

Ke-USB24A

Быстрый старт с модулем Ke-USB24A

Версия 1.01

2 марта 2008

Содержание

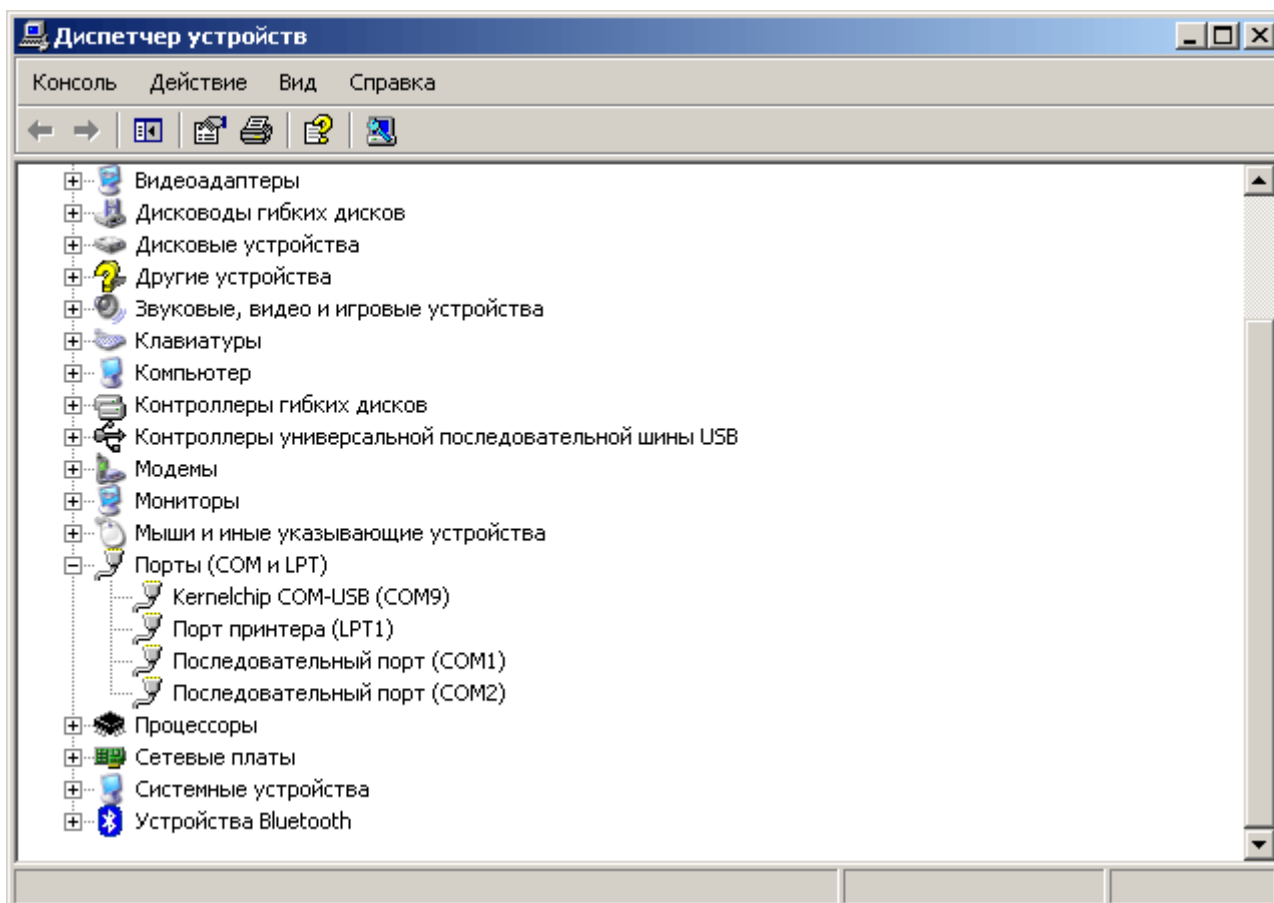
Введение	3
Первое подключение. HyperTerminal и KeTerm	3
Управление линиями ввода/вывода	9
Управление выходными линиями	9
Настройка направлений ввода/вывода	11
Управление входными линиями	11
Модуль АЦП	14
Сохранение пользовательских данных в памяти модуля	16
Управление строковым дескриптором USB	16

Введение

Данный документ поможет Вам в быстром освоении модуля Ke-USB24A, познакомит Вас на практических примерах с основными возможностями модуля.

Первое подключение. HyperTerminal и KeTerm.

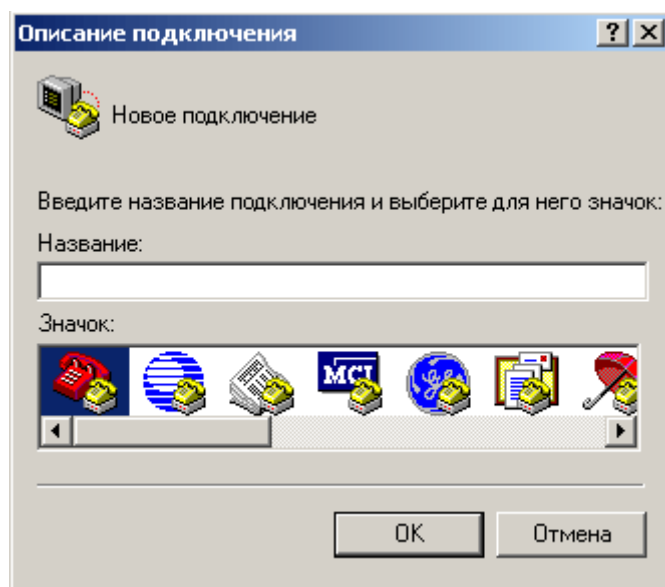
Итак, начнем. Первым делом для проведения опытов с модулем необходимо установить драйвера. Подробные инструкции по установке описаны в отдельном документе, доступном на сайте. По окончании установки, необходимо уточнить номер виртуального COM порта, который операционная система выделила для модуля. Именно через этот виртуальный порт будет возможен обмен информацией с модулем. Для этого необходимо открыть диспетчер устройств, например, через *Пуск -> Настройка -> Панель управления -> Система -> Оборудование -> Диспетчер устройств*. В ветке "Порты (COM и LPT)" должна появиться запись о новом устройстве – Kernelchip COM-USB. В данном случае, виртуальному порту был присвоен номер 9 (COM9). Запомним этот номер.



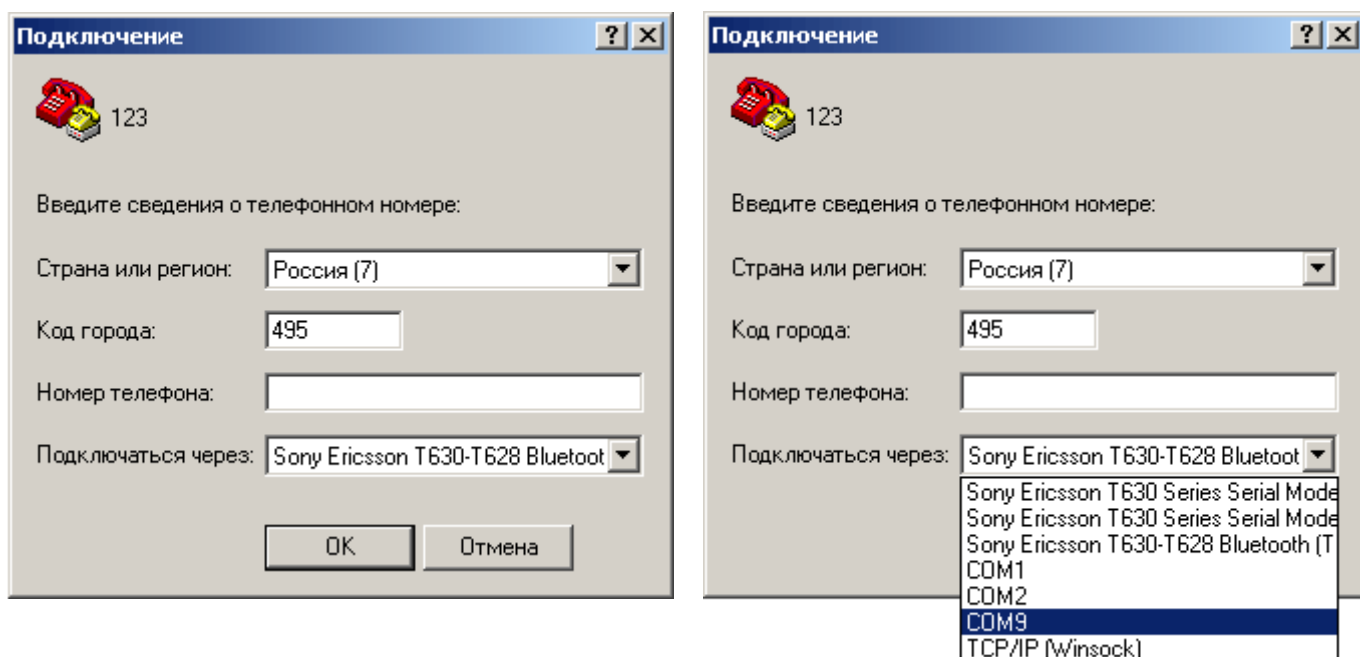
Для управления модулем в общем случае можно использовать любую терминальную программу, имеющую возможность работы с COM портами. Здесь мы рассмотрим использование программы *HyperTerminal*, входящей в состав ОС Windows и программу *KeTerm* специально предназначенную для работы с Ke-USB24A.

Начнем с *HyperTerminal* т.к. она есть в любой версии Windows и всегда под рукой. Для ее запуска выберите *Пуск -> Программы -> Стандартные -> Связь -> HyperTerminal*. Тотчас же

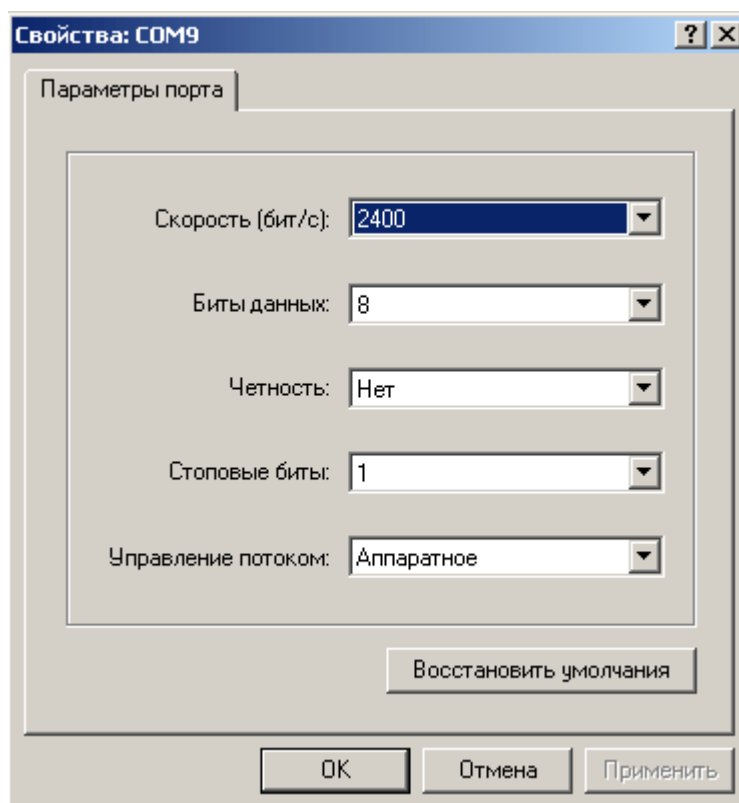
запуститься мастер создания нового соединения с предложением указать имя соединения. Укажите любое название, нажмите **"OK"**.



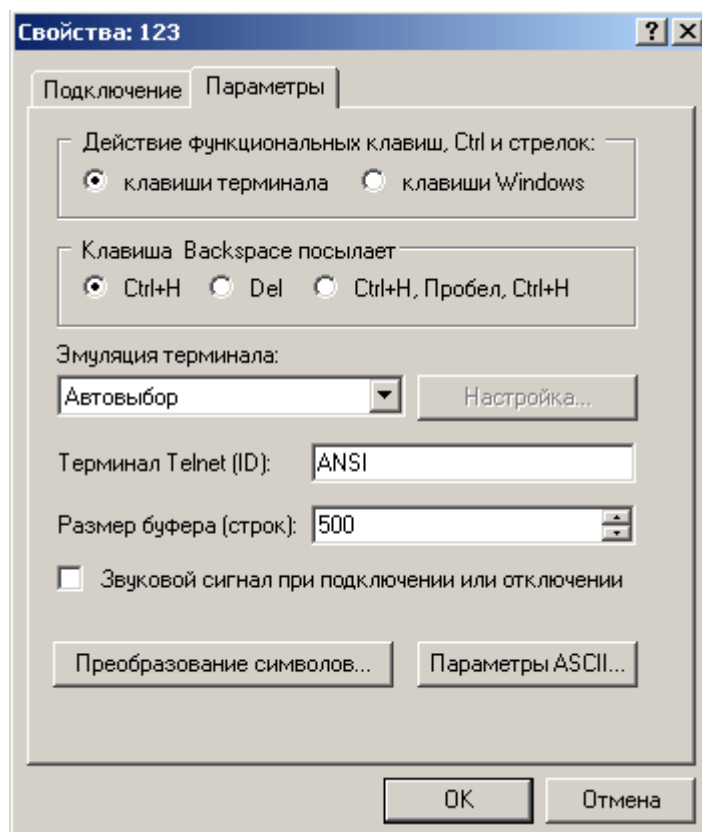
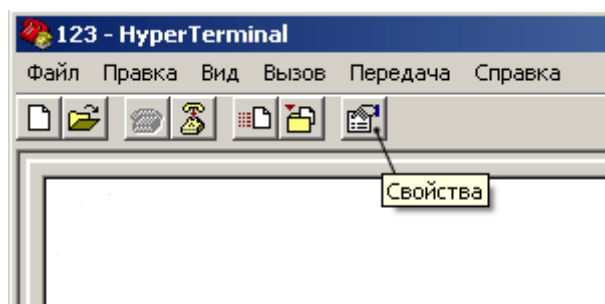
Следующим шагом необходимо выбрать COM порт через который необходимо установить соединение. В списке выбираем COM9 (именно такой номер порта был выделен Windows для модуля). Нажмите **"OK"**.



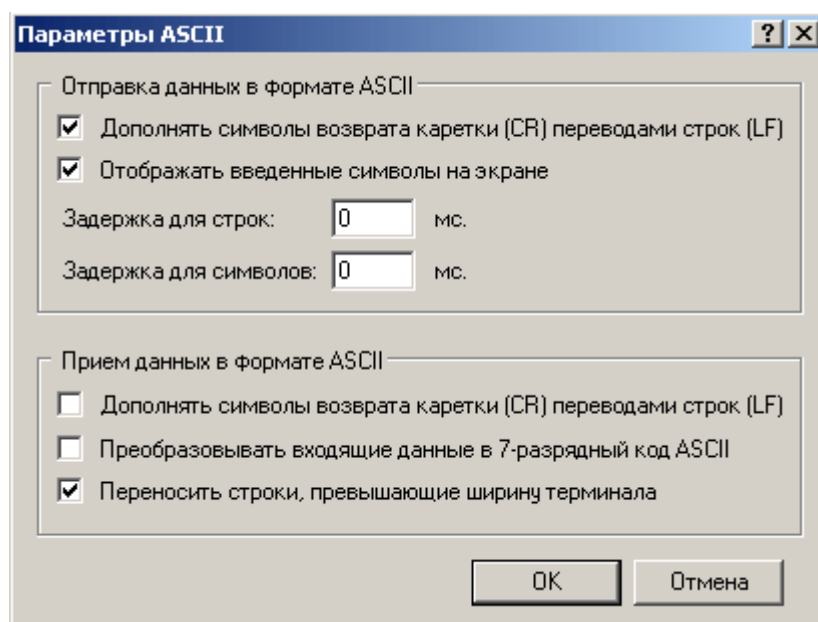
Далее мастер предложит изменить настройки порта. Здесь необходимо сделать важное замечание – такие параметры как скорость, количество битов данных, битов четности и остальные настройки не имеют ни какого значения для модуля, т.к. фактически данные передаются по шине USB. Можно смело оставить все настройки по умолчанию. Нажмите **"OK"**.



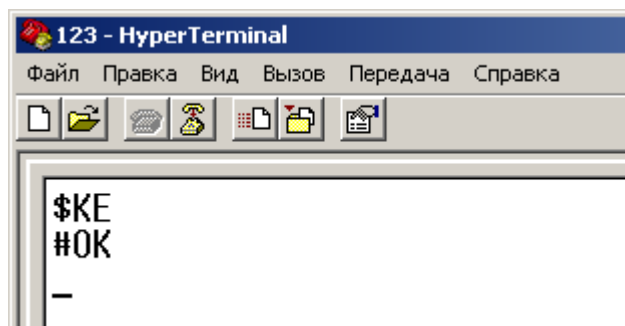
Итак, соединение установлено. Далее настоятельно рекомендуется изменить некоторые настройки в программе для удобства работы. В главной панели программы нажмите на кнопку "Свойства". В открывшемся окне перейдите на вкладку "Параметры" и нажмите на кнопку "Параметры ASCII...".



Установите флажки как на рисунке ниже. Обратите внимание на первый флажок. Синтаксис команд модуля требует чтобы все команды заканчивались символом возврата каретки и перевода на новую строку. Нажмите "OK".



Все настройки выполнены. Можно приступать к работе. Для начала отправим модулю самую простую команду **\$KE** – команду проверки. Набираем текст **\$KE** и нажимаем Enter. В ответ мы получаем строку **#OK** – это значит что модуль исправен и готов к работе.

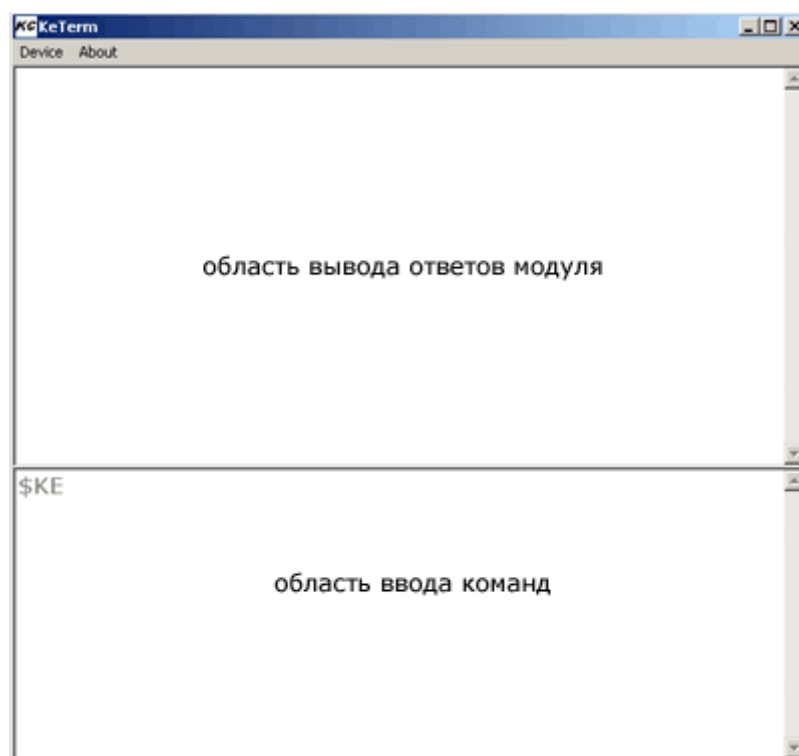


Давайте используем еще одну простую команду – команду получения серийного номера модуля. Каждый модуль имеет свой собственный уникальный серийный номер, хранящийся в энергонезависимой памяти. Набираем **\$KE,SER** и например, получаем вот такой ответ:

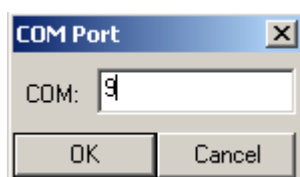
```
$KE,SER
#SER,H59K-I93L-AM89-P1NZ v1.0
```

Специально для работы с модулем Ke-USB24A была разработана небольшая программа *KeTerm*. Давайте рассмотрим как ее использовать. KeTerm можно скачать с сайта www.kernelchip.ru. Там же доступны ее исходные коды (C++, Microsoft Visual C++ 6.0).

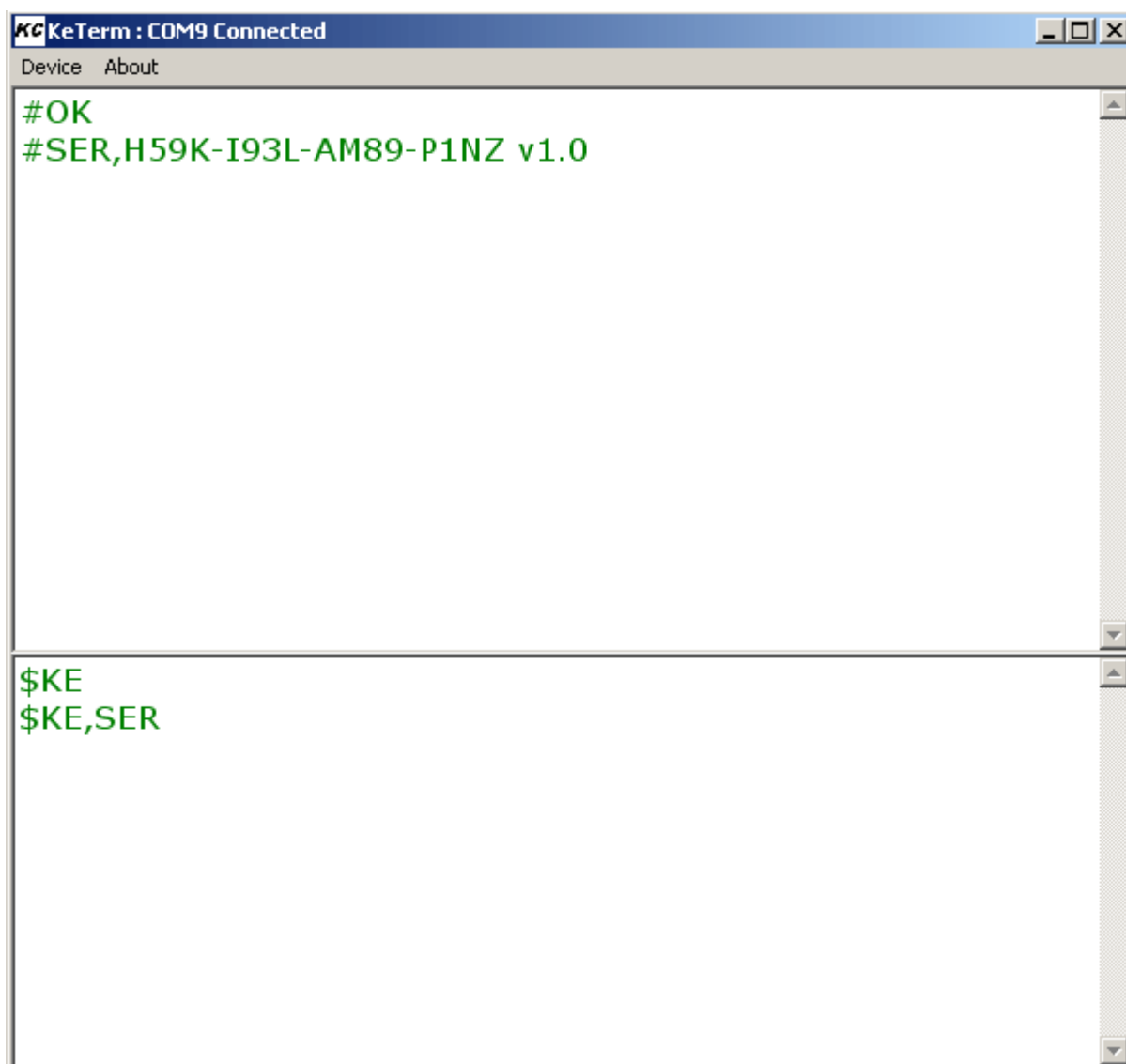
Окно программы *KeTerm* разделено на две части – нижняя предназначена для ввода команд модулю, верхняя – для отображения ответов модуля.



После запуска программы область ввода команд неактивна. Для начала работы необходимо подключиться к модулю, указав номер COM порта к которому он подключен. Для этого выбираем в меню *Device -> Connect*. В появившемся окне указываем номер порта 9 (у Вас он может быть другим; определить можно в диспетчере устройств, см. выше) и нажимаем кнопку "OK". Если порт свободен и модуль подключен, область ввода команд будет активирована. В противном случае получим сообщение об ошибке открытия порта.



Повторяем набор команд введенный ранее:



Какую программу использовать – выбор за Вами. Однако, *KeTerm* имеет ряд преимуществ по сравнению с *HyperTerminal*. *KeTerm* не требует настроек, он сразу же готов к работе. Далее, он имеет разделенные области ввода команд и вывода ответов, позволяя при этом использовать команды введенные ранее. Одним из самых важных принципиальных различий программ является механизм отправки команд в модуль – *HyperTerminal* отправляет в порт каждый вновь введенный символ (по нажатию Enter отправляются символы возврата каретки и перевода на новую строку), *KeTerm* – всю команду целиком по нажатию клавиши Enter. Поэтому, если Вы допустили неточность в синтаксисе команды используя *HyperTerminal* – исправить команду уже нельзя. *KeTerm* же позволяет отредактировать команду до ее отправки в модуль.

Управление линиями ввода/вывода

Следующим шагом мы познакомимся с управлением линиями ввода/вывода модуля Ke-USB24A. Модуль имеет 24 линий ввода/вывода на которых мы можем программно устанавливать высокий или низкий уровень напряжения если линия настроена на выход и считывать входную информацию, если линия настроена на вход. Любая из линий может быть сконфигурирована на вход или выход с возможностью сохранения настройки в энергонезависимой памяти.

По умолчанию, все линии ввода/вывода настроены на выход. Для установки направления передачи данных линии предназначена команда **\$KE,IO,SET**, для получения **\$KE,IO,GET** (см. описание команд модуля Ke-USB24A). Если линия настроена на выход, для записи данных на нее предназначена команда **\$KE,WR**. Если на вход, с помощью команды **\$KE,RD** можно производить чтение данных.

Управление выходными линиями

По умолчанию, все линии ввода/вывода настроены на выход, поэтому соберем схему, в которой ко всем линиям ввода/вывода подключена некая нагрузка, например светодиод (если Вы ранее меняли направления линий, установите их на выход командой **\$KE,IO,SET**, или подайте команду **\$KE,RST**, которая вернет все настройки модуля в значение по умолчанию). Подавая команду записи на ту или иную линию мы сможем зажигать/тушить светодиоды. Соберите схему как показано на рисунке ниже. Как видно, задействованы все 24 линии ввода/вывода модуля.

Для записи данных используем команду **\$KE,WR** в которой используется два параметра – номер линии которой адресуется эта команда и значение для записи (0 или 1). Соответственно, для того чтобы зажечь все светодиоды необходимо подать команду записи для каждой линии ввода/вывода.

```
$KE,WR,1,1
#WR,OK
```

```
$KE,WR,2,1
#WR,OK
```

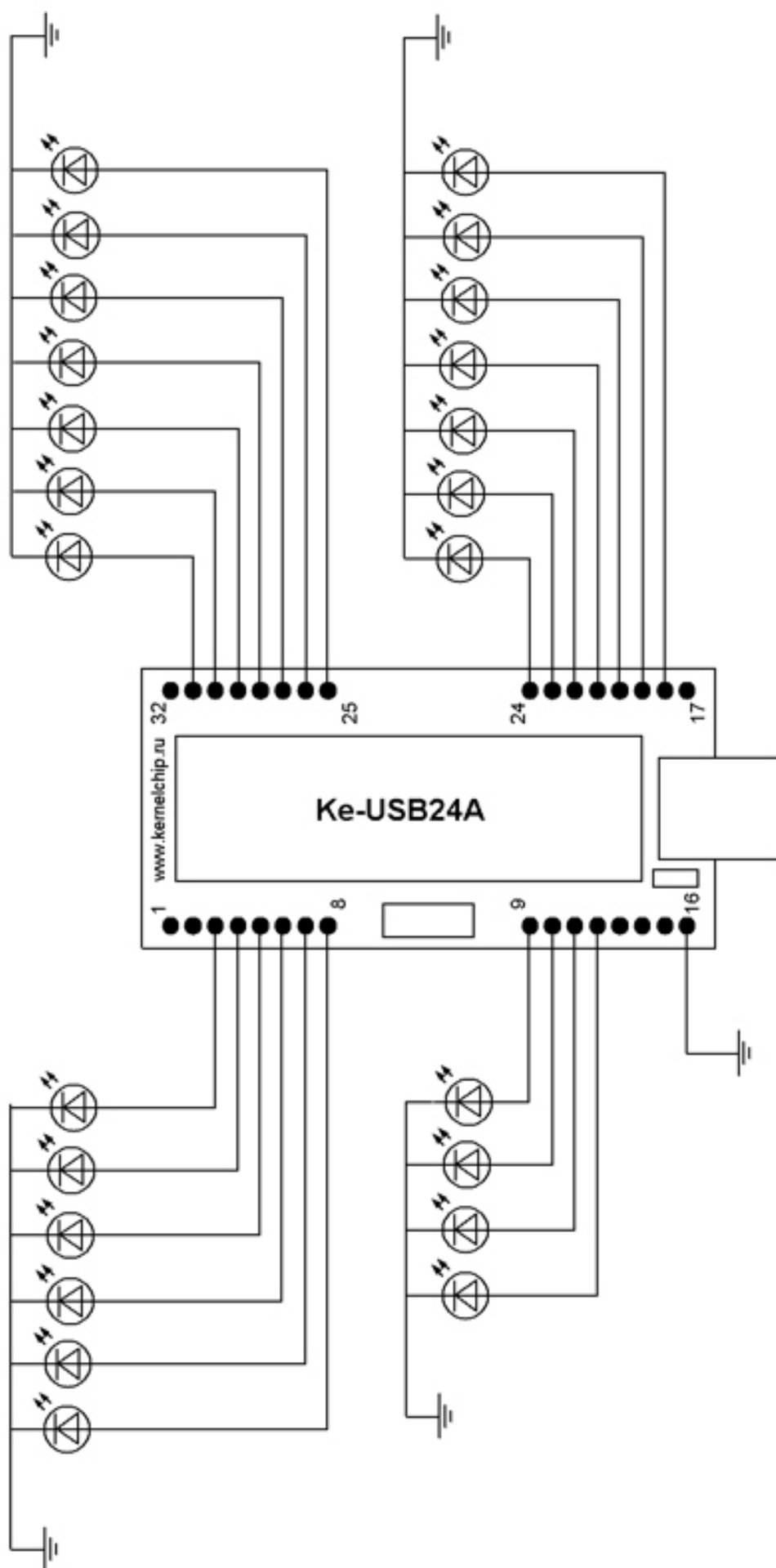
```
$KE,WR,3,1
#WR,OK
```

```
.....
```

```
$KE,WR,23,1
#WR,OK
```

```
$KE,WR,24,1
#WR,OK
```

В ответ на каждую команду записи модуль отвечает **#WR,OK** сообщая что запись успешно произведена. Если схема собрана верно, все светодиоды должны загореться.



Чтобы потушить тот или иной светодиод, необходимо установить на соответствующей линии низкий уровень напряжения, путем отправки команды записи с последним параметром равным нулю. Например, потушим светодиод, подключенный к линии ввода/вывода под номером 1 (вывод модуля 3):

```
$KE,WR,1,0  
#WR,OK
```

Если линия настроена на вход, а мы пытаемся произвести запись данных используя команду **\$KE,WR** в ответ модуль сообщит об этом, выдав ответ **#WR,WRONGLINE**.

Настройка направлений ввода/вывода

Для установки направления ввода/вывода необходимо использовать команду **\$KE,IO,SET**. В качестве параметров необходимо указать номер линии, направление линии (0 – выход, 1 - вход) и необязательный флаг (S) – сохранение настройки. Если флаг задан, то произведенная настройка линии будет сохранена в энергонезависимой памяти модуля, т.е. при следующем подключении модуля линия будет автоматически сконфигурирована в соответствии с сохраненным направлением.

Изменение направлений линий желательно производить в таком состоянии схемы, когда к линиям ввода-вывода не подключена нагрузка, т.е. желательно извлечь модуль из схемы. Поясним с чем связана подобная рекомендация. Предположим, линия была сконфигурирована на вход и на нее подан низкий уровень напряжения (фактически, подключена к земле схемы). Теперь подается команда настройки этой линии на выход, а затем команда записи данных, а именно высокого логического уровня. В результате мы получаем короткое замыкание между землей и шиной питания.

Управление входными линиями

Для чтения данных с входных линий предназначена команда **\$KE,RD** в которой в качестве параметра используется номер линии для чтения. Для того чтобы экспериментировать с чтением данных, необходимо настроить часть линий на вход. Например, пусть линии 18, 19, 20, 21, 22, 23 и 24 будут входными. Извлекаем модуль из предыдущей схемы и с помощью терминальной программы подаем следующие команды:

```
$KE,IO,SET,18,1,S  
#IO,SET,OK
```

```
$KE,IO,SET,19,1,S  
#IO,SET,OK
```

```
.....
```

\$KE,IO,SET,24,1,S
#IO,SET,OK

Теперь линии ввода/вывода с номерами 18 - 24 (7 штук) настроены на вход, причем направление сохранено в памяти, т.к. был задан флаг **S**. В этом можно убедиться, подав команду **\$KE,IO,GET**, предназначенную для чтения направлений линий (подробнее см. описание команд управления модулем Ke-USB24A).

\$KE,IO,GET,CUR
#IO,000000000000000001111111

\$KE,IO,GET,MEM
#IO,000000000000000001111111

Хорошо видно, что как и в текущей конфигурации (CUR) так и в энергонезависимой памяти (MEM) линии с номерами 1 – 17 настроены на выход, а линии 18 – 24 на вход.

После настройки линий можно приступить к модернизации схемы. Для данного примера она должна иметь вид как на рисунке ниже. Линии которые мы настроили как входные через переключатель могут быть подсоединены либо к земле схемы (логический ноль) либо к шине питания (логическая единица).

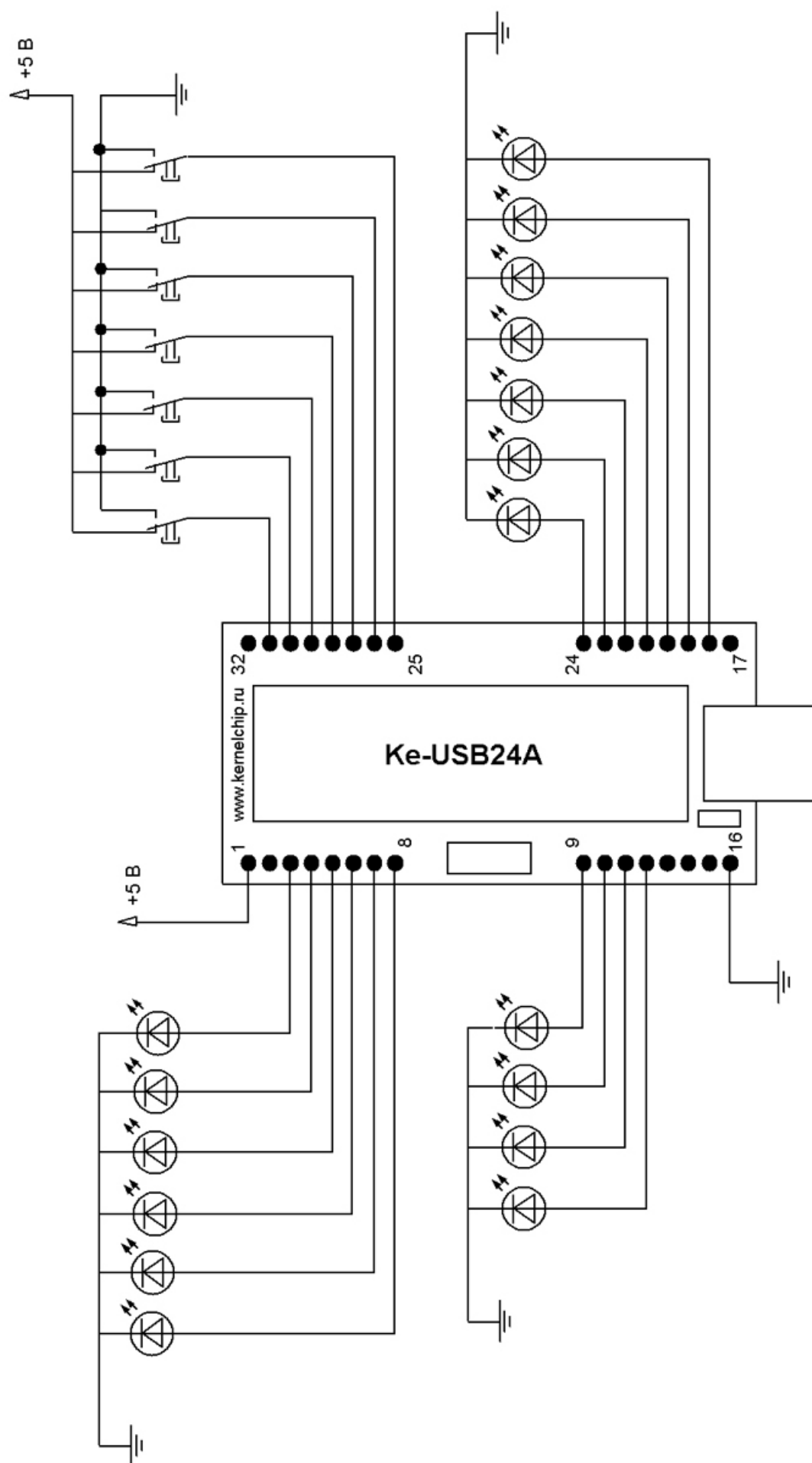
Например, соединим линию 18 с землей схемы и подадим команду чтения:

\$KE,RD,18
#RD,18,0

Получаем ответ, в котором сообщается что на линию под номером 18 подан логический ноль. Теперь подключим линию к шине питания и подадим команду еще раз:

\$KE,RD,18
#RD,18,1

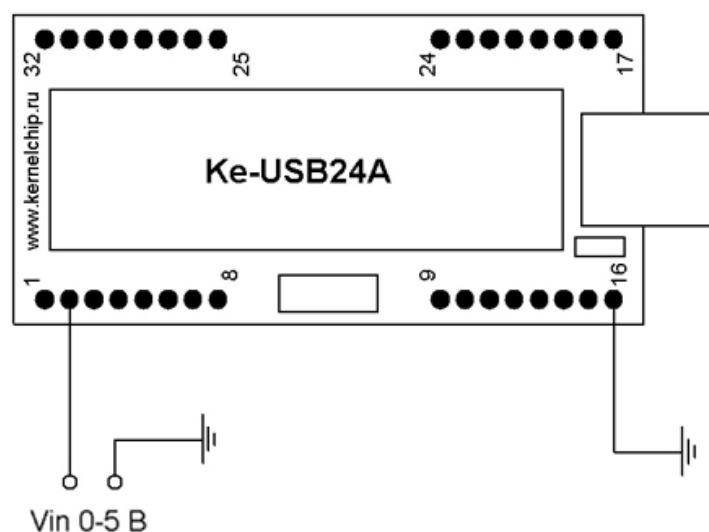
Соответственно, можно попробовать подавать команды чтения и для других линий, настроенных на вход. Если линия настроена на выход, и производится попытка чтения с нее данных, модуль выдаст ответ **#RD,WRONGLINE** сообщая нам что для данной линии в текущий момент времени операция чтения не возможна.



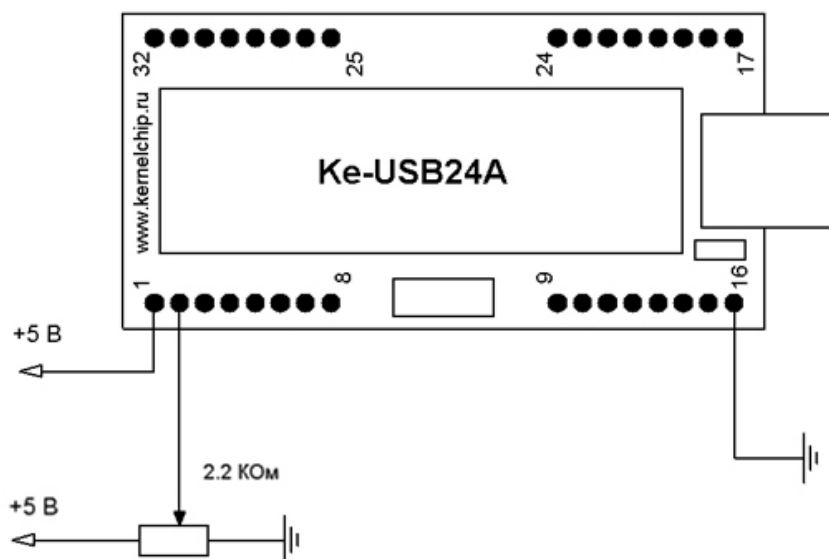
Модуль АЦП

Ke-USB24A содержит в себе встроенный 10-ти разрядный АЦП с динамическим диапазоном входного напряжения от 0 до 5 В и гарантированной частотой дискретизации до 400 Гц. Модуль АЦП может быть применен для построения цифрового USB вольтметра, USB осциллографа с изменяемой частотой дискретизации, измерения выходного напряжения различных датчиков и схем.

Рассмотрим схемотехнические особенности модуля АЦП. Для измерения напряжения входного аналогового сигнала необходимо подать это напряжение на вывод модуля Vadc (вывод номер 2) относительно земли модуля (выводы 16 или 17).



Для стартового ознакомления с возможностями АЦП модуля Ke-USB24A предлагается следующая схема, показанная на рисунке ниже. На вход АЦП будет подаваться напряжение, снимаемое с движка переменного резистора. Изменяя положение движка можем получать любое напряжение в диапазоне от 0 до 5 В.



Для единичного получения цифрового кода, соответствующего напряжению на входе АЦП необходимо подать команду \$KE,ADC. Мы получим, например, вот такой ответ:

```
$KE,ADC
#ADC,0031
```

Для получения величины напряжения входного сигнала необходимо воспользоваться следующей формулой:

$$V_{in} = \frac{Value}{1023} \cdot 5 [B]$$

где *Value* – результат работы АЦП.

Например, в данном случае на входе АЦП было установлено напряжение

$$V_{in} = \frac{31}{1023} \cdot 5 = 0.151 [B]$$

Установим движок переменного резистора в одно из крайних положений, подадим команду чтения АЦП. Затем установим движок в другое крайнее положение, еще раз прочтем АЦП. Мы получим следующие результаты. Соответственно, в первом случае на входе АЦП было нулевое напряжение, во втором 5 В.

```
$KE,ADC
#ADC,0000
```

```
$KE,ADC
#ADC,1023
```

Ke-USB24A позволяет настроить АЦП в режим автоматической выдачи результатов оцифровки входного напряжения с определенной частотой. Например, если подать команду **\$KE,ADC,100** – модуль будет с частотой 100 Гц выдавать ответ в виде **#ADC,Value**. Такой режим крайне удобен для построения USB компьютерного осциллографа, поскольку все вопросы временной синхронизации уже решены модулем, остается только указать необходимую частоту дискретизации и считывать результаты работы АЦП. Для остановки автоматического опроса АЦП необходимо подать команду **\$KE,ADC,0**.

Сохранение пользовательских данных в памяти модуля

Модуль Ke-USB24A позволяет сохранить в своей энергонезависимой памяти до 32 байт произвольной информации пользователя. Запись в энергонезависимую память подразумевает что при отключении питания сохраненные данные не теряются.

Например, сохраним в памяти модуля строку данных *My Data for storage*:

```
$KE,UD,SET,My Data for storage  
#UD,SET,OK
```

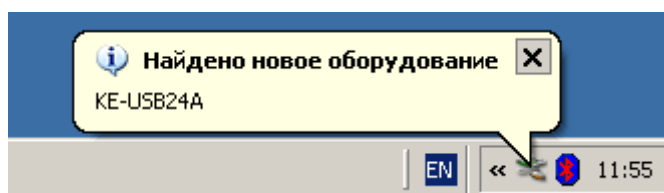
Теперь попробуем отключить питание модуля (желательно при этом сначала закрыть терминальную программу), включить его снова и прочесть ранее сохраненные данные:

```
$KE,UD,GET  
#UD,My Data for storage
```

Подобная возможность сохранения данных может быть использована, например, для ведения счетчика подключений модуля, для сохранения специфичных настроек.

Управление строковым дескриптором USB

Информация содержащаяся в строковом дескрипторе USB устройства отображается в виде сообщения операционной системы при подключении модуля к шине USB компьютера. По умолчанию используется строка *KE-USB24A*.

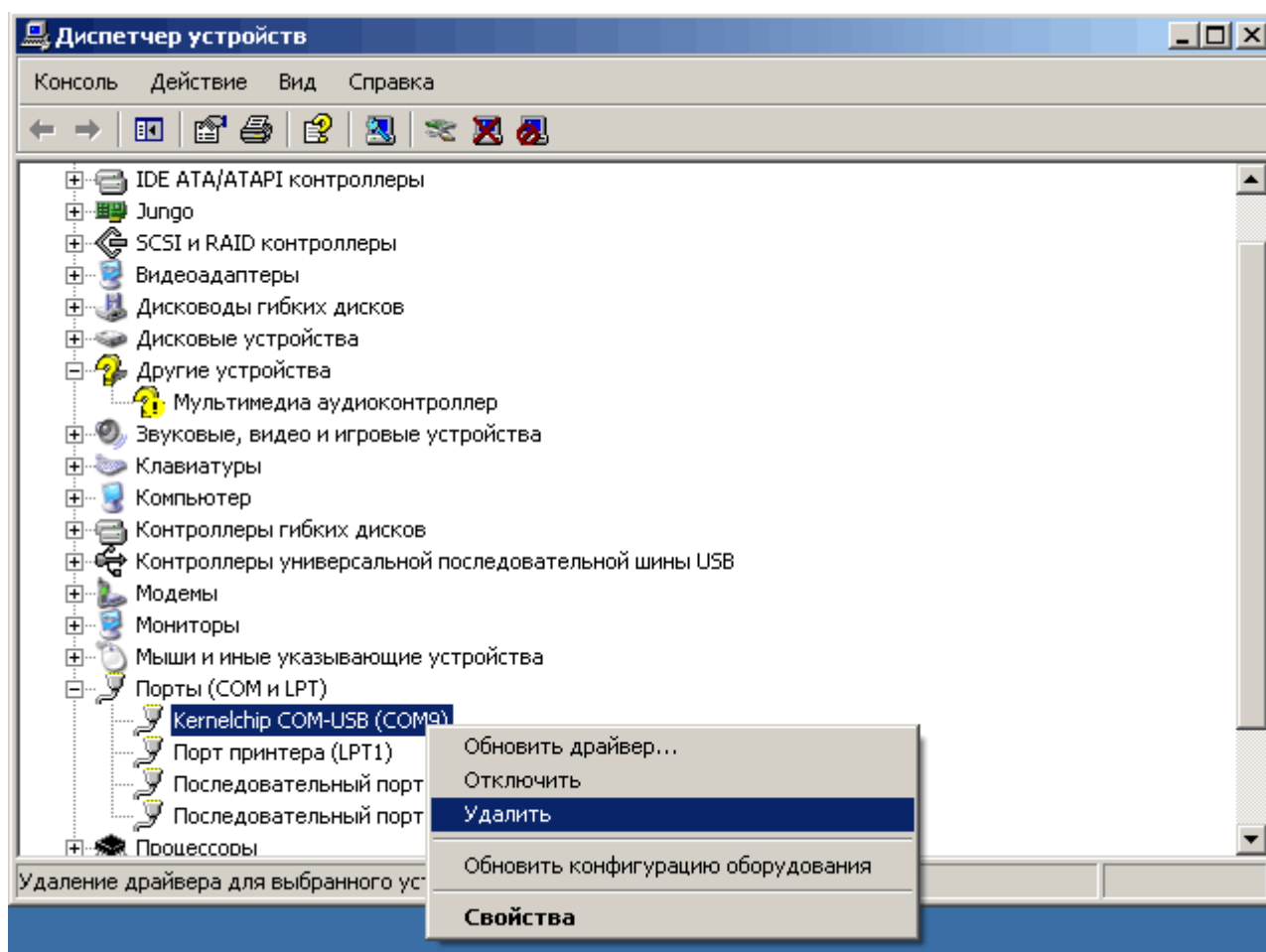


Однако, модуль позволяет изменять эту строку и устанавливать любую другую по усмотрению пользователя! Подобные возможности на данный момент не предоставляет ни один другой производитель интерфейсных модулей. Длина строкового дескриптора ограничена по длине 32 символами.

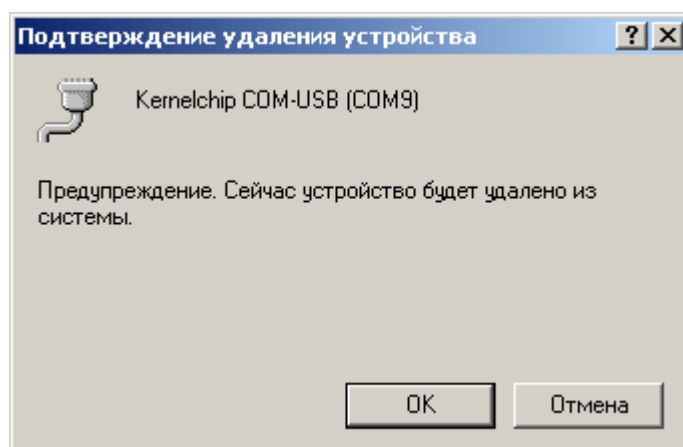
Давайте для примера изменим строковый дескриптор USB устройства на строку вида *My own Usb device*. Для этого в терминале подаем соответствующую команду:

```
$KE,USB,SET,My own USB device  
#USB,SET,OK
```

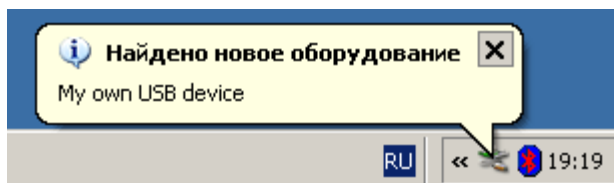

Далее, нам необходимо удалить устройство Ke-USB24A из состава системы и запустить заново установку устройства. Для этого, сначала, необходимо закрыть программы, использующие порт модуля, иначе система не позволит удалить устройство. Затем открываем Диспетчер устройств, находим виртуальный порт модуля и в выпадающем меню выбираем пункт **Удалить**.



Появится сообщение с предупреждением. Нажимаем **"ОК"**. При этом устройство исчезнет из диспетчера устройств и мы услышим характерный звуковой сигнал отключающегося USB устройства.



Теперь необходимо физически отключить модуль от шины USB и подключать вновь. Windows обнаружит новое устройство и отобразит следующее сообщение:

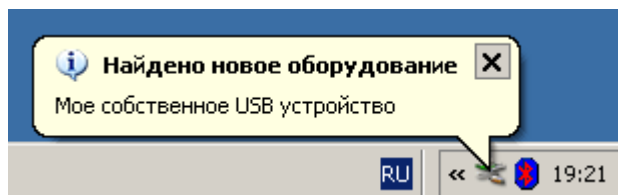


Вновь потребуется установить драйвера для модуля. Эта процедура описана в соответствующем документе, доступном на сайте.

Следует обратить внимание, что Ke-USB24A позволяет использовать для строкового дескриптора помимо латиницы также и кириллицу. Преобразование кириллических символов в UNICODE символы автоматически производится модулем в момент обработки команды. Например, если подать команду следующего вида:

\$KE,USB,SET,Мое собственное USB устройство
#USB,SET,OK

и произвести всю описанную ранее последовательность действий по удалению старого устройства и его повторного переподключения мы увидим следующее сообщение Windows:



KERNELCHIP

Компоненты для управления и мониторинга

Россия, Москва
+7 917 516 99 51

Mail: port@kernelchip.ru
Web: <http://www.kernelchip.ru>

