

Ethernet модуль Jerome

ТСР/IP команды управления



Версия 1.03

23 декабря 2011

История документа:

Данная редакция документа соответствует версии внутреннего программного обеспечения модуля (“прошивки”) Jm04 и выше.

Версия	Дата	Описание
1.03	23-12-11	Добавлена секция команд управления для последовательного порта модуля (USART)
1.02	01-06-2011	Добавлены следующие новые команды: <ul style="list-style-type: none">• \$KE,PSW• \$KE,MAC
1.01	11-04-2011	Исходная версия документа

Содержание

Введение.....	4
Команда \$KE.....	5
Команда \$KE,WR.....	5
Команда \$KE,WRA.....	6
Команда \$KE,RD.....	7
Команда \$KE,RID.....	8
Команда \$KE,ADC.....	9
Команда \$KE,IMPL.....	10
Группа Команд \$KE,PWM.....	11
Команда \$KE,PWM,SET.....	11
Команда \$KE,PWM,GET.....	12
Группа Команд \$KE,IO.....	12
Команда \$KE,IO,SET.....	12
Команда \$KE,IO,GET.....	13
Команда \$KE,DAT.....	14
Команда \$KE,EVT.....	15
Группа Команд \$KE,PSW.....	16
Команда \$KE,PSW,SET.....	16
Команда \$KE,PSW,NEW.....	16
Группа Команд \$KE,IP.....	17
Команда \$KE,IP,SET.....	17
Команда \$KE,IP,GET.....	18
Группа Команд \$KE,MAC.....	18
Команда \$KE,MAC,SET.....	18
Команда \$KE,MAC,GET.....	19
Команда \$KE,INF.....	19
Команда \$KE,RST.....	20
Команда \$KE,DEFAULT.....	20
Приложение: KE команды поддерживаемые USART.....	21
Команда \$KE.....	21
Команда \$KE,DEFAULT.....	21

Введение

Для управления модулем Jerome предназначен набор команд в текстовом формате, называемых KE командами. Для управления модулем с помощью KE-команд необходимо установить сетевое соединение с адресом 192.168.0.101 (по умолчанию) по порту 2424. После успешного установления соединения можно отправлять управляющие команды и получать ответы модуля.

В качестве программного обеспечения можно использовать любую терминальную программу позволяющую устанавливать сетевое соединение по протоколу TCP/IP, например программу *HyperTerminal*, по умолчанию входящую в состав ОС Windows.

Для защиты модуля от несанкционированного доступа в нем реализована система контроля доступа с помощью пароля. Модуль не выполняет команды управления до тех пор, пока не будет введен пароль (можно провести аналогию с консолью ОС Linux).

Любая KE команда, отсылаемая модулю, должна начинаться с символов '\$KE'. Также все команды должны заканчиваться символом возврата каретки <CR> и символом перехода на новую строку <LF> (в шестнадцатеричном формате эти символы имеют коды 0x0D и 0x0A соответственно). Ответы модуля на команды, а также отдельные информационные блоки выдаваемые модулем всегда начинаются с символа '#' (шестнадцатеричный код 0x23) и заканчиваются символами возврата каретки <CR> и перехода на новую строку <LF>.

Далее по тексту документа символы <CR><LF>, которыми должна заканчиваться любая команда модулю и любой ответ выдаваемый модулем, опускаются.

В том случае, если, синтаксис команды, отправленной модулю не является верным, модуль выдает сообщение об ошибке:

#ERR

Команда \$KE

Синтаксис: \$KE

Команда проверки работоспособности модуля. Это простая тестовая команда, на которую модуль должен ответить '#OK'.

Ответ на запрос:

#OK

Пример:

Тестовая проверка модуля:

запрос: \$KE
ответ: #OK

Команда \$KE,WR

Синтаксис: \$KE,WR,<LineNumber>,<Value>

С помощью данной команды можно установить высокий ($Value = 1$) или низкий уровень напряжения ($Value = 0$) на линии ввода/вывода модуля под номером *LineNumber*. Линия *LineNumber* должна быть настроена на выход (по умолчанию линии ввода/вывода настроены на выход. Изменение направления линии производится с помощью команды \$KE,IO,SET).

Параметры:

LineNumber – номер линии ввода/вывода. Может быть в пределах от 1 до 22 включительно.

Value – значение для записи на линию. 1 – высокий уровень напряжения (+3.3 В), 0 – низкий уровень напряжения (0 В).

Ответ на запрос:

#WR,OK – значение успешно установлено.

#WR,WRONGLINE – команда сформирована верно, но линия *LineNumber* сконфигурирована на вход, и значение не может быть установлено.

Пример:

Установим высокий уровень напряжения на линии ввода/вывода под номером 16:

запрос: \$KE,WR,16,1
ответ: #WR,OK

Команда \$KE,WRA

Синтаксис: \$KE,WRA,<ArrayOfValues>

Команда \$KE,WRA позволяет за одно обращение установить высокий или низкий уровень напряжения на всех линиях модуля, настроенных на выход.

Параметры:

ArrayOfValues – строка длиной от 1 до 22 символов. Может содержать символы '0' или '1'. Нумерация символов в строке производится слева на право. Значение первого символа строки будет установлено на линии вывода номер 1, значение второго символа - на линии номер 2 и т.д. Если соответствующая линия настроена на вход, то операция записи производится не будет. Строка может содержать меньшее число символов, чем суммарное число линий модуля, например, строка из 4-х символов позволит установить первые четыре линии модуля.

Ответ на запрос:

#WRA,OK,<UpdCount> – где *UpdCount* содержит количество успешно записанных значений.

Пример 1:

Считаем, что все линии модуля настроены на выход. Установим на всех линиях логическую единицу:

запрос: \$KE,WRA,11111111111111111111
ответ: #WRA,OK,22

Пример 2:

Считаем, что все линии модуля настроены на выход кроме линий номер 2, 3 и 10 которые соответственно настроены на вход. Установим на всех линиях логическую единицу:

запрос: \$KE,WRA,11111111111111111111
ответ: #WRA,OK,19

Входные линии действием этой команды затронуты не будут. При этом значения на всех выходных линиях будут обновлены. Как видно, ответ содержит информацию о том, что были успешно обновлены только 19 линий.

Пример 3:

Считаем, что все линии модуля настроены на выход. Установим на первых 8-ми линиях модуля логический ноль:

запрос: \$KE,WRA,00000000
ответ: #WRA,OK,8

Команда \$KE,RD

Синтаксис (Вариант 1): \$KE,RD,<LineNumber>

С помощью данной команды можно считать информацию с линии ввода/вывода *LineNumber*. Линия должна быть настроена на вход.

Параметры:

LineNumber – номер линии ввода/вывода. Может быть в пределах от 1 до 22 включительно.

Ответ на запрос:

#RD,<LineNumber>,<Value> – чтение линии *LineNumber* произведено успешно, результат *Value*. *Value = 0* – на вход линии установлен низкий уровень напряжения, *Value = 1* – соответственно, высокий уровень напряжения.

#RD,WRONGLINE – команда сформирована верно, но линия *LineNumber* сконфигурирована на выход, и значение не может быть считано.

Пример:

Считать информацию с линии 2:

запрос: \$KE,RD,2
ответ: #RD,02,1

Синтаксис (Вариант 2): \$KE,RD,ALL

По данной команде модуль произведет последовательный перебор всех линий ввода/вывода. Если линия настроена на вход будет произведено считывание информации с этой линии. Если линия настроена на выход, считывание производиться не будет. Результат выводится в виде сводной строки данных, состоящей из 22 символов (по суммарному числу линий ввода/вывода). Нумерация позиции символа в строке осуществляется слева на право и соответствует номеру линии. Если линия под номером *N* настроена на вход, то символ в сводной строке под номером *N* будет содержать результат чтения. Если линия настроена на выход – в соответствующей позиции строки будет выставлен символ 'x'.

Ответ на запрос:

#RD,<Line1 Value>< Line2 Value>< Line3 Value>.... <Line22 Value>

Пример:

Считать информацию со всех линий модуля:

запрос: \$KE,RD,ALL
ответ: #RD,xxx10xxx0xxx1xxxxxxxxxx

Данный пример показывает, что линии под номером 4, 5, 9 и 13 настроены на вход и на них установлены логические уровни 1, 0, 0, 1 соответственно. Остальные линии настроены на выход.

Команда \$KE,RID

Синтаксис (Вариант 1): \$KE,RID,<LineNumber>

С помощью данной команды можно считать информацию с линии ввода/вывода *LineNumber*. Линия может быть настроена как на вход, так и на выход. Если линия настроена на вход, то команда возвращает значение логического уровня (0 или 1) установленного на линии “снаружи” модуля, т.е. действие команды полностью аналогично команде \$KE,RD. Если линия настроена на выход, то команда вернет последнее значение, которое было установлено для этой линии с помощью команд \$KE,WR или \$KE,WRA.

Параметры:

LineNumber – номер линии ввода/вывода. Может быть в пределах от 1 до 22 включительно.

Ответ на запрос:

#RID,<LineNumber>,<Value> – чтение линии *LineNumber* произведено успешно, результат *Value*. *Value* = 0 – на линии установлен низкий логический уровень, *Value* = 1 – соответственно, высокий логический уровень.

Пример 1:

Считаем значение с выходной линии модуля, предварительно записав на нее различные значения:

```
запрос: $KE,WR,5,1
ответ:  #WR,OK
запрос: $KE,RID,5
ответ:  #RID,05,1
```

```
запрос: $KE,WR,5,0
ответ:  #WR,OK
запрос: $KE,RID,5
ответ:  #RID,05,0
```

Пример 2:

Считать информацию с входной линии номер 2:

```
запрос: $KE,RID,2
ответ:  #RID,02,1
```

Синтаксис (Вариант 2): \$KE,RID,<Type>

С помощью данной команды можно считать информацию со всех линий (как ввода, так и вывода), только с входных линий или только с выходных. Выбор линий осуществляется параметром *Type*.

Параметры:

Type – выбор линий для чтения. ALL – чтение всех линий, IN – чтение только входных линий, OUT – только выходных.

Ответ на запрос:

#RID,<Type>,<Line1 Value>< Line2 Value>< Line3 Value>.... <Line22 Value>

Ответ за запрос содержит информацию по всем 22 линиям в виде сводной строки данных. Нумерация в строке производится слева на право. Первому символу в строке соответствует линия номер 1, второму символу линия номер 2 и т.д. *Line Value = 0* – на линии установлен низкий логический уровень, *Line Value = 1* – соответственно, высокий логический уровень. Если направление линии не соответствует запрошенному, соответствующий символ в ответной сводной строке будет установлен в значение 'x'.

Пример:

Считать информацию со всех линий модуля:

```
запрос:  $KE,RID,ALL
ответ:   #RID,ALL,0001011100111110011111

запрос:  $KE,RID,IN
ответ:   #RID,IN,xxx10xxx0xxx1xxxxx1111

запрос:  $KE,RID,OUT
ответ:   #RID,OUT,000xx111x011x11001xxxx
```

Данный пример показывает, что линии под номером 4, 5, 9, 13, 19-22 настроены на вход и на них установлены логические уровни 1, 0, 0, 1, 1, 1, 1, 1 соответственно. Остальные линии настроены на выход.

Команда \$KE,ADC

Синтаксис (Вариант 1): \$KE,ADC,<ChannelNumber>

Считывание результата АЦП с канала модуля под номером *ChannelNumber*.

Параметры:

ChannelNumber – номер канала АЦП. Может быть в пределах от 1 до 4 включительно.

Ответ на запрос:

#ADC,<ChannelNumber>,<Value> – аналоговому напряжению на входе *ChannelNumber* канала АЦП модуля соответствует число *Value*. Для получения величины напряжения входного сигнала в вольтах необходимо воспользоваться следующей формулой:

$$V_{in} = \frac{Value}{1023} \cdot 3.3 [B]$$

Пример 1:

Получить результат преобразования АЦП с 3-го канала:

запрос: \$KE,ADC,3
ответ: #ADC,3,0645

В данном примере на входе АЦП в момент подачи запроса было напряжение:

$$V_{in} = \frac{645}{1023} \cdot 3.3 = 2.08 [B]$$

Синтаксис (Вариант 2): \$KE,ADC,ALL

Считывание результатов со всех АЦП каналов модуля за один запрос.

Параметры:

Отсутствуют.

Ответ на запрос:

#ADC,ALL,<Channel1_Value>,<Channel2_Value>,<Channel3_Value>,<Channel4_Value>

Команда \$KE,IMPL**Синтаксис (Вариант 1): \$KE,IMPL,<ImplChannel>**

Считывание значения счетчика импульсов под номером *ImplChannel*.

Ответ на запрос:

#IMPL,<SystemTime>,<ImplChannel>,<Value> – значение счетчика импульсов под номером *ImplNumber* равно *Value* в момент времени *SystemTime*.

Параметры:

SystemTime – текущее системное время модуля в секундах.

ImplChannel – номер счетчика импульсов. Может принимать значения от 1 до 4 включительно.

Value – значение счетчика импульсов.

Пример:

Получить значение счетчика импульсов под номером 3:

запрос: \$KE,IMPL,3
ответ: #IMPL,14615,3,208

Синтаксис (Вариант 2): \$KE,IMPL,ALL

Считывание значений со всех счетчиков импульсов модуля за один запрос.

Ответ на запрос:

#IMPL,ALL,<SystemTime>,<Chan1_Value>,<Chan2_Value>,<Chan3_Value>,<Chan4_Value>

Пример:

Получить значения со всех счетчиков импульсов:

запрос: \$KE,IMPL,ALL
ответ: #IMPL,14615,0,2,234,89

Синтаксис (Вариант 3): \$KE,IMPL,RST

Обнуление значения всех счетчиков импульсов.

Пример:

Произведем обнуление значений счетчиков импульсов:

запрос: \$KE,IMPL,RST
ответ: #IMPL,RST,OK

Группа Команд \$KE,PWM

Команды группы PWM (Pulse Width Modulation) предназначены для управления ШИМ выводом модуля.

Команда \$KE,PWM,SET

Синтаксис: \$KE,PWM,SET,<PowerValue>

Управление ШИМ выходом модуля. Команда задает выходную мощность ШИМ сигнала.

Параметры:

PowerValue – параметр, задающий выходную мощность сигнала на ШИМ выходе. Может принимать значения от 0 до 100. При значении равном 100 – ШИМ сигнал имеет 100% теоретическую мощность и 0% при значении равном 0.

Ответ на запрос:

#PWM,SET,OK

Пример:

Установить 60% уровень мощности ШИМ сигнала:

запрос: \$KE,PWM,SET,60

ответ: #PWM,SET,OK

Команда \$KE,PWM,GET

Синтаксис: \$KE,IO,GET

Возвращает текущее значение мощности ШИМ сигнала.

Ответ на запрос:

#PWM,<PowerValue>

Параметры:

PowerValue – выходная мощность сигнала на ШИМ выходе. Может принимать значения от 0% до 100% включительно.

Пример 1:

Получить значение мощности ШИМ сигнала на текущий момент времени:

запрос: \$KE,PWM,GET

ответ: #PWM,60

Группа Команд \$KE,IO

Команды группы IO (Input/Output) предназначены для управления режимом ввода/вывода линий модуля, позволяя настраивать линии на вход или на выход.

Команда \$KE,IO,SET

Синтаксис: \$KE,IO,SET,<LineNumber>,<IoDirection>

Команда позволяет настроить линию ввода/вывода *LineNumber* в качестве выходной (*IoDirection* = 0) или входной (*IoDirection* = 1). Настройка сохраняется в энергонезависимой памяти модуля. При последующих включениях модуля линия *LineNumber* будет автоматически сконфигурирована в ранее установленное направление.

Параметры:

LineNumber – номер линии ввода/вывода. Может быть в пределах от 1 до 22 включительно.

IoDirection – направление ввода/вывода. 1 – (input) на вход, 0 – (output) на выход.

Ответ на запрос:

#IO,SET,OK

Пример 1:

Установить линию 5 в качестве выходной:

запрос: \$KE,IO,SET,5,0

ответ: #IO,SET,OK

Пример 2:

Установить линию 3 в качестве входной

запрос: \$KE,IO,SET,3,1

ответ: #IO,SET,OK

Команда \$KE,IO,GET

Синтаксис (Вариант 1): \$KE,IO,GET,ALL

Вывод значений направления ввода/вывода для всех 22 линий модуля в виде сводной строки за один запрос.

Ответ на запрос:

#IO,ALL,<Line1 Direction><Line2 Direction>...<Line22 Direction>

Line Direction = 1 – линия настроена на вход

Line Direction = 0 – линия настроена на выход

Пример 1:

Получить значения направлений ввода/вывода линий модуля:

запрос: \$KE,IO,GET,ALL

ответ: #IO,ALL,0001000011000000000000

В данный момент времени линии с номерами 4, 9 и 10 настроены на вход, остальные на выход.

Синтаксис (Вариант 2): \$KE,IO,GET,<LineNumber>

Вывод значения направления ввода/вывода для линии *LineNumber*.

Параметры:

LineNumber – номер линии ввода/вывода. Может быть в пределах от 1 до 22 включительно.

Ответ на запрос:

#IO,<LineNumber>,<Value>

Value – направление ввода/вывода, 0 – выход, 1 – вход.

Пример 1:

Получить значение направления ввода/вывода линий модуля 13:

запрос: \$KE,IO,GET,13
ответ: #IO,13,1

Команда \$KE,DAT

Синтаксис: \$KE,DAT,<Sate>

Команда включает/выключает выдачу сводной информации по аппаратным ресурсам модуля с частотой 1 Гц. Выводится следующая информация: текущее системное время, значения всех входных дискретных линий, всех выходных линий, значения со всех каналов АЦП и значения счетчиков импульсов.

Параметры:

Sate – если равен *ON* – производится включение выдачи сводной информации, *OFF* – выдача информации соответственно выключается.

Ответ на запрос:

#DAT,OK

Пример:

Включить периодическую выдачу сводной информации по аппаратным ресурсам:

запрос: \$KE,DAT,ON
ответ: #DAT,OK
#TIME,614

```
#RID,IN,0xxxx0xxxxxx0xxxx0xxxx
#RID,OUT,x0000x000000x0000x0000
#ADC,ALL,610,529,514,606
#INT,ALL,614,29,0,0,0
```

```
#TIME,615
```

```
.....
```

Информация выводится с частотой в 1 Гц.

Команда \$KE,EVT

Синтаксис: \$KE,EVT,<Sate>

Команда включает/выключает режим автоматического отслеживания изменения состояний входных дискретных линий (система eCatcher). Если такой режим включен и на любой из входных линий происходит изменение состояния, в автоматическом режиме производится выдача информационного сообщения об обнаруженном событии.

Параметры:

Sate – если равен *ON* – режим включен, *OFF* – режим выключается.

Ответ на запрос:

```
#EVT,OK
```

Пример:

Включить режим отслеживания изменений на входных линиях:

```
запрос: $KE,EVT,ON
```

```
ответ: #EVT,OK
```

Например, в некий момент времени произошло изменение состояния входной линии под номером 4. Новое состояние – логическая единица. При этом в порт будет выдано сообщение в следующем формате:

```
#EVT,IN,<SystemTime>,<LineNumber>,<CurrentValue>
```

где

SystemTime – текущее системное время модуля в секундах.

LineNumber – номер входной дискретной линии, на которой было обнаружено событие

CurrentValue – текущее значение на входной линии

В описываемом примере ответ может быть таким:

```
#EVT,IN,567,4,1
```

Группа Команд \$KE,PSW

Команды группы PSW предназначены для установки/изменения пароля модуля, который необходим для доступа к Web-странице и командному интерфейсу. Команды этой группы доступны начиная с версии Jm02 программного обеспечения модуля (“прошивки”).

Команда \$KE,PSW,SET

Синтаксис: \$KE,PSW,SET,<Password>

С помощью команды можно ввести пароль доступа к командному интересу модуля.

Параметры:

Password – Пароль для доступа к модулю

Ответ на запрос:

#PSW,SET,OK – команда сформирована верно, пароль верный, доступ к командному интерфейсу разблокирован
 \$PSW,SET,BAD – неверный пароль. Доступ по-прежнему заблокирован

Пример:

Введем пароль доступа к модулю (по умолчанию - Jerome):

запрос: \$KE,PSW,SET,Jerome
 ответ: #PSW,SET,OK

Команда \$KE,PSW,NEW

Синтаксис: \$KE,PSW,NEW,<CurrPassword>,<NewPassword>

С помощью этой команды можно установить новый пароль, который будет использоваться для разблокировки доступа к командному интерфейсу и в качестве пароля доступа к Web-интерфейсу. Новый пароль сохраняется в энергонезависимой памяти.

Параметры:

CurrPassword – Текущий пароль доступа
NewPassword – Новый пароль, длиной не более 9 символов

Ответ на запрос:

#PSW,NEW,OK – новый пароль успешно установлен
 \$PSW,NEW,BAD – текущий пароль указан неверно

Пример:

Установить новый пароль “SimSim” (при условии, что текущий пароль соответствует паролю по умолчанию – “Jerome”):

запрос: \$KE,PSW,NEW,Jerome,SimSim
ответ: #PSW,NEW,OK

Замечание:

В том случае, если вы забыли новый пароль или произошел сбой во время его записи в энергонезависимую память (отключение питания) – единственный выход из сложившейся ситуации является аппаратный сброс настроек. Для сброса всех настроек в энергонезависимой памяти модуля в исходное значение по умолчанию необходимо использовать вывод модуля под номером 36 (RST).

Группа Команд \$KE,IP

Команды группы IP предназначены для установки/чтения IP адреса модуля.

Команда \$KE,IP,SET

Синтаксис: \$KE,IP,SET,<IpAddress>

Команда позволяет установить IP адрес модуля. По умолчанию, IP адрес модуля равен 192.168.0.101. Параметр сохраняется в энергонезависимой памяти. Изменения вступают в силу после перезагрузки модуля (команда \$KE,RST или сброс питания).

Параметры:

IpAddress – IP адрес в формате X.X.X.X (в качестве X могут быть использованы числа от 0 до 255)

Ответ на запрос:

#IP,SET,OK

Пример:

Установить IP адрес модуля равным 192.168.0.115:

запрос: \$KE,IP,SET,192.168.0.115
ответ: #IP,SET,OK

Замечание:

Будьте внимательны при изменении IP адреса модуля. Если адрес будет указан некорректно, вы не сможете подключиться к модулю через сетевое соединение. Для сброса IP адреса в исходное значение по умолчанию необходимо подать команду *\$KE,DEFAULT* через последовательный порт модуля (USART).

Команда \$KE,IP,GET

Синтаксис: *\$KE,IP,GET*

Возвращает текущий IP адрес модуля.

Ответ на запрос:

#IP,<IpAddress>

Пример:

Получить текущее значение IP адреса модуля:

запрос: *\$KE,IP,GET*
ответ: *#IP,192.168.0.115*

Группа Команд \$KE,MAC

Команды группы MAC предназначены для установки/чтения MAC адреса модуля. Команды этой группы доступны начиная с версии *Jm02* программного обеспечения модуля (“прошивки”).

Команда \$KE,MAC,SET

Синтаксис: *\$KE,MAC,SET,<MacAddress>*

Команда позволяет установить MAC адрес модуля. По умолчанию, MAC адрес модуля равен 00-04-A3-00-00-0B (в десятичном формате 0-4-163-0-0-11). Параметр сохраняется в энергонезависимой памяти. Изменения вступают в силу после перезагрузки модуля (команда *\$KE,RST* или сброс питания).

Параметры:

MacAddress – MAC адрес в формате X.X.X.X.X.X (в качестве X могут быть использованы числа от 0 до 255)

Ответ на запрос:

#MAC,SET,OK

Пример:

Установить MAC адрес модуля равным 0-4-163-0-0-15:

запрос: \$KE,MAC,SET,0.4.163.0.0.15

ответ: #MAC,SET,OK

Замечание:

Будьте внимательны при изменении MAC адреса модуля. Если адрес будет указан некорректно, вы не сможете подключиться к модулю через сетевое соединение. Для сброса MAC адреса в исходное значение по умолчанию необходимо подать команду *\$KE,DEFAULT* через последовательный порт модуля (USART).

Команда \$KE,MAC,GET

Синтаксис: \$KE,MAC,GET

Возвращает текущий MAC адрес модуля.

Ответ на запрос:

#MAC,<MacAdress>

Пример:

Получить текущее значение MAC адреса модуля:

запрос: \$KE,MAC,GET

ответ: #MAC, 0.4.163.0.0.15

Команда \$KE,INF

Синтаксис: \$KE,INF

Команда возвращает сводную информацию об имени устройства, версии программного обеспечения и серийном номере.

Ответ на запрос:

#INF,<DeviceName>,<FW Version>,<SerialNumber>

Параметры:

DeviceName – имя устройства. Установлено в значение “Jerome”.
FW Version – номер версии программного обеспечения модуля
SerialNumber – серийный номер модуля

Команда \$KE,RST

Синтаксис: \$KE,RST

Программный сброс модуля. После подачи команды модуль начинает работу как после отключения питания. Настройки в энергонезависимой памяти не стираются.

Команда \$KE,DEFAULT

Синтаксис: \$KE,DEFAULT

Программный сброс модуля с очисткой энергонезависимой памяти. После подачи команды модуль начинает работу как после отключения питания. Настройки в энергонезависимой памяти возвращаются в значение по умолчанию (заводские настройки).

Приложение: KE команды поддерживаемые USART

В составе модуля Jerome встроен последовательный порт (USART). Помимо передачи данных по сетевому интерфейсу (Ethernet-2-COM интерфейс), USART также может принимать и обрабатывать ряд текстовых команд управления. Общие правила построения команд управления для последовательного порта ни чем не отличаются от правил и принципов, описанных в начале данного документа. Скорость USART постоянна, не подлежит изменению и равна 19200 бит/с. Именно на этой скорости необходимо производить информационный обмен с последовательным портом модуля.

Команда \$KE

Синтаксис: \$KE

Команда проверки соединения с модулем по последовательному порту. Это простая тестовая команда, на которую модуль должен ответить '#OK'.

Ответ на запрос:

#OK

Пример:

Тестовая проверка соединения с модулем по последовательному порту:

запрос: \$KE
ответ: #OK

Команда \$KE,DEFAULT

Синтаксис: \$KE,DEFAULT

Программный сброс модуля с очисткой энергонезависимой памяти. После подачи команды модуль начинает работу как после отключения питания. Настройки в энергонезависимой памяти возвращаются в значение по умолчанию (заводские настройки):

- IP адрес: 192.168.0.101
- MAC адрес: 00-04-A3-00-00-0B
- Все линии ввода/вывода настроены на выход
- Режим обнаружения изменений на входных линиях отключен

Замечание:

Отправка данной команды через последовательный порт – единственный доступный способ сброса некорректно установленного IP/MAC адреса модуля.

KERNELCHIP

Компоненты для управления и мониторинга

Россия, Москва
+7 917 516 99 51

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

