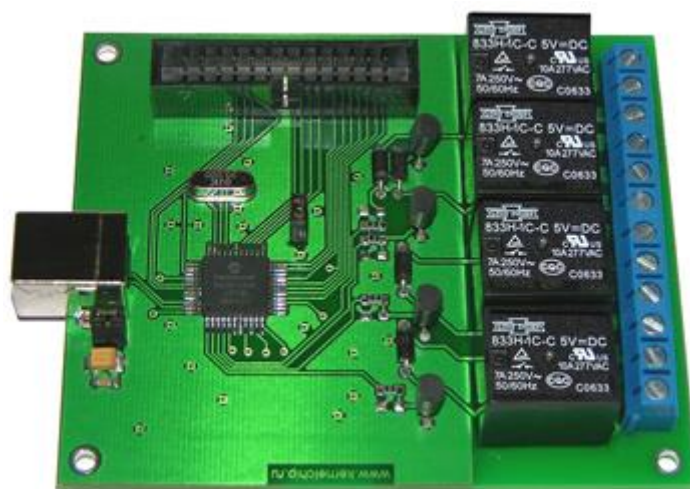


# Управление нагрузкой и контроль через USB

## Модуль MP714

Техническое описание



**История документа:**

<b>Версия</b>	<b>Дата</b>	<b>Описание</b>
1.06	20-12-2012	Стилистические коррекции документа
1.05	26-11-2010	Описание обновлено с учетом поддержки ОС Windows Vista и 7 32/64 bit
1.04	13-12-2009	Добавлены правила и условия эксплуатации модуля
1.03	19-10-2009	N/A

## Содержание

1.	Общее описание .....	4
2.	Отличительные особенности .....	4
3.	Управление модулем .....	5
3.	Функциональные характеристики .....	6
4.	Электрические характеристики .....	6
5.	Габаритные размеры .....	7
6.	Назначение выводов .....	8
7.	Правила и условия эксплуатации .....	11

## 1. Общее описание

Модуль МР714 предназначен для сопряжения внешних цифровых и аналоговых устройств, датчиков и исполнительных механизмов с компьютером через шину USB. Благодаря наличию четырех мощных реле появляется возможность управлять силовыми цепями с напряжениями до 230 В непосредственно с компьютера.

На компьютере под управлением операционной системы Windows или ОС семейства Linux модуль определяется как дополнительный (виртуальный) COM порт. Это означает, что с программной точки зрения обмен информацией между модулем и компьютером осуществляется через интерфейс обычного последовательного порта. Последнее обстоятельство существенно облегчает программирование, т.к. нет необходимости в применении дополнительных динамических библиотек, сложных интерфейсов и непосредственного общения с драйвером. Для написания собственных программ управления модулем возможно применение любого языка/среды программирования, поддерживающих возможность работы с COM портами.



Рис.1. Общий вид модуля МР714

Модуль имеет 18 дискретных линии ввода / вывода (либо лог. 0 либо лог. 1) с возможностью независимой настройки направления передачи данных (вход/выход), 4 мощных электромагнитных реле для непосредственного управления высоковольтными цепями и 4 встроенных 10-ти разрядных АЦП.

## 2. Отличительные особенности

- интерфейсный модуль для сопряжения по шине USB
- определяется ОС Windows как виртуальный COM порт
- не требует дополнительных схемных элементов. Сразу готов к работе
- питание от шины USB
- 18 дискретных линий ввода/вывода с возможностью независимой настройки направления передачи данных (вход/выход) и сохранения настроек в энергонезависимой памяти модуля

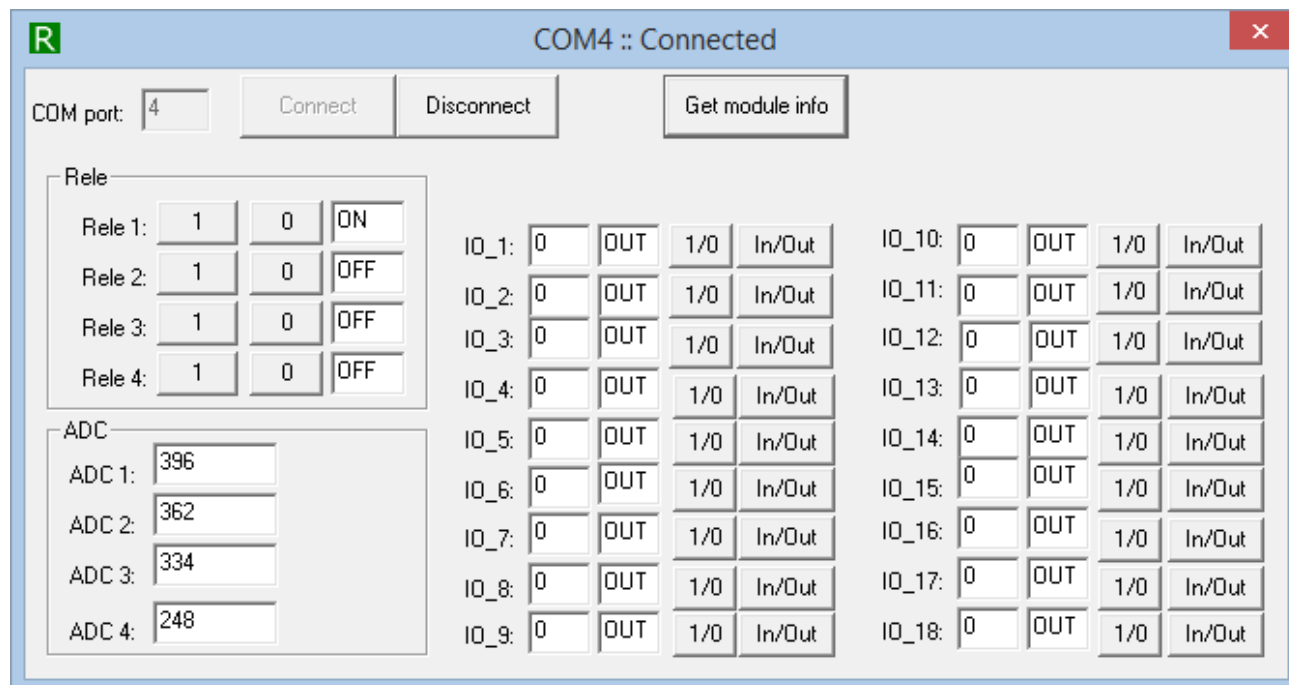
- 4 реле для управления высоковольтными цепями и нагрузками
- 4 встроенных 10-ти разрядных АЦП
- динамический диапазон напряжения входного аналогового сигнала для АЦП от 0 до 5 В
- набор готовых текстовых команд управления высокого уровня (КЕ - команды)
- открытый командный интерфейс
- возможность сохранения данных пользователя в энергонезависимой памяти модуля (до 32 байт)
- возможность изменения строкового дескриптора USB устройства
- каждый модуль имеет уникальный серийный номер доступный программно
- поддержка ОС Windows 2000, XP 32/64 bit, Vista 32/64 bit и Windows 7 32/64 bit
- поддержка OS Linux

### 3. Управление модулем

Для наглядного управления возможностями модуля on-line управление модулем разработана программа *Boxer24*. С ее помощью можно:

- управлять включением и выключением реле;
- наблюдать за значениями напряжения на входах АЦП;
- менять направление линии входа/выхода;
- наблюдать за состоянием линий ввода;
- менять состояние линии выхода.

Предварительно модуль надо подключить к программе, указав номер COM-порта, который назначила система. Номер порта можно узнать через Диспетчер устройств Windows



Существует еще один способ управления модулем, с помощью набора высокоуровневых текстовых команд управления (КЕ - команды). Формируемая команда отправляется в порт, процессор модуля декодирует ее, выполняет необходимую операцию и отправляет обратно ответ в текстовом формате о статусе выполненной задачи или другую необходимую информацию, специфичную для конкретной команды. Применение текстовых команд позволяет в общем случае обойтись без разработки дополнительного программного обеспечения. Достаточно использовать любую терминальную программу позволяющую передавать данные через COM порт, например *HyperTerminal* входящую в состав ОС Windows XP. Возможно, более удобной программой

окажется *KeTerm* которая была специально разработана для работы с модулем (ее можно найти на сайте [www.masterkit.ru](http://www.masterkit.ru) в карточке товара).

KE-команды можно использовать в собственных разработках программ управления модулем на любом языке, который поддерживает передачу данных через последовательный СОМ-порт.

#### 4. Функциональные характеристики

Общее количество выводов на IDC разъеме модуля .....	26
Количество линий ввода/вывода .....	18
Число каналов АЦП .....	4
Число реле .....	4
Разрядность АЦП .....	10 бит

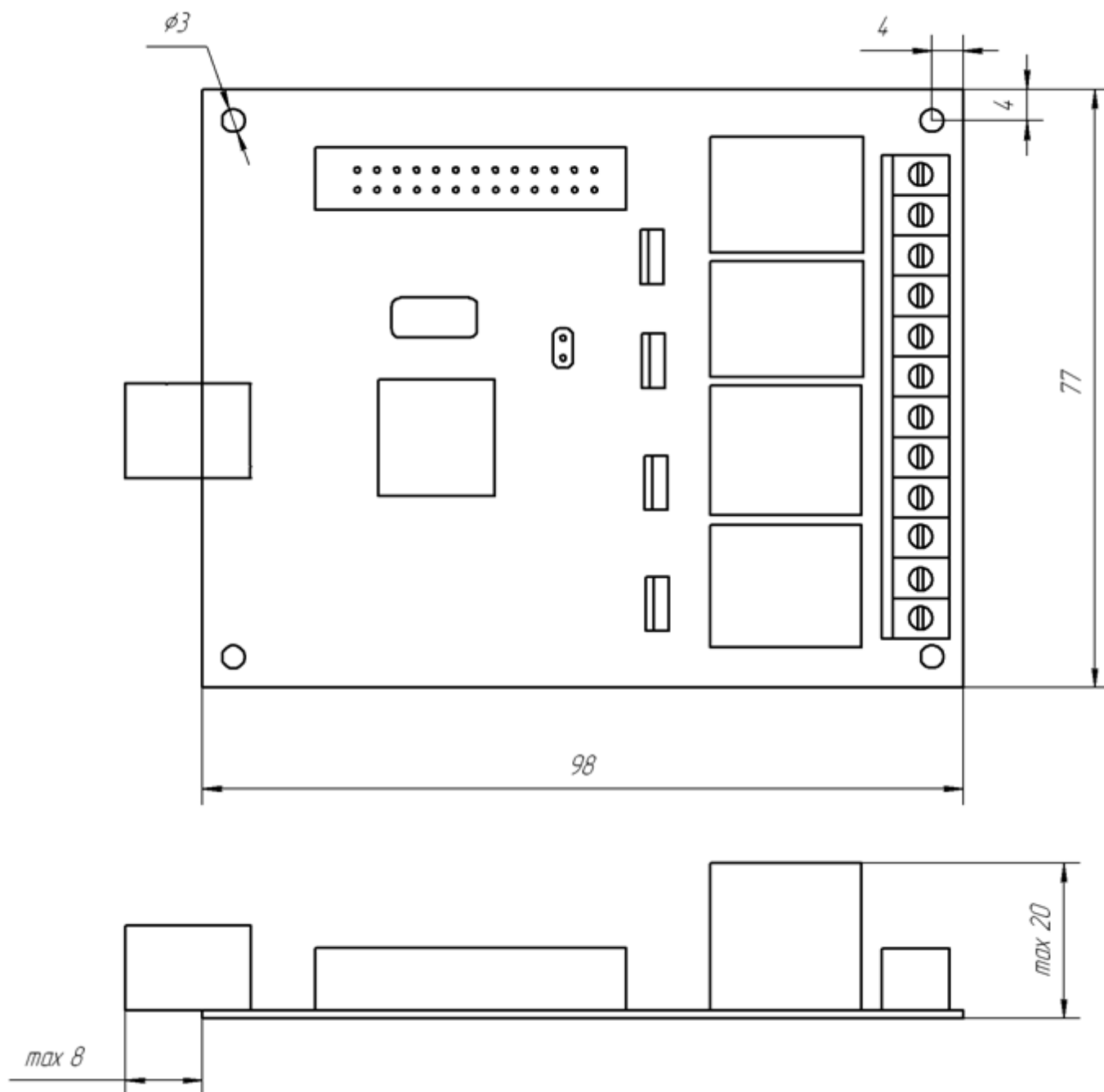
#### 5. Электрические характеристики

Напряжение питания модуля .....	5 В
Низкий уровень напряжения на линии ввода/вывода .....	$\leq 0.3$ В
Высокий уровень напряжения на линии ввода/вывода .....	$\geq 4$ В
Максимальный ток нагрузки для одной линии ввода/вывода .....	25 мА
Максимальный суммарный ток нагрузки для всех линии ввода/вывода .....	200 мА
Диапазон напряжения входного сигнала для АЦП .....	0 – 5 В
Реле: максимальное коммутируемое постоянное напряжение .....	48 В
Реле: максимальный коммутируемый постоянный ток .....	7 А
Реле: максимальное коммутируемое переменное напряжение .....	230 В

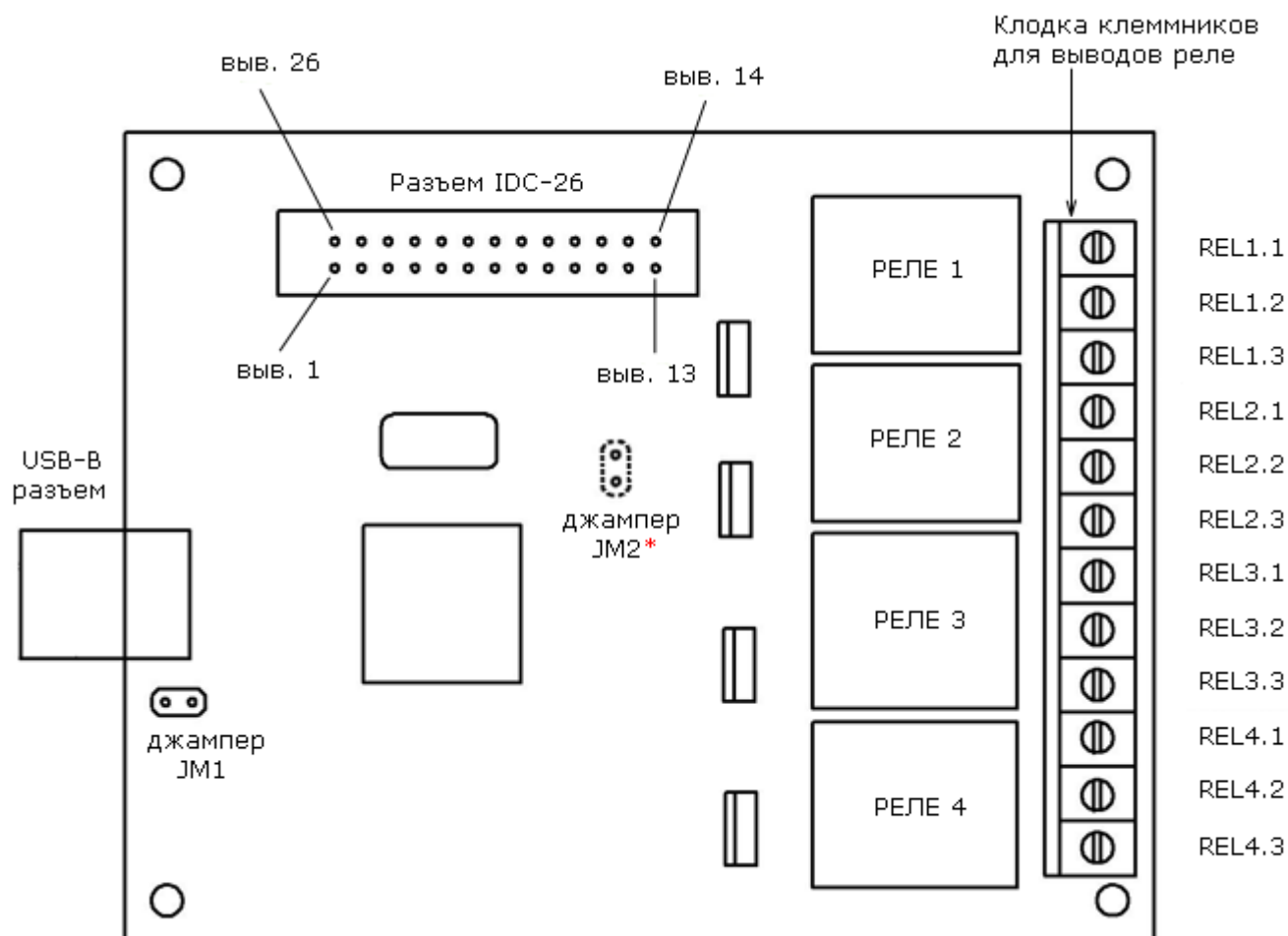
Реле: максимальный  
коммутируемый переменный  
ток

7 А

## 6. Габаритные размеры



## 7. Назначение выводов



Примечание:

\* - джампер JM1 и JM2 используется в ходе тестирования модуля на производстве. В обычном режиме работы модуля оба джампера должны быть установлены. На некоторых модулях они может отсутствовать. В этом случае на их место впаивается перемычка.

Таблица 1. Описание выводов колодки разъема IDC-26 модуля **MP714**

Номер вывода	Обозначение	Описание вывода
1	+ 5 В	Напряжение +5 В от шины USB
2	I/O1	Линия ввода/вывода 1
3	I/O2	Линия ввода/вывода 2
4	I/O3	Линия ввода/вывода 3
5	I/O4	Линия ввода/вывода 4
6	I/O5	Линия ввода/вывода 5
7	I/O6	Линия ввода/вывода 6
8	I/O7	Линия ввода/вывода 7
9	ADC4	Аналоговый вход 4-го канала АЦП
10	ADC3	Аналоговый вход 3-го канала АЦП
11	ADC1	Аналоговый вход 1-го канала АЦП



12	I/O8	Линия ввода/вывода 8
13	I/O9	Линия ввода/вывода 9
14	I/O10	Линия ввода/вывода 10
15	NC	Вывод не используется (Not Connected)
16	ADC2	Аналоговый вход 2-го канала АЦП
17	I/O11	Линия ввода/вывода 11
18	I/O12	Линия ввода/вывода 12
19	I/O13	Линия ввода/вывода 13
20	I/O14	Линия ввода/вывода 14
21	I/O15	Линия ввода/вывода 15
22	I/O16	Линия ввода/вывода 16
23	I/O17	Линия ввода/вывода 17
24	I/O18	Линия ввода/вывода 18
25	GND	Земля
26	GND	Земля

Модуль МР714 имеет в своем составе 4 электромагнитных реле. Каждое реле коммутирует три контакта. В исходном состоянии модуля контакты 1, 2 каждого реле замкнуты между собой; 2, 3 разомкнуты. С помощью команд управления можно изменить состояние на противоположенное.

Таблица 2. Описание выводов колодки клеммников реле модуля МР714

Обозначение	Описание вывода
REL1.1	НЗ контакт 1-го реле
REL1.2	Общий контакт 1-го реле
REL1.3	НО контакт 1-го реле
REL2.1	НЗ контакт 2-го реле
REL2.2	Общий контакт 2-го реле
REL2.3	НО контакт 2-го реле
REL3.1	НЗ контакт 3-го реле
REL3.2	Общий контакт 3-го реле
REL3.3	НО контакт 3-го реле
REL4.1	НЗ контакт 4-го реле
REL4.2	Общий контакт 4-го реле
REL4.3	НО контакт 4-го реле

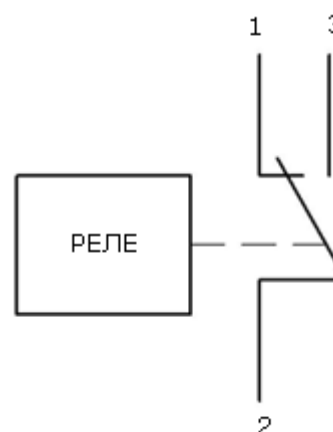


Рис. Нумерация выводов реле

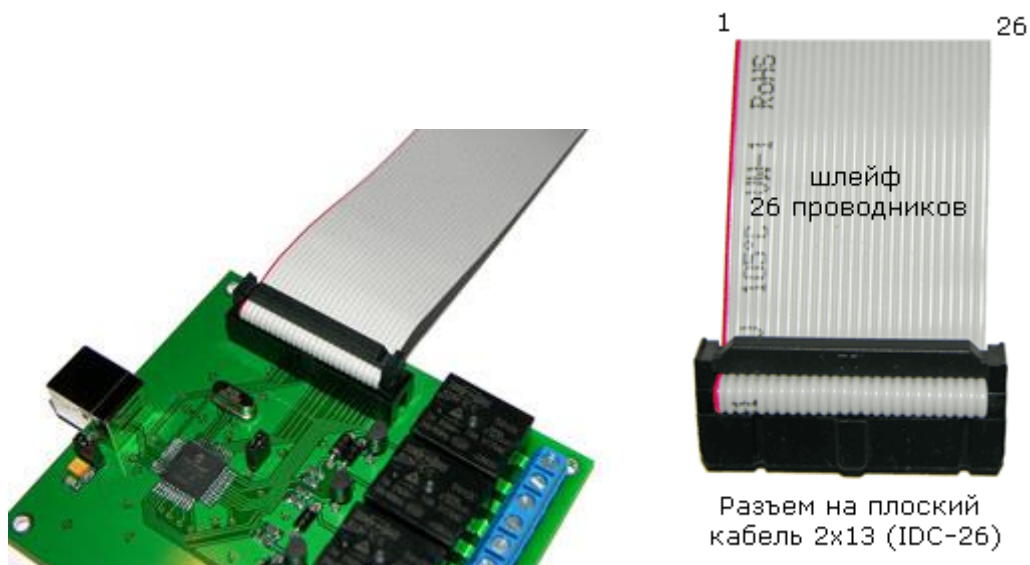


Таблица 3. Соответствие выводов разъема IDC модуля и проводников плоского шлейфа, подключаемого к разъему.

Номер проводника шлейфа	Обозначение	Номер проводника шлейфа	Обозначение
1	+ 5 В	14	I/O14
2	GND	15	I/O7
3	I/O1	16	I/O13
4	GND	17	ADC4
5	I/O2	18	I/O12
6	I/O18	19	ADC3
7	I/O3	20	I/O11
8	I/O17	21	ADC1
9	I/O4	22	ADC2
10	I/O16	23	I/O8
11	I/O5	24	NC
12	I/O15	25	I/O9
13	I/O6	26	I/O10

## 8. Правила и условия эксплуатации

Распаковать модуль из упаковки. Убедиться в отсутствии видимых механических повреждений или производственного брака. В случае обнаружения оных обратиться по месту покупки. Подключить модуль к USB порту компьютера, установить драйвера согласно инструкции. Настройку направления линий проводить только при отключенных внешних нагрузках. Изменение направления передачи данных линии с подключенной внешней нагрузкой может привести к повреждению модуля.

Рабочее напряжение логических уровней линий ввода/вывода и входа АЦП составляет +5 В. Превышение этой величины может привести к полному или частичному выходу модуля из строя.

Рекомендуемые условия эксплуатации:

- интервал температур от -30°C до 70°C
- относительная влажность воздуха до 80%



Если модуль транспортировался или эксплуатировался при температуре ниже 3°C а затем был перенесен в помещение с нормальной (комнатной) температурой, перед его включением рекомендуется выдержка в новых климатических условиях не менее 1 часа во избежание потенциального замыкания от конденсирующейся влаги.



При подключении к реле высоковольтных нагрузок и цепей соблюдать повышенную осторожность при эксплуатации и обращении, поскольку часть проводников платы оказывается под высоким напряжением. Превышение допустимых параметров нагрузки для реле (напряжение, ток) может привести модуль к выходу из строя.

## Приложение 1. Система команд модуля МР714

### Введение

Для управления модулем *МР714* предназначен набор команд в текстовом формате, называемых КЕ командами.

В качестве программного обеспечения можно использовать любую терминальную программу, например программу *HyperTerminal*, по умолчанию входящую в состав ОС Windows XP.

Любая КЕ команда, отсылаемая модулю, должна начинаться с символов '\$КЕ'. 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): ***\$КЕ,Команда<CR><LF>***

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>: ***#Ответ модуля<CR><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***

### \$КЕ

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

**Синтаксис: \$КЕ**

**Ответ на запрос: #ОК**

**Пример:**



Тестовая проверка модуля:

запрос: \$КЕ  
ответ: #ОК

### \$КЕ,FW

**Синтаксис: \$КЕ,FW**

Команда возвращает номер версии программного обеспечения модуля

**Ответ на запрос: #FW,<Version>** – значение успешно установлено.

## \$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 должна быть настроена на выход.

**Параметры:**

- LineNumber* – номер выходной дискретной линии. Может быть в пределах от 1 до 18 включительно.
- Value* – значение для записи на линию. 1 – высокий уровень напряжения, 0 – низкий уровень напряжения (0 В).

**Ответы на запрос:**

#WR,OK – значение успешно установлено.

#WR,WRONGLLINE – команда сформирована верно, но линия сконфигурирована на вход и значение не может быть установлено

**Пример:**



Установим высокий уровень напряжения на выходной дискретной линии OUT\_6:

запрос: \$KE,WR,6,1  
ответ: #WR,OK

## \$KE,WRA

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

**Синтаксис:** \$KE,WRA,<ArrayOfValues>

**Параметры:**

*ArrayOfValues* – строка длиной от 1 до 18 символов. Может содержать символы '0' (низкий уровень), '1' (высокий уровень). Нумерация символов в строке производится слева на право. Значение первого символа строки будет установлено на выходной линии 1, значение второго символа - на линии 2 и т.д. Строка может содержать меньшее число символов, чем суммарное число выходных линий, например, строка из 4-х символов позволит установить значение на первых четырех выходных линиях 1 – 4.

**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,111111111111111111  
ответ: #WRA,OK,18

**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,1111111111111111  
ответ: #WRA,OK,15

**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

**\$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 входной дискретной линии. Может быть в пределах от 1 до 18 включительно.

**Ответ на запрос:**

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

**Пример:**

Считаем 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 18 символов. 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>.... <Line18 Value>>

**Пример:**

Считать информацию со всех линий модуля:

запрос: \$KE,RD,ALL  
ответ: #RD,xxx10xxx0xxx1xxxxx

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

**\$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. От 1 до 18 включительно.

**Ответ на запрос:**

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

**Пример:**

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

**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>.... <Line18 Value> >

Ответ за запрос содержит информацию по всем 18 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,000101110011111001
запрос: $KE,RID,IN
ответ:  #RID,IN,xxx10xxx0xxx1xxxxx
запрос: $KE,RID,OUT
ответ:  #RID,OUT,000xx111x011x11001
```

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.

## \$KE,REL

Команда предназначена для управления реле модуля (включение/выключение).

**Синтаксис:** \$KE,REL,<ReleNumber>,<Value>

### Параметры:

- ReleNumber* – номер реле. Может быть в пределах от 1 до 4 включительно.
- Value* – управляющее значение. 0 – реле выключено, 1 - реле включено

### Ответ на запрос:

#REL,OK – значение успешно установлено.

### Пример:



Включим второе реле:

```
запрос: $KE,REL,2,1
ответ:  #REL,OK
```

## \$KE,RDR



Команда позволяет определить, в каком сейчас состоянии находится реле под номером *ReleNumber* – включено оно или выключено.

**Синтаксис (Вариант 1) :** `$KE,RDR,<ReleNumber>`

**Параметры:**

*ReleNumber* – номер реле. Может быть в пределах от 1 до 4 включительно.

**Ответ на запрос:**

`#RID,<ReleNumber>,<State>` – запрос состояния реле *ReleNumber* произведено успешно.

*State* = 0 – реле выключено,

*State* = 1 – реле включено.

**Пример:**



Запросим состояние 3-го реле модуля:

запрос: `$KE,RDR,3`

ответ: `#RDR,3,1`

Ответ показывает, что в данный момент 3-е реле включено.

**Синтаксис (Вариант 2) :** `$KE,RDR,ALL`

Команда позволяет получить состояние всех четырех реле за один запрос.

**Параметры:**

отсутствуют

**Ответ на запрос:**

`#RID,ALL,<Rele1 State>,<Rele2 State>,<Rele3 State>,<Rele4 State>` – запрос состояния реле произведено успешно. *State* = 0 – реле выключено, *State* = 1 – соответственно, реле включено.

**Пример:**

Запросим состояние всех реле модуля:

запрос: `$KE,RDR,ALL`

ответ: `#RDR,ALL,0,1,1,1`

**Пример:**



Запросим состояние всех реле модуля:

запрос: `$KE,RDR,ALL`

ответ: `#RDR,ALL,0,1,1,1`

## **\$KE, AFR**

Команда устанавливает частоту автоматического опроса каналов АЦП модуля. Модуль МР714 позволяет настроить автоматическую отправку значений АЦП в порт USB. Для этого необходимо задать частоту опроса для всех каналов с помощью этой команды. Затем нужно разрешить автоматический опрос необходимого канала/каналов АЦП. (см. описание команды `$KE,ADC`)

**Синтаксис:** \$KE,ADC,AFR,<Frequency>

**Параметры:**

*Frequency* – Frequency – частота опроса в герцах. Может принимать значения [0 - 400].

**Ответ на запрос:**

#AFR,OK – значение успешно установлено.

**Пример:**



Установим частоту автоматического опроса каналов АЦП равной 150 Гц:

запрос: \$KE,AFR,150

ответ: #AFR,OK

**\$KE,ADC**

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

**Синтаксис:** \$KE,ADC,<ChannelNumber>[,<AutoFlag>]

**Параметры:**

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

*AutoFlag* – опциональный параметр. Флаг, разрешающий или запрещающий автоматический опрос данного канала АЦП. Может принимать значения 0 или 1. 1 – автоматический опрос разрешен, 0 – запрещен.

**Ответ на запрос:**

#ADC,<ChannelNumber>,<Value> – аналоговому напряжению на входе *ChannelNumber* канала

АЦП модуля соответствует число *Value*. Для получения величины напряжения входного сигнала необходимо воспользоваться следующей формулой:

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

**Пример:**



Получить результат преобразования АЦП с 3-го канала:

запрос: \$KE,ADC,3

ответ: #ADC,3,0645

В данном примере на входе АЦП ADC\_1 присутствует напряжение

$$V_{in} = \frac{645}{1023} \cdot 5 = 3.152 [B]$$

## 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>[,S]

Команда позволяет настроить линию ввода/вывода *LineNumber* в качестве выходной (*IoDirection* = 0) или входной (*IoDirection* = 1). Если установлен необязательный флаг S (Save), то направление (вход/выход) линии *LineNumber* будет сохранено в энергонезависимой памяти модуля. При последующих включениях модуля линия *LineNumber* будет автоматически сконфигурирована в то направление, которое было сохранено ранее.

#### Параметры:

*LineNumber* – номер линии ввода/вывода. Может быть в пределах от 1 до 18 включительно.

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

S – флаг сохранения настройки в энергонезависимой памяти модуля.

#### Ответ на запрос:

#IO,SET,OK

#### Пример 1:



Установить линию 5 в качестве выходной, в памяти не сохранять:

запрос: \$KE,IO,SET,5,0

ответ: #IO,SET,OK

#### Пример 2:



Установить линию 3 в качестве входной, сохранить настройку в памяти:

запрос: \$KE,IO,SET,3,1,S

ответ: #IO,SET,OK

### Команда \$KE,IO,GET

**Синтаксис (Вариант 1):** \$KE,IO,GET,<Location>

Вывод значений направления ввода/вывода для всех 18 линий модуля. Информация будет соответствовать текущему состоянию, если поле *Location* равно 'CUR' или будет взята из энергонезависимой памяти ('MEM'), т.к. в общем случае значения могут отличаться (если подавалась команда \$KE,IO,SET без указания сохранять в энергонезависимой памяти)

#### Параметры:

*Location* – значение этого параметра определяет будет ли информация о направлении линий соответствовать текущему состоянию, или будет прочитана из энергонезависимой памяти.

*CUR* – текущее состояние

*MEM* – энергонезависимая память

#### Ответ на запрос:

#IO, <Line1 Direction><Line2 Direction>...<Line18 Direction>

*Line Direction* = 1 – линия настроена на вход

*Line Direction* = 0 – линия настроена на выход

### Пример 1:



Получить значения направлений ввода/вывода линий модуля на текущий момент времени:

запрос: \$KE,IO,GET,CUR

ответ: #IO,000100001100000000

### Пример 2:



Получить значения направлений ввода/вывода линий модуля, сохраненных в энергонезависимой памяти модуля:

запрос: \$KE,IO,GET,MEM

ответ: #IO,100000001000000000

В энергонезависимой памяти модуля для линий с номерами 1, 9 сохранено направление на вход, для остальных – на выход.

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

Вывод значения направления ввода/вывода для линии LineNumber. Информация будет соответствовать текущему состоянию, если поле Location равно 'CUR' или будет взята из энергонезависимой памяти ('MEM'), т.к. в общем случае значения могут отличаться (если подавалась команда \$KE,IO,SET без указания сохранять в энергонезависимой памяти).

### Параметры:

*LineNumber* – номер линии ввода/вывода. Может быть в пределах от 1 до 18 включительно.

*Location* – значение этого параметра определяет будет ли информация о направлении линий соответствовать текущему состоянию, или будет прочитана из энергонезависимой памяти.

*CUR* – текущее состояние

*MEM* – энергонезависимая память

### Ответ на запрос:

#IO,<Value>

*Line Direction* = 1 – линия настроена на вход

*Line Direction* = 0 – линия настроена на выход

### Пример 1:



Получить значение направления ввода/вывода линий модуля 13 на текущий момент времени:

запрос: \$KE,IO,GET,CUR,13

ответ: #IO,1

### Пример 2:



Получить значение направления ввода/вывода линий модуля 13, сохраненного в энергонезависимой памяти модуля:

запрос: \$KE,IO,GET,MEM,13

ответ: #IO,0

## Группа Команд \$KE,UD

Команды группы UD (User Data) предназначены для управления данными пользователя, которые модуль MP712 позволяет сохранять в своей энергонезависимой памяти.

### Команда \$KE,UD,SET

**Синтаксис:** \$KE,UD,SET,<Data>

Позволяет сохранить произвольные данные размером до 32 байт в энергонезависимой памяти модуля.

**Параметры:**

*Data* – данные для записи в память; не более 32 байт (символов).

**Ответ на запрос:**

#UD,SET,OK

**Пример 1:**



Сохранить в энергонезависимой памяти модуля строку ‘My Data for storage’:

запрос: \$KE,UD,SET,My Data for storage

ответ: #UD,SET,OK

### Команда \$KE,UD,GET

**Синтаксис:** \$KE,UD,GET

Чтение ранее сохраненных пользователем данных из энергонезависимой памяти модуля.

**Ответ на запрос:**

#UD,<Data> - чтение прошло успешно, поле Data содержит прочтенные данные

#UD,NOTSET – данные отсутствуют.

**Пример:**



Считать данные энергонезависимой памяти модуля:

запрос: \$KE,UD,GET

ответ: #UD, My Data for storage

## Группа Команд \$KE,USB

Команды группы USB предназначены для управления строковым дескриптором USB устройства. Информация содержащаяся в строковом дескрипторе устройства отображается в виде сообщения операционной системы при подключении модуля к шине USB компьютера. По умолчанию используется строка ‘MP712’. Команды этой группы позволяют установить свою собственную строку дескриптора устройства и сохранить ее в памяти модуля

### Команда \$KE,USB,SET

**Синтаксис: \$KE,UD,SET,<Data>**

Позволяет сохранить строку дескриптора USB

**Параметры:**

*Data* – данные для записи в память; не более 32 байт (символов).

**Ответ на запрос:**

#USB,SET,OK

**Пример 1:**

Установить в качестве строки дескриптора USB устройства текст ‘My USB Device’::

запрос: \$KE,USB,SET, My USB Device

ответ: #USB,SET,OK

**Команда \$KE,USB,GET****Синтаксис: \$KE,UD,GET**

Чтение строки дескриптора USB устройства.

**Ответ на запрос:**

*#USB,<Data>* - чтение прошло успешно, поле *Data* содержит строковый дескриптор.

**Пример:**

Считать строковый дескриптор из памяти модуля:

запрос: \$KE,USB,GET

ответ: #USB, My USB Device

**Команда \$KE,SER**

Чтение серийного номера модуля. Каждый модуль имеет свой собственный уникальный серийный номер.

**Синтаксис: \$KE,SER**

#SER,<Serial Number>

**Команда \$KE,RST**

Сброс всех настроек модуля в значение по умолчанию. При подаче команды все линии ввода/вывода устанавливаются на выход; на всех линиях устанавливается логический ноль; сохраненные настройки линий ввода/вывода в энергонезависимой памяти стираются и устанавливаются на выход; реле возвращаются в исходное состояние; стираются пользовательские

данные; восстанавливается строковый дескриптор по умолчанию.

**Синтаксис:** **\$KE,SER**

#SER,<Serial Number>

**Ответ на запрос:**

#RST,OK