

Протокол обмена данными адаптера для платежных систем EW-UsbMDB

Версия 1.3 (20.05.2011), данная версия поддерживается оборудованием с прошивкой версии 1.3 и выше

Взаимодействие адаптера и ПК происходит в текстовом режиме по каналу USB (эмуляция RS232 скорость 115200 8N1).

При первом подключении требуется установить драйверы устройства по ссылке <http://www.ftdichip.com/Drivers/VCP.htm>

Каждая команда от ПК и ответ ведомого представляют из себя текстовую строку, начинающуюся с символа '#', и завершающуюся символами '\r' '\n'

Все передаваемые числовые значения представляются в 16-ричном формате (строчные буквы).

Список команд стандартного набора (модификация -S)

Команда	Описание	Ответ	Описание
#vreset	Перезагрузка переходника	#vreset=ok	
#vserial=<serial> (#vserial?)	Установка серийного номера (макс. 32 символа)	#vok #vver=<text> (#vserial=<serial>)	
#vdev	Тип устройства	#vdev= usbmdb	
#vver	Версия ПО	#vver=<string>	
#vmanu	Информация о производителе	#vmanu=<string>	
#vpoll=<t>	Задание интервала поллинга MDB-шины	#vok	t = по умолчанию 500мс, единица = 10мс
#vcardtm=<t>	Задание интервала ожидания MDB-модуля безнал.оплаты	#vok	t = по умолчанию 10с, макс. 20с, единица = 1с
#vdog=<t>	Включение сторожевого таймера потери связи с ПК (после включения данной функции переходник в случае отсутствия любых команд от ПК в течение времени более чем <t> прекращает опрос всех подключенных платежных систем и выдает событие #veve=DOG	#vok	Единица = 1с. Если задано 0 – то функция отключается
пустая строка (только символы \r\n)	Пустая команда. Предназначена лишь для сброса сторожевого таймера заданного командой #vdog	Без ответа	
#vpow	Измерение напряжения питания платежных систем	#vpow=<v>	<v> - напряжение питания в мВ
#vin	Вывод информации о состоянии входов концевиков	#vin=<a>,	Состояния концевиков <a>, : 0 – разомкнут 1 - замкнут
#vcena=<ena> (#vcena?)	Включение поддержки монетоприемника <ena> = 0, 1	#vok	
#vcask	Запрос статуса монетоприемника	#vcask=<state>	<state> - состояние монетоприемника:
#vcinfo	Запрос информации о монетоприемнике	#vcinfo=<routing>,<level>,<scale>,<decimal>,<coin type credits... (up to 16 channels separated with commas)>	<routing> - битовое поле номеров каналов, которые направляются в тубы монетоприемника (2 байта) <level> - уровень совместимости монетоприемника <scale> - масштаб денежных величин монетоприемника

			<decimal> - положение десятичной точки денежных величин монетоприемника <coin type credits> - массив от 1 до 16 переменных через запятую, указывающих номиналы монет для туб
#vctubes	Запрос заполнения туб	#vctubes=<full>,<n0>,...<n15>	<full> - битовое поле флагов переполнения каналов (2 байта) <nх> - количество монет в канале х
#vcini=<ena>,<manual>	Разрешение приема монет для каждого канала <ena> - битовое поле разрешенных каналов (2 байта) <manual> - битовое поле каналов для которых разрешена ручная выдача монет (2 байта)	#vok	
#vcdisp=<chn>[,<amount>]	Выдача монет из указанного канала <chn> - канал <amount> - количество монет к выдаче (не более 15шт, если параметр опущен то принимается значение по умолчанию 1 монета)	#vok	
#vbena=<ena> (#vbena?)	Включение поддержки купюроприемника <ena> = 0, 1	#vok	
#vbask	Запрос статуса купюроприемника	#vbask=<state>	
#vbinfo	Запрос информации о купюроприемнике	#vbinfo=<escrow>,<stacker>,<level>,<scale>,<decimal>,<bill credit type (up to 16 channels separated with commas)>	<escrow> - указание наличия функции определения номинала купюры перед ее укладкой (ff – наличие, 00 – отсутствие) <stacker> - емкость стекера в купюрах <level> - уровень совместимости купюроприемника <scale> - масштаб денежных величин купюроприемника <decimal> - положение десятичной точки денежных величин купюроприемника <bill type credits> - массив от 1 до 16 переменных через запятую, указывающих номиналы купюр для каналов
#vbsecu=<high>	Задание уровней высокой безопасности детектирования купюр <high> - битовое поле каналов в которых задан уровень высокой безопасности (2 байта)	#vok	
#vbini=<enable>,<escrow>	Задание разрешенных купюр к приему и функции предварительного определения номинала купюры Escrow <enable> - битовое поле разрешенных к приему купюр (2 байта) <escrow> - битовое поле купюр для которых разрешено предварительное определение номинала	#vok	

#vbaccept=<acc>	Ответ забирать купюру (1) или нет (0) на событие что купюра в позиции Escrow	#vok	
#vbstacker	Запрос количества купюр в стекере	#vbstacker=<fullflag>,<amount>	<fullflag> = 1 – стекер переполнен <amount> - количество купюр в стекере
#vaena=<ena> (#vaena?)	Включение поддержки безналичного модуля <ena> = 0, 1	#vok	
#vaask	Запрос статуса безналичного модуля	#vaask=<state>	
#vainfo	Запрос информации о безналичном модуле	#vainfo=<level>,<scale>,<decimal>,<options>,<maxresponsetime>	<level> - уровень совместимости купюроприемника <scale> - масштаб денежных величин безналичного модуля <decimal> - положение десятичной точки денежных величин безналичного модуля <options> - битовые поля опций безналичного модуля: бит 0 – возможность пополнения носителя бит 1 – наличие режима многократной продажи в одной сессии (Multivend) бит 2 – модуль имеет собственный дисплей бит 3 – модуль поддерживает Vend/cash sale команду
#vaprices=<min>,<max>	Установка минимальной и максимальной цен товаров	#vok	
#vaini=<ena>	Разрешение работы и поиска карт	#vok	
#vavend=<price>,<item>	Команда начала продажи для вычитания стоимости товара с кредита карты	#vok – сначала, а затем после транзакции с картой #vavend=<success>,<deducted> #verr=NORESP – нет ответа по таймауту	<price> - цена продаваемого товара <success> - подтверждение списания кредита с карты: 1 – успешно списан 0 – ошибка <deducted>: сумма средств, реально списанная с карты (с учетом возможной скидки)
#vacancel	Команда отмены начатой продажи. Например клиент передумал покупать. Модулю следует попытаться вернуть списанный кредит на карту	#vok	
#varesult=<success>[,<item>]]	Команда результата продажи (для возврата кредита на карту в случае ошибки при продаже)	#vok	Значения <success>: 0 – ошибка продажи 1 – продажа осуществлена успешно <item> - номер товара который был продан (если продажа была успешно выполнена)
#vacomplete	Команда завершения сессии	#vok	

#vareval=<credit>	Пополнение кредита карты на заданную сумму	#vok – сначала, а затем после транзакции с картой #vareval=<success> #verr=NORESP – нет ответа по таймауту	
-------------------	--	--	--

Список команд расширенного набора (модификации –S и –SX)

#vevmcena=<ena> (#vevmcena?)	Разрешение работы адаптера в режиме EХЕ мастер (эмуляция монетоприемника в режиме EХЕ), опрос модуля аппарата, включение поллинга событий #veve=EХЕVМC		
#vevmcask	Запрос текущего статуса модуля аппарата	#vevmcask=<state>	
#vevmcvend	Отправка команды в VMC на выдачу товара	#vok – подтверждение приема команды Затем с задержкой до 120сек: #vevmcvend=<success>	Значения <success>: 0 – ошибка продажи 1 – продажа осуществлена успешно
#vevmcid	Запрос команды идентификации VMC	#vevmcid=<Size>	Чтение данных идентификации командой #vevmcaudit
#vevmcaudit	Запрос размера накопленных данных аудита (в парах адрес-данные)	#vecpuaudit=<Size>	
#vevmcrdaudit	Чтение пары адрес-данные аудита	#vecpurdaudit=<addr>,<data> #verr=NODATA	
#vevmcdisplay=<BaseUnit>,<Scale>,<DecPt>,<NoChange>	Отправка данных для отображения на дисплее VMC	#vok	
#vecpuena=<ena> (#vecpuena?)	Разрешение работы адаптера в режиме EХЕ мастер (эмуляция монетоприемника в режиме EХЕ), опрос безналичного модуля, включение поллинга событий #veve=EХЕСPУ	#vok	
#vecpuask	Запрос текущего статуса безналичного модуля	#vevmcask=<state>	
#vecpuread	Запрос на загрузку данных карты (если <CardInAperture>=1	#vecpuread=<CardCredit> #verr=NOCARD	
#vecpuaccept=<Value>	Передать денежное значение в модуль	#vok #verr=NOCARD	
#vecpudec	Уменьшить кредит карты на загруженное ранее денежное значение по команде #vecpuaccept		
#vecpuret	Вернуть карту (предварительно прибавив к кредиту значение <Value> ранее выполненной #vecpuaccept)	#vok #verr=NOCARD	
#vecpuundo	Откатить последнюю операцию DEC (вернуть кредит на карту в случае ошибки выдачи товара)	#vok #verr=NOCARD	

#vecpuid	Запрос команды идентификации CPP	#vecpuid=<Size>	Чтение данных идентификации командой #vecpuidaudit
#vecpuidaudit	Запрос размера накопленных данных аудита (в парах адрес-данные)	#vecpuidaudit=<Size>	
#vecpurdaudit	Чтение пары адрес-данные аудита	#vecpurdaudit=<addr>,<data> #verr=NODATA	
#vecpuidisplay=<BaseUnit>,<Scale>	Отправка данных для отображения на дисплее CPU	#vok	
#vxena=<ena>	Разрешение работы адаптера в режиме EXE слейв (эмуляция модуля аппарата)	#vok	
#vxask	Запрос статуса EXE-шины	#vxask=<state>	
#vxready=<ena>,<free>	Установка состояния аппарата в режим разрешения продажи: <ena> = 1 (продажи разрешены), 0 (продажи запрещены) <free> = 1 (бесплатные продажи разрешены), 0 (бесплатные продажи запрещены)	#vok	
#vxcredit=<req>[,<value>]	Задание запроса на покупку и цены текущего выбранного товара <req> = 1 – запрос на покупку от пользователя <value> = цена товара (или номер товара если PriceHolding), если <req>=1	#vok	<credit> - значение текущего кредита
#vxvend=<success>	Результат выполнения продажи (после получения события #veve=EXE,VENDREQ) <success> - результат, если продажа выполнена то 1, если ошибка, то 0	#vok #verr=VENDFAIL	
#vxdataadd=<addr>,<data>	Задание пар данных аудита <addr>,<data> - пара аудита (в устройстве имеется свой FIFO буфер в котором хранятся эти пары до их чтения мастером_	#vxdatapair=<empty> <empty> - сколько ячеек еще свободны #verr=MEMFULL – память заполнена	
#vxdataclear	Очистка FIFO буфера аудита		
#vlisten=<type>	Разрешение функции прослушивания шины (для всех режимов)	#vok	Значения <type>: OFF – запретить прослушивание MDB – режим MDB EXE – режим EXE

Примечание: если поступившая команда не распознана или не соответствует режиму шины, который сейчас активен, переходник возвращает строку **#vunknown**

Список возможных событий

Адаптер платежных систем EW-UsbMDB

Событие	Описание
#veve=DOG	Событие, выдаваемое при срабатывании сторожевого таймера. Опрос всех платежных систем остановлен до очередной команды от ПК
#veve=COIN,STATE,<state>	Текущее состояние монетоприемника (после его изменения)
#veve=COIN,ROUTE,<accepted>,<routing>,<channel>,<intube>	Обнаружение опущенной монеты, <accepted> - если монета была принята то 1, если монета была возвращена, то 0 <routing> - направление: CASHBOX – принята в денежный ящик, TUBES – принята в тубы, REJECT – не распознана и возвращена, NOTUSED – неизвестная монета, возвращена <channel> - канал монеты <intube> - сколько монет теперь в указанной тубе
#veve=COIN,DEPOSIT,<depositflag>,<routing>,<channel>,<intube>	Монета опущена <depositflag> = 1 если монета была принята, = 0 если была возвращена <routing> - направление <channel> - канал монеты <intube> - общее кол-во монет в тубе с учетом этой Значения параметра <routing>: CASHBOX – монета в накопительный ящик TUBES – монета в тубы для выдачи сдачи REJECT – монета не распознана и возвращена назад NOTUSED – неизвестный код отработки
#veve=COIN,DISPENSE,<channel>,<number>	Монета выдана в ручном режиме <channel> - канал монеты <number> - количество выданных монет
#veve=COIN,SLUG	Неопознанная монета опущена и возвращена назад
#veve=BILL,STATE,<state>	Текущее состояние купюроприемника (после его изменения)
#veve=BILL,STACK,<billtype>	Прием купюры <billtype> - канал купюры
#veve=BILL,ESCROW,<billtype>	Купюра в позиции определения номинала, ожидается дальнейший вызов функции #vbaccept <billtype> - канал купюры
#veve=CARD,STATE,<state>	Текущее состояние безналичного модуля (после его изменения)
#veve=CARD,BEGIN,<credit>,<serial>,<pmttype>,<pmdata>	К безналичному модулю поднесена карта <credit> - текущий кредит карты <serial> - серийный номер карты, максимум 32 символа <pmttype> - тип карты <pmdata> - данные типа карты
#veve=CARD,APPROVED[,<value>]	Подтверждение выполненной продажи (<value> - списанный кредит)
#veve=CARD,DENIED	Подтверждение отмененной продажи
#veve=CARD,END	Подтверждение завершения сессии
#veve=CARD,COMPLETEREQ	Запрос от модуля на завершение сессии
#veve=EXEVMC,<FreeVend>,<VendInhib>,<AuditAvail>[,<VendReq>]	Текущее состояние платы автомата (режим EXE)
#veve=EXECPU,<FreeVend>,<CardRetReq>,<AuditAvail>,<CardLoa	Текущее состояние модуля безналичной оплаты (режим EXE)

ded>,<CardInAperture>	
#veve=EXE,STATE,<state>	Текущее состояние модуля (после его изменения)
#veve=EXE,VENDREQ	От монетоприемника пришел запрос на выполнение выдачи текущего выбранного товара
#veve=EXE,AUDITREAD,<addr>,<data>,<remainder>	Была прочитана пара адрес-данные аудита
#veve=EXE,DISPLAY,<Value>,<Scale>,<DecPt>,<NoChange>	Выдача информации на дисплей VMC
#veve=LISTENA,<data>	Прослушивание шины, слово канала А
#veve=LISTENB,<data>	Прослушивание шины, слово канала В

Список ошибок

Параметр #verr	Описание
OFF	Устройство, к которому обращается текущая команда, отключено
LOST	Ошибка команды
MODE	Ошибка конфликта режимов (например если активен MDB режим а мы отдаем команду для EXE)
NORESP	Нет ответа от ведомого устройства

Список состояний устройств (параметр <state>)

Параметр	Описание
LOST	Устройство исчезло с шины
ERROR[:abbr]	Ошибка (необязательный текстовый параметр аббревиатура ошибки)
READY	Готово к работе
BUSY	Занято (операция выдачи или приема)
RESET	Перезагружается
DISABLED	Устройство готово к работе, прием денег запрещен
CARDNEAR	Карта у считывателя (только для модуля безналичной оплаты)
UNKNOWN	Неизвестное состояние