальный и уровень приложений.

Области промышленного применения: организация связи датчиков и исполнительных механизмов с контроллером, связь контроллеров и управляющих компьютеров, связь с датчиками, контроллерами и корпоративными сетями, в SCADA системах.

#### 1.4. Типы мультиплексных шин бортового оборудования

Любая промышленная сеть, в том числе автомобильная, представляет собой совокупность датчиков, исполнительных механизмов, вычислительных устройств и органов управления, объединенных системой передачи данных и взаимодействующих по правилам, задаваемым протоколом. Протокол - центральный элемент, определяющий характеристики и возможности связанных им систем.

Шинные системы автомобиля можно разделить на три класса [2]: А, В и С, отличающихся скоростью передачи данных и областями применения (таблица 1.1).

Таблица 1.1 – Классификация шинных систем автомобиля

Класс А	
Скорость передачи данных	Низкая до 10 кбит/с
Применение метода	Объединение в сеть исполнительных органов и датчиков
Реализация через	LIN, PCI5
Класс В	
Скорость передачи данных	Средняя до 125 кбит/с
Применение метода	Сложные механизмы обработки ошибок. Объединение в сеть блоков управления в области обеспечения комфорта
Реализация через	Низкоскоростную шину CAN
Класс С	
Скорость передачи данных	Высокая до 1 Мбит/с
Применение метода	Требования к возможностям работы в реальном времени Объединение в сеть блоков управления в области управления приводом и шасси
Реализация через	Высокоскоростную шину CAN

Класс С+	
Скорость передачи данных	Очень высокая до 10 Мбит/с
Применение метода	Требования к возможностям работы в реальном времени Объединение в сеть блоков управления в области управления приводом и шасси
Реализация через	FlexRay

К сетям класса С предъявляются особо жесткие требования, поскольку они по одному каналу связи обслуживают наиболее ответственные системы автомобиля, а передаваемые по ним сообщения могут быть как периодическими, так и случайными. Такие сети должны быть не только надежными и защищенными от внешних воздействий, но и обеспечивать возможность расстановки приоритетов различным сообщениям, сигнализировать об ошибках в передаче управляющих сигналов, иметь скорость реакции на важное сообщение определенной длительности.

Автомобильные системы, независимо от их класса, выполняются по различным топологическим схемам сетей. Основные виды топологий: «звезда», «кольцо» и «шина».

В схеме «звезда» есть центральный узел, связанный с каждым устройством системы отдельным каналом связи, т.е. для связи двух или более таких устройств необходимо, чтобы информация прошла через «центр». Плюс у схемы один - простота протоколов обмена информацией, недостатков, к сожалению, гораздо больше, и они явно перекрывают этот плюс. В их числе: большое время задержки и значительное число проводов; ограниченное число коммутируемых устройств; низкая надежность из-за наличия центрального узла. Схема используется редко.

В схеме «кольцо» все устройства равноправны, так как последовательно объединены в кольцо. Значит, передаваемые сигналы должны проходить по нескольким звеньям, этим обусловлены и недостатки схемы: потеря работоспособности при разрыве цепи или выходе из строя одного устройства; большая задержка и ее увеличение при добавлении нового звена.

Схема «шина» позволяет устройствам функционировать в общей среде передачи данных, используя широкополосную пере-

дачу; не требует доработок при подключении дополнительных устройств; в ней возможна реализация любого типа доступа к среде передачи данных, а время их передачи невелико. Самая важная задача протокола здесь - решение вопросов доступа в среду передачи данных.

Очевидно, что для автомобиля предпочтительнее именно эта схема: она экономит провода, обеспечивает высокую надежность системы управления. Схема «шина» реализует доступ трех типов: основной узел по определенным правилам опрашивает дочерние узлы; получив от синхронизирующего пакета сигнал, отправляет данные тому дочернему узлу, который соответствует полученному от пакета сигналу; получив сигнал от дочернего узла, открывает последнему доступ в сеть.

В электропроводке автомобиля есть несколько типов шин передачи данных, по которым происходит обмен командами и другой информацией между электронными блоками управления [21]. Конфигурация шин зависит от модели и уровня комплектации конкретного автомобиля. В таблице 1.2 перечислены типы мультиплексных шин бортового оборудования:

Таблица 1.2 – Параметры мультиплексных шин бортового оборудования

<b>Шина</b>	<b>Скорость передачи данных</b>	<b>Протокол</b>
Высокоскоростная шина сети контроллеров CAN	500 Кбит/с	11 898
Среднескоростная шина CAN	125 Кбит/с	11 898
Мультимедийная шина MOST	24 Мбит/с	Объединение MOST
Видеоинтерфейс GVIF	1,95 Гбит/с	Оригинальная разработка Sony
Локальная шина LIN	9,6 Кбит/с	Консорциум LIN

Шина CAN является высокоскоростной линией передачи данных, по которой ECU автоматически передают информацию по шине каждые несколько микросекунд. Скорость передачи информации по другим шинам намного ниже. Эти шины, главным образом, реагиру-

ют на «события», т.е. передают сообщения только в ответ на запросы ECU или при изменении состояния переключателей и датчиков.

В автомобиле используются две шины CAN:

- ↯ Среднескоростная шина
- ↯ Высокоскоростная шина

Среднескоростная шина соединяет перечисленные ниже электронные устройства:

- ↯ Объединенный головной блок аудиосистемы (IHU) или стандартный головной блок
- ↯ Панель управления отопителем и вентиляцией салона
- ↯ Отопитель, работающий на топливе
- ↯ Система помощи при парковке
- ↯ Контроллер системы наблюдения за давлением воздуха в шинах
- ↯ Центральный коммутационный блок

Высокоскоростная шина соединяет перечисленные ниже электронные устройства:

- ↯ Датчик угла поворота рулевого колеса
- ↯ Пневматическая подвеска
- ↯ Стояночный тормоз с электроприводом
- ↯ Контроллер дифференциала заднего моста
- ↯ Контроллер системы TERRAIN RESPONSE™
- ↯ Контроллер системы пассивной безопасности
- ↯ Электронный блок управления двигателем
- ↯ Контроллер коробки передач
- ↯ Контроллер адаптивных передних фар
- ↯ Контроллер антиблокировочной системы тормозов (ABS)

Обе шины CAN (высоко- и среднескоростная) соединены с одного конца с приборной панелью, а с другого - с диагностическим разъемом. Оконечным устройством среднескоростной шины является в центральный коммутационный блок (CJB), а высокоскоростной шины - контроллер ABS.

Мультимедийная шина MOST предназначена для передачи данных в рамках информационно-развлекательной системы.

MOST - синхронная сеть. Ведущее устройство сети задает время, а все остальные устройства работают согласно синхронизиру-

щим импульсам этого таймера. Для сети MOST ведущим устройством является объединенный головной блок аудиосистемы.

Видеоинтерфейс GVIF - разработка, принадлежащая компании Sony, предназначенная для передачи видеoinформации от передатчика к дисплею. В данном случае она используется только для передачи видеосигналов от компьютера навигационной системы к сенсорному дисплею (TSD).

Локальная шина LIN. В автомобиле есть две шины локальной сети передачи данных (LIN). Одна из них соединяет датчик дождя, потолочный люк и устройство запоминания положения сиденья с центральным коммутационным блоком, а вторая соединяет VbUs (тревожная сирена с автономным питанием) также с центральным коммутационным блоком.

Для правильного понимания принципов работы электронных узлов автомобиля, необходимо четко представлять назначение электронных узлов и структуру информационных шин, которые объединяют узлы в единую информационно-управляющую систему автомобиля.

На рисунке 1.1 в качестве примера представлена структурная схема электронного оборудования автомобиля OPEL [20]

Информационно-управляющая система автомобиля фактически является локальной сетью микроконтроллеров, которые объединены набором информационных шин, основанных на CAN протоколе. Набор этот состоит из трех основных шин, несколько отличающиеся по скорости обмена и по физической реализации.

HS CAN (High-Speed CAN) - это высокоскоростная двухпроводная шина, к которой подключены основные модули, отвечающие за работу двигателя, коробки передач и системы рулевого управления. Эти модули и обеспечивают главную функцию автомобиля, функцию движения, управления двигателем (ECM) и коробкой передач (TCM). К ней же подключены модули управление тормозной системой (ABS/ESP), усилителя руля (EHPS) и головным светом (AHL/AFL). Обмен информации между модулями очень интенсивный и поэтому шина имеет высокую скорость обмена.

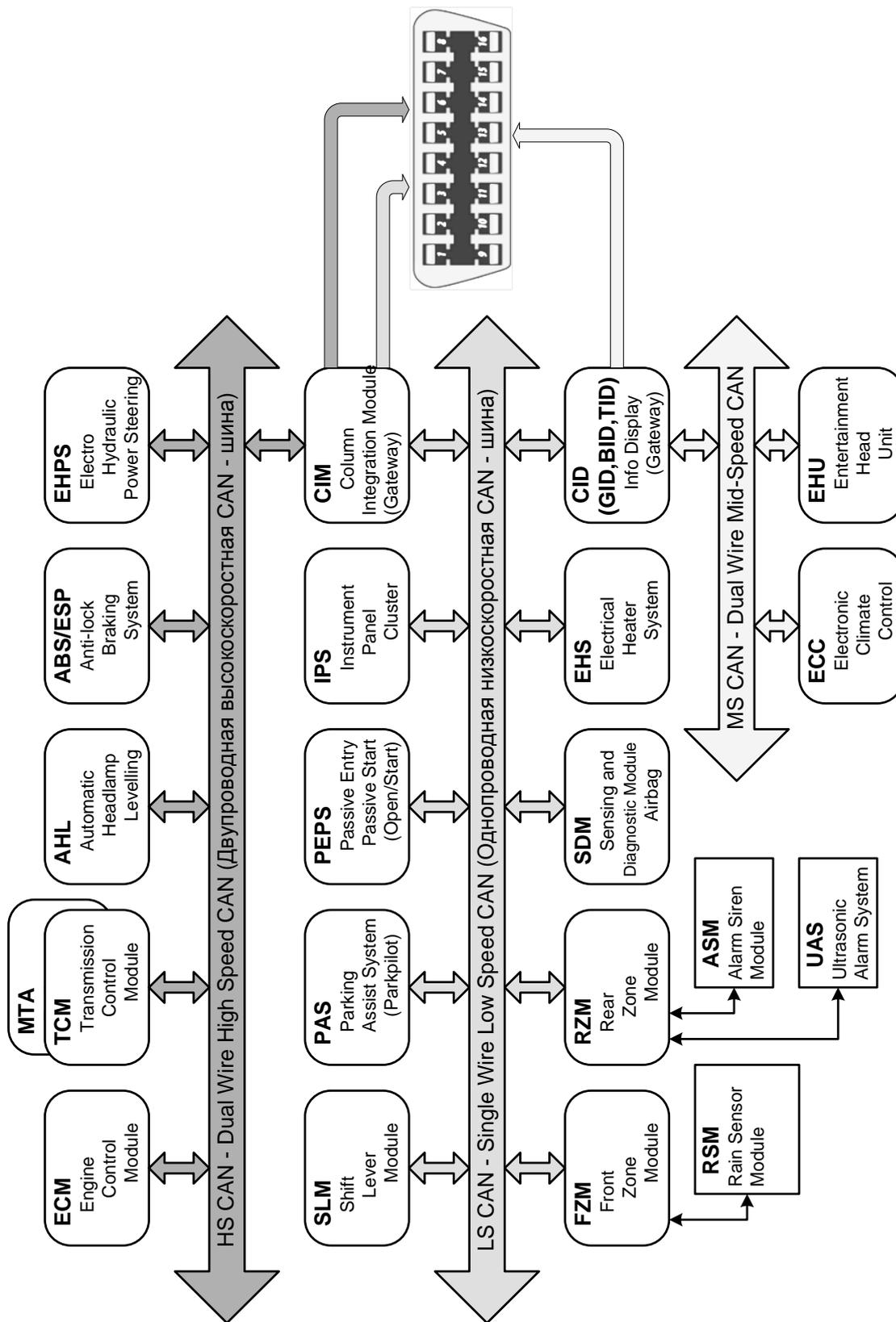


Рисунок 1.1 – Мультиплексные шины бортового оборудования

MS CAN (Mid-speed CAN) - это среднескоростная двухпроводная шина, к которой подключаются модули, отвечающие за работу информационно-развлекательных систем автомобиля. Передача информации между этими модулями так же требует повышенной скорости, но информация не так критична для работы автомобиля в целом. На этой шине находятся музыкальный центр, управление климатической установкой и информационный дисплей.

LS CAN (Low-speed CAN) - низкоскоростная однопроводная шина. Часто эту шину называют «комфорт». Эта шина объединяет блоки управления, отвечающие за автоматизацию кузовных элементов. Это различное управление электромеханикой замков, стеклоподъемников, индикацию приборной панели. Эта шина объединяет укрупненные блоки, такие как подкапотный центр электроники и задний центр электроники. Непосредственный опрос датчиков и управление исполнительными устройствами происходит по дополнительным последовательным шинам (LIN)

Шины данных сообщаются между собой при помощи шлюзов, построенных на блоках CIM и DIS. Эти шлюзы позволяют объединить все три шины в единую информационную систему автомобиля.

На представленном рисунке приведены следующие аббревиатуры и сокращения [18]:

**ECM** (Engine Control Module) — Контроллер системы управления двигателем

**MTA** (Manual Transmission — Automatic Shift), механическая коробка передач с автоматическим переключением

**TCM** (Transmission Control Module) — Контроллер автоматической коробки передач

**AHL** (Automatic Headlamp Levelling) Автоматическая система регулирования угла наклона фар

**ABS** (Antilock Braking System) Антиблокировочная система

**ESP** (Electronic Stability Program) — Электронная система динамической стабилизации

**EHPS** (Electro Hydraulic Power Steering) — Рулевое управление с электрогидравлическим сервоусилителем

**SLM** (Shift Lever Module) — Контроллер рычага переключения передач

**PAS** (Parking Assist System) Система облегчения парковки

**PEPS** Open & Start (система управления центральным замком и пуском двигателя)

**IPC** (Instrument Panel Cluster) — Комбинация приборов

**CIM** (Column Integration Module) — Контроллер рулевой колонки

**FZM** (Front Zone Module) — Контроллер моторного отсека

**RSM** (Rain Sensor Module) — Модуль датчика дождя

**RZM** (Rear Zone Module) — Контроллер задней части автомобиля

**ASM** (Alarm Siren Module) — Контроллер системы сигнализации

**UAS** (Ultrasonic Alarm System) — Система ультразвуковой сигнализации

**SDM** (Sensing & Diagnostic Module), модуль датчиков и диагностики

**EHS** (Electrical Heater System) — Дополнительный электронагреватель

**CID** (Colour Info Display) — Цветной информационный дисплей

**GID** (Graphic Info Display) — Графический информационный дисплей(монохромный)

**VID** (Board Computer Info Display) — Информационный дисплей бортового компьютера

**TID** (Triple Info Display) — Трехстрочный информационный дисплей

**ECC** (Electronic Climate Control) — Электронный климат-контроль

**EHU** (Entertainment Head Unit) — Главный блок аудиосистемы

Более подробный список аббревиатур и сокращений, принятый в автомобилях представлен в Приложении 1.

### **1.5. Автоматизированное рабочее место водителя многоосного колёсного шасси**

Автоматизированное рабочее место водителя многоосного колёсного шасси (АРМВ), разработанное в научно-исследовательском центре спцткхники и конверсии «Континент» [1] – объединенная система, включающая в себя две независимые системы: систему диагностики технологических параметров автомобиля и систему управления электрооборудованием.

АРМВ обеспечивает:

↯ выработку команд для исполнительных органов шасси (уменьшение частоты вращения коленчатого вала двигателя, глушение двигателя, торможение шасси и т.д.);

↯ вывод в кабине водителя аудио- и видеоинформации от датчиков шасси, камер телевизионного наблюдения дорожной обстановки и других устройств;

↗ накопление информации о состоянии шасси, водителя, условиях движения и возникших аварийных ситуациях в бортовом накопителе информации.

На рисунке 1.2 представлена структурная схема АРМВ.

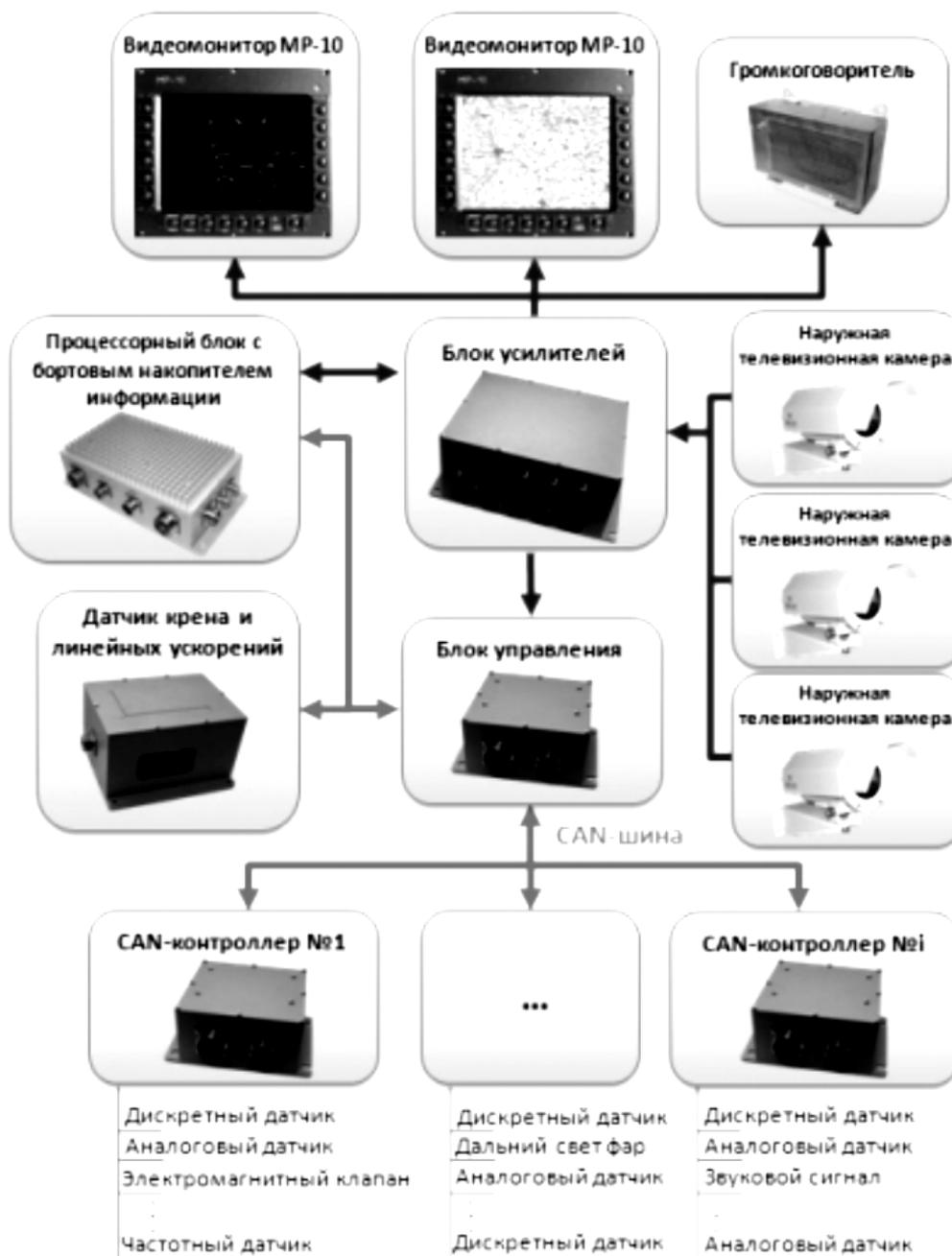


Рисунок 1.2 – Структурная схема АРМВ.

В состав АРМВ входят блок управления (БУ), процессорный блок с бортовым накопителем информации (ПБНИ), цветные видеомониторы типа МР-10, блок усилителей, наружные телевизионные

камеры высокого разрешения VNN-743, CAN-контроллеры, датчик крена и линейных ускорений (ДКЛЮ), комплект средств объективного контроля состояния бодрствования водителя (СОК СБ), комплект программных средств АРМВ.

*Блок управления обеспечивает:*

↯ сбор и обработку информации от CAN-контроллеров (датчики шасси) и передачу её в процессорный блок;

↯ формирование сигналов для исполнительных механизмов: стоп сигналы, указатели поворота, ближний и дальний свет, габаритный свет, звуковой сигнал, очистители ветрового стекла, омыватель ветрового стекла, отопитель и т.д.;

↯ сбор информации от датчика крена и ускорений;

↯ формирование сигналов для модуляторов нагнетания и сброса давления в тормозных контурах;

↯ формирование сигналов при аварийных ситуациях для глушения двигателя и перевода на холостые обороты.

Блок управления имеет следующие технические характеристики:

↯ Напряжение питания – 22...29 В.

↯ Информационные каналы: CAN – 2 шт., RS-232 – 1 шт.

↯ 16 силовых выходов управления с максимальным длительным током по каждому каналу 40 А.

↯ Количество адресных входов - 4 шт.

↯ Количество счётных входов - 2 шт.

↯ 16 универсальных входов (аналоговые и дискретные).

↯ Диапазон измеряемого напряжения аналоговых входов – 0÷30 В.

↯ Максимальное напряжение дискретного входа – 30 В.

↯ Максимальное напряжение счетного входа – 30 В.

↯ Максимальная частота входного сигнала счетного входа – 1 МГц.

↯ Диапазон рабочих температур - от –50 до +45 оС.

*Процессорный блок с бортовым накопителем информации (ПБНИ) состоит из герметичного корпуса, внутри которого размещены адаптер 4xRS-232/422/485, преобразователь USB-CAN, два адаптера Compact Flash, блок питания, два одноплатных компьютера, плата видеозахвата. Процессорный блок обеспечивает:*

↯ преобразование информации, получаемой от датчиков шасси, формирование экранных форм и вывод их на видеомонитор;

- ↯ хранение информации, полученной от датчиков шасси;
- ↯ формирование голосовых сообщений;
- ↯ оцифровка видеоинформации, получаемой от видеокамер;
- ↯ оцифровка звуковой информации и её хранение.

Технические характеристики ПБНИ:

- ↯ Напряжение питания – 22...29 В.
- ↯ Потребляемая мощность - не более 20 Вт.
- ↯ Информационные каналы: CAN – 2 шт., RS-232/422/485 – 4 шт.

- ↯ Время выхода на рабочий режим – не более 3 мин.
- ↯ Диапазон рабочих температур - от –50 до +45 оС.

*Блок усилителей* обеспечивает:

- ↯ усиление видеосигналов от камер;
- ↯ усиление видеосигналов от процессорного блока к видеомониторам;
- ↯ усиление звукового сигнала от процессорного блока к громкоговорителю;
- ↯ защиту аппаратуры от просадок напряжения, переплюсовки и короткого замыкания.

*CAN-контроллеры* обеспечивают:

- ↯ сбор информации от датчиков шасси, передачу её в блок управления;
- ↯ формирование сигналов для исполнительных механизмов: стоп сигналы, указатели поворота, ближний и дальний свет, габаритный свет, звуковой сигнал, очистители ветрового стекла, омыватель ветрового стекла, отопитель и т.д.

*Датчик крена и линейных ускорений (ДКЛУ)* обеспечивает:

- ↯ определение углов крена, тангажа, азимута;
- ↯ определение ускорений по трём осям.

Технические характеристики:

- ↯ Напряжение питания – 22...29 В.
- ↯ Защита от переплюсовки и перенапряжения.
- ↯ Информационные каналы: CAN – 2 шт., RS-232/422/485 – 4 шт.
- ↯ Диапазон измерения угловой скорости вращения –  $\pm 300$  град/с  $\pm 15\%$ .
- ↯ Рабочий диапазон измерений, g – 0÷2 (0÷19,6) м/с<sup>2</sup>.
- ↯ Предел допускаемой основной погрешности в рабочем диапазоне измерений – не более  $\pm 2\%$  от верхнего предела измерения.

Σ Суммарная погрешность, складывающаяся из основной и дополнительной погрешности, вызванной изменением температуры окружающего воздуха, во всем рабочем диапазоне температур – 5 %.

На рисунке 1.3. показан пример размещения АРМВ на четырехосном тягаче.

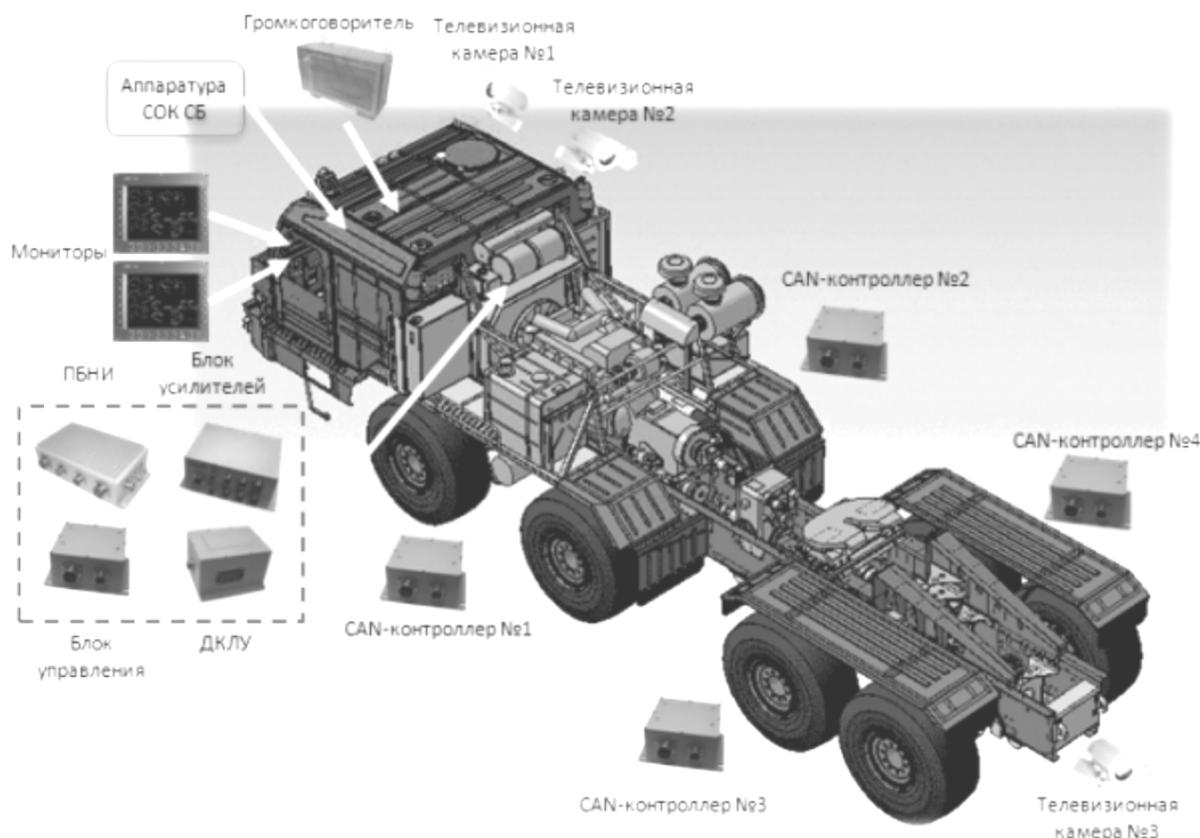


Рисунок 1.3 – Размещение АРМВ на четырехосном тягаче

Видеомониторы, аппаратура СОК СБ и громкоговоритель устанавливаются в кабине, остальные составные части – на шасси.

### Контрольные вопросы к главе 1

1. Что представляет собой БИУС?
2. Каковы основные функции БИУС?
3. Какие основные разработчики БИУС для автомобилей?
4. Какие стандарты протоколов высокого уровня используются для организации обмена информацией между блоками БИУС по CAN-сети?
5. Что такое прикладной уровень?

6. Что такое представительский уровень?
7. Чем управляет сеансовый уровень?
8. Что организует транспортный уровень?
9. Что определяет сетевой уровень?
10. Для чего предназначен канальный уровень?
11. Каково назначение физического уровня?
12. Какие основные типы мультиплексных шин бортового оборудования?
13. Что такое шина CAN?
14. Каково предназначение оптоволоконной шины MOST?
15. Для передачи какой информации предназначена шина GVIIF?
16. Каково назначение шины локальной сети передачи данных LIN?

## 2. СТРУКТУРНАЯ ОРГАНИЗАЦИЯ ВЫЧИСЛИТЕЛЬНЫХ СЕТЕЙ ПОДВИЖНЫХ ОБЪЕКТОВ

### 2.1. Сеть MODBUS

Протокол MODBUS и сеть MODBUS являются самыми распространенными в мире. Несмотря на свой возраст (стандартом де-факто MODBUS стал еще в 1979 году), MODBUS не только не устарел, но, наоборот, существенно возросло количество новых разработок и объем организационной поддержки этого протокола. Миллионы MODBUS-устройств по всему миру продолжают успешно работать, а последняя версия описания протокола появилась в декабре 2006 г.

*Среди основных преимуществ MODBUS следует выделить:*

- отсутствие необходимости в специальных интерфейсных контроллерах (PROFIBUS и CAN требуют для своей реализации заказные микросхемы);
- простота программной реализации и элегантность принципов функционирования.

Все это снижает затраты на освоение стандарта как системными интеграторами, так и разработчиками контроллерного оборудования. Высокая степень открытости протокола обеспечивается также полностью бесплатными текстами стандартов, которые можно найти на сайте [www.modbus.org](http://www.modbus.org).

В России MODBUS по распространенности конкурирует только с PROFIBUS. Популярность протокола в настоящее время объясняется, прежде всего, совместимостью с большим количеством оборудования, которое имеет протокол MODBUS. Кроме того, MODBUS имеет высокую достоверность передачи данных, связанную с применением надежного метода контроля ошибок. MODBUS позволяет унифицировать команды обмена благодаря стандартизации номеров (адресов) регистров и функций их чтения-записи.

*Основным недостатком MODBUS является сетевой обмен по типу «ведущий/ведомый», что не позволяет ведомым устройствам передавать данные по мере их появления и поэтому требует интенсивного опроса ведомых устройств ведущим.*

Разновидностями MODBUS являются протоколы MODBUS PLUS - многомастерный протокол с кольцевой передачей маркера и

MODBUS TCP, рассчитанный на использование в сетях ETHERNET и интернет.

Протокол MODBUS имеет два режима передачи: RTU (Remote Terminal Unit – «удаленное терминальное устройство») и ASCII. Стандарт предусматривает, что режим RTU в протоколе MODBUS должен присутствовать обязательно, а режим ASCII является опциональным. Пользователь выбирает необходимый режим вместе с другими параметрами (скорость передачи, режим паритета и т.д.) во время конфигурации каждого контроллера. Пользователь может выбирать любой из них, но все модули, включенные в сеть MODBUS, должны иметь один и тот же режим передачи.

В стандарте MODBUS имеются обязательные требования, рекомендуемые и опциональные (необязательные). Существует три степени соответствия стандарту: «полностью соответствует» - когда протокол соответствует всем обязательным и всем рекомендуемым требованиям, «условно соответствует» - когда протокол соответствует только обязательным требованиям и не соответствует рекомендуемым, и «не соответствует». Модель OSI протокола MODBUS содержит три уровня: физический, канальный и прикладной. В таблице 2.1 представлена модель OSI для MODBUS.

Таблица 2.1 – Модель OSI для MODBUS

Номер уровня	Название уровня	Реализация
7	Прикладной	MODBUS Application Protocol
6	Уровень представления	Нет
5	Сеансовый	Нет
4	Транспортный	Нет
3	Сетевой	Нет
2	Канальный (передачи данных)	Протокол «ведущий/ведомый» Режимы RTU и ASCII
1	Физический	RS-485 или RS-232

Стандарт MODBUS предусматривает применение физического интерфейса RS-485, RS-422 или RS-232. Наиболее распространенным для организации промышленной сети является 2-проводной интерфейс RS-485. Для соединений точка-точка может быть использован интерфейс RS-232 или RS-422.

### **2.1.1. Физический уровень MODBUS**

В новых разработках на основе MODBUS стандарт рекомендует использовать интерфейс RS-485 с двухпроводной линией передачи, но допускается применение четырехпроводной линии и интерфейса RS-232.

MODBUS-шина должна состоять из одного магистрального кабеля, от которого могут быть сделаны отводы. Магистральный кабель MODBUS должен содержать 3 проводника в общем экране, два из которых представляют собой витую пару, а третий соединяет общие ("земляные") выводы всех интерфейсов RS-485 в сети. Общий провод и экран должны быть заземлены *в одной точке*, желательно около ведущего устройства.

Устройства могут подключаться к кабелю тремя способами:

- непосредственно к магистральному кабелю;
- через пассивный разветвитель (тройник);
- через активный разветвитель (содержащий развязывающий повторитель интерфейса).

В документации на устройство и на тройник должны быть указаны наименования подключаемых цепей.

На каждом конце магистрального кабеля должны быть установлены резисторы для согласования линии передачи, как это требуется для интерфейса RS-485. В отличие от физического интерфейса RS-485, в котором терминальные резисторы на низких скоростях обмена можно не использовать, стандарт на протокол MODBUS формально требует применения терминальных резисторов для всех скоростей обмена. Их номинал может быть равным 150 Ом и мощность 0,5 Вт. Стандарт требует, чтобы в руководствах по эксплуатации устройств MODBUS было сказано, имеются ли указанные резисторы внутри устройства, или их необходимо устанавливать при монтаже сети. Если требуются внешние резисторы, то они должны иметь номинал в интервале от 450 до 650 Ом и быть установлены только в одном месте в пределах каждого сегмента сети (сегментами считаются части сети между повторителями интерфейса).

MODBUS-устройство обязательно должно поддерживать скорости обмена 9600 бит/с и 19200 бит/с, из них 19200 бит/с устанавливается «по умолчанию». Допускаются также скорости 1200, 2400, 4800, ..., 38400 бит/с, 65 кбит/с, 115 кбит/с.

Скорость передачи должна выдерживаться в передатчике с погрешностью не хуже 1%, а приемник должен принимать данные при отклонении скорости передачи до 2%.

Сегмент сети, не содержащий повторителей интерфейса, должен допускать подключение до 32 устройств, однако их количество может быть увеличено, если это допустимо исходя из нагрузочной способности передатчиков и входного сопротивления приемников, которые приведены в документации на интерфейсы. Указание этих параметров в документации является обязательным требованием стандарта.

Максимальная длина магистрального кабеля при скорости передачи 9600 бит/с и сечении жил более 0,13 кв. мм (AWG26) составляет 1 км. Отводы от магистрального кабеля не должны быть длиннее 20 м. При использовании многопортового пассивного разветвителя с N отводами длина каждого отвода не должна превышать значения 40 м/N.

MODBUS не устанавливает конкретных типов разъемов, но, если используются разъемы RJ45, mini-DIN или D-Shell, они должны быть экранированными, а цоколевки должны соответствовать стандарту.

Для минимизации ошибок при монтаже рекомендуется использовать провода следующих цветов: желтый - для положительного вывода RS-485 (на котором устанавливается логическая "1", когда через интерфейс выводится логическая "1"); коричневый - для второго вывода интерфейса RS-485; серый - для общего провода.

Типовым сечением кабеля является AWG 24 (0,2 кв. мм, диаметр провода 0,51 мм). При использовании кабеля категории 5 его длина не должна превышать 600 м. Волновое сопротивление кабеля желательно выбирать более 100 Ом, особенно для скорости обмена более 19200 бит/с.

### ***2.1.2. Канальный уровень MODBUS***

Протокол MODBUS предполагает, что только одно ведущее устройство (контроллер) и до 247 ведомых (модулей ввода-вывода) могут быть объединены в сеть. Обмен данными всегда инициируется ведущим. Ведомые устройства никогда не начинают передачу данных, пока не получают запрос от ведущего. Ведомые устройства также не могут обмениваться данными друг с другом. Поэтому в любой момент времени в сети MODBUS может происходить только один акт обмена.

Адреса с 1 по 247 являются адресами MODBUS устройств в сети, а с 248 по 255 зарезервированы. Ведущее устройство не должно иметь адреса и в сети не должно быть двух устройств с одинаковыми адресами. Ведущее устройство может посылать запросы всем устройствам одновременно ("широковещательный режим") или только одному. Для широковещательного режима зарезервирован адрес "0" (при использовании в команде этого адреса она принимается всеми устройствами сети).

#### *2.1.2.1. Описание кадра (фрейма) протокола MODBUS*

##### **ASCII фрейм**

В ASCII-режиме, сообщение начинается с "двоеточия" (:, ASCII 3A hex), и заканчивается последовательностью «возврат каретки - перевод строки» (CRLF, ASCII 0D и 0A hex). Допустимые символы для передачи - это шестнадцатеричные цифры 0-9, A-F. Монитор сетевого устройства в сети непрерывно отслеживает символ «двоеточие». Когда он принят, каждое устройство декодирует следующие поле сообщения (поле адреса) и т.д.

Интервалы между символами сообщения могут быть до 1 сек. Если интервал больше, то принимающее устройство распознает это как ошибку.

Исключение: В контроллерах типа 584 и 984A/B/X ASCII-сообщение может нормально заканчиваться после контрольной суммы без CRLF последовательности. Интервалы меньше 1 с. допускаются.

Каждый байт сообщения передается как два ASCII символа. Главное преимущество этого способа время между передачей символов может быть до 1 с. без возникновения ошибок при передаче.

##### **Формат каждого байта в ASCII-режиме:**

Система кодировки: шестнадцатеричная, ASCII-символы 0-9, A-F

Назначение битов: 1 стартовый бит

7 бит данных, младшим битом вперед

1 бит паритета; нет бита паритета

1 стоповый бит, если есть паритет; 2 бита если нет паритета

Контрольная сумма: Longitudinal Redundancy Check (LRC)

### *RTU фрейм*

Сообщения MODBUS RTU передаются в виде кадров, для каждого из которых известно начало и конец. В протоколе MODBUS RTU сообщение начинает восприниматься как новое после паузы (тишины) на шине длительностью не менее 3,5 символов (14 бит), т. е. величина паузы в секундах зависит от скорости передачи. Кадр должен передаваться непрерывно. Если при передаче кадра обнаруживается пауза продолжительностью более 1,5 шестнадцатеричных символа (6 бит), то считается, что кадр содержит ошибку и должен быть отклонен принимающим модулем. Эти величины пауз должны строго соблюдаться при скоростях ниже 19200 бит/с, однако при более высоких скоростях рекомендуется использовать фиксированные значения паузы, 1,75 мс и 750 мкс соответственно. Формат кадра показан на рисунке 2.1. На рис. обозначено: PDU – «Protocol Data Unit» - «элемент данных протокола»; ADU – «Application Data Unit» - «элемент данных приложения». Поле адреса всегда содержит только адрес ведомого устройства, даже в ответах на команду, посланную ведущим. Благодаря этому ведущее устройство знает, от какого модуля пришел ответ.

Поле «Код функции» говорит о действии, которое нужно выполнить. Поле «Данные» может содержать произвольное количество байт. В нем может содержаться информация о параметрах, используемых в запросах контроллера или ответах модуля. Поле «Контрольная сумма» содержит контрольную сумму CRC длиной 2 байта.



Рисунок 2.1 – Формат кадра протокола MODBUS RTU

### *Содержание адресного поля*

Адресное поле фрейма содержит два символа (ASCII) или 8 бит (RTU). Допустимый адрес передачи находится в диапазоне 0 - 247. Каждому подчиненному устройству присваивается адрес в пределах от 1 до 247.

Адрес 0 используется для широковещательной передачи, его распознает каждое устройство. Когда MODBUS протокол используется на более высоком уровне сети, широковещательная передача может не поддерживаться или может быть реализована другими методами.

#### *Содержание поля функции*

Поле функции фрейма содержит два символа (ASCII) или 8 бит (RTU). Диапазон числа 1 -255. Когда подчиненный отвечает главному, он использует поле кода функции для фиксации ошибки. В случае нормального ответа подчиненный повторяет оригинальный код функции. Если имеет место ошибка, возвращается код функции с установленным в 1 старшим битом. В дополнение к изменению кода функции, подчиненный размещает в поле данных уникальный код, который говорит главному какая именно ошибка произошла или причину ошибки.

#### *Содержание поля данных*

Поле данных в сообщении от главного к подчиненному содержит дополнительную информацию, которая необходима подчиненному для выполнения указанной функции. Оно может содержать адреса регистров или выходов, их количество, счетчик передаваемых байтов данных.

Например, если главный запрашивает у подчиненного прочитать группу регистров (код функции 03), поле данных содержит адрес начального регистра и количество регистров. Если главный хочет записать группу регистров (код функции 10 hex), поле данных содержит адрес начального регистра, количество регистров, счетчик количества байтов данных и данные для записи в регистры.

Поле данных может не существовать (иметь нулевую длину) в определенных типах сообщений.

#### *Содержание поля контрольной суммы*

В MODBUS - сетях используются два метода контроля ошибок передачи. Содержание поля контрольной суммы зависит от выбранного способа передачи. ASCII Когда используется ASCII-режим поле контрольной суммы содержит два ASCII-символа. Контрольная сумма является результатом вычисления Longitudinal Redundancy Check (LRC) сделанного над содержанием сообщения начиная с ":" и заканчивая CRLF. RTU Когда используется RTU-режим поле контрольной суммы содержит 16-ти битовую величину. Контрольная сумма является результатом вычисления Cyclical Redundancy Check сделанного

над содержанием сообщения. CRC добавляется к сообщению последним полем младшим байтом вперед.

### 2.1.2.2. Структура данных

Передача данных осуществляется младшим разрядами вперед. Последовательность передачи битов в ASCII- и RTU-режимах представлена на рисунках 2.2, 2.3. По умолчанию бит паритета устанавливают равным 1, если количество двоичных единиц в байте нечетное, и равным 0, если оно четное. Такой паритет называют четным (even parity) и метод контроля называют контролем четности.

Стартовый бит	1	2	3	4	5	6	7	Бит паритета	Стоп-бит
---------------	---	---	---	---	---	---	---	--------------	----------

a)

Стартовый бит	1	2	3	4	5	6	7	Стоп-бит	Стоп-бит
---------------	---	---	---	---	---	---	---	----------	----------

b)

Рисунок 2.2 – Последовательность битов в режиме ASCII

- a) последовательность передачи битов в режиме ASCII с контролем четности;
- b) последовательность передачи битов в режиме ASCII без контроля четности

Стартовый бит	1	2	3	4	5	6	7	8	Бит паритета	Стоп-бит
---------------	---	---	---	---	---	---	---	---	--------------	----------

a)

Стартовый бит	1	2	3	4	5	6	7	8	Стоп-бит	Стоп-бит
---------------	---	---	---	---	---	---	---	---	----------	----------

b)

Рисунок 2.3 – Последовательность битов в режиме RTU

- a) последовательность передачи битов в режиме RTU с контролем четности;
- b) последовательность передачи битов в режиме RTU без контроля четности

При четном количестве двоичных единиц в байте бит паритета может быть равен 1. В этом случае говорят, что паритет является нечетным (odd parity). Контроль четности может отсутствовать вообще.

В этом случае вместо бита паритета должен использоваться второй стоповый бит. И для обеспечения максимальной совместимости с другими продуктами рекомендуется использовать возможность замены бита паритета на второй стоповый бит. Ведомые устройства могут воспринимать любой из вариантов: четный, нечетный паритет или его отсутствие.

### *2.1.2.3. Методы контроля ошибок*

Стандартная MODBUS сеть использует два метода контроля ошибок. Контроль паритета (even/odd) и контрольная сумма. Обе эти проверки генерируются в головном устройстве. Подчиненное устройство проверяет каждый байт и все сообщение в процессе приема. Пользователь может устанавливать продолжительность интервала таймаута, в течении которого головное устройство будет ожидать ответа от подчиненного. Если подчиненный обнаружил ошибку передачи, то он не формирует ответ главному.

**Контроль паритета.** Пользователь может конфигурировать контроллеры на проверку четного или нечетного паритета (Even/Odd). Например, 8 бит RTU-режима содержат следующую информацию:

1100 0101 Общее количество единиц - 4. Если используется четный паритет, то бит паритета будет равен 0, и общее количество единиц будет по-прежнему четным числом. Если используется нечетный паритет, то бит паритета будет равен 1, тогда общее количество единиц вместе с битом паритета будет равно 5, т.е. нечетному числу.

**Контрольная сумма LRC.** Метод LRC проверяет содержание сообщения исключая начальный символ «:» и пару CRLF.LRC это 1 байт. LRC вычисляется передающим устройством и добавляется в конец сообщения. Принимающее устройство вычисляет LRC в процессе приема сообщения и сравнивает его с принятым от главного. Если есть несовпадение, то имеет место ошибка.

**Контрольная сумма CRC.** Контрольная сумма CRC состоит из двух байт. Контрольная сумма вычисляется передающим устройством и добавляется в конец сообщения. Принимающее устройство вычисляет контрольную сумму в процессе приема и сравнивает ее с полем CRC принятого сообщения.

Счетчик контрольной суммы предварительно инициализируется числом FF hex. Только восемь бит данных используются для вычис-

ления контрольной суммы CRC. Старт и стоп биты, бит паритета, если он используется, не учитываются в контрольной сумме.

Во время генерации CRC каждый байт сообщения складывается по исключающему ИЛИ с текущим содержимым регистра контрольной суммы. Результат сдвигается в направлении младшего бита, с заполнением нулем старшего бита. Если младший бит равен 1, то производится исключающее ИЛИ содержимого регистра контрольной суммы и определенного числа. Если младший бит равен 0, то исключающее ИЛИ не делается.

Процесс сдвига повторяется восемь раз. После последнего (восьмого) сдвига, следующий байт складывается с текущей величиной регистра контрольной суммы, и процесс сдвига повторяется восемь раз как описано выше. Конечное содержание регистра и есть контрольная сумма CRC.

### ***2.1.3. Прикладной уровень MODBUS***

Данный уровень обеспечивает коммуникацию между устройствами типа «ведущий/ведомый». Прикладной уровень является независимым от физического и канального, в частности, он может использовать протоколы Ethernet TCP/IP (MODBUS TCP/IP), MODBUS Plus (многомастерная сеть с передачей маркера), интерфейсы RS-232, RS-422, RS-485, оптоволоконные, радиоканалы и другие физические среды для передачи сигналов.

Прикладной уровень MODBUS основан на запросах с помощью *кодов функций*. Код функции указывает ведомому устройству, какую операцию оно должно выполнить. При использовании протокола прикладного уровня с различными протоколами транспортного и канального уровня сохраняется неизменным основной блок MODBUS-сообщения, включающий код функции и данные (этот блок называется PDU – «Protocol Data Unit» – «элемент данных протокола»). К блоку PDU могут добавляться дополнительные поля и тогда он называется «ADU» – «Application Data Unit» – «элемент данных приложения».

Стандартом MODBUS предусмотрены три категории кодов функций: установленные стандартом, задаваемые пользователем и зарезервированные.

Коды функций являются числами в диапазоне от 1 до 127. Коды в диапазоне от 65 до 72 и от 100 до 110 относятся к задаваемым пользователем функциям, в диапазоне от 128 до 255 зарезервированы для пересылки кодов ошибок в ответном сообщении. Код «0» не используется. Коды ошибок используются ведомым устройством, чтобы определить, какое действие предпринять для их обработки.

Поле данных в сообщении, посланном от ведущего устройства ведомому, содержит дополнительную информацию, которую ведомое использует, чтобы выполнить функцию, указанную в поле «код функции». Например, если код функции указывает, что необходимо считать данные из группы регистров устройства ввода (код функции 03 hex), то поле данных содержит адрес начального регистра и количество регистров. Если ведущее устройство посылает команду записи данных в группу регистров (код функции 10 hex), то поле данных должно содержать адрес начального регистра, количество регистров, количество байтов данных и данные для записи в регистр. Конкретное содержание поля данных устанавливается стандартом для каждой функции отдельно. В некоторых сообщениях поле данных может иметь нулевую длину.

Поле данных может содержать значения состояний дискретных входов/выходов, адреса регистров, из которых надо считывать (записывать) данные, количество байт данных, ссылки на переменные, количество переменных, код подфункций и т. п.

Если ведомый выполнил принятую от ведущего функцию, то в ответе поле «код функции» содержит ту же информацию, что и в запросе. В противном случае ведомый выдает код ошибки. В случае ошибки код функции в ответе равен коду функции в запросе, увеличенному на 128.

#### ***2.1.4. Передача в сети MODBUS***

Стандартные MODBUS-порты в контроллерах MODICON используют RS-232C совместимый последовательный интерфейс. Контроллеры могут быть соединены на прямую или через модем.

Контроллеры соединяются, используя технологию главный-подчиненный, при которой только одно устройство (главный) может

инициировать передачу (сделать запрос). Другие устройства (подчиненные) передают запрашиваемые главным устройством данные, или производят запрашиваемые действия. Типичное главное устройство включает в себя ведущий (HOST) процессор и панели программирования. Типичное подчинённое устройство - программируемый контроллер.

Главный может адресоваться к индивидуальному подчиненному или может инициировать широкую передачу сообщения на все подчиненные устройства. Подчиненное устройство возвращает сообщение в ответ на запрос, адресуемый именно ему. Ответы не возвращаются при широковещательном запросе от главного.

### **Контрольные вопросы к разделу 2.1**

1. Каковы основные преимущества протокола MODBUS?
2. Какие режимы передачи данных использует протокол MODBUS?
3. Какие существуют степени соответствия стандарту MODBUS?
4. Что представляет собой физический уровень MODBUS?
5. Какой принцип работы интерфейса RS-485?
6. Какие скорости обмена поддерживают MODBUS-устройства?
7. Как работает канальный уровень MODBUS?
8. Как описывается фрейм протокола MODBUS?
9. Какова структура данных кадра протокола MODBUS?
10. Какие используются методы контроля ошибок при передаче данных?
11. Каковы особенности прикладного уровня MODBUS?
12. Какие категории кодов функций предусмотрены стандартом MODBUS?
13. Как организована передача данных в сети MODBUS?

## 2.2. Сеть PROFIBUS

Слово PROFIBUS получено из сокращений PROcess FieLd BUS, что приблизительно переводится как «промышленная шина для технологических процессов». Стандарт PROFIBUS был первоначально принят в Германии в 1987 году, затем, в 1996 году, он стал международным (EN 50170 и EN 50254).

Сеть PROFIBUS использует только первый и второй уровни модели OSI. Один из вариантов сети, PROFIBUS FMS, использует также уровень 7. В таблице 2.2 представлена модель OSI для различных вариантов сети PROFIBUS.

Таблица 2.2 – PROFIBUS в соответствии с моделью OSI

№	Название уровня	Profibus DP	Profibus FMS	Profibus PA
7	Прикладной	Нет	Fieldbus Message Specification (FMS)	Нет
6	Представления	Нет		
5	Сеансовый			
4	Транспортный			
3	Сетевой			
2	Канальный (передачи данных)	FDL	FDL	IEC 1158-2
1	Физический	RS-485, оптоволоконный интерфейс	RS-485, оптоволоконный интерфейс	Интерфейс IEC 1158-2

PROFIBUS имеет три модификации: PROFIBUS DP, PROFIBUS FMS и PROFIBUS PA.

PROFIBUS DP (PROFIBUS for Decentralized Peripherals – «PROFIBUS для децентрализованной периферии») использует уровни 1 и 2 модели OSI, а также пользовательский интерфейс, который в модель OSI не входит. Непосредственный доступ из пользовательского приложения к канальному уровню осуществляется с помощью DDLM (Direct Data Link Mapper – «прямой преобразователь для канального уровня»). Пользовательский интерфейс обеспечивает функции, необходимые для связи с устройствами ввода-вывода и контрол-

лерами. PROFIBUS DP в отличие от FMS и PA построен таким образом, чтобы обеспечить наиболее быстрый обмен данными с устройствами, подключенными к сети.

PROFIBUS FMS (PROFIBUS с FMS протоколом) использует кроме первого и второго уровня, еще и седьмой уровень модели OSI и применяется для обмена данными с контроллерами и компьютерами на регистровом уровне. PROFIBUS FMS предоставляет большую гибкость при передаче больших объемов данных, но проигрывает протоколу DP вследствие своей сложности. PROFIBUS FMS и DP используют один и тот же физический уровень, основанный на интерфейсе RS-485, и могут работать в общей сети.

PROFIBUS PA (PROFIBUS for Process Automation – «для автоматизации технологических процессов») использует физический уровень на основе стандарта IEC 1158-2, который обеспечивает питание сетевых устройств через шину и не совместим с RS-485. Особенностью PROFIBUS PA является возможность работы во *взрывоопасной зоне*.

PROFIBUS является многомастерной сетью (с несколькими ведущими устройствами). В качестве ведомых устройств выступают обычно устройства ввода-вывода, клапаны, измерительные преобразователи. Они не могут самостоятельно получить доступ к шине и только отвечают на запросы ведущего устройства.

### ***2.2.1. Физический уровень PROFIBUS***

На физическом уровне PROFIBUS DP и FMS используют стандарт RS-485 при скорости передачи до 12 Мбит/с и с размерами сегментов сети до 32 устройств. Количество устройств можно увеличить с помощью повторителей интерфейса.

Особые требования установлены к сетевому кабелю. Он должен иметь волновое сопротивление от 135 до 165 Ом при погонной емкости не более 35 пФ/м, площадь поперечного сечения проводников более 0,34 кв. мм. и погонное сопротивление не более 110 Ом/км. Кабель должен иметь одну или две витые пары с медным экраном в виде оплетки или фольги.

Стандартом для шины PROFIBUS рекомендуется разъем D-sub (DB-9) с 9-ю контактами, цоколевка разъема приведена в таблице 2.3. На устройствах устанавливается разъем с гнездами, на кабеле - со штырьками. При необходимости иметь степень защиты IP65/67 рекомендуется

использовать цилиндрический разъем типа M12 (IEC 947-5-2), HAN-BRID или гибридный разъем фирмы Siemens.

Таблица 2.3 – Цоколевка разъема DB-9 для PROFIBUS

Контакт	Сигнал	Примечание
1	Shield	Экран
2	M24	-24 В
3	Rx/Tx-DP	Прием/передача данных (положительный вывод, провод В)
4	CNTR-P	Сигнал для управления направлением передачи, положительный вывод
5	DGND	Общий провод данных
6	VP	Напряжение питания, "+"
7	P24	+24 В
8	Rx/Tx-N	Прием/передача данных (отрицательный вывод, провод А)
9	CNTR-N	Сигнал для управления направлением передачи, отрицательный вывод

С обеих сторон линии передачи подключаются согласующие резисторы, которые конструктивно установлены во все сетевые разъемы и подключаются с помощью микропереключателей. При скоростях передачи более 1,5 Мбит/с для согласования линии дополнительно используются плоские (печатные) катушки индуктивности.

Для увеличения дальности передачи в PROFIBUS предусмотрена возможность работы с оптоволоконным кабелем. При использовании стеклянного оптоволокна дальность связи может быть увеличена до 15 км. Оптоволоконные интерфейсы выполняются в виде сменных модулей для контроллеров.

PROFIBUS PA использует физический уровень, соответствующий стандарту IEC 1158-2. Данные передаются с помощью уровней тока +9 мА и -9 мА («токовая петля»). Используется манчестерский код (логический ноль соответствует смене отрицательного тока на положительный, а логическая единица - положительного на отрицательный). Скорость передачи составляет 31,25 кбит/с, в качестве линии передачи используется витая пара в экране или без него. Один

сегмент сети может содержать до 32 устройств. Максимальная длина кабеля достигает 1,9 км. В каждом сегменте с обеих сторон кабеля подключены RC-цепочки, состоящие из последовательно соединенных конденсатора емкостью 1 мкФ и резистора сопротивлением 100 Ом. Благодаря низкой энергии передаваемого сигнала PROFIBUS PA является искробезопасной электрической цепью и может быть использован во взрывоопасных зонах.

Для передачи данных в сети PROFIBUS используется NRZ-кодирование и 11-битный формат, идентичный формату HART-протокола, включающему стартовый бит («0»), 8 бит данных младшими разрядами вперед, бит паритета (четный) и стоп-бит («1»). Бит паритета равен нулю, если количество бит в слове четное и равен единице в противном случае. Такой формат используется для всех без исключения слов, включая заголовки сообщений. При передаче слов длиннее 8 бит старший байт отправляется первым, за ним остальные в порядке старшинства.

### ***2.2.2. Канальный уровень PROFIBUS DP***

Далее будем рассматривать только PROFIBUS DP, поскольку он получил несравненно более широкое распространение, чем FMS и PA.

Канальный уровень модели OSI в PROFIBUS DP называется FDL- уровнем (Fieldbus Data Link –«промышленный канал связи»). Объект MAC (Medium Access Control –«управление доступом к каналу») на канальном уровне определяет процедуру передачи данных устройствами, включая управление правами на передачу данных через сеть. Протокол канального уровня обеспечивает выполнение следующих важных требований:

- в процессе коммуникации между ведущими устройствами необходимо обеспечить выполнение каждым из них своей задачи в течение заранее определенного интервала времени;
- взаимодействие ведущих устройств (контроллеров) с ведомыми должно происходить максимально быстро.

В сети PROFIBUS для доступа ведущих устройств к сети используется метод передачи маркера. В этом методе сеть имеет логическую топологию кольца (т. е. кольца на уровне адресов устройств) и каждое ведущее устройство получает доступ к сети только при получении маркера. Маркер выполняет роль арбитра, который предостав-

ляет устройству право доступа. По истечении определенного времени это устройство должно передать маркер следующему ведущему устройству, которое получает доступ также на время, пока маркер находится у него. Таким образом, каждому ведущему устройству выделяется точно заданный интервал времени. Этот интервал может быть установлен при конфигурировании системы.

Каждому мастеру в сети назначаются свои ведомые устройства (рисунок 2.4) В методе «ведущий/ведомый» процедуру коммуникации с ведомыми устройствами выполняет мастер, который обладает маркером. На время обладания маркером мастер становится ведущим также по отношению к другим мастерам, т.е. может выполнять с ними коммуникацию типа «мастер-мастер».

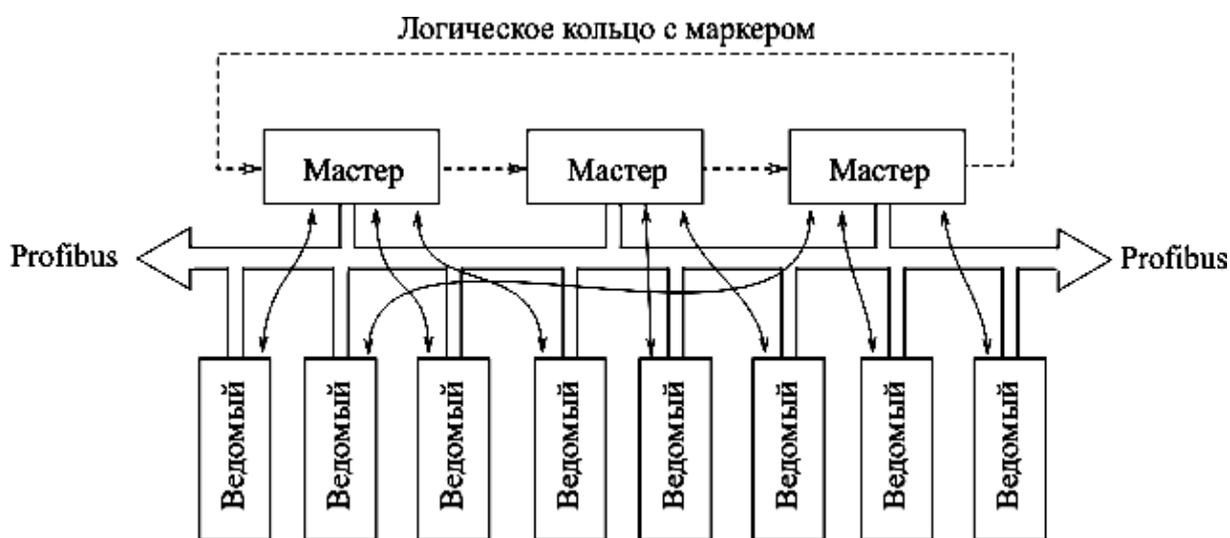


Рисунок 2.4 – Принцип работы многомастерной сети

PROFIBUS имеет также *широковещательный* режим работы, когда ведущее устройство посылает сообщение «всем», не ожидая уведомления о получении, и *многоабонентский режим*, когда ведущее устройство посылает одно и то же сообщение сразу нескольким участникам сети.

В задачи объекта MAC активного устройства (получившего маркер) входит обнаружение наличия или отсутствия маркера сразу после начала работы сети, передача маркера следующему устройству в порядке возрастания адресов, удаление адресов, вышедших из строя или выключенных устройств и добавление новых, восстановление потерянного маркера, устранение дубликатов маркеров, устранение

дублирования сетевых адресов и обеспечение заданного периода обращения маркера по сети.

Master-устройство DP (DP-master) связывает CPU контроллера с периферийными устройствами. DP-master обменивается данными с периферийными устройствами посредством PROFIBUS-DP и контролирует систему шин PROFIBUS-DP. Децентрализованные периферийные системы (slave-устройства DP) готовят данные датчиков и исполнительных элементов на месте, чтобы их можно было передать с помощью PROFIBUS-DP в CPU контроллера.

К PROFIBUS-DP можно подключать самые разнообразные устройств как в качестве master-устройств DP, так и в качестве slave-устройств DP, при условии, что их поведение соответствует стандарту IEC 61784-1:2002 Ed1 CP 3/1. Могут применяться следующие устройства:

- SIMATIC S5;
- SIMATIC S7/M7/C7;
- устройство программирования SIMATIC или ПК;
- человеко-машинный интерфейс SIMATIC, или ЧМИ НМИ (панель оператора, OP);
- станция оператора, OS;
- текстовый дисплей, TD);
- устройства других производителей.

Рисунок 2.5 иллюстрирует типичную структуру сети PROFIBUS-DP.

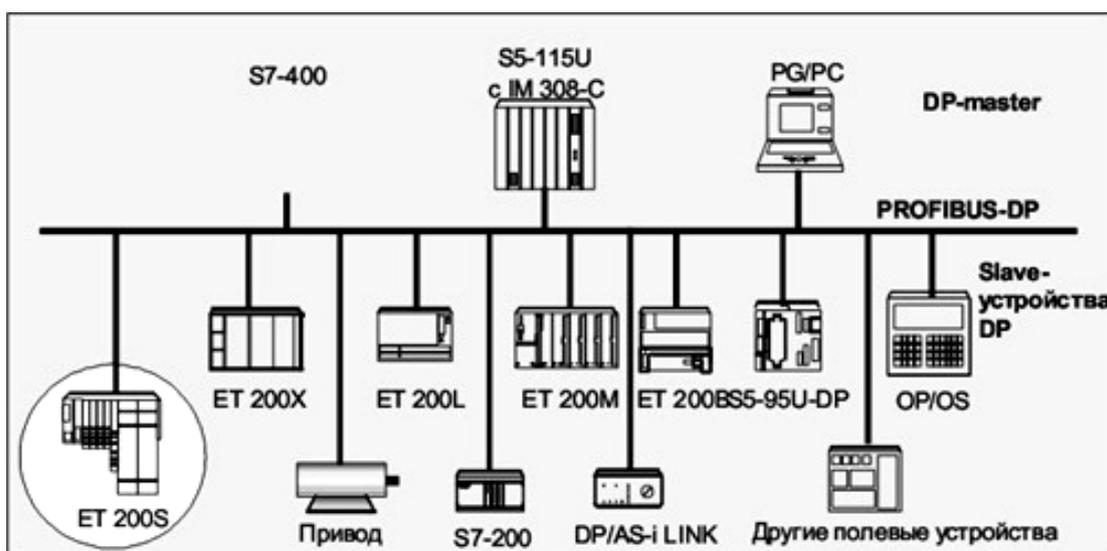


Рисунок 2.5 – Структура сети PROFIBUS-DP

DP-master встраивается в соответствующее устройство, например, в S7-400 имеется интерфейс PROFIBUS-DP, интерфейсный master-модуль IM 308-C вставляется в S5-115U. Slave-устройствами DP являются устройства децентрализованной периферии, связанные с master-устройствами DP с помощью PROFIBUS-DP.

#### *2.2.2.1. Коммуникационный профиль DP*

Основная функция коммуникационного профиля DP состоит в эффективном обмене данными ПЛК или компьютера с датчиками и исполнительными устройствами. Обмен данными с этими устройствами обычно выполняется периодически, но коммуникационный профиль DP представляет также дополнительный сервис аperiodического обмена для установки параметров, контроля режимов работы и обработки сигналов тревоги (алармов).

В обычном режиме центральный контроллер (ведущий) периодически считывает информацию, поступающую на входы ведомых устройств, и записывает информацию для их выходов. Дополнительно к этому периодическому обмену данными DP обеспечивает мощные средства для диагностики и инсталляции системы, а также для обеспечения устойчивости к внешним дестабилизирующим факторам.

Средства диагностики DP обеспечивают быстрое обнаружение места появления ошибки и пересылку соответствующего сообщения ведущему устройству. Диагностические сообщения делятся на три уровня:

- уровень устройства (касающиеся общих признаков работоспособности устройства, таких как перегрев, уход напряжения за допустимые границы и др.);
- уровень модуля ввода-вывода, входящего в состав устройства модульной конструкции;
- уровень канала модуля (например, «к. з. входа 8»).

К одной сети могут быть подсоединены до 128 устройств (но не более 32-х в одном сегменте). Спецификация для конфигурирования системы включает количество узлов сети, распределение адресов устройств, формат диагностических сообщений, параметры шины.

В сети могут использоваться устройства трех типов:

- DP мастер класса 1 (DPM1) - центральный контроллер, который циклически обменивается информацией с ведомыми устройствами с заранее определенным периодом;

- DP мастер класса 2 (DPM2) - устройство, предназначенное для конфигурирования системы, настройки, обслуживания или диагностики;
- ведомое устройство - устройство, которое выполняет сбор информации или выдачу ее исполнительным устройствам. Эти устройства могут отсылать или принимать не более 256 байт информации за один цикл обмена.

Коммуникационный профиль DP позволяет сконфигурировать как одномастерную, так и многомастерную сеть. В одномастерной сети ведущее устройство (мастер) может посылать запросы и получать ответы только от ведомых устройств.

В многомастерной сети имеется несколько ведущих устройств, которые имеют свои одномастерные подсети и в пределах подсети являются устройствами класса DPM1. Ведущие устройства в многомастерной сети могут быть также устройствами класса DPM2. Входные и выходные данные подчиненных устройств могут быть прочитаны любым мастером сети. Однако записывать данные в устройства может только один мастер, который при конфигурировании системы был обозначен как DPM1.

Ведущий контроллер (DPM1) может находиться в одном из трех состояний: «Stop» - когда не происходит обмена данными; «Clear» - когда DPM1 может считывать данные, но не может записывать их и выходы всех устройств переводятся в безопасные состояния; «Operate» - обычное рабочее состояние.

Если в системе появляется сообщение об ошибке, то DPM1 устанавливает выходы всех устройств вывода в безопасное состояние, а сам переходит в состояние «Clear». *Безопасным* считается такое состояние, при котором исполнительные устройства находятся в безопасном (для человека или системы) состоянии. Такое состояние самоконтроля системы может быть установлено или нет при ее конфигурировании. При отключенном состоянии самоконтроля система продолжает работать, несмотря на появление ошибок.

При конфигурировании системы пользователь назначает каждому ведущему свои ведомые устройства и очередность их опроса, а также указывает устройства, которые не надо опрашивать периодически.

Передача данных между мастером DPM1 и ведомыми делится на три фазы: параметризация, конфигурирование и передача данных.

В фазе параметризации и конфигурирования проверяется, соответствует ли конфигурация и параметры ведомого устройства запланированным в DPM1 установкам. Проверяется тип устройства, формат и длина передаваемых сообщений, количество входов или выходов.

PROFIBUS DP имеет режим синхронизации вывода. Для этого посылается широковещательная управляющая команда синхронизации, при получении которой происходит одновременная смена состояний выходов всех устройств вывода. Имеется также команда «замораживание», при поступлении которой входы всех устройств ввода сохраняют свое текущее состояние и перестают реагировать на изменение поступающих на входы сигналов, пока не поступит команда «размораживание». Эти команды используются для синхронизации ввода. Команды синхронизации могут посылаться всем устройствам сети, группе или одному устройству.

Для обнаружения ошибок в передающих устройствах предусмотрен механизм временного мониторинга (наблюдения), который действует как в ведомых, так и ведущих устройствах. Интервал мониторинга устанавливается при конфигурировании системы. Ведущий (DPM1) контролирует процесс передачи данных ведомым устройством с помощью таймера. Для каждого подчиненного используется свой таймер. Если в течение интервала наблюдения не приходят корректные данные, выдается диагностическое сообщение для пользователя. Если включен режим автоматической реакции на ошибки, то ведущий устанавливает все выходы в безопасные состояния, а сам переходит в состояние «Clear».

Ведомый также выполняет контроль ведущего устройства или линии передачи. Для этого используется сторожевой таймер. Если от ведущего не приходят данные в течение периода сторожевого таймера, ведомый автоматически переводит свои выходы в безопасные состояния. Для большей степени защиты выходов в многомастерной системе только один (уполномоченный) мастер имеет прямой доступ к изменению состояний выходов устройства. Все другие мастера могут считывать только "изображения" сигналов на входах и выходах устройства.

В PROFIBUS имеются также расширенные DP функции, которые позволяют передавать аperiodические функции чтения и записи, а также сигналы тревог параллельно и независимо от периодической

пересылки данных, установленной пользователем при конфигурировании системы. Это позволяет, например, с помощью DPM2 изменять параметры вновь подсоединенных ведомых устройств или считывать состояние любых устройств сети без остановки системы. Эти служебные функции выполняются аperiodически с низким приоритетом, параллельно с рабочим процессом передачи данных в системе. Для обеспечения такой возможности при параметризации всей сети устанавливают увеличенный цикл обращения маркера, чтобы шина на была загружена на 100%.

#### 2.2.2.2. Передача сообщений

PROFIBUS использует два типа сервисов для передачи сообщений: SRD (Send and Receive Data with acknowledge – «отправка и прием данных с уведомлением») и SND (Send Data with No acknowledge – «отправка данных без уведомления»).

Сервис SRD позволяет отправить и получить данные в одном цикле обмена. Этот способ обмена наиболее распространен в PROFIBUS и очень удобен при работе с устройствами ввода-вывода, поскольку в одном цикле можно и отправить, и получить данные.

Сервис SND используется, когда надо отправить данные одновременно группе ведомых устройств (многоабонентский режим) или всем ведомым устройствам (широковещательный режим). При этом ведомые устройства не отправляют свои уведомления мастеру.

Сообщение в PROFIBUS называется телеграммой. Телеграмма может содержать до 256 байт, из них 244 байта данных, плюс 11 служебных байт (заголовки телеграммы). Все телеграммы имеют заголовки одинаковой длины, за исключением телеграммы с названием Data\_Exchange. Заметим, что 11 байт служебной информации делают PROFIBUS очень неэффективным при передаче коротких сообщений. Однако при больших объемах данных такой формат телеграммы достаточно эффективен. Структура сообщения в PROFIBUS представлена на рисунке 2.6.

SD	LE	LEr	SD	DA	SA	FC	DSAP	SSAP	DU	FCS	ED
----	----	-----	----	----	----	----	------	------	----	-----	----

Рисунок 2.6 – Структура телеграммы PROFIBUS

Поля телеграммы имеют следующее содержание:

SD - стартовый разделитель. Используется для указания начала телеграммы и ее формата. Имеется четыре типа разделителей для телеграмм запроса и ответа, и один тип для короткого уведомления. Короткое уведомление имеет поле SD, но не в начале телеграммы;

LE - длина передаваемых данных (DA+SA+FC+DSAP+SSAP+DU);

LEr - повторение поля LE с целью его резервирования;

DA - адрес устройства-получателя телеграммы;

SA - адрес отправителя;

FC - код типа телеграммы (запрос, уведомление, ответ, диагностические данные, тип устройства - мастер или ведомый, приоритет, уведомление);

DSAP - устройство-получатель использует это поле чтобы определить, какой тип сервиса нужно выполнить;

SSAP - COM порт отправителя;

DU - данные длиной от 1 до 244 байт;

FCS - контрольная сумма телеграммы (сумма значений полей DA+SA+FC+DU, по модулю 255);

ED - признак конца.

### ***2.2.3. Резервирование в PROFIBUS***

С целью повышения надежности в PROFIBUS предусмотрено резервирование, выполненное следующим образом:

- ведомые устройства содержат два различных PROFIBUS-интерфейса, основной и резервный. Они могут быть либо в одном устройстве, либо в двух одинаковых устройствах (основном и резервном);
- устройства снабжаются двумя независимыми стеками протоколов со специальным расширением для резервирования;
- процесс резервирования стеков протоколов осуществляется путем запуска специального программного объекта резервирования RedCom.

В нормальном режиме коммуникация выполняется только через основное устройство, которое посылает диагностическую информацию резервному устройству. В случае, когда основное устройство дает сбой, резервное устройство берет на себя его функции. Кроме того, мастер контролирует все ведомые устройства и выдает диагностиче-

ское сообщение на верхний уровень АСУ ТП, как только в системе вышло из строя основное устройство и не осталось резервного, или, когда вышло из строя резервное. Резервное устройство может работать на основной PROFIBUS линии, или на двух, если имеется резервная.

Подход к резервированию в PROFIBUS имеет следующие свойства:

- одна и та же модификация устройств используется для реализации различных вариантов резервирования;
- ведущее, ведомое устройство и шина могут быть резервированы независимо друг от друга;
- не требуется особого дополнительного конфигурирования резервного устройства;
- возможен полный мониторинг обоих ведомых устройств.

Резервирование обеспечивает высокий коэффициент готовности, короткое время восстановления, отсутствие потерь данных и нечувствительность системы к отказам.

#### ***2.2.4. Описание устройств сети PROFIBUS***

Современные модули ввода-вывода являются интеллектуальными устройствами и выполняют многие функции, которые ранее выполнялись только контроллерами. Однако, чтобы выполнить эти функции, устройства требуют сложной настройки при инсталляции системы, при обслуживании и параметризации. Поэтому необходимо иметь точное и полное описание сведений об устройствах, таких как тип выполняемых функций, количество входов/выходов, диапазон изменения переменных, единицы измерения, значения по умолчанию, идентифицирующие параметры устройства и т. д.

PROFIBUS предлагает несколько методов и средств для описания устройств, которые обеспечивают унификацию описания. По историческим причинам в промышленной автоматизации используется в основном формат GSD (General Station Data – «общие данные об устройстве»). Описание устройств в этом формате создается их изготовителем и поставляется вместе с устройством.

Характеристики устройства описываются с помощью языка описания электронных устройств Electronic Device Description Language (EDDL) и поставляются в виде текстового файла EDD

(Electronic Device Description – «описание электронного устройства»). Интерпретатор этого описания очень хорошо апробирован для приложений средней сложности. Для сложных приложений Profibus предлагает другой программный компонент – Device Type Manager (DTM).

Текстовый файл GSD содержит как общую, так и специфичную для конкретного устройства информацию. С помощью ключевых слов средство конфигурирования может прочесть идентификационные записи, настраиваемые параметры, типы данных, допустимые значения параметров. Некоторые из ключевых слов обязательны (по стандарту), например, имя изготовителя, другие являются опционными. GSD файл делится на три секции:

- 1) *общие параметры* - содержит имя поставщика и имя устройства, версия аппаратуры и программного обеспечения, идентификационный номер, поддерживаемые скорости передачи;
- 2) *спецификация ведущего устройства* - указывает допустимое количество подключаемых ведомых устройств, параметры передачи и приема сообщений;
- 3) *спецификация ведомого устройства* - указывает количество и тип каналов ввода-вывода, перечень диагностических сообщений и список модулей при модульной конструкции устройства.

GSD файл загружается в средство конфигурирования системы «PROFIBUS Configurator» и используется при ее инсталляции.

Более мощным средством описания устройств является язык EDD, который является частью международного стандарта IEC 61804-2 и позволяет описывать устройства средней сложности. Еще более мощными являются независимые от конкретной промышленной сети средства описания устройств FDT/DTM Field Device Tool/Device Type Manager – «средство для устройств полевого уровня/менеджер типа устройства»), которые позволяют описывать очень сложные устройства.

### **2.2.5. Протоколы сети PROFIBUS**

Одни и те же каналы связи сети PROFIBUS допускают одновременное использование нескольких протоколов передачи данных.

1. PROFIBUS DP (Decentralized Peripheral - Распределенная периферия) — протокол, ориентированный на обеспечение скоростного

обмена данными между системами автоматизации (ведущими DP-устройствами) и устройствами распределённого ввода-вывода (ведомыми DP-устройствами). Протокол характеризуется минимальным временем реакции и высокой стойкостью к воздействию внешних электромагнитных полей. Оптимизирован для высокоскоростных и недорогих систем. Эта версия сети была спроектирована специально для связи между автоматизированными системами управления и распределенной периферией.

С помощью PROFIBUS DP могут быть реализованы Mono и MultiMaster системы. Основной принцип работы заключается в следующем: центральный контроллер (ведущее устройство) циклически считывает входную информацию с ведомых устройств и циклически записывает на них выходную информацию. При этом время цикла шины должно быть короче, чем время цикла программы контроллера, которое для большинства приложений составляет приблизительно 10 мс. В дополнение к циклической передаче пользовательских данных Profibus DP предоставляет широкие возможности по диагностике и конфигурированию. Коммуникационные данные отображаются специальными функциями, как со стороны ведущего, так и со стороны ведомого устройства.

Диагностические функции PROFIBUS DP позволяют быстро локализовать сбои в системе. Диагностические сообщения передаются по шине мастеру и делятся на три уровня:

- 1) Диагностика, связанная со станцией – сообщения определяют состояние всего устройства (перегрев, низкое напряжение и т.д.)
- 2) Диагностика, связанная с модулем – сообщения, связанные с ошибками в том или ином входном/выходном модуле.
- 3) Диагностика, связанная с каналом – сообщения определяют ошибку конкретного бита входа/выхода.

Поведение системы при использовании протокола DP определяется состоянием ведущего устройства. Существует три основных состояния:

- 1) **ОСТАНОВ** – в этом состоянии не происходит передачи данных между ведущим устройством и периферией;
- 2) **ОЧИСТКА** – ведущее устройство считывает информацию с ведомых устройств и держит выходы в состоянии защиты от сбоев;
- 3) **РАБОТА** – ведущее устройство находится в состоянии приёма или передачи данных с периферии.

Ведущее устройство циклически посылает информацию о своём состоянии всем ведомым устройствам, присоединённым к нему. Передача данных между ведущим и ведомым устройствами делится на три фазы:

- 1) Параметризация;
- 2) Конфигурирование;
- 3) Передача данных.

На 1 и 2 фазах ведомое устройство сравнивает свою текущую конфигурацию с конфигурацией ожидаемой ведущим устройством и только если они совпадают, происходит передача данных. В дополнение к обычной передаче пользовательских данных, ведущее устройство может посылать управляющие команды одному, группе или всем своим ведомым устройствам. Существует две таких команды. Одна переводит ведомые устройства в режим **sync** (все выходы блокируются в текущем состоянии), другая – переводит в режим **freeze** (все входы блокируются в текущем состоянии). Вывод из этих режимов происходит с помощью команд **unsync** и **unfreeze** соответственно.

В дополнение к данной системе передачи, существуют расширенные DP функции, которые позволяют производить ациклическое чтение и запись параллельно циклической передаче данных.

2. PROFIBUS PA (Process Automation – Автоматизация процесса) – протокол обмена данными с оборудованием полевого уровня, расположенным в обычных или Ex-зонах (взрывоопасных зонах). Протокол отвечает требованиям международного стандарта IEC 61158-2. Позволяет подключать датчики и приводы на одну линейную шину или кольцевую шину. PROFIBUS PA (Process Automation) – промышленная сеть, служит для соединения систем автоматизации и систем управления процессами с полевыми устройствами (например, датчиками давления, температуры и уровня). Может использоваться для аналоговой (от 4 до 20 мА) технологии. PROFIBUS PA использует основные PROFIBUS DP функции передачи измеренных величин и состояния контроллера, а также расширенные функции PROFIBUS DP для параметризации и операций с полевыми устройствами.

3. PROFIBUS FMS (Fieldbus Message Specification – Спецификация сообщений полевого уровня) – универсальный протокол для решения задач по обмену данными между интеллектуальными сетевыми устройствами (контроллерами, компьютерами/программаторами,

системами человеко-машинного интерфейса) на полевом уровне. Некоторый аналог промышленного Ethernet, обычно используется для высокоскоростной связи между контроллерами и компьютерами верхнего уровня и используемыми диспетчерами. Скорость до 12 Мбит/с. PROFIBUS FMS – протокол предназначен в основном для связи программируемых контроллеров друг с другом и станциями оператора. Он используется в тех областях, где высокая степень функциональности более важна, нежели чем быстрое время реакции системы.

При связи через FMS используются отношения типа клиент-сервер. Клиент является процессом приложения, который в качестве заказчика услуги обращается к объектам. Сервер является исполнителем услуги «Объекты».

В распоряжение клиенту предоставляются объекты связи. В качестве примера устройств, соединенных по FMS протоколу можно взять из оборудования фирмы Сименс – SIMATIC S7 с FMS-CP или, например, SIMATIC S5 с CP 5431FMS. Очень часто используется комбинированный режим работы устройств PROFIBUS FMS и PROFIBUS DP, в этом случае между мастерами и ведомыми устройствами используется протокол DP, а между самими мастерами протокол FMS:

Основная нагрузка в протоколе FMS приходится на уровень приложений. Им предоставляются коммуникационные службы, которые могут использоваться непосредственно пользователем, которые отвечают за выполнение запросов в системе клиент-сервер. Коммуникационная модель PROFIBUS FMS допускает объединение распределенных процессов приложений в общий процесс с использованием коммуникационных связей. Часть процесса приложения в полевом устройстве, которая может быть достигнута через коммуникацию, называется виртуальным полевым устройством VFD. В нем находится словарь так называемых коммуникационных объектов, через которые и происходит связь между устройствами с помощью служб. Словарь содержит описание, структуру и типы данных, а также связи между адресами внутреннего устройства коммуникационных объектов и их назначение на шине (индекс/имя).

Более подробно, словарь состоит из следующих объектов:

- ♦ заголовок - информация по структуре словаря;

- ♦ список статических типов данных – список поддерживаемых статических типов данных;
- ♦ словарь статических объектов в нем – все статические коммуникационные объекты;
- ♦ динамический список списка переменных – список всех списков переменных;
- ♦ динамический список программ – список всех программ.

Все протоколы используют одинаковые технологии передачи данных и общий метод доступа к шине, поэтому они могут функционировать на одной шине. Дополнительно к перечисленным протоколам, поддерживаются следующие возможности обмена данными:

- ♦ службы FDL (Field Data Link – Канал полевых данных), SEND/RECEIVE – Отправить/Получить, позволяют легко и быстро установить соединение с любым устройством, поддерживающим FDL;
- ♦ функции S7 позволяют оптимизировать соединение с устройствами семейства.

### 2.2.6. Программирование PROFIBUS-DP с помощью STEP 7

Программный пакет STEP 7 – базовое программное обеспечение для программирования и проектирования систем SIMATIC S7. Основные средства базового пакета STEP 7, применяемые к PROFIBUS-DP представлены на рисунке 2.7.

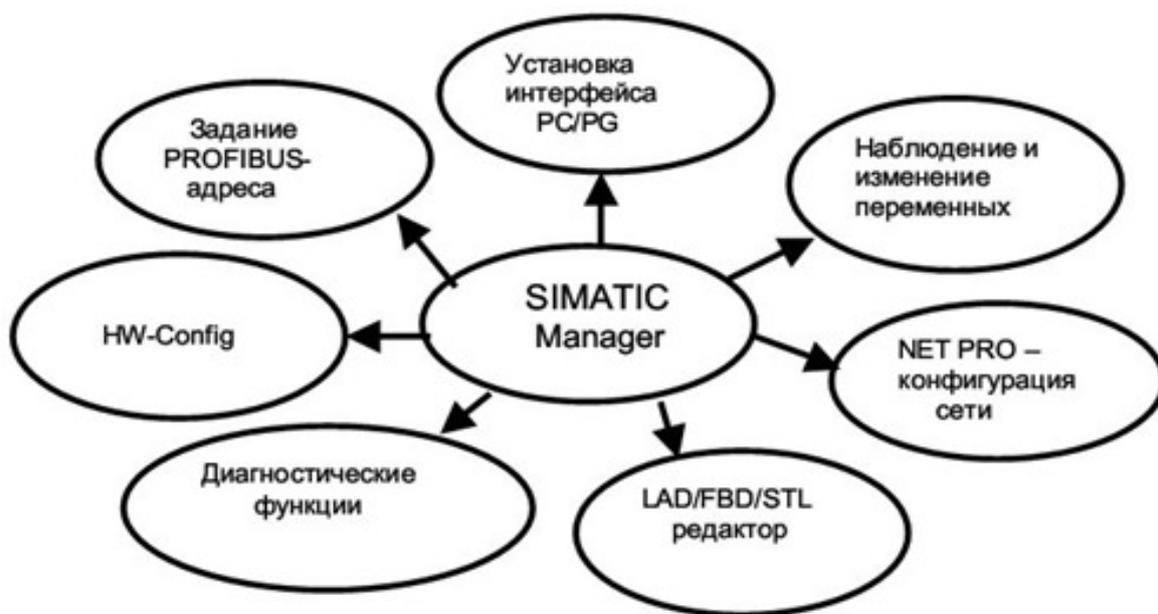


Рисунок 2.7 – STEP7-приложения для PROFIBUS-DP, вызываемые из SIMATIC Manager

Данные приложения применяются для:

- Конфигурирования и параметрирования аппаратуры;
- Конфигурирования сетей и соединений;
- Загрузки и тестирования пользовательской программы.

Благодаря ряду дополнительных пакетов, например, таких, как языки программирования SCL, S7GRAPH или HiGraph, базовый пакет STEP 7 расширяется для соответствующих приложений. С помощью центрального инструмента SIMATIC Manager все необходимые приложения могут быть вызваны в графическом виде. Все данные и установки для системы автоматизации структурированы внутри проекта и представлены в виде объектов. Пакет STEP 7 снабжен обширной online-помощью вплоть до контекстной помощи для указанного контейнера, объекта и появляющегося сообщения об ошибке.

### **Контрольные вопросы к разделу 2.2**

1. Дайте определение стандарту PROFIBUS.
2. Какие модификации имеет сеть PROFIBUS?
3. Как реализован физический уровень PROFIBUS?
4. Каковы особенности работы канального FDL-уровня модели OSI в PROFIBUS DP?
5. Как реализуется интервально-маркерный метод доступа в сети PROFIBUS?
6. Как подключаются master-устройства DP и slave-устройства DP?
7. Какова основная функция коммуникационного профиля DP?
8. Чем отличаются одномастерная и многомастерная сети?
9. Какой предусмотрен механизм для обнаружения ошибок в передающих устройствах сети PROFIBUS?
10. Какие типы сервисов для передачи сообщений использует PROFIBUS?
11. Как выполнено резервирование в сети PROFIBUS?
12. Какие методы и средства для описания устройств предлагает PROFIBUS?
13. Какие протоколы передачи данных используются в каналах сети PROFIBUS?
14. Каково назначение программного пакета STEP 7?

### 2.3. Сеть CAN

CAN протокол уже более 10 лет активно используется во всем мире. Он показал свою качественность и надежность. Сейчас на рынке представлено тысячи различных CAN изделий. CAN поддерживается развитой системой инструментальных средств, позволяющих быстро и легко проектировать CAN сети. CAN протокол реализован в кремнии, что позволяет быстро проектировать высокоэффективные, высокоскоростные, высоконадежные и в то же время дешевые системы. [0]

Основная среда передачи в CAN сетях – витая пара. Но CAN может также работать и на одном проводе (второй – корпус). Имеются примеры систем, использующие в качестве среды передачи коаксиальный кабель, оптоволокно, ИК и радио каналы, силовые линии электропередачи. CAN может работать на физическую среду RS-485.

Помимо высоконадежного алгоритма передачи и обработки ошибок, CAN протокол имеет механизм, позволяющий отключать удаленный узел и тем самым не допускать блокирование сети.

Использование глобальных часов, широковещательный (broadcast) способ передачи сообщений, мультимастерность (multiMaster) позволяет создавать полностью синхронные системы.

Так как все CAN узлы слушают все сообщения, проходящие по сети, просто реализуются приложения, работающие по управлению событиями.

CAN протокол очень хорошо подходит для построения территориально-распределенных систем управления. Используемый метод арбитража для определения приоритета сообщения и широковещательная передача позволяет просто и оптимально проектировать такие системы.

CAN-протокол разработан фирмой Robert Bosch GmbH для использования в автомобильной электронике. Данный протокол отличается повышенной помехоустойчивостью и надежностью, обладает следующими возможностями:

- конфигурационная гибкость;
- получение сообщений всеми узлами с синхронизацией по времени;
- неразрушающий арбитраж доступа к шине;
- режим мультимастер;
- обнаружение ошибок и передача сигналов об ошибках;

- автоматическая передача сбойных сообщений при получении возможности повторного доступа к шине;
- различие между случайными ошибками и постоянными отказами узлов с возможностью выключения дефектных узлов;
- работает по витой паре на расстоянии до 1 км.

Основные стандарты CAN:

- ISO 11898-1 - CAN протокол;
- ISO 11898-2 - CAN высокоскоростная физическая структура;
- ISO 11898-3 - CAN низкоскоростная физическая структура;
- ISO 11898-4 - CAN запуск;
- ISO 11898-5 - Высокоскоростное низковольтное устройство;
- ISO 11519-2 – заменен на 11898-3;
- ISO 15765 – Диагностический протокол по CAN;
- J1939 - Основной CAN протокол для грузовиков и автобусов;
- ISO 11783 - J1939 дополнение для сельхоз машин;
- ISO 11992 – определяет интерфейс тягачей и прицепов;
- ГОСТ Р ИСО 15765-1-2014;
- ГОСТ Р ИСО 15765-4-2014;
- ГОСТ Р ИСО 11898-1-2015;
- ГОСТ Р ИСО 11898-2-2015.

CAN-протокол распространяется на следующие уровни:

1. *Физический уровень* определяет, как именно будут передаваться сигналы, их электрические уровни и скорость передачи ввода/вывода.

2. *Транспортный уровень* представляет собой ядро CAN-протокола. Он отвечает за синхронизацию, арбитраж, доступ к шине, разделение посылок на фреймы, определение и передачу ошибок, и минимизацию неисправностей.

3. *Объектный уровень* обеспечивает фильтрацию сообщений и обработку сообщений и состояний.

### **2.3.1. Физический уровень CAN**

Физический уровень протокола CAN определяет сопротивление кабеля, уровень электрических сигналов в сети и т.п. В большинстве случаев используется физический уровень CAN определенный в стандарте ISO 11898.

Типичные значения напряжений в сети приведены на рисунке 2.8, причем доминирующим является логический ноль, а рецессивным, соответственно, логическая единица. Напряжение питания в сети +5В.

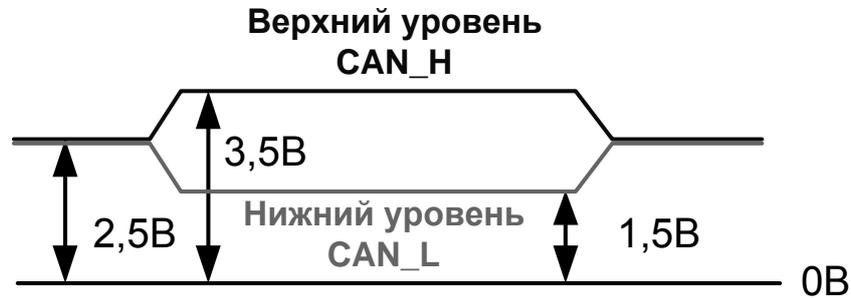


Рисунок 2.8 – Определение доминантного и рецессивного битов

CAN устройства соединяются с помощью дифференциальной шины, которая имеет две линии CAN\_H (can-high) и CAN\_L (can-low), по которым передаются сигналы. Логический ноль регистрируется, когда на линии CAN\_H сигнал выше, чем на линии CAN\_L. Логическая единица - в случае, когда сигналы CAN\_H и CAN\_L одинаковы (отличаются менее чем на 0,5 В). Использование такой дифференциальной схемы передачи делает возможным работу CAN сети в очень сложных внешних условиях. Логический ноль - называется доминантным битом (d), логическая единица – рецессивным (r) (Изначально в спецификации CAN рассматривались именно доминантный (d) и рецессивный (r) сигнал, а не лог. «0» и лог.«1»). Эти названия отражают приоритет логической единицы и нуля на шине CAN. При одновременной передаче в шину логического нуля и единицы, на шине будет зарегистрирован только логический ноль (доминантный сигнал), а логическая единица будет подавлена (рецессивный сигнал).

CAN шина использует NRZ (Non-Return To Zero) с bit-stuffing. Уровень NRZ сигнала неизменен в течение каждого битового времени. Это может привести к рассинхронизации различных узлов вследствие смещения опорной частоты. Структура фрейма сети CAN представлена на рисунке 2.9.

Bit-stuffing (бит-стаффинг) представляет собой следующую процедуру: при наличии последовательности более 5 одноименных бит вставляется дополнительный инвертированный бит. CAN – кадр при этом удлиняется (рисунок 2.10). При приеме сообщения лишние биты удаляются. Процедура бит-стаффинга определяется протоколом верхнего уровня.

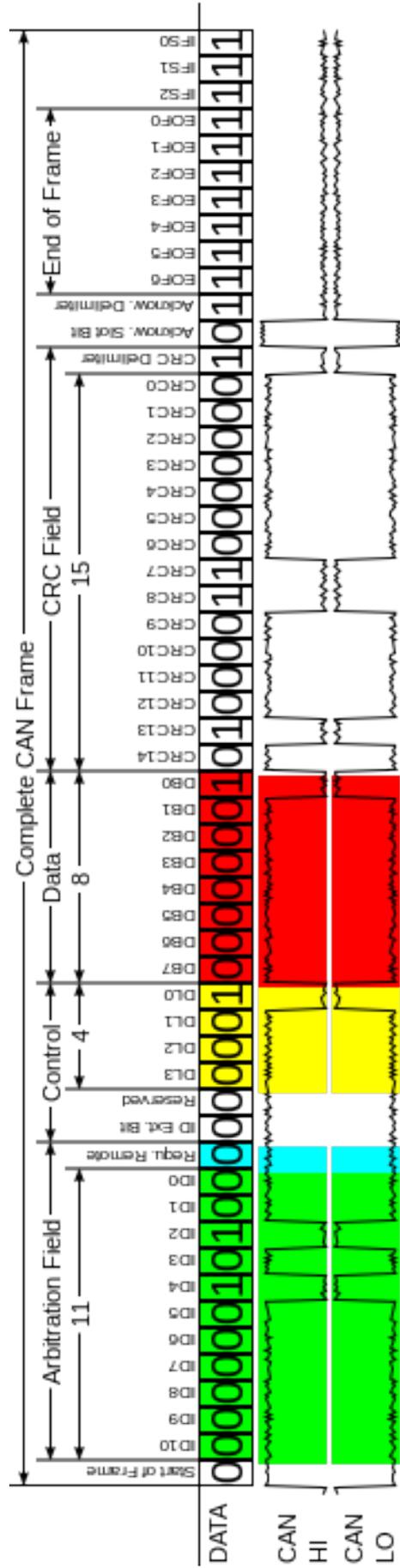


Рисунок 2.9 – Структура фрейма сети CAN

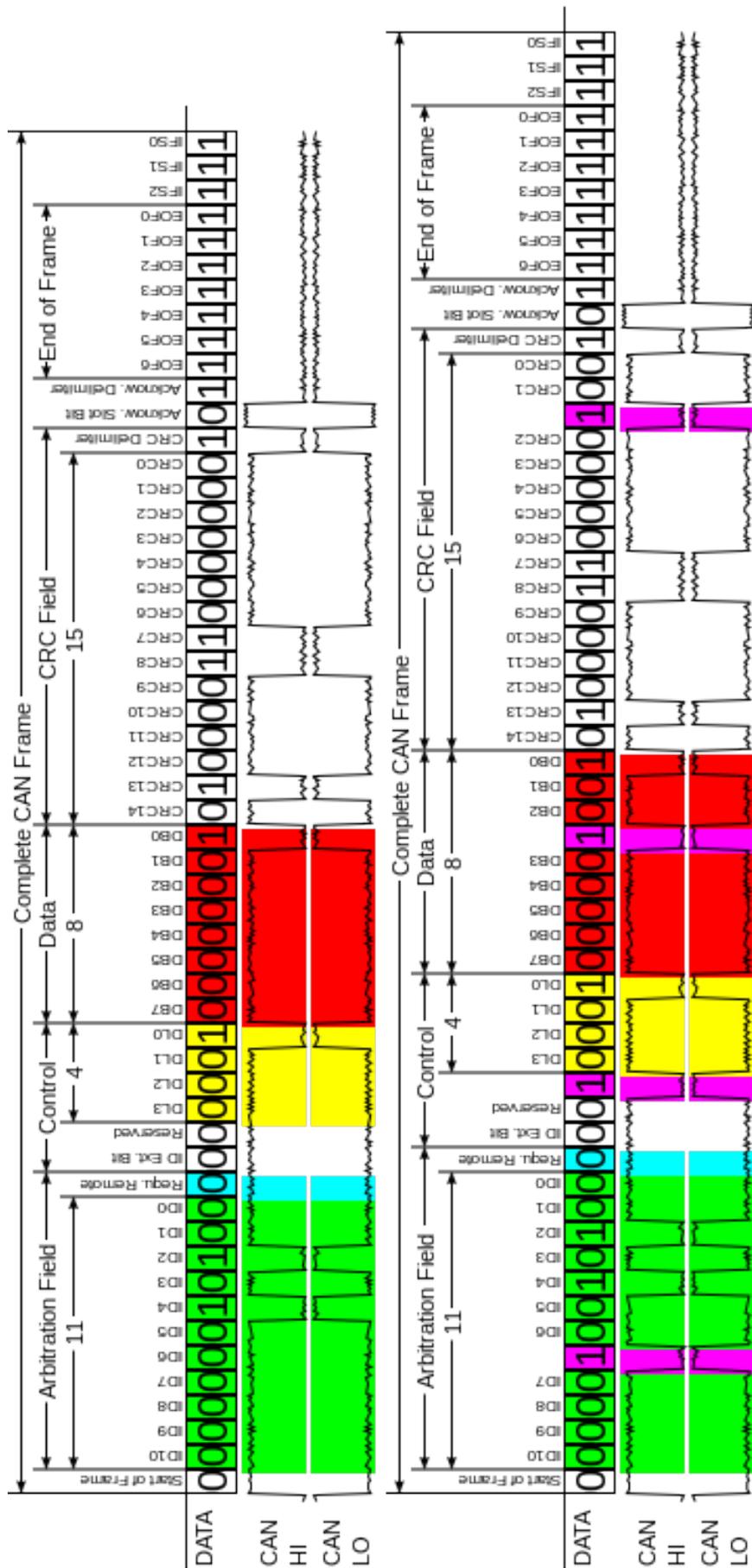


Рисунок 2.10 – Процедура Bit-stuffing

Номинальная скорость передачи информации в битах - число битов за секунду, передаваемое в отсутствии пересинхронизации идеальным передатчиком.

Номинальное время передачи бита – величина обратная номинальной скорости передачи информации в битах. Номинальное время передачи бита можно представить, как время, разделенное на отдельные не перекрывающиеся сегменты времени, представленное на рисунке 2.11. Это сегменты:

- сегмент синхронизации (SYNC\_SEG);
- сегмент времени распространения (PROP\_SEG);
- сегмент PHASE\_SEG1;
- сегмент PHASE\_SEG2.

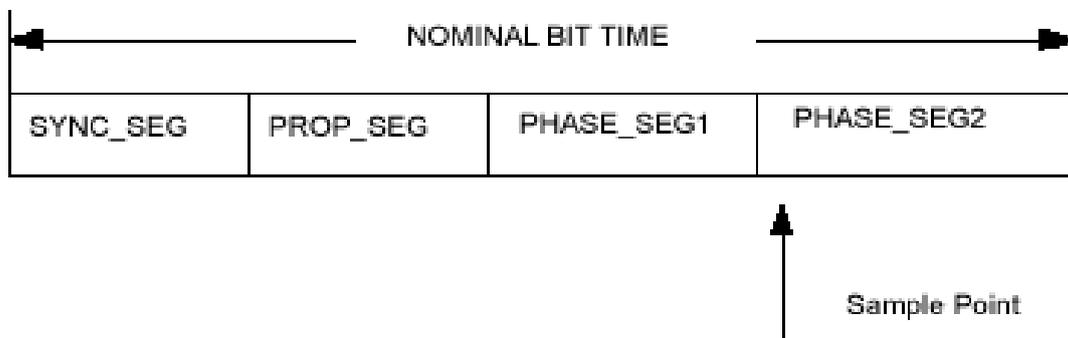


Рисунок 2.11 – Сегментное представление времени передачи бита

**SYNC SEG** – часть времени передачи бита, используемая для синхронизации различных узлов на шине. Ожидается, что фронт сигнала находится внутри этого сегмента.

**PROP SEG** – часть времени передачи бита используется, чтобы компенсировать физическую задержку времён внутри сети. Это удвоенная сумма времени распространения сигнала по шине, входной задержки компаратора, и выходной задержки формирователей.

**PHASE SEG1 и PHASE SEG2.** Эти сегменты используются, чтобы компенсировать ошибки смещения фазы сигнала. Эти сегменты могут быть удлинены или укорочены пересинхронизацией.

**Точка считывания (Sample point)**– точка времени, в которой уровень шины читается и интерпретируется, как величина соответствующего бита. Ее место - в конце PHASESEG1.

**Время обработки информации** представляет собой отрезок времени, начинающийся с точки считывания, зарезервированный для вычисления уровня бита.

**Квант времени Time Quanta (TQ)** – фиксированная единица времени, полученная из периода генератора. Существует программируемый делитель, оперирующий целочисленными величинами, изменяющимися, по крайней мере, от 1 до 32. Начиная с минимального кванта времени, квант времени может иметь длительность:

$$TQ = 1 / F_{osc},$$

$F_{osc}$ , МГц – частота генератора.

Так, например, контроллер 1886BE5 ЗАО «Миландр» имеет время  $TQ$  определяемое периодом генератора и программируемым прескалером BRP со значением от 1 до 64 в дополнение к  $F_{osc}$ :

$$TQ(\mu s) = \frac{4(BRP + 1)}{F_{osc}}.$$

*Тогда длительность отрезков времени*

- SYNC\_SEG - 1 квант времени.
- PROP\_SEG - программируется, 1,2, ..., 8 квантов.
- PHASE\_SEG1 - программируется, 1,2, ..., 8 квантов.
- PHASE\_SEG2 - максимум из PHASE\_SEG1 и времени обработки информации.

- время обработки информации - меньше или равно 2 квантам.

После аппаратной синхронизации внутреннее время передачи бита перезапускается с SYNC\_SEG. Таким образом, аппаратная синхронизация вынуждает фронт находиться внутри сегмента SYNC\_SEG.

В результате пересинхронизации PHASE\_SEG1 может быть удлинен, или PHASE\_SEG2 может быть сокращен. Синхронизация информации может быть получена из переходов от одного бита к другому. Максимальная длина между двумя переходами, которые могут использоваться для пересинхронизации - 29 времен передачи бита.

Фазовая ошибка фронта определяется позицией фронта относительно SYNC\_SEG, измеряется в квантах времени. Знак фазовой ошибки определяется следующим образом:

- $\varepsilon = 0$ , если фронт сигнала находится внутри SYNC\_SEG.
- $\varepsilon > 0$ , если фронт сигнала перед точкой считывания.
- $\varepsilon < 0$ , если фронт сигнала после точки считывания предыдущего бита.

Эффект пересинхронизации – также, как и от аппаратной синхронизации, когда величина фазовой ошибки фронта сигнала, которая вызывает пересинхронизацию – меньше или равна программируемой величине ширины перехода пересинхронизации.

Когда величина ошибки фазы больше чем ширина перехода пересинхронизации, и – если ошибка фазы положительна, то PHASE\_SEG1 удлиняется на время  $t_1 = t_b h$ , – и если ошибка фазы отрицательна, то PHASE\_SEG2 сокращается также на время  $t_1$ , где  $t_b$  – время передачи бита;  $h$  – ширину перехода пересинхронизации.

Синхронизация и пересинхронизация - две формы синхронизации в CANсетях. На них действуют следующие правила:

1. Позволяется только одна синхронизация внутри одного интервала передачи бита.

2. Для синхронизации будет использоваться фронт только, если величина, полученная в предыдущей точке считывания (предыдущая величина на шине) отличается от величины на шине сразу после фронта.

3. Синхронизация выполняется всякий раз, по фронту «1» → «0» в течение простоя шины.

4. Все другие фронты «1» → «0» (и фронты «0» → «1» в случае низких скоростей), выполняемые по правилам 1 и 2, будут использоваться для пересинхронизации.

CAN контроллер подключается к приемо-передатчику (трансиверу) через последовательный порт. Приемопередатчик присоединяется к шине CAN через свои терминалы CAN\_H и CAN\_L, которые обеспечивают дифференциальный сигнал между CAN\_H и CAN\_L дифференциальное напряжение - 2 В (рисунок 2.12).

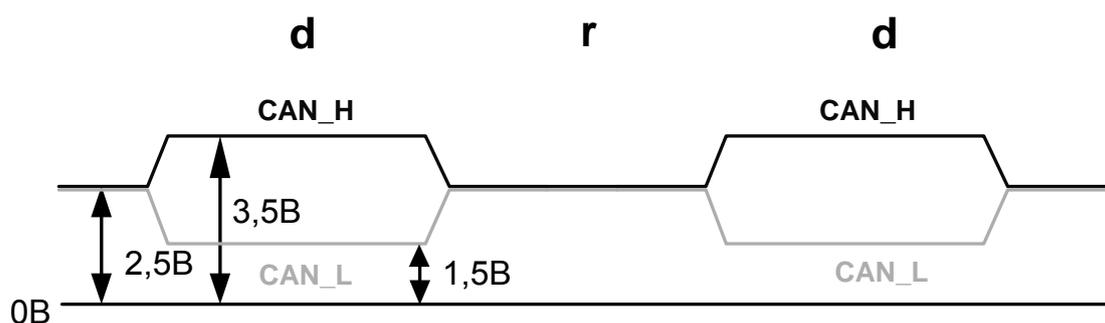


Рисунок 2.12 – Уровни сигналов при подключении CAN контроллера к приемо-передатчику

Помехоустойчивый приемо-передатчик был первоначально определен Daimler-Benz и другими производителями легковых автомобилей. Доминирующий бит представлен 3.6 В (CAN\_H) и 1.4 В

(CAN\_L) дифференциальное напряжение - 2.2 В; Рецессивный бит 5 В (CAN\_H) и 0 В (CAN\_L). дифференциальное напряжение - 5 В (рисунок 2.13). Помехоустойчивые приемо-передатчики первоначально предназначались для медленнодействующих вспомогательных сетей (до 125 kbit/s), в легковых автомобилях. Имеется возможность однопроводной передачи данных. Стандартизовано в ISO 11898-3.

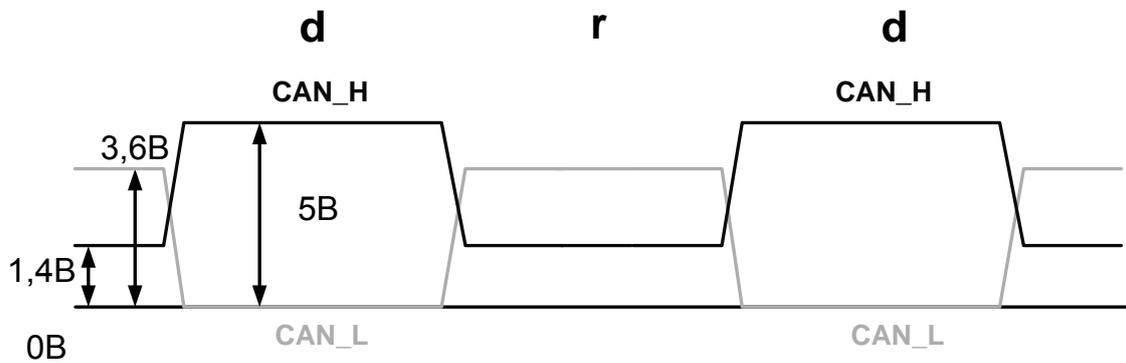


Рисунок 2.13 – Уровни сигналов при подключении CAN контроллера к помехоустойчивому приемо-передатчику

### 2.3.2. Передающая среда

В качестве передающей среды ISO 11898 определяет двухпроводную дифференциальную линию, экранированную или неэкранированную.

CAN шина должна быть закрыта концевыми резисторами 120 Ом на каждой оконечности шины для исключения отражения сигнала на конце шины (рисунок 2.14).

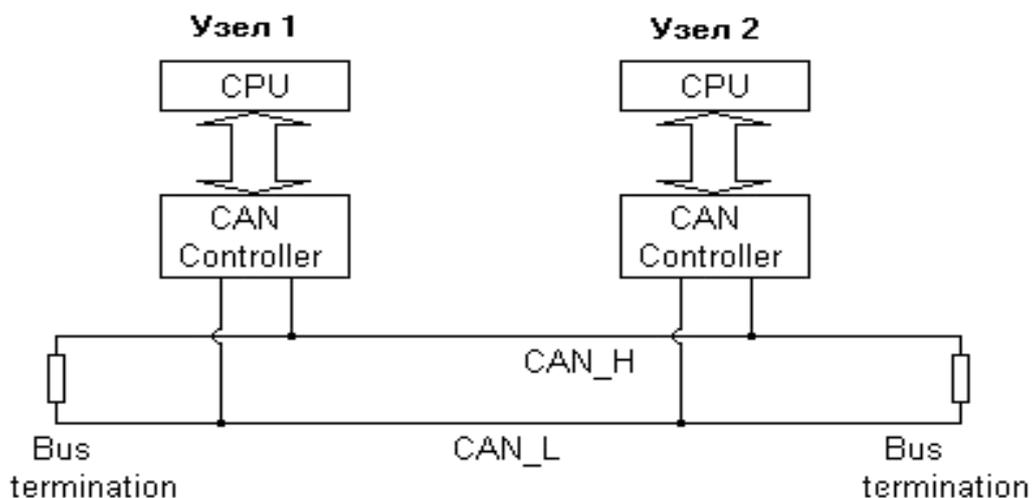


Рисунок 2.14 – Передающая среда CANсети

На автомобиле CAN-шина чаще всего представляет собой скрученную (витую) пару проводов (по 30 витков на один погонный метр) с разветвителями для подключения электронных блоков управления и конечными резисторами-терминаторами с номинальным сопротивлением 120 Ом на концах шины. Резисторы могут устанавливаться отдельно или быть встроенными в ЭБУ. Провод SHLD, как правило, отсутствует.

Соотношение между скоростью передачи и максимальной длиной кабеля представлены в таблице 2.4.

Таблица 2.4 – Соотношение между скоростью передачи и максимальной длиной кабеля

Скорость передачи	Максимальная длина сети
1000 Кбит/сек	40 метров
800 Кбит/сек	50 метров
500 Кбит/сек	100 метров
250 Кбит/сек	200 метров
125 Кбит/сек	500 метров
62,5 Кбит/сек	1000 метров
20 Кбит/сек	2500 метров
10 Кбит/сек	6 километров

Максимальная скорость сети CAN в соответствии с протоколом равна 1Мбит/с. При скорости в 1 Мбит/с максимальная длина кабеля равна примерно 40 метрам. Ограничение на длину кабеля связано с конечной скоростью света и механизмом побитового арбитража (во время арбитража все узлы сети должны получать текущий бит передачи одновременно, т.е. сигнал должен успеть распространиться по всему кабелю за единичный отсчет времени в сети). Теоретически возможная максимальная скорость CAN до 1,5 Мбит/сек. на 10 метрах (в стандарт не входит).

Стандарт ISO 11992 первоначально был разработан для соединения CAN - сетей грузовика и прицепа. Номинальная скорость передачи данных – 125 Кбит/с. Доминирующие (d) битовые величины: на линии CAN\_H = 18В, на линии CAN\_L = 9В (дифференциальное напряжение  $V_{diff} = 9 В$ ). Рецессивный (r) бит имеет также дифференциальное напряжение  $V_{diff} = 9 В$ , но линия CAN\_H = 9В, а линия

CAN\_L = 18В (рисунок 2.15). Нужно отметить, что уровни напряжения не являются жестко фиксированными, а зависят от напряжения бортовой сети ( $V_s$ ). На грузовых автомобилях номинальное напряжение бортовой электросети 24В, реальное напряжение при работающем двигателе  $V_s \approx 27В$ . Таким образом, рецессивный бит имеет напряжение:

$$V_{CAN\_H} = 1/3 V_s (\approx 9В);$$

$$V_{CAN\_L} = 2/3 V_s (\approx 18В).$$

Доминантный бит имеет напряжение:

$$V_{CAN\_H} = 2/3 V_s (\approx 18В);$$

$$V_{CAN\_L} = 1/3 V_s (\approx 9В).$$

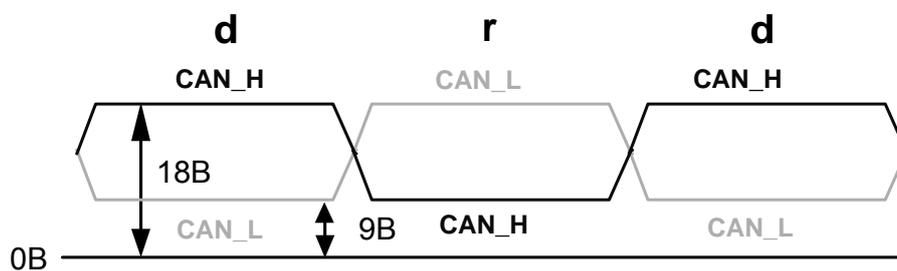


Рисунок 2.15 – Организация передающей среды в сети CANсогласно ISO 11992

Для реализации уровней напряжения необходимо использовать четырехпроводной соединитель, представленный на рисунке 2.16.

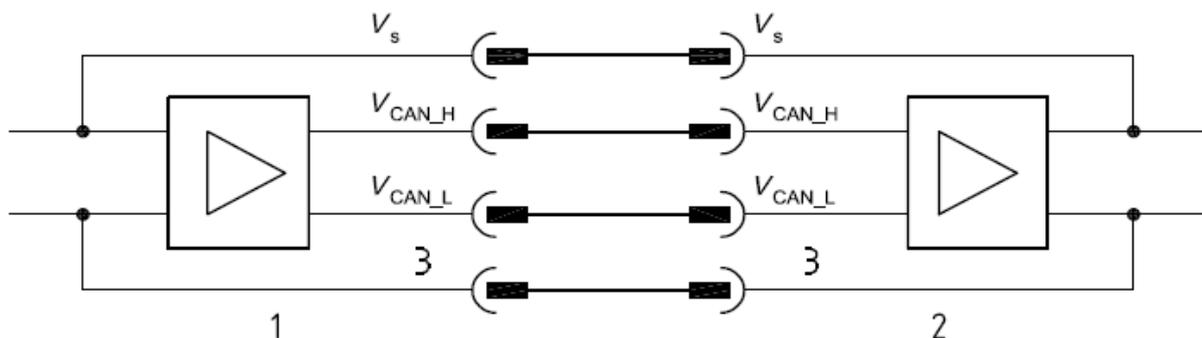


Рисунок 2.16 – Четырехпроводной соединитель:  
1 – интерфейс тягача; 2 – интерфейс прицепа; 3 – GND

Логическая организация сообщений соответствует стандарту J1939. Сеть реального времени CAN представляет собой одноранговую сеть с общей средой передачи данных, т.е. центральное устройство отсутствует. Все узлы сети одновременно принимают сигналы,

передаваемые по шине (широковещательный принцип). Невозможно послать сообщение какому-либо конкретному узлу. Не существует приоритета узла, но существует приоритет сообщения. Каждое сообщение содержит идентификатор, определяющий приоритет сообщения. Наивысший приоритет имеет сообщение, идентификатор которого имеет наибольшую последовательность доминантных бит. При этом один и тот же узел может генерировать сообщения как с высоким, так и с низким приоритетом. Все узлы сети принимают весь трафик, передаваемый по шине. Однако, CAN - контроллеры предоставляют аппаратную возможность фильтрации CAN - сообщений.

Логическая организация ISO 11783 в целом аналогична стандарту J1939, так же организовано распределение полей идентификатора, однако группы параметров разработаны для специфики сельхозтехники. Полное описание групп параметров приведено в тексте стандарта ISO 11783. Большое внимание в стандарте ISO 11783 уделено терминалу управления (VT), подробно описаны функции, типовые размеры экрана, отображаемые символы. При этом стандарт никак не определяет программное обеспечение и не содержит никаких исходных кодов, регламентируя только формат сообщений.

J1939 является основным CAN-протоколом для грузовиков и автобусов. Стандарт ISO 11783 - J1939 вносит дополнение для тракторов и других сельхозмашин.

Трактор по своему назначению, является носителем различных навесных и буксируемых механизмов. Для управления этими механизмами требуется установка консолей управления в кабине трактора. В отсутствие единого стандарта это приводит к тому, что для каждого механизма требуется своя консоль (рисунок 2.17).



а)



б)

Рисунок 2.17 – Кабина ТС

Для всего этого оборудования приходится прокладывать кабельные трассы с самыми разными разъемами.

В 2001 году производители сельскохозяйственных машин пришли к соглашению относительно введения стандартов, которые бы позволили использовать общие CAN интерфейсы связи в тракторах, приспособлениях и системах управления для сельскохозяйственных механизмов. Благодаря этому продукты различных производителей смогут обмениваться между собой данными, при этом меньше будет требоваться специальных устройств управления, предназначенных для конкретных приспособлений, терминалов и дисплеев. Один монитор в кабине трактора позволит осуществлять управление всеми приспособлениями – рулонным прессом-подборщиком, распылителем, разбрасывателем и т. п., и устанавливать связь с трактором и системой управления сельскохозяйственными работами.

Были разработаны очень четкие стандарты: ISO 11783, часто называемый ISOBUS. Стандарты довольно сложные и их применение во всей промышленности займет много времени. Европа занимает лидирующие позиции потому, что многие производители оборудования предлагают элементы управления приспособлениями с применением стандарта ISOBUS и решения по мониторингу.

### **Преимущества стандартизации ISOBUS**

- ♦ большая эффективность – один терминал для всех механизмов, поэтому не требуется покупать отдельную консоль управления для каждого приспособления;
- ♦ простота использования – достаточно изучить только один терминал.

Стандарт ISOBUS 11783 позволяет отображать на экране консоли систему управления навесным оборудованием. Примеры программных интерфейсов показаны на рисунке 2.18. Отображаемую на терминале консоли информацию по любому навесному оборудованию, независимо от его производителя, можно одинаково просматривать с помощью меню, кнопок выбора и активации. Это экономит время и деньги в связи с отсутствием необходимости устанавливать в кабине дополнительные терминалы.

Простота установки – благодаря стандартизированным заглушкам, кабелям и программному обеспечению функция автоматической настройки при подключении становится реальностью.

Чтобы воспользоваться этими преимуществами, следует учесть следующее:

- ← Совместимость трактора или комбайна со стандартом ISOBUS.
- ← Совместимость с ISOBUS навесных приспособлений
- ← Наличие монитора ISOBUS.

Возможно использование совместимых с ISOBUS приспособлений на несовместимом со стандартом ISOBUS тракторе, но при этом невозможно реализовать все преимущества использования этой системы.

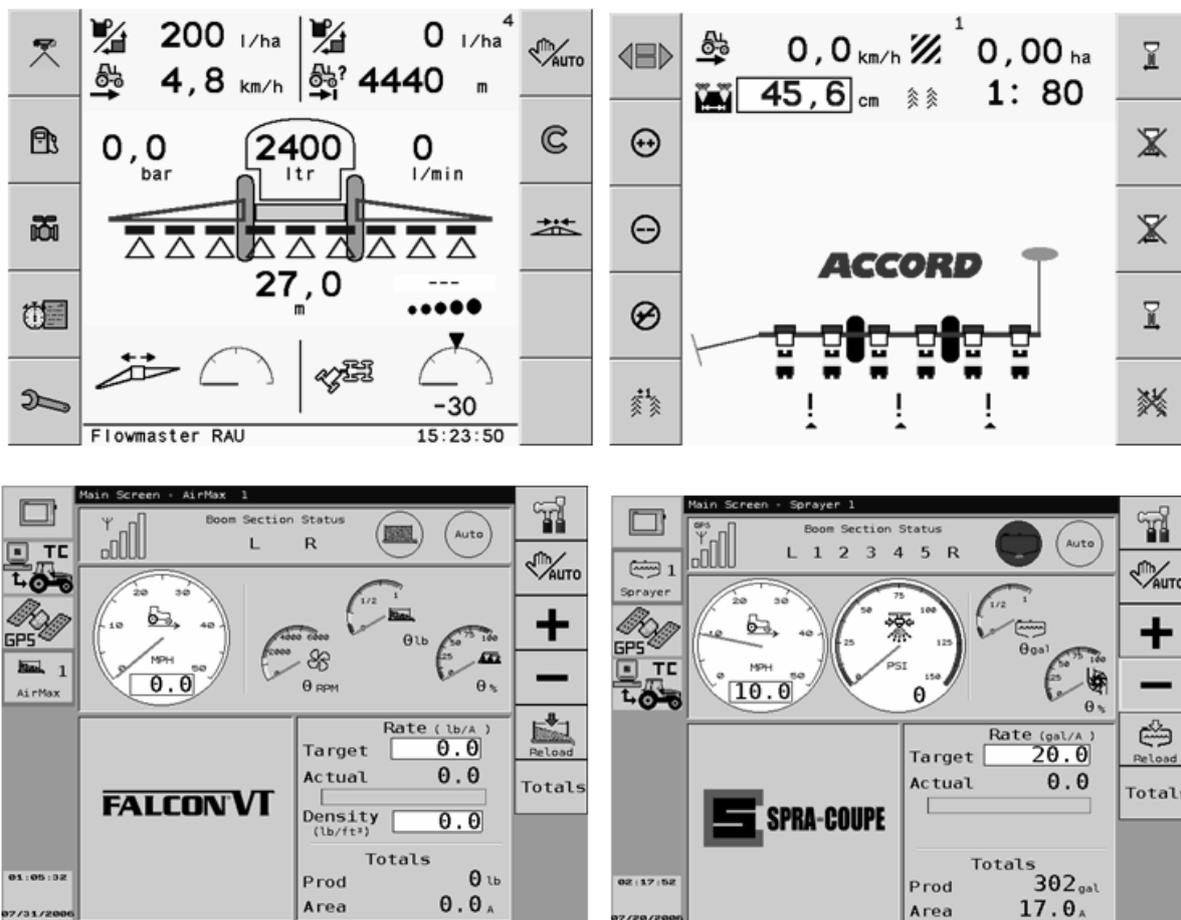


Рисунок 2.18 – Программный интерфейс

Стандарт ISO 11783 определяет максимальную длину сети 40 метров. Стандартный показатель скорости 250 Кбит/с. Допускает максимум 30 узлов в сети. Уровни сигнала соответствуют стандарту ISO 11898.

Большинство сообщений может транслироваться по сети ISO 11783. Поэтому данные передаются на сеть без направления ее в

определенное место назначения. Такая конфигурация позволяет любую функцию управления в ECU для использования данных без использования дополнительных сообщений запроса. ISO 11783 также определяет, что адрес назначения должен быть включен в идентификатор CAN сообщения при направлении сообщения на определенную функцию управления. Таким образом, формат сообщения для конкретного назначения отличается от широковещательного формат сообщения. Проприетарное сообщение также разрешено в ISO 11783, используя любой пункт назначения сообщения или форматы широковещательных сообщений.

ISO 11783-2 определяет физический уровень сети передачи данных. Среда передачи – витая четверка проводов соединенных с каждым электронным блоком управления (ЭБУ). К проводам CAN\_H и CAN\_L добавлены провода питания TBC\_PWR(+) и TBC\_RTN(-).

ISO 11783-2 также определяет разъемы, необходимые для подключения дополнительного оборудования к тракторам, дополнительных ЭБУ, и сервисных средств для сети. Также определяются источники питания, необходимые для работы сети и их соединений.

Типовая сеть управления и передачи данных ISO 11783 показана на рисунке 2.19.

ISO 11783 поддерживает два или более сегментов сети. Один сегмент идентифицируется как тракторная сеть. Этот сегмент предназначен для обеспечения управления и передачи данных для трансмиссии и шасси трактора или основного силового агрегата в системе. Второй сегмент определяется как сеть инструментов, которая обеспечивает контроль и передачу данных между инструментами, а также между инструментами и трактором или основным силовым агрегатом в системе. ЭБУ трактора необходим для объединения сети трактора и сетью инструмента. Рисунок 2.20 иллюстрирует два сегмента сети, соединенные с ЭБУ трактора.

Сеть трактора обеспечивает управление и передачу данных между функциями управления на приводе и шасси трактора. Данный сегмент сети контролируется производителем тракторов. Рекомендуется, что физический уровень сети трактора исполняет с ISO 11783-2.

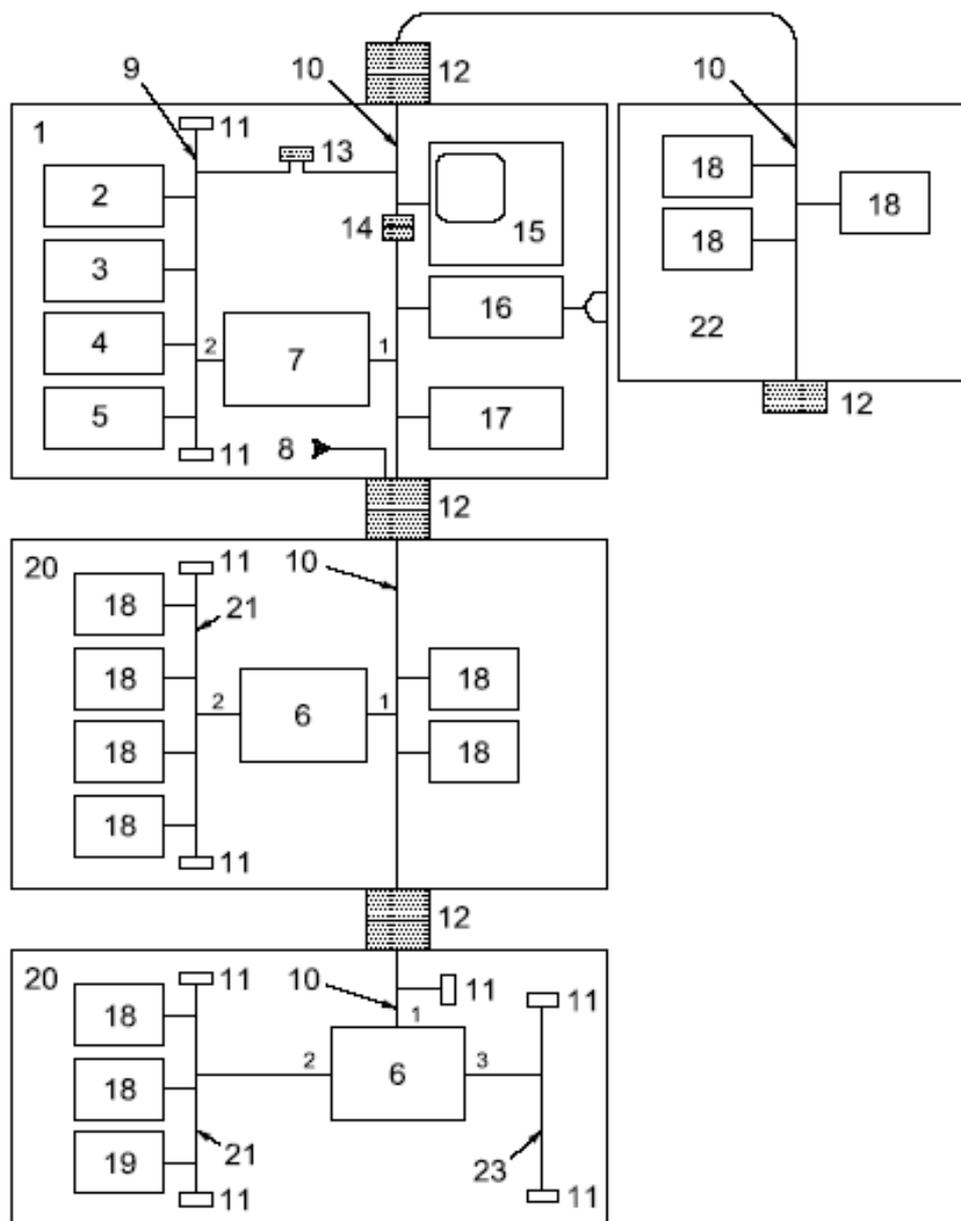


Рисунок 2.19 – Типовая структура физического сетевого соединения трактора и дополнительного оборудования:

1 – трактор; 2 – двигатель; 3 – трансмиссия; 4 – тормоза; 5 – контроллер сцепки; 6 – блок сетевого соединения; 7 – ЭБУ трактора; 8 – входное питание; 9 – сеть трактора; 10 – сеть инструмента; 11 – терминатор (оконечная нагрузка); 12 – разъем отвода шины инструмента; 13 – диагностический разъем; 14 – разъем расширения шины; 15 – виртуальный терминал; 16 – шлюз управляющего компьютера; 17 – задающий контроллер; 18 – ЭБУ; 19 – контроллер освещения; 20 – расположенное в задней части или буксируемое оборудование; 21 – ISO 11783 или другая сеть; 22 – переднее или боковое оборудование; 23 – сети другого стандарта

ПРИМЕЧАНИЕ. Меньшие номера указывают на части на взаимосвязанных устройствах в ЭБУ трактора.

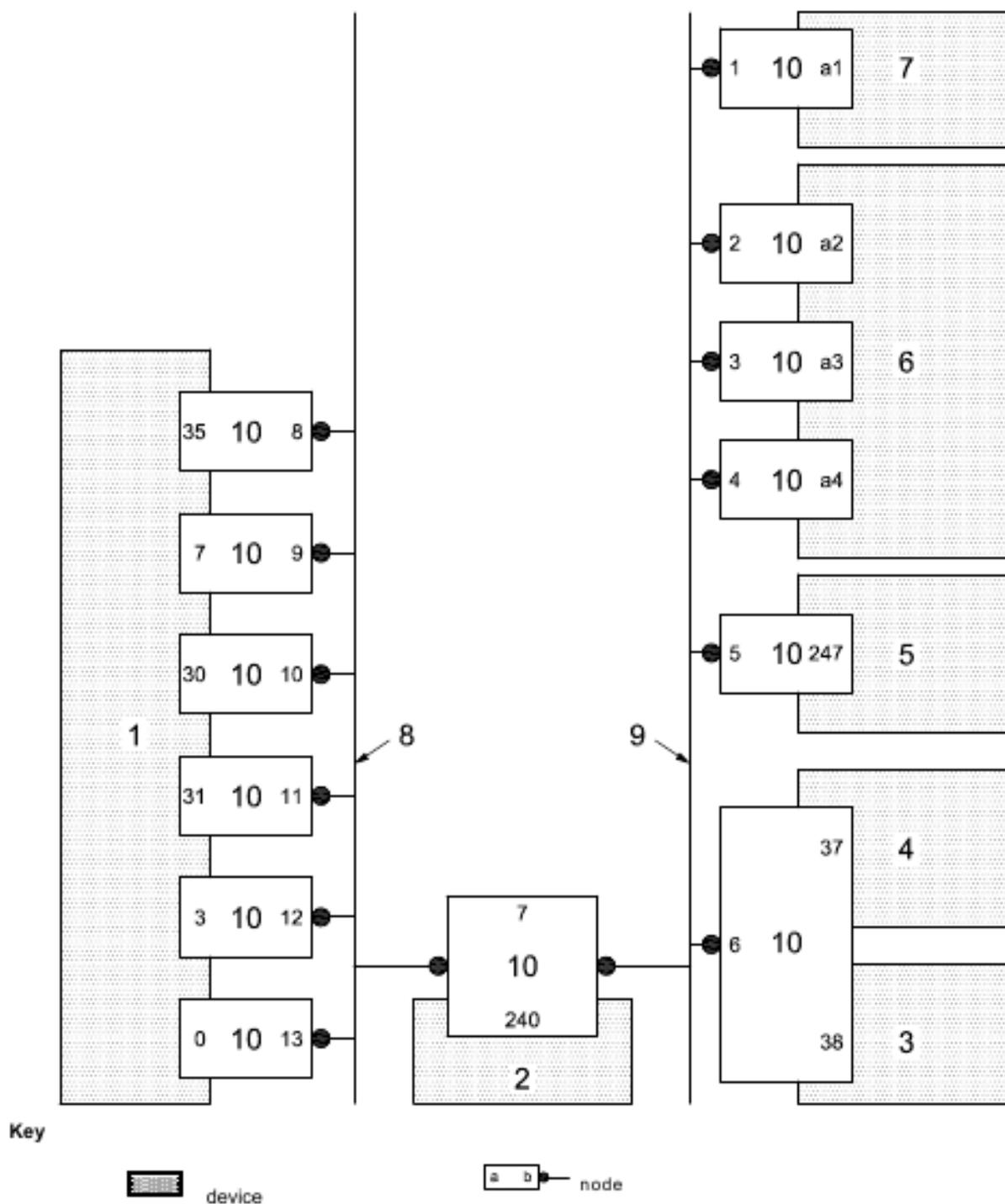


Рисунок 2.20 – Типовая топология сети ISO 11783

1– трактор; 2 – ЭБУ трактора; 3 – виртуальный терминал; 4 – шлюз компьютера управления; 5 – задающий контроллер; 6 – инструмент 1; 7 – инструмент 2; 8 – сеть трактора; 9 – сеть инструмента; 10 – ЭБУ;  $a$  – адрес функции управления  $b$  – ЭБУ, содержащий функции управления

Примечание 1:  $a_1, a_2, a_3, a_4$  – автоконфигурируемые адреса.

Тракторная сеть показана относительно сети агрегата на тракторе. И сеть трактора, и сеть инструмента могут быть встроенный бортовой трактор. Предполагается, что функции управления не будут подключены к тракторной сети, если только не получено согласие производителя трактора. Сеть трактора может использовать сообщения приложения определенными в ISO 11783-8 и ISO 11783-7 а также собственные сообщения, указанного в ISO 11783-3.

Диапазон напряжения в шине определяется максимальным и минимальным допустимым напряжением верхнего CAN\_Hand и нижнего уровня CAN\_L, измеренным по отношению к определенной земле каждого ЭБУ, для которых правильная работа гарантируется для всех ЭБУ, подключенных к сигнальным линиям шины.

Сигнальные линии шины сегмента электрически завершаются на каждом конце оконечной схемой смещения (ТВС). Этот элемент располагается внешне от ЭБУ, для того чтобы обеспечить смещение и закрытие шины когда ЭБУ будет отключен (рисунок 2.21).

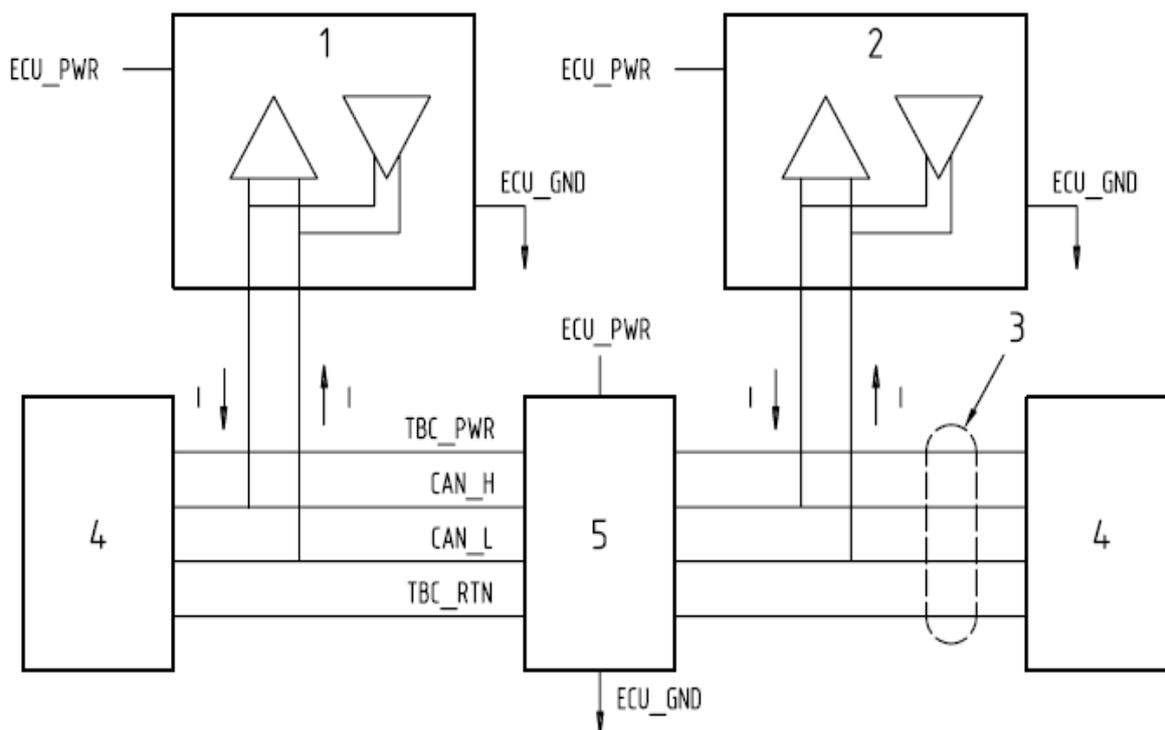


Рисунок 2.21 – Функциональная схема физического уровня  
 1 – ЭБУ №1; 2 – ЭБУ №n; 3 – витая четверка проводов; 4 – оконечная схема смещения (ТВС); 5 – питание для TBC\_PWR и TBC\_RTN

Чтобы избежать отражения волн в кабеле, топология проводки сегмента шины должна иметь, насколько это возможно, прямолинейную организацию. На практике может потребоваться подключение коротких заглушек к магистральному кабелю, как показано на рисунке 2.22.

Для минимизации стоячих волн узлы не должны быть равномерно расположены на сегменте шины, а длина заглушки не должна быть одинаковой. Размерные параметры этой топологии приведены в таблице 2.5.

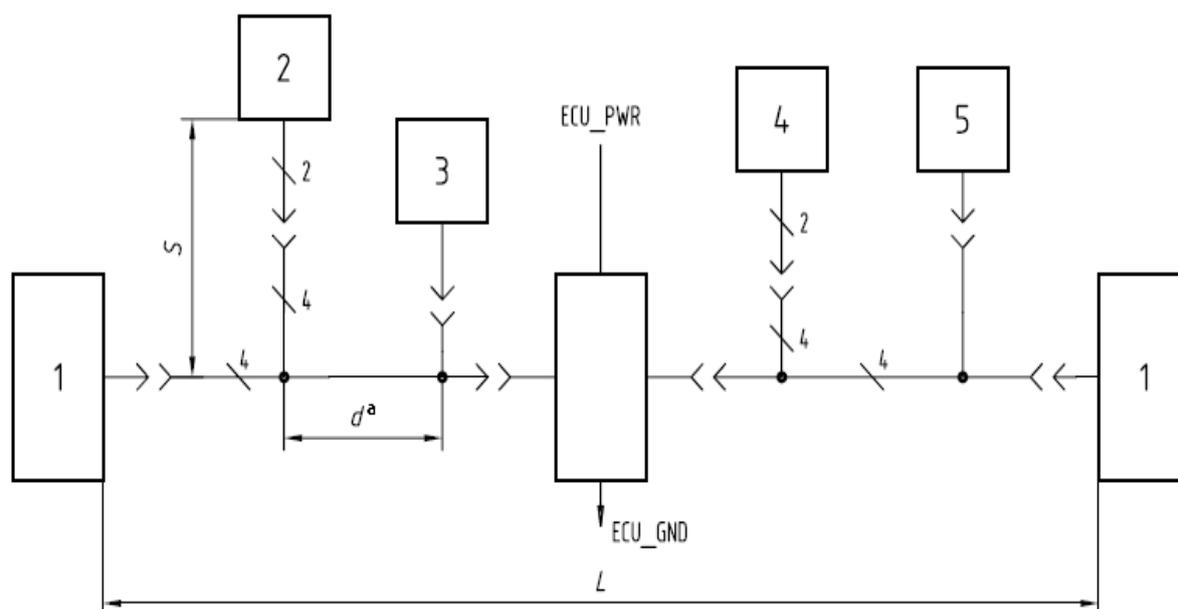


Рисунок 2.22 – Топология проводного соединения:  
 1 – ТВС; 2 – ECU 1; 3 – ECU 2; 4 – ECU n-1; 5 – ECU n;  
 дистанция  $d$  должна быть произвольной, но не менее 0,1 м.

Таблица 2.5 – Размерные параметры топологии

Параметр	Символ	Min.	Max.	Ед. изм.	Условия
Длина шины	L	0	40	м	Без учёта заглушек
Длина заглушки	S	0	1	м	-
Расстояние между узлами	d	0.1	40	м	-

Подключение дополнительных устройств, необходимых для работы в полевых условиях, используется разъем сигнальных линий шины

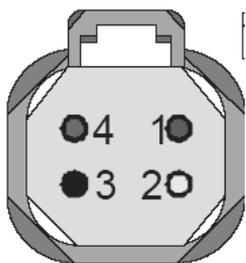


Рисунок 2.23 – Разъем ISO 11783

трактора, который располагается внутри трактора. Он находится в кабине с правой стороны сиденья оператора, впереди от внешних органов управления оборудованием.

Розетка разъема расширителя шины трактора показана на рисунке 2.23, а назначение выводов разъема в таблице 2.6

Таблица 2.6. Назначение выводов

Номер вывода	Назначение
Вывод 1 – красный	TBC_PWR(+Terminator)
Вывод 2 – желтый	CAN_H (Data H)
Вывод 3 – черный	TBC_RTN(ground Terminator)
Вывод 4 – зеленый	CAN_L (Data L)

На задней части трактора установлена розетка, для подключения дополнительного навесного или буксируемого оборудования, в соответствии с ISO 1724. Розетка закрывается крышкой, защищающей разъем от пыли влаги, когда буксируемое оборудование не подключено.

Дополнительный разъем может быть установлен на передней части трактора рядом с передними гидравлическими выходами, когда навесное оборудование размещается впереди трактора. Этот разъем должен быть идентичен заднему разьему.

Вилка и розетка разъема расширителя инструментальной шины показана на рисунке 2.24, а назначение выводов разъема в таблице 2.7

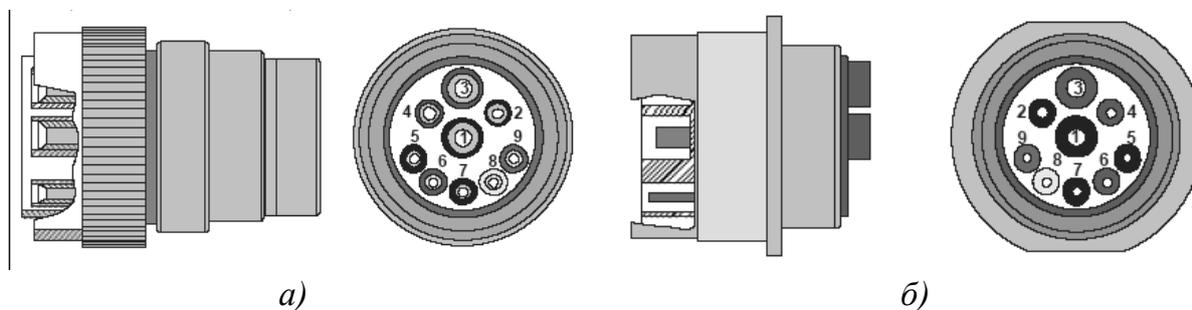


Рисунок 2.24: а) вилка; б) приёмник

Таблица 2.7 – Назначение выводов разъема

№	Обозначение	Цвет	Описание
1	GND	black	Связан с корпусным заземлением на тракторе и оборудовании. Все основные автоматические нагрузки, включая фары, двигатели, и т.д. будут использовать эту обратную связь. Связь с корпусным заземлением гарантирует, что нет никакого различия в разности потенциалов или статическом заряде между оборудованием и трактором
2	ECU_GND	black	Этот вывод должен быть электрически изолирован от корпусного заземления оборудования
3	PWR	red	Питание для всех фар, двигателей, и т.д. которые обычно требуют значительной мощности и имеют тенденцию генерировать переходные процессы на линии питания
4	ECU_PWR	red	Положительное питание для ЭБУ, установленных на оборудовании
5	TBC_DIS	black	Существует только в пределах соединителей (т.е. не для внешних связей), чтобы управлять реле для автоматического подключения/отключения. Подключается к выводу 4 на штепселе оборудования
6	TBC_PWR	red	Напряжение для ТВС. Это напряжение не используется для чего-либо кроме ТВС
7	TBC_RTN	black	Обеспечивает обратную связь для ТВС. Это связь не используется для чего-либо кроме ТВС
8	CAN_H	yellow	Линия передачи данных, которая подтянута к высокому уровню напряжения.
9	CAN_L	green	Линия передачи данных, которая протянута к низкому уровню напряжения.

На рисунке 2.25 приведен пример сетевых соединений, иллюстрирующий использование разъемов и соединения узла к различным ЭБУ, который использует промышленную CAN-сеть, соответствующую стандарту ISO 11783.

На рисунке приняты следующие обозначения:

*a* – трактор; *b* – оборудование 1; *c* – оборудование 2;  
*1* – Электронный блок управления (ЭБУ) 1, (ISO 11783-2); *2* – ЭБУ 2 (ISO 11783-2); *3* – ЭБУ n (ISO 11783-2); *4* – ЭБУ n -2 (ISO 11783-2);  
*5* – ЭБУ n -1 (ISO 11783-2); *6* – ЭБУ z (ISO 11783-2); *7* – ЭБУ трактора; *8* – ЭБУ y (SAEJ1939-11); *9* – ЭБУ x (ISO 11783-2); *10* – батарея;  
*11* – ТВС; *12* – соединительный узел (стык); *13* – дополнительная заглушка ЭБУ; *14* – четырехпроводная линия; *15* – трехпроводная ли-

ния; 16 – шина трактора; 17 – подключение питания к сети; 18 – инструментальная шина; 19 – диагностический разъем; 20 – опциональная автоматическая ТВС; 21 – контроллер шины; 22 – автоматическая ТВС с подключением сетевого питания; 23 – контроллер отвода шины; 24 – автоматическая ТВС.

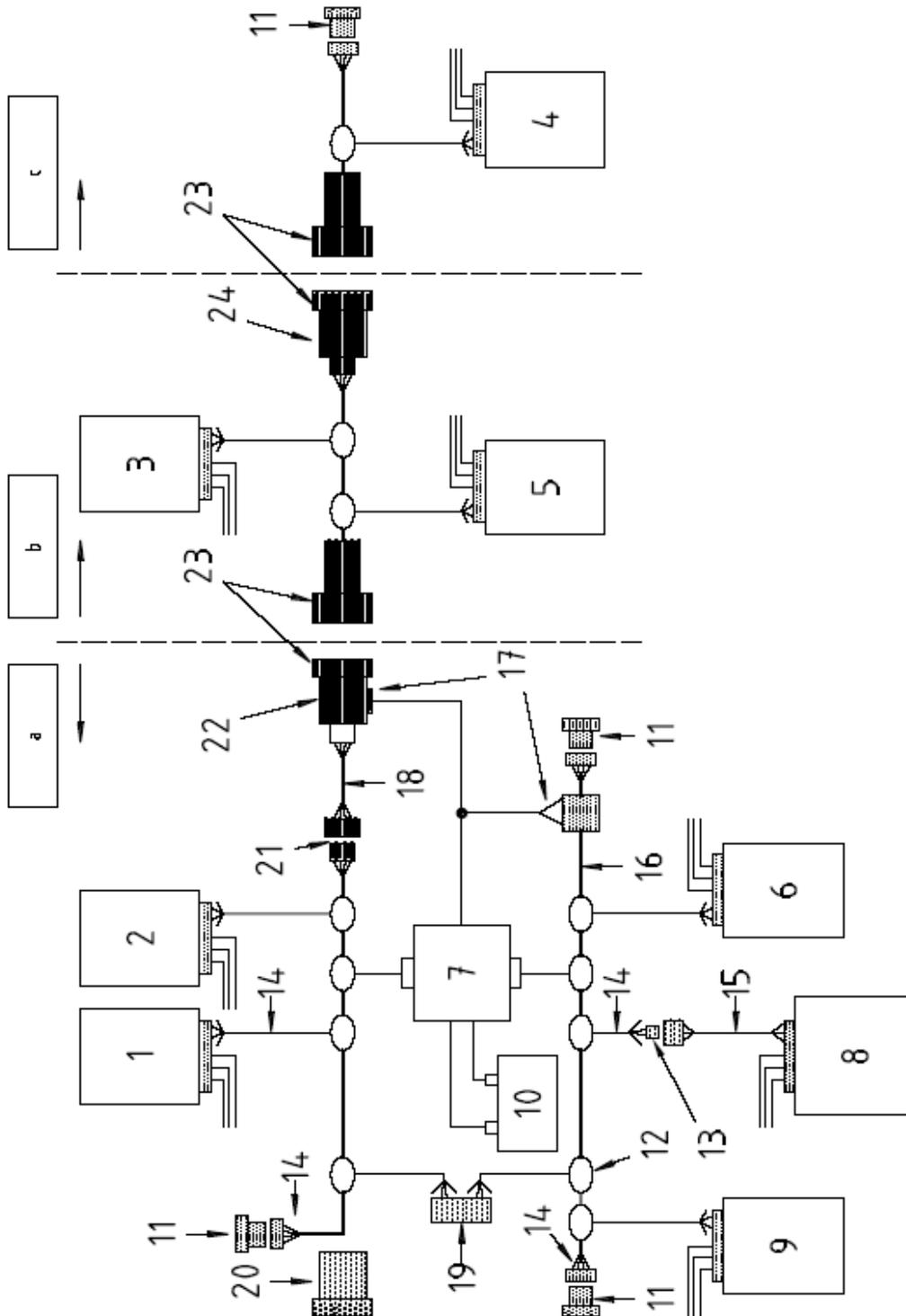


Рисунок 2.25 – Пример сетевых подключений

### 2.3.3. Формат кадра в сети CAN

Кадр данных состоит из 7 различных полей: «Начало кадра», «поле арбитража», «поле контроля», «поле данных», «поле CRC», «поле подтверждения», «конец кадра» (рисунок 2.26).

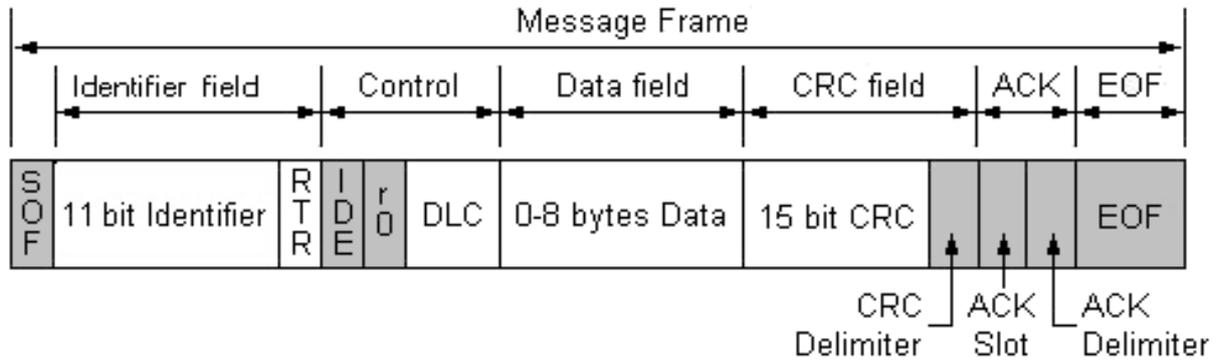


Рисунок 2.26 – Формат кадра в сети CAN

**Начало кадра (StartofFrame)** отмечает начало кадра данных или кадра удаленного запроса данных. Состоит из бита с лог. «0». Узлу разрешено начинать передачу только при свободной шине. Все узлы должны быть синхронизированы по началу фронта, вызванного полем «начало кадра» узла, начавшего работу первым.

**Поле арбитража (Arbitration Field)** состоит из идентификатора и RTR-бита.

**Идентификатор (Identifier, ID)** имеет длину 11 бит. Эти биты должны быть переданы от ID10. Самый старший бит ID0. 7 старших битов не должны быть все битами с лог «1». Идентификатор определяет приоритет сообщения (пример):

Узлы передают в шину сообщения с соответствующими идентификаторами, доминантный уровень – лог. «0»

- Узел 1:           01101000110
- Узел 2:           01101001X
- Узел 3:           0111X

X свидетельствует о прекращении передачи.

**RTR-бит (RTR - bit) Бит запроса передачи.** В кадре данных RTR-бит –«0». Внутри кадра удаленного запроса данных –«1».

**Поле контроля (CONTROL FIELD)** включает 6 бит. Это - код длины данных (4 бита) и 2 бита зарезервированные под будущие расширения. Зарезервированные биты должны быть «0».

**Код длины данных (DLC)** показывает количество байт в поле данных. Код длины данных имеет размер 4 бита и передается внутри контрольного поля. Допустимые значения: 0.....8. Другие значения использоваться не могут.

**Поле данных (DATA FIELD)** включает данные, передаваемые внутри кадра данных. Оно может содержать от 0 до 8 байт, каждый из которых содержит 8 бит.

**CRC поле (CRC FIELD)** содержит CRC - последовательность, сопровождаемую разделителем.

Для вычисления CRC полинома, полином, коэффициенты которого задаются потоком, состоящим из значений бит - полей: «начало кадра», «поле арбитража», «поле контроля», «поле данных» должен быть разделён на полином следующего вида:

$$x^{15}+x^{14}+x^{10}+x^8+x^7+x^4+x^3+1.$$

Остаток этого полиномиального деления есть CRC-последовательность, передаваемая по шине.

CRC-последовательность сопровождается CRC-разделителем, который всегда равен лог. «1».

**Поле подтверждения (ACK FIELD)** имеет длину в 2 бита. Содержит область подтверждения (1 бит) и разделитель подтверждения (1 бит). В поле подтверждения передающий узел посылает два бита с лог. «1». Приемник, получивший правильное сообщение, информирует об этом передатчик, посылая бит с лог. «0» (т.е. перезаписывая бит в области подтверждения с лог. «1» на бит с лог. «0»).

**Область подтверждения (ACK Slot).** Все узлы, получившие соответствующую CRC-последовательность, сообщают об этом внутри области подтверждения перезаписью бита с лог. «1» на бит с лог. «0».

**Разделитель подтверждения (ACK Delimiter).** Второй бит области подтверждения должен быть - лог. «1». Следовательно, область подтверждения окружена битами с лог. «1» (CRC-разделитель и разделитель подтверждения).

**Конец кадра (END OF FRAME).** Каждый кадр данных и кадр удаленного запроса данных разграничены последовательностью флагов, состоящей из семи битов с лог. «1».

На рисунке 2.27 представлены форматы кадров данных: а) стандартный формат кадра; б) расширенный формат кадра.

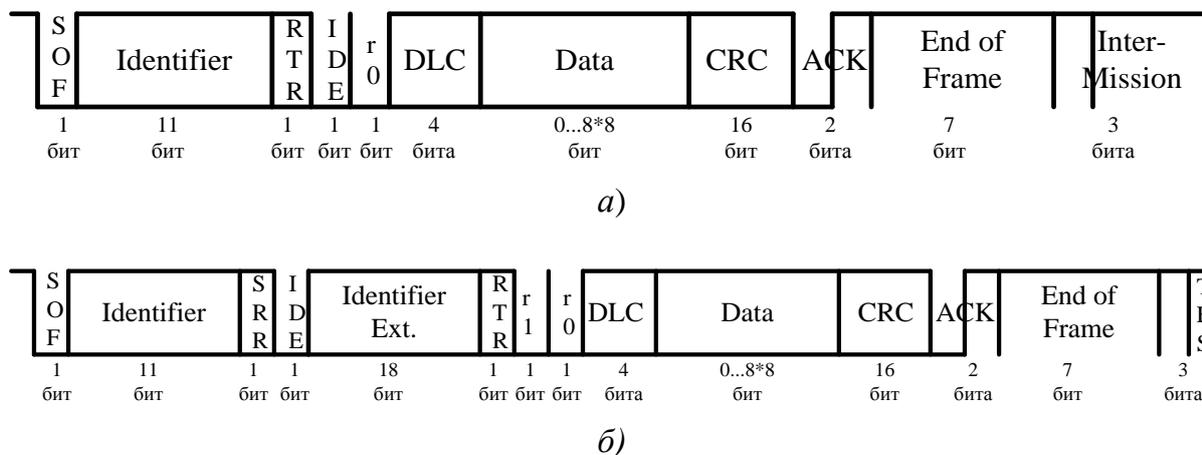


Рисунок 2.27 – Форматы кадра данных

В дополнение к описанным выше полям, кадр расширенного формата содержит следующие поля: подмена запроса на передачу (SRR) (должен быть рецессивным); бит расширения идентификатора (IDE) (должен быть рецессивным – определяет длину идентификатора); зарезервированные биты (r1 и r0)

### 2.3.4. Аппаратная реализация сети CAN

#### 2.3.4.1. CAN разъемы

Не существует стандарта для CAN разъемов. Обычно это определяется NLP протоколом. Ниже, на рисунках 2.28 – 2.31 представлены наиболее распространенные соединительные разъемы.

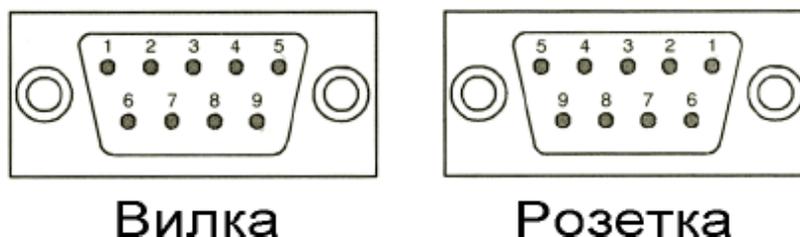


Рисунок 2.28 – 9-pinDSUB:  
 2 - CAN\_L; 3 - CAN\_GND; 5 - CAN\_SHLD;  
 6 – GND; 7 - CAN\_H; 9 - CAN\_V+

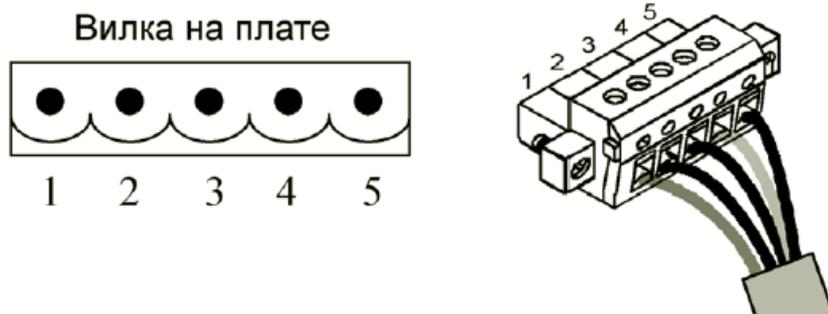


Рисунок 2.29 –5-pin Mini-C и/или Micro-C, используется DeviceNet и SDS: 1 -V-; 2 -CAN\_L; 3 -Drain; 4 -CAN\_H;5 -V+

Фирма DEUTSCH выпускает CAN шину из специального 3-х жильного кабеля (CAN-H, CAN-L и сигнальная «земля») с диаметром наружной оболочки от 7 до 12 мм и специальными разветвителями, и конечными устройствами – терминаторами, которые служат для согласования волнового сопротивления при передаче сообщений в шине и для подавления помех.

В таблице 2.8 приведено назначение выводов разъема CAN.

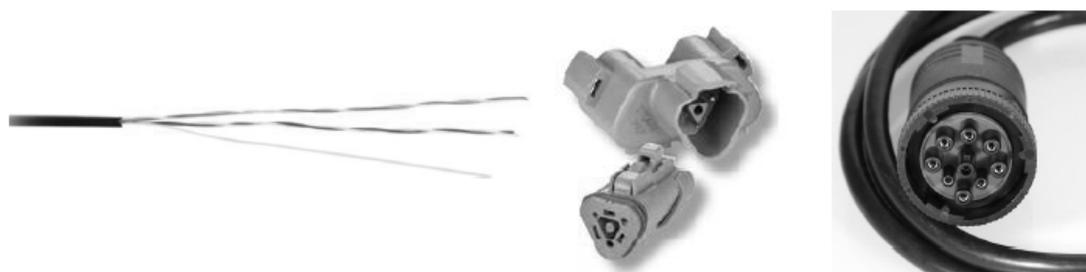


Рисунок 2.30 – Элементы шины CAN фирмы DEUTSCH

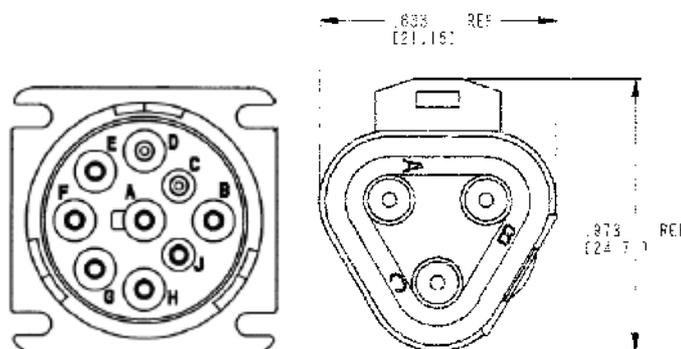


Рисунок 2.31 – Разъемы элементов шины CAN фирмы DEUTSCH

Таблица 2.8 – Назначение выводов разъема CAN

A	GROUND	
B	+12V	
C	J1939CAN H	
D	J1939CAN L	
E	Shield	not connected
F	J 1708 +	not connected
G	J 1708 -	not connected
H	CAN2 H	NOT USED
J	CAN2 L	NOT USED

#### 2.3.4.2. CAN устройства

В настоящее время доступны различные устройства с CAN-интерфейсом, которые помимо передачи данных из одной точки в другую позволяют реализовать синхронизацию процессов и обслуживание по приоритетам.

Более ранние реализации CAN-контроллеров используют кадры с 11-разрядным идентификатором и возможностью адресации до 2048 сообщений и соответствуют спецификации CAN V. 2.0A. Такие контроллеры носят название Basic CAN и характеризуются сильной загруженностью центрального процессора (ЦПУ), так как каждое входящее сообщение запоминается в памяти и ЦПУ решает, нужны ему данные сообщения или нет.

Контроллеры Basic CAN содержат один передающий буфер и один или два приемных буфера сообщений. Чтобы послать или получить сообщение, требуется задействовать ЦПУ через прерывания «сообщение-послано» и «сообщение-получено».

В результате проверки каждого входящего сообщения загрузка ЦПУ очень велика, что ограничивает реальную скорость обмена по сети.

По этой причине такие контроллеры используются в сетях CAN с низкой скоростью обмена и/или малым количеством сообщений.

Большинство выпускаемых сегодня CAN-контроллеров используют расширенные кадры сообщений с идентификатором длиной 29 разрядов, что позволяет адресовать до 536 млн. сообщений. Такие

контроллеры соответствуют спецификации CAN V. 2.0B (active) и называются контроллеры Full-CAN.

В них предусмотрен буфер для нескольких сообщений, причем каждое сообщение имеет свою маску, и фильтрация осуществляется по соответствию идентификатора маске.

В случае Full-CAN ЦПУ максимально разгружено, поскольку не обрабатывает ненужные сообщения.

При приеме сообщения с идентификатором, соответствующим маске, оно запоминается в специальной зоне двухпортового ОЗУ, и работа ЦПУ прерывается. Full-CAN имеет также специальный тип сообщения, которое означает: «у кого бы ни находилась эта информация, пожалуйста, пошлите ее сейчас же». Контроллер Full-CAN автоматически прослушивает все сообщения и посылает запрошенную информацию.

### **Входные фильтры**

Входные фильтры представляют собой решета, или идентификационные экраны. Любое сообщение, которое проходит через входные фильтры, должно быть обработано процессором обслуживания CAN-контроллера. Чем большее количество единиц может быть отфильтровано, тем меньше нагрузка на процессор. Микросхемы, поддерживающие CAN-протокол, могут иметь одиночный фильтр или многократные фильтры, в зависимости от конкретной реализации.

Существуют следующие два типа входных фильтров:

- фиксированные – фильтры, которые требуют, чтобы биты соответствовали точно один к одному (one-for-one).
- Mask-and-Match (маскируемые) – фильтры, которые применяют маску к полю идентификатора, прежде чем он сравнивается с приемным регистром кода.

До недавнего времени был широко распространен Basic CAN с 11-разрядным идентификатором. Этот протокол допускает простую связь между микроконтроллерами и периферийными устройствами при скорости обмена вплоть до 250 Кбит/с.

Однако при стремительном удешевлении CAN-контроллеров использование Full-CAN стало оправданным и для связи с медленными устройствами.

Если требуется высокоскоростной (до 1 Мбит/с) обмен данными, то следует использовать только Full-CAN.

CAN-устройства могут иметь разную степень интеграции в корпусе микросхем. Однако наиболее распространены устройства, где главный контроллер (ЦПУ) интегрирован с CAN-контроллером в одну микросхему, а трансивер подключается отдельно, что позволяет использовать разные типы трансиверов под конкретные условия, например, если это необходимо, использовать трансиверы с внутренними или внешними гальваническими развязками и/или оптронными развязками. На рисунке 2.32 приведена функциональная схема аппаратной организации CAN сети

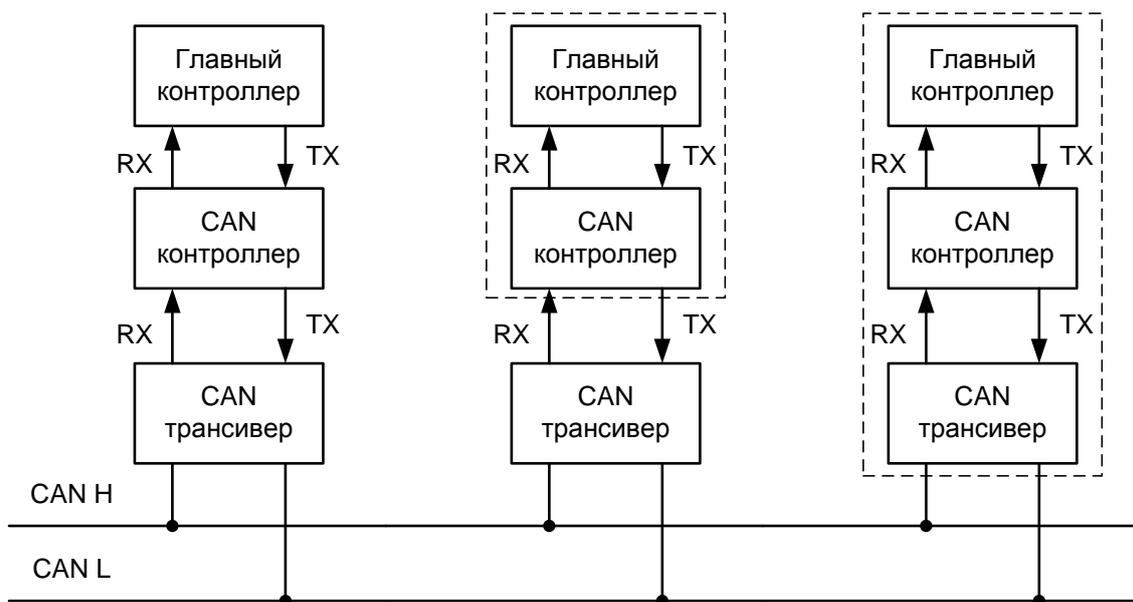


Рисунок 2.32 – Аппаратная организация CAN сети

Непосредственная связь шин CAN силового агрегата и системы «Комфорт» невозможна ввиду различного уровня напряжений и различных нагрузочных сопротивлений. При этом имеют место еще различные скорости передачи данных по шинам, что исключает обработку разнотипных сигналов в одном устройстве. Поэтому необходимо применение преобразователя для связи между шинами. Таким преобразователем является так называемый межсетевой интерфейс (Gateway). На автомобиле такой интерфейс может быть встроен в комбинацию приборов или в блок управления бортовой сетью, а также выделен в отдельный прибор межсетевой связи.

Так как в интерфейс поступает вся передаваемая через шины информация, его используют также для вывода диагностической информации.

### **2.3.5. Протоколы верхнего уровня HLP**

#### **2.3.5.1. J1939**

Протокол J1939 первоначально был разработан для тяжелых грузовиков и автобусов. В настоящее время применяется и на легковых автомобилях. Полный комплект спецификации J1939 можно приобрести в SAE, ниже приведен перечень основных документов:

- ◆ J1939/11 – Физический уровень, 250 кбит/с, на основе экранированной витой пары.
- ◆ J1939/13 – Диагностические разъемы, сопряжение с внешним диагностическим оборудованием.
- ◆ J1939/15 – Физический уровень, 250 кбит/с, не экранированная витая пара.
- ◆ J1939/21 – Канальный уровень.
- ◆ J1939/31 – Сетевой уровень.
- ◆ J1939/71 – Прикладной уровень транспортного средства.
- ◆ J1939/73 – Прикладной уровень диагностики.
- ◆ J1939/81 – Протоколы управление сетью.

Физический уровень протокола J1939 повторяет основные положения стандарта ISO 11898. В качестве эталонного трансивера рекомендуется PCA82C250. Приведенная на рисунке 2.33 трехпроводная схема используется в основном на тяжелых грузовиках. Возможно двухпроводное подключение без провода SHLD. Применяется для легковых автомобилей.

Стандарт определяет максимальную длину сети 40 метров (~120 фут.). Стандартный показатель скорости 250 Кбит/с. Допускает максимум 30 узлов в сети (рисунок 2.34).

Поддерживает адресованные и широковещательные сообщения. Поддерживает передачу данных длиной до 1785 байт.

J1939 основан на стандартном CAN кадре с 29-битовым идентификатором. Формат этого кадра (рисунок 2.35) определен стандартом ISO 11898-1. А распределение отдельных бит идентификатора и правила заполнения поля данных относятся к ведению собственно стандарта J1939. Именно он определяет то, каким образом задается трехбитовый приоритет сообщения (P), выбор страницы данных (DP), адреса источника (SA) и получателя (DA) кадра и другие параметры CAN сообщения. Отображение данных в CAN-кадры задается по-

средством номеров групп параметров (PGN), которые состояются из зарезервированного бита (R), бита выбора страницы (DP), PF поля и параметра расширения группы PS.

Идентификатор ID ( $ID=P+PGN+SA$ )

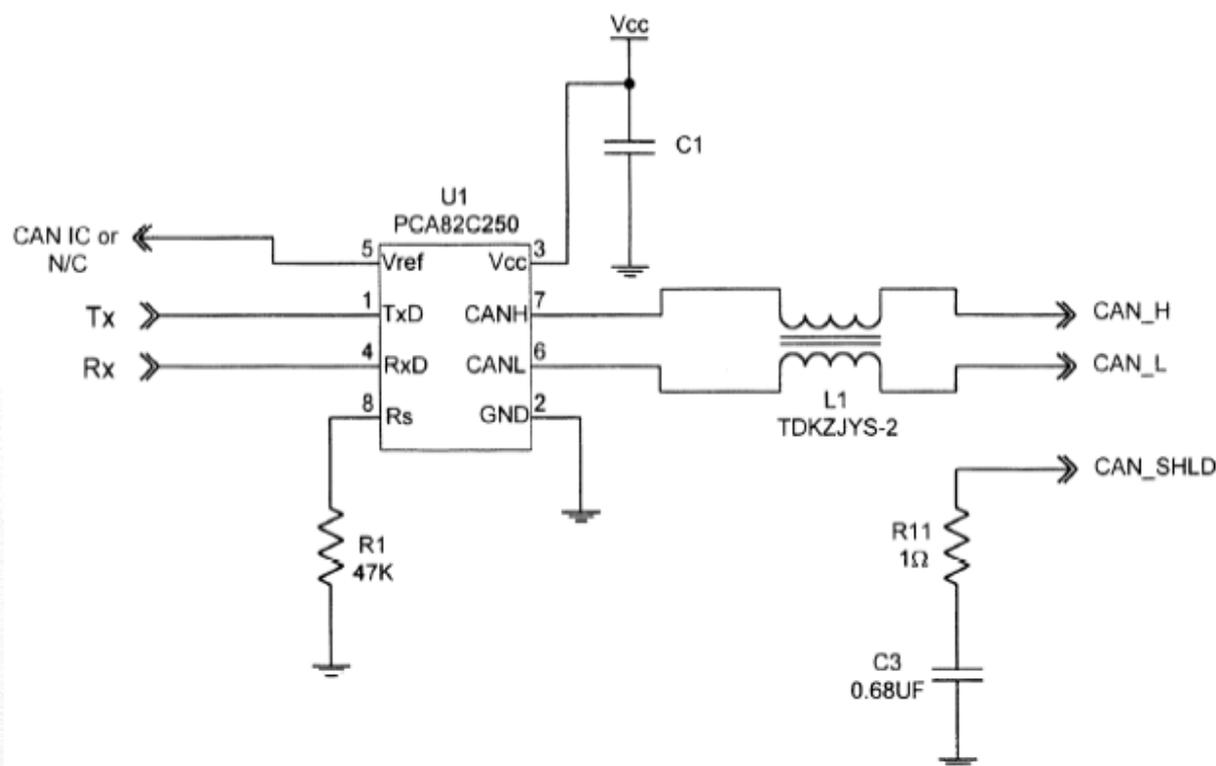


Рисунок 2.33– Трехпроводная схема протокола J1939

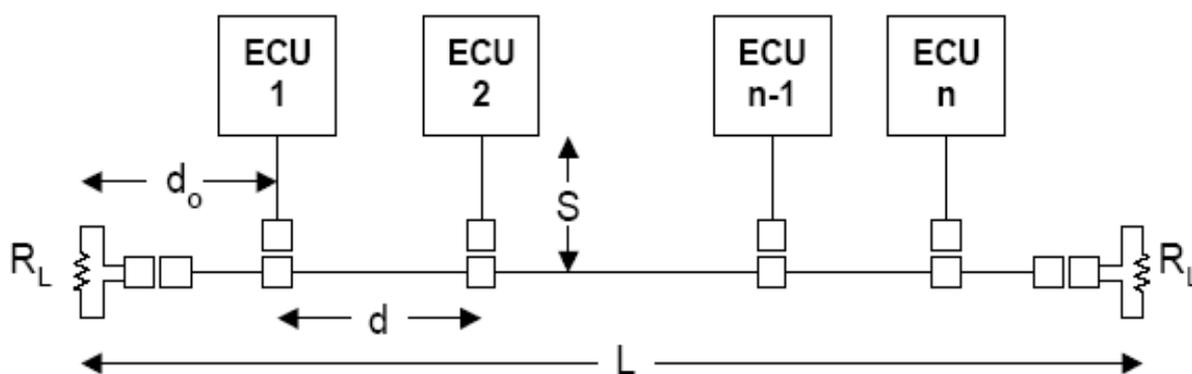
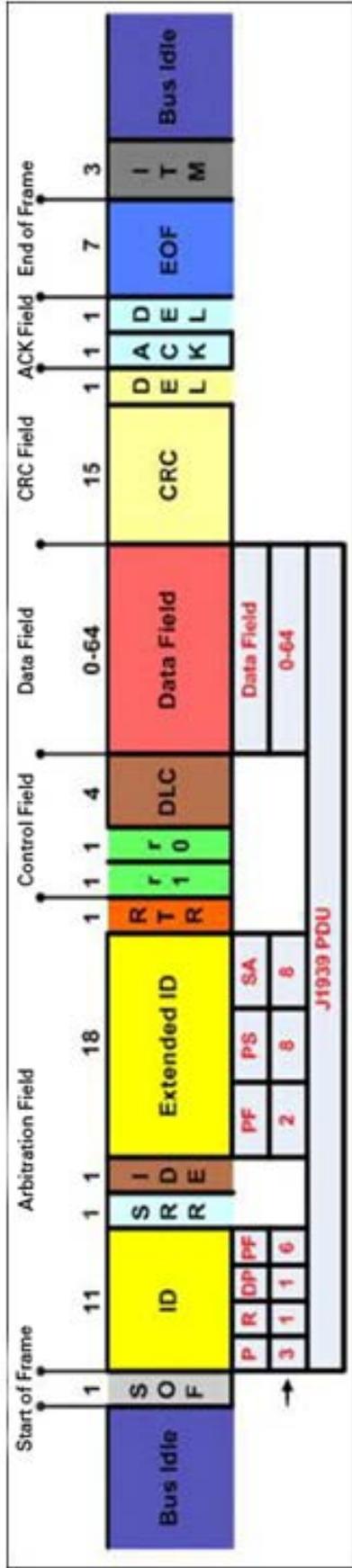


Рисунок 2.34– Физический уровень протокола J1939:

$L$  – максимальная длина сети (40 м),  $S$  – максимальная длина ответвлений (1 м),  $d$  – максимальная дистанция между устройствами 40 м, минимальная 0.1 м.  $d_0$  – минимальное расстояние до концевого резистора (0 м).



- SOF, Start of Frame
- RTR, Remote Transmission Request
- IDE, Identifier Extension
- DLC, Data Length Code
- ACK, Acknowledgement
- EOF, End of Frame
- ITM, Intermission
- SRR, Substitute Remote Request

**ID=Priority+PGN+SA**  
**PGN=R(0)+DP(0)+PF+PS**  
**PDU=ID-DF(data field)**

	PF Data Range	PS Definition
PDU1	0 – 239	Target Address
PDU2	240 – 255	Group Extension

## J1939 PGN

P	R	DP	PF	PS	SA
3	1	1	8	8	8
PGN					
ID					

Рисунок 2.35 – Формат кадра протокола J1939

Первые 3 бита идентификатора используются для определения приоритета сообщения (поле P). При  $P=000_2$  сообщение имеет самый высокий, а при  $P=111_2$  самый низкий приоритет.

Поле R (1 бит) является зарезервированным. Для передаваемых сообщений значение R всегда равно  $0_2$ .

Поле DP (1 бит) служит для выбора страницы данных из всего диапазона PGN (может принимать значения  $0_2$  или  $1_2$ ).

PGN (Parameter Group Number) — это номер группы параметров, определяющий содержимое соответствующего сообщения согласно SAE J1939. ( $PGN=R+DP+PF+PS$ )

Однobaйтовые поля PF (PDUFormat - задает формат PDU) и PS (PDUspecific- задает уточнение PDU). PDU (Protocol Data Unit) — это пакет передачи данных согласно протоколу SAE J1939 ( $PDU=ID+DATA$ , Идентификатор + данные). Если поле PS принимает значения от 0 до 239, то оно содержит десятичный идентификационный адрес устройства-получателя сообщения DA (Destination Address), который обозначается PDU1. Формат PDU1 позволяет передать данные напрямую указанному в адресе DA устройству-получателю сообщения. Если поле PS принимает значения от 240 до 255, то оно обозначает широковещательную адресацию, обозначаемую PDU2. Формат PDU2 может использоваться только для обмена сообщениями, у которых не указан конкретный адрес DA устройства-получателя сообщения. Последнее однobaйтовое поле SA (Source Address) содержит десятичный идентификационный адрес устройства-отправителя сообщения.

Поля DEL (Delimiter), ACK(Acknowledgment), EOF (End of Frame) не подлежат бит-стаффингу.

В протоколе J1939 формализовано не только распределение полей идентификатора, но и заполнение области данных (рисунок 2.36). Передаваемые CAN – кадры имеют как правило, одинаковую длину поля данных – 8 байт. Заполнение области данных определяется кодами отображаемых параметров (SPN), длиной отображаемых данных (Length) и стартовыми позициями (StartPosition) внутри области данных.

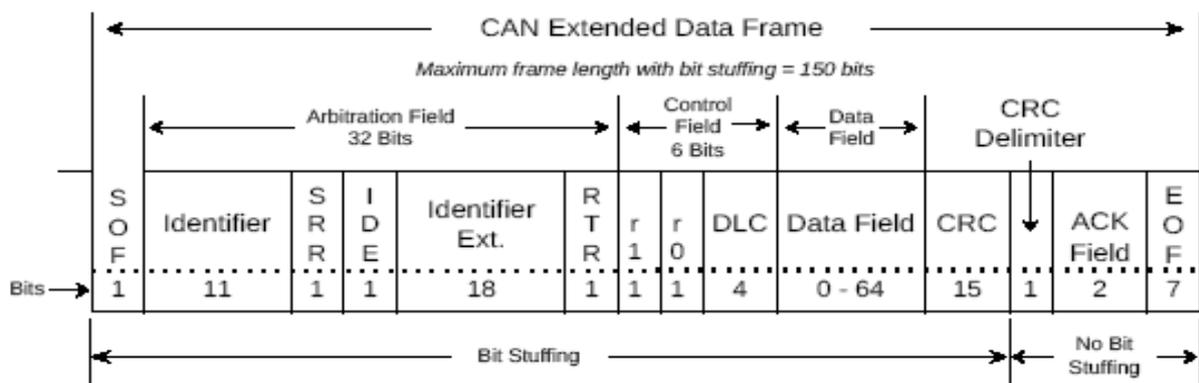


Рисунок 2.36 – Структура кадра данных CAN

На рисунке 2.37 приведен пример заполнения области данных для группы параметров 65213, включающей в себя три параметра: расчетный процент скорости вентилятора (SPN 975), состояние привода вентилятора (SPN 977), скорость вентилятора (SPN 1693).

**PGN 65213 Fan Drive – FD**  
Data Length:8

Start Position	Length	Parameter Name	SPN
1	1 byte	Estimated Percent Fan Speed	975
2.1	4 bits	Fan Drive State	977
3-4	2 bytes	Fan Speed	1639

Byte 1								Byte 2								Byte 3								Byte 4							
8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1
SPN 975								1	1	1	1	SPN 977				SPN 1639															

Byte 5								Byte 6								Byte 7								Byte 8							
8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Рисунок 2.37 – Заполнение области данных для группы параметров 65213

Как видно из примера, в тех случаях, когда размер передаваемых данных менее 8 байт, неиспользованные разряды заполняются логическими единицами. Регламентированный таким образом формат области данных позволяет передавать данные из одной сети в другую непосредственно «бит в бит», не декодируя информацию и соответственно не тратя время и ресурсы на ее обработку. Так же важным свойством подобного формата данных является то, что можно зачитывать только конкретные байты, не считывая область данных полно-

стью, что опять же экономит время и ресурсы устройств и упрощает их программирование.

Протокол J1939 содержит формализованное описание параметров (SPN) и групп параметров (PGN), которыми оперируют электронные блоки автомобилей.

*Пример: температура двигателя.*

Обратимся к тексту стандарта J1939-71, находим группу параметров (PGN), отображающих температуру двигателя. Группа параметров, отображающих температуру двигателя, в соответствии со стандартом, представлена в таблице 2.9.

Таблица 2.9 – Группа параметров, отображающих температуру двигателя

<i>PGN 65262</i>		<i>Engine Temperature 1</i>	<i>-ET1</i>
Transmission Repetition Rate:		1 s	
Data Length:		8	
Extended Data Page:		0	
Data Page:		0	
PDU Format:		254	
PDU Specific:		238 PGN Supporting Information	
Default Priority:		6	
Parameter Group Number:		65262 (0xFEE)	
Start Position	Length	Parameter Name	SPN
1	1 byte	Engine Coolant Temperature	110
2	1 byte	Engine Fuel Temperature 1	174
3-4	2 bytes	Engine Oil Temperature 1	175
5-6	2 bytes	Engine Turbocharger Oil Temperature	176
7	1 byte	Engine Intercooler Temperature	52
7	1 byte	Engine Intercooler Thermostat Opening	1134

Описание группы параметров содержит временной интервал обновления, длину области данных, PDU Format - задает формат PDU, PDU Specific - задает уточнение PDU, приоритет. Для формирования идентификатора не хватает только адреса источника сообщения (SA), который определяется номером устройства, посылающего сообщение. Номер устройства может быть присвоен производителем конкретного транспортного средства.

Область данных заполняется в соответствии с кодами SPN, их стартовыми позициями и длиной (таблица 2.10)

Таблица 2.10 – Заполнение области данных для группы параметров 65262

<b>Byte 1</b>	<b>Byte 2</b>	<b>Byte 3</b>	<b>Byte 4</b>
SPN 110	SPN 174	SPN 175	
<b>Byte 5</b>	<b>Byte 5</b>	<b>Byte 7</b>	<b>Byte 8</b>
SPN 176		SPN 52	SPN 1134

Снова обратимся к тексту стандарта и найдем параметры из данной группы (таблицы 2.11 – 2.16).

Таблица 2.11 – SPN 110

<b>SPN 110</b>	<b>Engine Coolant Temperature</b>
Temperature of liquid found in engine cooling system	
Data Length:	1 byte
Resolution:	1 deg C/bit, -40 deg C offset
Data Range:	-40 to 210 deg C
Operational Range	same as data range
Type:	Measured
Supporting information:	
PGN	65262

Таблица 2.12 – SPN 174

<b>SPN 174</b>	<b>Engine Fuel Temperature 1</b>
Temperature of fuel (or gas) passing through the first fuel control system. See SPN 3468 for the second control system	
Data Length:	1 byte
Resolution:	1 deg C/bit, -40 deg C offset
Data Range:	-40 to 210 deg C
Operational Range	same as data range
Type:	Measured
Supporting information:	
PGN	65262

Таблица 2.13 – SPN 175

<b>SPN 175</b>	<b>Engine Oil Temperature 1</b>
Temperature of the engine lubricant.	
Data Length:	2 byte
Resolution:	0,03125 deg C/bit, -273 deg C offset
Data Range:	-273 to 1735deg C
Operational Range	same as data range
Type:	Measured
Supporting information:	
PGN	65262

Таблица 2.14 – SPN 176

<b>SPN 176</b>	<b>Engine Turbocharger Oil Temperature</b>
Temperature of the engine lubricant.	
Data Length:	2 byte
Resolution:	0,03125 deg C/bit, -273 deg C offset
Data Range:	-273 to 1735deg C
Operational Range	same as data range
Type:	Measured
Supporting information:	
PGN	65262

Таблица 2.15 – SPN 52

<b>SPN 52</b>	<b>Engine Intercooler Temperature</b>
Temperature of liquid found in the intercooler located after the turbocharger.	
Data Length:	1 byte
Resolution:	1 deg C/bit, -40 deg C offset
Data Range:	-40 to 210 deg C
Operational Range	same as data range
Type:	Measured
Supporting information:	
PGN	65262

Таблица 2.16 – SPN 1134

SPN 1134	Engine Intercooler Thermostat Opening
The current position of the thermostat used to regulate the temperature of the engine intercooler. A value of 0% represents the thermostat being completely closed and 100% represents the thermostat being completely open.	
Data Length:	1 byte
Resolution:	0,4 %/bit, 0 offset
Data Range:	0 to 100 %
Operational Range	same as data range
Type:	Measured
Supporting information:	
PGN	65262

Описание SPN содержит длину данных, разрешение, диапазон измерения, тип данных (измеряемые или статусные). Таким образом, идентификатор сообщения определяется PGN, область данных – SPN. (рисунок 2.38).

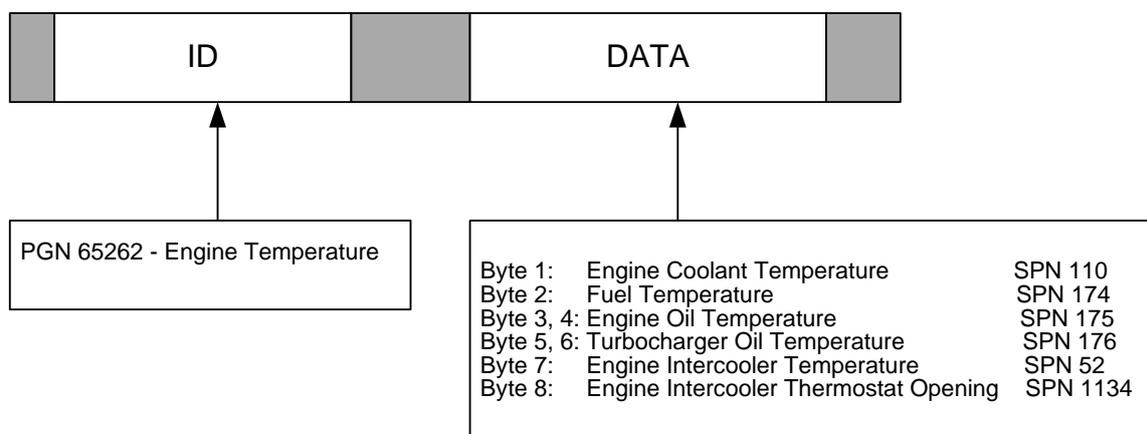


Рисунок 2.38 – Определение идентификатора сообщения и требуемой области данных

Передача пакета данных длиной до 8 байт включительно осуществляется одним сообщением. Если требуется передать более 8 байт данных, то пакет данных делится на несколько сообщений длиной по 8 байт. Максимальная длина пакета данных для одного PGN составляет 1785 байт. При этом первый байт в области данных используется для нумерации сообщений (от 1 до 255) оставшиеся 7 байт используются для передачи данных ( $255 \times 7 = 1785$ ).

Протокол J1939 предназначен в основном для использования в автомобилях, однако, возможно создание собственного протокола J1939, для этого нужно только разработать свои группы параметров и описать параметры в них входящие, т.е. разработать собственные PGN и SPN.

CAN протокол определяет безопасную передачу небольших пакетов данных, используя общую линию коммуникации. Протокол не содержит средств контроля потока, адресацию, не предоставляет передачу сообщений более чем 8 бит, не осуществляет установку соединения и т.д. Перечисленные свойства определяются HLP, higher layer protocol (протоколами высшего порядка).

HLP обычно определяет:

- ◆ Параметры запуска;
- ◆ Распределение идентификатора сообщения среди различных устройств в системе;
- ◆ Интерпретация содержимого блоков данных;
- ◆ Статус взаимодействия в системе

Применение HLP обязательно, в противном случае придется написать собственный HLP.

#### 2.3.5.2. *CAN Kingdom*

CAN Kingdom поддерживается организацией CAN Kingdom International полная спецификация доступна на сайте организации. CAN Kingdom обычно упоминается как CAN (Controller Area Network) протокол высшего порядка.

В реальности наиболее упорядоченный протокол. Модули в системе соединены сетью, в которой один из модулей является главным (King). Например, для организации plug & play системы, главный модуль определяет какое устройство и при каких обстоятельствах может быть добавлено, разрешено добавление только специфицированных устройств. CAN Kingdom обеспечивает простую уникальную идентификацию устройств в системе, для этого используется стандарт идентификации EAN/UPC, индивидуальный идентификатор устройства определяется серийным номером устройства.

CAN Kingdom предоставляет разработчику все потенциальные возможности CAN. Разработчик не ограничен мультимастер протоколом CSMA/AMP и может создавать виртуальные системы управления

шинами всевозможных разновидностей и топологии. Предоставляет возможность создания общих модулей без учета обстоятельств таких как, зависимость от HLP и свойств системы. Разработчик может определить использование только специфических модулей, совмещая тем самым ценности открытой системы с преимуществами системы с ограниченным и безопасным доступом.

Потому как идентификатор в CAN сообщениях не только идентифицирует сообщение, но также управляет доступом к шине, ключевое значение имеет нумерация сообщений. Другой важный фактор – это идентичность структуры данных в поле данных, как в передающем, так и принимающем модулях. Введением небольших, простых правил, указанные факторы полностью контролируются и коммуникации оптимизированы для любой системы. Это выполняется во время короткой фазы установки при инициализации системы. Так же возможно включение устройств, не следующих CAN Kingdom правилам, в CAN Kingdom систему. CAN Kingdom сопровождается соответствующей документацией по модулям и системам.

#### *2.3.5.3. CAL and CANopen*

CAL сокращенно от «CAN Application Layer» порядок или слой CAN приложений, протокол поддерживается CiA.

CAL разделен на несколько составных частей:

- ◆ CMS (CAN-based Message Specification) определяет протоколы передачи данных между CAN устройствами;
- ◆ NMT (Network Management Service) определяет протоколы запуска и выключения, определения неисправностей, и т.д.;
- ◆ DBT (Distributor Service) определяет протокол распределения идентификаторов различных устройств в системе.

CAL протокол отличный от OSI модели (Open Systems Interconnect (OSI) Model).

CANopen является подразделом CAL, и скомпонован как набор профилей, которые не завершены окончательно. CAL и CANopen спецификации в полном объеме доступны и поддерживаются CiA.

#### *Прикладной уровень: CANopen*

Прикладной уровень модели OSI обеспечивает интерфейс между сетью и программным приложением, которое может взаимодей-

ствовать с аналогичными приложениями в других устройствах сети. На прикладном уровне реализуется также механизм синхронизации между устройствами.

К сожалению, разработка CAN закончилась на первых двух уровнях модели OSI. Это привело к появлению множества несовместимых между собой протоколов прикладного уровня, среди которых самыми распространенными и поддерживаемыми организацией SiA являются CANopen и DeviceNet

Канальный уровень CAN, рассмотренный выше, практически невозможно использовать в SCADA-пакетах, поскольку он оперирует битами, фреймами, полями. Для написания же прикладных программ нужно использовать понятия: переменная, массив, событие, клиент, сервер, имя устройства и т. п.

Рассмотрим наиболее распространенный стандарт прикладного уровня CANopen. Для упрощения применения стандарта вводятся несколько специфических для CANopen понятий. Все функциональные возможности прикладного уровня делятся между так называемыми сервисами (элементами услуг). Программные приложения взаимодействуют между собой путем вызова соответствующих сервисов прикладного уровня. Сервисы обмениваются данными с равными им (одноранговыми) сервисами через CAN-сеть с помощью определенного протокола. Этот протокол описывается в спецификации протокола сервиса.

Вводится понятие сервисного примитива, который представляет собой средство (языковую конструкцию), с помощью которого программное приложение взаимодействует с прикладным уровнем. В CANopen существует четыре различных примитива:

- 1) запрос приложения к прикладному уровню, публикуемый приложением для вызова сервиса;
- 2) индикация, публикуемая прикладным уровнем для приложения, чтобы сообщить о внутренних событиях, обнаруженных прикладным уровнем или чтобы показать, что сервис запрошен;
- 3) ответ, публикуемый приложением для прикладного уровня, чтобы ответить на ранее полученную индикацию;
- 4) подтверждение, публикуемое прикладным уровнем для приложения, чтобы отчитаться о результатах ранее изданного запроса.

Сервисы также делятся на несколько типов сервисов:

- 1) локальный сервис - который выполняет запрос приложения без взаимодействия с другими сервисами того же ранга;

2) неподтвержденный сервис - который вовлекает в выполнение запроса один или более других одноранговых сервисов. Приложение посылает запрос к локальному сервису. Этот запрос передается далее сервису (сервисам) того же ранга;

3) подтвержденный сервис может вовлечь только один сервисный объект того же ранга. Приложение издает запрос к его локальному сервису. Этот запрос передается сервису того же ранга, который передает его другому приложению как индикацию. Другое приложение издает ответ, который передается исходному сервису, который передает его как подтверждение запрашивающему приложению;

4) сервис, инициированный провайдером - вовлекает только локальный сервис.

CANopen предлагает серию стандартизованных коммуникационных механизмов и функций, выполняемых устройствами в сети (профилей). Серия профилей доступна и поддерживается организацией CiA (CAN in Automation); для ее использования не требуется лицензий.

Устройство в сети CANopen представляется состоящим из трех частей:

- ◆ коммуникационный интерфейс (к шине CAN) и программный протокол обмена;
- ◆ словарь объектов;
- ◆ интерфейс к устройствам ввода-вывода и прикладная программа.

Коммуникационный интерфейс и программный протокол обеспечивают сервис по передаче и получению через сеть коммуникационных объектов. Словарь объектов описывает типы данных, коммуникационные объекты и прикладные объекты, использованные в устройстве для обмена через интерфейс к устройствам ввода вывода. Прикладная программа обеспечивает внутреннее управление функциями устройства и интерфейсом к устройствам ввода-вывода.

Наиболее важной частью устройства в CANopen является словарь объектов. Под объектами понимаются типы данных, профили устройств, коммуникационные объекты, регистр ошибок. Каждый объект в словаре адресуется 16-битным индексом.

В CANopen используются следующие типы данных: Boolean, Integer, UnsignedN, Float, Date, Time, которые имеют общепринятый смысл. Имеется также несколько сложных типов данных для PDO и

SDO параметров (Process Data Object - объект данных технологического процесса и Service Data Object - объект сервисных данных).

Объекты PDO и SDO используются для передачи данных. Сообщения PDO позволяют передавать данные в реальном времени. Существует два типа объектов PDO. Первый из них выполняет передачу данных (Transmit-PDO или TPDO), второй - прием данных (Receive-PDO или RPDO). Коммуникационные параметры PDO определяют его коммуникационные возможности и описываются в словаре объектов.

Объект SDO обеспечивает доступ к словарю объектов. SDO может использоваться также для передачи групп данных от клиента к серверу и наоборот.

Имеются также объекты специального назначения (объекты для синхронизации, объекты меток времени, объекты аварийных ситуаций), а также объекты управления сетью (объекты начальной загрузки, объекты контроля ошибок и сообщения для управления сетью).

#### *Коммуникационные модели*

Коммуникационная модель CANopen определяет различные коммуникационные объекты и сервисы, а также доступные режимы запуска передачи сообщений, поддерживает передачу синхронных и асинхронных сообщений. Синхронные сообщения используются для сбора данных или управления исполнительными устройствами. Синхронные сообщения передаются относительно сообщений синхронизации, которые определяются заранее; асинхронные сообщения могут передаваться в любое время.

В CANopen используют три типа взаимодействий между передающим и принимающим устройством:

- 1) ведущий/ведомый;
- 2) клиент/сервер;
- 3) производитель/потребитель.

#### *Электронные спецификации устройств CANopen*

Поскольку устройства, используемые в сети, являются программируемыми, перед их включением в сеть необходимо задать параметры, необходимые для их коммуникаций с сетью и функционирования. CANopen устанавливает для этого стандартизованный метод. Метод

предполагает наличие электронного описания устройств в текстовом формате, для обработки которого достаточно несложного компилятора. CANopen определяет формат EDS (Electronic Data Sheet – «электронный список параметров»), который описывает конфигурацию и параметры устройств, в том числе контроллеров с модульной архитектурой.

EDS поддерживается и поставляется производителем устройства. В противном случае используется EDS «по умолчанию», общий для определенного класса устройств, например, модулей аналогового ввода.

EDS является текстовым файлом, использующим ASCII-коды (набор символов по стандарту ISO 646). Длина строки файла - 255 символов, строки должны оканчиваться символами CR или LF.

Файл содержит несколько секций:

- ◆ информация о самом файле (имя файла, версия, дата создания, версия EDS, описание, кем создан, дата модификации и др.);
- ◆ общая информация об устройстве (имя производителя, идентификационный код производителя, имя устройства, код устройства, номер версии, функции устройства, список поддерживаемых скоростей обмена, наличие программы начальной загрузки и др.);
- ◆ конфигурационные параметры (длительность цикла обмена, тип устройства, тип данных, нижний и верхний предел изменения переменных, значения по умолчанию, количество каналов ввода-вывода и др.).

#### 2.3.5.4. DeviceNet

Протокол развивается «Rockwell Automation nowadays», определен организацией ODVA (Open DeviceNet Vendor Association). DeviceNet один из протоколов, которые поддерживает CiA.

##### Smart Distributed System (SDS)

SDS система, на основе шины для интеллектуальных датчиков и исполнительных устройств, с упрощенным процессом установки, предоставляет широкие возможности управления вводом – выводом. Посредством одного четырехпроводного кабеля SDS система может быть оборудована до 126 приборами с индивидуальным адресом. Дополнительная информация и спецификация по SDS доступна на сайте разработчика Honeywell. SDS один из действующих четырех протоколов, поддерживаемых CiA.

### ***2.3.6. Сравнительная характеристика основных HLP протоколов***

DeviceNet, SDS и CAN Kingdom основаны на ISO 11898 CAN коммуникационном протоколе и функционируют согласно требований CAN спецификации. Каждый CAN модуль, следующий определенному протоколу, может быть подключен к CAN шине, следующей тому же протоколу. В любом случае при подключении модулей, которые действуют по различными протоколам, в большинстве случаев проблемы возникают по причине конфликта интерпретации сообщений на уровне приложений.

CAN Kingdom отличается от SDS и DeviceNet основным принципом: CAN Kingdom организуется главным узлом коммуникации («King») при запуске, в отличие от SDS и DeviceNet. Такая организация позволяет упростить разработку комплекса систем реального времени и сокращает необходимое количество модулей, координирующих спецификации, часто обозначаемые как профили.

SDS эффективен при подключении I/O устройств, различных выключателей и датчиков к PLC, фактически представляет собой соединение между основным модулем и удаленными I/O устройствами.

DeviceNet открытая система, в которой все модули имеют равные права по пользованию шиной, и порядок пользования шиной определяется небольшим набором инструкций. Разработчик модулей системы может передать полномочия по управлению коммуникацией другим модулям, например, основному модулю в предопределенном режиме главный/подчиненный, но DeviceNet не имеет возможности передать полномочия другим модулям. Характеристики SDS с использованием I/O устройств и DeviceNet в режиме главный/подчиненный сходны.

Can Kingdom протокол, ориентированный на системы производства, соединения и контроля и не поддерживает профили для цифровых и аналоговых устройств. Основная особенность протокола заключается в том, что модуль, подключенный к системе, только ожидает инструкции от главного устройства. Все CAN приоритеты и идентификаторы принадлежат и предоставляются главным устройством. Во время запуска каждый модуль конфигурируется основным устройством, определяются приоритеты и идентификаторы объектов, продуцирующих и потребляющих. Основное устройство является

главным, но только в период конфигурирования системы. Главное устройство не может быть внедрено в период коммуникационной сессии между работающими приложениями различных модулей. Основное устройство может быть удалено после конфигурирования и проверки комплектности, притом каждый модуль запоминает полученные инструкции в памяти.

CAN не требуется физическая адресация и номер узла. Это является важным свойством CAN, так как модули не должны иметь сведений о системе, в которой функционируют. Но во время конфигурирования системы и во время технического обслуживания сети, специфические узлы должны быть адресованы. Для этих целей, по крайней мере, один CAN приоритет/идентификатор должен быть зарезервирован и каждому модулю присвоен номер узла. CAN Kingdom, DeviceNet и SDS используют подобную технологию.

**DIP переключатель.** Свойства: повышенная возможность возникновения ошибок, допущенных человеком, и ошибок по причине неисправностей соединения. Обычно отсутствуют ошибки, которые возможно проверить чек суммой. Зачастую переключатель должен быть изолирован от окружающей среды, поэтому требуется использование отвертки или подобного инструмента для доступа к переключателю. Установки открытого переключателя могут быть случайно изменены. *Рекомендации:* номер узла должен быть доступен без применения, какого-либо сервисного инструмента.

**Кодирование контактами разъема.** Свойства: повышенная возможность возникновения ошибок, допущенных человеком, и ошибок по причине неисправностей соединения. Обычно отсутствуют ошибки, которые возможно проверить чек суммой. Установки разъема могут быть доступны или в открытом состоянии, либо посредством смотрового окна в корпусе. Требуется документация по изменению установок разъема. Ограниченное количество комбинаций. Высокая стоимость контактов разъема. *Рекомендации:* номер узла должен быть доступен без применения, какого-либо сервисного инструмента.

**Хранение во внутренней памяти.** Свойства: номер узла должен быть установлен в модуль до подключения к системе. *Рекомендации:* Обычно номер узла сохраняется с чек суммой в энергонезависимой памяти. Низкая стоимость и наличие нескольких бит памяти. В

памяти так же можно сохранить дополнительную информацию. Такой способ более защищен от сбоев и ошибок в отличие от контактов и переключателей.

**Хранение номера в памяти разъема.** Свойства: отсутствие в продаже разъемов со встроенной памятью. Потребуется дополнительные контакты для считывания информации из памяти. *Рекомендации:* тоже, что для внутренней памяти. Если номер узла сохраняется и в памяти разъема, и в энергонезависимой внутренней памяти, достигается высокий уровень защиты от замены и перемещения модулей. Не требуется применение сервисных инструментов.

Ни в одном из HLP протоколов не специфицирован метод установки номера узла. SDS и DeviceNet описывают применение переключателей с применением каких-либо разновидностей инструментов.

В CAN сети важно чтобы все модули были установлены для обмена данными на одной скорости. Следующим, после короткого замыкания CAN шины, наипростейшим способом разрушить коммуникации является установка модуля с некорректной или очень низкой скоростью обмена данными. Обычные последствия заключаются в том, что остальные модули отключаются. CAN Kingdom и Device Net рекомендуют различные способы, чтобы избежать подобные последствия.

SDS описывает рекомендованную методику автоматической настройки скорости обмена данными. Скорость обмена устанавливается главным устройством, которому присвоен наименьший адрес. Другие модули проверяют своевременность обмена по шине и устанавливают собственную скорость соответственно. Рекомендованные значения 125 Кбит/с, 250Кбит/с, 500Кбит/с и 1 Мбит/с.

Device Net определяет три значения скорости 125Кбит/с, 250Кбит/с и 500Кбит/с, но не обеспечивает защиту от сбоев установки скорости обмена данными. Если не определено, некоторые Device Net устройства используют автоматическую настройку скорости.

CAN Kingdom не определяет скорость обмена или автоматическую настройку. Модуль после включения питания в течение первых 200 мс должен «слушать» пассив на скорости 125 Кбит/с с фиксированными и определенными установками. Во время этой процедуры неправильно настроенный модуль может быть всегда доступен при

таких условиях. Пассивная коммуникация означает, что модуль только «слушает» CAN сеть, но не предоставляется возможность передачи доминантных битов по шине.

Приоритет доступа к CAN шине дается первыми 11 или 29 битами сообщения, и называется «Поле идентификатора» или более корректное название «Поле приоритета». 11 бит ID называется Standard ID и 29 бит Extended ID. SDS и Device Net используют Std. IDs. CAN Kingdom использует и Std и Ext. IDs. SDS позволяет подключение 125 модулей к сети, и каждому модулю присваивается набор IDs относительно каждого номера узла.

И SDS и Device Net определяют профили, для большого количества различных устройств, включая поведение и структуру данных принимающих и передающих модулей. Необходимо что бы модуль, не принадлежащий системе, соответствовал протоколу системы. В CAN Kingdom всегда присутствует модуль, который отвечает за систему, по крайней мере, при запуске системы первый раз. Определяется возможности модуля приспособляться к конкретным условиям, т. е. профилированием системных данных, таких как скорость обмена, номер узла и приоритеты.

Сравнительная характеристика HLP протоколов SDS, Device Net и CAN Kingdom представлена в таблицах 2.17-2.20.

Таблица 2.17 – Скорость передачи данных

	<b>SDS</b>	<b>Device Net</b>	<b>CAN Kingdom</b>
Возможная скорость передачи данных	125 Кбит/с, 250 Кбит/с, 500 Кбит/с, 1 Мбит/с	125 Кбит/с, 250 Кбит/с, 500 Кбит/с	любая, при обслуживании 125 Кбит/с
Защита от устройств с некорректной скоростью передачи	да	нет	да
Автоматическая настройка скорости	да, специфицирована	возможно, но не специфицировано	возможно, но не специфицировано
Возможность изменения протоколом HLP	нет, только автоматическая настройка	да	нет, если установлен переключатель кода

## Выводы:

### SDS

- ◆ компактный и эффективный способ соединить небольшие устройства к основному контроллеру;
- ◆ главное устройство имеет полный контроль над остальными модулями;
- ◆ отсутствует поддержка соединения между модулями без основного PLC;
- ◆ поддерживается только Std. CAN.

Таблица 2.18 – Количество узлов

	<b>SDS</b>	<b>DeviceNet</b>	<b>CAN Kingdom</b>
Возможное количество устройств	0-125	0-63	(0) 1-255
Рекомендованное количество устройств	125	63	не определено; устанавливается в сервисном режиме или определяется коннектором
Защита от дублирования номеров	нет/ да. HLP поддерживает проверку контролирующим устройством	Да, с помощью Duplicate MAC ID Check	нет/ да, HLP поддерживает проверку главным контроллером
Возможность изменения протоколом HLP	да	да, нет, если установлен переключатель	да

### DeviceNet

- ◆ открытая система, где каждый модуль является локально главным;
- ◆ разработчик системы не имеет возможности контроля над модулями;
- ◆ поддержка модели производительности и потребления при передаче от модуля к модулю;
- ◆ ограниченное количество, 27 свободно используемых приоритетов в одном модуле;
- ◆ компактное ядро в предопределенной коммуникационной установке главный/подчиненный;
- ◆ поддерживается только Std. CAN, CAN Kingdom

- ◆ разработчик системы имеет полный контроль над всеми модулями посредством главного контроллера;
- ◆ поддержка модели производительности и потребления при передаче от модуля к модулю;
- ◆ возможности контроля в режиме реального времени за состоянием CAN шины;
- ◆ полная утилизация функции приоритета в CAN протоколе;
- ◆ поддержка формирования преобразования структуры данных в HLP;
- ◆ компактное ядро, обычно 500-1500 байт кода и 24-48 байт RAM;
- ◆ Поддерживается только Std. CAN

Таблица 2.19 – Базовые данные приоритетов и идентификаторов

	<b>SDS</b>	<b>Device Net</b>	<b>CAN Kingdom</b>
Приоритеты, присваиваемые модулю при запуске	8 + (8)	31 + (63)	0/1 + (2)
Приоритеты открытые для общего пользования	нет	26 для каждого модуля	любые, которые еще не используются
CAN запрос дистанционной передачи	нет	нет	да
Extended CAN	нет	нет	да
Системный контроль приоритетов	нет, предоставляется выбором номера узла	3 группы содержат 16, 5 и 5 данных приоритетов из которых можно выбрать	да, определяется дизайном системы
Свободные приоритеты	нет	нет	все
Старт с предопределенными установками	фиксированы HLP, всегда предопределены	не поддерживается HLP, каждый модуль свободно оперирует присвоенными 27 приоритетами/IDs	King может отдать команду с установками из энергонезависимой памяти
Автоматический запуск при включении питания	да, после автоматической настройки скорости обмена	каждый модуль свободно оперирует присвоенными 27 приоритетами/IDs	да, если предварительно разрешено King

	<b>SDS</b>	<b>Device Net</b>	<b>CAN Kingdom</b>
Предопределенный приоритет/IDs при запуске и зарезервированные для модуля	<i>при запуске:</i> 8Tx, 8 Rx; <i>от основного устройства:</i> 8N <sup>*</sup> -8N+7; <i>к основному устройству:</i> 1024+ 8N-8N+7	<i>при запуске:</i> 2 Tx, 3 Rx; Grp1:N+64 <sup>2</sup> M <sup>**</sup> ; M=0-15 Grp2:8N+1024+M; M=0-7; Grp3:N+1536+64M <sup>2</sup> =0-6;	<i>при запуске:</i> во время первых 200 мс: 0 Tx, 2Rx0 and 2031; далее любой номер, предварительно установленный главным узлом.
*N – любой номер узла используемый модулем в системе **M - ID сообщения определенный в Device Net спецификации			

Таблица 2.20 – Управление системой

	<b>SDS</b>	<b>DeviceNet</b>	<b>CAN Kingdom</b>
Восстановление исходных настроек модуля	нет	да, в период установленной связи	да
Организация модуля в группу	да, 1 группа	нет	да, 255 минус количество модулей в системе
Установка CAN маски приема	нет	нет	да

### 2.3.7. Элементная база для реализации сети CAN

2.3.7.1. Микросборка приемопередатчика по стандарту CAN с гальванической развязкой 2011BB034.

Микросборка 2011-BB034 (далее – МСБ) предназначена для использования в аппаратуре специального назначения (рисунок 2.39 –2.41), в качестве приемопередатчика сигналов цифрового интерфейса CAN. МСБ может использоваться для создания устройств высоковольтной гальванической развязки.

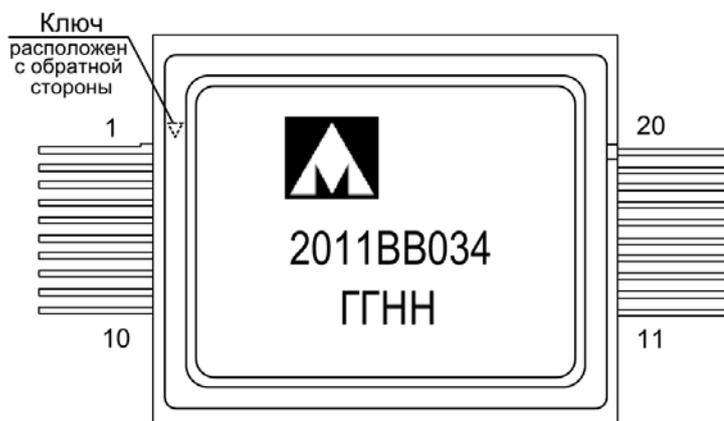


Рисунок 2.39 – Микросборка 2011BB034  
ГГ – год выпуска, НН – неделя выпуска

Основные характеристики микросборки:

- ↻ Напряжение источника питания, UCC,  $5,0 \pm 10\%$  В;
- ↻ Выходное напряжение дифференциальное рецессивного состояния, UO\_DIFF\_REC, на выводах CANH и CANL (без нагрузки) от минус 500 до 50 мВ;
- ↻ Входное напряжение дифференциальное высокого уровня, UIDH, на выводах CANH, CANL при CANH>CANL от 0,9 до 5 В;
- ↻ Входное напряжение дифференциальное низкого уровня, UIDL, на выводах CANH, CANL при CANH>CANL от 0 до 0,5 В;
- ↻ Динамический ток потребления, IOCC, не более 170 мА;
- ↻ Ток потребления в состоянии «Выключено», ICCZ не более 560 мкА;
- ↻ Скорость передачи битов данных, VDR, не более 10 Мбит/с;
- ↻ Температурный диапазон:

2011BB034	$-60 \div +85\text{ }^{\circ}\text{C}$ ;
K2011BB034	$-60 \div +85\text{ }^{\circ}\text{C}$
K2011BB034K	$0 \div +70\text{ }^{\circ}\text{C}$

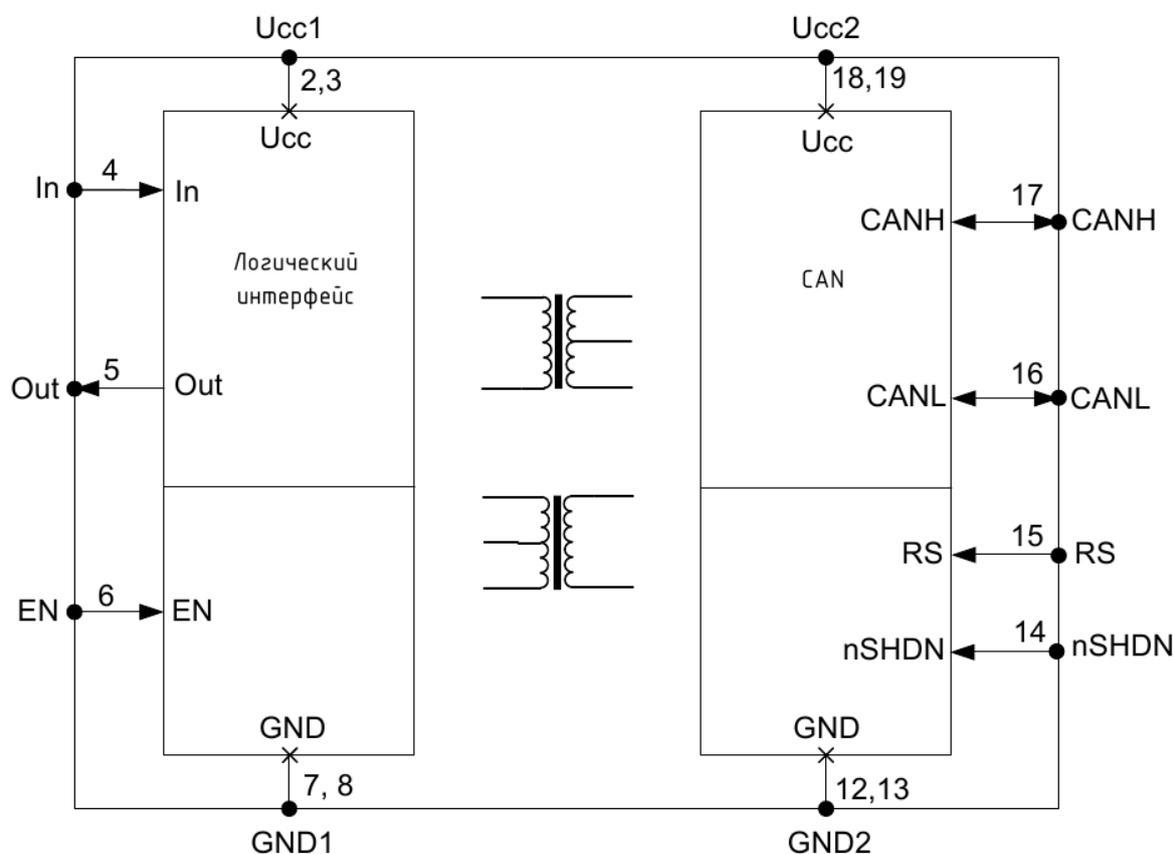


Рисунок 2.40 – Структурная блок-схема МСБ

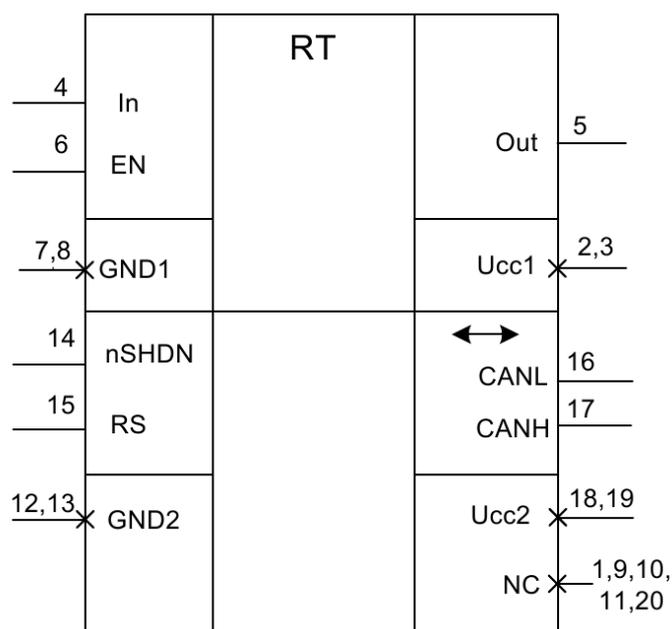


Рисунок 2.41 – Условное графическое обозначение

Таблица 2.21 – Описание выводов

№ вывода корпуса	Обозначение вывода	Функциональное назначение выводов
1	NC	Не используется
2, 3	Ucc1	Питание логического интерфейса
4	In	Вход логического информационного сигнала передатчика
5	Out	Выход логического информационного сигнала приемника
6	EN	Вход разрешения работы логического интерфейса приемника
7, 8	GND1	Общий
9	NC	Не используется
10	NC	Не используется
11	NC	Не используется
12, 13	GND2	Общий
14	nSHDM	Вход выбора режима «Выключено». Отключает выходы передатчика CAN. (Активный низкий уровень сигнала)
15	RS	Вход (аналоговый) выбора режима работы
16	CANL	Вход приемника CAN/выход передатчика CAN. Инверсный
17	CANH	Вход приемника CAN /выход передатчика CAN. Прямой
18, 19	Ucc2	Питание интерфейса CAN
20	NC	Не используется

МСБ 2011ВВ034 представляет собой преобразователь интерфейса CAN в цифровой сигнал и обратно.

МСБ предназначена для преобразования передаваемого сигнала интерфейса CAN в дифференциальный импульсный сигнал, подаваемый на первичную обмотку развязывающего трансформатора, а также преобразования принимаемого импульсного сигнала со вторичной обмотки трансформатора в выходной сигнал интерфейса CAN. Используется для создания устройств высоковольтной гальванической развязки передаваемых сигналов с использованием импульсного трансформатора.

МСБ содержит приемопередатчик интерфейса CAN и кодер/декодер трансформаторного интерфейса. При использовании МСБ можно получить гальванически развязанную дуплексную линию связи CAN.

Блок-схема вариантов приемопередатчика с гальванической развязкой представлена на рисунке 2.42.

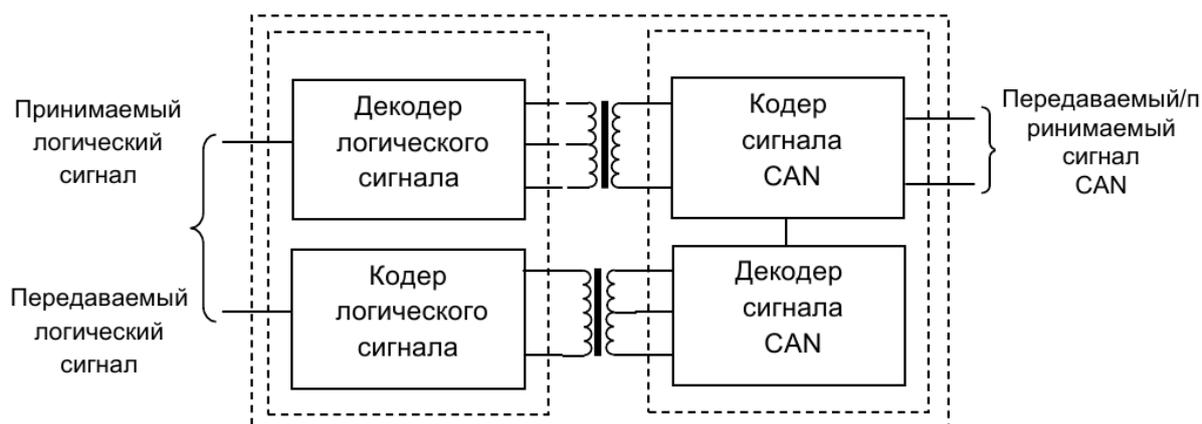


Рисунок 2.42 – Блок-схема преобразователя логического интерфейса в интерфейс CAN

Для выбора режима «Контроль скорости» необходимо подключить резистор между входом RS и выводом «Общий». В этом режиме номинал резистора определяет величину скорости нарастания/спада выходного сигнала, что необходимо для уменьшения уровня электромагнитных помех, а также отражений при неидеально согласованной шине. Таким образом, обеспечивается стабильная передача информации со скоростью от 40 до 500 Кбит/с.

Величину подключаемого резистора  $R_{RS}$  можно рассчитать по формуле:  $R_{RS} [\text{кОм}] = 12\,000 / \text{Скорость передачи} [\text{Кбит/с}]$ .

### 2.3.7.2. Микросхема приемопередатчика интерфейса CAN 5559ИН14.

Микросхема приемопередатчика интерфейса CAN предназначена для организации полудуплексного канала связи с максимальной скоростью передачи данных до 1 Мбит/с (рисунок 2.43, 2.44). Микросхема доступна в трех исполнениях: с выходом опорного напряжения UREF, с входами управления режимом «выключено» SHDN или /SHDN.

Основные области применения: автомобильные и промышленные системы управления.

Основные характеристики микросхемы:

- ↯ Соответствует стандарту ISO 11898-2;
- ↯ Напряжение питания от 4,5 В до 5,5 В;
- ↯ Защита выходов передатчика  $\pm 40$  В от короткого замыкания и перегрева для применения в 12/24 В автомобильных и промышленных системах управления
- ↯ Быстродействующий дифференциальный приемник с диапазоном входного синфазного напряжения от минус 10 В до 10 В;
- ↯ Четыре режима работы:
  - ↯ Нормальный режим, максимальная скорость передачи данных до 1 Мбит/с;
  - ↯ Режим контроля скорости нарастания/спада выходного дифференциального напряжения передатчика для улучшения электромагнитной совместимости, скорость передачи данных от 40 Кбит/с до 500 Кбит/с;
  - ↯ Режим «Ожидание» с пониженным потреблением;
  - ↯ Режим «Выключено»;
- ↯ Входы TXD, SHDN и /SHDN совместимы с 3,3 В логическими уровнями;
- ↯ Рабочий диапазон температур:

5559ИН14А(Б,В)У	– 60 ÷ + 125 °С
К5559ИН14А(Б,В)У	– 60 ÷ + 125 °С

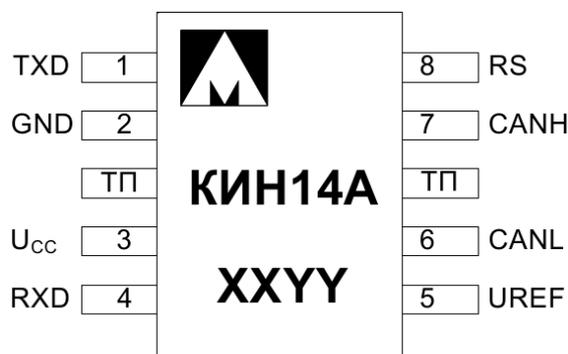


Рисунок 2.43 – Микросхема приемопередатчика

K5559ИH14A(Б,В)УК	0 ÷ + 70 °С
K5559ИH14ASI	- 45 ÷ + 125 °С
K5559ИH14BSI	- 45 ÷ + 125 °С
K5559ИH14BSI	- 45 ÷ + 125 °С

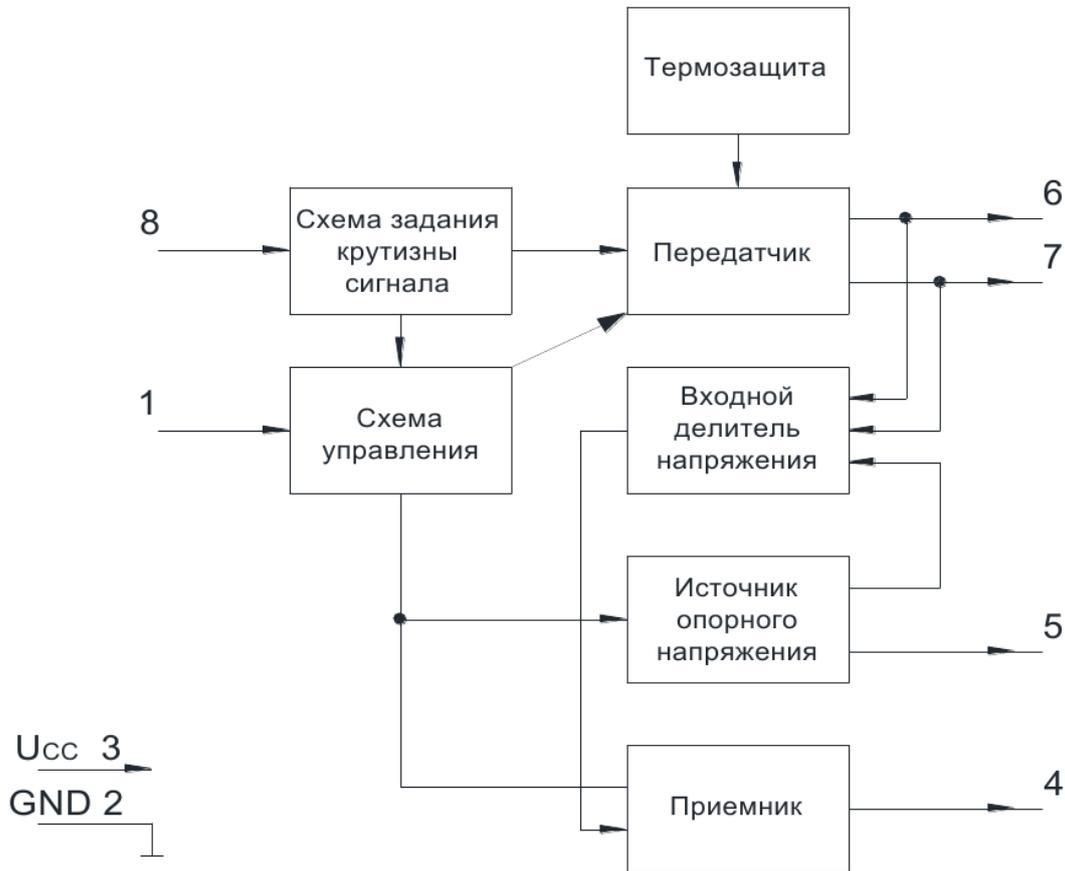


Рисунок 2.44 – Структурная блок-схема микросхемы

CAN передатчик имеет три режима работы: нормальный режим, режим контроля скорости нарастания/спада выходного сигнала и режим «Выключено».

В нормальном режиме работы выходы передатчика переключаются с максимальной возможной скоростью для обеспечения скорости передачи данных до 1 Мбит/с.

Для выбора режима «Контроля скорости» необходимо подключить резистор между входом RRS и потенциалом земли. В этом режиме номинал резистора определяет величину скорости нарастания/спада выходного сигнала, что необходимо для уменьшения уровня электромагнитных помех, а также отражений при неидеально со-

гласованной шине. Таким образом обеспечивается стабильная передача информации со скоростью от 40 до 500 Кбит/с.

Величину подключаемого резистора можно рассчитать по формуле:  $RRS [кОм] = 12000 / \text{Скорость передачи [Кбит/с]}$ .

Таблица 2.22 – Описание выводов

№ вывода	Условное обозначение	Описание
1	TXD	Вход передатчика
2	GND	Общий
3	Ucc	Питание
4	RXD	Выход приемника
5	UREF	Выход источника опорного напряжения
6	CANL	Вход приемника/выход передатчика низкого уровня
7	CANH	Вход приемника/выход передатчика высокого уровня
8	RS	Вход управления режимом работы «нормальный»/«ожидание»/«контроль скорости»

Выход приемника активен во всех режимах работы схемы. Выходной высокий уровень соответствует рецессивному состоянию на линии передачи, а также режиму «Выключено», выходной низкий уровень соответствует доминантному состоянию на линии передачи. Дифференциальный порог переключения приемника около 0,7 В и имеет гистерезис порядка 80 мВ. Допустимый диапазон синфазных напряжений для приемника составляет от минус 10 В до 10 В.

Приемник рассчитан на прием данных со скоростью до 1 Мбит/с. Приемник имеет входной фильтр, что повышает стойкость приемника к дифференциальным помехам.

### Контрольные вопросы к разделу 2.3

1. Интеграция датчиков в бортовую сеть. Классификация датчиков по виду выходного сигнала. Объединение устройств съема и обработки информации в одном блоке.

2. Продукционные правила. Применение нечеткой логики управления.
3. Автомобильные мультимплексные системы передачи информации. Понятие об автомобильных мультимплексных системах.
4. Локальные вычислительные сети. Эталонные модели взаимодействия систем.
5. Протоколы компьютерных сетей. Правила и последовательность обмена данными.
6. Аппаратные драйверы.
7. Физическая передающая среда в ЛВС. Оптоволоконный кабель. Коаксиальный кабель. Одиночный провод. Витая пара.
8. Пропускная способность. Подключение и разветвление.
9. Протоколы высоких уровней.
10. Формат сообщений.
11. Управление сетью.
12. Поддержание корректной работы сети.
13. Обработка ошибок. Контроль конфигурации.
14. Протоколы низкого уровня (шинные). Скорость и расстояние.
15. Электромагнитная совместимость. Шинная топология.
16. Протокол CAN для автомобильных мультимплексных систем. CAN – сеть равноправного доступа. Версии 2.0A и 2.0B.
17. Архитектура протокола CAN.
18. Объектный уровень CAN.
19. Фильтрация сообщений и обработка сообщений и состояний.
20. Транспортный уровень. Синхронизация, арбитраж, доступ к шине, разделение посылок на кадры, определение и передача ошибок.
21. Физический уровень. Передача сигналов, электрические уровни. Скорость передачи ввода/вывода.
22. Формат сообщений CAN. Идентификаторы. Данные.
23. Ограничение распространения ошибок CAN. Программные и аппаратные ошибки. Счетчики ошибок. Повтор сообщений. Частичная работоспособность.

## 2.4. Сеть 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 Volvo Car Corporation. Протокол LIN предназначен для создания дешевых локальных сетей обмена данными на коротких расстояниях. Он служит для передачи входных воздействий, состояний переключателей на панелях управления и так далее, 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 автомобильных подсистем и узлов (таких как дверные замки, стеклоочистители, стеклоподъемники, управление магнитолой и климат-контролем, электролюк и так далее) в единую электронную систему.

LIN-протокол утвержден Европейским Автомобильным Консорциумом как дешевое дополнение к сверхнадежному протоколу CAN. 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 и скорость; область же применения LIN - объединение дешевых узлов, работающих с малыми скоростями передачи информации на коротких дистанциях и сохраняющих при этом универсальность, многофункциональность, а также простоту разработки и отладки. Стандарт LIN включает технические требования на протокол и на среду передачи данных. Как последовательный протокол связи, LIN эффективно поддерживает управление электронными узлами в автомобильных системах с шиной класса «А» (двунаправленный полудуплексный), что подразумевает наличие в системе одного главного (master) и нескольких подчиненных (slave) узлов.

Протокол LIN поддерживает двунаправленную передачу данных по одному проводу длиной до 40 м, используя недорогой микроконтроллер с генератором на RC-цепочке, без 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», обеспечивающая дешевое исполнение, основанное на обычных по-

следовательных интерфейсах UART/SCI; как программная, так и аппаратная возможность реализации, самосинхронизирующаяся тактирующая 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/с. Возможен перевод шины в режим микропотребления «Sleep», когда она 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 низкой стоимости за счет пониженной эффективности. Структура шины представляет собой нечто среднее между I<sup>2</sup>C и RS232. 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.

Концепции интерфейса LIN:

- ◆ дешевый однопроводной интерфейс в соответствии с ISO 9141;
- ◆ скорость в линии до 20 кбит/с, ограниченная для улучшенной электромагнитной совместимости;
- ◆ один ведущий микроконтроллер («master») — множество ведомых («slave»);
- ◆ не требует арбитража;
- ◆ не требует дополнительных аппаратных средств, в основе использования обычный порт UART/SCI;
- ◆ ведомые микроконтроллеры производят самосинхронизацию по ведущему микроконтроллеру сети;
- ◆ гарантированная задержка при передаче сообщений.

Шина LIN применяется в тех приложениях, где требуется управление оборудованием при низкой стоимости подключения к сети. Это позволяет стандартизировать проектирование и значительно сокращает трудозатраты при подключении к сети таких приборов, как датчики и приводы. В спецификации шины LIN 2.0 появилась поддержка режима Plug and Play.

### 2.4.1. Конфигурация шины LIN

На рисунке 2.45 показана типовая конфигурация шины LIN. Для обмена данными используется один сигнальный провод, в каждом узле подтянутый к источнику питания через резистор. В качестве выходного каскада используется транзистор с открытым коллектором. Активным состоянием является низкий уровень на шине данных, в это состояние ее может перевести любой узел. В пассивном состоянии напряжение на шине близко к  $V_{пит}$  (9...18 В). Это означает, что все узлы находятся в неактивном состоянии. Диапазон изменений напряжения питания - 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.

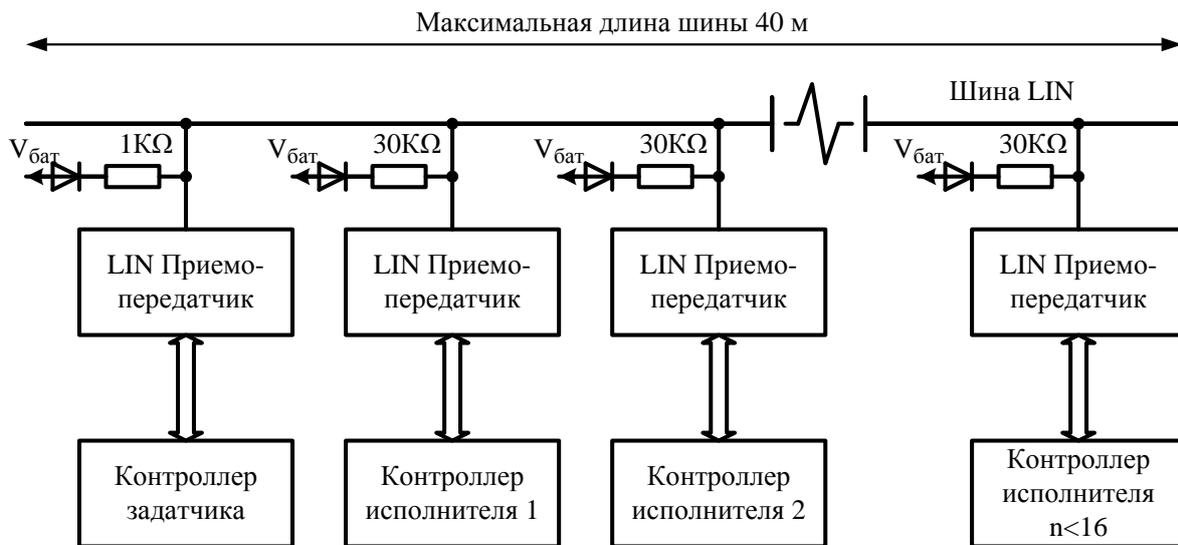


Рисунок 2.45 – Конфигурация LIN

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

Каждый передаваемый байт обрамлен стартовым и стоповым битами, как показано на рисунке 2.46. Передача начинается с младше-

го бита. Стартовый бит равен нулю, а стоповый — пассивному состоянию шины (единице).



Рисунок 2.46 – 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 (рисунок 2.47) начинается задатчиком с посылки сигнала «Synch Break», которое 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).

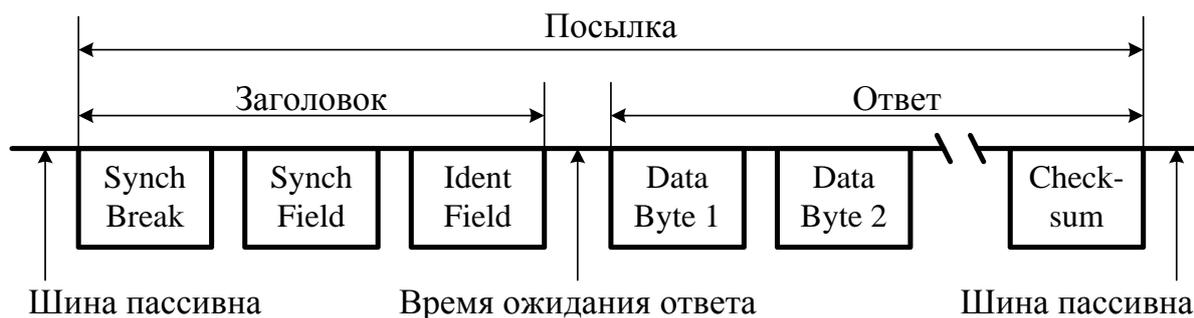


Рисунок 2.47 – Формат посылки

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 (рисунок 2.48).

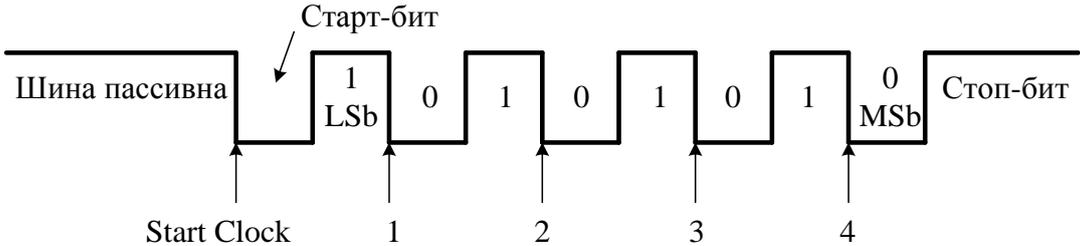


Рисунок 2.48 – Байт синхронизации (SynchField)

В идентификационном поле сообщается информация о том, что же, собственно, последует дальше. Поле идентификации (рисунок 2.49) 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 из них может отвечать 2-мя, 4-мя, 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.

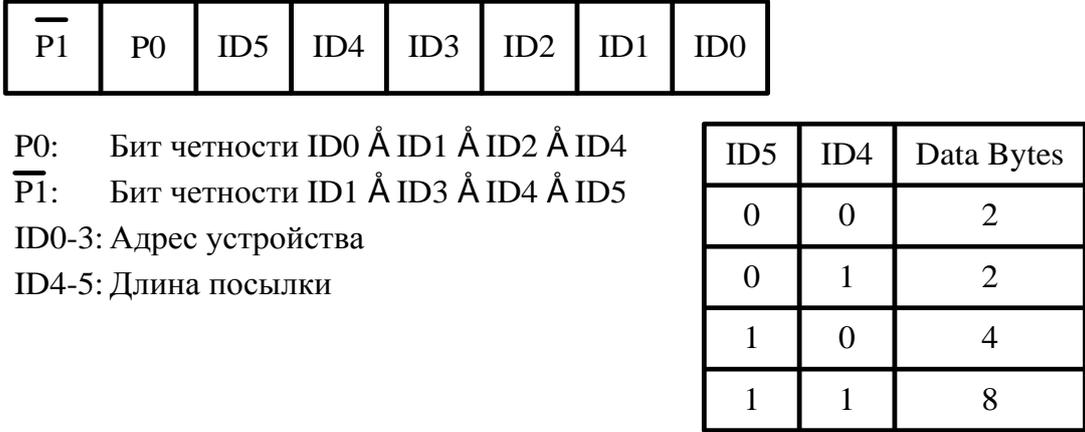


Рисунок 2.49 – Поле идентификации

Задатчик может послать команду всем исполнителям перейти в микромощный режим (Sleep), выставив в поле идентификации байт 0x80 (рисунок 2.50). 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.

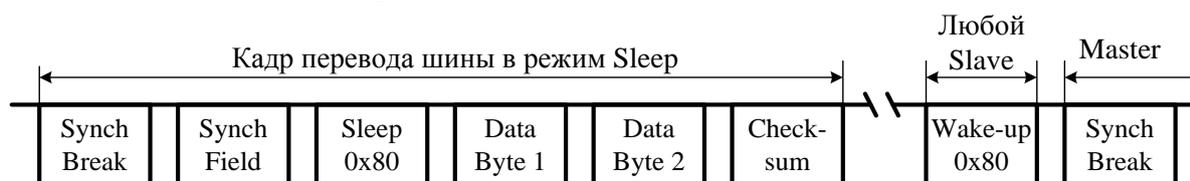


Рисунок 2.50 – Включение и выключение режима «Sleep»

#### 2.4.2. Физический уровень интерфейса

Прием и передача осуществляются по однопроводному интерфейсу в соответствии со стандартом ISO 9141. Схема выходного каскада и диаграмма уровней напряжения в линии для приемника и передатчика показаны на рисунках 2.51 и 2.52 соответственно. Выходные напряжения для передатчика должны соответствовать следующему условию: низкий/высокий уровни передатчика — макс. 20%/мин. 80%  $V_{BAT}$ , где  $V_{BAT}$  — напряжение бортовой сети электропитания.

Входные напряжения для приемника должны соответствовать следующему условию: низкий/высокий уровни приемника — мин. 40%/макс. 60%  $V_{BAT}$ . Скорость нарастания напряжения в линии 1–2 В/мкс.

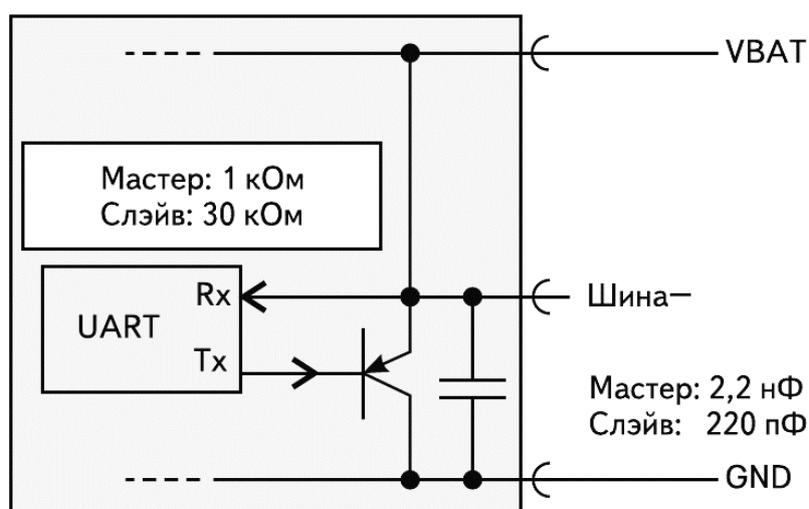


Рисунок 2.51 – Схема выходного каскада трансивера LIN

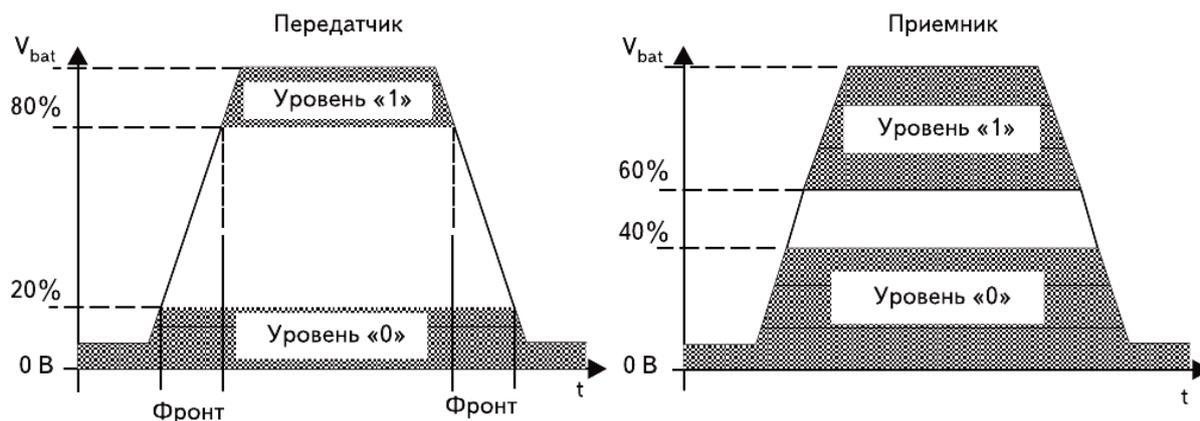


Рисунок 2.52 – Диаграмма уровней напряжений в линии для приемника и передатчика трансивера LIN

В передаче данных по сети участвуют как минимум два абонента сети: «master» и «slave». Соответственно, протокол передачи данных может быть описан как последовательность действий, выполняемая двумя этими абонентами (рисунок 2.53).

«Master» сети выполняет управление обменом данными по шине. Он определяет, кто и какое сообщение будет передавать по шине. Он также ведет обработку ошибок, возникающих при работе шины.

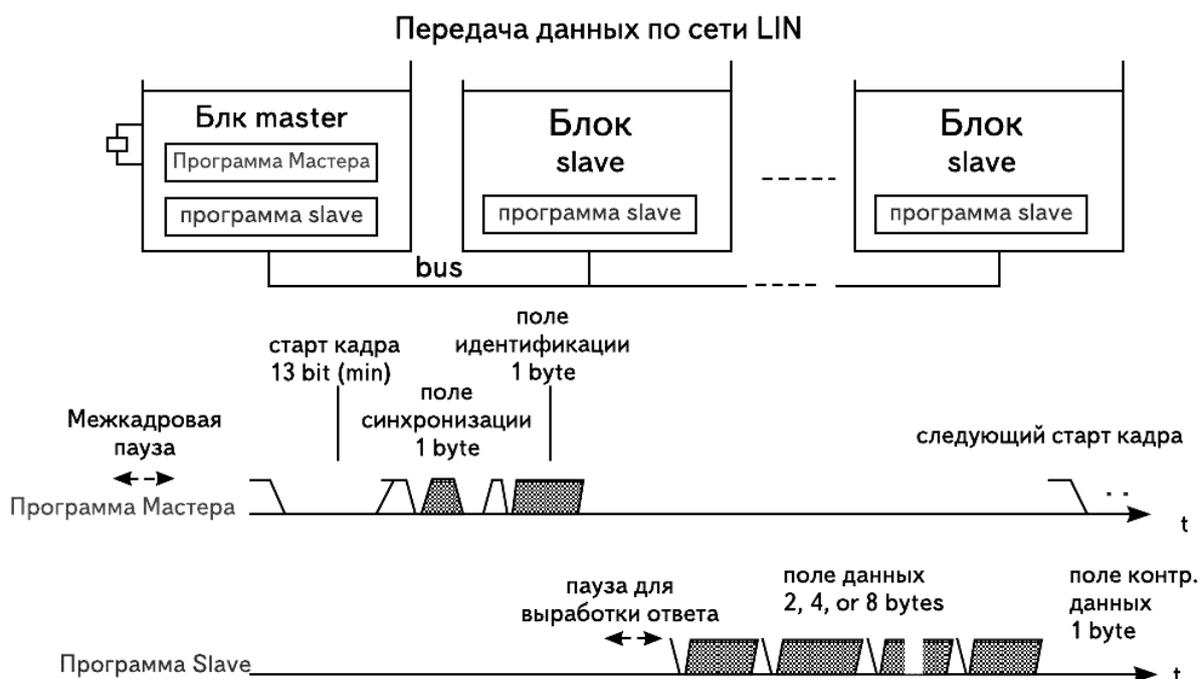


Рисунок 2.53 – Передача данных по сети LIN

Для организации обмена данными по шине «master» выполняет следующие действия:

- посылает синхронизирующую паузу;
- посылает синхронизирующий байт;
- посылает поле идентификатора;
- контролирует байты данных и контрольный байт;
- когда шина неактивна, получает сигнал «WakeUp» от подчиненных узлов, запрашивающих некоторое действие;
- генерирует опорную синхрочастоту для сети (синхрочастота определяется по продолжительности синхронизирующего байта).

Заголовок кадра формируется мастером сети следующим образом: «master» выполняет перерыв сигнала, этим он формирует состояние начала кадра; «master» выдает байт сигнала начала блока — этот байт служит для определения базы времени (определение интервала времени между двумя положительными фронтами); «master» выдает байт «поле идентификатора». Этот байт содержит информацию об отправителе, получателе (или получателях), цели и длине поля данных. Шесть битов этого поля содержат информацию о длине посылки. Возможны посылки, содержащие 2, 4 или 8 байтов данных. Кодирование длины посылки находится в двух старших битах поля идентификатора. Всего возможно 64 идентификатора сообщений. 2 дополнительных бита четности защищают поле идентификатора от ошибок.

Скорость передачи данных составляет 1–20 Кбит/сек и заложена в программное обеспечение блоков управления LIN. Это составляет одну пятую (20%) скорости передачи данных шины CAN-комфорт.

#### *Рецессивный уровень*

Если на шину данных LIN не будет послана телеграмма или рецессивный бит, то на шину данных подается напряжение, практически равное напряжению аккумуляторной батареи.

#### *Доминантный уровень*

Для передачи доминирующего бита по шине данных LIN в передающем блоке управления шина данных замыкается на массу через приемопередатчик (трансивер).

### 2.4.3. Программная реализация

Протокол LIN можно организовать программно на любом микроконтроллере, выпускаемом фирмой Microchip (рисунок 2.54).

Очень удобно для этих целей применять малогабаритные и дешевые PIC16C508 и PIC16C505. На сайте компании [www.microchip.com](http://www.microchip.com) находится пример такой конструкции и приведен исходный текст программы микроконтроллера (Application Note AN729).

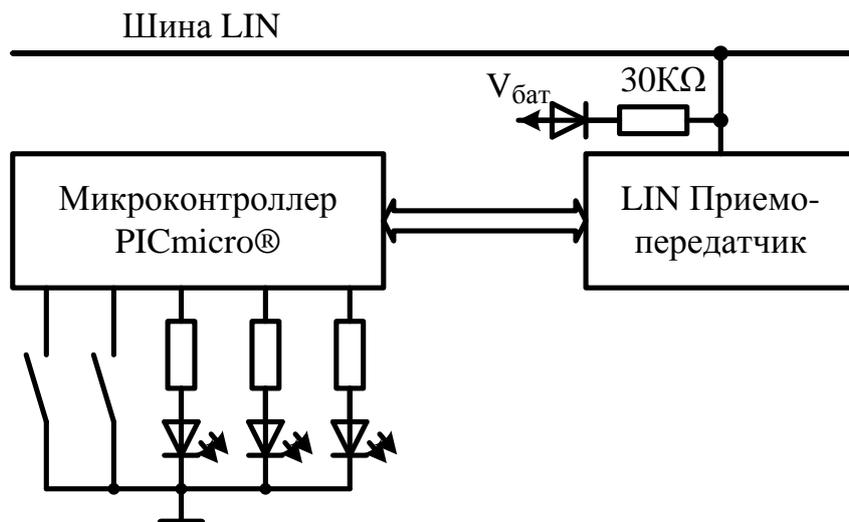


Рисунок 2.54 – Пример реализации Slave устройства

### 2.4.4. Аппаратная реализация

Для удобства проектирования встроенных систем управления для автомобильной электроники, Microchip Technology Inc. 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 классической архитектуре, PIC16C432/433 имеет 2К x 14 бит слов однократно программируемой программной памяти и 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

4-канальных 8-бит АЦП позволяет обрабатывать аналоговые сигналы.

Другими встроенными модулями являются 8-разрядный счетчик/таймер реального времени с 8-разрядным предварительным делителем, выход из "спящего" режима по изменению сигнала на шине и встроенный приемопередатчик LIN, для работы которого необходимо лишь наличие напряжения питания на шине 12 В. Специальные возможности микроконтроллеров поддерживают внутрисхемное программирование (ICSP™), power-on reset (POR), power-up timer (PWRT), oscillator start-up timer (OST), режим пониженного энергопотребления «Sleep», возможность выбора типа задающего генератора и сторожевой таймер (WDT) с отдельным RC-генератором для повышения надежности. PIC16C432 также имеет функцию brown-out reset (BOR).

#### *2.4.4.1. Реализация сетей LIN на микроконтроллерах фирмы NEC*

Микроконтроллеры фирмы NEC Electronics, содержащие новый усовершенствованный LIN UART, работают в сетях LIN как «master» или «slave». Фирма NEC Electronics дополнила ядро UART специальными функциями для того, чтобы облегчить обработку заголовков кадров LIN микроконтроллерами, оптимизированными как по скорости выполнения, так и по объему кода.

LIN UART поддерживает все специфические для LIN скорости передачи и все типы кадров LIN, длительность сигнала синхронизации может устанавливаться программно (стандартный UART) или в специальном регистре (LIN UART). Поскольку прием сообщений ведется кадрами, то при приеме кадра необходимо определить, что именно принимает приемник в текущий момент времени: паузу между кадрами или поле синхронизации. Для этого стандартный UART перестраивается на пониженную тактовую частоту и производится прием данных. Далее, определив, что принята информация, соответствующая полю синхронизации, необходимо снова перестроить UART на номинальную тактовую частоту. В случае применения LIN UART эта процедура значительно упрощается, поскольку LIN UART имеет специальный таймер для определения поля синхронизации. Блок-схема ядра UART приведена на рисунке 2.55.

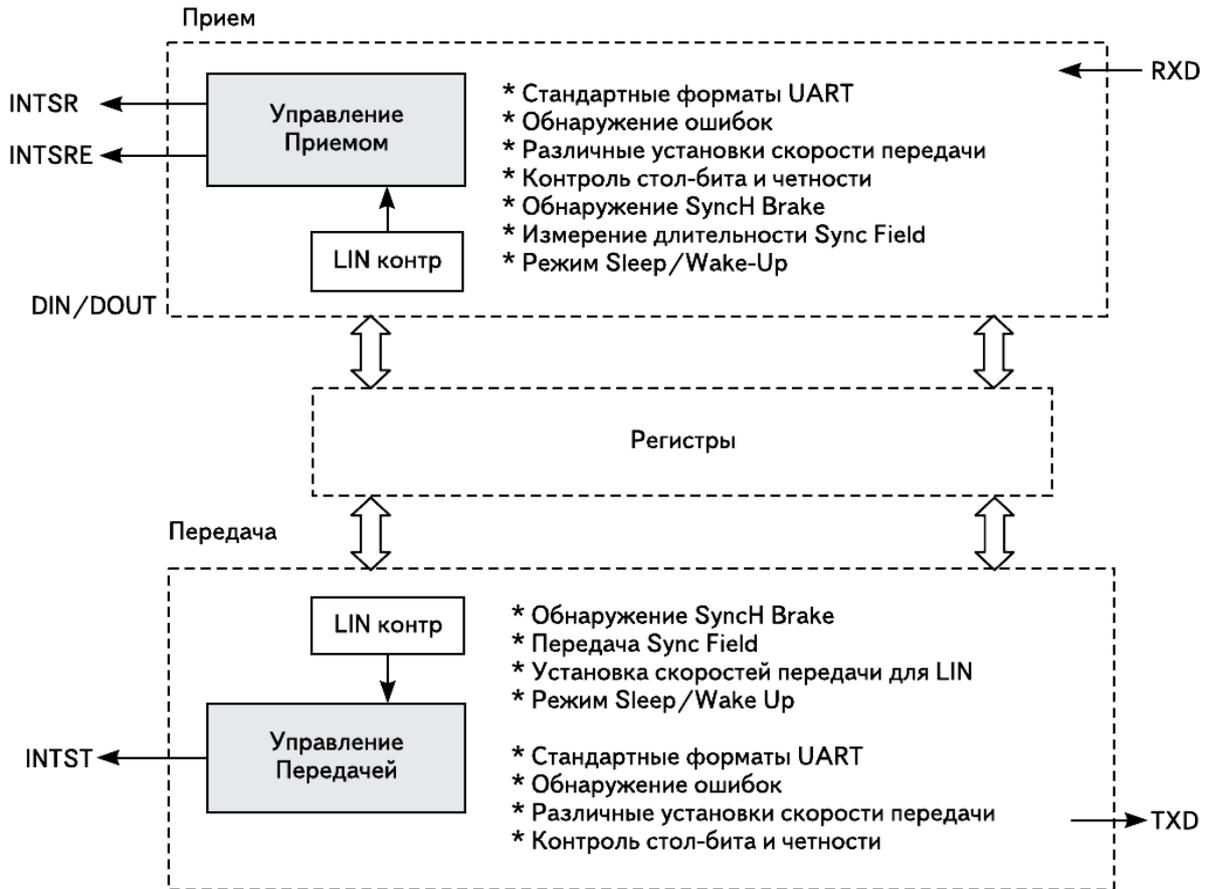


Рисунок 2.55 – Блок схема ядра UART микроконтроллеров 78K

Семейство 8-разрядных микроконтроллеров 78K0 фирмы NEC Electronic используется в сети LIN в основном, в качестве устройств «slave», в то время как 32-разрядное семейство V850 оптимизировано для использования в качестве устройств «master» для создания меж-сетевого контроллера CAN/LIN.

### 2.4.5. Элементная база для реализации сети LIN

#### 2.4.5.1. Микросхема приемо-передатчика интерфейса LIN 5559IH15U.

Микросхема приемопередатчика интерфейса LIN служит для организации интерфейса между микроконтроллером, поддерживающим протокол LIN, и физической линией передачи (рисунок 2.56). Микросхема предназначена для использования в сетях стандарта LIN со скоростями передачи данных от 2,4 Кбит/с до 20,0 Кбит/с.

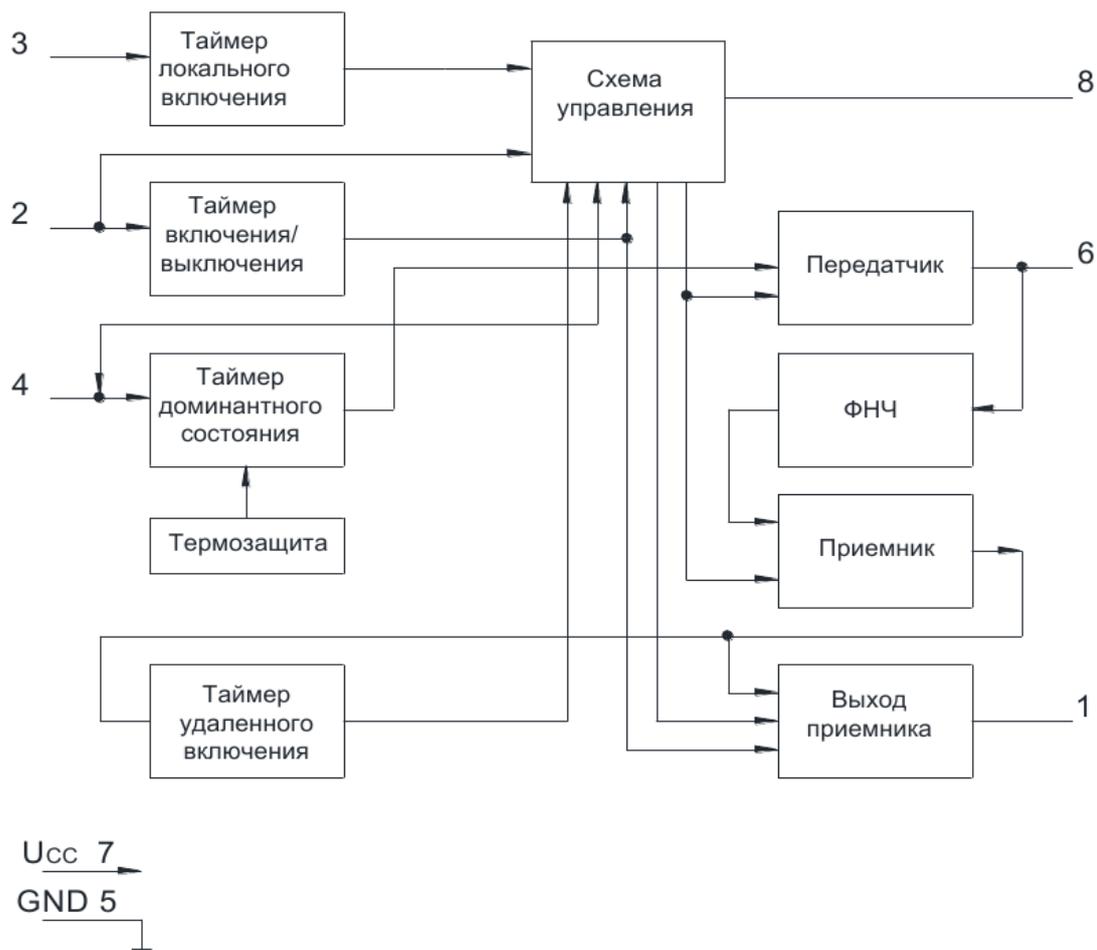


Рисунок 2.56 – Структурная блок-схема микросхемы

Основные области применения: Автомобильные бортовые системы и системы промышленной автоматики с напряжением питания 12В или 24В.

Таблица 2.23 – Описание выводов

Вывод	Условное обозначение	Описание
1	RXD	Выход приемника (открытый сток)
2	NSLP	Вход управления режимом «выключено»
3	NWAKE	Вход локального включения
4	TXD	Вход передатчика
5	GND	Общий
6	LIN	Вход приемника/Выход передатчика
7	Ucc	Питание
8	INH	Выход для управления внешним регулятором напряжения, активный высокий уровень

Основные характеристики микросхемы:

- ↯ Соответствует спецификации LIN 2.1
  - ↯ Напряжение питания от 5,0 до 27,0 В
  - ↯ Передатчик с ограничением скорости изменения выходного сигнала, скорость передачи данных до 20 Кбит/с, пониженный уровень электромагнитного излучения (ЕМЕ)
  - ↯ Низкоскоростной режим работы передатчика со скоростью передачи данных до 10,4 Кбит/с, пониженный уровень электромагнитного излучения
  - ↯ Защита от короткого замыкания и перегрева выхода передатчика, функция ограничения времени доминантного состояния выхода передатчика
  - ↯ Диапазон входных напряжений со стороны линии передачи от минус 27 до 40 В
  - ↯ Низкое потребление в режиме «выключен» с возможностью удаленного и локального включения схемы
  - ↯ Распознавание источника сигнала включения
  - ↯ Отсутствие влияния на линию передачи в режиме «выключено»
  - ↯ Интегрированный согласующий резистор для использования в роли ведомого узла LIN сети
  - ↯ Логические уровни интерфейса с контроллером совместимы с 3,3 В и 5,0 В логическими уровнями;
  - ↯ Рабочий диапазон температур
- |             |                 |
|-------------|-----------------|
| 5559ИН15У   | – 60 ÷ + 125 °С |
| К5559ИН15У  | – 60 ÷ + 125 °С |
| К5559ИН15УК | 0 ÷ + 70 °С     |

Данный приемопередатчик может работать в одном из четырех возможных режимов: нормальный режим, низкоскоростной режим, режим «ожидание» и режим «выключен». В нормальном режиме скорость изменения выходного напряжения передатчика оптимизирована для скорости передачи данных до 20 Кбит/с с низким уровнем электромагнитного излучения (ЕМЕ). В низкоскоростном режиме дополнительное ограничение скорости изменения выходного напряжения передатчика позволяет осуществлять передачу данных со скоростью до 10,4 Кбит/с с еще более низким уровнем электромагнитного излучения. В режиме «ожидание» приемник и передатчик схемы находятся в выключенном состоянии, схема ожидает действия микро-

контроллера. В режиме «выключен» схема характеризуется низким током потребления и отсутствием нагрузки на линию передачи.

Возможные варианты смены режима работы схемы приемопередатчика проиллюстрированы на диаграмме состояний (рисунок 2.57).

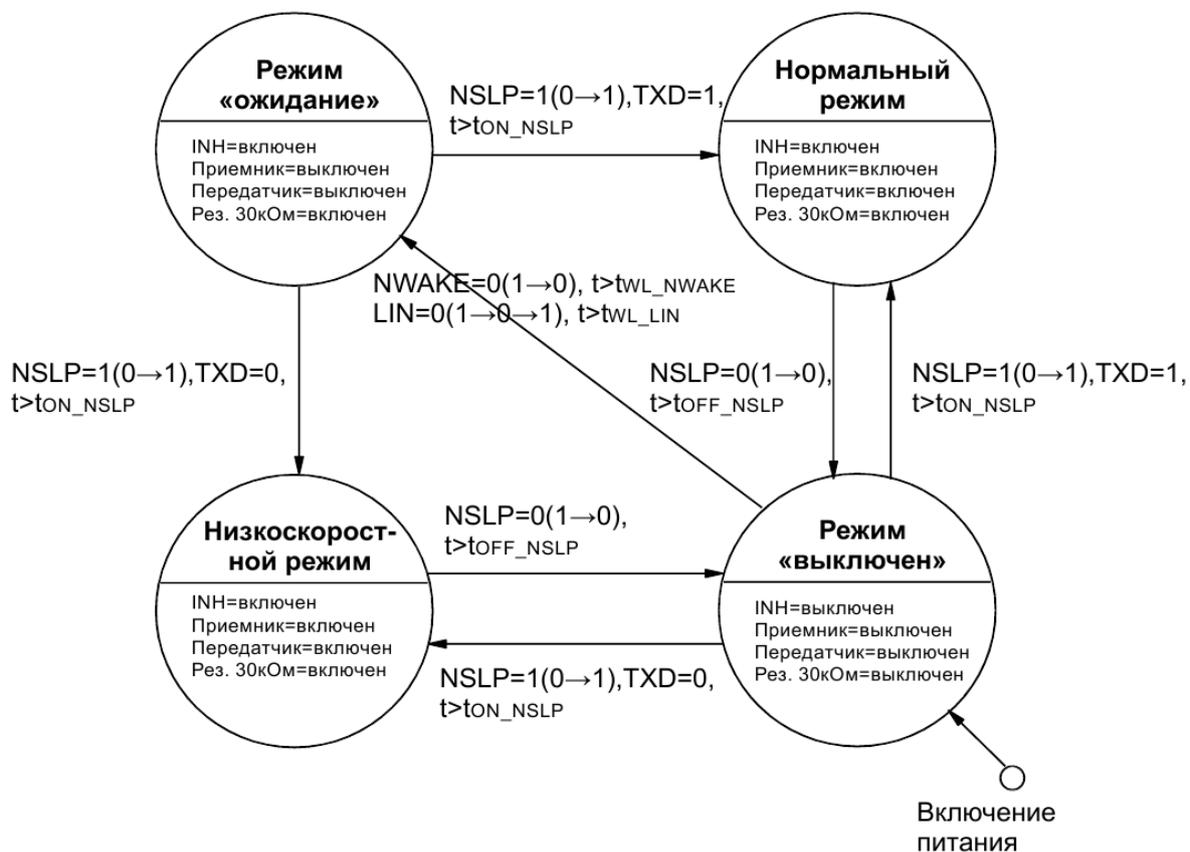


Рисунок 2.57 – Диаграмма состояний приемопередатчика

### Режим «выключен»

При подаче питания, схема первоначально переходит в режим «выключен».

Режим «выключено» характеризуется низким током потребления. В этом режиме полностью выключены LIN приемник и передатчик. Выходы RXD и INH находятся в состоянии с высоким выходным сопротивлением. Внутренний согласующий резистор между выводами BAT и LIN - выключен, остается включенной только «подтяжка» током 5 мкА вывода LIN к напряжению  $U_{BAT}$ .

Из режима «выключен» схема может перейти в режим «ожидание» только при возникновении событий удаленного или локального пробуждения.

В условиях применения приемопередатчика с постоянно включенным микроконтроллером, схему приемопередатчика можно так же перевести из режима «выключен» в нормальный или низкоскоростной режимы и обратно.

Удерживая высокий логический уровень на входе NSLP дольше  $t_{ON\_NSLP}$  можно перевести схему приемопередатчика в нормальный режим, если одновременно на входе TXD присутствует высокий логический уровень. Или можно перевести в низкоскоростной режим, если одновременно на входе TXD удерживается низкий логический уровень.

Если схема находится в нормальном или низкоскоростном режиме работы, ее можно перевести в режим «выключен» удерживая низкий логический уровень на входе NSLP дольше  $t_{OFF\_NSLP}$ .

#### *Режим «ожидание»*

В режим «ожидание» схема приемопередатчика может перейти только из режима «выключен» и только при возникновении событий удаленного или локального пробуждения.

В режиме «ожидание» LIN приемник и передатчик остаются в выключенном состоянии. Внутренний согласующий резистор подключается между выводами BAT и LIN. Потенциал выхода INH подтягивается к BAT.

В режиме «ожидание» на выходе RXD формируется низкий логический уровень, который отображает состояние внутреннего флага «запрос на прерывание» схемы. Это сигнал можно использовать для прерывания работы микроконтроллера. Состояние выхода TXD соответствует флагу «источник сигнала пробуждения». Низкий логический уровень на входе TXD (включается ключ подтягивающий вход к земле) сигнализирует о локальном пробуждении, высокий логический уровень (на входе остается подтягивающий резистор 330 кОм) сообщает об удаленном пробуждении схемы приемопередатчика.

Флаги «запрос на прерывание» и «источник сигнала пробуждения» сбрасываются сразу после подачи высокого логического уровня на вход NSLP независимо от того перейдет схема в нормальный/низкоскоростной режим или нет.

Из режима «ожидание» схему приемопередатчика можно перевести в нормальный или низкоскоростной режимы работы. Удерживая высокий логический уровень на входе NSLP дольше  $t_{ON\_NSLP}$  мож-

но перевести схему приемопередатчика в нормальный режим, если одновременно на входе TXD присутствует высокий логический уровень. Или можно перевести в низкоскоростной режим, если одновременно на входе TXD удерживается низкий логический уровень.

#### *Нормальный режим*

В нормальном режиме внутренний согласующий резистор подключается между выводами VAT и LIN. Потенциал выхода INH подтягивается к VAT.

В этом режиме приемопередатчик LIN обеспечивает микроконтроллер интерфейсом к линии передачи сети LIN через выводы TXD и RXD. Низкий логический уровень на входе TXD соответствует доминантному состоянию на выходе передатчика LIN. При высоком логическом уровне на входе TXD состояние на выводе LIN определяется состоянием линии передачи. Доминантное состояние на выводе LIN отражается как низкий логический уровень на выходе RXD (выход с открытым стоком).

В нормальный режим работы схему приемопередатчика можно перевести удерживая высокий логический уровень на входе NSLP дольше  $t_{ON\_NSLP}$ , если одновременно на входе TXD присутствует высокий логический уровень. Если схема находится в нормальном режиме работы, ее можно перевести в режим «выключен» удерживая низкий логический уровень на входе NSLP дольше  $t_{OFF\_NSLP}$ .

#### *Низкоскоростной режим*

Низкоскоростной режим идентичен нормальному режиму, за исключением того, что скорость изменения напряжения на выходе передатчика ограничена и оптимальна для скоростей передачи данных до 10,4 Кбит/с. В этом режиме еще больше снижается уровень электромагнитного излучения.

В низкоскоростной режим работы схему приемопередатчика можно перевести удерживая высокий логический уровень на входе NSLP дольше  $t_{ON\_NSLP}$ , если одновременно на входе TXD присутствует низкий логический уровень. Если схема находится в низкоскоростном режиме работы, ее можно перевести в режим «выключен» удерживая низкий логический уровень на входе NSLP дольше  $t_{OFF\_NSLP}$ .

#### *Локальное и удаленное пробуждение*

Схема приемопередатчика может определить источник сигнала, который привел к включению схемы в режим «ожидание».

Локальное пробуждение происходит в случае, когда вход NWAKE удерживается в состоянии с низким логическим уровнем дольше  $t_{WL\_NWAKE}$ . Вход NWAKE имеет внутреннюю «подтяжку» к потенциалу ВАТ током 10 мкА. Если функция локального пробуждения не требуется, вход NWAKE следует соединить с ВАТ. Для улучшения стойкости к электромагнитным помехам лучше всего вход NWAKE соединить с ВАТ через резистор 5 кОм.

Удаленное пробуждение происходит в случае, когда на вход приемника LIN поступает последовательность рецессивное-доминантное-рецессивное состояние. Длительность доминантного состояния должна быть дольше  $t_{WL\_LIN}$ . Переход приемопередатчика в режим «ожидание» происходит по фронту входного сигнала доминантное-рецессивное состояние (рисунок 2.58).

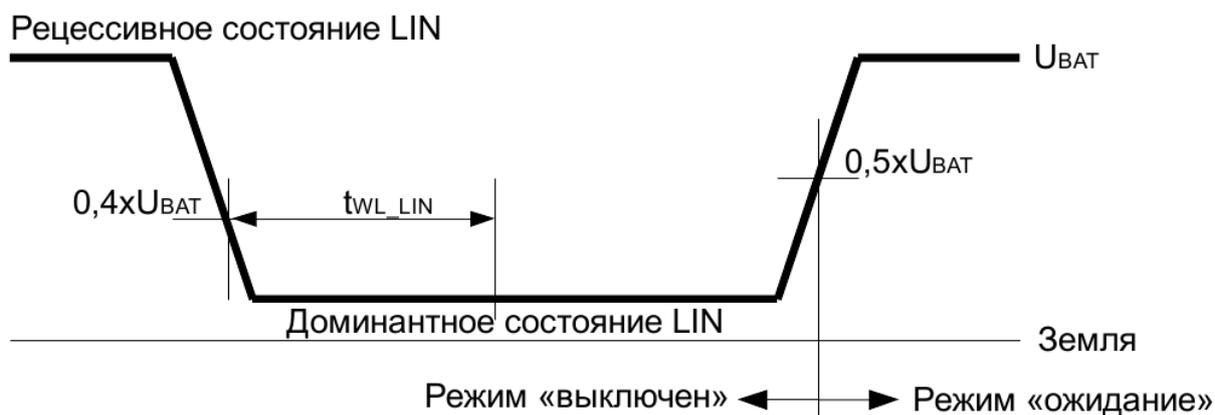


Рисунок 2.58 – Локальное и удаленное пробуждение

#### *Определение источника сигнала включения*

После удаленного или локального пробуждения схема приемопередатчика переходит в режим «ожидание». В режиме «ожидание» на выходе RXD выставляется низкий логический уровень, который отображает состояние внутреннего флага «запрос на прерывание» схемы.

Состояние выхода TXD соответствует флагу «источник сигнала пробуждения». Низкий логический уровень на входе TXD (включается ключ подтягивающий вход к земле) сигнализирует о локальном пробуждении, высокий логический уровень (на входе остается подтягивающий резистор 330 кОм) сообщает об удаленном пробуждении схемы приемопередатчика. Для того, что бы считать состояние флага

«источник сигнала пробуждения» микроконтроллером необходимо использовать внешний резистор, подключенный между выводом TXD и питанием микроконтроллера.

Флаги «запрос на прерывание» и «источник сигнала пробуждения» сбрасываются, сразу после подачи высокого логического уровня на вход NSLP независимо от того перейдет схема в нормальный/низкоскоростной режим или нет.

#### *Защита от сбоев*

Типовая схема включения микросхемы показана на рисунке 2.59. Схема приемопередатчика содержит схемы «подтяжки» входов TXD, NSLP, NWAKE, LIN, наличие которых исключает возможность возникновения сбойных ситуаций в случае отсутствия подключения этих входов.

Выход передатчика имеет защиту от короткого замыкания на случай возникновения замыкания на потенциал ВАТ. В схеме передатчика реализовано 2 механизма защиты: ограничение выходного тока и защита от перегрева. Схема защиты от перегрева срабатывает при температуре кристалла около 165°C и переводит схему передатчика в рецессивное состояние. Гистерезис порога включения порядка 15°C, но работа передатчика возобновляется только после прихода фронта сигнала по входу TXD.

Если на вход TXD подан логический сигнал низкого уровня, то передатчик формирует на линии передачи доминантное состояние. Тем самым передатчик блокирует обмен информацией по линии передачи для других устройств. Во избежание захвата линии передачи сбойным узлом, если доминантное состояние на выходе передатчика длится дольше  $t_{OFF\_DOM}$ , передатчик переводится в рецессивное состояние внутренним таймером. Работа передатчика возобновляется после прихода фронта сигнала по входу TXD.

В случае обрыва цепей GND или ВАТ, когда схема приемопередатчика утрачивает питание, вывод LIN находится в состоянии с высоким выходным сопротивлением и не нагружает линию передачи. Кроме того, в состоянии с высоким выходным сопротивлением так же находится выход RXD, таким образом, предотвращается возникновение пути протекания тока от подключенного к этому выводу микроконтроллера.

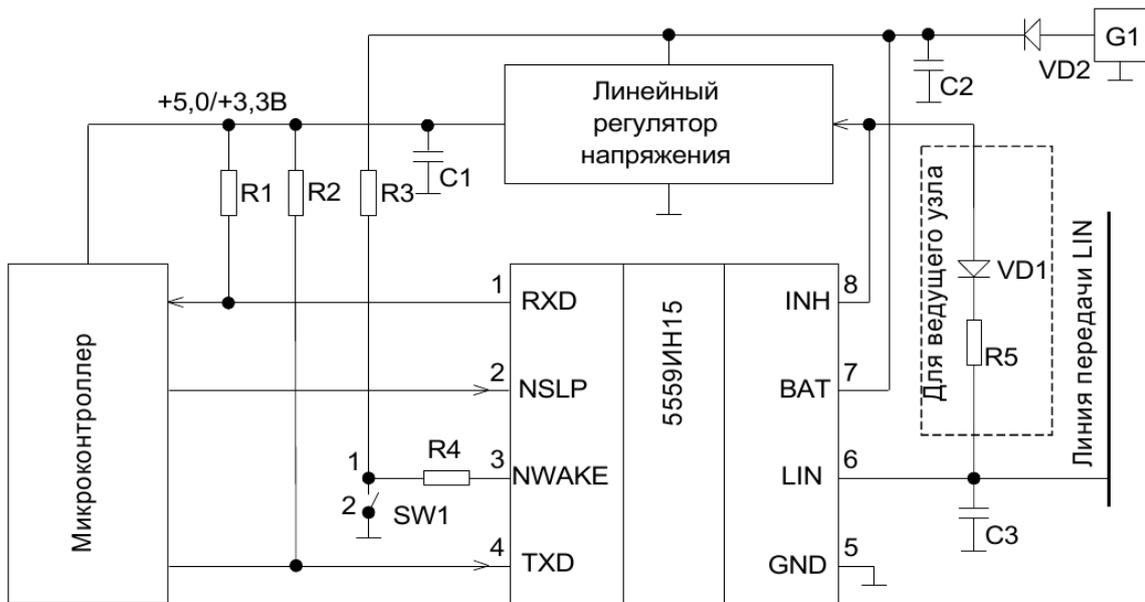


Рисунок 2.59 – Типовая схема включения микросхемы 5559IH15 - включаемая микросхема (5559IH15У)  
 G1, - источник постоянного напряжения,  $U_{CC}=(7,0\dots27,0)$  В;  
 SW1 - переключатель;  
 C1...C3 - конденсаторы, C1\*= 100 нФ ± 10 %;  
 C2= 100 нФ ± 10 %;  
 C3= 1 нФ - для ведущего узла;  
 C3= 220пФ - для ведомого узла;  
 R1...R5 - резисторы, R1= R2= 5 кОм ± 5 %;  
 R3= 10 кОм ± 5 %;  
 R4= 33 кОм ± 5 %;  
 R5= 1 кОм ± 5 %.  
 VD1, VD2 -диоды; VD1\*\*= VD2\*\*=2Д510А (КД243Г)

Развязывающий конденсатор C2 необходимо располагать как можно ближе к микросхеме.

Резистор R2 необходим для считывания микроконтроллером флага «источник сигнала включения». Если данная функция не нужна резистор R2 можно исключить из схемы включения.

Резистор R5 и диод VD1 подключаются только на стороне ведущего узла LIN сети. Номинал конденсатора C4 также зависит (см. рисунок) от того ведущий или ведомый узел LIN сети.

В условиях применения приемопередатчика с постоянно включенным микроконтроллером, вход NWAKE рекомендуется соединить с выводом BAT. Для улучшения стойкости к электромагнитным помехам следует соединить вход NWAKE с выводом BAT через резистор 5 кОм.

### 2.4.5.2. Микросхема однокристалльной микро-ЭВМ с ЭСППЗУ EEPROM-типа с CAN и LIN интерфейсами.

Микросхема (рисунки 2.60, 2.61) предназначена для широкого применения в аппаратуре общего назначения, автомобильной технике, железнодорожном, водном и воздушном транспорте в качестве периферийного контроллера организующего сбор и первичную обработку и передачу информации и сигналов управления по интерфейсам CAN и LIN. Может использоваться для организации малопроизводительных вычислительных систем и в качестве устройства совмещения различных типов интерфейсов, так же предназначена для обработки информации от группы датчиков.

Основные области применения:

- ↯ интеллектуальные датчики;
- ↯ автомобильная техника;
- ↯ промышленные системы управления;
- ↯ телекоммуникационное оборудование;
- ↯ системы безопасности;
- ↯ измерительное оборудование.

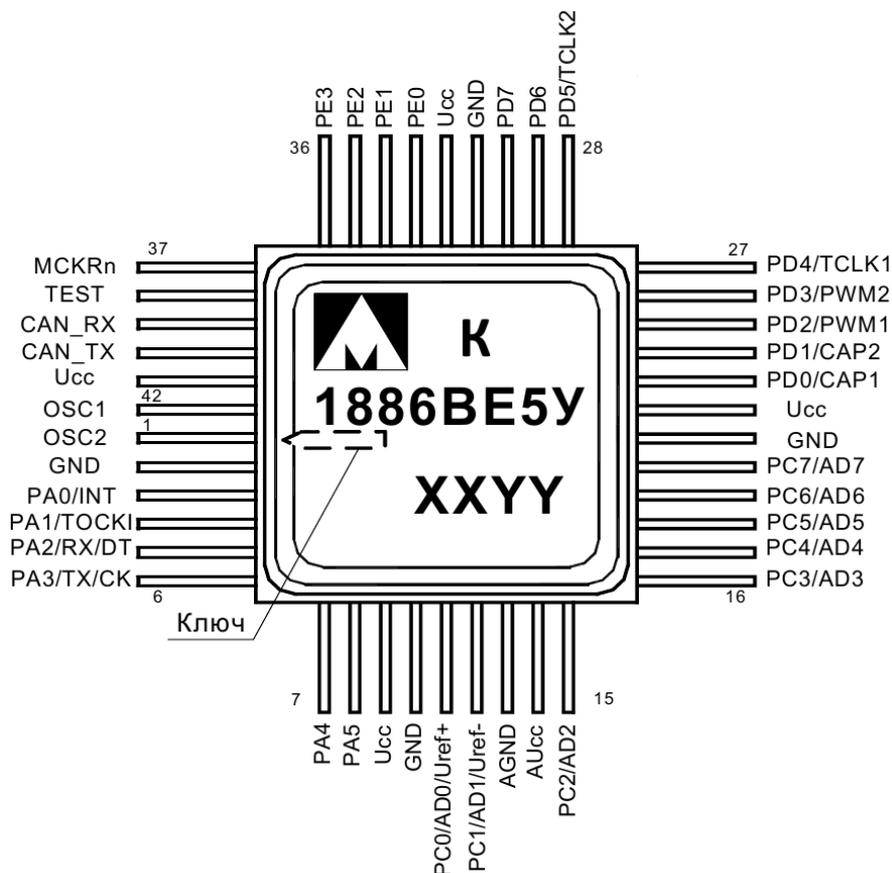


Рисунок 2.60 – Микросхема однокристалльной микро-ЭВМ

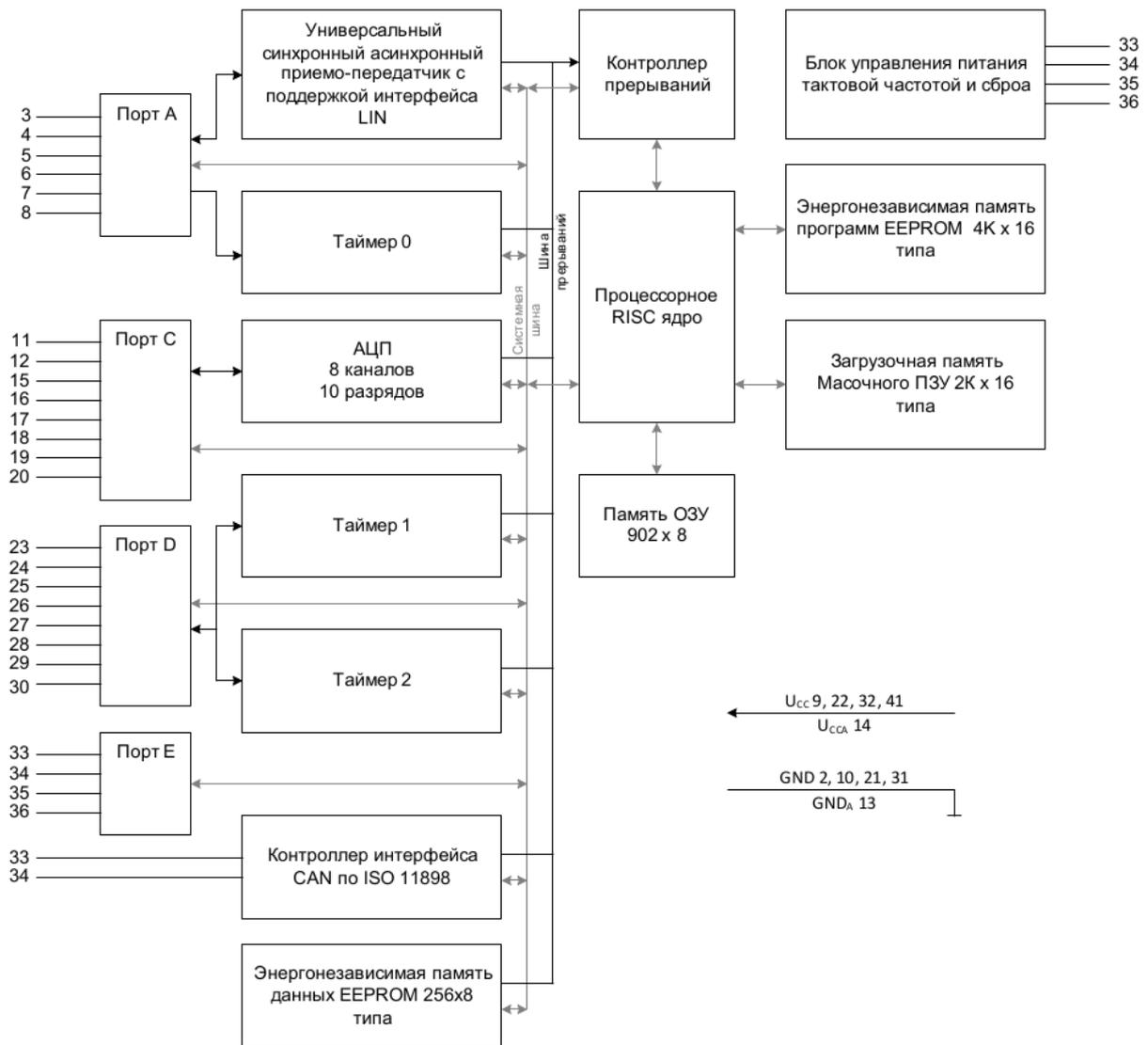


Рисунок 2.61 – Структурная блок-схема

### Основные параметры микросхемы

- ↻ Тактовая частота до 35 МГц;
- ↻ Минимальная длительность цикла выполнения команды 115 нс;
- ↻ 58 однословных инструкций (коды инструкций 16-ти разрядные);
- ↻ 8-ми разрядное АЛУ;
- ↻ Аппаратно реализованная инструкция 8 x 8 битного умножения;
- ↻ Поддержка прямого, косвенного и относительного режимов адресации;
- ↻ Наличие инструкций одновременно работающих с двумя регистрами;

↯ Контроллер интерфейса CAN2.0b удовлетворяющий стандарту ISO 11898;

↯ Аппаратная поддержка интерфейса LIN;

↯ Функция самостоятельной модификации кода программ;

↯ 10 разрядный АЦП;

↯ Диапазон напряжения питания 4,5 – 5,5 В;

↯ Температурный диапазон:

1886BE5AY – 60 ÷ + 125 °C

1886BE5BY – 60 ÷ + 125 °C

K1886BE5AY – 60 ÷ + 125 °C

K1886BE5BY – 60 ÷ + 125 °C

K1886BE5BY 0 ÷ + 70 °C

K1886BE5GY 0 ÷ + 70 °C

Особенности ядра микроконтроллера:

↯ 58 однословных инструкций.

↯ Все инструкции выполняются за один цикл, за исключением инструкций переходов и инструкций чтения/записи таблиц выполняемых за два цикла.

↯ Скорость работы: тактовая частота до 35 МГц, минимальная длительность цикла 115 нс.

↯ 8 x 8 битный аппаратный умножитель, за цикл.

↯ Поддержка прерываний.

↯ 16 словный аппаратный стек.

↯ Прямая, косвенная и относительная модель адресации.

Внутренняя типа EEPROM память программ размером 4Кx16

Краткое описание периферии:

↯ До 26 пользовательских выводов.

↯ 16-ти битный таймер/счетчик с 8-ми битным предварительным делителем.

↯ Таймер 1 и 2 (ШИМ/Захват/Таймеры).

↯ Один универсальный синхронный асинхронный приемник передатчик (USART) с программируемой скоростью передачи и поддержкой режима LIN.

↯ Универсальный контроллер CAN 2.0b интерфейса с шестью буферами сообщений, со скоростью передачи до 1 Мбит/с, соответствующий ISO 11898-1.

↯ Универсальный контроллер внутренней памяти типа CMOS EEPROM размером 256x8.

↻ 8 канальное 10 разрядное АЦП последовательного приближения.

Специализированные особенности:

- ↻ Сброс по снижению питания.
- ↻ Отложенный запуск по подаче питания и тактовой частоты.
- ↻ Сторожевой таймер.
- ↻ Защищенный режим.
- ↻ Режим энергосбережения (SLEEP).
- ↻ Уровень напряжения питания микроконтроллера от 4,5В до 5,5В.

Таблица 2.24 – Описание выводов

Обозначение вывода	Номер вывода	Тип вывода	Назначение вывода
OSC1	42	вход	Вход для сигналов тактовой синхронизации, от внешнего кварцевого генератора или резонатора
OSC2	1	выход	Выход обратной связи для внешнего кварцевого резонатора
Порта А			Дополнительное назначение выводов:
PA0/INT	3	вход	Вывод порта А, разряд 0/Вход внешнего прерывания. Только входной контакт
PA1/T0CLK	4	вход	Вывод порта А, разряд 1/Вход тактового сигнала для таймера 0 и внешнего прерывания (T0CKIF). Только входной контакт
PA2/RX/DT	5	вход/выход	Вывод порта А, разряд 2/Вход асинхронного приемника/Вход (выход) линии данных в синхронном режиме USART
PA3/TX/CK	6	вход/выход	Вывод порта А, разряд 3/Выход асинхронного передатчика/Вход (выход) тактовых импульсов в синхронном режиме USART.
PA4	7	вход/выход	Вывод порта А, разряд 4
PA5	8	вход/выход	Вывод порта А, разряд 5
Порт С – параллельный двунаправленный порт ввода/вывода совмещен с АЦП			Дополнительное назначение выводов:

Продолжение табл. 2.24

Обозначение вывода	Номер вывода	Тип вывода	Назначение вывода
PC0/ADC0/Vref+	11	вход/выход	Двунаправленный порт общего назначения разряд 0/Аналоговый канал 0 АЦП/ Вход верхнего опорного напряжения АЦП
PC1/ADC1/Vref-	12	вход/выход	Двунаправленный порт общего назначения разряд 1/Аналоговый канал 1 АЦП/ Вход нижнего опорного напряжения АЦП
PC2/ADC2	15	вход/выход	Двунаправленный порт общего назначения разряд 2/Аналоговый канал 2 АЦП/
PC3/ADC3	16	вход/выход	Двунаправленный порт общего назначения разряд 3/Аналоговый канал 3 АЦП/
PC4/ADC4	17	вход/выход	Двунаправленный порт общего назначения разряд 4/Аналоговый канал 4 АЦП/
PC5/ADC5	18	вход/выход	Двунаправленный порт общего назначения разряд 5/Аналоговый канал 5 АЦП/
PC6/ADC6	19	вход/выход	Двунаправленный порт общего назначения разряд 6/Аналоговый канал 6 АЦП/
PC7/ADC7	20	вход/выход	Двунаправленный порт общего назначения разряд 7/Аналоговый канал 7 АЦП/
Порта D – параллельный двунаправленный порт ввода/вывода совмещен с Timer12			Дополнительное назначение выводов:
PD0/CAP1	23	вход/выход	Двунаправленный порт общего назначения, разряд 0/Вход схемы захвата 1
PD1/CAP2	24	вход/выход	Двунаправленный порт общего назначения, разряд 1/Вход схемы захвата 2
PD2/PWM1	25	вход/выход	Двунаправленный порт общего назначения, разряд 2/Выход схемы ШИМ 1

Окончание табл. 2.24

Обозначение вывода	Номер вывода	Тип вывода	Назначение вывода
PD3/PWM2	26	вход/выход	Двунаправленный порт общего назначения, разряд 3/Выход схемы ШИМ 2
PD4/T1CLK	27	вход/выход	Двунаправленный порт общего назначения, разряд 4/Вход внешней тактовой частоты Timer 1
PD5/T2CLK	28	вход/выход	Двунаправленный порт общего назначения, разряд 5/Вход внешней тактовой частоты Timer 2
PD6	29	вход/выход	Двунаправленный порт общего назначения, разряд 6
PD7	30	вход/выход	Двунаправленный порт общего назначения, разряд 7
Порта E – параллельный двунаправленный порт ввода/вывода			Дополнительное назначение выводов:
PE0	33	вход/выход	Двунаправленный порт общего назначения, разряд 0
PE1	34	вход/выход	Двунаправленный порт общего назначения, разряд 1
PE2	35	вход/выход	Двунаправленный порт общего назначения, разряд 2
PE3	36	вход/выход	Двунаправленный порт общего назначения, разряд 3
Интерфейс CAN			
CAN_TX	40	выход	Выход контроллера интерфейса CAN
CAN_RX	39	вход	Вход контроллера интерфейса CAN
Питание и управление			
UCC	9, 22, 32, 41	напряжение питания	Питание ядра микроконтроллера кристалла.
AUCC	14	напряжение питания	Питание АЦП микроконтроллера кристалла.
GND	2, 10, 21, 31	общий	Общий.
AGND	13	общий	Общий АЦП.
TEST	38	вход	Вывод используемый при тестировании микросхемы.
MCLRn/Upp	37	вход	Вход внешнего сброса кристалла.

Микроконтроллер содержит один модуль синхронно-асинхронного приемопередатчика с поддержкой LIN интерфейса USART/LIN. Универсальный синхронно-асинхронный приемопередатчик может работать в следующих режимах:

- ↔ асинхронный (полный дуплекс);
- ↔ асинхронный (полный дуплекс) с автоматическим приемом LIN заголовка;
- ↔ синхронный ведущий (полудуплекс);
- ↔ синхронный ведомый (полудуплекс).

Бит SPEN (RCSTA1<7>) должен быть установлен, чтобы выводы PA2/RX/DT и PA3/TX/CK сконфигурировались как выводы последовательного интерфейса. Модуль USART/LIN будет управлять направлением выводов PA2/RX/DT и PA3/TX/CK, в зависимости от состояния битов конфигурации в регистрах RCSTA и TXSTA. Следующие биты контролируют направление выводов: SPEN, TXEN, SREN, CREN, CSRC. При установке бита LINEN в асинхронном режиме работы, модуль автоматически принимает заголовок пакета (поля BREAK и SYNCH) с вычислением скорости передачи.

## **Контрольные вопросы к разделу 2.4**

1. Каковы технические требования протокола LIN?
2. В чем суть концепции «single-master/multi-slave»?
3. Дать описание типовой конфигурации шины LIN.
4. Каким образом осуществляются прием и передача данных по шине LIN?
5. Как программно реализовать протокол LIN (на примере PIC-контроллера)?
6. В чем особенности аппаратной реализации шины LIN на микроконтроллерах?

## **2.5. FlexRay**

В 1999 году компания BMW совместно с фирмой Daimler Chrysler и фирмами изготовителями полупроводникового оборудования Freescale (в то время Motorola) и Philips основала консорциум Flex Ray с целью разработки коммуникационной технологии нового типа. В дальнейшем в консорциум вошли Bosch и General Motors. С 2002 к

настоящему моменту его членами стали также Ford Motor Company, Mazda, Elmos и Siemens VDO. На сегодняшний день почти все ведущие мировые производители автомобилей и поставщики комплектующих являются членами консорциума FlexRay.

FlexRay—довольно новая коммуникационная система, задача которой состоит в обеспечении надежной и высокоэффективной передачи данных в реальном времени между электрическими и мехатронными компонентами для сетевого взаимодействия не только уже существующих, но и будущих инновационных функций автомобиля. Создатели коммуникационной системы FlexRay учли растущие технические требования к системам, предназначенным для объединения в сеть блоков управления на транспортном средстве, а также опыт, согласно которому при разработке инфраструктурных систем следует стремиться к решениям, предполагающим возможность дальнейшего расширения и стандартизации. В новой системе используется протокол, обеспечивающий высокую скорость передачи данных для отдельных систем автомобиля в реальном времени (рисунок 2.62).

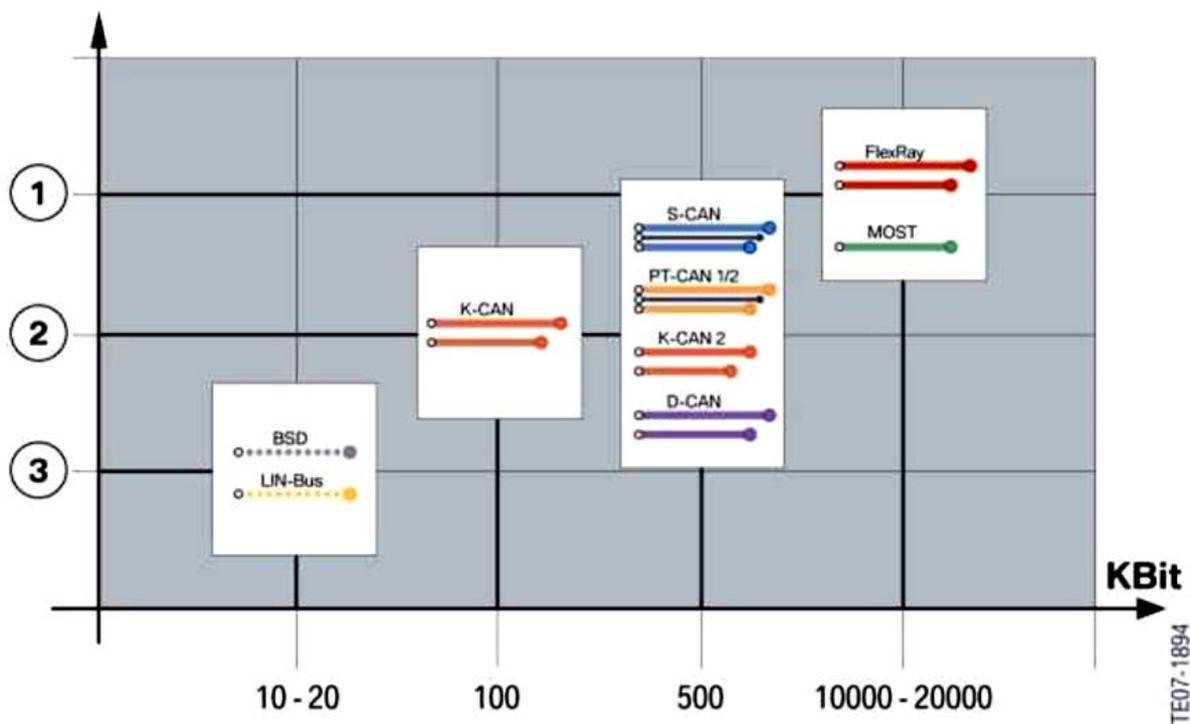


Рисунок 2.62 – Скорости шинных систем:

1 - передача в реальном времени, детерминированность (строгая определенность) и резервирование (многоканальная передача); 2 - передача условно в реальном времени - достаточно для регулировки; 3 - данные не передаются в реальном времени

Скорость передачи данных по шине FlexRay, составляющая макс. 10 Мбит/с, значительно выше по сравнению с аналогичным параметром шин передачи данных, которые использовались прежде на автомобилях в области кузова и трансмиссии/ходовой части. Данная скорость передачи данных до сих пор обеспечивалась только при помощи оптоволоконных кабелей. FlexRay имеет более значительную ширину полосы пропускания, обеспечивает детерминированную (строго определенную) передачу данных и устойчива к отказам конфигурации, т. е. даже после выхода из строя отдельных компонентов обеспечивается надежная работа остальной части коммуникационной системы.

Среди основных преимуществ шины FlexRay следует выделить:

- большая ширина полосы пропускания (10 Мбит/с по сравнению с 0,5 Мбит/с шины CAN);
- детерминированная (в реальном времени) передача данных;
- надежный обмен данными;
- возможность интеграции системы.

Шинная система FlexRay на F01/F02 используется для объединения в сеть систем регулировки динамики движения и системы управления двигателем на транспортном средстве. Центральный модуль межсетевых обмена обеспечивает связь между различными шинными системами и FlexRay. Различные варианты топологии шины FlexRay показаны на рисунках 2.63, 2.64.

На рисунках приняты следующие обозначения: AL – активное рулевое управление; BD – шинный драйвер; DME – цифровая электронная система управления двигателем; DSC – система динамического контроля стабильности; EDCSHL – спутник электронной системы регулирования жесткости амортизаторов, левый задний; EDCSHR – спутник электронной системы регулирования жесткости амортизаторов, правый задний; EDCSVL – спутник электронной системы регулирования жесткости амортизаторов, левый передний; EDCSVV – спутник электронной системы регулирования жесткости амортизаторов, правый передний; HSR – регулировка угла бокового увода заднего моста; ICM – встроенное управление ходовой частью; SZL – коммутационный центр в рулевой колонке; VDM – система управления вертикальной динамикой; ZGM – центральный модуль межсетевых обмена

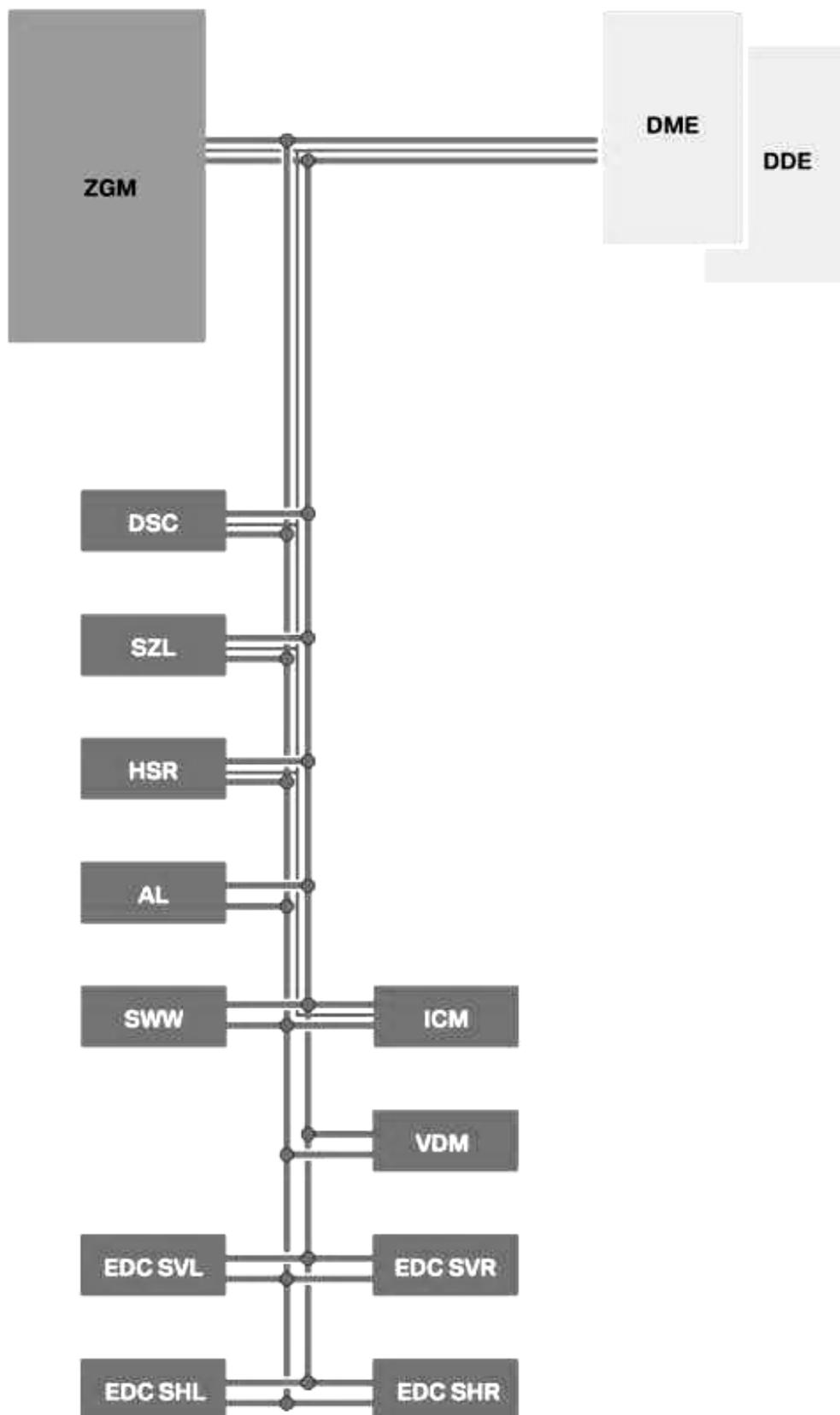
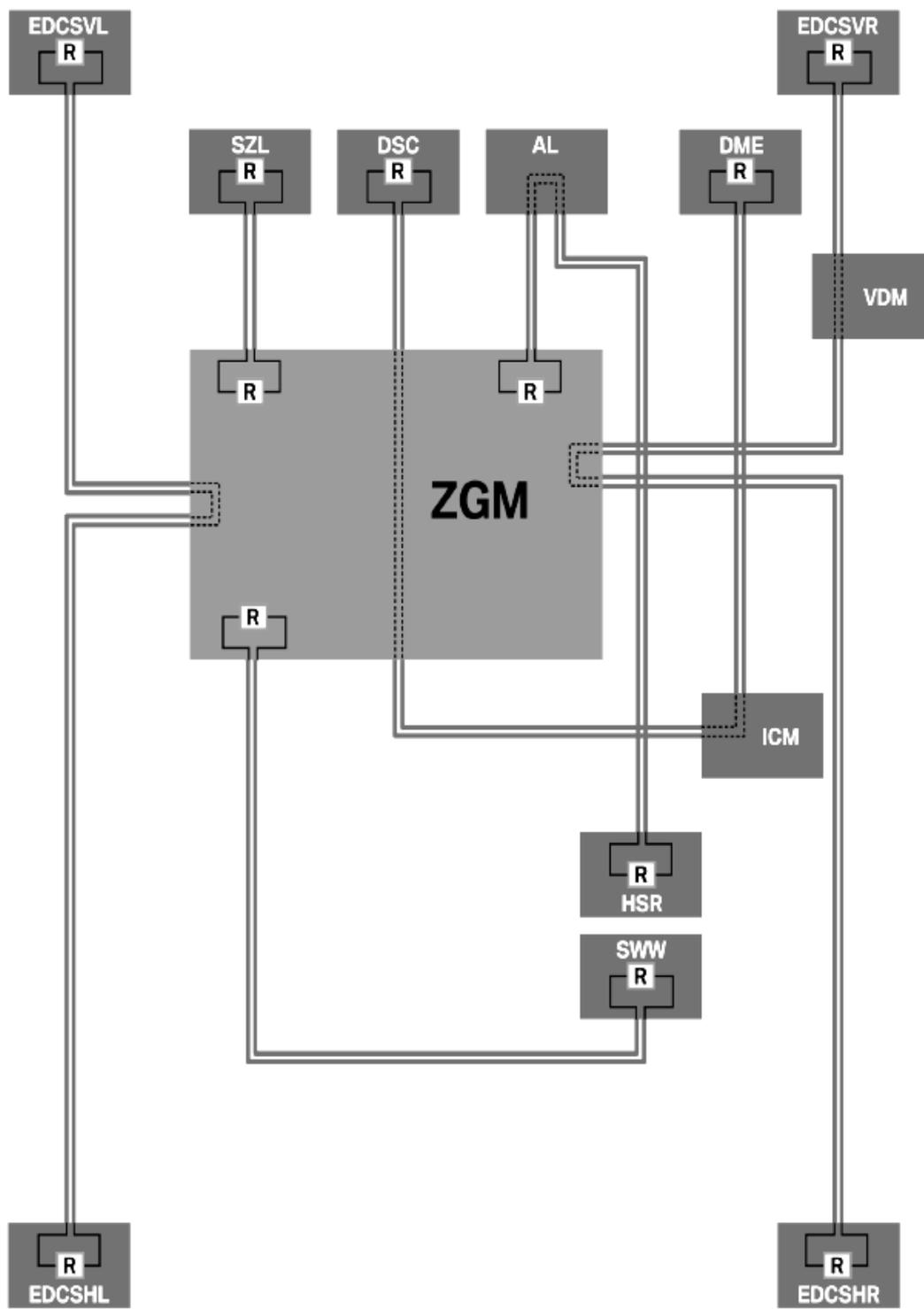


Рисунок 2.63 – Физическая структура FlexRay



11 - Топология FlexRay на F01

Рисунок 2.64 – Физическая структура FlexRay F01/F02 (топология)

В зависимости от комплектации в модуле ZGM имеется один или два так называемых концентратора с четырьмя шинными драйверами каждый. Шинные драйверы передают данные блоков управле-

ния через контроллер коммуникационной системы в центральный модуль межсетевого обмена (ZGM). К данным шинным драйверам блоки управления FlexRay в зависимости от типа их замыкания подключаются двумя разными способами. Как и в большинстве шинных систем, на FlexRay для предотвращения отражений на обоих концах линий передачи данных для замыкания (в качестве оконечной нагрузки шины) используются резисторы. Коэффициент сопротивления данных согласующих резисторов определяется по скорости передачи данных и длине кабелей. Согласующие резисторы находятся в блоках управления.

Если к шинному драйверу подключен только один блок управления (например, SZL к шинному драйверу BD0), разъемы на шинном драйвере и на блоке управления имеют по одному согласующему резистору. Такой тип подключения к центральному модулю межсетевого обмена называется «Замыкание выходного полюса» (рисунок 2.65).

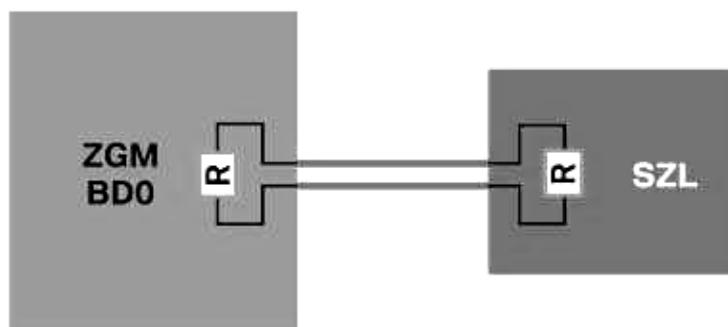


Рисунок 2.65 – Замыкание выходного полюса

Если подсоединение на блоке управления не является в физическом смысле выходным полюсом (например, DSC, ICM и DME на шинном драйвере BD2), то его обозначают как подводящий и передающий провод FlexRay. В этом случае оба компонента на концах соответствующего участка шины должны быть замкнуты согласующими резисторами (рисунок 2.66).

Данный вариант подсоединения используется как для центрального модуля межсетевого обмена, так и для некоторых блоков управления. Правда, блок управления с подводящим и передающим проводом также имеет «Замыкание выходного полюса по логической схеме HE» для съема данных. Однако данный тип замыкания нельзя про-

контролировать измерительными средствами на разъеме блока управления из-за его емкостно-резистивной схемы.

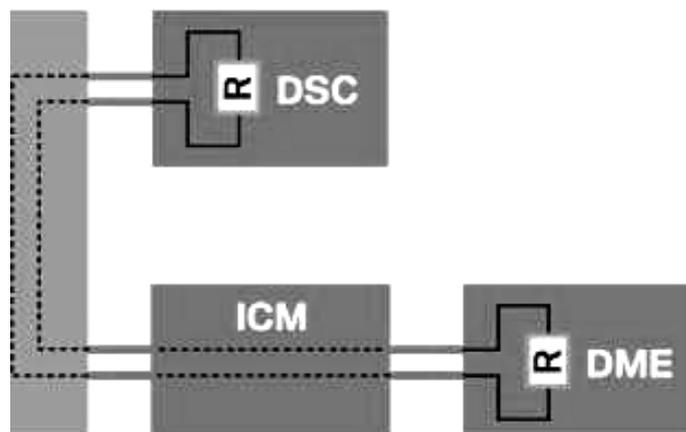


Рисунок 2.66 – Проложенная шина FlexRay

Для измерения всех параметров (обесточенной) шины FlexRay с целью определения сопротивления линии или согласующего сопротивления обязательно следует использовать электромонтажную схему транспортного средства.

### ***2.5.1 Основные характеристики FlexRay***

#### ***2.5.1.1. Топология шинной системы***

Шинная система FlexRay может иметь различную топологию и предлагаться в различных исполнениях. Существуют следующие виды топологии шинной системы:

- линейная;
- звездообразная;
- смешанная.

При линейной топологии все блоки управления (например, SG1 - SG3) соединены между собой двухпроводной шиной. В качестве среды передачи используется витая пара медных проводов. Такой вид соединения применяется также в шинах CAN. По обоим проводам передается одинаковая информация, однако, с разным уровнем напряжения. Передаваемый разностный сигнал нечувствителен к помехам. Линейная топология шинной системы может быть использована только для передачи данных посредством электрических сигналов (рисунок 2.67).

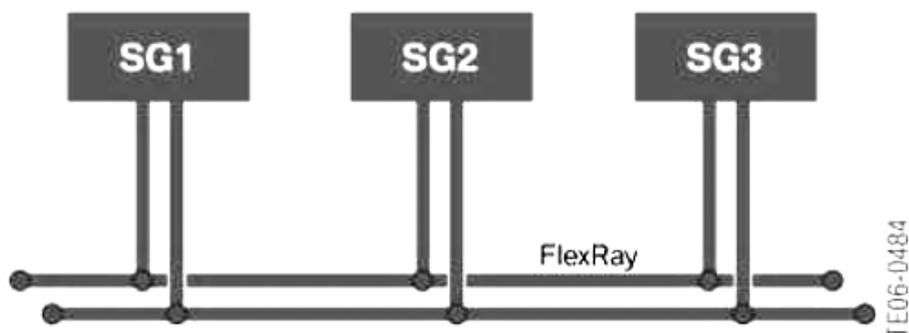


Рисунок 2.67 – Линейная топология шинной системы

При звездообразной топологии каждый сателлит (блоки управления SG2 - SG5) соединен отдельным проводом с центральным ведущим устройством (SG1). Звездообразная топология может быть использована для передачи данных, как посредством электрических сигналов, так и с помощью световых импульсов (рисунок 2.68).

При смешанной топологии разные ее виды применяются в рамках одной шинной системы. Шинная система частично имеет линейную, частично звездообразную конфигурацию. Данный вид топологии применяется на F01/F02. В центральном модуле межсетевого обмена (в зависимости от комплектации) имеются одна или две активные нулевые точки с четырьмя шинными драйверами каждая. Таким образом, существует до восьми вариантов подсоединения.

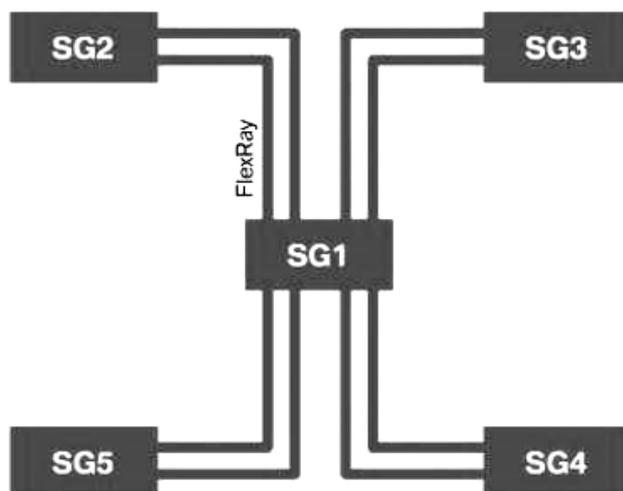


Рисунок 2.68 – Звездообразная топология шинной системы

### 2.5.1.2. Резервирование при передаче данных

В отказоустойчивых системах даже при обрыве одной линии шины должна быть обеспечена надежность передачи данных. Это достигается за счет резервирования, т. е. возможности передачи данных по второй линии (рисунок 2.69).

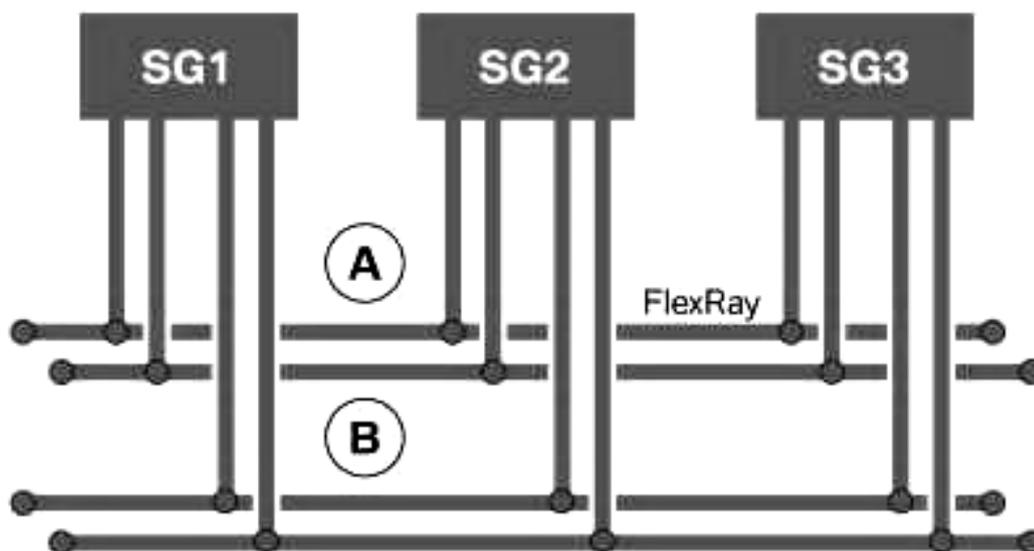


Рисунок 2.69 – Резервирование при передаче данных:  
А – канал 1; В – канал 2

В шинной системе с резервированием передачи данных предусмотрены два независимых канала. Каждый канал состоит из двухпроводного соединения. В случае повреждения одного канала информация, поступавшая по нему, передается по второму каналу, работающему в нормальном режиме. При резервированной передаче данных шина FlexRay также может иметь смешанную топологию.

### 2.5.1.3. Средства передачи - свойства сигналов

Сигнал шины FlexRay должен находиться в пределах определенных границ. Ниже на рисунке 2.70 представлены нормальная и искаженная характеристики сигналов. Электрический сигнал не должен входить во внутреннюю область ни по оси времени, ни по оси напряжения. Шинная система FlexRay характеризуется высокой скоростью передачи данных и, следовательно, быстрой сменой уровней напряжения.

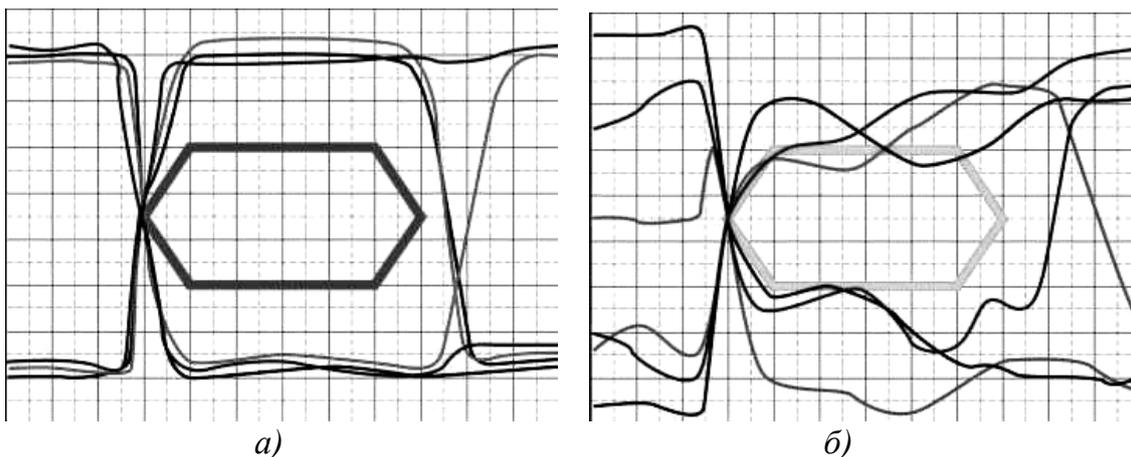


Рисунок 2.70 – Нормальные и искаженные характеристики:  
 а) нормальная характеристика; б) искаженная характеристика

Уровень напряжения, а также его повышение или падение (крутизна фронта сигнала) являются строго заданными параметрами и должны соответствовать определенным значениям. Пересечение границ показанных «полей» (зеленый и красный шестиугольники) недопустимо. Электрические помехи, возникающие из-за неправильной прокладки кабеля, переходных сопротивлений и т. д., могут привести к нарушению передачи данных.

Диапазоны напряжения шинной системы FlexRay:

- *система включена*– связь по шине не осуществляется 2,5 В;
- *сигнал высокого уровня*– 3,1 В (сигнал напряжения повышается на 600 мВ);
- *сигнал низкого уровня*– 1,9 В (сигнал напряжения понижается на 600 мВ).

#### 2.5.1.4. Детерминированная передача данных

Система CAN является событийно управляемой шинной системой. Передача данных осуществляется, если имеет место событие. При большом скоплении событий возможны задержки в передаче данных. В случае неудачной или неправильной передачи информации попытки ее отправки повторяются до тех пор, пока адресат не подтвердит прием.

Сбои в шинной системе могут привести к скоплению подобной событийно управляемой информации и к перегрузке шинной системы, когда передача отдельных сигналов происходит со значительной

задержкой по времени. Это может стать причиной искажения регулировочных характеристик отдельных систем. FlexRay- шинная система с управлением по времени, которая дополнительно позволяет осуществлять в отдельных поддиапазонах передачу данных, ориентированную на события. В диапазоне, управляемом по времени, кванты времени (timeslots) закрепляются за конкретной информацией. Квант времени - заданный интервал, выделенный для передачи определенной информации (например, частоты вращения).

Благодаря этому в шинной системе FlexRay передача важной периодической информации осуществляется в строгом соответствии с неизменным временным растром, что исключает перегрузки FlexRay.

Другие, менее срочные сообщения передаются в событийно-управляемом диапазоне.

Ниже приводится пример детерминированной передачи данных (рисунок 2.71). Благодаря детерминированной передаче данных любое сообщение в управляемом по времени диапазоне передается в реальном масштабе времени. Это означает, что передача осуществляется в течение строго определенного времени.

Таким образом, исключается задержка отправки важных сообщений по причине перегрузки шинной системы. В случае потери данных из-за временного сбоя в шинной системе (например, в результате электромагнитного воздействия) их повторная отправка не производится.

В пределах следующего кванта времени (временного диапазона), выделенного для передачи данной информации, будет отправлено текущее значение.

Шинная система FlexRay обеспечивает максимальную скорость передачи данных до 10 Мбит/с на канал. Это значение в 20 раз превышает скорость передачи данных по PT-CAN 2 или D-CAN.

Несмотря на то, что блоки управления FlexRay могут активизироваться сигналом шины, активизация большинства блоков управления FlexRay происходит через дополнительный провод активизации от системы CAS.

Активное рулевое управление и система управления вертикальной динамикой активизируются не через провод активизации, а через сигнал шины. Затем четыре сателлита в амортизаторах активизируются

ся посредством включения напряжения питания непосредственно системой управления вертикальной динамикой.

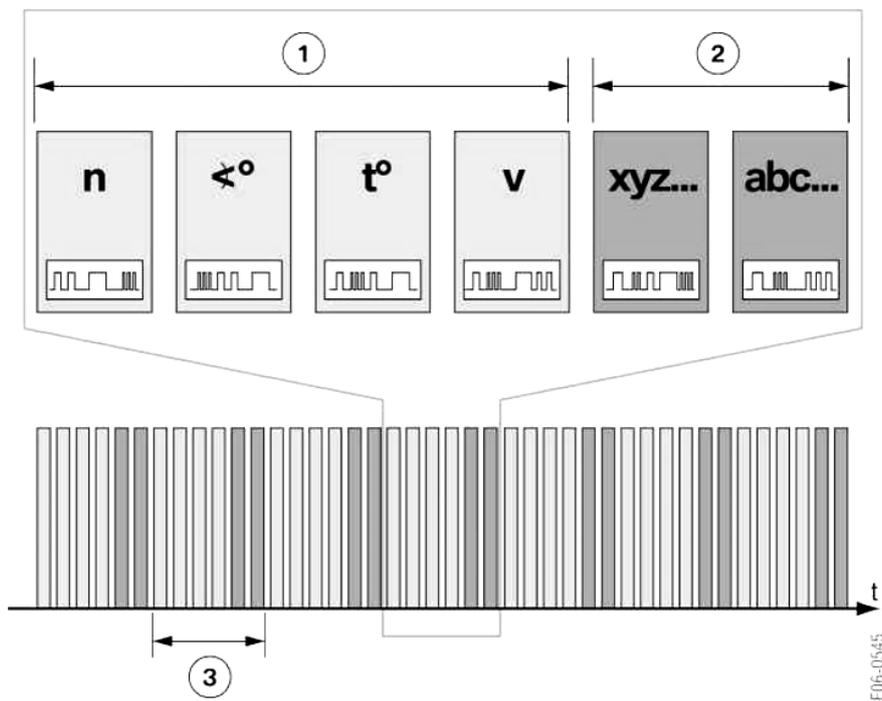


Рисунок 2.71 – Детерминированная передача данных:

1 – управляемый по времени диапазон циклической передачи данных;  
 2 – событийно-управляемый диапазон циклической передачи данных;  
 3 – цикл [общая продолжительность цикла 5 мс, из них 3 мс в статическом режиме (с управлением по времени) и 2 мс в динамическом режиме (событийно-управляемая передача)];

$n$  - частота вращения;  $\alpha^\circ$  – угол;  $t^\circ$  – температура;  $v$  – скорость движения; xyz... abc... – событийно-управляемая информация;  $t$  – время

### 2.5.1.5. Синхронизация

Для обеспечения синхронности выполнения отдельных функций объединенными в сеть блоками управления требуется общая точка отсчета времени. Так как все блоки управления внутри своей системы работают от собственного тактового генератора, согласование по времени должно производиться через шину. Блоки управления измеряют продолжительность определенных синхронизирующих битов, рассчитывают на основании этого среднее значение и приводят свой тактовый режим в соответствие с данным значением. Синхронизирующие биты передаются в статической части сообщения по шине. Синхронизация начинается на шине FlexRay после запуска системы

между двумя из блоков управления, имеющих разрешение на активизацию (в шинной структуре обозначены буквой «S»), после отправки блоком управления CAS сигнала активизации. После завершения данного процесса остальные блоки управления регистрируются последовательно на FlexRay и производят расчет своего определенного разностного значения. Дополнительно во время работы путем расчетов происходит корректировка синхронизации. За счет этого даже самые незначительные расхождения по времени за относительно длительный период не приводят к ошибкам при передаче.

#### *2.5.1.6. Обработка неисправностей*

В случае неисправностей в шинных кабелях (например, короткое замыкание или короткое замыкание на массу) или в блоках управления FlexRay возможно отключение отдельных блоков управления или целых линий от системы связи по шинам. Исключение составляет линия с четырьмя блоками управления FlexRay (ZGM, DME, DSC, ICM), имеющими разрешение на активизацию. Если между данными блоками управления нарушается связь, пуск двигателя невозможен.

Дополнительно так называемое реле контроля шины, имеющееся в блоках управления, препятствует передаче сообщений в неразрешенные моменты времени. Тем самым предотвращается запись одних сообщений поверх других.

Кабель FlexRay на F01/F02 выполнен в виде витой пары проводов в оболочке. Оболочка защищает кабель от механических повреждений. Согласующие резисторы находятся частично в центральном модуле межсетевого обмена и в терминалах. Так как волновое сопротивление (полное сопротивление высокочастотных проводов) зависит от внешних факторов воздействия, согласующие резисторы очень точно подобраны с учетом необходимого значения сопротивления. При помощи омметра (мультиметра) относительно просто можно проверить участки проводов, ведущие к терминалам. Измерение должно производиться от центрального модуля межсетевого обмена.

В шине FlexRay согласующие резисторы находятся в следующих блоках управления:

- центральный модуль межсетевого обмена (только выходные полюсы);
- сателлиты электронной системы регулирования жесткости амортизаторов;

- цифровая электронная система управления двигателем;
- система динамического контроля стабильности;
- регулировка угла бокового увода заднего моста;
- коммутационный центр в рулевой колонке;
- сигнализация смены полосы движения.

## **Контрольные вопросы к разделу 2.5**

1. Каково назначение коммуникационной системы FlexRay?
2. Каковы основные преимущества шины FlexRay?
3. Каковы особенности F01/F02 топологии FlexRay?
4. Как осуществляется резервирование при передаче данных по шине FlexRay?
5. Каковы основные свойства сигналов шины FlexRay?
6. Каким образом реализована детерминированная передача данных?
7. Как обеспечивается синхронизация при передаче данных на шине FlexRay?
8. Как производится обработка неисправностей в шинных или в блоках управления FlexRay?

## **2.6. Сеть EtherCAT**

EtherCAT – это промышленная шина, основанная на сети Ethernet. EtherCAT отличается от других промышленных шин, в том числе базирующихся на Ethernet, в первую очередь своей беспрецедентной производительностью. Благодаря гибкой топологии и простой конфигурации и минимальной стоимости аппаратных средств шина EtherCAT может быть реализована с очень высокой экономической эффективностью, она позволяет использовать промышленные шины в тех областях, где ранее возможность их применения даже не рассматривалась.

EtherCAT – это промышленная шина на базе Ethernet, впервые представленная для использования в производстве полупроводниковых приборов в 2004 г. Она стандартизована такими ассоциациями, как SEMI, IEC и ISO. Функциональный принцип протокола устройств EtherCAT существенно отличает эту технологию от других, базирующихся на Ethernet-решениях. В технологии EtherCAT не использу-

ется такой режим, в котором пакет Ethernet сначала принимается, затем интерпретируется в ответный пакет от устройства ввода/вывода, содержащий данные технологического процесса. Вместо этого пакет Ethernet обрабатывается «на лету», без какой-либо буферизации (рисунок 2.72).

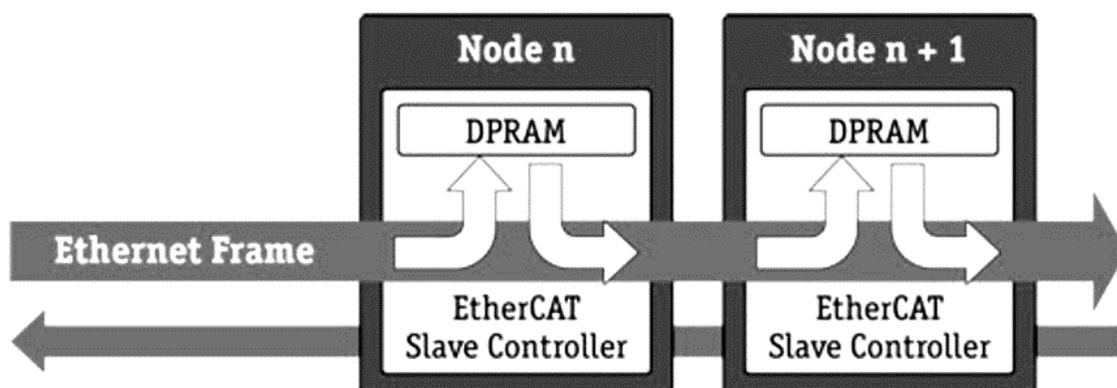


Рисунок 2.72 – Обработка кадра «на лету»

Специальная микросхема каждого ведомого EtherCAT-устройства, оснащенная двухпортовой памятью DPRAM, считывает адресованные ему данные во время передачи кадра следующему устройству. Входные данные также «на лету» вставляются во время прохождения кадра. Кадры при этом практически не задерживаются, аппаратная задержка измеряется наносекундами. Кадр, отправленный ведущим устройством, передается каждому следующему устройству, пока не достигнет конца сегмента (или ветви). Последнее устройство определяет, что к его выходному порту не подключено никаких устройств, и отправляет кадр назад ведущему устройству по второй витой паре. При определенных условиях возможна даже такая ситуация, при которой задержка распространения пакета через ряд ведомых устройств окажется меньше длительности самого пакета. Иными словами, ведущее устройство начнет получать ответ от устройств распределенного ввода/вывода раньше, чем успеет отправить «хвост» пакета, как это ни парадоксально звучит.

Протокол EtherCAT реализован на самом нижнем логическом уровне стека протоколов Ethernet. Структура телеграмм Ethernet и физический уровень интерфейса сохранены, что обеспечивает совместимость со стандартным сетевым оборудованием: коммутаторы, преобразователи медь/оптоволокно и т.п.

Из всех устройств, подключенных к шине EtherCAT, настоящим участником Ethernet со своим MAC-адресом является только одно устройство – это мастер шины EtherCAT. Только мастер может быть инициатором телеграмм. Все остальные устройства «на лету» модифицируют проходящую через них телеграмму, читая и записывая в нее данные технологического процесса. Аппаратная задержка на прохождение телеграммы через одно slave-устройство составляет всего несколько наносекунд. Таким образом, всего одна единственная большая ширококвещательная телеграмма Ethernet, посланная мастером в сеть, возвращается мастеру модифицированной. И она уже несет в себе всю информацию. Это позволяет максимально эффективно (на 80-97%) использовать физический трафик среды передачи 100 Мбит/с. Данное обстоятельство коренным образом отличает EtherCAT от прочих решений для промышленной шины реального времени на основе Ethernet. Каждый отдельный модуль ввода/вывода, а не контроллер, является самостоятельным slave-устройством. В головном блоке контроллера происходит лишь преобразование физического уровня сигнала.

Структура данных в телеграмме EtherCAT соответствует не топологии подключения slave-устройств к шине, а структуре прикладной управляющей программы, каждый модуль ввода/вывода «знает» в какое место телеграммы EtherCAT он должен вставить свои данные. В сочетании с использованием DMA (Direct Memory Access) контроллера, интегрированного в адаптер Ethernet, функция адресации данных (mapping) перекладывается с центрального процессора мастера на slave-устройства. Их коллективные действия обеспечивают этот самый «mapping» на уровне модулей ввода/вывода. Таким образом, аппаратная реализация мастера шины EtherCAT (контроллер DMA), плюс аппаратная реализация slave-распределенных устройств ввода. EtherCAT – технология будущего: новые технологии вывода в сочетании с опросом одновременно всех периферийных устройств в рамках всего одной телеграммы Ethernet приводит к рекордному, не имеющему аналогов среди других промышленных шин быстродействию шины EtherCAT (таблица 2.25). Типичное время цикла EtherCAT составляет 50–250 мкс, в то время как в традиционных промышленных шинах на каждое обновление требуется 5–15 мс, т.е. традиционные промышленные шины работают медленнее от 20 до 300 раз.

Таблица 2.25 – Быстродействие шины EtherCAT

Данные процесса	Время обновления
256 распределенных дискретных входов/ выходов	11 мкс
1000 распределенных дискретных входов/ выходов	30 мкс
200 аналоговых входов/выходов (16 бит)	50 мкс или 20 кГц
100 сервоприводов, входные и выходные данные которых имеют разрядность 8 байт	100 мкс
1 главный шлюз промышленной шины (1486 байт входных и выходных данных)	150 мкс

Для синхронизации работы объединенных в сеть узлов в EtherCAT применяется точное согласование распределенных часов EtherCAT – внутренних часов микросхем EtherCAT-контроллеров. В отличие от полностью синхронной связи, где при возникновении ошибки сразу же страдает качество синхронизации, распределенные согласованные часы имеют высокую степень устойчивости к возможным задержкам из-за ошибок и кратковременных сбоев связи, за счет локального хранения точного времени внутри каждого из распределенных устройств ввода/вывода.

В сети EtherCAT механизм синхронизации времени реализован полностью аппаратно. Каждые часы могут легко и точно измерить задержку относительно других часов, так как для связи используется логическая и полnodуплексная физическая кольцевая структура Ethernet, т.е. каждый пакет EtherCAT проходит дважды через каждое ведомое устройство (прямой и обратный путь по разным витым парам). На основе этого значения задержки производится подстройка распределенных часов, что позволяет достичь очень точной временной базы с разбросом существенно меньше 1 мкс в масштабах всей сети. Это идеальная точность для приложений синхронизированного управления движением и интеграции измерительных задач внутри одной сети.

Ведущие EtherCAT-устройства обычно реализуются программно с использованием стандартного оборудования Ethernet и не требуют выделенного коммуникационного сопроцессора. Ведомые устройства содержат встроенные контроллеры на основе БИС, обеспечивающие обработку пакета, и все ключевые временные параметры, по-

этому мощные микроконтроллеры на стороне ведомых устройств не нужны. Инфраструктура EtherCAT не нуждается в коммутаторах и других активных компонентах – в ней используются стандартные кабели и разъемы. Трудозатраты на настройку и инженерно-конструкторские работы также уменьшаются, поскольку отпадает нужда в настройке сети. И наконец, функции диагностики позволяют точно локализовать ошибки и снизить затраты времени на поиск и устранение неисправностей.

Сети EtherCAT не имеют практических ограничений на топологию – они могут иметь линейную, звездообразную, древовидную, резервированную кольцевую топологию или даже представлять собой комбинацию всего вышеперечисленного и позволяют использовать до 65 535 узлов на сегмент. Если расстояния между двумя узлами спецификации 100BaseTX, равного 100 м, недостаточно, то его можно увеличить до 2 км за счет волоконно-оптических кабелей. Поддержка «горячего подключения» позволяет подключать и отключать узлы или даже целые технологические камеры во время работы.

Протокол устройств EtherCAT может транспортировать по одной и той же физической сети и другие сервисы и протоколы, основанные на Ethernet. Они туннелируются через протокол EtherCAT, поэтому показатели реального времени не ухудшаются. Таким образом, в среде EtherCAT можно использовать все IP-технологии: HTTP, SECS/GEM, EDA, HSMS-SS, передачу по FTP и т.д.

Для реализации гарантированного противоаварийного обмена данными по EtherCAT членами ассоциации EtherCAT Technology Group был разработан функциональный протокол ПАЗ по EtherCAT (FSoE). EtherCAT используется как одноканальная система связи для передачи защищенной и незащищенной информации. Среда передачи считается «черным каналом» и не учитывается в соображениях безопасности. Кадр обеспечения безопасности, содержащий защищенные данные технологического процесса и необходимые резервные копии данных, включен в данные технологического процесса EtherCAT. Этот «контейнер» защищенным образом анализируется в устройствах на уровне приложений. Связь остается одноканальной.

Одна из новейших добавок к технологии EtherCAT – протокол автоматизации EtherCAT (EAP). Протокол EAP объединяет EtherCAT-протоколы с классическими Ethernet-топологиями для интеграции ве-

душих устройств шины EtherCAT, инструментов конфигурирования сетей и беспроводных компонентов. В то время как протокол устройств EtherCAT – с обработкой «на лету» – работает абсолютно детерминировано (обычно в микросекундном диапазоне), время цикла EAP лежит в диапазоне миллисекунд, в зависимости от топологии сети, но позволяет организовывать многоуровневую сеть EtherCAT с использованием стандартных маршрутизаторов и прочего стандартного сетевого оборудования и ПО.

Объединение производительности EtherCAT с вычислительной мощностью и пропускной способностью современных средств управления на базе ПК позволяет реализовать новый подход к архитектуре управления. Эта шина позволяет передавать не только параметры и команды для рассредоточенных контроллеров или данные относительно медленных контуров управления, но и данные быстродействующих контуров управления.

Перенос алгоритмов управления на уровень центрального управляющего ПК позволяет «вскрыть черные ящики» (встроенный в периферийное устройство алгоритм управления задан производителем устройства и недоступен для системного интегратора), характерные для прошлых подходов: теперь новые передовые алгоритмы можно разработать, испытать и в конце концов реализовать в инструментах без особых сложностей и без привлечения поставщика подсистемы.

Вдобавок существенно упрощается интерфейс шины: вместо сложного и специфического для каждого производителя списка конфигурационных параметров происходит лишь обмен простыми, лишенными избыточности командами / фактическими значениями наряду со стандартизованными командами и словами состояния. Сервоприводы, контроллеры температуры и т.д. становятся менее избыточными и более дешевыми, что, в свою очередь, сводит к минимуму зависимость от поставщика.

EtherCAT обеспечивает быстрый обмен данными не только вне устройств ввода/вывода, но и внутри них. Управление дискретными сигналами ввода/вывода напрямую осуществляется подчиненным EtherCAT-контроллером, оно идет без задержек благодаря локальному микропрограммному обеспечению, и не зависит от производительности установленного микроконтроллера (рисунок 2.73).

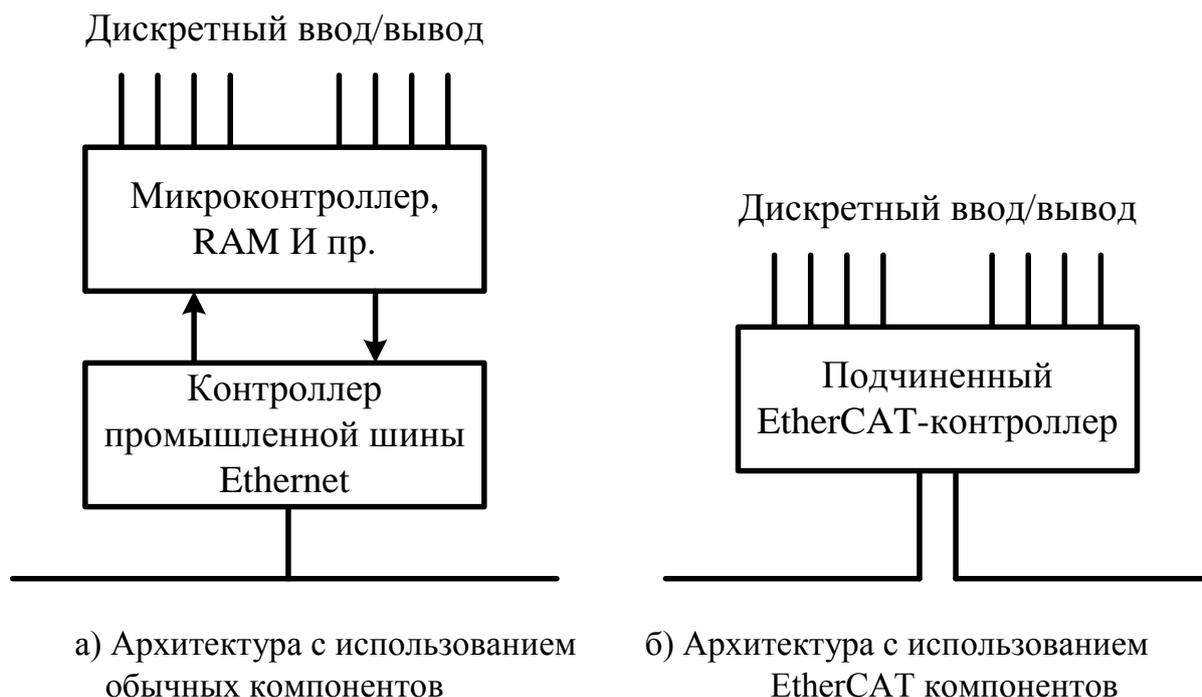


Рисунок 2.73 – Архитектура сети

### Контрольные вопросы к разделу 2.6

1. Каковы основные преимущества промышленной шины EtherCAT?
2. Какой функциональный принцип работы протокола устройств EtherCAT?
3. На каком логическом уровне стека протоколов Ethernet реализован EtherCAT?
4. Как работает мастер шины EtherCAT?
5. Какова структура данных в телеграмме EtherCAT?
6. Каково время цикла EtherCAT?
7. Какой принцип положен в основу синхронизации работы объединенных в сеть узлов в EtherCAT?
8. Каковы возможности топологии сетей EtherCAT?
9. Как реализован гарантированный противоаварийный обмен данными по EtherCAT?

### **3. РЕКОМЕНДАЦИИ ПО СТРУКТУРНОЙ ОРГАНИЗАЦИИ ВЫЧИСЛИТЕЛЬНОЙ СЕТИ**

Рассмотрев существующие решения по информационному взаимодействию, можно выявить следующие подходы структурной организации сетей обмена данными подвижных средства - автомобилей, сельхозтехники и др.:

- Бортовые сети, как правило, имеют многоуровневую организацию – основная быстродействующая сеть, объединяющая двигатель, трансмиссию, тормоза и дополнительная сеть (сети), объединяющая системы комфорта в легковом автомобиле, электропневмоавтоматику прицепов в грузовом автомобиле, оборудование салона автобусов, навесные орудия сельхозтехники и т.п.

- Количество узлов в одном уровне сети не превышает 30, имеется тенденция деления бортовой сети по функциональному признаку на подсети с малым количеством узлов (10..15).

- Бортовые сети имеют шинную архитектуру, оптимальную с точки зрения кабельных соединений, надежности и расположения на борту мобильного объекта. На периферийных узлах вспомогательных сетей может иметь место ограниченное применение звездной архитектуры, что вызвано прежде всего попыткой удешевления компонентов, но с технической точки зрения такое решение не целесообразно.

- Смешанные сети широко применяются на подвижных объектах, это обусловлено либо объединением групп узлов и агрегатов по функциональному признаку, либо экономическими причинами – компоненты медленных вспомогательных сетей дешевле.

#### **Протоколы низкого уровня**

Из существующих в настоящее время протоколов низкого уровня, используемых на подвижных объектах можно выделить следующие:

1. - CAN
2. - FlexRay
3. - RS-485
4. - LIN

Первые три протокола используют дифференциальный принцип передачи сигнала, что дает определенную помехоустойчивость. Протокол LIN использует однопроводную линию и был разработан как

дешёвое дополнение CAN интерфейса в тех областях, где не требуется высокая скорость передачи данных и высокая надёжность. LIN разработан для создания дешёвых локальных сетей обмена данными на небольшие расстояния. В автомобиле, например, такие устройства как стеклоочистители, электростеклоподъемники, замки дверей, приводы и обогрев зеркал, приводы сидений объединены в единую сеть LIN. Целесообразно в серийном производстве. При выпуске единичных образцов оборонного или специального назначения экономия незначительна в сравнении с программным и аппаратным усложнением бортовой сети, связанным с использованием разнотипных компонентов.

Протокол CAN изначально был разработан для использования в автомобилях. Протокол CAN характеризуется высокой скоростью передачи данных (до 1 Мбит/с), высокой надёжностью системы и её отказоустойчивостью в экстремальных условиях. Хорошим примером использования шины CAN может служить современный автомобиль. CAN используется в таких ответственных системах автомобиля как управление двигателем, обеспечение безопасности (подушки безопасности, ABS, контроль давления шин) и т.д. Протокол CAN занял прочные позиции не только в автомобилестроении, но и в авиации, военной технике, на железной дороге, системах управления технологическим оборудованием. В качестве протокола низко уровня CAN может быть рекомендован для построения бортовой сети подвижного объекта по следующим соображениям:

1. Проверенный эксплуатацией на транспортных средствах протокол, на практике доказавший свою жизнеспособность.
2. Высокая живучесть: при распаде сети на части, образовавшиеся фрагменты внутри себя сохраняют работоспособность.
3. Наличие протоколов высокого уровня и возможность создания собственных протоколов.
4. Наличие элементной базы отечественного производства (ЗАО «Миландр»).
5. Совместимость с отечественными БИУС.
6. Совместимость с системами управления ДВС, автомобильными агрегатами.

Протокол FlexRay, несмотря на высокие заявленные характеристики (скорость передачи данных до 10 Мбит/с, конфигурационная

гибкость), имеет ограниченное применение на легковых автомобилях премиум-класса. Объясняется это высокой стоимостью компонентов и сложностью программирования Time-Trigged интерфейсов. Появившись впервые на легковых автомобилях BMW в 2005 году, протокол FlexRay до настоящего времени так и не получил широкого распространения. Для сравнения: протокол CAN, впервые появившись на серийных автомобилях в 1999 году к 2001 году имел широкое распространение. Протокол FlexRay не рекомендован для построения бортовой сети подвижного объекта по следующим соображениям:

1. Отсутствие отечественной и ограниченность импортной элементной базы.
2. Полное отсутствие в открытом доступе протоколов высокого уровня и какой-либо информации о них.
3. Крайняя сложность самостоятельного программирования.
4. Несовместимость с электронными системами отечественного производства.

Протокол низкого уровня RS-485 хорошо известен и применялся на тяжелых грузовиках и автобусах до недавнего времени, в настоящее время заменен на CAN протокол. Однако, применение RS-485 на подвижном объекте вполне возможно.

### **Протоколы высокого уровня**

Для CAN сетей рассмотрены следующие протоколы высокого уровня:

1. DeviceNet
2. SDS
3. CANopen
4. CANKingdom
5. J1939

Протоколы DeviceNet и SDS ориентированы на использование готовых контроллеров, имеющих аппаратную поддержку данных протоколов, пользователю остается только собрать в сеть готовые устройства. Если же разработчик намерен самостоятельно создавать устройство, начиная с уровня микросхем, то применение высокоуровневых протоколов DeviceNet и SDS скорее усложняет процесс построения сетей подвижного объекта. Протоколы CANopen и CANKingdom более подходят для построения сетей подвижного объекта, однако и здесь имеются определенные сложности. Для протокола CANopen

необходимы EDS - ElectronicDataSheet "электронный список параметров", который описывает конфигурацию и параметры устройств. Конечно, разработчик может создать и записать в память контроллера собственный EDS. Протокол CANKingdom использует в качестве идентификаторов индивидуальные номера устройств в соответствии с кодом EAN/UPC. Таким образом, вышеперечисленные протоколы высокого уровня ориентированы на компоненты иностранного производства.

Протокол J1939 никак не связан с конкретными аппаратными средствами и не является в полном смысле протоколом, а скорее является набором правил, регламентирующих распределение полей идентификатора и поля данных, а так же набором детализированных параметрических групп (PGN). Программная реализация остается на усмотрение разработчика.

## **ЗАКЛЮЧЕНИЕ**

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Автоматизированное рабочее место водителя многоосного колёсного шасси // ООО «НИЦ СК «Континент»» [Электронный ресурс]. URL: [http://nic-kontinent.ru/index.php?option=com\\_content&view=article&id=8:2009-09-04-09-19-21&catid=1:2009-09-04-09-07-45&Itemid=6](http://nic-kontinent.ru/index.php?option=com_content&view=article&id=8:2009-09-04-09-19-21&catid=1:2009-09-04-09-07-45&Itemid=6) (дата обращения: 21.12.2018).

2. Автомобильный справочник. [Электронный ресурс]. URL: <http://press.ocenin.ru/shiny-can-v-avtomobilyah/>

3. Бортовая информационно-управляющая система (БИУС) // Унитарное предприятие «ИЦТ “Горизонт”» [Электронный ресурс]. URL: <http://www.vpk.gov.by/catalog/niihor/910> (дата обращения: 05.07.2016).

4. Бортовая информационно-управляющая система (БИУС) транспортного средства [Электронный ресурс]. URL: <https://www.bibliofond.ru/view.aspx?id=896491> © Библиофонд

5. Бортовая информационно-управляющая система (БИУС). ОАО «СКБ “Камертон”» // Государственный военно-промышленный комитет Республики Беларусь [Электронный ресурс]. URL: <http://www.vpk.govby/catalog/kamerton/246> (дата обращения: 05.07.2016).

6. Будущий триумф «Тайфуна» // Военное обозрение [Электронный ресурс]. URL: <http://topwar.ru/8969-buduschiy-triumf-tayfuna.html> (дата обращения: 05.07.2016).

7. Вознесенский А. Н. Бортовые информационно-управляющие системы // Машиностроение: энциклопедия. – М.: Машиностроение, 1997. – Раздел IV. Расчёт и конструирование машин. – Т. IV – С. 578 – 589.

8. ВПК-3927 «Волк». «Волк» проходит финальные испытания // OfRoadClub.ru — территория общения [Электронный ресурс]. URL: [http://offroadclub.ru/automobiles/trucks/review/vpk-3927\\_volk.html](http://offroadclub.ru/automobiles/trucks/review/vpk-3927_volk.html) (дата обращения: 05.07.2016).

9. КамАЗ-63968 «Тайфун». Защищённый автомобиль // Русская сила – современное оружие [Электронный ресурс]. URL: <http://xn----7sbb5ahj4aiadq2m.xn--p1ai/guide/army/tr/kamaz63969.shtml> (дата обращения: 20.12.2018).

10. Мультиплексы для спецтехники. Бортсеть нового поколения (БНП) // ООО «Научно-техническое предприятие “ДЕКА”» [Элек-

тронный ресурс]. URL:<http://deka-ntp.ru/multi.htm> (дата обращения: 05.07.2016).

11. Научно-конструкторское бюро вычислительных систем. Основные направления деятельности. [Электронный ресурс]. [http://www.nkbvs.ru/products/sistemy\\_upravleniya\\_vzaimodeystviem/](http://www.nkbvs.ru/products/sistemy_upravleniya_vzaimodeystviem/) (дата обращения: 20.12.2018).

12. Обновлённый «Тигр» // ООО «Военно-промышленная компания» [Электронный ресурс]. URL: [milindcom.ru/news/archive/?id=357](http://milindcom.ru/news/archive/?id=357) (дата обращения: 05.07.2016).

13. Овсиенко О. М., Шахнович И. В. Без CAN российским инженерам не выжить. О CAN-технологии и не только // Электроника: наука, технология, бизнес. – 2010. – № 5. – С. 12-17.

14. Операционная система реального времени «Багет 3.0» // Программные продукты и системы [Электронный ресурс]. URL: <http://www.swsys.ru/index.php?page=article&id=2603> (дата обращения: 05.07.2016).

15. Операционные системы реального времени для 32-разрядных микропроцессоров // Современная электроника [Электронный ресурс]. Систем. требования: AdobeAcrobatReader. URL: <http://www.soel.ru/cms/f/?/326514.pdf> (дата обращения: 05.07.2016).

16. ОТТ 9.1.9-2006. Система общих технических требований к видам вооружения и военной техники. Военная автомобильная техника. Бортовые информационно-управляющие системы. Общие технические требования. – Бронницы, 2006. – 38 с.

17. Системная модернизация // Пятое колесо: российский автомобильный журнал [Электронный ресурс]. URL: <http://5koleso.ru/content/sistemnaya-modernizaciya> (дата обращения: 05.07.2016).

18. Список аббревиатур и сокращений Опель [Электронный ресурс]. URL: <https://www.drive2.ru/b/2212657/>

19. СПМ-2Э «Тигр»: brutальный, мощный, надёжный // Новый оборонный заказ. Стратегии [Электронный ресурс]. URL: <http://dfnc.ru/arhiv-zhurnalov/c142-2009-2-5/spm-2e> (дата обращения: 20.12.2018).

20. Структура информационных шин и электронных модулей в автомобилях OPEL. [Электронный ресурс]. URL: <http://www.autopro.spb.ru/?artun=00000067>

21. Устройство и техническое описание коммуникационной сети. [Электронный ресурс]. URL: [http://zinref.ru/avtomobili/Land\\_Rover/004\\_03\\_Land\\_Rover\\_Discaveri\\_3/283.htm](http://zinref.ru/avtomobili/Land_Rover/004_03_Land_Rover_Discaveri_3/283.htm)

22. Шипилевский Г. Б., Вознесенский А. Н. Пути электронизации отечественной тракторной техники // Тракторы и сельскохозяйственные машины. – 2006. – № 8. – С. 6-9.

23. Шумилин С. Советские бронетранспортёры БТР-60, 70, 80, 90. Киров: Торнадо, Кировское общество любителей военной техники и моделизма, 2000. – Ч. 2. – 56 с.

24. Электронная система общего управления, контроля и диагностики // ОАО «Научно-конструкторское бюро вычислительных систем» [Электронный ресурс]. URL: <http://web.nkbvs.ru/index.php?page=esukd> (дата обращения: 05.07.2016).

25. Электронные системы автомобилей: учебное пособие / О. Л. Коваленко; Сев. (Арктич.) федер. ун-т им. М.В. Ломоносова. – Архангельск: ИПЦ САФУ, 2013. – 80 с.: ил. ISBN 978-5-261-00762-3 <https://narfu.ru/university/library/books/0900.pdf>

26. Элинс. Проектирование, разработка и производство промышленных компьютерных продуктов. [Электронный ресурс]. <https://www.zelenograd.ru/catalog/directory/498051/сpec/> (дата обращения: 20.12.2018).

## ПРИЛОЖЕНИЯ

### Приложение 1

#### АВТОМОБИЛЬНЫЕ АББРЕВИАТУРЫ, СОКРАЩЕНИЯ И ИХ РАСШИФРОВКА

**ABS (Anti-lock Braking System)** – Антиблокировочная система при торможении. Предотвращает блокировку колес при торможении автомобиля, что сохраняет его курсовую устойчивость и управляемость.

**AC (Air Condition)** – Кондиционер, см. также «ЕСС».

**ACC (Active Cornering Control)** – Иногда ACE, BCS, CATS. Автоматическая система стабилизации поперечного положения кузова в поворотах, а в некоторых случаях и изменяемого хода подвесок, главную роль в которой играют активные элементы подвески.

**ACC (Adaptive Cruise Control)** – Адаптивный круиз-контроль, круиз-контроль.

**Active Select** – Функция установленная на автоматической коробке передач МТА, позволяющая управлять переключением передач, как в автоматическом, так и в ручном режиме.

**ADR (Automatic Distance Regulation)** – система по поддержанию безопасного расстояния до впереди идущего автомобиля. В основе системы лежит радар, установленный в передней части автомобиля.

**AFL (Adaptive Forward Lighting)** – адаптивная система головного освещения, связанная с системой рулевого управления. В систему AFL входятдвигающиеся горизонтально биксеноновые фары, способные освещать дорогу в направлении, перпендикулярном направлению движения автомобиля.

**AFU (Assistance au freinage d'urgence – фр.)** – система помощи при экстренном торможении (на французских автомобилях Renault, Peugeot, Citroen) – система посредством датчика скорости нажатия на педаль тормоза распознает ситуацию экстренного торможения. После чего действует непосредственно на тормозную систему, обеспечивая максимальную силу торможения.

**AGS (Adaptive Getriebe-Steuerung)** – Самонастраивающаяся система автоматической коробки передач. «Индивидуальная» коробка передач. AGS в процессе движения выбирает самую подходящую для водителя передачу.

**AHL (Automatic Headlamp Leveling)** – Автоматическая система регулирования угла наклона фар.

**AHR (Active Head Restraints)** — Активные подголовники. Система пассивной безопасности, которая снижает вероятность повреждения шейных позвонков пассажира в случае удара сзади.

**AHS (Auxiliary Heater System)** – Вспомогательная система отопления. Вспомогательный отопитель. Работает автономно. Принцип аналогичен известным отопителям Vebasto.

**Airbag** – подушка безопасности, см. «Full-Size Airbag».

**ALC (Automatic Lighting Control)** – Система автоматического управления фарами с распознаванием тоннелей и датчиком наружной освещенности. Автоматически включает и выключает ближний свет.

**ALDL (Assembly Line Diagnostic Link)** – разъем для подключения диагностического прибора TECH 2, см. «Разъем для подключения диагностического прибора».

**APP (Accelerator Pedal Positioning)** – датчик положения педали акселератора.

**AQS (Air Quality Sensor)** – Датчик качества воздуха.

**ARP (Anti Rollover Protection)** – Система предотвращения опрокидывания, защита от опрокидывания. Варианты названий: ARP (Active Rollover Protection) – активная защита от опрокидывания; RSC (Roll Stability Control) – контроль устойчивости к опрокидыванию; ROP (Roll Over Prevention) – предотвращение опрокидывания. Является частью системы ESP.

**ASC (Anti-Slip Control)** – Антипробуксовочная система или, как иногда ее называют, «трэкшн-контроль» (traction-control). Она же **ASR (Antriebsschlupfregelung - нем., Automatic Slip Regulation, Acceleration Slip Regulation, Anti-Slip Regulation - англ.)** (в автомобилях германского производства), а также **DTS (Dynamic Traction Control)**, **ETC**, **TCS (Traction Control System)**, **STC**, **TRACS**, **ASC+T (Automatic Stability Control + Traction)**. В машинах Toyota ее аналог именуется **TRC**, в Opel – **DSA**, в Range Rover – **ETC**, в Honda – **TSC**, в Volvo – **STC**.

**ASM (Alarm Siren Module)** – Контроллер системы сигнализации.

**A/T (Automatic Transmission)** – АКПП, автоматическая коробка переключения передач.

**ATC (Automatic Traction Control)** – Автоматическое управление тягой

**ATWS (Anti-Theft Warning System)** – система противоугонной сигнализации. Это сочетание двух систем. С одной стороны это полнофункциональная охранная система, которая имеет связь с основной противоугонной системой (иммобилайзером). С другой это центральный замок с функциями управления стеклоподъемниками и люком.

**AU** – контроль токсичности выхлопных газов; правила проведения проверок разных двигателей.

**AUS (Audio Systems)** – аудиосистемы.

**AWD (All Wheel Drive)** – полный привод.

**BA (Brake Assist)** – Иногда **BAS**, **PA** или **PABS**. Электронная система управления давлением в гидравлической системе тормозов, которая в случае необходимости экстренного торможения и недостаточного при этом усилия на педали тормоза самостоятельно повышает давление в тормозной магистрали, делая это во много раз быстрее, чем на то способен человек.

**BCM (Body Control Module), BEC (Body Electrical Center)** – Центр управления электроникой кузова. Электронный блок управления, который контролирует и управляет всеми системами в кузове автомобиля: электрические окна, электрические зеркала, электрические замки, системы безопасности и т.д.

**BDU (Body Distribution Unit)** – Коробка реле и предохранителей.

**BIC** – разъем/интерфейс для дополнительного аккумулятора.

**BID (Board Computer Info Display)** – Информационный дисплей бортового компьютера.

**CAN-Bus (Controller Area Network)** – контроллерная локальная сеть. Протокол шины CAN, приоритеты кадров данных, приоритеты, обработка ошибок, фильтры кадров данных.

**Car Pass** – паспорт автомобиля. В паспорте автомобиля содержатся сведения, связанные с безопасностью автомобиля. Он вручается покупателю при передаче автомобиля. В случае потери необходимо обращаться в «Центр технической помощи».

**CBC (Cornering Brake Control)** – Система контроля торможения на поворотах. Эта функция антиблокировочной тормозной системы, регулируя тормозные силы на каждом колесе, компенсирует де-

стабилизирующие моменты сил, стремящихся повернуть автомобиль вокруг вертикальной оси при торможении в повороте.

**CDC (Clutch Control Module)** – контроллер сцепления. Контролирует и управляет узлом сцепления на автомобиле.

**CDC (Continuous Damping Control)** – Система непрерывного управления демпфированием. Электронная система управления демпфированием, основу которой составляют четыре амортизатора с регулируемыми электромагнитными клапанами, непрерывно и точно управляет характеристиками амортизаторов с учетом состояния дорожного покрытия и индивидуального стиля вождения.

**CDC (Compact Disc Changer)** – CD-чейнджер.

**CDC (Continuous Damping Control)** – Система динамического управления амортизаторами, пример: Astra-H, МГ 2004,5 , Vectra-C, Zafira-B.

**CDLS (Central Door Locking System)** – Система центрального замка

**CDTI (Common Rail Diesel Turbo Injection)** – Турбодизель с общей топливной рампой

**Check Control** – функция контроля, пример: Omega-B, сообщения о неисправностях и приоритет индикации.

**CID (Color Info Display)** – Цветной информационный дисплей.

**CIM (Column Integration Module)** – Центр управления электроникой, расположенный на рулевой колонке. CIM обычно отвечает за следующие функции: управление подрулевыми переключателями, транспондерная антенна, замок зажигания, сигналы поворотов, управление круиз контролем, управление клавишами инфодисплея на руле, управление стеклоочистителями, звуковой сигнал.

**СКР (Crankshaft Position)** – датчик положения коленчатого вала.

**СМР (Camshaft Position)** – датчик положения распределительного вала.

**CNG (Compressed Natural Gas)** – Природный газ.

**Competitive Mode** – конкурентный режим, дополнительная функция электронной системы динамической стабилизации на модели Opel GT. Активизируется дополнительным подключением и обеспечивает полный контроль над задними колесами при выключенной системе управления тягой. Регулирование системы ESP остается активным.

**CRIP-2-MI (Common Rail ...)** – прецизионная форсунка для впрыска топлива на дизельных двигателях с системой Common Rail.

**CRP (Car phone)** – встроенный в радиосистему автомобильный телефон.

**Crash-Box** – деформируемая коробка – поглощает энергию удара, защищает детали кузова и узлы

**CSC (Corner Stability Control)** – Система стабилизации движения на поворотах.

**CSM (Column Switch Module)** – Блок выключателей на рулевой колонке

**CSV (Caster System Vehicle)** – система помощи при недостаточной поворачиваемости – система контроля курса – аналоги: UCL.

**CTI (Central Tire Inflation System)** – система централизованной подкачки шин.

**CVCP (Continuous Variable Camshaft Phasing)** – Система непрерывного регулирования фаз газораспределения Система непрерывного управления обоими распределительными валами для четырехцилиндровых двигателей с четырьмя клапанами на цилиндр.

**CVT (Continuously Variable Transmission)** – вариатор или постоянно изменяющаяся трансмиссия. В отличие от традиционного автомата бесступенчатые трансмиссии не имеют внутри набора разных шестерен, что означает отсутствие зацепляющихся друг с другом зубчатых колес.

**DAB (Digital Audio Broadcast)** – цифровое радиовещание.

**DBC (Dynamic Brake Control)** — Система динамического контроля за торможением.

**DCS (Downhill Control System)** – система облегчения езды на спусках. Служит для ограничения и поддержания постоянной скорости на крутых спусках.

**DDM (Driver Door Module)** – Контроллер двери водителя.

**DDS (Deflation Detection System)** – система контроля падения давления в шинах, Astra-H, МГ 2004,5 , Zafira-B.

**DIC (Driver Information Centre)** – информационный центр водителя, дисплей для сообщений на модели Opel GT.

**DID (Dual Info Display)** – Двухстрочный информационный дисплей, в зависимости от исполнения содержит индикаторы времени / даты / аудиосистемы или времени / температуры.

**DIS (Display Information System)** – информационный дисплей, ранее также «ID»; см. также «BID, DID, TID, MID, GID, CID»

**DLC (Diagnostic Link Connector)** – диагностический разъем, см. «TECH 1 / TECH 2».

**DME (Digital Motor Electronics)** — Цифровая электронная система управления работой двигателя. Осуществляет контроль за «правильным» зажиганием и впрыском топлива и другими дополнительными функциями, такими, как регулировка состава рабочей смеси.

**DOT (Department Of Transport)** – департамент транспорта США. Стандарт DOT 4 на тормозную жидкость.

**DPF (Diesel Particulate Filter)** – сажевый фильтр для дизельного двигателя.

**Drop-Down** – функция Автоматического опускания безрамного стекла (Astra Coupe) при открытии двери и автоматическое поднятие стекла при закрытии двери.

**DSC (Dynamic Stability Control)** — система динамического контроля устойчивости (см. ESP).

**DSG (Direct-Shift Gearbox (нем. Direkt-Schalt-Getriebe))** – преселективная коробка передач. Больше известная в обиходе как автоматическая коробка передач с двойным сцеплением. Аналоги у разных производителей могут звучать по-разному: PDK, SST, DCG, PSG, S-tronic.

**Driveline** – ведущий привод: AWD (All Wheel Drive) – все колёса ведущие; FWD (Front Wheel Drive) – передние ведущие колёса; RWD (Rear Wheel Drive) – задние ведущие колёса; 4WD-OD (Four Wheel Drive – On Demand) – четыре колеса ведущие (не постоянно включенный полный привод); 4WD-FT (Four Wheel Drive – Full Time) – четыре колеса ведущие (постоянно включенный полный привод).

**DSTC (Dynamic Stability and Traction Control)** — см. ESP.

**DSM (Driver Seat Module)** – Контроллер сиденья водителя

**DTC (Drag Torque Control)** – Контроль крутящего момента в поворотах.

**Dual Ram System** – двухканальная регулируемая впускная система, на одном валу расположены переключающие заслонки, которые управляются в зависимости от оборотов и нагрузки.

**DVD (Digital Versatile Disk)** – цифровой универсальный диск.

**DWA** – Система противоугонной сигнализации.

**E-Gas** – электронная педаль акселератора.  
**Easy-Entry** – сиденье.  
**Easytronic** – Автоматизированная механическая коробка передач, которая позволяет выбрать полностью автоматический режим переключения передач или переключать их вручную.

**EBD (Electronic Brake Distribution)** – В немецком варианте – EBV (Elektronische Bremskraftverteilung). Электронная система распределения тормозных сил. Обеспечивает наиболее оптимальное тормозное усилие на осях, изменяя его в зависимости от конкретных дорожных условий.

**ECC (Electronic Climate Control)** – Электронный климат-контроль.

**ECM (Engine Control Module)** – Контроллер системы управления двигателем. Микрокомпьютер задаёт продолжительность впрыска и количество впрыскиваемого топлива для каждого цилиндра. Способствует получению от двигателя оптимальной мощности и крутящего момента в соответствии с заложенной в него программой.

**ECO Service-Flex** – график техобслуживания.

**ECOTEC (Emission Consumption Optimization Technology)** – Технология оптимизации поглощения выхлопа.

**ECT (Engine Coolant Temperature)** – датчик температуры охлаждающей жидкости.

**ECT (Electronically Controlled Transmission)** – Электронная система управления переключениями передач в автоматических КПП последнего поколения.

**ECU (Electronic Control Unit)** – Контроллер (общее название) блок электронного управления работой двигателя.

**EDC (Electronic Damper Control)** – электронная система регулирования жесткости амортизаторов. Иначе ее можно назвать системой, заботящейся о комфорте.

**EDIS (Electronic Distributorless Ignition System)** – электронная бесконтактная система зажигания (без прерывателя — распределителя).

**EDL (Electronic Differential Lock)** – система электронной блокировки дифференциала. В немецком варианте EDS (Elektronische Differentialsperre) – электронная блокировка дифференциала. Представляет собой логичное дополнение к функциям антиблокировочной системы (АБС).

**EGR (Exhaust Gas Recirculation)** – Система рециркуляции выхлопных газов.

**EHPS (Electro Hydraulic Power Steering)** – Рулевое управление с электрогидравлическим сервоусилителем.

**EHS (Electrical Heater System)** – Дополнительный электронагреватель.

**EHU (Entertainment Head Unit)** – Главный блок аудиосистемы.

**EMS (Engine Management System)** – Электронная система управления двигателем.

**EOL (Engine Oil Level)** – датчик уровня моторного масла.

**EON (Enhanced Other Network)** – встроенная навигационная система.

**EOT (Engine Oil Temperature)** – датчик температуры моторного масла.

**EPB (Electronic Parking Brake)** – Электрический стояночный тормоз.

**EPS (Electronic Power Steering)** – Электроусилитель рулевого управления.

**ESCL (Electric Steering Column Lock)** – Электрический замок блокировки рулевой колонки.

**ESP (Electronic Stability Program)** – Она же ATTS, ASMS (Automatisches Stabilitäts Management System), DSTC, DSC (Dynamic Stability Control), FDR (Fahr-dynamik-Regelung), VDC, VSC (Vehicle Stability Control), VSA (Vehicle Stability Assist). Электронная система динамической стабилизации (противозаносная система (ПЗС)). Система контролирует и сравнивает приблизительно 25 раз в секунду режим движения автомобиля с характером воздействия на рулевое управление водителем. Опасность потери устойчивости, например, при недостаточном или чрезмерном воздействии на рулевое управление, немедленно обнаруживается датчиками ESP. Путем целенаправленного подтормаживания отдельных колес и вмешательства в управление двигателем автомобиль снова стабилизируется.

**ESP-Plus (Electronic Stability Program)** – Электронная система динамической стабилизации. Система обеспечения курсовой устойчивости новейшего поколения.

**ETC (Electronic Throttle Control)** – Электронная педаль управления положением дроссельной заслонки.

**ETCS (Electronic Throttle Control System)** – Электронная система управления положением дроссельной заслонки.

**EUC (Enhanced Understeering Control)** – Система контроля усиления поворота передних колес.

**EVAP (Evaporation System)** – система улавливания паров топлива.

**Flex Fix System** – Несущая конструкция для велосипедов, лыж, сноубордов и проч.

**Follow-me-home** – подсветка дороги домой.

**FSI (Fuel Stratified Injection)** – «послойный» впрыск топлива. Разработка специалистов Volkswagen..

**Full-Size Airbag** – полноразмерные надувные подушки водителя и переднего пассажира.

**FZM (Front Zone Module)** – Контроллер моторного отсека, см. также «UEC».

**GCU (Glow Plug Control Unit)** – контроллер свечей предпускового подогрева.

**GID (Graphic Info Display)** – Графический информационный дисплей (монокромный).

**GMLAN (General Motors Local Area Network)** – локальная сеть фирмы General Motors).

**GPS (Global Positioning System)** – спутниковая система определения местоположения, см. также «Навигационная система».

**GSM (Groupe Spécial Mobile)**, позже переименован в Global System for Mobile Communications) – глобальный стандарт цифровой мобильной сотовой связи с разделением каналов по времени и частоте.

**HBA (Hill Brake Assist)** –помощь при трогании в гору, кратковременно предотвращает скатывание назад автомобиля при трогании на подъеме – дополнительная функция системы ABS/ESP.

**HBA (Hydraulic Brake Assistant)** – гидравлический тормозной ассистент. Для помощи водителю поддерживает требуемое давление в тормозной системе при резком торможении.

**HDC (Hill Descent Control)** – Система контроля тяги для спуска с крутых и скользких уклонов. Работает примерно по тому же принципу, что и антипробуксовочная.

**HLA (Hill Launch Assist)** – Система помощи при трогании в гору или с горы.

**HPS (Hydraulic Power Steering)** – гидравлический сервоусилитель рулевого управления.

**HSA (Hill Start Assist)** – помощь при трогании в гору.

**HSCAN (High Speed Controller Area Network)** – высокоскоростная контроллерная локальная сеть CAN.

**HTC (Hard Top Control)** – Контроллер стального откидного люка в крыше.

**HVAC (Heating, Ventilation, Air Condition)** – Обогрев, вентиляция, кондиционер.

**IAT (Intake Air Temperature)** – датчик температуры впускного воздуха.

**IPBEC (Instrument Panel Body Electrical Centre)** – контроллер приборной панели и кузова), см. «IPBEC».

**IC (Inflatable Curtain)** – Надуваемая занавеска.

**IDS (Interactive Driving System)** – Интерактивная динамическая система вождения.

**IDSPLUS (Interactive Driving System Plus)** – Интерактивная система управления IDSPlus. Предлагаемая в качестве дополнительного оборудования система шасси с использованием новейших технологий, включающая электронную систему динамического управления амортизаторами CDC, спортивный режим переключения передач и объединяющая все динамические системы управления для достижения оптимального баланса между плавностью хода, динамическими характеристиками и безопасностью.

**IPBEC (Instrument Panel Body Electrical Centre)** – Электрический центр панели управления, контроллер салона.

**IPC (Instrument Panel Cluster)** – Комбинация приборов.

**ISS (Input Shaft Speed)** – датчик частоты оборотов входного вала.

**IVLAN (In Vehicle Local Area Network)** – автомобильная локальная сеть.

**LCD (Liquid Crystal Display)** – Жидкокристаллический дисплей.

**LHD (Left Hand Drive)** – автомобиль с левосторонним рулевым управлением (ЛП).

**LSCAN (Low Speed Controller Area Network)** – низкоскоростная контроллерная локальная сеть CAN.

**MAF (Mass Airflow sensor)** – датчик массового расхода воздуха.

**MATC (Mitsubishi Active Traction Control)** – активная противобуксовочная система Мицубиси.

**MAP (Manifold Absolute Pressure)** – абсолютное давление во впускном коллекторе.

**MID (Multi Info Display)** – Многострочный информационный дисплей.

**MIL (Malfunction Indicator Light)** – Контрольная лампа неисправности, связанная с системой встроенной диагностики, см. также «OBD».

**MKM** – модуль охлаждения двигателя.

**MSCAN (Mid Speed Controller Area Network)** – среднескоростная контроллерная локальная сеть CAN.

**MSR (Motor Schleppmoment Regelung)** – система контроля за торможением двигателем. Предотвращает блокирование ведущих колес при торможении двигателем.

**M/T (Manual Transmission)** – РКПП, ручная коробка переключения передач.

**MTA (Manual Transmission – Automatic Shift)** – механическая коробка передач с автоматическим переключением (автоматизированная механическая трансмиссия). См. также «Easytronic».

**MULTEC** – Система впрыска топлива. Представляет собой систему одноточечного (центрального) прерывистого впрыска.

**NHU (Navigation Head Unit)**, главный блок навигационной системы, см. «Навигационная система».

**NTC (Negative Temperature Coefficient)** – датчик с отрицательным температурным коэффициентом (постоянным множителем при переменной величине).

**OBD (On-Board Diagnostics)** – встроенная система бортовой диагностики автомобиля.

**OHC (Overhead Camshaft)** – расположенный сверху распределительный вал

**OnStar** – система обслуживания с применением мобильной связи и GPS.

**OPC (Opel Performance Center)** – центр усовершенствования Opel), [www.opel-performance.de](http://www.opel-performance.de). См. также «OSV» и национальные

веб-сайты Opel – веб-сайт, [www.opel.com](http://www.opel.com), см. также национальные веб-сайты.

**Opel Fix** – типы детских сидений безопасности

**Open & Start** – система управления центральным замком и пуском двигателя.

**OSRVM (Out Side Rear View Mirror)** – Наружные зеркала.

**OSS (Output Shaft Speed)** – датчик частоты оборотов выходного вала на Antara с автоматической коробкой передач AISIN 55-51 LE

**OSV (Opel Special Vehicles)** – специальные автомобили Opel, [www.opel-osv.de](http://www.opel-osv.de), См. также «ОПС» и национальные веб-сайты

**PAM (Park Assist Module)** – Система помощи при парковке. см. «PAS».

**PAS (Parking Assist System)** – Система облегчения парковки, см. «Система помощи при парковке».

**Parkpilot** – парковочный пилот, см. «Система помощи при парковке».

**PCV (Positive Crankcase Ventilation)** – клапан вентиляции картера.

**PDA (Port Deactivation)** – Система управления сечением впускного коллектора. Управление геометрией впускного коллектора при помощи дроссельных клапанов, установленными в одном из двух впускных портов каждого клапана. PDA – так называлась эта система в первых разработках OPEL. В дальнейшем технология была переименована в TWINPORT.

**PDM (Passenger Door Module)** – Контроллер двери пассажира.

**Ped-Pro (Pedestrian Protection)** – Защита пешехода.

**PHS (Parking Heater System)** – Стояночный обогреватель.

**PMM (Power Mode Master)** – режим электропитания, функция контроллеров. Связывает различные функции и данные разных контроллеров. Контролируется и управляется контроллером кузова (BCM).

**Power Sounder** – специальная мощная сирена для автомобилей Vauxhall.

**PRS (Pedal Release System)** – автоматически отсоединяющийся (травмобезопасный) блок педалей.

**PSOC (Passenger Seat Occupant Classification)** – Система распознавания занятости сиденья переднего пассажира.

**PTS (Parktronic System)** – в немецком варианте ADK (Abstandsdistanzkontrolle) — система контроля дистанции при парковке, которая посредством ультразвуковых сенсоров, расположенных в бамперах, определяет расстояние до ближайшего препятствия.

**PTU (Power Take-Off Unit)** – раздаточная коробка.

**PWM (Pulse Width Modulation)** – Широтно-импульсная модуляция (ШИМ).

**RAP (Retained Accessory Power)** – резервное электропитание для принадлежностей. Служит для поддержания различных функций в автомобиле и после выключения зажигания.

**RDC (Reifen Druck Control)** – Система контроля за давлением воздуха в шинах.

**RDM (Rear Drive Module)** – модуль заднего привода. Узел для распределения тягового усилия на задние колеса.

**REC (Rear Electrical Centre)** – Задний центр управления электроникой.

**RFA (Remote Function Actuation)** – приемный блок для дистанционного управления центральным замком.

**RHD (Right Hand Drive)** – Автомобиль с правосторонним рулевым управлением (ПР)

**RSA (Rear Seat Audio)** – Аудиосистема для пассажиров на заднем сиденье.

**RSM (Rain Sensor Module)** – Модуль датчика дождя.

**Run Flat Tire** – шины, не боящиеся проколов.

**RZM (Rear Zone Module)** – Контроллер задней части автомобиля, см. также REC.

**SAC (Semi-Automatic Climate Control)** – полуавтоматический климат-контроль.

**SAC (self adjusting clutch)** – Сцепление с автоматической регулировкой рабочего зазора.

**SADS (Semi-Active Damping System)** – полуактивная система амортизации, см. «CDC».

**SAFETEC (Safety System)** – Система безопасности SAFETEC.

**SBE** – система распознавания занятости сиденья.

**SDM (Sensing & Diagnostic Module)** – модуль датчиков и диагностики.

**Security Code** – защитный код, см. «WFS (иммобилизатор) с транспондером».

**Servotronic** – Рулевое управление, регулируемое в зависимости от скорости автомобиля.

**SIMTEC 56** – система приготовления горючей смеси.

**SIPS (Side Impact Protection System)** — Система защиты от бокового удара. Состоит из усиленных и энергопоглощающих элементов кузова и боковых подушек безопасности, которые обычно располагаются во внешнем крае спинки переднего сидения.

**Skyhook** – принцип работы амортизаторов.

**SLM (Shift Lever Module)** – Контроллер селектора переключения передач.

**SLS (Straight-Line Stability Control)** – Система поддержания прямолинейности движения.

**SLS (Self-Levelizing Suspension)** — Система самовыравнивания подвески. Особая конструкция амортизаторов и/или пневморессор.

**SOR (Seat Occupancy Recognition)** – система распознавания занятости сиденья. См. также SBE (распознавание занятости сиденья).

**Sport Mode** – спортивный режим.

**SRM (Sun Roof Module)** – Контроллер сдвижного люка в крыше.

**SSPS (Speed Sensitive Power Steering)** – сервоусилитель рулевого управления с зависимым от скорости автомобиля передаточным числом.

**SVM (Special Vehicle Module)** – Контроллер для автомобилей специального назначения.

**SWCAN (Single Wire CAN)** – Однопроводная шина CAN.

**TC (Traction Control), TC-Plus** – система управления тягой.

**T** **TCM (Transmission Control Module)** – Контроллер автоматической коробки передач.

**TCS (Traction Control System)** – Система управления тягой.

**TDI (Turbocharged Direct Injection)** – турбонагнетатель с непосредственным впрыском. Турбонаддув двигателя.

**Tec shift** – обозначение автоматизированной коробки передач (МТА) на моделях Vivaго и Movano.

**Telefon-Mute** – автоматическое отключение остальных источников аудиосигнала во время разговора по телефону.

**Tempomat** – устройство, поддерживающее постоянную скорость автомобиля, автоматически прибавляя газ при снижении скорости движения и уменьшая при её увеличении, к примеру, на спусках, без участия водителя. Удобен в дальних дорогах, когда утомительно удерживать на большом протяжении времени педаль газа в одном и том же положении. См. «Круиз-контроль»

**TFT (Transmission Fluid Temperature)** – датчик температуры трансмиссионного масла.

**Thorax** – подушка безопасности, см. «Full-Size Airbag».

**TID (Triple Info Display)** – Трехстрочный информационный дисплей. Монохромная (одноцветная) индикация.

**TMS (Telematics System)** – Система телематики.

**TP (Throttlevalve Positioning)** – датчик положения дроссельной заслонки. Пример: Astra-G.

**TPMS (Tire Pressure Monitoring System)** – Система контроля давления воздуха в шинах.

**TR (Traction Range)** – датчик ступени коробки передач..

**Travel Assistant** – Спутник путешественника.

**TSA (Trailer Stability Assist)** – система стабилизации прицепа. Система функционирует совместно с системой ESPPlus и следит за интенсивностью рыскания автопоезда. см. TSP (Trailer Stability Program).

**TSP (Trailer Stability Program)** – Система стабилизации прицепа, предотвращает опасность срыва и повреждения прицепа путем целенаправленного автоматического притормаживания отдельных колес – дополнительная функция системы ABS/ESP.

**Twin Audio** – см. «Информационно-развлекательная система»

**TWINPORT (TWINPORT)** – Технология TWINPORT. Остроумная технология, разработанная компанией Opel для бензиновых двигателей рабочим объемом до 1,6 литра с четырьмя клапанами на цилиндр, позволяющая экономить топливо.

**TXV (Thermostatic eXpansion Valve)** – термостатически регулируемый расширительный клапан, см. «Кондиционер».

**UART (Universal Asynchronous Receive Transmit)** –

**U** двунаправленная последовательная шина данных. См. также «DWA».

**UAS (Ultrasonic Alarm System)** – Система ультразвуковой сигнализации.

**UCL (Understeering Control Logic)** – Логическая схема контроля недостаточного рулевого воздействия, предотвращает недостаточное воздействие рулевого управления на поворотах – дополнительная функция системы ABS/ESP.

**UEC (Underhood Electrical Centre)** – Контроллер моторного отсека.

**UHP (Universal Handsfree Phone)** – Универсальное устройство громкой связи, контроллер блока телефона.

**UID (User Identification Device)** – Брелок идентификации владельца, электронный ключ, а также резервный ключ, часть системы «Open & Start».

**Vauxhall** – веб-сайт, [www.vauxhall.com](http://www.vauxhall.com), см. также национальные веб-сайты.

**V**

**VDC (Vehicle Dynamic Control)** – система стабилизации курсовой устойчивости.

**VIN (Vehicle Identification Number)** – Номер шасси (идентификационный номер автомобиля), пример: Astra-H.

**VNT Turbolader (Variable Nozzle Turbines)** – Турбонагнетатель с электропневматическим регулированием впускного аппарата.

**VSS (Vehicle Speed Sensor)** – датчик скорости автомобиля.

**VVT-i (Valve Variable Timing-intelligent)** – Или VTEC, VANOS, NVCS, VIS, CVVT. Системы изменяемых фаз газораспределения. Применяются для улучшения характеристик крутящего момента в широком диапазоне оборотов, а также для улучшения экономичности и экологических характеристик двигателя.

**WFS (Web Feature Service)** – Веб-Сервер. Инструмент для получения или изменения описания функций.

**WFS** – иммобилайзер (иммобилизатор) – вид электронного противоугонного устройства. Предназначен надежно блокировать некоторые из модулей машины или другого транспорта. Первичной задачей иммобилайзера является недопущения к управлению посторонних лиц, в том числе при попытках угона.

**WHIPS (Whiplash Protection System)** — Механическая система безопасности, которая встроена в оба передних сидения.

**X** **XBCM (Body Control Module – Export)** – контроллер системы противоугонной сигнализации на модели Antara.

**Y** **YRS (Yaw Rate Sensor)** – Датчик угловой скорости рыскания, Датчик ускорения, см. также «ESP».

**Z** **ZV** – центральный замок, см. также «Устройство дистанционного радиуправления».

**CAN словарь**

Словарь позволит каждому компетентно разобраться в свойствах и особенностях CAN технологий. Словарь дает возможность не только понять CAN терминологию, но и уяснить многие тонкости и особенности работы CAN сетей.

Статьи словаря размещены в порядке английского алфавита. Для терминов, оставленных в латинском написании, приведена рекомендованная русскоязычная транскрипция.

**A**

<i>acceptance filter</i> <b>приемный фильтр</b>	Приемный фильтр CAN контроллера служит для селекции входных сообщений в зависимости от их идентификаторов. Большинство микросхем CAN контроллеров поддерживают аппаратный приемный фильтр, который производит отбор CAN сообщений с определенным идентификатором или группой идентификаторов. Задаваемая пользователем фильтрация разгружает микроконтроллер от дополнительного выполнения функций приемного фильтра.
<i>acknowledge (ACK) delimiter</i> <b>разделитель подтверждения</b>	Второй бит поля подтверждения CAN кадра. Является рецессивным. Доминантное значение разделителя считается нарушением формата и вызывает передачу кадра ошибки.
<i>acknowledge error</i> <b>ошибка подтверждения</b>	Обнаружение передающим узлом рецессивного состояния в слоте подтверждения интерпретируется как ошибка подтверждения. Ошибки подтверждения не вызывают состояния отключения от шины. Как правило, такие ошибки происходят, если в сети имеется лишь один узел и он начинает передачу CAN сообщений.
<i>acknowledge (ACK) field</i> <b>поле подтверждения</b>	Поле подтверждения состоит из двух бит: слота подтверждения и разделителя подтверждения.
<i>acknowledge (ACK) slot</i> <b>слот подтверждения</b>	Первый бит поля подтверждения CAN кадра. Устанавливается рецессивным со стороны передающего узла и доминантным со стороны всех получателей, которые осуществили успешную проверку CRC (циклического избыточного кода). Если передающий узел обнаруживает доминантное состояние этого бита, он может быть уверен, что хотя бы один узел принял сообщение без ошибок.

<i>active error flag</i> <b>активный флаг ошибки</b>	Активный флаг ошибки является начальной частью активного кадра ошибки и состоит из шести последовательных доминантных бит.
<i>application layer</i> <b>прикладной уровень</b>	Прикладной уровень является составной частью семиуровневой эталонной модели OSI (интерфейс открытых систем). Он обеспечивает коммуникационную поддержку прикладных программ.
<i>application objects</i> <b>прикладные объекты</b>	Прикладными объектами являются функции и параметры прикладной программы, которые видимы для интерфейса программирования (API) прикладного уровня.
<i>application profile</i> <b>прикладной профиль</b>	Прикладной профиль определяет все коммуникационные и прикладные объекты для каждого устройства CAN сети.
<i>arbitration field</i> <b>поле арбитража</b>	Поле арбитража включает в себя 11 либо 29 битовый идентификатор, бит RTR (удаленный запрос), а в случае расширенного формата кадра также биты SRR (замена удаленного запроса) и IDE (флаг расширенного формата).
<i>assembly object</i> <b>сборочный объект</b>	Объект протокола DeviceNet, определяющий содержимое сообщения ввода-вывода.
<i>asynchronous PDO</i> <b>асинхронный PDO</b> <i>[пэдэо]</i>	Передача асинхронного PDO производится при возникновении определенного внутреннего события. Таковым может быть, в частности, истечение таймера PDO события. При получении асинхронного PDO прикладная программа должна немедленно обновить в объектном словаре отображаемые этим PDO объекты.
<i>automatic retransmission</i> <b>автоматическая повторная передача</b>	Повторная передача искаженных сообщений (кадров данных и удаленного запроса) производится автоматически после успешной передачи кадров ошибок.
<i>auto bit rate detection</i> <b>автодетектирование скорости</b>	В режиме автодетектирования скорости каждый CAN узел прослушивает сетевой трафик и при обнаружении не искаженного сообщения подтверждает прием кадра. Если такового не обнаружено, CAN узел автоматически переключается на следующую predetermined скорость. При выполнении этой процедуры в сети должен быть только один передающий узел. Автодетектирование скорости поддерживается рядом CAN контроллеров. Этого можно достичь и с помощью дополнительной внешней схемотехники.

# В

<i>Bandwidth</i> <b>Полоса пропускания</b>	Полоса пропускания определяет величину, которая задает объем передаваемой в единицу времени информации.
<i>BasicCAN</i> <b>BasicCAN [бэйсиккан]</b>	Термин, применявшийся в начале становления CAN технологий. Описывает реализацию, которая использует только два приемных буфера сообщений, заполняемых и считываемых попеременно.
<i>basic frame format</i> <b>основной формат кадра</b>	В кадрах данных и кадрах удаленного запроса основного формата используются 11 битовые идентификаторы.
<i>basic cycle</i> <b>основной цикл</b>	В системе TTCAN основной цикл всегда начинается с опорного сообщения, за которым следуют несколько окон: явных, арбитража либо пустых. Один или несколько основных циклов формируют матричный цикл TTCAN.
<i>bit encoding</i> <b>кодировка бит</b>	В CAN биты кодируются без возврата к нулю (NRZ код).
<i>bit error</i> <b>ошибка бита</b>	Случай, когда бит передается доминантным уровнем, а принимается рецессивным или наоборот, рассматривается как ошибка бита и вызывает передачу кадра ошибки в очередном битовом интервале. Такая ситуация, однако, не является ошибочной во время передачи поля арбитража или слота подтверждения.
<i>bit monitoring</i> <b>мониторинг бит</b>	Каждый передающий CAN контроллер прослушивает сеть и при этом осуществляет мониторинг состояния бит.
<i>bit rate</i> <b>битовая скорость</b>	Определяет число передаваемых бит за единицу времени, независимо от их представления. Битовая скорость в CAN сетях ограничена значением 1 мегабит в секунду.
<i>bit resynchronization</i> <b>битовая ресинхронизация</b>	Ограниченная точность опорных генераторов, может привести к выходу из синхронизации какого-либо узла. CAN контроллер осуществляет ресинхронизацию по каждому переходу сигнала с рецессивного на доминантный уровень.
<i>bit stuffing</i> <b>бит стаффинг</b>	Вставка дополнительных бит в поток данных для обеспечения смены уровня сигнала на шине и возможности периодической ресинхронизации.
<i>bit time</i> <b>битовое время</b>	Длительность одного бита.

<i>bit timing</i> <b>битовое хронирование</b>	Установка регистров битового хронирования CAN контроллера определяется квантом времени, который зависит от частоты опорного генератора и параметров делителя (pre-scaler) битовой скорости узла.
<i>Bridge</i> <b>мост</b>	Устройство, обеспечивающее связь двух сетей на канальном уровне.
<i>broadcast transmission</i> <b>широковещательная передача</b>	Коммуникационный протокол, выполняющий одновременную передачу данных от одного узла ко всем остальным.
<i>boot-up message</i> <b>сообщение загрузки</b>	Коммуникационное сообщение CANopen, передаваемое при переходе узла сети в предоперационное состояние после инициализации.
<i>Bus</i> <b>шина</b>	Топология коммуникационной сети, при которой все узлы объединяются посредством пассивных каналов связи, позволяющих вести передачу в обоих направлениях.
<i>bus access</i> <b>доступ к шине</b>	Если шина свободна, любой узел сети может начинать передачу кадра. В сети CAN доступ к шине инициируется передачей доминантного SOF бита (бит начала кадра).
<i>bus analyzer</i> <b>анализатор шины</b>	Инструментарий, позволяющий отслеживать и отображать передаваемые по сети данные. Анализаторы шины могут работать на физическом, канальном или различных прикладных уровнях, например, CANopen или DeviceNet.
<i>bus arbitration</i> <b>арбитраж шины</b>	Процедура арбитража необходима для разрешения ситуаций, когда несколько узлов пытаются получить одновременный доступ к шине. После ее выполнения доступом к шине обладает лишь один узел. В CAN протоколе используется алгоритм арбитража CMA/CD (множественный доступ с обнаружением несущей / детектирование коллизий) совместно с AMP (арбитраж по приоритету сообщения). Этот протокол позволяет проводить арбитраж без разрушения сообщений.
<i>bus comparator</i> <b>компаратор шины</b>	Элемент, который осуществляет преобразование физических сигналов, распространяемых в коммуникационной среде, в логическую информацию или данные.
<i>bus driver</i> <b>драйвер шины</b>	Элемент, который осуществляет преобразование логической информации или данных в физические сигналы, которые могут передаваться в коммуникационной среде

<i>bus idle</i> <b>шина свободна</b>	При свободном состоянии шины передача CAN кадра не производится, а все узлы сети выставляют рецессивные биты.
<i>bus latency</i> <b>латентность шины</b>	Представляет собой время между возникновением запроса на передачу и передачей SOF бита (бит начала кадра). В CAN сетях максимальная латентность составляет: время одного сообщения минус длительность одного бита.
<i>bus length</i> <b>длина шины</b>	Представляет собой длину сетевого кабеля между двумя терминаторами. Длина шины для CAN сетей ограничена используемой скоростью передачи. При скорости 1 Мбит/с максимальная длина шины теоретически составляет 40 метров. При меньших скоростях передачи возможная длина шины возрастает: при 50 Кбит/с она может достигать 1 километра.
<i>Busload</i> <b>загрузка шины</b>	Загрузка шины определяется отношением числа передаваемых бит к числу бит, когда шина свободна, за определенный промежуток времени.
<i>bus monitoring mode</i> <b>режим мониторинга шины</b>	В режиме мониторинга CAN контроллер отключает вывод Tx. Это означает, что не могут передаваться флаг ошибки и слот подтверждения (ACK).
<i>bus-off state</i> <b>состояние bus-off [басоф]</b>	Переключение CAN контроллеров в состояние bus-off (отключение от шины) производится, когда счетчик ошибок передачи (TEC) достигает значения 255. При нахождении в состоянии bus-off, CAN контроллеры передают только рецессивные биты.
<i>bus state</i> <b>состояние шины</b>	Одно из двух дополнительных состояний шины: доминантное или рецессивное.

## С

<b>CAN</b> <b>CAN [кан]</b>	Сеть контроллеров CAN является системой на основе последовательной шины. Первоначально была разработана фирмой Роберт Бош GmbH. Сеть CAN утверждена международным стандартом ISO 11898-1. CAN реализована многими производителями микроэлектроники.
<i>CANaerospace</i> <b>CANaerospace [кан-аэро-спэйс]</b>	Протокол высокого уровня для авиационных и аэрокосмических приложений.
<i>CAN Application Layer</i> <b>(CAL)</b> <b>прикладной уровень CAN</b>	Прикладной уровень, разработанный членами CiA (CAN в автоматизации). Поддерживает ряд коммуникационных сервисов и соответствующих протоколов.

<i>CAN common ground</i> <b>общее заземление CAN</b>	Для каждой сети CAN необходимо наличие общего заземления, что позволяет избежать проблем подавления синфазного сигнала. Однако, это создает условия для возникновения нежелательных контуров тока по общей земле.
<i>CAN device</i> <b>CAN устройство</b>	Устройство с CAN интерфейсом.
<i>CAN_H</i> <b>CAN_H [кан хай]</b>	Линия CAN_H трансивера, соответствующего стандарту ISO 11898-2, находится в рецессивном состоянии при напряжении 2.5 V и в доминантном при 3.5 V.
<i>CAN in Automation</i> <b>(CiA) CiA [циа]</b>	CiA (CAN в автоматизации) является международным сообществом пользователей и производителей решений на основе CAN. CiA поддерживает также ряд протоколов высокого уровня. Основано в 1992 году.
<i>CAN Kingdom</i> <b>CAN Kingdom [кан киндэм]</b>	Структура высокоуровневого протокола, оптимизированная для глубоко встроенных сетей. CAN Kingdom подходит, в частности, для приложений реального времени.
<i>CAN_L</i> <b>CAN_L [кан лоу]</b>	Линия CAN_L трансивера, соответствующего стандарту ISO 11898-2, находится в рецессивном состоянии при напряжении 2.5 V и в доминантном при 1.5 V.
<i>CAN Message Specification (CMS)</i> <b>спецификация CAN сообщения</b>	Часть спецификации прикладного уровня CAN, определяющая коммуникационные сервисы.
<i>CAN module</i> <b>CAN модуль</b>	Реализация контроллера CAN протокола совместно с аппаратным фильтром входных кадров и буферами сообщений в составе микроконтроллера или специализированной интегральной схемы (ASIC).
<i>CAN node</i> <b>CAN узел</b>	Синоним CAN устройства.
<i>CANopen</i> <b>CANopen [каноупэн]</b>	Семейство профилей, используемых для построения встроенных сетей в машиностроении, медицинском оборудовании, автоматизации зданий (системы управления лифтами, электронными дверьми, интегрированные системы контроля помещений), в железнодорожной и морской электронике, грузовых и внедорожных автомобилях и т.д.

<i>CANopen application layer</i> <b>прикладной уровень</b> <b>CANopen</b>	Прикладной уровень и коммуникационный профиль CANopen определяются стандартом EN 50325-2. Этот стандарт описывает коммуникационные сервисы и объекты. Кроме того, он дает спецификацию объектного словаря и протокола управления сетью (NMT).
<i>CANopen manager</i> <b>CANopen менеджер</b>	CANopen менеджер отвечает за управление сетью. В CANopen устройстве, поддерживающем такое управление, находится NMT мастер (мастер управления), менеджер SDO (сервисный объект данных) и конфигурационный менеджер.
<i>CANopen Safety</i> <b>безопасный CANopen</b>	Коммуникационный протокол, позволяющий осуществлять безопасную передачу данных. Для его реализации достаточно одной физической CAN сети. Избыточность достигается путем передачи каждого сообщения дважды с побитно инвертированными данными и идентификаторами, различающимися как минимум двумя битами.
<i>CAN protocol controller</i> <b>контроллер CAN протокола</b>	Контроллер CAN протокола является составной частью CAN модуля и осуществляет инкапсуляцию/декапсуляцию данных, битовое хронирование, подсчет CRC, бит стаффинг, обработку ошибок, локализацию отказов и др.
<i>CAN transceiver</i> <b>CAN трансивер</b>	CAN трансивер служит сопряжению CAN контроллера и физической шины. Он обеспечивает прием и передачу сигнала в линию. Существуют высокоскоростные, отказоустойчивые и однопроводные трансиверы, а также трансиверы для силовых линий и опто-волоконных кабелей.
<i>CANPL</i> <b>CANPL [канпл]</b>	Язык программирования доступа к CAN (CANPL) используется при разработке инструментария для анализа CAN сетей.
<i>CCP</i> <b>CCP [сисипи]</b>	CAN протокол калибровки (CCP) используется для обмена калибровочными данными в системах автомобильных двигателей.
<i>Certification</i> <b>сертификация</b>	Официальный тест соответствия компонентов и устройств какому-либо стандарту. Университет прикладных наук Германии проводит сертификацию микросхем CAN контроллеров. Организация ODVA официально сертифицирует изделия DeviceNet, а CiA осуществляет официальную сертификацию CANopen устройств.

<i>CiA 102</i> <i>CiA 102 [циа 102]</i>	Проектный стандарт для высокоскоростной передачи данных в соответствии с ISO 11898-2, предусматривающий использование 9- выводных D-sub разъемов.
<i>CiA 201..207</i> <i>CiA 201..207 [циа 201..207]</i>	Спецификации прикладного уровня CAN, определяющие CMS, DBT, NMT и LMT сервисы и протоколы.
<i>CiA 301</i> <i>CiA 301 [циа 301]</i>	Спецификация прикладного уровня и коммуникационного профиля CAN, определяющая функциональность CANopen NMT (протокол управления сетью) slave устройств.
<i>CiA 302</i> <i>CiA 302 [циа 302]</i>	План проектного стандарта для программируемых CANopen устройств. Включает в себя функциональность CANopen менеджера, динамические SDO соединения, стандартизированную процедуру загрузки NMT slave устройств, а также загрузку программ.
<i>CiA 303</i> <i>CiA 303 [циа 303]</i>	Проектные рекомендации для CANopen, касающиеся типов кабеля и назначения выводов разъемов, кодировки префиксов и единиц международной системы СИ, а также использования светодиодов.
<i>CiA 304</i> <i>CiA 304 [циа 304]</i>	Спецификация безопасного CANopen протокола, одобренная полномочными органами Германии и отвечающая требованиям приложений SIL класса 3 в соответствии со стандартом МЭК 61508.
<i>CiA 305</i> <i>CiA 305 [циа 305]</i>	Сервис установки сетевого уровня (LSS) определяет, каким образом назначать номер узла и скорость передачи с использованием CANopen сети.
<i>CiA 306</i> <i>CiA 306 [циа 306]</i>	План проектного стандарта определяет формат и содержимое электронных спецификаций (EDS), используемых в конфигурационном инструментарии.
<i>CiA 308</i> <i>CiA 308 [циа 308]</i>	Технический доклад CANopen, который определяет временные измерения, такие как время оборота PDO, джиттер объекта синхронизации и время отклика SDO. Кроме того, этот доклад определяет стандартную загрузку шины.
<i>CiA 309</i> <i>CiA 309 [циа 309]</i>	Набор спецификаций, определяющий службы и протоколы сети TCP/ IP, сопряженной с CANopen сетью. Также определены протоколы для сети ModbusTCP и протоколы ASCII команд.
<i>CiA 401</i> <i>CiA 401 [циа 401]</i>	CANopen профиль для модулей ввода/вывода общего назначения определяет цифровые и аналоговые устройства ввода/вывода.

<i>CiA 402</i> <i>CiA 402 [циа 402]</i>	CANopen профиль контроллеров движения и приводов определяет интерфейс для инверторов частоты, серво-контроллеров, а также шаговых двигателей.
<i>CiA 404</i> <i>CiA 404 [циа 404]</i>	CANopen профиль измерительных устройств и контроллеров с обратной связью. Поддерживает также многоканальные устройства.
<i>CiA 405</i> <i>CiA 405 [циа 405]</i>	CANopen профиль устройств и интерфейсов для IEC 61131-3 совместимых контроллеров, основанный на спецификации CiA DSP 302. Используется отображение сетевых параметров в PDO, функциональные блоки для SDO обмена и т.д.
<i>CiA 408</i> <i>CiA 408 [циа 408]</i>	Профиль CANopen устройства для гидравлических контроллеров и пропорциональных клапанов. Совместим с независимым от сети VDMA профилем.
<i>CiA 410</i> <i>CiA 410 [циа 410]</i>	Профиль CANopen устройства для уклонометров. Поддерживает как 16-и, так и 32-х разрядные датчики.
<i>CiA 412</i> <i>CiA 412 [циа 412]</i>	Профили CANopen устройств для медицинского оборудования. Определяют интерфейсы рентгеновских коллиматоров, генераторов рентгеновского излучения, медицинских стоек и столов.
<i>CiA 413</i> <i>CiA 413 [циа 413]</i>	Профиль CANopen интерфейса для шлюзов внутренних сетей грузовых автомобилей. Определяет шлюзы к ISO 11992, J1939 и другим внутренним сетям. Сеть CANopen как правило используется для управления дополнительным оборудованием грузовиков или трейлеров, например в мусороуборочных машинах, подъемных кранах, бетономешалках.
<i>CiA 414</i> <i>CiA 414 [циа 414]</i>	Профиль CANopen устройства для ткацких машин. Определяет интерфейс подающего механизма.
<i>CiA 415</i> <i>CiA 415 [циа 415]</i>	Прикладной профиль CANopen для асфальтоукладчиков. Определяет интерфейсы различных устройств, используемых в дорожно-строительной технике.
<i>CiA 416</i> <i>CiA 416 [циа 416]</i>	Прикладной профиль CANopen для дверей зданий. Определяет интерфейсы замков, сенсоров и других устройств, используемых в управляемых электроникой дверях.
<i>CiA 417</i> <i>CiA 417 [циа 417]</i>	Прикладной профиль CANopen для лифтов. Определяет интерфейсы контроллеров кабины, дверей, контроллера вызова и других, а также узлов кабины и дверей, кнопочных панелей, дисплеев и др.

<i>CiA 418</i> <b>CiA 418 [циа 418]</b>	Профиль CANopen устройства для батарейных модулей. Определяет интерфейс для работы с зарядными устройствами.
<i>CiA 419</i> <b>CiA 419 [циа 419]</b>	Профиль CANopen устройства для зарядных устройств. Определяет интерфейс для работы с батарейными модулями.
<i>CiA 420</i> <b>CiA 420 [циа 420]</b>	Семейство профилей CANopen устройства для экструдеров. Определяет интерфейсы для вытягивателя, гофрирователя и распиловщика.
<i>CiA 422</i> <b>CiA 422 [циа 422]</b>	Прикладной профиль CANopen для муниципальных транспортных средств. Используется, в частности, для управления дополнительным оборудованием мусороуборочных машин и определяет интерфейсы таких подсистем, как прессовщик, весы и др.
<i>CiA 425</i> <b>CiA 425 [циа 425]</b>	CANopen профиль для вспомогательных медицинских устройств. Определяет plug-and-play интерфейсы инжектора контрастного вещества и электрокардиографа.
<i>Client SDO</i> <b>SDO клиента [эсдэо]</b>	Иницирует и производит обмен данными с помощью SDO протокола путем чтения или записи объектного словаря серверного устройства.
<i>client/server communication</i> <b>режим клиент/сервер</b>	При передаче данных в режиме клиент/сервер инициализация связи осуществляется клиентом. Этот обмен всегда реализуется методом точка-точка.
<i>COB-ID</i> <b>COB-ID [кобид]</b>	COB-ID является объектом, определяющим CAN идентификатор, а также дополнительные параметры, такие, как действительный / недействительный и поддержка удаленного запроса.
<i>communication object (COB)</i> <b>коммуникационный объект</b>	Коммуникационным объектом называется одно или несколько CAN сообщений с определенным назначением, например, PDO, SDO, срочное сообщение, временная метка или сообщения контроля ошибок.
<i>communication profile</i> <b>коммуникационный профиль</b>	Коммуникационный профиль определяет содержимое коммуникационных объектов CANopen, таких, как срочное сообщение, временная метка, объекты синхронизации, сердцебиения, NMT объекты и др.
<i>Configuration Manager</i> <b>Конфигурационный Менеджер</b>	Конфигурационный Менеджер (CMT) поддерживает механизмы конфигурирования CANopen устройств при загрузке.

<i>confirmed communication</i> <b>обмен с подтверждением</b>	Для обмена с подтверждением необходима двунаправленная передача информации, когда узел-получатель высылает подтверждение успешного приема сообщения.
<i>conformance test plan</i> <b>план теста соответствия</b>	Определение протоколов тестирования, которые должны быть успешно пройдены для того, чтобы было достигнуто соответствие коммуникационному стандарту. План теста соответствия для CAN стандартизирован ISO 16845.
<i>conformance test tool</i> <b>инструментарий теста соответствия</b>	Инструментарием теста соответствия является реализация плана теста соответствия.
<i>Connector</i> <b>разъем</b>	Электро-механический компонент, используемый для подсоединения устройства к шине CAN, либо для удлинения шины. Назначение выводов CAN разъема для CAN и CANopen определяется CiA, а для DeviceNet – ODVA.
<i>Consumer</i> <b>потребитель</b>	Потребителем называется получатель сообщения в CAN сетях. Это подразумевает, что для данного сообщения приемный фильтр открыт.
<i>contention-based arbitration</i> <b>арбитраж на основе состязания</b>	Процедура арбитража на основе множественного доступа с обнаружением несущей (CMSA), при которой попытка захвата шины осуществляется многими узлами одновременно и для получения доступа проводится их состязание.
<i>control field</i> <b>контрольное поле</b>	Контрольное поле в кадрах данных и удаленного запроса имеет длину 6 бит и включает четыре бита длины поля данных, бит флага расширенного идентификатора и зарезервированный бит.
<i>CRC</i> <b>CRC [цээрцэ]</b>	Циклический избыточный код (CRC) в CAN кадрах данных и удаленного запроса является числом, рассчитанным на основе содержимого блока данных и предназначенным для контроля искажений этих данных. Рассчитав CRC для блока принятых данных и сравнив его со значением, полученным от передающей стороны, принимающая сторона может определить некоторые типы ошибок передачи данных.
<i>CRC delimiter</i> <b>CRC разделитель [цээрцэ]</b>	Бит CRC разделителя является последним битом поля CRC в кадрах данных и удаленного запроса. Этот бит всегда устанавливается рецессивным уровнем.

<i>CRC error</i> <b>ошибка CRC</b> <i>[цээрцэ]</i>	Ошибкой CRC является ненулевой результат подсчета циклического кода на стороне приемного CAN узла. При обнаружении этой ошибки вслед за полем подтверждения в сеть передается кадр ошибки.
<i>CRC field</i> <b>поле CRC</b> <i>[цээрцэ]</i>	Поле CRC в кадрах данных и удаленного запроса включает в себя 15 битовую CRC последовательность и 1 бит CRC разделителя. CRC последовательность позволяет обнаружить до пяти случайно распределенных искажений отдельных бит в полях начала кадра, арбитража, контрольном поле, поле данных или до 15 бит пакетной ошибки. Расстояние Хэмминга определено равным 6, без учета бит стаффинга.
<i>CSMA/CD+AMP</i> <b>CSMA/CD+AMP</b> <i>[сиэс-эмэй/сиди + амн]</i>	В CAN используется алгоритм арбитража по приоритету сообщения на основе множественного доступа с обнаружением несущей и детектированием коллизий. Этот метод обслуживает и разрешает одновременные запросы доступа к шине.
<i>cyclic redundancy check</i> <b>(CRC)</b> <b>циклический избыточный код (CRC)</b> <i>[цээрцэ]</i>	Проверка циклического избыточного кода выполняется на основе полинома, рассчитываемого как в передающем, так и в принимающем узлах.

## D

<i>data consistency</i> <b>состоятельность данных</b>	Механизм глобализации локальной ошибки гарантирует состоятельность данных для всех CAN узлов сети, находящихся в активном к ошибке состоянии.
<i>data field</i> <b>поле данных</b>	Поле данных CAN кадра содержит от 0 до 8 байт информации. Его длина указывается кодом DLC.
<i>data frame</i> <b>кадр данных</b>	CAN кадр данных передает информацию от поставщика одному или нескольким потребителям. Он состоит из бита начала кадра, поля арбитража, контрольного поля, поля данных, поля CRC, поля подтверждения и поля конца кадра.
<i>data length code (DLC)</i> <b>длина поля данных (DLC)</b> <i>[диэлси]</i>	Длина поля данных CAN кадра указывается 4-х битовым кодом в контрольном поле. В кадрах удаленного запроса DLC должна совпадать с длиной поля данных запрашиваемого кадра!
<i>data type</i> <b>тип данных</b>	Атрибут CANopen объекта, определяющий формат. Например: unsigned8 (беззнаковое 8 бит), integer16 (целое со знаком 16 бит), boolean (булево/логическое).

<i>data link layer</i> <b>канальный уровень</b>	Второй уровень эталонной модели ISO/OSI, обеспечивающий базовые коммуникационные сервисы. Канальный уровень CAN определяет кадры данных, удаленного запроса, ошибки и перегрузки.
<i>DBT дистрибутор (DBT) [дибити]</i>	Дистрибутор является составной частью спецификации прикладного уровня CAN (CAL) и определяет метод автоматического распределения идентификаторов при загрузке сети.
<i>default value</i> <b>значение по умолчанию</b>	Атрибут CANopen объекта, определяющий предустановленные значения для не конфигурируемых пользователем объектов после включения питания либо перезапуска приложения.
<i>DeviceNet</i> <b>DeviceNet [девайснет]</b>	Протокол высокого уровня и определение профилей устройств на основе CAN. DeviceNet был разработан для автоматизации промышленных предприятий и детально описывает физический уровень CAN для обеспечения высокой совместимости и взаимозаменяемости готовых устройств. Спецификации DeviceNet поддерживаются не коммерческой организацией ODVA ( <a href="http://www.odva.org">www.odva.org</a> ).
<i>device profile</i> <b>профиль устройства</b>	Профиль устройства определяет специфичные для этого устройства коммуникационные сервисы, в том числе детально описывает процедуры конфигурирования.
<i>Diagnostics on CAN</i> <b>CAN диагностика</b>	Стандарт ISO 15765 определяет протоколы и сервисы CAN диагностики, которые применяются в диагностическом CAN интерфейсе пассажирских автомобилей.
<i>dominant bit</i> <b>доминантный бит</b>	Доминантное состояние бита на CAN шине. Логическое значение доминантного бита - '0'.
<i>dominant state</i> <b>доминантное состояние</b>	Доминантное состояние по определению вытесняет (перезаписывает) рецессивное.
<i>double-reception of message</i> <b>дублированный прием сообщения</b>	Искажение последнего бита поля конца кадра (EOF) в передающем узле вызовет повторную передачу кадра. Так как после предпоследнего бита поля конца кадра получатели уже приняли сообщение, они получают его дважды.
<i>DR (draft recommendation)</i> <b>проектные рекомендации</b>	Проектные рекомендации еще не определены окончательно, но уже опубликованы. Проектные рекомендации CiA неизменны в течение одного года.

<i>DS (draft standard)</i> <b>проектный стандарт</b>	Проектный стандарт еще не определен окончательно, но уже опубликован. Проектные стандарты CiA неизменны в течение одного года.
<i>DSP (draft standard proposal)</i> <b>план проектного стандарта</b>	План проектного стандарта является только опубликованным предложением. Планы проектного стандарта CiA могут изменяться в любое время без каких-либо извещений.
<i>D-sub connector</i> <b>разъем D-sub</b>	Стандартные разъемы. Наиболее широко распространен девяти штырьковый разъем D-sub (DIN 41652). Назначение его выводов для CAN сетей описано в CiA DS 102.

## Е

<i>EDS checker</i> <b>средство проверки EDS [идээс]</b>	Программный инструментарий, позволяющий тестировать согласованность электронных спецификаций. Средство проверки EDS для CANopen может быть загружено с сайта CiA.
<i>EDS generator</i> <b>EDS генератор</b>	Программный инструментарий, позволяющий создавать электронные спецификации. Доступен для CANopen и DeviceNet.
<i>electronic data sheet (EDS)</i> <b>электронная спецификация (EDS) [идээс]</b>	Электронная спецификация описывает функциональность устройства стандартным образом. CANopen и DeviceNet используют различные форматы EDS.
<i>Emergency</i> <b>срочное сообщение</b>	Предопределенный коммуникационный сервис CANopen, поддерживаемый одним 8-ми байтовым кадром данных. Кадр состоит из двух байт стандартного кода ошибки, одного байта регистра ошибки и пяти байт информации производителя устройства. Срочное сообщение используется для информирования об аппаратных и программных отказах устройства.
<i>EN 50325</i> <b>EN 50325 [иэн]</b>	Набор стандартов CENELEC [синелек], определяющий DeviceNet (часть 2), интеллектуальные распределенные системы (часть 3) и CANopen (часть 4).
<i>entry category</i> <b>категория записи</b>	Атрибут объекта CANopen, определяющий, является ли этот объект обязательным или опциональным.
<i>end of frame (EOF)</i> <b>конец кадра (EOF) [иоф]</b>	EOF состоит из семи рецессивных бит в CAN кадрах данных и удаленного запроса.
<i>error active state</i> <b>активное к ошибке состояние</b>	В активном к ошибке состоянии CAN контроллер может передавать активные кадры ошибки, содержащие активный флаг ошибки. Если все CAN узлы активны к ошибке, то гарантируется состоятельность данных во всей сети.

<i>error code</i> <b>код ошибки</b>	В CANopen определены стандартные коды ошибок, передаваемые в составе срочного сообщения.
<i>error control message</i> <b>сообщение контроля ошибки</b>	Сообщение контроля ошибки CANopen представляет собой однобайтовый кадр данных, которому назначен фиксированный идентификатор, зависящий от номера CAN узла. Этот кадр передается в качестве сообщения загрузки при переходе узла сети в предоперационное состояние после инициализации, при удаленном запросе от NMT мастер (протокол охраны узла) или периодически самим устройством (протокол сердцебиения).
<i>error counter</i> <b>счетчик ошибок</b>	В каждом CAN контроллере имеется два счетчика ошибок: один для принимаемых сообщений и один для передаваемых. Значения счетчиков увеличиваются и уменьшаются прозрачно для пользователя, обеспечивая выполнение правил, изложенных в ISO 11898-1. Счетчики ошибок используются для определения текущего состояния CAN модуля (активное к ошибке, пассивное к ошибке и bus-off).
<i>error delimiter</i> <b>разделитель ошибки</b>	Конечный сегмент кадра ошибки, состоящий из 8-ми рецессивных бит.
<i>error detection capability</i> <b>способность обнаружения ошибок</b>	В CAN протоколе используются пять различных механизмов обнаружения ошибок, которые дают возможность зафиксировать практически любое искажение CAN сообщения. Вероятность не обнаружения ошибки зависит от интенсивности ошибок, скорости передачи, загрузки сети, числа узлов и коэффициента способности обнаружения ошибок.
<i>error flag</i> <b>флаг ошибки</b>	Начальный сегмент кадра ошибки, состоящий из шести бит одного знака. Один флаг ошибки может перекрываться другим, передаваемым еще одним узлом.
<i>error frame</i> <b>кадр ошибки</b>	Передается при обнаружении ошибки. Состоит из флага ошибки и разделителя ошибки.
<i>error globalization</i> <b>глобализация ошибки</b>	Передача кадра ошибки, вызванная локальным отказом, будет воспринята как ошибка бит стаффинга, что вызовет передачу флага ошибки другими узлами. Таким образом, происходит глобализация локальной ошибки и гарантируется состоятельность данных для всех активных к ошибке узлов сети.

<i>error passive state</i> <b>пассивное к ошибке состояние</b>	В пассивном к ошибке состоянии CAN контроллер может передавать только пассивные кадры ошибки, содержащие пассивный флаг ошибки.
<i>error signaling</i> <b>сигнализация ошибки</b>	Сигнализация ошибки обеспечивается путем передачи кадров ошибки.
<i>event-driven</i> <b>управляемый событием</b>	Управляемые событием сообщения передаются, когда в устройстве происходят определенные перемены. Это может быть изменение состояния входных параметров, истечение промежутка времени или любое другое локальное событие.
<i>event-driven PDO</i> <b>управляемый событием PDO</b>	Управляемое событием PDO передается, когда в устройстве происходит определенное внутреннее событие. Таковым может быть истечение таймера события PDO. При получении управляемого событием PDO программное обеспечение немедленно обновляет заданные в PDO отображении записи объектного словаря.
<i>event timer</i> <b>таймер события</b>	Таймер события может быть назначен в CANopen каждому TPDO. Он задает частоту передачи этого PDO объекта.
<i>expedited SDO</i> <b>ускоренный SDO [эсдэо]</b>	Ускоренный SDO является коммуникационным сервисом с подтверждением в CANopen (равный к равному). Включает в себя одно иницирующее SDO сообщение со стороны узла клиента и соответствующее подтверждающее сообщение узла сервера. Ускоренные SDO могут использоваться для передачи до четырех байт данных.
<i>Explicit Message</i> <b>явное сообщение</b>	Явное сообщение является коммуникационным сервисом с подтверждением в DeviceNet, который используется в конфигурационных целях. Для обмена информацией объемом свыше 8 байт, явное сообщение поддерживает сегментированную передачу.
<i>extended frame format</i> <b>расширенный формат кадра</b>	В кадрах данных и удаленного запроса расширенного формата используются 29 битовые идентификаторы.

## **F**

<i>fault confinement</i> <b>локализация отказа</b>	CAN узлы в состоянии отличить кратковременные нарушения в своей работе от постоянного отказа. Неисправные передающие узлы отключаются, что означает их логическое отсоединение от сети.
---	---

<i>fault-tolerant transceiver</i> <b>отказоустойчивый трансивер</b>	В соответствии со спецификациями ISO 11898-3 и ISO 11992-1 трансиверы могут работать по одной линии шины совместно с CAN землей, в случае если в другой линии произошел обрыв или короткое замыкание или плохо подсоединены терминальные резисторы.
<i>flying master</i> <b>переключаемый мастер</b>	В критических по безопасности приложениях может возникать потребность автоматического замещения отказавшего NMT мастер другим резервным NMT мастером. Такой подход к избыточности называется переключаемым мастером.
<i>form error</i> <b>ошибка формата</b>	Искажение одного из predetermined рессивных бит (CRC разделителя, разделителя подтверждения, EOF) рассматривается как ошибка формата, которая вызывает передачу кадра ошибки в очередном интервале битового времени.
<i>Frame</i> <b>кадр</b>	Сущность протокола канального уровня, определяющая порядок и значение бит или битовых полей в передаваемой последовательности.
<i>frame coding</i> <b>кодировка кадра</b>	Последовательность полей в CAN кадрах, например, для кадра данных: SOF, поле арбитража, контрольное поле, поле данных, поле CRC, поле подтверждения и EOF. Кодировка кадра также включает в себя бит стаффинг.
<i>frame format</i> <b>формат кадра</b>	Стандарт CAN различает основной формат кадра, использующий 11 битовые идентификаторы, и расширенный формат с 29 битовыми идентификаторами.
<i>frame types</i> <b>типы кадра</b>	В CAN используются четыре типа кадров: кадр данных, кадр удаленного запроса, кадр ошибки и кадр перегрузки.
<i>FullCAN</i> <b>FullCAN [фулкан]</b>	Реализация CAN, в которой используются единые буферы принимаемых и передаваемых сообщений, имеющих различные идентификаторы.
<i>function code</i> <b>код функции</b>	Первые четыре бита CAN идентификатора в predetermined распределении идентификаторов CANopen, указывающие функциональное назначение коммуникационного объекта (например, TPDO_1 или сообщение контроля ошибок).

## G

<i>galvanic isolation</i> <b>гальваническая изоляция</b>	Для гальванической изоляции в CAN сетях используются оптрона или трансформаторы, установленные между CAN контроллером и трансивером.
---	--

<i>Gateway</i> <b>шлюз</b>	Шлюзом называется устройство, обладающее минимум двумя сетевыми интерфейсами и преобразующее все семь уровней протоколов эталонной модели ISO/OSI, например, шлюз между CANopen и Ethernet.
<i>global error</i> <b>глобальная ошибка</b>	Глобальная сетевая ошибка оказывает влияние на все подсоединенные CAN устройства.
<i>global fail-safe command</i> <b>глобальная команда безопасного отказа</b>	Глобальная команда безопасного отказа (GFC) является высоко приоритетным CAN сообщением, определенным в безопасном CANopen протоколе. Ее использование сокращает время реакции на отказ. Вслед за GFC командой должно быть передано соответствующее SRDO.

## Н

<i>Hamming distance</i> <b>расстояние Хэмминга</b>	Расстояние Хэмминга равно 6 (теоретическое значение для CAN сетей), позволяет обнаружить до пяти случайно распределенных искажений отдельных бит.
<i>hard synchronization</i> <b>аппаратная синхронизация</b>	Все CAN узлы аппаратно синхронизируются фронтом бита начала кадра при его обнаружении на шине. Аппаратная синхронизация производится когда шина свободна, при отложенной передаче и в третьем бите межкадрового промежутка.
<i>Heartbeat</i> <b>сердцебиение</b>	Для информирования о работоспособности узла CANopen и DeviceNet используют сообщение сердцебиения. Его передача осуществляется периодически.
<i>heartbeat consumer time</i> <b>период сердцебиения потребителя</b>	Период сердцебиения потребителя определяет время, по истечении которого считается, что узел, который не передал сообщение сердцебиения, потерял работоспособность.
<i>heartbeat producer time</i> <b>период сердцебиения поставщика</b>	Период сердцебиения поставщика определяет частоту передачи сообщения сердцебиения.
<i>higher-layer protocol (HLP)</i> <b>протокол высокого уровня</b>	Протоколы высокого уровня соответствуют коммуникационным протоколам транспортного, сеансового, уровня представления данных или прикладного уровня эталонной модели ISO/OSI.
<i>high-speed transceiver</i> <b>высокоскоростной трансивер</b>	Трансиверы, поддерживающие скорость передачи до 1 Мбит/с в соответствии с ISO 11898-2.

# I

<i>Identifier</i> <b>идентификатор</b>	CAN идентификатор определяет содержимое кадра данных. Идентификатор кадра удаленного запроса соответствует идентификатору запрашиваемого кадра. Идентификатор неявно задает приоритет арбитража сообщения в CAN сети.
<i>identifier extension flag (IDE)</i> <b>флаг расширенного формата</b>	Флаг расширенного формата (бит IDE) указывает, каким образом интерпретируются последующие биты: как контрольные или же задающие вторую часть 29 битового идентификатора.
<i>identifier field</i> <b>поле идентификатора</b>	Поле идентификатора содержит 11 бит для кадра основного формата и дополнительно 18 бит для расширенного формата кадра.
<i>Index</i> <b>индекс</b>	Индексом является 16 битовый адрес, используемый для доступа к объектному словари CANopen. Для массивов и записей этот адрес расширяется 8 битовым субиндексом.
<i>inhibit timer</i> <b>таймер подавления</b>	CANopen объект для PDO и срочного сообщения, который осуществляет запрет передачи в течение определенного времени (время подавления) соответствующего коммуникационного объекта.
<i>initialization state</i> <b>состояние инициализации</b>	Состояние NMT slave устройства CANopen, в которое оно переходит автоматически после включения питания и при сбросе коммуникаций либо приложения.
<i>inner priority inversion</i> <b>внутренняя инверсия приоритета</b>	К внутренней инверсии приоритета приводит ситуация, когда низко приоритетное сообщение не может быть передано в CAN сеть вследствие значительного трафика высокоприоритетных сообщений. При этом, если в устройстве возникает запрос на передачу высокоприоритетного сообщения, он не может быть удовлетворен CAN контроллером, поскольку низко приоритетное сообщение по-прежнему ожидает своей очереди на отправку.
<i>interface profile</i> <b>профиль интерфейса</b>	CANopen профиль, который определяет только интерфейс, но не прикладное поведение устройства, например профиль шлюзов и мостов.
<i>interframe space</i> <b>межкадровый промежуток</b>	Межкадровый промежуток формируется тремя рецессивными битами. Он разделяет все CAN кадры, в том числе кадры ошибки и перегрузки.

<i>intermission field</i> <b>поле паузы</b>	Синоним межкадрового промежутка (interframe space).
<i>I/O Message</i> <b>Сообщение ввода-вывода (I/O [uco])</b>	Коммуникационный объект DeviceNet (из объектов транспортного приложения), представляющий ввод или вывод. I/O сообщения отображаются в один или несколько CAN кадров данных с поддержкой сегментированной передачи.
<i>ISO 11898-1</i> <b>ISO 11898-1 [uco]</b>	Международный стандарт, определяющий канальный уровень CAN, в том числе подуровни LLS (службы настройки уровня), MAC (контроль доступа к коммуникационной среде) и PLS (физические сигналы).
<i>ISO 11898-2</i> <b>ISO 11898-2 [uco]</b>	Международный стандарт, определяющий высокоскоростное CAN MAU (устройство подсоединения к коммуникационной среде).
<i>ISO 11898-3</i> <b>ISO 11898-3 [uco]</b>	Международный стандарт, определяющий низкоскоростное, отказоустойчивое CAN MAU (устройство подсоединения к коммуникационной среде).
<i>ISO 11898-4</i> <b>ISO 11898-4 [uco]</b>	Международный стандарт, определяющий синхронной коммуникационный протокол на основе CAN.
<i>ISO 11992</i> <b>ISO 11992 [uco]</b>	Международный стандарт, определяющий прикладной CAN профиль для грузовиков и трейлеров. Вторая часть этого стандарта содержит спецификации для тормозных устройств и коробок передач, часть 3 описывает профили остальных устройств, а четвертая часть - диагностические системы.
<i>ISO 11745-2</i> <b>ISO 11645-2 [uco]</b>	Международный стандарт, определяющий структуру интеграции приложений для систем управления на основе ISO 11898, таких, как CANopen и DeviceNet.
<i>ISO 11783</i> <b>ISO 11783 [uco]</b>	Международный стандарт, определяющий прикладной CAN профиль, который используется в сельскохозяйственных и лесоводческих машинах. Основан на профиле J1939.
<i>ISO 16844</i> <b>ISO 16844 [uco]</b>	Международный стандарт, определяющий тахограф на основе CAN, который используется в грузовиках и автобусах.
<i>ISO 16845</i> <b>ISO 16845 [uco]</b>	Международный стандарт, определяющий план теста соответствия для реализаций ISO 11898-1.

## J

<i>J1939 application profile</i> <b>прикладной профиль</b> <b>J1939 [джэй 1939]</b>	Прикладной профиль, разработанный Сообществом Инженеров Автомобилестроения (SAE), задает спецификацию внутренних информационных коммуникаций для грузовиков и автобусов. Профиль J1939 определяет коммуникационные сервисы и сигналы, в том числе отображение данных в CAN кадры посредством PGN (номеров групп параметров).
<i>J2284 bit timing</i> <b>битовое время [джэй 2284]</b>	Определение битового времени для внутренних сетей легковых автомобилей при скоростях шины 250 кбит/с и 500 кбит/с. Предложено SAE.
<i>J2411 single-wire CAN</i> <b>однопроводной CAN</b> <b>J2411 [джэй 2411]</b>	Спецификация однопроводной передачи для CAN сетей, разработанная SAE. Битовая скорость ограничена 40 кбит/с.

## L

<i>layer-2 protocol</i> <b>протокол уровня 2</b>	Протокол уровня 2 использует коммуникационные сервисы CAN непосредственно, не опираясь на определенный высокоуровневый протокол.
<i>layer-7 protocol</i> <b>протокол уровня 7</b>	Протокол уровня 7 использует коммуникационные сервисы CAN стандартизированным образом. Это предоставляет возможность повторного использования прикладного программного обеспечения без переработки коммуникационного CAN ПО.
<i>layer setting services (LSS)</i> <b>сервисы установки уровня (LSS) [элэсэс]</b>	CANopen сервисы установки уровня (LSS) определяют коммуникационные сервисы для конфигурирования номера узла и битовой скорости с использованием CAN сети.
<i>life guarding</i> <b>охрана работоспособности</b>	Метод, позволяющий обнаружить, что NMT мастер больше не наблюдает работоспособности NMT slave устройства. Используется на прикладном уровне CAN и в CANopen.
<i>line topology</i> <b>линейная топология</b>	Топология сети, где все узлы непосредственно подсоединяются к единой шине. Теоретически CAN сети используют только линейную топологию без каких-либо ответвлений. Тем не менее, на практике встречаются также древовидные и звездообразные топологии.

<i>LMT protocols</i> <b>LMT протокол [элэмти]</b>	Протоколы, определенные в прикладном уровне CAN (CAL) для задания номера узла и битовой скорости с использованием CAN сети.
<i>local bus error</i> <b>локальная ошибка шины</b>	Локальная ошибка шины оказывает влияние на один либо несколько, но не все узлы сети.
<i>logical link control (LLC)</i> <b>контроль логического соединения (LLC) [элэлси]</b>	Подуровень контроля логического соединения (LLC) описывает верхние компоненты канального уровня ISO/OSI. Он касается тех аспектов протокола, которые не зависят от метода доступа к коммуникационной среде.
<i>low-power mode</i> <b>режим пониженного энергопотребления</b>	CAN контроллер и CAN трансивер могут поддерживать режим пониженного энергопотребления (режим stand-by), когда мощность питания меньше, чем в рабочем режиме.
<i>low-speed transceiver</i> <b>низкоскоростной трансивер</b>	Синоним отказоустойчивого трансивера.
<i>LSS protocols</i> <b>протоколы LSS [элэсэс]</b>	Сервисы установки уровня (LSS) определяют протоколы для задания номера узла и битовой скорости с использованием CAN сети.

## **М**

<i>Master</i> <b>мастер</b>	Коммуникационная или прикладная сущность, которой разрешено контролировать определенные операции. В сетевых приложениях такой операцией может быть, например, инициализация коммуникационных сервисов.
<i>master/slave communication</i> <b>мастер/slave коммуникации [слэйв]</b>	В мастер/slave коммуникациях инициализацию и контроль процесса обмена осуществляет мастер. Slave-у не разрешено выступать инициатором никаких коммуникационных операций.
<i>matrix cycle</i> <b>матричный цикл</b>	В TTCAN матричный цикл состоит из одного или нескольких основных циклов. Каждый основной цикл начинается с реперной посылки, но за ней могут следовать различные окна.
<i>medium access control (MAC)</i> <b>контроль доступа к коммуникационной среде (MAC) [мак]</b>	Подуровень контроля доступа к коммуникационной среде (MAC) представляет нижние компоненты канального уровня ISO/OSI. Он обслуживает интерфейсы к LLC подуровню и физическому уровню. MAC включает в себя правила и функции, относящиеся к инкапсуляции/декапсуляции, обнаружению ошибок и выдаче сигналов.

<i>medium dependent interface (MDI)</i> <b>интерфейс, зависящий от среды (MDI)</b> [эмдиай]	MDI определяет требования к разъемам, кабелям и терминаторам.
<i>Message</i> <b>сообщение</b>	CAN сообщением является кадр данных или кадр удаленного запроса.
<i>message buffer</i> <b>буфер сообщений</b>	Микросхемы CAN контроллеров поддерживают буферы сообщений для принимаемых и/или передаваемых кадров. Реализация и алгоритмы работы с буферами сообщений не стандартизированы.
<i>multicast transmission</i> <b>мультивещательная передача</b>	Метод адресации, когда один кадр направляется нескольким узлам одновременно.
<i>multi-master communication</i> <b>мульти-мастерные коммуникации</b>	В мульти-мастерных коммуникационных системах каждый из узлов может временно контролировать обмен данными в сети. Это означает, что каждый узел теоретически обладает правом доступа к шине в любое время, при условии что она свободна.
<i>multiplexed PDO (MPDO)</i> <b>мультиплексированный PDO (MPDO)</b> [эмпэдэо]	Мультиплексированный PDO состоит из 8 байт, в том числе одного контрольного байта, трех байт мультиплексора (содержит 16 битовый индекс и 8 битовый субиндекс) и четырех байт собственно данных объекта.

## N

<i>network length</i> <b>длина сети</b>	См. bus length (длина шины).
<i>network management</i> <b>менеджер сети</b>	Менеджер сети несет ответственность за процедуру загрузки и опциональное конфигурирование узлов. Менеджер может также выполнять функции контроля состояния узлов сети, например, охраны узла.
<i>network variables</i> <b>сетевые переменные</b>	Сетевые переменные используются в программируемых CANopen устройствах для отображения в PDO по завершении программирования устройства.
<i>NMT</i> <b>NMT</b> [энэмтэ]	Менеджер сети (сетевой менеджер) в прикладном уровне CAN и CANopen.
<i>NMT master</i> <b>NMT мастер</b> [энэмтэ]	NMT мастер осуществляет менеджмент сети путем передачи NMT сообщений. С их помощью мастер управляет NMT машинами состояний всех подключенных NMT slave устройств.

<i>NMT slave</i> <b><i>NMT slave [энэмтэ слэйв]</i></b>	NMT slave принимает NMT сообщения, содержащие команды управления для NMT машины состояний, которая реализована на прикладном уровне CAN и CANopen устройств.
<i>NMT slave state machine</i> <b><i>машина состояний NMT slave [энэмтэ]</i></b>	Машина (диаграмма) состояний NMT slave устройства определена для прикладного уровня CAN и CANopen. NMT мастер управляет переходом устройства из одного состояния в другое путем передачи CAN сообщения с наивысшим приоритетом.
<i>Node</i> <b><i>узел</i></b>	Модуль, подсоединенный к CAN сети и способный осуществлять коммуникации в соответствии с CAN протоколами.
<i>node guarding</i> <b><i>охрана узла</i></b>	Механизм, используемый на прикладном уровне CAN и CANopen для обнаружения отсоединенных устройств, либо находящихся в состоянии bus-off. NMT мастер посылает кадр удаленного запроса для NMT slave, на который тот отвечает соответствующим сообщением контроля ошибки.
<i>node-ID</i> <b><i>идентификатор узла</i></b>	Уникальный идентификатор устройства, используемый различными высокоуровневыми CAN протоколами, например, CANopen или DeviceNet, при назначении этому устройству CAN идентификаторов. Для предопределенного распределения идентификаторов CANopen либо DeviceNet идентификатор узла является частью CAN идентификатора.
<i>nominal bit-rate</i> <b><i>номинальная битовая скорость</i></b>	Номинальная битовая скорость определяется числом бит в секунду, которое передается идеальным передатчиком в отсутствие ресинхронизации.
<i>nominal bit-time</i> <b><i>номинальное битовое время</i></b>	Номинальное битовое время может быть представлено как разделенное на отдельные, не перекрывающиеся временные сегменты.
<i>non-return to zero (NRZ)</i> <b><i>без возврата к нулю (NRZ) [энэрзэт]</i></b>	Метод представления двоичного сигнала, при котором его уровень неизменен в течение одного битового времени. При кодировании без возврата к нулю необходима возможность ресинхронизации, поскольку битовый поток может иметь один и тот же логический знак, а значит в сигнале не будет каких-либо перепадов уровня.

# О

<i>object dictionary</i> <b>объектный словарь</b>	Объектный словарь является сердцевиной каждого CANopen устройства и содержит описания всех коммуникационных и прикладных объектов.
<i>operational state</i> <b>операционное состояние</b>	В операционном NMT состоянии доступны все коммуникационные сервисы CANopen.
<i>OSEK/VDX</i> <b>OSEK/VDX</b> <b>[осек/видиэкс]</b>	Набор спецификаций, определяющих коммуникации (COM), менеджмент сети (NM), операционную систему реального времени (OS), и язык реализации (OIL). OSEK/VDX частично реализован в легковых автомобилях.
<i>OSI (open system inter-connection) reference model</i> <b>эталонная модель ISO/OSI [исо/оси]</b>	Семиуровневая эталонная модель протоколов передачи данных. Определяет уровни: физический, канальный, сетевой, транспортный, сеансовый, представительский и прикладной. В CAN сетях обычно реализуются только физический, канальный и прикладной уровни.
<i>outer priority inversion</i> <b>внешняя инверсия приоритета</b>	Если узлу необходимо передать два высоко приоритетных CAN сообщения, но он не может отослать второе из них сразу вслед за межкадровым промежутком, другой узел сети может успеть передать низко приоритетное сообщение. Такая ситуация называется внешней инверсией приоритета.
<i>overload condition</i> <b>состояние перегрузки</b>	Ситуации, при которых CAN контроллер передает кадр перегрузки, например, доминантный уровень среди первых двух бит межкадрового промежутка, доминантное значение последнего бита EOF, неверное значение бит в кадрах ошибки или перегрузки.
<i>overload delimiter</i> <b>разделитель перегрузки</b>	Конечный сегмент кадра перегрузки, состоящий из 8-ми рецессивных бит.
<i>overload flag</i> <b>флаг перегрузки</b>	Начальный сегмент кадра перегрузки, состоящий из шести доминантных бит. Один флаг перегрузки может перекрываться другим, передаваемым еще одним узлом.
<i>overload frame</i> <b>кадр перегрузки</b>	Кадр для обозначения состояния перегрузки.

# Р

<i>parameter group number</i> <b>(PGN)</b> <b>номер группы параметров (PGN) [пиджиэн]</b>	<p>В стандартах SAE J1939, ISO 11783 и ISO 11992 определены номера групп параметров (PGN-ы), которые определяют содержимое соответствующего CAN сообщения. PGN отображается в 29 битовый идентификатор.</p>
<i>passive error flag</i> <b>пассивный флаг ошибки</b>	<p>Пассивный флаг ошибки является начальной частью пассивного кадра ошибки и состоит из шести последовательных рецессивных бит.</p>
<i>PDO mapping</i> <b>PDO отображение</b> <b>[пэдэо]</b>	<p>В каждый PDO может быть отображено до 64 объектов. PDO отображение описывается параметрами PDO отображения.</p>
<i>pending transmission request</i> <b>ожидающий запрос на передачу</b>	<p>Вследствие занятости шины или по причине проигрыша узлом процедуры арбитража в CAN контроллере может храниться одно и более сообщений, которые ожидают передачи в сеть.</p>
<i>phase error</i> <b>фазовая ошибка</b>	<p>Фазовая ошибка фронта сигнала задается его положением относительно сегмента синхронизации (Sync_Seg). Измеряется в квантах времени (time quanta).</p>
<i>phase segment 1</i> <b>(Phase_Seg 1)</b> <b>фазовый сегмент 1</b>	<p>Часть битового времени, используемая для компенсации фазовой ошибки фронта сигнала. Может быть удлинен при ресинхронизации.</p>
<i>phase segment 2</i> <b>(Phase_Seg 2)</b> <b>фазовый сегмент 2</b>	<p>Часть битового времени, используемая для компенсации фазовой ошибки фронта сигнала. Может быть укорочен при ресинхронизации.</p>
<i>physical layer</i> <b>физический уровень</b>	<p>Низший уровень эталонной модели ISO/OSI, определяющий разъемы, типы кабеля, электрические или оптические сигналы, задающие уровень бит, а также синхронизацию и ресинхронизацию.</p>
<i>physical signaling (PLS)</i> <b>физические сигналы</b> <b>(PLS) [пиэлэс]</b>	<p><i>Подуровень физического уровня. Принимает и передает в схему трансивера битовый поток. Выполняет кодирование/декодирование бит, контролирует битовое хронирование и синхронизацию.</i></p>
<i>pin assignment</i> <b>назначение выводов</b>	<p>Определение назначения выводов разъемов.</p>
<i>pre-defined connection set</i> <b>предопределенное распределение идентификаторов</b>	<p>Набор CAN идентификаторов, используемых по умолчанию для различных коммуникационных протоколов CANopen либо DeviceNet.</p>

<i>Pre-operational state</i> <b>предоперационное состояние</b>	В предоперационном NMT состоянии коммуникационные объекты PDO запрещены.
<i>Priority</i> <b>приоритет</b>	Атрибут кадра, определяющий его арбитражный ранг. В CAN кадрах данных и удаленного запроса приоритет задается идентификатором. Чем меньше его значение, тем выше приоритет кадра.
<i>Process Data Object (PDO)</i> <b>объект данных процесса (PDO) [пэдэо]</b>	Коммуникационный объект, определяемый коммуникационным PDO параметром и параметром PDO отображения. Относится к не подтверждаемым коммуникационным сервисам и не приводит к избыточности протокола.
<i>Producer</i> <b>поставщик</b>	В CAN сетях поставщиком называется устройство, которое осуществляет передачу сообщений.
<i>propagation segment (Prop_Seg)</i> <b>сегмент распространения</b>	Часть битового времени, используемая для компенсации физических задержек сигнала в сети. Полное время задержки включает в себя время распространения сигнала по шине и время внутренней задержки сигнала в узлах сети.
<i>Protocol</i> <b>протокол</b>	Формальный набор соглашений и правил, описывающий обмен информацией между узлами сети, в том числе спецификации управления кадром, передачи кадра и физического уровня.
<i>priority inversion</i> <b>инверсия приоритета</b>	Инверсия приоритетов происходит, когда объект с низким приоритетом обрабатывается или передается прежде объекта высокого приоритета. В недостаточно хорошо спроектированных CAN устройствах может наблюдаться внутренняя либо внешняя инверсия приоритета.

## R

<i>receive error counter (REC)</i> <b>счетчик ошибок приема (REC) [рек]</b>	Внутренний счетчик ошибок приема CAN контроллера. В некоторых контроллерах значение этого счетчика доступно для чтения.
<i>Receiver</i> <b>приемный узел</b>	CAN узел называется приемным узлом или потребителем, если он не является передающим узлом и шина не свободна.
<i>reception buffer(s)</i> <b>приемные буферы</b>	Локальная память CAN контроллера, используемая для промежуточного хранения принятых сообщений.
<i>recessive bit</i> <b>рецессивный бит</b>	Рецессивное состояние бита на CAN шине. Логическое значение рецессивного бита - '1'.

<i>recessive state</i> <b>рецессивное состояние</b>	По определению, рецессивное состояние вытесняется (перезаписывается) доминантным.
<i>recovery time</i> <b>время восстановления</b>	Промежуток времени между первым битом флага ошибки и моментом, когда может быть начата автоматическая повторная передача. В активных к ошибке узлах максимальное время восстановления составляет 23 интервала битового времени, в пассивных к ошибке узлах – 31 интервал битового времени.
<i>redundant networks</i> <b>избыточные сети</b>	В ряде критических по безопасности приложений (например, морские системы) могут использоваться избыточные сети, которые обеспечивают возможность переключения с одной сети на другую при обнаружении коммуникационных ошибок.
<i>Reference message</i> опорное сообщение	В системе TTCAN каждый основной цикл начинается с опорного сообщения.
<i>remote frame</i> <b>кадр удаленного запроса</b>	С помощью кадра удаленного запроса узел сети запрашивает передачу другим узлом соответствующего кадра данных с тем же самым значением идентификатора. Код длины поля данных (DLC) удаленного запроса должен иметь то же значение, что и DLC соответствующего кадра данных. Длина поля данных кадра удаленного запроса – ноль байт.
<i>remote transmit request (RTR)</i> <b>удаленный запрос {на передачу} (RTR) [эртээр]</b>	Бит поля арбитража, указывающий, является ли данный кадр кадром удаленного запроса (рецессивный уровень) или кадром данных (доминантный уровень).
<i>Repeater</i> <b>репитер</b>	Пассивное устройство, повторяющее сигналы на CAN шине. Репитер используется для увеличения максимально возможного числа подсоединяемых узлов или для увеличения протяженности сетей (свыше 1 км) или для реализации древовидной либо сложной топологий.
<i>Reset</i> <b>сброс</b>	Сброс CAN контроллера осуществляется с помощью команды (возможно, подаваемой аппаратно). До перехода в активное к ошибке состояние CAN контроллер должен зарегистрировать 128 одиннадцати битовых рецессивных сегментов.
<i>reset application</i> <b>сброс приложения</b>	NMT команда сброса приложения устанавливает значения по умолчанию или сохраняемые конфигурируемые значения для всех объектов CANopen устройства.

<i>reset communication</i> <b>сброс коммуникаций</b>	NMT команда сброса коммуникаций устанавливает значения по умолчанию или сохраняемые конфигурируемые значения только для коммуникационных объектов CANopen устройства.
<i>re-synchronization jump width (SJW)</i> <b>скачок ресинхронизации</b>	Число квантов времени, на которое может быть удлинен фазовый сегмент 1 (Phase_Seg 1) или укорочен фазовый сегмент 2 (Phase_Seg 2).
<i>RPDO</i> <b>RPDO [эриэдэо]</b>	RPDO (принимаемый объект данных процесса) является коммуникационным объектом, который принимается CANopen устройством.

## S

<i>sample point</i> <b>точка выборки</b>	Точкой выборки является момент времени, когда считывается сигнал на шине и определяется значение соответствующего бита. Она располагается между фазовым сегментом 1 и фазовым сегментом 2.
<i>SCT</i> <b>SCT [эссити]</b>	Длительность цикла безопасности (Safe-guard Cycle Time) определяет максимальный интервал времени между двумя периодически передаваемыми SRDO (безопасный объект данных).
<i>SDO block transfer</i> <b>блочная передача SDO [эсдэо]</b>	Блочная передача SDO - это коммуникационный сервис CANopen, предназначенный для увеличения производительности SDO обмена. При блочной передаче подтверждение высылается лишь после приема определенного числа SDO сегментов.
<i>SDO manager</i> <b>SDO менеджер [эсдэо]</b>	SDO менеджер управляет динамической установкой SDO соединений. Он размещается на том же узле, что и NMT мастер.
<i>segmented SDO</i> <b>сегментированный SDO [эсдэо]</b>	Сегментированный SDO используется для передачи объектов длиной свыше 4 байт. Полное число сегментов в цикле передачи теоретически не ограничено.
<i>Server SDO</i> <b>SDO сервера [эсдэо]</b>	SDO сервера принимает SDO сообщения от соответствующего SDO клиента и возвращает ответ на каждое сообщение либо на блок сообщений (блочная передача SDO).
<i>Service Data Object (SDO)</i> <b>сервисный объект данных (SDO) [эсдэо]</b>	Сервисные объекты данных обеспечивают доступ к записям объектного словаря CANopen. Каждый SDO формируется как минимум двумя CAN сообщениями с различными идентификаторами. SDO – это подтверждаемый коммуникационный сервис типа точка-точка.

<i>single-shot transmission</i> <b>однократная передача</b>	Ряд CAN контроллеров поддерживает режим однократной передачи, когда в случае обнаружения ошибки не производится автоматическая повторная посылка сообщения. Такой режим необходим для протокола TTCAN.
<i>single-wire CAN (SWC)</i> <b>однопроводной CAN</b>	Физический уровень, где используется только одна линия шины и CAN земля. Сообщество Инженеров Автомобилестроения (SAE) разработало спецификацию SWC трансивера (J2411).
<i>SI unit</i> <b>Система СИ</b>	Международная система единиц физических величин, определенная стандартом ISO 1000:1983.
<i>sleep mode</i> <b>спящий режим</b>	CAN контроллер и трансивер могут находиться в спящем режиме, либо режиме сниженного энергопотребления, отключаясь при этом от шины.
<i>SRDO</i> <b>SRDO[эсэрдэо]</b>	В безопасном CANopen протоколе определен безопасный объект данных (Safety Related Data Object), который состоит из двух CAN сообщений. Поле данных второго сообщения формируется из побитно инвертированных данных первого сообщения.
<i>SRVT</i> <b>SRVT[эсэрвити]</b>	Время достоверности безопасного объекта данных (Safety Related object Validation Time) определяет максимальный интервал времени между двумя CAN сообщениями, составляющими SRDO.
<i>start of frame (SOF)</i> <b>начало кадра (SOF)</b> <b>[соф]</b>	Самый первый бит каждого кадра данных и кадра удаленного запроса. Состояние SOF всегда доминирующее.
<i>star topology</i> <b>топология звезда</b>	В некоторых легковых автомобилях используется звездообразная топология CAN сетей, а терминаторы сети подключены в центре этой звезды.
<i>Stopped state</i> <b>состояние останова</b>	NMT состояние, в котором обрабатываются только NMT сообщения и при некоторых условиях также передаются сообщения контроля ошибок.
<i>stuff-bit</i> <b>стаф-бит</b>	Всякий раз, когда CAN передатчик обнаруживает в битовом потоке 5 последовательных бит одного знака, он автоматически производит вставку бита противоположного знака (стаф-бит). CAN приемник автоматически удаляет стаф-биты, так что принятое сообщение становится идентичным переданному. Бит-стафинг используется для автоматической ресинхронизации в схеме битового хронирования CAN модуля.

<i>stuff error</i> <b>ошибка бит стафинга</b>	Ошибка бит-стафинга происходит, когда на шине обнаруживается шестой последовательный бит одного знака в полях начала кадра, арбитража, контрольном поле, поле данных и поле CRC.
<i>sub-index</i> <b>субиндекс</b>	8-битовый субадрес в объектном словаре для реализации доступа к элементам массивов и записей.
<i>substitute remote request (SRR)</i> <b>замена удаленного запроса (SRR) [эсэрэр]</b>	Бит кадра расширенного формата, замещающий RTR (бит удаленного запроса), вслед за первой частью идентификатора (11 бит). SRR бит имеет рецессивный уровень.
<i>suspend transmission</i> <b>отложенная передача</b>	В пассивном к ошибке состоянии CAN контроллер должен дополнительно ожидать 8 единиц битового времени до того, как начинать передачу очередного кадра данных или удаленного запроса.
<i>SYNC message</i> <b>посылка синхронизации SYNC [синк]</b>	Специальное CANopen сообщение, дающее указание принимающим узлам считать значения входов, отображенных в синхронные TPDO. Получение этого сообщения также предписывает узлу установить значения выходов, которые были приняты предшествующим синхронным RPDO.
<i>sync segment (Sync_Seg)</i> <b>сегмент синхронизации</b>	Часть битового времени, служащая для синхронизации различных узлов сети. В пределах сегмента синхронизации ожидается появление фронта сигнала.

## T

<i>termination resistor</i> <b>резистор-терминатор</b>	В высокоскоростных CAN сетях с линейной топологией для подавления отражений сигнала к обоим концам шины подсоединяются резисторы-терминаторы.
<i>thick cable</i> <b>толстый кабель</b>	Толстый кабель используется в DeviceNet сетях, протяженностью свыше 100 метров.
<i>thin cable</i> <b>тонкий кабель</b>	Тонкий кабель используется в DeviceNet сетях для ответвлений и в случае, если протяженностью сети не превышает 100 метров.
<i>TIME message</i> <b>посылка временной метки TIME [тайм]</b>	Стандартизированное CANopen сообщение, которое содержит значение абсолютного времени, представленное шестью байтами в виде числа миллисекунд после полуночи и количества дней после 1 января 1984 года.
<i>time quanta</i> <b>квант времени</b>	Единица атомарного времени для CAN сети.

<i>time stamp</i> <b>временная метка</b>	Некоторые CAN контроллеры обеспечивают возможность присвоения временной метки каждому принятому сообщению. В протоколе TTCAN уровня 2 передающие узлы также должны обладать способностью захватывать значение времени и включать временную метку в поле данных текущего сообщения.
<i>time-triggered</i> <b>синхро-временной</b>	Синхро-временные сообщения передаются в predetermined интервалы времени (временные слоты). Это приводит к необходимости глобальной синхронизации времени и предотвращению автоматической повторной передачи искаженных сообщений. Синхро-временные CAN коммуникации стандартизированы в ISO 11898-4 (TTCAN).
<i>Topology</i> <b>топология</b>	Физическая структура сетевых соединений, например, шина, кольцо, звезда, древовидная топология.
<i>TPDO</i> <b>TPDO [тэпэдэо]</b>	TPDO (передаваемый объект данных процесса) является коммуникационным объектом, который передается CANopen устройством.
<i>transmission buffer(s)</i> <b>передающие буферы</b>	Локальная память CAN контроллера, используемая для хранения ожидающих передачи сообщений.
<i>transmission request</i> <b>запрос на передачу</b>	Внутреннее событие в CAN контроллере, предвещающее передачу сообщения.
<i>transmission time capture</i> <b>захват момента передачи</b>	В протоколе TTCAN уровня 2 должен обеспечиваться захват момента времени передачи бита начала кадра (SOF) опорного сообщения.
<i>transmission type</i> <b>тип передачи</b>	Объект CANopen, определяющий режимы приема и передачи PDO.
<i>transmit error counter (TEC)</i> <b>счетчик ошибок передачи (TEC) [тэж]</b>	Внутренний счетчик ошибок передачи CAN контроллера В некоторых контроллерах значение счетчика TEC доступно для чтения.
<i>Transmitter</i> <b>передающий узел</b>	Узел, порождающий кадр данных или удаленного запроса. Он остается передающим узлом, пока шина вновь не станет свободной либо этот узел не проиграет арбитраж.
<i>tree topology</i> <b>древовидная топология</b>	Сетевая топология с магистральной соединительной шиной и линиями ответвлений. Не снабженные терминаторами ответвления могут вызывать отражения сигнала, уровень которых не должен превышать критического значения.

<i>TSEG1</i> <b><i>TSEG1 [mce21]</i></b>	Значение TSEG1 включает в себя сегмент распространения (Prop_Seg) и фазовый сегмент 1 (Phase_Seg 1) битового времени.
<i>TSEG2</i> <b><i>TSEG2 [mce22]</i></b>	Значение TSEG2 равно длительности фазового сегмента 2 (Phase_Seg 2) битового времени.
<i>TTCAN</i> протокол <b><i>протокол TTCAN [микан]</i></b>	Протокол высокого уровня, определяющий синхро-временные коммуникационные режимы в сети CAN. При использовании TTCAN протокола CAN контроллеры должны обладать возможностью отключать автоматическую повторную передачу искаженных сообщений. Кроме того, контроллеры должны быть способны захватывать 16-битовое значение таймера при передаче бита начала кадра (SOF) для того, чтобы включить это значение в текущее сообщение.

## V

---

<i>value definition</i> <b><i>определение значений</i></b>	Подробное описание диапазона значений в CANopen профилях.
<i>value range</i> <b><i>диапазон значений</i></b>	Атрибут CANopen объекта, определяющий его допустимые значения.

## W

---

<i>wake-up procedure</i> <b><i>процедура пробуждения</i></b>	Специальная процедура может быть использована для пробуждения CAN узлов, находящихся в спящем режиме, либо режиме сниженного энергопотребления.
---	---

КОБЗЕВ Александр Архипович,  
МИШУЛИН Юрий Евгеньевич,  
НЕМОНТОВ Владимир Александрович  
и др.

## БОРТОВЫЕ СЕТИ ТРАНСПОРТНЫХ СРЕДСТВ

Учебное пособие

*Издается в авторской редакции*

Подписано в печать . . .

Формат 60■84/16. Бумага для множит. техники. Гарнитура Таймс.  
Печать на ризографе. Усл. печ. л. , . Уч.-изд. л. , . Тираж 50 экз.

Заказ .

Издательство

Владимирского государственного университета.  
600000, Владимир, ул. Горького, 87.