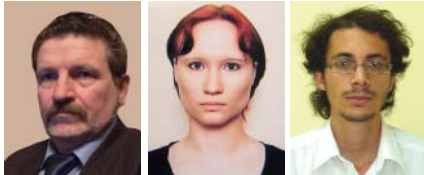


# Открываем сетевые ресурсы матроники транспорта

А. Б. ШАДРИН, академик РАН, доктор технических наук ИО и ИПК РАН, профессор СЗТУ,

С. А. РОМАШОВА, инженер ФГОУ ВПО «Северо-Западный государственный заочный технический университет»,

И. КАСТИЛЬО ЧАГИН, инженер ФГОУ ВПО «Северо-Западный государственный заочный технический университет»



**Сетевые ресурсы электроники — net&eLearning: mecha auto, teletronice (e-технологии в элементах матроники) — требуют совершенствования построения всех уровней виртуальных сетей (VPNT)**

**для освоения ресурсов управления качеством сервисов на всех этапах обучения и управления динамическими процессами транспорта.**

С учетом совершенствования оптических сетей для верхнего уровня сервисных сетевых процессов обучения и управления динамическими процессами транспорта (VPNT) [1,2] исследована и выбрана технология — Multiprotocol Label-Switching (MPLS, мультипротокольной коммутации по меткам):

1. Формируются таблицы маршрутизации в маршрутизаторах или коммутаторах.
2. Протокол распределения меток — Label Distribution Protocol (LDP) использует отраженную в таблицах топологию маршрутизации для определения значений меток, указывающих на соседние устройства, и формируются маршруты с коммутацией по меткам — Label Switched Paths (LSP).
3. Входящий пакет поступает на пограничный Label Switch Router (LSR) — выбирает и присваивает метку, которая записывается в заголовок пакета, после чего пакет передается дальше.
4. LSR, находящийся в опорной сети, считывает метки каждого пакета, заменяет старые метки новыми (метки определяются по локальной таблице) и передает пакет дальше. Эта операция повторяется в каждой точке передачи пакета по опорной сети.
5. На выходе пакет попадает в пограничный LSR, который удаляет метку, считывает заголовок пакета и передает его по месту назначения.

MPLS разделяет трафик по меткам. Верхняя метка направляет пакет VPNT к оконечной части сети провайдера (PE) — маршрутизатору. Вторая метка показывает, куда PE-маршрутизатор должен направить пакет VPNT. Для формирования VPNT на базе MPLS между PE и оконечной частью заказчика VPNT CE-маршрутизаторами надо использовать схемы IP forwarding: PE связывает CE с таблицей передачи (forwarding table), в

которой хранятся только те маршруты, которые доступны данному CE. CE используют External Border Gateway Protocol (EBGP) для обмена данными о маршрутной информации с PE. Адрес VPN-IPv4 состоит из 12 байтов: 8 байтов — «различители маршрута» — Route Distinguisher (RD); 4 байта — адрес IPv4. IP-заголовок пакета не рассматривается, пока пакет не поступит на оконечный PE.

PE и CE общаются между собой с помощью Routing Information Protocol (RIPv2) или статических маршрутов. Членство в VPN зависит от BGP — присваивает Route Distinguisher (RD) — неизвестны конечным узлам. BGP надо использовать для распространения таблицы — Forwarding Information Base (FIB) и достижения безопасности VPNT с помощью логического разделения трафика.

Трафик при построении VPNT на основе MPLS не шифруется. PE «делится»

между различными узлами CE сети и процессами в VPNT.

На среднем уровне VPNT следует использовать Any Transport over MPLS (AToM) — передаются не пакеты, а целые блоки второго уровня. В таких VPNT упрощается синтез защищенных «псевдопроводов» — логических каналов. На базе AToM исследованы и реализованы для задач VPNT модификации: Ethernet over MPLS (EoMPLS) и Virtual Private LAN Service (VPLS). Облако MPLS/VPLS — «большой виртуальный коммутатор». Пограничные устройства узлов — коммутаторы. С помощью Point-to-Point Protocol (PPP) узлы подключаются к точке присутствия — Point of Presence (POP) и формируют на основе MPLS элементы VPNT. Используя имя домена в ходе аутентификации PPP или DNIS коммутатора, создается туннель с помощью Layer 2 Tunneling Protocol (L2TP). Затем сессия PPP передается PE. Интерфейсы группируются в VPN на основании имени домена или номера DNIS. В ходе сессии VPNT используется технология аутентификации и учета RADIUS. После этого сессия PPP полностью принадлежит данной VPNT и может направлять пакеты к месту назначения. При этом может использоваться любой узел доступа, поддерживающий L2TP, и реализуются современные технологии взвешенных очередности —

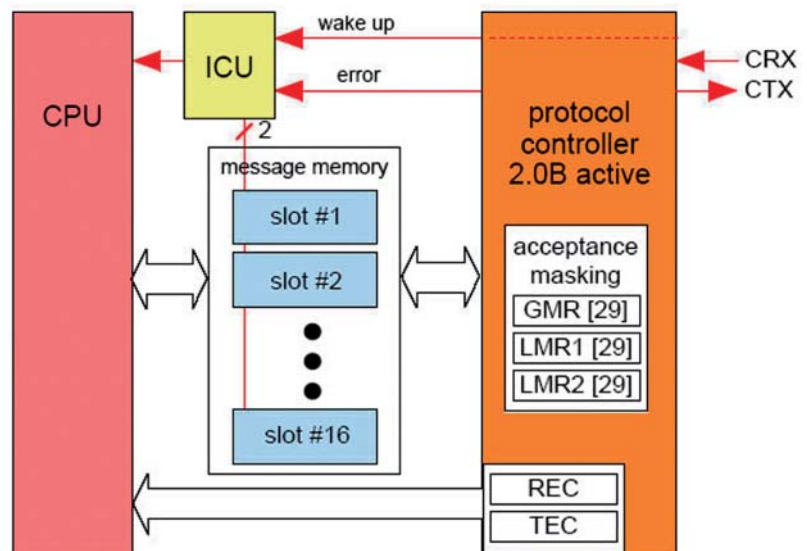


Рис. 1. Схема управляемого бортового микросервера

Weighted Fair Queuing (WFQ), справедливой очередности с учетом классов — Class Based Weighted Fair Queuing (CBWFQ).

Через web-интерфейс узла можно получить доступ к меню сервисов для различных услуг в VPNT.

На уровне оптических/коаксиальных участков VPNT использовался Cisco uBR72xx/VXR с беспроводным модулем действует как PE. Трафик с беспроводных узлов (CPE) передается в VPNT с учетом Service Identifiers (SID) из фреймов DOCSIS. Каждый CPE может поддерживать множество идентификаторов SID.

С учетом распространенности телефонных сетей в элементах обучающих технологий на базе VPNT применяются абонентские линии DSL для выделенного доступа к VPNT.

Отладку и моделирование сетевых сервисов с учетом особенностей построения VPNT на верхнем уровне целесообразно выполнять на основе: Dynamips (эмулятор) и Dynagen (front-end интерфейс Dynamips для слежения за конфигурированием портов, мостов и управления устройствами).

Для совершенствования хранения данных в VPNT [1,2] выбираем Fibre Channel over Ethernet (FCoE) через full duplex IEEE 802.3 (10Gbps Ethernet) и Convergence Enhanced Ethernet (CEE).

FCoE представляет в плане Ethernet транспортируемый протокол верхнего уровня, а в плане сетевой техники — расширение транспорта через Ethernet без потерь.

Инкапсуляция фрейма выполняется посредством построения соответствия, или отображения на Ethernet. Протокол имеет стек уровней, каждый из которых предоставляет определенный набор функций. Стек FC содержит пять уровней, от FC-0 до FC-4. В стек Ethernet входит два уровня: физический и канальный. FCoE обеспечивает трансляцию данных с уровня FC-2 в стек протоколов Ethernet и обеспечивает управление на уровнях FC-3 и FC-4 через Ethernet. Отметим, что в фрейме FCoE первые 12 байт — MAC отправителя и получателя. 32-битовый IEEE 802.1Q Tag расширяет возможности создания VPNT. 16 бит отведены для параметра Ethertype = FCoE. 4-битовое поле Ver — определяет версию. 100 зарезервированных битов и за ними 8-битовый флаг Start of Frame (SOF), а затем данные — FC и далее 32-битовый Frame Check Sequence (FCS). При подсоединении FC SAN к коммутатору FCoE фрейм извлекается и обрабатывается без дополнительного шлюза.

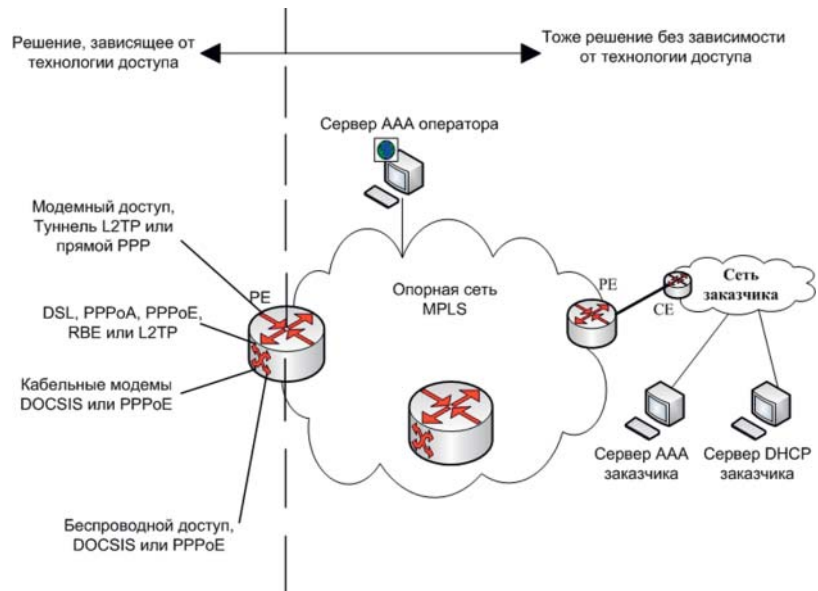


Рис. 2. Элементы верхнего уровня VPNT на базе MPLS

FCoE — объединение двух сетевых технологий. Коммутаторы FCoE поддерживают обе и имеют консолидированный ввод-вывод: реализованы коммутационные функции как Ethernet, так и FC; логический модуль FCoE внутри коммутатора FC и инкапсулирует FC-фреймы в FCoE-фреймы, и выполняет обратный процесс; FC-трафик поступает в модуль из FC-портов, инкапсулируется в FCoE-фреймы и выходит из Ethernet-портов уже в формате FCoE. Важно, что реализуется и обратная последовательность операций. Модуль FCoE имеет MAC-адрес, который служит адресом отправителя или получателя в зависимости от направления трафика.

FCoE упрощает консолидацию ввода-вывода на уровне «доступ — распределение — ядро». На базе решений Brocade коммутатор серии 8000 24 порта FCoE 10Gbit и 8 портов 8Gbit FC, Converged Network Adapters (CNA), Brocade 1010 и 1020 с двумя портами упрощается центр хранения данных VPNT в части соединения на скорости 10 Гбит/с, подключения по технологии FCoE к серверам выделенных и локальных сетей. Результат — освоение ресур-

сов FCoE, CEE, 10 Gigabit Ethernet. FCoE реализует возможность FC и интегрирует преимущества Ethernet, расширяя возможности администрирования Ethernet и FC-сети в VPNT.

Виртуализация серверов увеличивает требования к вычислительным ресурсам, а внедрение в VPNT технологии FCoE позволяет использовать больше ресурсов Ethernet, и отпадает необходимость полной зависимости от FC. Консолидация на базе FCoE упрощает инфраструктуру центров данных в части затрат на электропитание, управление, обслуживание, обучение, запасные части и охлаждение элементов. Fibre Channel сокращает расходы на кабели, адаптеры и коммутаторы и выгодно отличается от типовых центров хранения данных на базе протокола iSCSI.

Для реализации автономных и бортовых сервисов удаленного многоцелевого обучения и управления на нижних уровнях VPNT необходимо использовать технологии: Special Equipment of Logical Management of Signals (SELMS); Converter of Signals (CS), Transceiver of a Signal of Satellite Management (TSSM), Transmitter of Global Positioning System

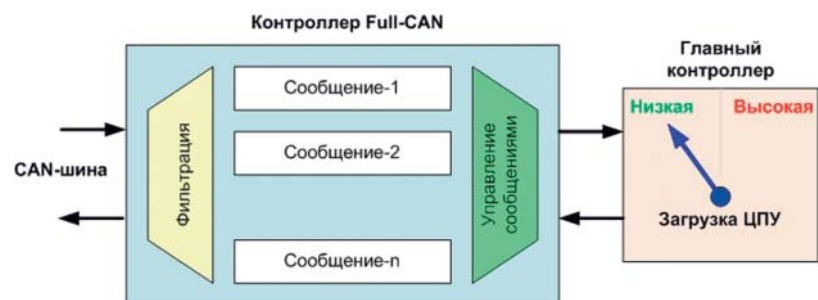


Рис. 3. Элементы фильтрации сообщений в схеме контроллера CAN

Критерии	MPLS/VPN	ATM/Frame Relay	IPSec	GRE
Масштабируемость	Высокая	Низкая	Низкая	Низкая
Требования к оператору	Поддержка технологий MPLS/VPN	Поддержка ATM/Frame Relay	Нет	Нет
Требования к клиенту	Нет	Поддержка ATM/Frame Relay	Наличие средств шифрования	Поддержка туннелирования трафика
Обеспечение целостности и конфиденциальности	Нет	Нет	Да	Нет
Пересечение адресных пространств узлов из разных VPN	Допускается	Допускается	Необходимость в NAT со стороны клиента	Необходимость в NAT со стороны клиента

Рис. 4. Сравнение основных технологий по организации VPN

(TGPS); Wireless Radio Transmitter (WRT)[1].

Для комплексного обучения всем элементам серверно-сетевого управления процессами на базе Control Area Network (CAN) на нижних уровнях VPNT целесообразно использовать микроконтроллеры типа Infineon XC164. Взаимодействие CPU с памятью организуется PMU (управляет доступом к блоку памяти программ — IMB и к внешней памяти) и DMU (управляет поступлением данных из памяти данных и периферии), связанных с контроллером внешней шины (EBC). Блоки встроенной памяти (программ, двухпортовая RAM, SRAM данных) и подсистема встроенной периферии соединены отдельными шинами. LxBus используется для соединения с внешними ресурсами и рядом дополнительных встроенных ресурсов. Расширенная структура шин повышает общую производительность системы, обеспечивая одновременную работу различных подсистем. Основные блоки CPU: Instruction Fetch Unit (IFU), располагающий широкополосным интерфейсом выборки, FIFO команд, стеком возврата и обеспечивающий высокоэффективную обработку ветвлений, вызовов и циклов с предсказанием потока команд; блок обработчика включений/исключений — Hardware Traps; конвейер команд — Instruction Pipeline (IPIP); блок адресов и данных — Address and Data Unit (ADU); Arithmetic and Logic Unit (ALU); Multiply and Accumulate Unit (MAC) с 16-разрядным умножителем с формированием 32-разрядного результата. В MAC 75% всех команд выполняется за один цикл CPU. Register File (RF) — 5-портовый регистровый файл с тремя независимыми банками регистров. 3-входовый буфер обратной записи — Write Back Buffer (WB). C166S-V2 оптимизи-

рован под выполнение высокопроизводительных команд и минимальное время реакции (отклика) на внешние прерывания. Интеллектуальные периферийные модули уменьшают загрузку CPU и расширяют возможности XC166. Память микроконтроллеров серии XC166 реализована по комбинированной архитектуре — Harvard и von Neumann. В Harvard память кодов и данных разделена и позволяет организовать одновременный доступ к кодам и к данным. В von Neumann коды и данные размещаются в едином адресном пространстве, составляющем 16 Мбайт, что позволяет эффективно использовать память. Расположенная в блоке IMB встроенная flash-память кодов и констант микроконтроллеров XC166 занимает 128 Кбайт адресного пространства и подразделяется на секторы — четыре по 8 Кбайт, 32 Кбайта и 64 Кбайта. Flash-память, оснащенная встроенным механизмом коррекции ошибок (ECC), взаимодействует с CPU (через PMU) посредством 64-разрядного ин-

терфейса, что позволяет CPU считывать четыре 16-разрядные команды инструкции за один раз. Через блок PMU с CPU взаимодействуют кэш-память программ (RAM емкостью 2 Кбайта) и ROM начального запуска (Start up ROM).

Память данных: регистровое двухпортовое ОЗУ (DPRAM) с обращением короткой адресацией и возможностью одновременного доступа для записи и чтения; блок памяти данных. К блоку памяти данных CPU обращается посредством 16-разрядного блока DMU. Такая организация памяти позволяет CPU одновременно взаимодействовать с 64-разрядным кодом flash-памяти и с тремя 16-разрядными источниками данных (два в двухпортовом ОЗУ и один источник данных памяти данных или периферии).

Модуль таймеров общего назначения (GPT12E) — структура многофункциональных таймеров/счетчиков, которую можно использовать в большом количестве связанных с временными промежутками задач — определения момента события и счета, определения ширины импульса и измерения скважности, генерации импульсов или их размножения. Модуль GPT12E содержит пять 16-разрядных таймеров, организованных в два отдельных модуля — GPT1 и GPT2. Каждый таймер каждого модуля может работать автономно или же может быть объединен с другим таймером того же самого модуля. Каждый таймер может быть индивидуально сконфигурирован в один из четырех основных режимов работы — таймера, стробируемого таймера, в режим счетчика и в режим инкрементного интерфейса (только таймеры GPT1). Максимальное разрешение таймеров модуля

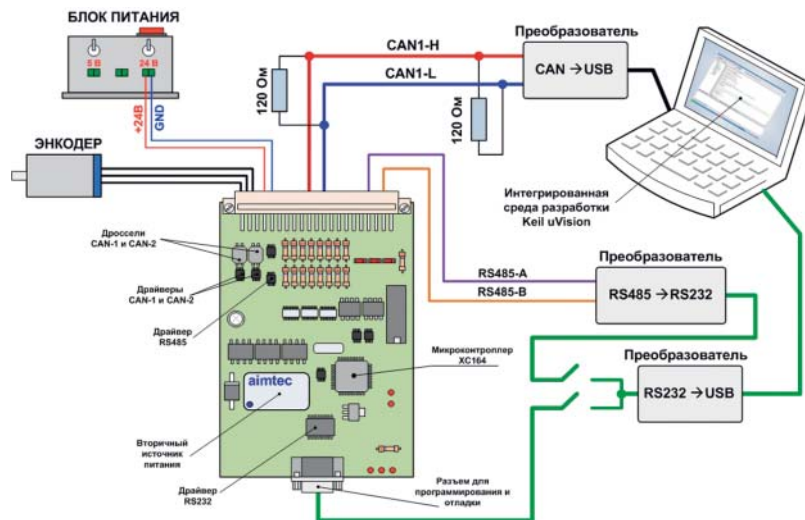


Рис. 5. Схема стенда для настройки контроллеров CAN

GPT1 — 8 циклов сигнала тактирования CPU. Максимальное разрешение таймеров модуля GPT2 составляет 4 цикла сигнала тактирования CPU.

Для обеспечения многопроцессорной связи организован механизм распознавания адреса среди байтов данных. 13-разрядный генератор скорости поступления импульсов обеспечивает формирование ASC0 отдельного последовательного синхросигнала с очень точной настройкой предварительным делителем, выполненным по схеме дробного делителя. Модулем формируются прерывания по пустому буферу передатчика, по последнему биту фрейма, по полному буферу приемника, по ошибке и по запуску и окончанию автоматического детектирования скорости поступления импульсов.

Модуль TwinCAN, заменивший используемый в микроконтроллерах семейства S166 модуль CAN, взаимодействует с CPU посредством шины LXBus, работающей аналогично внешней шине.

К шине LXBus подсоединены два (как опция) или один узел Full-CAN. Передача и прием CAN фреймов обрабатываются в соответствии со спецификацией CAN V2.0 part B (active). Каждый из двух Full-CAN-узлов может получать и передавать стандартные фреймы с 11-разрядными идентификаторами и расширенные фреймы с 29-разрядными идентификаторами. Два узла CAN используют ресурсы TwinCAN-модуля совместно, оптимизируя обработку трафика шины CAN и минимизируя нагрузку CPU.

Каждый из 32 возможных объектов сообщений может быть индивидуально назначен к одному из двух узлов CAN. Функцией шлюза обеспечивается автоматический обмен данными между двумя отдельными системами CAN, что способствует снижению загруженности CPU и улучшает поведение всей системы в реальном времени.

Сигнал тактирования битов в узлах CAN формируется из сигнала тактирования периферии (fCAN) и может быть запрограммирован в диапазоне до 1 Мбод. Каждый узел CAN соединяется с приемопередатчиком шины.

Разработку и отладку программ для нижних уровней (бортовых сетей) в VPNT целесообразно проводить в среде IDE uVision 3 фирмы Keil. В uVision имеется C/C++ компилятор для XC164 с возможностью отладки (встроенный debugger) через интерфейсы JTAG или RS232. При разработке программ среда IDE uVision 3 настраивается на микроконтроллер из семейства XC164.

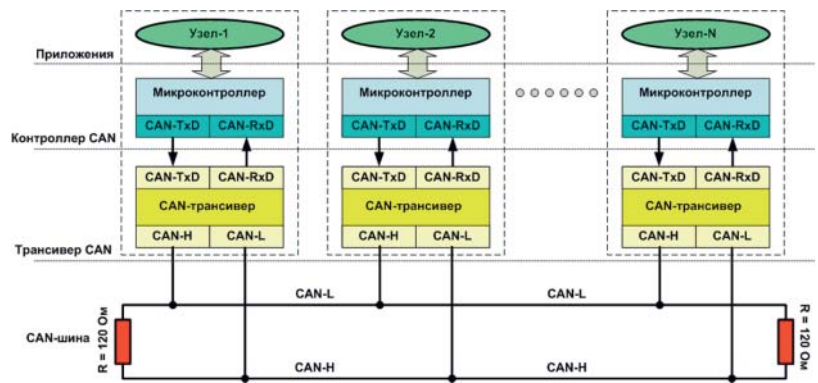


Рис. 6. Схема серверно-сетевое управления

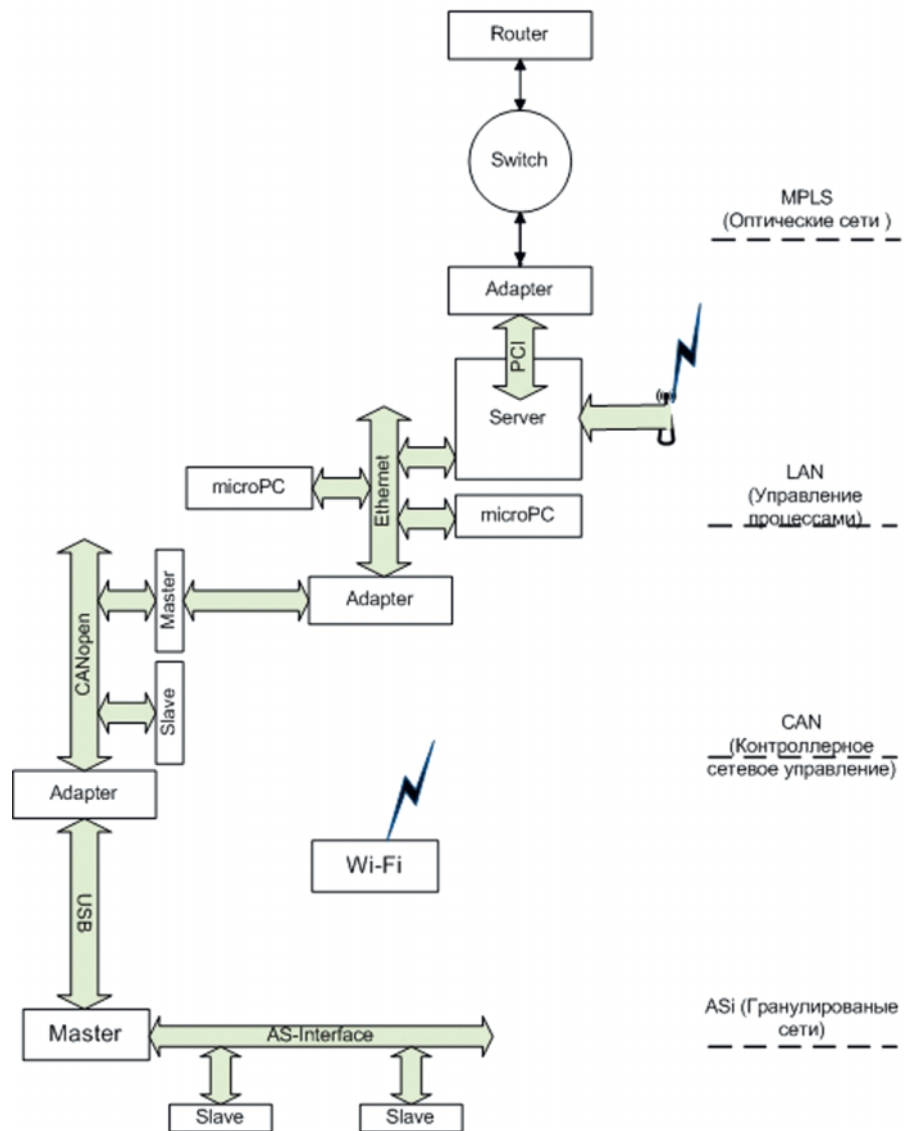


Рис. 7. Схема элементов иерархии VPNT

С учетом увеличения до 75% элементов матроники в современном транспорте и развития оптических сетей необходимо своевременно открывать и обучать ресурсы e-технологий для реализации защищенных сетей (VPNT) и развития на этой базе новых элементов серверно-сетевое управления процессами в транспортном комплексе.

ЛИТЕРАТУРА

1. Тузов Л. В. Роль автотроники и телематики в интеграции сервисов для транспорта / Л. В. Тузов, А. Б. Шадрин // *Транспорт*. — 2008. — № 1. С. 54–57.
2. Тузов Л. В. Возможности матроники транспорта / Л. В. Тузов, А. Б. Шадрин // *Исследования, проектирование и эксплуатация судовых ДВС: тр. II-го Международного научно-технического семинара*. — СПб: ГУВК, 2008. — С. 342–350.