

HL-KBD56RS/IP/S/A

Мембранная клавиатура оператора

**Руководство пользователя
V1.2/26/09**



ООО «ХОЛИТ Дэйта Системс»

✉ 03056, Украина, Киев-56,
ул. Политехническая 16, уч.корп.12, к.019
(044) 241-8739, 241-67-54, 492-31-08, 491-31-09
support@holit.ua
www.holit.ua

1. НАЗНАЧЕНИЕ



Рис.1. Внешний вид клавиатуры

Клавиатура HL-KBD56RS представляет собой малогабаритное устройство оперативного ввода информации, предназначенное для применения во встраиваемых системах управления, контроля и автоматизации, а также для организации интерфейса между оператором и контроллером (компьютером) в том случае, когда применение стандартных интерфейсных средств затруднено или нецелесообразно по тем или иным причинам.

Информационный обмен между HL-KBD56RS контроллером (компьютером) производится через интерфейс RS-485 или RS-232 в полудуплексном режиме, скорость обмена устанавливается в пределах от 300 до 115200 бод, обмен данными производится в ASCII-кодах.

Входящий в состав клавиатуры микроконтроллер, способен:

- производить опрос клавиатурной матрицы;
- выдавать звуковое подтверждение нажатия клавиши на клавиатуре пульта;

! *Следует помнить, что клавиатура является пассивным устройством, управляемым дистанционно со стороны удаленного компьютера или контроллера по стандартному последовательному интерфейсу RS485/RS232 с помощью набора предопределенных команд в ASCII формате.*

2. КОМПЛЕКТ ПОСТАВКИ

1. Клавиатура HL-KBD56RS.
2. CD-диск с программным обеспечением и руководством пользователя.

3. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

3.1. СТРУКТУРНАЯ СХЕМА

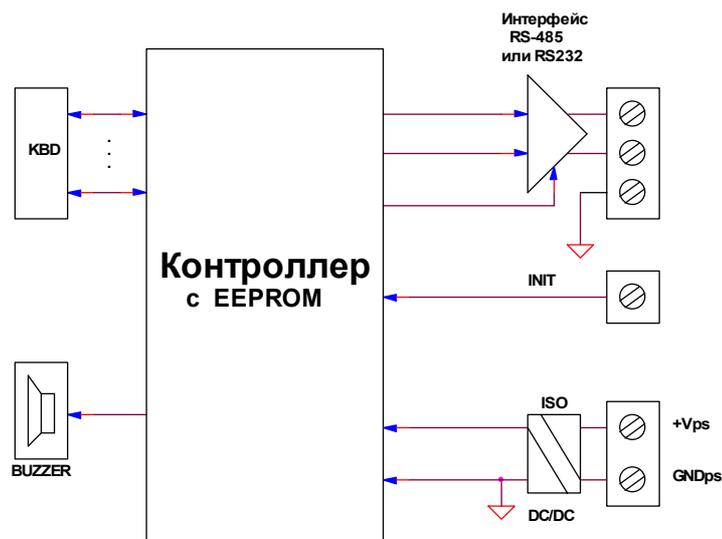


Рис.2. Структурная схема построения модуля

ISO	гальваническая развязка
KBD	мембранная клавиатура
INIT	линия перевода модуля в режим инициализации
КОНТРОЛЛИЕР	автомат управления
DC/DC	преобразователь уровней напряжений питания с гальванической развязкой
BUZZER	внутренний BUZZER

3.2 ОБЩИЕ ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

Клавиатура	Пленочная мембранная, пылевлагозащищенная. 30 кнопок (6 строк по 5 кнопок).
Количество срабатываний клавиш, не менее	1 500 000
Электрическая прочность лицевой пленки, КВ/мм, не более	250
Сопротивление изоляции	100МОм
Интерфейс связи	RS-485/RS-232
Скорость обмена	от 300 до 115200 бод
Напряжение питания	От 10 до 30В постоянного тока (номинальное 24В)
Потребляемая мощность:	Не более 0,75 Вт
Напряжение гальванической изоляции между цепями питания и интерфейса	500В
Исполнение передней панели (по степени пылевлагозащиты)	Не уступает IP54
Рабочая температура окружающего воздуха для модуля	от -20 до +60°C
Относительная влажность воздуха	5~90% без конденсации влаги
Температура хранения	от -30 до +70°C без конденсации влаги
Материал корпуса	Алюминий
Цвет корпуса	Черный (RAL9005st)
Габаритные размеры	179x153x37мм
Масса, не более	0,65кг

! По отдельному заказу возможна поставка модуля с функциональным обозначением кнопок по желанию заказчика.

! Не допускается воздействие на лицевую пленку клавиатуры колющих и режущих предметов. Очистку поверхности клавиатур от возможных загрязнений производить мягкой хлопчатобумажной ветошью, слегка увлажненной очищенным бензином или этиловым спиртом.

3.4. КОНФИГУРАЦИЯ

HL-KBD56RS поставляется со следующими настройками (параметры по умолчанию):

- Скорость обмена информацией с компьютером – 9600bps
- Адрес устройства в сети – 01h
- Тип интерфейса – RS-485
- Формат данных – 1 старт-бит, 8 бит данных, 1 стоп-бит
- Контрольная сумма – не используется
- Размер буфера клавиатуры – 8 нажатий
- Имя модуля – HMICKBD
- Звуковое подтверждение нажатия клавиши на клавиатуре пульта – отключено.

При необходимости часть из этих параметров можно изменить. Выбор параметров конфигурации производится путем установки соответствующих перемычек и переключателей на плате, а также путем перепрограммирования внутренней энергонезависимой памяти EEPROM.

3.4.1. РАСПАЙКА ОТВЕТНОГО РАЗЪЕМА ТИПА D-SUB DB-9

№ контакта	Наименование сигнала
1	RxD,
2	GND rs232
3	TxD,
4	GNDps(-)
5	+Vps.
6	DATA-
7	DATA+
8	GND RS485
9	INIT

3.4.2. ВЫБОР ТИПА ПОСЛЕДОВАТЕЛЬНОГО ИНТЕРФЕЙСА

При помощи переключателей JP1, JP2, находящихся на плате модуля, устанавливается тип последовательного интерфейса, к которому подключается модуль.

Первый контакт имеет квадратную контактную площадку.

Наименование	RS-485	RS-232
JP1	2-3	1-2
JP2	2-3	1-2

При замыкании на обоих переключателях контактов 1-2 будет выбран для работы интерфейс RS-232.

При замыкании на обоих переключателях контактов 2-3 будет выбран для работы интерфейс RS-485.

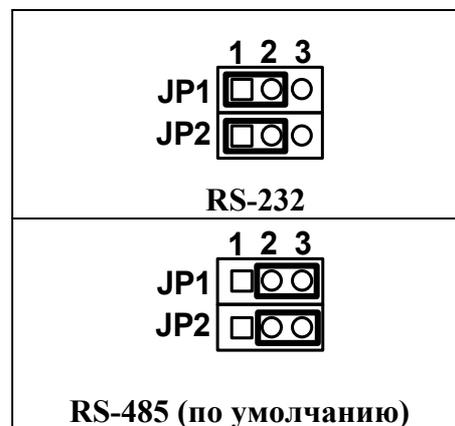
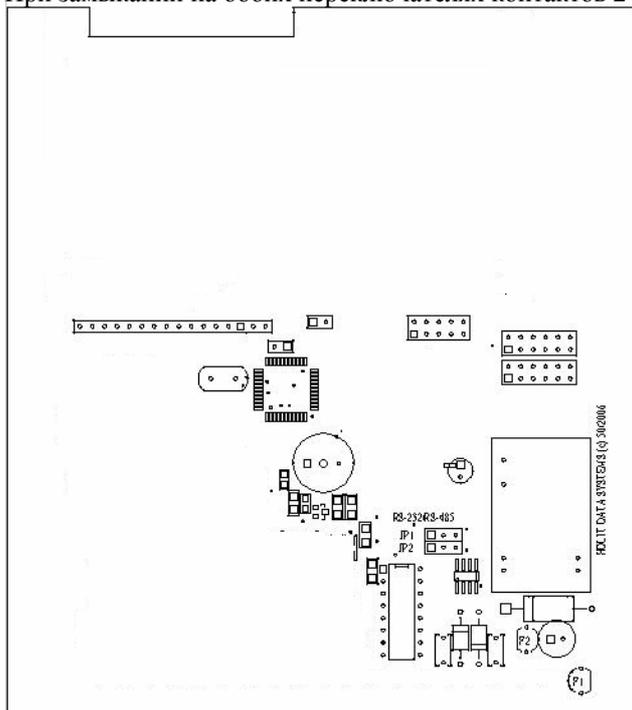


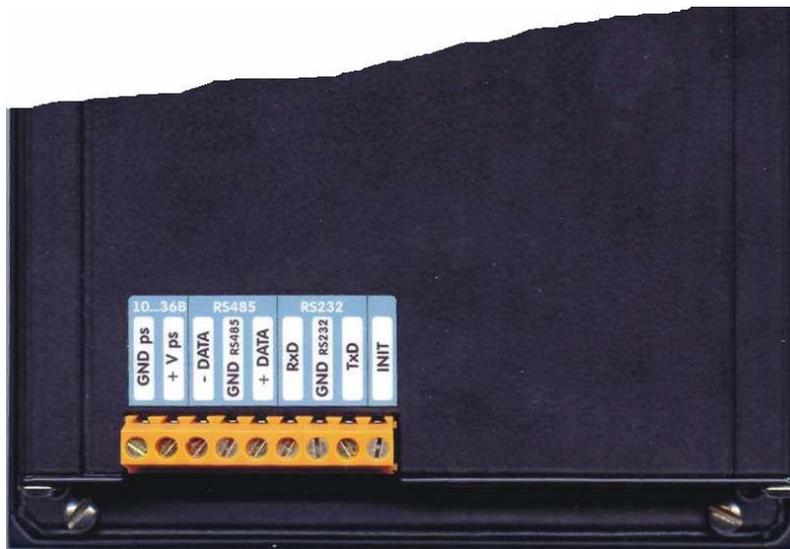
Рис.3. Монтажная схема

3.4.3. УСТАНОВКА РЕЖИМА ИНИЦИАЛИЗАЦИИ

Используется при необходимости изменения определенных, редко перенастраиваемых параметров модуля. Внешнее электрическое соединение контакта INIT с контактом GND RS485 или GND RS232 на входном разъеме модуля переведет модуль в режим *инициализации* при последующем включении питания.

Применяя команды изменения параметров модуля (см. раздел “Программирование”), можно перезаписать новые значения во внутреннюю энергонезависимую память EEPROM. Выключив модуль, следует отсоединить перемычку между контактами. Таким образом, модуль при последующем включении питания будет переведен в *нормальный* режим работы.

3.5. ПОДКЛЮЧЕНИЕ



3.5.1. ПОДКЛЮЧЕНИЕ ИСТОЧНИКА ПИТАНИЯ

Для подключения источника питания к модулю предназначены контакты с обозначением GNDps(-) и +Vps .

3.5.2. ПОДКЛЮЧЕНИЕ ИНТЕРФЕЙСА RS232

Для подключения модуля к интерфейсу RS-232 предназначены контакты с обозначением Rx D, Tx D, GND rs232 (см. Рис.4).

Пример подключения модуля к интерфейсу RS-232 управляющего компьютера

HL-KBD56RS / x / x

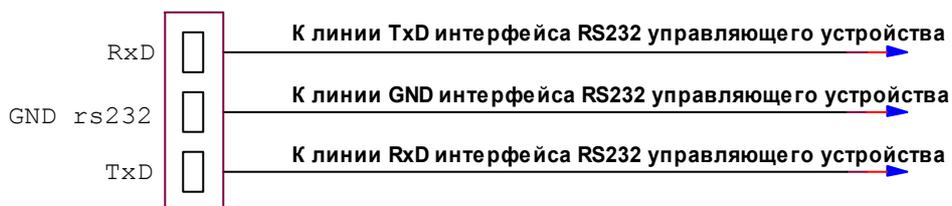


Рис.4. схема подключения интерфейса

3.5.3. ПОДКЛЮЧЕНИЕ ИНТЕРФЕЙСА RS485

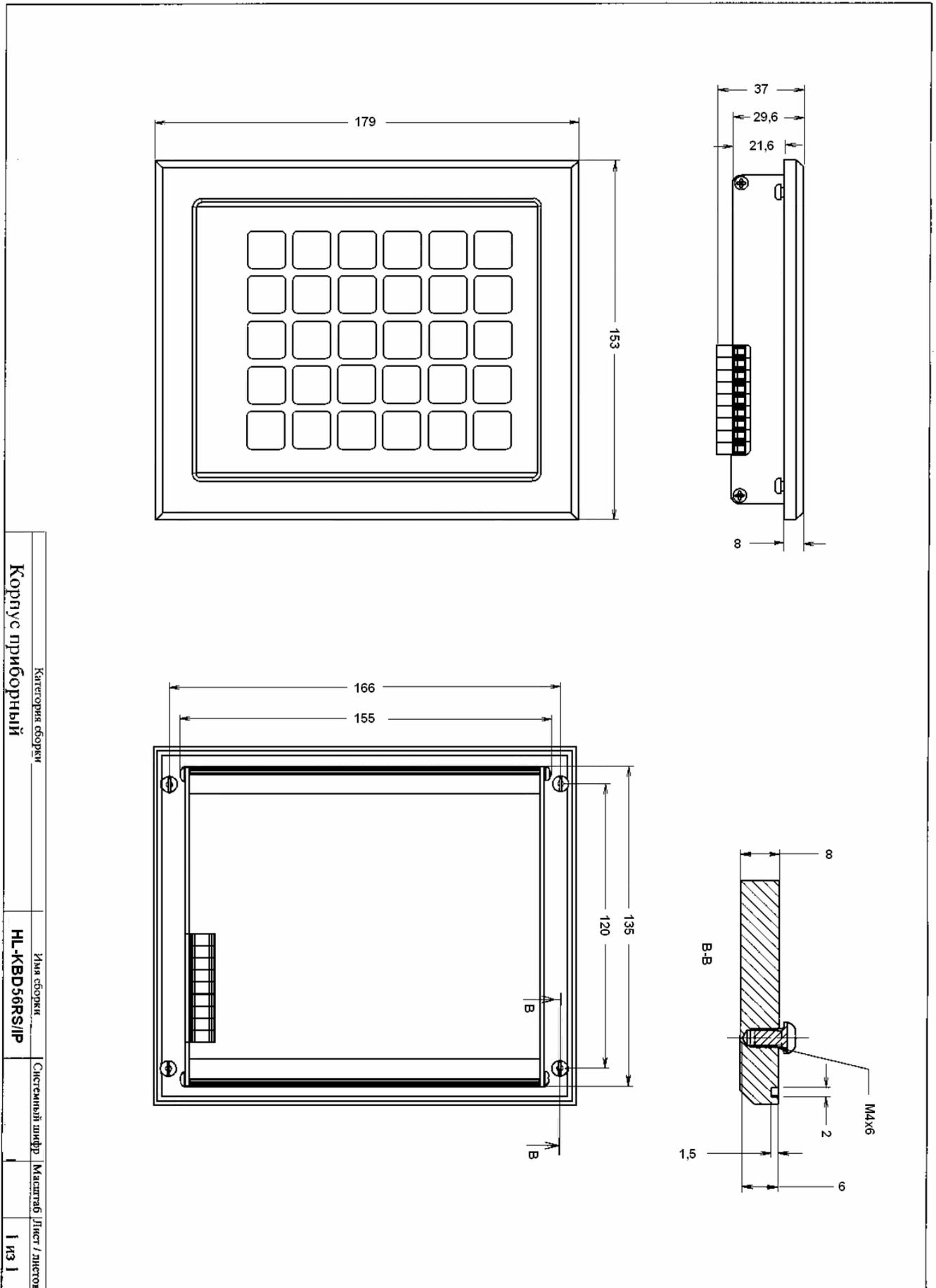
Для подключения модуля к интерфейсу RS485 предназначены контакты с обозначением DATA-, DATA+, GND RS485.

3.6. СКАН - КОДЫ КЛАВИАТУРНОЙ МАТРИЦЫ

Наименование кнопки	Скан код	Наименование кнопки	Скан код
0	05	F5	1A
1	04	F6	1B
2	0A	Ins	16
3	10	Del	1C
4	03	-	17
5	09	+	1D
6	0F	Space	18
7	02	Esc	01
8	08	Mode	07
9	0E	Bsp	0D
.	11	Enter	1E
F1	13	↑	0B
F2	14	↓	0C
F3	15	←	06
F4	19	→	12

3.7. ГАБАРИТНЫЕ И УСТАНОВОЧНЫЕ РАЗМЕРЫ

3.7.1. ЩИТОВОЕ ИСПОЛНЕНИЕ



4. РАБОТА С УСТРОЙСТВОМ

4.1. ПОРЯДОК ВКЛЮЧЕНИЯ

1. Проверьте компоненты на отсутствие механических повреждений.
2. Установите необходимые переключатели на плате.
3. Убедитесь в правильном подключении внешних сигналов.
4. Включите питание модуля.



ВНИМАНИЕ!

Все операции по установке параметров функционирования модуля (установка перемычек, подключение сигналов и т.п.) должны выполняться при отключенном питании.

4.2. ОБЩИЕ УКАЗАНИЯ ПО ПРИМЕНЕНИЮ

Рекомендации при работе с интерфейсом RS-485

В модуле установлен драйвер интерфейса RS-485 - SP485R (фирмы SIPEX), который благодаря высокому входному сопротивлению допускает подключение до 256 приемопередатчиков.

Во многих случаях может потребоваться включение двух согласующих резисторов R_T (рис.5) на обоих концах сегмента сети RS-485. Особенно это касается сети, где устройства работают на высокой скорости и длина кабеля велика.

Если протяженность сети RS-485 не более 100м, то скорее всего в применении терминирующих (согласующих) резисторов нет большой необходимости.

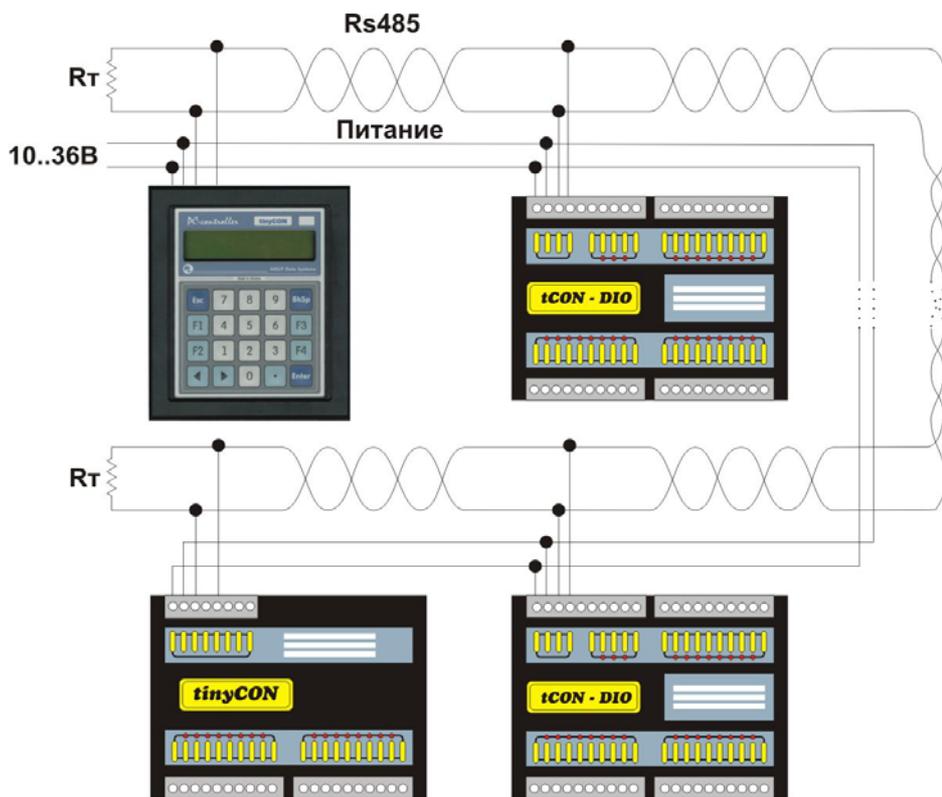


Рис.5. Типовое включение контроллера в сеть из нескольких устройств

Для каждой сети номинал резисторов может существенно отличаться и зависит от типа применяемого кабеля, а также его длины. Рассчитать сопротивление согласующего резистора, учитывая все необходимые параметры сети непросто, поэтому, для достижения наилучшей характеристики работы сети номиналы согласующих резисторов рекомендуется подобрать по форме сигнала с помощью осциллографа.



Сопротивление резистора подобрано правильно



Сопротивление резистора необходимо увеличить



Сопротивление резистора необходимо уменьшить

Рекомендации для начального подбора номинала резистора:

- а). Если длина линии около 300 м, то подбор резистора начинается с номинала 330 Ом;
- б). Если длина линии около 600 м, то подбор резистора начинается с номинала 220 Ом;
- в). Если длина линии около 1200 м, то подбор резистора начинается с номинала 110 Ом.

Примечание!

При совместной работе с модулями серий I-7000, I-8000 (ICP DAS) рекомендуется применять следующую схему включения: при питании модуля и подключаемых к нему модулей серий I-7000, I-8000 от одного источника питания, рекомендуется соединить контакты GNDps и GNDrs485. При этом необходимо помнить, что гальваническая изоляция между цепями питания и интерфейса модуля пропадет.

4.2.1. ПАРАМЕТРЫ МОДУЛЯ И НАЧАЛЬНЫЕ УСТАНОВКИ

Модуль клавиатуры имеет в своем составе перепрограммируемое запоминающее устройство с электрическим стиранием информации (EEPROM) для хранения параметров конфигурации модуля. Поэтому пользователю невозможно визуально определить установленную для модуля скорость обмена, адрес и другие параметры. Для решения этой проблемы каждый модуль имеет внешний контакт с обозначением INIT. При подаче питания на модуль, у которого этот контакт соединен с контактом GND RS485 или GND RS232, активизируется **режим инициализации**, при этом параметры конфигурации модуля временно примут следующие значения:

- **адрес** - **00**
- **скорость обмена** - **9600bps (06)**
- **контрольная сумма** - **отключена (00)**

Теперь, обратившись к модулю командой “Чтение конфигурации” (\$002), можно узнать истинные его параметры. При этом не происходит каких-либо изменений параметров в EEPROM модуля, т.е. если произвести повторное включение модуля при разомкнутой цепи между контактами INIT и GND RS485(GND RS232), то параметры конфигурации вновь примут те значения, которые были ранее сохранены в EEPROM либо изменены пользователем.

4.2.1.1. ОПРЕДЕЛЕНИЕ ПАРАМЕТРОВ МОДУЛЯ

1. Отключить питание от модуля и соединить между собой контакты INIT и GND RS485(GND RS232)
2. Подать питание на модуль
3. Послать команду чтения конфигурации \$002[0x0D]
4. Ответ содержит параметры конфигурации модуля
!010Z0600(cr) ⇒
⇒ адрес - 01, скорость обмена - 9600 bps, контрольная сумма - отключена
5. Отключить питание от модуля и разорвать связь между контактами INIT и GND RS485 (GND RS232)
6. Подать питание на модуль

Заводская конфигурация:

- адрес** - **01**
- скорость обмена** - **9600bps (06),**
- контрольная сумма** - **отключена (00)**

4.2.1.2. УСТАНОВКА ПАРАМЕТРОВ КОНФИГУРАЦИИ МОДУЛЯ

Изменение адреса модуля (может быть произведено в любой момент, результат хранится во внутренней Flash памяти контроллера):

1. Включить питание модуля
2. Послать команду чтения конфигурации \$012(cr)
3. Получить ответ !010Z0600(cr) ⇒
⇒ адрес - 01, скорость обмена - 9600 bps, контрольная сумма - отключена
4. Послать команду установки конфигурации %01020Z0600(cr) ⇒
⇒ новый адрес - 02, скорость обмена - 9600 bps, контрольная сумма - отключена
5. Ответ !02(cr) - успех.
6. Послать команду чтения конфигурации \$022(cr)
7. Ответ !020Z0600(cr) ⇒
⇒ адрес - 02, скорость обмена - 9600 bps, контрольная сумма - отключена

Изменение скорости обмена и подсчета контрольной суммы (производится только в режиме инициализации с сохранением результата в EEPROM):

1. Отключить питание от модуля и замкнуть контакты INIT и GND RS485 (GND RS232)
2. Включить питание модуля
3. Послать команду чтения конфигурации \$002(cr)

4. Ответ !020Z0600(cr) ⇒
⇒ адрес - 02, скорость обмена - 9600 bps, контрольная сумма - отключена
5. Послать команду установки конфигурации %00020Z0940(cr) ⇒
⇒ адрес - 02, новая скорость обмена - 57600 bps, контрольную сумму - включить
6. Ответ !02(cr) - успех.
7. Послать команду чтения конфигурации \$002(cr)
8. Ответ !020Z0940(cr) ⇒
⇒ адрес - 02, скорость обмена - 57600 bps, контрольная сумма - включена
9. Выключить питание модуля
10. Разомкнуть контакты INIT и GND RS485 (GND RS232)

! ВНИМАНИЕ!

Новый адрес модуля, код скорости обмена, код формата команд сохраняется во внутренней энергонезависимой памяти EEPROM, поэтому не рекомендуется часто менять эти параметры, так как максимальное количество циклов перезаписи EEPROM ограничено (типовое - 100 000).

4.3. ПРИНЦИП РАБОТЫ СДВОЕННОГО СТОРОЖЕВОГО ТАЙМЕРА

Модуль клавиатуры имеет в своем составе два сторожевых устройства:

1. Аппаратный сторожевой таймер модуля
2. Программный сторожевой таймер ведущего устройства.

Модуль предназначен для использования в системах промышленной автоматизации и поэтому может работать в жестких производственных условиях, в том числе при наличии электромагнитных помех и некачественном электропитании. Однако при значительном уровне таких дестабилизирующих факторов может произойти "зависание" модуля. Для вывода модуля из такого состояния используется внутренний аппаратный сторожевой таймер, осуществляющий перезапуск модуля. Кроме того, иногда может произойти нарушение связи или нормальной работы контроллера (компьютера), управляющего сетью модулей. Для выявления подобных ситуаций предназначен сторожевой таймер ведущего устройства.

Наличие двух сторожевых устройств в модуле обеспечивает многократное увеличение надежности работы всей системы.

4.3.1. ОСОБЕННОСТИ ИСПОЛЬЗОВАНИЯ СТОРОЖЕВОГО ТАЙМЕРА ВЕДУЩЕГО УСТРОЙСТВА

Сторожевой таймер ведущего устройства предназначен для отслеживания состояния контроллера (компьютера), управляющего сетью модулей, в процессе работы системы.

В случае обрыва линии связи интерфейса RS-485 или нарушении нормальной работы контроллера (компьютера) управляющие команды перестанут поступать к удаленным модулям системы. Это чрезвычайно опасная ситуация для реальных промышленных систем, смягчить последствия которой призван сторожевой таймер ведущего устройства. Программирование интервала этого таймера осуществляется командой ~AA3ETT. Активность управляющего контроллера определяется по команде ~** регулярно поступающей к модулям. В случае отсутствия этой команды в течение периода сторожевого таймера статус модуля примет значение 04.

Пример возможной командной последовательности для управляющего компьютера применительно к описанной ситуации приводится ниже:

1. Подключите модуль в соответствии с описанием и подайте питание
2. Подайте команду \$012(cr), в ответ Вы должны получить !010Z0600(cr)
3. Подайте команду @0100550055(cr) в ответ Вы должны получить >(cr)
4. Подайте команду ~015(cr) в ответ Вы должны получить >!01(cr)
5. Подайте команду @0100AA00AA(cr) в ответ Вы должны получить >(cr)
6. Подайте команду ~01311E(cr) в ответ Вы должны получить >!01(cr)
7. Подайте команду ~011(cr) в ответ Вы должны получить >!01(cr)

4.3.2. ОСОБЕННОСТИ ИСПОЛЬЗОВАНИЯ СТОРОЖЕВОГО ТАЙМЕРА МОДУЛЯ

Команда \$AA5 предназначена для определения факта перезагрузки модуля внутренним аппаратным сторожевым таймером. В случае непредвиденной остановки исполнения встроенного программного обеспечения модуля ("зависания"), вызванной электромагнитными помехами или сбоями по цепи питания, входящий в состав модуля сторожевой таймер обеспечит перезапуск модуля.

5. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

Раздел освещает основные вопросы программного взаимодействия с модулем .

Обмен данными с контроллером пульта оператора осуществляется по сериальному интерфейсу RS485/RS232 в режиме полудуплекса (1start+8data+1stop = 10bits) посредством предопределенного набора команд. Команда представляет собой последовательность символов ASCII, завершающихся кодом возврата каретки 0x0d (cr).

Таблица 3.1. Набор команд

Посылка	Ответ	Описание
КОМАНДЫ УПРАВЛЕНИЯ МОДУЛЕМ		
%AANN0ZCCFF	!NN	Установка параметров конфигурации модуля
\$AA2	!AA0ZBBFF	Чтение конфигурации
\$AA5	!AAS	Чтение статуса перезагрузки модуля
\$AABKX	!AA	Включение / отключение звукового сигнала подтверждения нажатия кнопки на клавиатуре
\$AABTXXXX	!AA	Вывод звукового сигнала с заданной частотой на внутренний SPEAKER модуля.
\$AAKLX	!AA	Установка размера буфера клавиатуры.
КОМАНДЫ ОБЩЕГО НАЗНАЧЕНИЯ		
\$AAK	!AAVkeys	Чтение буфера клавиатуры
\$AAF	!AAVVVV	Чтение номера версии резидентной программы
\$AAOxxxxxxxxxxxxxxxxxxxx	!AA	Команда присваивает имя модуля
\$AAM	!AA(имя)	Чтение имени модуля
КОМАНДЫ ПОВЫШЕНИЯ НАДЕЖНОСТИ СИСТЕМЫ		
~**	без ответа	Управляющее устройство в норме
~AA0	!AASS	Чтение статуса модуля
~AA1	!AA	Сброс статуса модуля
~AA2	!AASTT	Чтение настроек сторожевого таймера ведущего устройства
~AA3ETT	!AA	Установка сторожевого таймера ведущего устройства

Ситуация, когда ответ отсутствует по истечении некоторого времени после отправки команды (timeout), означает либо ошибку коммуникации, либо неверный адрес модуля в команде. Если будет послана команда с неверными параметрами, то ответ будет содержать ?AA(cr).

5.1. ОСНОВНЫЕ ПАРАМЕТРЫ КОМАНД

5.1.1. АДРЕС МОДУЛЯ

Адрес модуля AA (см. команду “Установка параметров конфигурации модуля”) задается двумя символами, которые соответствуют числовому значению в шестнадцатеричном виде от 0x00 до 0xFF. Данный параметр сохраняется в энергонезависимой памяти EEPROM.

5.1.2. СКОРОСТЬ ОБМЕНА ПО ПОСЛЕДОВАТЕЛЬНОМУ ИНТЕРФЕЙСУ

Скорость обмена (baud rate) CC (см. команду “Установка параметров конфигурации модуля”) задается для каждого модуля в пределах от 300bps до 115200bps. Данный параметр сохраняется в энергонезависимой памяти EEPROM и может быть изменен только в режиме инициализации.

CC	Baud Rate
01	300
02	600
03	1200
04	2400
05	4800
06	9600
07	19200
08	38400
09	57600
0A	115200

5.1.3. ФОРМАТ КОМАНД

Параметр формата команд FF (см. команду “Установка параметров конфигурации модуля”) определяет включение (ON) или выключение (OFF) использования контрольной суммы (checksum) в командах при обмене. Данный параметр сохраняется в энергонезависимой памяти EEPROM и может быть изменен только в режиме инициализации.

FF	Состояние
00	OFF
40	ON

Контрольная сумма представляет собой шестнадцатеричное число, состоящее из двух ASCII символов в конце команды перед кодом возврата каретки (cr). Контрольная сумма вычисляется как сумма ASCII кодов всех символов команды, при этом результатом выступает младший байт суммы.

Например, *command* = %0001880600(cr), тогда

$$\begin{aligned} \text{sum} &= \% + 0 + 0 + 0 + 1 + 8 + 8 + 0 + 6 + 0 + 0 = \\ &= 0x25 + 0x30 + 0x30 + 0x30 + 0x31 + 0x38 + 0x38 + 0x30 + 0x36 + 0x30 + 0x30 = \mathbf{0x21C} \end{aligned}$$

$$\text{checksum} = \text{sum} \& 0xFF = 0x21C \& 0xFF = \mathbf{0x1C}$$

$$\text{ASCII '1'} = 0x31$$

$$\text{ASCII 'C'} = 0x43$$

В результате *command with checksum* = %00018806001C(cr).

5.2. ОПИСАНИЕ КОМАНД УПРАВЛЕНИЯ МОДУЛЕМ

5.2.1. %AANN0ZCCFF

- Назначение: Установка параметров конфигурации модуля
- Описание: Устанавливает адрес, входной диапазон, скорость обмена, формат команд и статус контрольной суммы
- Синтаксис: %AANN0ZCCFF[chk](cr)

%	-	разделительный символ
AA	-	шестнадцатеричный адрес модуля в диапазоне от 00 до FF (в режиме инициализации адрес модуля всегда равен 00)
NN	-	новый шестнадцатеричный адрес модуля в диапазоне от 00 до FF
CC	-	код скорости обмена (*)
FF	-	код формата команд (*)
[chk]	-	двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
(cr)	-	символ возврата каретки (0x0D), используемый в качестве признака конца команды

* - данный параметр может быть изменен только в режиме инициализации.

- Ответ:

!AA[chk](cr)	⇒	команда выполнена успешно
?AA[chk](cr)	⇒	ошибка в команде (команда не воспринята)
нет ответа	⇒	синтаксическая или коммуникационная ошибка либо неверный адрес модуля

!	-	разделительный символ, обозначающий прием корректной команды
?	-	разделительный символ, обозначающий передачу неверной команды
AA	-	шестнадцатеричный адрес модуля в диапазоне от 00 до FF
[chk]	-	двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
(cr)	-	символ возврата каретки (0x0D), используемый в качестве признака конца команды

Следует помнить, что в режиме инициализации: скорость обмена информацией с компьютером – 9600bps, адрес в сети – 00h, контрольная сумма – не используется

▪ Пример:

На модуле с номером 01 установить новый адрес 02, скорость обмена 115200bps и отключить вычисление контрольной суммы:

Команда	-	%00020Z0A00(cr)
ответ	-	!02(cr)

ВНИМАНИЕ!!!

Новый адрес модуля, код скорости обмена, код формата команд сохраняется во внутренней энергонезависимой памяти EEPROM, поэтому не рекомендуется часто менять эти параметры, так как максимальное количество циклов перезаписи EEPROM ограничено (типовое - 100 000).

5.2.2. \$AA2

- Назначение: Чтение параметров конфигурации модуля
- Описание: Выполняет запрос текущих параметров конфигурации модуля
- Синтаксис: \$AA2[chk](cr)

\$	-	разделительный символ
AA	-	шестнадцатеричный адрес модуля в диапазоне от 00 до FF
[chk]	-	двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
(cr)	-	символ возврата каретки (0x0D), используемый в качестве признака конца команды

- Ответ:

!AA0ZCCFF[chk](cr)	⇒	команда выполнена успешно
?AA[chk](cr)	⇒	ошибка в команде
нет ответа	⇒	синтаксическая или коммуникационная ошибка или неверный адрес модуля

!	-	разделительный символ, обозначающий прием корректной команды
?	-	разделительный символ, обозначающий передачу неверной команды
AA	-	шестнадцатеричный адрес модуля в диапазоне от 00 до FF
CC	-	код скорости обмена
FF	-	код формата команд
[chk]	-	двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
(cr)	-	символ возврата каретки (0x0D), используемый в качестве признака конца команды

- Пример:

Определить конфигурацию модуля с номером 01:

команда	-	\$002(cr)
ответ	-	!000Z0600(cr)

На модуле с номером 01 установлена скорость обмена 9600bps и отключено вычисление контрольной суммы.

5.2.3. \$AA5

- Назначение: Чтение статуса перезагрузки модуля
- Описание: Используется для определения, был ли модуль перезагружен внутренним сторожевым таймером. В случае непредвиденной остановки исполнения встроенного программного обеспечения модуля ("зависания"), вызванной электромагнитными помехами или сбоями по цепи питания, входящий в состав модуля сторожевой таймер обеспечит перезапуск модуля.
- Синтаксис: \$AA5[chk](cr)

\$	-	разделительный символ
AA	-	шестнадцатеричный адрес модуля в диапазоне от 00 до FF
[chk]	-	двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)

- (cr) - символ возврата каретки (0x0D), используемый в качестве признака конца команды
- Ответ:
 - !AAS[chk](cr) ⇨ команда выполнена успешно
 - ?AA[chk](cr) ⇨ ошибка в команде
 - нет ответа ⇨ синтаксическая или коммуникационная ошибка либо неверный адрес модуля
- ! - разделительный символ, обозначающий прием корректной команды
- ? - разделительный символ, обозначающий передачу неверной команды
- AA - шестнадцатеричный адрес модуля в диапазоне от 00 до FF
- S - код статуса перезагрузки, возвращаемый модулем. Если S=1, то это означает, что модуль был перезагружен с момента выдачи последней команды чтения статуса перезагрузки, в противном случае S=0.
- (data) - данные, записанные по последней команде #**
- [chk] - двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
- (cr) - символ возврата каретки (0x0D), используемый в качестве признака конца команды

▪ Пример:

Считывание статуса перезагрузки модуля с номером 01.

команда	-	\$015(cr)
ответ	-	!011(cr)

5.2.4. \$AABKX

- Назначение: Включение / отключение звукового сигнала подтверждения нажатия кнопки на клавиатуре
- Описание: Команда позволяет установить короткий звуковой сигнал, звучащий при нажатии на клавишу устройства
- Синтаксис: \$AABKX[chk](cr)
 - \$ - разделительный символ
 - AA - шестнадцатеричный адрес модуля в диапазоне от 00 до FF
 - X - E – разрешен, D - запрещен
 - [chk] - двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
 - (cr) - символ возврата каретки (0x0D), используемый в качестве признака конца команды
- Ответ:
 - !AA[chk](cr) ⇨ команда выполнена успешно
 - ?AA[chk](cr) ⇨ ошибка в команде
 - нет ответа ⇨ синтаксическая или коммуникационная ошибка либо неверный адрес модуля
- ! - разделительный символ, обозначающий прием корректной команды
- ? - разделительный символ, обозначающий передачу неверной команды
- AA - шестнадцатеричный адрес модуля в диапазоне от 00 до FF
- [chk] - двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
- (cr) - символ возврата каретки (0x0D), используемый в качестве признака конца команды

▪ Пример:

Включить звуковой сигнал:

команда	-	\$01BKE (cr)
ответ	-	!01(cr)

Выключить звуковой сигнал:

Команда	-	\$01BKD (cr)
ответ	-	!01(cr)

5.2.5. \$AABTXXXX

- **Назначение:** Вывод звукового сигнала, с заданной частотой, на внутренний SPEAKER модуля.
- **Описание:** Команда позволяет подать звуковой сигнал с заданной частотой в диапазоне 10 – 10000 Гц.

▪ **Синтаксис:** \$AABT XXXX[chk](cr)

\$	-	разделительный символ
AA	-	шестнадцатеричный адрес модуля в диапазоне от 00 до FF
XXXX	-	частота звукового сигнала в HEX представлении. Если XXXX=0000 то звук выключается
[chk]	-	двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
(cr)	-	символ возврата каретки (0x0D), используемый в качестве признака конца команды

- **Ответ:**
- | | | |
|--------------|---|---|
| !AA[chk](cr) | ⇒ | команда выполнена успешно |
| ?AA[chk](cr) | ⇒ | ошибка в команде |
| нет ответа | ⇒ | синтаксическая или коммуникационная ошибка либо неверный адрес модуля |

!	-	разделительный символ, обозначающий прием корректной команды
?	-	разделительный символ, обозначающий передачу неверной команды
AA	-	шестнадцатеричный адрес модуля в диапазоне от 00 до FF
[chk]	-	двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
(cr)	-	символ возврата каретки (0x0D), используемый в качестве признака конца команды

▪ **Пример:**

Включить звуковой сигнал с частотой 1000Гц:

Команда	-	\$01BT03E8 (cr)
ответ	-	!01(cr)

Выключить звуковой сигнал:

команда	-	\$01 BT0000 (cr)
ответ	-	!01(cr)

Примечание! Если у Вас активирована команда \$AABKE, то команда \$AABTXXXX работать не будет, пока не будет подана команда \$AABKD.

5.2.6. \$AAKLX

- **Назначение:** Установка размера буфера клавиатуры.
- **Описание:** Команда позволяет установить размер буфера клавиатуры 32 или 8 нажатий. Состояние сохраняется в EEPROM модуля.

▪ **Синтаксис:** \$AAKLX[chk](cr)

\$	-	разделительный символ
AA	-	шестнадцатеричный адрес модуля в диапазоне от 00 до FF
X	-	1 – размер буфера 32 нажатия, 0 - размер буфера 8 нажатий
[chk]	-	двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)

(cr) - символ возврата каретки (0x0D), используемый в качестве признака конца команды

- Ответ:
 - !AA[chk](cr) ⇨ команда выполнена успешно
 - ?AA[chk](cr) ⇨ ошибка в команде
 - нет ответа ⇨ синтаксическая или коммуникационная ошибка либо неверный адрес модуля

! - разделительный символ, обозначающий прием корректной команды
? - разделительный символ, обозначающий передачу неверной команды
AA - шестнадцатеричный адрес модуля в диапазоне от 00 до FF
[chk] - двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
(cr) - символ возврата каретки (0x0D), используемый в качестве признака конца команды

- Пример:

32 нажатия:

команда - \$01KLX1 (cr)
ответ - !01(cr)

8 нажатий:

команда - \$01KLX0 (cr)
ответ - !01(cr)

!

ВНИМАНИЕ!!!

Данные во внутренней энергонезависимой памяти EEPROM не рекомендуется часто менять, так как максимальное количество циклов перезаписи EEPROM ограничено (типовое - 100 000).

5.3. КОМАНДЫ ОБЩЕГО НАЗНАЧЕНИЯ

5.3.1. \$AAK

- Назначение: Чтение буфера клавиатуры, в котором находятся скан-коды нажатых на клавиатуре модуля клавиш.
- Описание: Служит для чтения буфера клавиатуры и последующей его очистки (стандартный объем буфера - 8 байт скан-кодов, тип – FIFO).
Команда возвращает количество нажатых клавиш, с момента предыдущего считывания этой же командой, и последовательность скан-кодов, соответствующую очередности нажатых клавиш. После выполнения команды буфер клавиатуры очищается. При переполнении буфера, если буфер не был считан этой командой, последующие нажатия игнорируются.

- Синтаксис: \$AAK[chk](cr)

\$ - разделительный символ
AA - шестнадцатеричный адрес модуля в диапазоне от 00 до FF
[chk] - двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
(cr) - символ возврата каретки (0x0D), используемый в качестве признака конца команды

- Ответ:
 - !AAVkeys[chk](cr) ⇨ команда выполнена успешно
 - ?AA[chk](cr) ⇨ ошибка в команде
 - нет ответа ⇨ синтаксическая или коммуникационная ошибка либо неверный адрес модуля

! - разделительный символ, обозначающий прием корректной команды
? - разделительный символ, обозначающий передачу неверной команды
AA - шестнадцатеричный адрес модуля в диапазоне от 00 до FF
V - кол-во скан-кодов в буфере (0..8 стандартно , 0..32 в расширенном режиме)
keys - последовательность скан-кодов (по два hex символа на скан-код). Длина последовательности 2*V ,байт.

- [chk] - двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
- (cr) - символ возврата каретки (0x0D), используемый в качестве признака конца команды

▪ Пример:

Считывание скан-кодов нажатых клавиш из буфера клавиатуры модуля 01:

команда	-	\$01K(cr)
ответ	-	!0134E012D(cr)

Буфер содержит 3 скан-кода: 4E, 01, 2D.

5.3.2. \$AAF

- Назначение: Чтение номера версии резидентной программы модуля.

- Синтаксис: \$AAF[chk](cr)

- \$ - разделительный символ
- AA - шестнадцатеричный адрес модуля в диапазоне от 00 до FF
- [chk] - двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
- (cr) - символ возврата каретки (0x0D), используемый в качестве признака конца команды

- Ответ:

!AAVVVV[chk](cr)	⇒	команда выполнена успешно
?AA[chk](cr)	⇒	ошибка в команде
нет ответа	⇒	синтаксическая или коммуникационная ошибка либо неверный адрес модуля

- ! - разделительный символ, обозначающий прием корректной команды
- ? - разделительный символ, обозначающий передачу неверной команды
- AA - шестнадцатеричный адрес модуля в диапазоне от 00 до FF
- VVVV - символы, содержащие версию резидентной программы
- [chk] - двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
- (cr) - символ возврата каретки (0x0D), используемый в качестве признака конца команды

▪ Пример:

Определить номер версии резидентной программы модуля 01:

команда	-	\$01F(cr)
ответ	-	!01V1.0(cr)

5.3.3. \$AAOXXXXXXXXXXXXXXXXXXXXX

- Назначение: Назначение имени модуля
- Описание: Команда присваивает имя модуля и сохраняет значение в EEPROM.

- Синтаксис: \$AAOXXXXXXXXXXXXXXXXXXXXX[chk](cr)

- \$ - разделительный символ
- AA - шестнадцатеричный адрес модуля в диапазоне от 00 до FF
- X - 1..20-ти символьное имя модуля
- [chk] - двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
- (cr) - символ возврата каретки (0x0D), используемый в качестве признака конца команды

- Ответ: !AA[chk](cr) ⇒ команда выполнена успешно

?AA[chk](cr)	⇒	ошибка в команде
нет ответа	⇒	синтаксическая или коммуникационная ошибка либо неверный адрес модуля

!	-	разделительный символ, обозначающий прием корректной команды
?	-	разделительный символ, обозначающий передачу неверной команды
AA	-	шестнадцатеричный адрес модуля в диапазоне от 00 до FF
[chk]	-	двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
(cr)	-	символ возврата каретки (0x0D), используемый в качестве признака конца команды

▪ Пример:

Присвоить имя НМІ01 для модуля с номером 01:

команда	-	\$01ОНМІ01 (cr)
ответ	-	!01(cr)

! **ВНИМАНИЕ!!!**

Данные во внутренней энергонезависимой памяти EEPROM не рекомендуется часто менять, так как максимальное количество циклов перезаписи EEPROM ограничено (типовое - 100 000).

5.3.4. \$AAM

▪ Назначение: Чтение имени модуля.

▪ Синтаксис: \$AAM[chk](cr)

\$	-	разделительный символ
AA	-	шестнадцатеричный адрес модуля в диапазоне от 00 до FF
[chk]	-	двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
(cr)	-	символ возврата каретки (0x0D), используемый в качестве признака конца команды

▪ Ответ:

!AАНМІСКВD[chk](cr)	⇒	выполнена успешно
?AA[chk](cr)	⇒	ошибка в команде
нет ответа	⇒	синтаксическая или коммуникационная ошибка либо неверный адрес

!	-	разделительный символ, обозначающий прием корректной команды
?	-	разделительный символ, обозначающий передачу неверной команды
AA	-	шестнадцатеричный адрес модуля в диапазоне от 00 до FF
[chk]	-	двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
(cr)	-	символ возврата каретки (0x0D), используемый в качестве признака конца команды

▪ Пример:

Прочитать имя модуля 01:

команда	-	\$01M(cr)
ответ	-	!01НМІСКВD(cr)

5.4. КОМАНДЫ ПОВЫШЕНИЯ НАДЕЖНОСТИ СИСТЕМЫ

5.4.1. ~**

▪ Назначение: Управляющее устройство в норме

▪ Описание: Сообщает всем модулям системы о том, что управляющее устройство (контроллер, компьютер) работает в штатном режиме.

▪

- **Синтаксис:** ~**[chk](cr)
- ~ - разделительный символ
- [chk] - двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
- (cr) - символ возврата каретки (0x0D), используемый в качестве признака конца команды

▪ **Ответ:** без ответа

▪ **Пример:**

Подать команду “контроллер в норме”:

команда	-	~**(cr)
ответ	-	без ответа

5.4.2. ~AA0

- **Назначение:** Чтение статуса модуля
- **Описание:** Производит чтение статуса адресуемого модуля. При нарушении нормальной работы управляющего устройства системы и при истечении интервала времени сторожевого таймера ведущего устройства, работа которого была разрешена, статус модуля примет значение 04.

▪ **Синтаксис:** ~AA0[chk](cr)

- ~ - разделительный символ
- AA - шестнадцатеричный адрес модуля в диапазоне от 00 до FF
- [chk] - двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
- (cr) - символ возврата каретки (0x0D), используемый в качестве признака конца команды

▪ **Ответ:**

!AASS[chk](cr)	⇒	команда выполнена успешно
?AA[chk](cr)	⇒	ошибка в команде
нет ответа	⇒	синтаксическая или коммуникационная ошибка либо неверный адрес модуля

- ! - разделительный символ, обозначающий прием корректной команды
- ? - разделительный символ, обозначающий передачу неверной команды
- AA - шестнадцатеричный адрес модуля в диапазоне от 00 до FF
- SS - код статуса модуля. Если SS=00, то все в норме, а если SS=04, то имело место срабатывание сторожевого таймера ведущего устройства.
- [chk] - двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
- (cr) - символ возврата каретки (0x0D), используемый в качестве признака конца команды

▪ **Пример:**

Считывание статуса модуля с номером 01.

команда	-	~010(cr)
ответ	-	!0100(cr)

5.4.3. ~AA1

- **Назначение:** Сброс статуса модуля.
- **Описание:** Производит сброс адресуемого модуля. Если статус модуля равен 04 пользователь должен осуществить сброс статуса модуля и убедиться, что статус модуля равен 00.
- **Синтаксис:** ~AA1[chk](cr)

- ~ - разделительный символ
- AA - шестнадцатеричный адрес модуля в диапазоне от 00 до FF
- [chk] - двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
- (cr) - символ возврата каретки (0x0D), используемый в качестве признака конца команды

- **Ответ:**

!AA[chk](cr)	⇒	команда выполнена успешно
?AA[chk](cr)	⇒	ошибка в команде
нет ответа	⇒	синтаксическая или коммуникационная ошибка либо неверный адрес модуля

- ! - разделительный символ, обозначающий прием корректной команды
- ? - разделительный символ, обозначающий передачу неверной команды
- AA - шестнадцатеричный адрес модуля в диапазоне от 00 до FF
- [chk] - двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
- (cr) - символ возврата каретки (0x0D), используемый в качестве признака конца команды

▪ **Пример:**

Осуществить сброс статуса модуля с номером 01.

команда	-	~011(cr)
ответ	-	!01(cr)

5.4.4. ~AA2

- **Назначение:** чтение настроек сторожевого таймера ведущего устройства.
- **Описание:** Производит чтение статуса и значения временного интервала сторожевого таймера ведущего устройства адресуемого модуля. Этот программно реализованный таймер предназначен для контроля состояния управляющего контроллера (компьютера) системы. После активизации (запуска) сторожевого таймера управляющий контроллер (компьютер) должен осуществить его сброс командой ~** до истечения сторожевого интервала. Принятая модулем команда ~** вызывает сброс сторожевого таймера и его перезапуск. Для управления работой сторожевого таймера ведущего устройства и установки сторожевого интервала, необходимо использовать команду ~AAЗЕТТ.

- **Синтаксис:** ~AA2[chk](cr)

- ~ - разделительный символ
- AA - шестнадцатеричный адрес модуля в диапазоне от 00 до FF
- [chk] - двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
- (cr) - символ возврата каретки (0x0D), используемый в качестве признака конца команды

- **Ответ:**

!AASTT[chk](cr)	⇒	команда выполнена успешно
?AA[chk](cr)	⇒	ошибка в команде
нет ответа	⇒	синтаксическая или коммуникационная ошибка либо неверный адрес модуля

- ! - разделительный символ, обозначающий прием корректной команды
- ? - разделительный символ, обозначающий передачу неверной команды
- AA - шестнадцатеричный адрес модуля в диапазоне от 00 до FF
- S - признак работы сторожевого таймера ведущего устройства:
S=0 - сторожевой таймер заблокирован,
S=1 - сторожевой таймер активизирован.
- TT - двухсимвольное шестнадцатеричное число в диапазоне от 00 до FF, определяющее значение интервала сторожевого таймера с разрешением 0,1с.
- [chk] - двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
- (cr) - символ возврата каретки (0x0D), используемый в качестве признака конца команды

▪ Пример:

Считывание настроек сторожевого таймера ведущего устройства модуля с номером 01.

команда	-	~012(cr)
ответ	-	!0111E(cr)

Результат: сторожевой таймер ведущего устройства модуля с номером 01 активизирован, его сторожевой интервал равен 3с (0,1*30=3).

5.4.5. ~AA3ETT

- Назначение: установка статуса и значения интервала сторожевого таймера ведущего устройства.
- Описание: Производит установку статуса и значения интервала сторожевого таймера ведущего устройства адресуемого модуля. Этот программно реализованный таймер предназначен для контроля состояния управляющего контроллера (компьютера) системы. После активизации (запуска) сторожевого таймера управляющий контроллер (компьютер) должен осуществить его сброс командой ~** до истечения сторожевого интервала. Принятая модулем команда ~** вызывает сброс сторожевого таймера и его перезапуск. Для чтения статуса и значения сторожевого интервала этого таймера необходимо использовать команду ~AA2..

▪ Синтаксис: ~AA3ETT[chk](cr)

~	-	разделительный символ
AA	-	шестнадцатеричный адрес модуля в диапазоне от 00 до FF
E	-	признак работы сторожевого таймера ведущего устройства: E=0 – (блокировка) работа сторожевого таймера блокируется E=1 – (активизация) работа сторожевого таймера разрешается
TT	-	двухсимвольное шестнадцатеричное число в диапазоне от 00 до FF, определяющее значение интервала сторожевого таймера с разрешением 0,1с.
[chk]	-	двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
(cr)	-	символ возврата каретки (0x0D), используемый в качестве признака конца команды

- Ответ:

!AA[chk](cr)	⇒	команда выполнена успешно
?AA[chk](cr)	⇒	ошибка в команде
нет ответа	⇒	синтаксическая или коммуникационная ошибка либо неверный адрес модуля

!	-	разделительный символ, обозначающий прием корректной команды
?	-	разделительный символ, обозначающий передачу неверной команды
AA	-	шестнадцатеричный адрес модуля в диапазоне от 00 до FF
[chk]	-	двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
(cr)	-	символ возврата каретки (0x0D), используемый в качестве признака конца команды

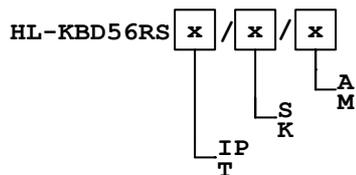
▪ Пример:

Для модуля с номером 01 запустить сторожевой таймер ведущего устройства и установить сторожевой интервал 3с.

команда	-	~01311E(cr)
ответ	-	!01(cr)

6. ИНФОРМАЦИЯ ДЛЯ ЗАКАЗА

HL-KBD56RS/x/x/x



Базовая версия пульта оператора

HL-KBD56RS/IP/S	мембранная клавиатура 30 клавиш (матрица 5x6), для ПК, интерфейс RS-485/RS-232, системой команд в ASCII формате (протокол DCON), питание от внешнего источника 10-36В), щитовое исполнение корпуса, защита передней панели IP54, материал корпуса алюминий
------------------------	---

Опции для заказа при отклонении от базовой версии

/IP	щитовое исполнение корпуса, защита передней панели IP54
/T	"настольное" исполнение корпуса (IP20)
/S	поставка с мембранной клавиатурой с "рисунком" производителя
/K	поставка с мембранной клавиатурой с "рисунком" заказчика
/A	поставка с системой команд в ASCII формате (протокол DCON)
/M	поставка с протоколом обмена MODBUS RTU SLAVE

ООО «ХОЛИТ Дэйта Системс» оставляет за собой право изменять данное руководство пользователя и модифицировать изделия без уведомления покупателей.

ООО «ХОЛИТ Дэйта Системс» не несет какой-либо ответственности за результат использования, информации представленной в настоящем руководстве, поскольку невозможно гарантировать, что данное изделие пригодно для всех целей, в которых оно может применяться покупателем.