

HMI-LCD2004A

Операторская индикаторная панель

Руководство пользователя

1.1/В1.8.



«ХОЛИТ Дэйта Системс, Лтд»

✉ 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.com.ua
info@holit.com.ua
www.holit.com.ua

Содержание

1. НАЗНАЧЕНИЕ.....	4
2. КОМПЛЕКТ ПОСТАВКИ.....	4
3. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ	5
3.1. СТРУКТУРНАЯ СХЕМА.....	5
3.2. ОБЩИЕ ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ.....	6
3.3. КОНФИГУРАЦИЯ.....	6
3.3.1. ВЫБОР ТИПА ПОСЛЕДОВАТЕЛЬНОГО ИНТЕРФЕЙСА	7
3.3.2. УСТАНОВКА РЕЖИМА ИНИЦИАЛИЗАЦИИ	7
3.3.3. РЕГУЛИРОВКА КОНТРАСТНОСТИ СВЕЧЕНИЯ LCD ДИСПЛЕЯ.....	7
3.4. ПОДКЛЮЧЕНИЕ.....	8
3.4.1. ПОДКЛЮЧЕНИЕ ИСТОЧНИКА ПИТАНИЯ	8
3.4.2. ПОДКЛЮЧЕНИЕ ИНТЕРФЕЙСА RS485.....	8
3.4.3. ПОДКЛЮЧЕНИЕ ИНТЕРФЕЙСА RS232.....	8
3.5. ИНДИКАТОР	9
3.5.1. ТИПОВЫЙ ЗНАКОГЕНЕРАТОР LCD ДИСПЛЕЯ.....	9
3.5.2. ТАБЛИЦА НАБОРОВ СИМВОЛОВ, СООТВЕТСТВУЮЩАЯ КОДОВОЙ СТРАНИЦЕ CP-866.....	10
3.6. ГАБАРИТНЫЕ И УСТАНОВОЧНЫЕ РАЗМЕРЫ	11
3.6.1. ЩИТОВОЕ ИСПОЛНЕНИЕ	11
4. РАБОТА С УСТРОЙСТВОМ	11
4.1. ПОРЯДОК ВКЛЮЧЕНИЯ	11
4.2. ОБЩИЕ УКАЗАНИЯ ПО ПРИМЕНЕНИЮ	12
4.2.1. ПАРАМЕТРЫ МОДУЛЯ И НАЧАЛЬНЫЕ УСТАНОВКИ	12
4.2.1.1. ОПРЕДЕЛЕНИЕ ПАРАМЕТРОВ МОДУЛЯ	12
4.2.1.2. УСТАНОВКА ПАРАМЕТРОВ КОНФИГУРАЦИИ МОДУЛЯ.....	12
4.3. ПРИНЦИП РАБОТЫ СДВОЕННОГО СТОРОЖЕВОГО ТАЙМЕРА.....	13
4.3.1. ОСОБЕННОСТИ ИСПОЛЬЗОВАНИЯ СТОРОЖЕВОГО ТАЙМЕРА ВЕДУЩЕГО УСТРОЙСТВА	13
4.3.2. ОСОБЕННОСТИ ИСПОЛЬЗОВАНИЯ СТОРОЖЕВОГО ТАЙМЕРА МОДУЛЯ	14
4.3.3. СТАТУС МОДУЛЯ	14
4.4. ИСПОЛЬЗОВАНИЕ СИМВОЛОВ КИРИЛЛИЦЫ ПРИ ВЫВОДЕ ТЕКСТОВОЙ НА LCD ДИСПЛЕЙ	15
5. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ	15
5.1. ОСНОВНЫЕ ПАРАМЕТРЫ КОМАНД.....	16



5.1.1.	АДРЕС МОДУЛЯ	16
5.1.2.	СКОРОСТЬ ОБМЕНА ПО ПОСЛЕДОВАТЕЛЬНОМУ ИНТЕРФЕЙСУ	16
5.1.3.	ФОРМАТ КОМАНД	16
5.2.	ОПИСАНИЕ КОМАНД УПРАВЛЕНИЯ МОДУЛЕМ.....	17
5.2.1.	%AANN0ZCCFF.....	17
5.2.2.	\$AA2.....	17
5.2.3.	\$AA5.....	18
5.3.	КОМАНДЫ УПРАВЛЕНИЯ LCD ИНДИКАТОРОМ МОДУЛЯ.....	19
5.3.1.	\$AASC.....	19
5.3.2.	\$AAGNLPPPPPPPPPPPPPPPP.....	20
5.3.3.	\$AALNC.....	20
5.3.4.	\$AALCCCC.....	21
5.3.5.	\$AAT.....	22
5.3.6.	\$AATYXX(str).....	22
5.3.7.	\$AAIV.....	23
5.3.8.	\$AAPYXX.....	23
5.4.	КОМАНДЫ ОБЩЕГО НАЗНАЧЕНИЯ.....	24
5.4.1.	\$AAF.....	24
5.4.2.	\$AAM.....	25
5.5.	КОМАНДЫ ПОВЫШЕНИЯ НАДЕЖНОСТИ СИСТЕМЫ	25
5.5.1.	~**.....	25
5.5.2.	~AA0.....	26
5.5.3.	~AA1.....	26
5.5.4.	~AA2.....	27
5.5.5.	~AA3ETT.....	28
6.	ИНФОРМАЦИЯ ДЛЯ ЗАКАЗА.....	29

1. НАЗНАЧЕНИЕ



Рис.1. Внешний вид индикаторной панели

Индикаторная панель HMI-LCD2004A представляет собой полнофункциональное малогабаритное устройство отображения информации, предназначенное для применения во встраиваемых системах управления, контроля и автоматизации, а также для организации процесса визуализации контролируемых параметров в том случае, когда применение стандартных средств отображения затруднено по тем или иным причинам. Информационный обмен между HMI-LCD2004A и компьютером производится через интерфейс RS-485 или RS-232, по протоколу, совместимому с серией I-7000.

Вывод на индикатор осуществляется компьютером путем выдачи набора простых команд в ASCII формате, которые передаются через последовательный порт.

Входящий в состав индикаторной панели модуль CB-LCD, выполненный на базе перепрограммируемого микроконтроллера AT89S8252, способен:

- осуществлять вывод текстовой информации на алфавитно-цифровой жидкокристаллический LCD дисплей латинскими и русскими символами, используя таблицу кодировки CP866 (предполагает работу пульта оператора с приложениями DOS); имеется возможность работы с украинскими символами (подгружаемые символы пользователя);
- осуществлять программное управление (включение-выключение) подсветкой LCD дисплея.

2. КОМПЛЕКТ ПОСТАВКИ

1. Индикаторная панель HMI-LCD2004A.
2. CD-диск с программным обеспечением и руководством пользователя.

3. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

3.1. СТРУКТУРНАЯ СХЕМА

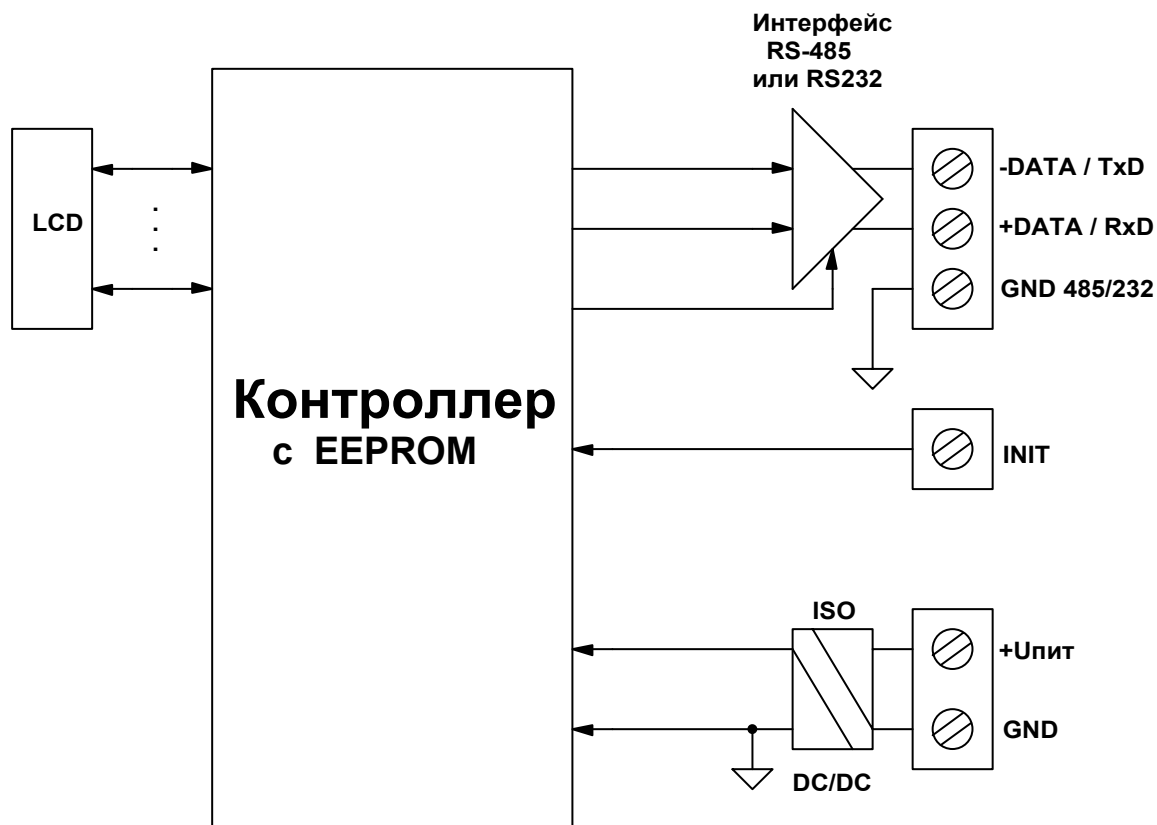


Рис.2. Структурная схема построения модуля

ISO	гальваническая развязка
LCD	знакосинтезирующий LCD индикатор
INIT	линия перевода модуля в режим инициализации
КОНТРОЛЛЕР	автомат управления
DC/DC	преобразователь уровней напряжений питания с гальванической развязкой

3.2. ОБЩИЕ ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

HMI-LCD2004ю	знакосинтезирующий LCD индикатор 4 строки по 20 символов со светодиодной подсветкой, высота символов 9.22 мм, синий фон подсветки, белые символы.
Кодовая таблица по ASCII	CP866 (DOS)
Напряжение питания	От 10 до 30В постоянного тока (номинальное 24В)
Потребляемая мощность при номинальном напряжении питания	Не более 1,0 Вт (без подсветки), не более 1,5 Вт (с подсветкой)
Напряжение гальванической изоляции между цепями питания и интерфейса	500В
Исполнение передней панели (по степени пылевлагозащиты)	IP54
Рабочая температура окружающего воздуха для модуля	от -20 до +60°C
Относительная влажность воздуха	90% при температуре +20°C.
Температура хранения	от -30 до +70°C.
Материал корпуса	Алюминий
Габаритные размеры	щитовое исполнение 171 x 88 x 39,4 мм
Масса, не более	0,3 кг.

Ввод-вывод данных производится через последовательный порт RS485 в полудуплексном режиме, скорость обмена устанавливается в пределах от 300 до 115200 бод, обмен данными производится в ASCII-кодах (возможность работы с модулями серии I-7000, I-8000, ADAM-4000 и NuDAM-6000, а также с семейством модулей tetraCON).

Питание устройства LCD2004A производится от источника постоянного тока. Наличие встроенного DC/DC преобразователя с гальванической развязкой позволяет использовать питающее напряжение в диапазоне 10..36 В. Питание LCD дисплея и его подсветки осуществляется через внутренний разъем на плате контроллера и не требует дополнительных источников питания.

Дисплей - жидкокристаллический буквенно-цифровой, 4 строки по 20 символов, светодиодная подсветка.

Конструкция модуля предполагает щитовой вариант крепления, при котором обеспечивается пылевлагозащита передней панели по классу IP54 (опция - IP65).

Многофункциональная операторская панель HMI-LCD2004 за счет своих малых размеров в сочетании с большими функциональными возможностями находит широкое применение для визуализации и ввода данных в различных промышленных системах.

3.3. КОНФИГУРАЦИЯ

HMI-LCD2004A поставляется со следующими настройками (параметры по умолчанию):

- Кодовая страница – CP866
- При включении питания подсветка дисплея – включена
- Курсор – включен, в виде мерцающего знакоместа 5x8
- Скорость обмена информацией с компьютером – 9600bps
- Адрес устройства в сети – 01h
- Тип интерфейса – RS485
- Формат данных – 1 старт-бит, 8 бит данных, 1 стоп-бит
- Контрольная сумма – не используется

При необходимости часть из этих параметров можно изменить. Выбор параметров конфигурации HMI-LCD2004 производится путем установки соответствующих переключателей и перемычек на плате, а также путем перепрограммирования внутренней энергонезависимой памяти EEPROM.

3.3.1. ВЫБОР ТИПА ПОСЛЕДОВАТЕЛЬНОГО ИНТЕРФЕЙСА

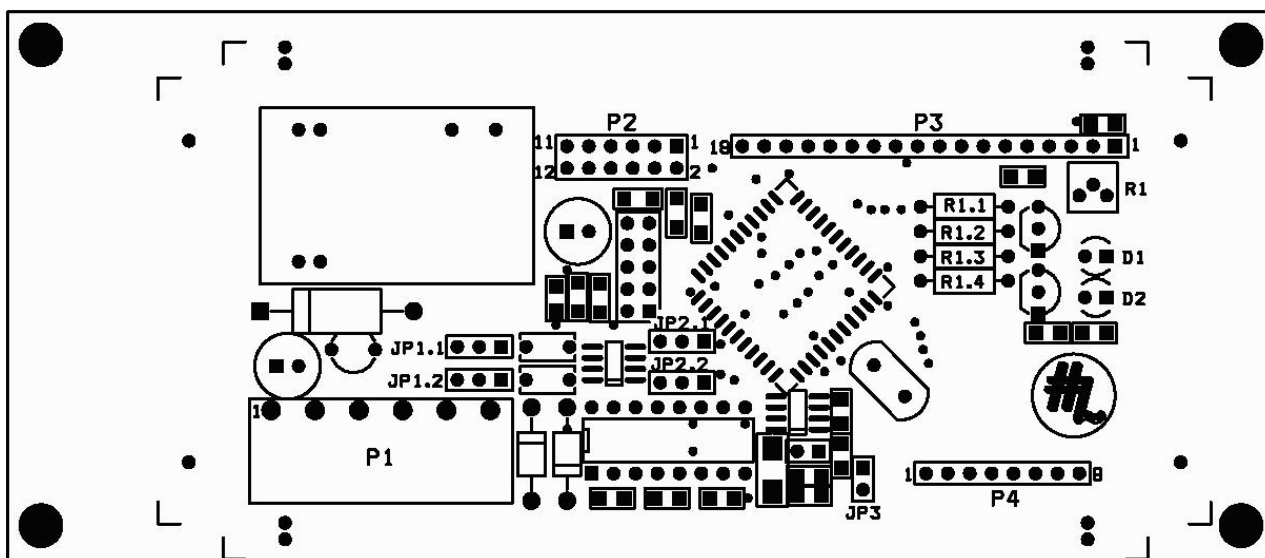
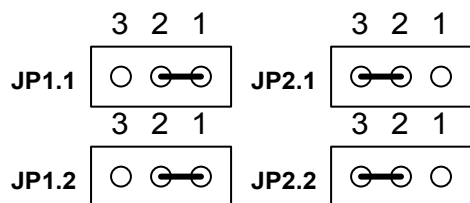


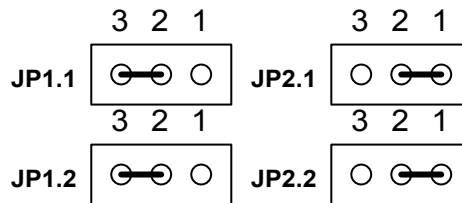
Рис.3. Монтажная схема

При помощи переключателей JP1.1, JP1.2, JP2.1, JP2.2 устанавливается тип последовательного интерфейса передачи команд и данных, (первый контакт имеет квадратную контактную площадку).

Установка интерфейса RS-485



Установка интерфейса RS-232



3.3.2. УСТАНОВКА РЕЖИМА ИНИЦИАЛИЗАЦИИ

Используется при необходимости изменения определенных редко перенастраиваемых параметров модуля. Внешнее электрическое соединение контактов внешнего разъема "под винт" P1 6(INIT) и 4(GND RS485/RS232) перед подачей питания при последующем включении переведет модуль в режим *инициализации*. Применяя команды изменения параметров модуля (см. раздел "Программирование"), можно перезаписать новые значения во внутреннюю энергонезависимую память EEPROM. Выключив модуль, следует отсоединить перемычку между контактами 6(INIT) и 4(GND RS485/RS232) разъема P1. Таким образом, модуль при последующем включении питания будет переведен в *нормальный* режим работы.

3.3.3. РЕГУЛИРОВКА КОНТРАСТНОСТИ СВЕЧЕНИЯ LCD ДИСПЛЕЯ

Переменный резистор R1 определяет контрастность свечения LCD дисплея. Регулировка контрастности доступна пользователю с задней стороны модуля.

! При первом включении жидкокристаллического дисплея следует подобрать такое положение переменного резистора R1, при котором четкость отображения символов будет максимальной.

3.4. ПОДКЛЮЧЕНИЕ

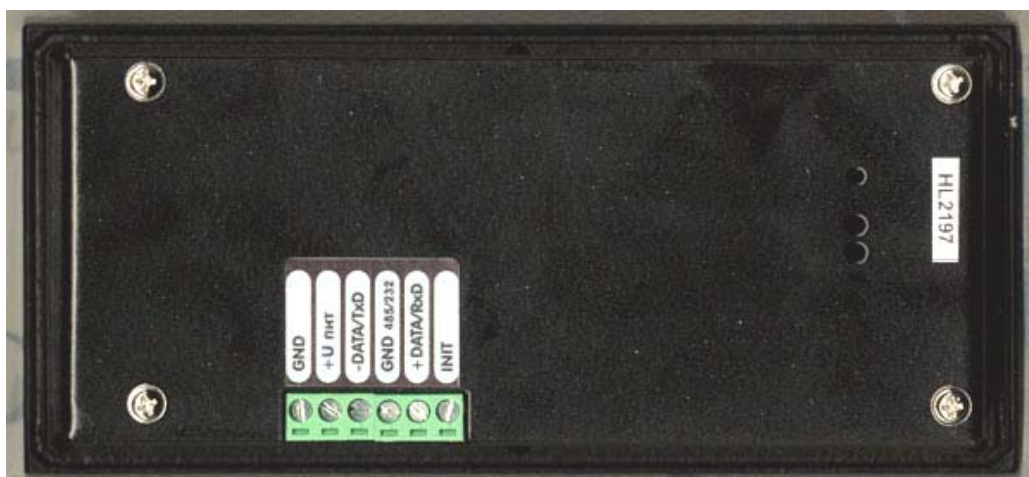


Рис.4. Вид модуля сзади

3.4.1. ПОДКЛЮЧЕНИЕ ИСТОЧНИКА ПИТАНИЯ

Для подключения источника питания служат контакты 1 и 2 внешнего разъема "под винт" P1 (см. Рис.3 и 4). К контакту 2 разъема подключается вывод питающего напряжения (+Uпит), а к контакту 1 - земляной вывод (GND).

№ контакта	Сигнал
1	GND(-)
2	+Uпит (10...36V)

3.4.2. ПОДКЛЮЧЕНИЕ ИНТЕРФЕЙСА RS485

Для подключения к интерфейсу RS485 предусмотрены контакты 3 и 5 внешнего разъема "под винт" P1 (см. Рис.3 и 4).

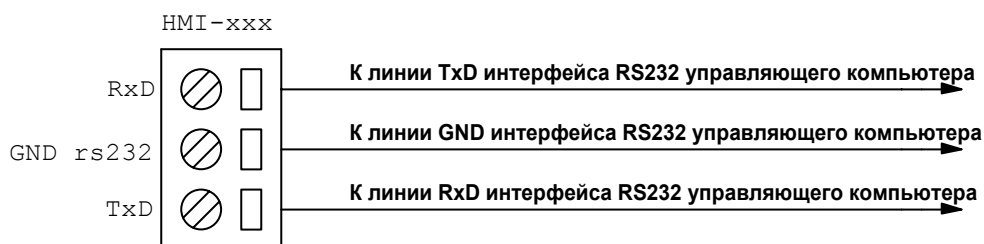
№ контакта	Сигнал
3	-DATA
4	GND RS485
5	+DATA

3.4.3. ПОДКЛЮЧЕНИЕ ИНТЕРФЕЙСА RS232

Для подключения к интерфейсу RS232 предусмотрены контакты 3 и 5 внешнего разъема "под винт" P1 (см. Рис.3 и 4).

№ контакта	Сигнал
3	TXD
4	GND RS232
5	RXD

Пример подключения к интерфейсу RS232 управляющего компьютера



3.5. ИНДИКАТОР

3.5.1. ТИПОВЫЙ ЗНАКОГЕНЕРАТОР LCD ДИСПЛЕЯ

Upper 4 bit Lower 4 bit	LLLL	LLH	LLHL	LLHH	LHLL	LHLH	LHHL	LHHH	HLLL	HLLH	HLHL	HLHH	HHLL	HHLH	HHHL	HHHH
LLLL	CG RAM (1)			0	1	2	3	4	5	6	7	8	9	A	B	C
LLH	CG RAM (2)		!	1	2	3	4	5	6	7	8	9	A	B	C	D
LLHL	CG RAM (3)		"	2	3	4	5	6	7	8	9	A	B	C	D	E
LLHH	CG RAM (4)		#	3	4	5	6	7	8	9	A	B	C	D	E	F
LHLL	CG RAM (5)		\$	4	5	6	7	8	9	A	B	C	D	E	F	G
LHLH	CG RAM (6)		%	5	6	7	8	9	A	B	C	D	E	F	G	H
LHHL	CG RAM (7)		&	6	7	8	9	A	B	C	D	E	F	G	H	I
LHHH	CG RAM (8)		'	7	8	9	A	B	C	D	E	F	G	H	I	J
HLLL	CG RAM (1)		(8	9	A	B	C	D	E	F	G	H	I	J	K
HLLH	CG RAM (2))	9	A	B	C	D	E	F	G	H	I	J	K	L
HLHL	CG RAM (3)		*	A	B	C	D	E	F	G	H	I	J	K	L	M
HLHH	CG RAM (4)		+	B	C	D	E	F	G	H	I	J	K	L	M	N
HHLL	CG RAM (5)		,	C	D	E	F	G	H	I	J	K	L	M	N	O
HHLH	CG RAM (6)		-	D	E	F	G	H	I	J	K	L	M	N	O	P
HHHL	CG RAM (7)		.	E	F	G	H	I	J	K	L	M	N	O	P	Q
HHHH	CG RAM (8)		/	F	G	H	I	J	K	L	M	N	O	P	Q	R

3.5.2. ТАБЛИЦА НАБОРОВ СИМВОЛОВ, СООТВЕТСТВУЮЩАЯ КОДОВОЙ СТРАНИЦЕ CP-866

Код символа		Старший полубайт													
		2	3	4	5	6	7	8	9	A	B	C	D	E	F
Младший полубайт	0		0	@	P	`	p	A	P	a				p	Ё
	1	!	1	A	Q	a	q	Б	С	б				с	ё
	2	"	2	B	R	b	r	В	Т	в				т	
	3	#	3	C	S	c	s	Г	У	г				у	
	4	\$	4	D	T	d	t	Д	Ф	д				ф	
	5	%	5	E	U	e	u	Е	Х	е				х	
	6	&	6	F	V	f	v	Ж	Ц	ж				ц	
	7	'	7	G	W	g	w	З	Ч	з				ч	
	8	(8	H	X	h	x	И	Ш	и				ш	
	9)	9	I	Y	i	y	Й	Щ	й				щ	
	A	*	:	J	Z	j	z	К	Ъ	к				ъ	
	B	+	;	K	[k	¹⁰	Л	Ы	л				ы	
	C	,	<	L	¢	l	¹²	М	Ь	м				ь	
	D	-	=	M]	m	¹⁵	Н	Э	н				э	
	E	.	>	N	^	n	↵	О	Ю	о				ю	
	F	/	?	O	_	o		П	Я	п				я	

3.6. ГАБАРИТНЫЕ И УСТАНОВОЧНЫЕ РАЗМЕРЫ

3.6.1. ЩИТОВОЕ ИСПОЛНЕНИЕ

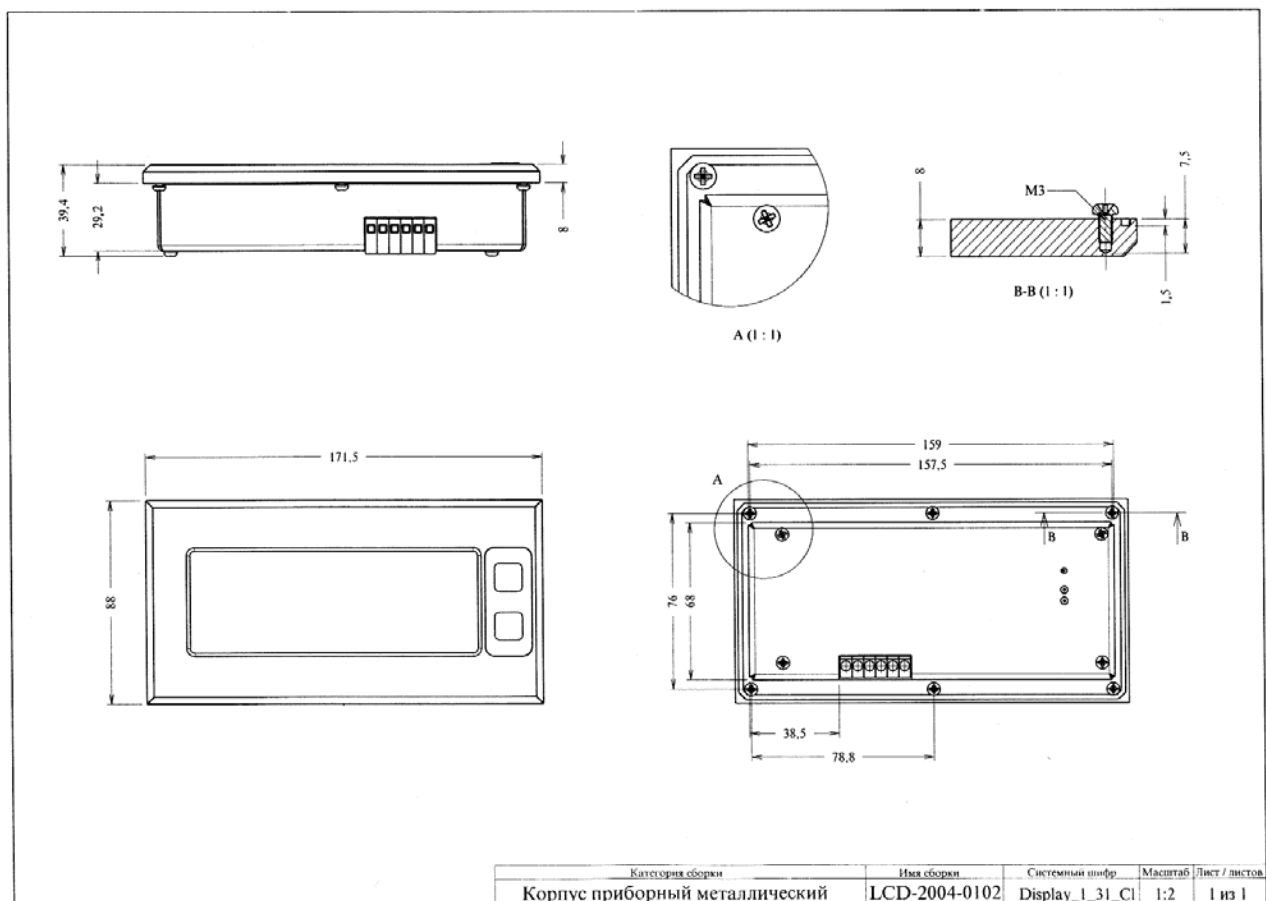


рис.5.

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.
5. Отрегулируйте контрастность свечения LCD дисплея.



ВНИМАНИЕ!!!

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

4.2. ОБЩИЕ УКАЗАНИЯ ПО ПРИМЕНЕНИЮ

4.2.1. ПАРАМЕТРЫ МОДУЛЯ И НАЧАЛЬНЫЕ УСТАНОВКИ

Все модули серии HMI имеют 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 485/232 последовательного порта, 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 485/232, 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 485/232 (режим инициализации)
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 485/232
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 485/232 (режим инициализации)
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 485/232



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

Модули серии HMI имеют в своем составе два сторожевых устройства:

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

Модули серии HMI предназначены для использования в системах промышленной автоматизации и поэтому могут 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, при этом выходные каналы модуля будут установлены в заранее определенное начальное состояние. Кроме того, иногда может произойти нарушение связи или нормальной работы контроллера (компьютера), управляющего сетью модулей. Для выявления подобных ситуаций предназначен сторожевой таймер ведущего устройства. При фиксации этим таймером отказа управляющего компьютера на выходах модулей серии HMI устанавливается заранее определенное значение безопасного уровня выходных сигналов для предотвращения создания аварийной ситуации.

Наличие двух сторожевых устройств в модулях серии HMI 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 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 на выходах модуля немедленно произойдет установка заранее определенного значения безопасного уровня выходных сигналов.

Пример возможной командной последовательности для управляющего компьютера применительно к описанной ситуации с использованием модуля 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. По истечении 3с красный светодиод модуля, расположенный на задней крышке модуля, станет мигать.
8. 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..

Блок-схема командной последовательности для определения факта срабатывания сторожевого таймера модуля приведена на рис. 10.

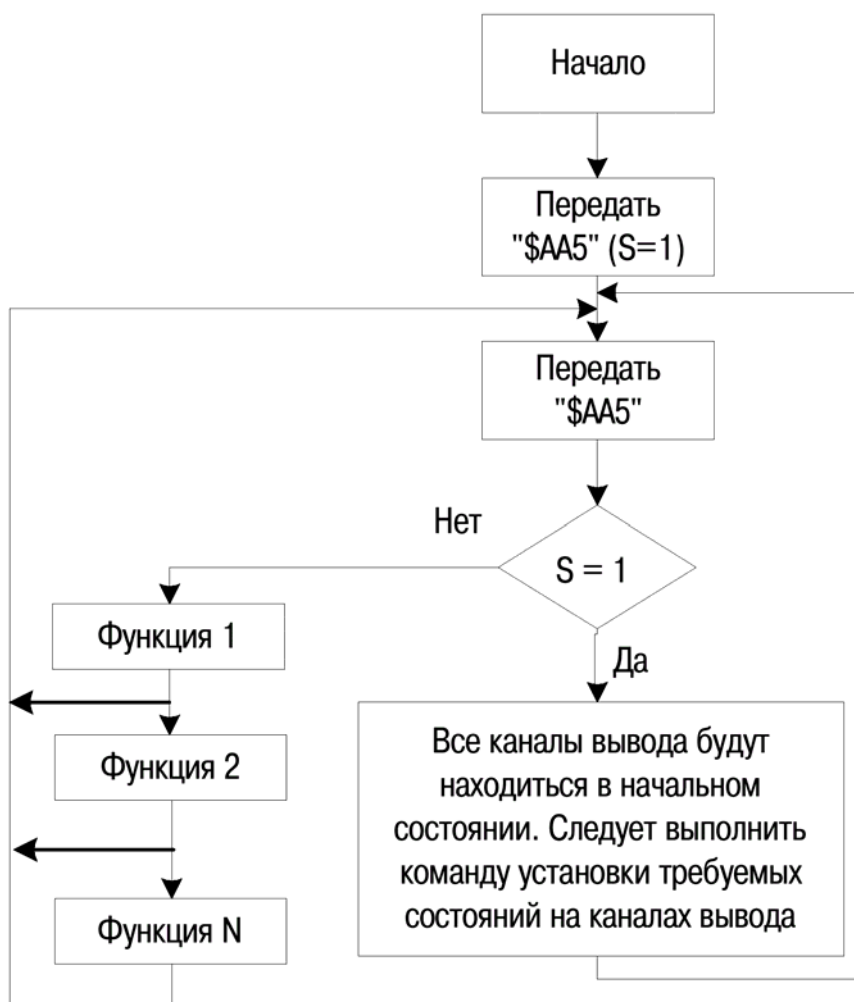


Рис. 6

4.3.3. СТАТУС МОДУЛЯ

Перезапуск модуля внутренним аппаратным сторожевым таймером:

- статус модуля не изменяется;
- разрешен прием команды от управляющего компьютера .

Срабатывание сторожевого таймера ведущего устройства:

- на линиях модуля устанавливаются безопасные значения сигналов;
- статус модуля принимает значение 04 (определяется по команде ~AA0);
- игнорируются все команды управляющего контроллера (компьютера) по установке нового состояния выходных сигналов до тех пор, пока статус модуля не будет сброшен в ноль командой ~AA1.

4.4. ИСПОЛЬЗОВАНИЕ СИМВОЛОВ КИРИЛЛИЦЫ ПРИ ВЫВОДЕ ТЕКСТОВОЙ ИНФОРМАЦИИ НА LCD ДИСПЛЕЙ

Для вывода русского текста на буквенно-цифровой LCD дисплей используются собственные символы знакогенератора, для вывода украинского текста используется метод перекодировки совместно с методом перепрограммирования символов пользователя в оперативной памяти LCD дисплея. **На данный момент в пользовательских ячейках памяти контроллера знакогенератора LCD дисплея находятся украинские символы ЄєІі.**

Метод перепрограммирования символов пользователя в оперативной памяти LCD дисплея (CGRAM) позволяет задавать произвольные начертания восьми символов. Это дает возможность отображать нестандартные символы и буквы.

Для отображения нестандартных символов в контроллере пульта оператора предусмотрена команда \$AAGNLPPPPPPPPPPPPPPPP, которая задает начертание **одного из восьми** символов пользователя (5x8 точек) и сохраняет его в EEPROM контроллера (при старте модуля восемь символов пользователя из EEPROM заносятся в оперативную память знакогенератора LCD дисплея CGRAM).

5. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

Обмен данными с модулем 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 5.1. Набор команд

Посылка	Ответ	Описание
%AANN0ZCCFF	!NN	Установка параметров конфигурации модуля
\$AA2	!AA0ZBBFF	Чтение конфигурации
\$AA5	!AAS	Чтение статуса перезагрузки модуля
\$AASC	!AA	Вкл./откл. дисплея и курсора
\$AAGNLPPPPPPPPPPPPPPPP	!AA	Задание начертания символа пользователя
\$AALNC	!AA	Замена одного символа пользователя
\$AALCCCC	!AA	Замена первых четырех символов пользователя
\$AAT	!AA	Очистка LCD дисплея
\$AATYXX(str)	!AA	Вывод на LCD дисплей
\$AAIV	!AA	Включение/выключение подсветки LCD дисплея
\$AAPYXX	!AA	Установка позиции курсора
\$AAF	!AAVVVV	Чтение номера версии программы
\$AAM	!AANMICLCD	Чтение имени модуля
~**	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

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

$$checksum = sum \& 0xFF = 0x21C \& 0xFF = 0x1C$$

$$ASCII 'I' = 0x31$$

$$ASCII 'C' = 0x43$$

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 <i>(в режиме инициализации адрес модуля всегда равен 00)</i>
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. Пользователь должен отслеживать статус перезагрузки 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)

5.3. КОМАНДЫ УПРАВЛЕНИЯ LCD ИНДИКАТОРОМ МОДУЛЯ

5.3.1. \$AASC

- Назначение: Включение/отключение дисплея и курсора

- Описание: Служит для управления включением и выключением LCD дисплея, установки параметров курсора.

- Синтаксис: \$AASC

- \$ - разделительный символ
- AA - шестнадцатеричный адрес модуля в диапазоне от 00 до FF
- C - число hex (0..F)
- бит C.0 = 1/0 - включить/отключить мигающий курсор
- бит C.1 = 1/0 - включить/отключить статический курсор
- бит C.2 = 1/0 - включить/отключить дисплей
- бит C.3 = 1/0 - сохранить/не сохранять установки в EEPROM
- [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: на модуле с номером 01 включить дисплей, включить мигающий и отключить статический курсор, сохранить установки в EEPROM:

команда - \$01SD(cr)
ответ - !01(cr)

Bit C.7	bit C.6	bit C.5	bit C.4	bit C.3	bit C.2	bit C.1	bit C.0	hex
X	X	X	X	1	1	0	1	D



ВНИМАНИЕ!!!

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

5.3.2. \$AAGNLPPPPPPPPPPPPPPPP

- Назначение: Задание начертания символа пользователя
- Описание: Задает начертание одного из восьми символов пользователя (5x8 точек) с сохранением в EEPROM (при старте модуля восемь символов пользователя из EEPROM заносятся в оперативную память знакогенератора LCD дисплея).

- Синтаксис: \$AAGNLPPPPPPPPPPPPPPPP(cr)

\$	-	разделительный символ
AA	-	шестнадцатеричный адрес модуля в диапазоне от 00 до FF
N	-	номер задаваемого символа в EEPROM (0...7)
L	-	ASCII символ, начертание которого задается
PP..PP	-	восемь байт (по два hex символа на байт), представляющих восемь строк начертания символа, начиная с верхней строки
[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:

В модуль с номером 01 в ячейку 0 EEPROM пользовательских символов занести начертание буквы кириллицы 'Ж' (код ASCII 0x86):

команда	-	\$01G0Ж15150E040E151500(cr)
ответ	-	!01(cr)

PP	b7	b6	b5	b4	b3	b2	B1	b0	hex
0	-	-	-	1	0	1	0	1	15
1	-	-	-	1	0	1	0	1	15
2	-	-	-	0	1	1	1	0	0E
3	-	-	-	0	0	1		0	04
4	-	-	-	0	1	1	1	0	0E
5	-	-	-	1	0	1	0	1	15
6	-	-	-	1	0	1	0	1	15
7	-	-	-	0	0	0	0	0	00

5.3.3. \$AALNC

- **Назначение:** замена одного из восьми отображаемых пользовательских символов в оперативной памяти знакогенератора LCD дисплея на символ, выбираемый из predetermined ряда украинских букв.
- **Посылка :** \$AALNC(cr)

\$	-	разделительный символ
AA	-	адрес модуля от 00 до FF
N	-	номер заменяемого символа (0..7)
C	-	новый символ (*)
[chk]	-	два hex символа байта контрольной суммы (если к.с. включена)
(cr)	-	0x0D (завершение команды)

* новый символ должен быть выбран из следующего ряда украинских символов: 'ЄĖİ'

- **Ответ :**

!AA[chk](cr)	⇒	команда выполнена успешно
?AA[chk](cr)	⇒	ошибка в команде
нет ответа	⇒	ошибка коммуникации, либо неверный адрес

!(?)	-	разделительный символ
AA	-	адрес модуля от 00 до FF
[chk]	-	два hex символа байта контрольной суммы (если к.с. включена)
(cr)	-	0x0D (завершение команды)

- **Пример :**
В модуле с номером 01 заменить отображаемый пользовательский символ с номером 5 на украинский символ 'İ'.

команда	-	\$01L5İ(cr)
ответ	-	!01(cr)

5.3.4. \$AALCCCC

- **Назначение:** замена первых четырех отображаемых пользовательских символа в оперативной памяти знакогенератора LCD дисплея на символы, выбираемые из predetermined ряда украинских букв.
- **Посылка :** \$AALCCCC(cr)

\$	-	разделительный символ
AA	-	адрес модуля от 00 до FF
C..C	-	четыре новых символа (*)
[chk]	-	два hex символа байта контрольной суммы (если к.с. включена)
(cr)	-	0x0D (завершение команды)

* новые символы должны быть выбраны из следующего ряда украинских символов: 'ЄĖİİ'

- **Ответ :**

!AA[chk](cr)	⇒	команда выполнена успешно
?AA[chk](cr)	⇒	ошибка в команде
нет ответа	⇒	ошибка коммуникации, либо неверный адрес

!(?)	-	разделительный символ
AA	-	адрес модуля от 00 до FF
[chk]	-	два hex символа байта контрольной суммы (если к.с. включена)
(cr)	-	0x0D (завершение команды)

- **Пример :**
В модуле с номером 01 заменить первые четыре отображаемые пользовательские символа на украинские символы 'ЄĖİİ':

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

5.3.5. \$AAT

- Назначение: Очистка LCD дисплея.
- Синтаксис: \$AAT[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	⇒	ошибка коммуникации, 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:

Очистить LCD дисплей модуля HMI с номером 01:

команда	-	\$01T(cr)
ответ	-	!01(cr)

5.3.6. \$AATYXX(str)

- Назначение: Вывод текстовой информации на буквенно-цифровой LCD дисплей.
- Синтаксис: \$AATYXX(str)[chk](cr)

\$	-	разделительный символ
AA	-	шестнадцатеричный адрес модуля в диапазоне от 00 до FF
Y	-	определяет строку LCD дисплея для вывода информации (0..3)
XX	-	определяет позицию в строке для вывода информации (0..40)
(str)	-	выводимая текстовая информация (*)
[chk]	-	двухсимвольное шестнадцатеричное значение контрольной суммы (используется, если контрольная сумма разрешена)
(cr)	-	символ возврата каретки (0x0D), используемый в качестве признака конца команды
- * параметр str может содержать до 40 символов из ряда 'прошитых' в знакогенераторе LCD дисплея, пользовательских символов и эмулируемых модулем (см. Раздел 4).
- 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:

На модуль с номером 01 вывести текст 'ПУЛЬТ ОПЕРАТОРА' в строчке дисплея 0 начиная с позиции 1 и 'HMI-LCD' в строчке дисплея 1 начиная с позиции 0:

команда	-	\$01T001ПУЛЬТ ОПЕРАТОРА(cr)
ответ	-	!01(cr)
команда	-	\$01T100HMI-LCD(cr)
ответ	-	!01(cr)

5.3.7. \$AAIV

- Назначение: Включение/выключение подсветки LCD дисплея.

▪ Синтаксис: \$AAIV[chk](cr)

\$	-	разделительный символ
AA	-	шестнадцатеричный адрес модуля в диапазоне от 00 до FF
V	-	0 – включить подсветку, 1 – выключить подсветку
[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:

Выключить подсветку LCD дисплея в модуле с номером 01:

команда	-	\$01I1(cr)
ответ	-	!01(cr)

Включить подсветку LCD дисплея в модуле с номером 01:

команда	-	\$01I0(cr)
ответ	-	!01(cr)

5.3.8. \$AAPYXX

- Назначение: Установка курсора в заданную позицию LCD дисплея.

▪ Синтаксис: \$AAPYXX[chk](cr)

\$	-	разделительный символ
AA	-	шестнадцатеричный адрес модуля в диапазоне от 00 до FF
Y	-	определяет строку LCD дисплея для установки курсора (0..3)

XX	-	определяет позицию (знакоместо) в строке для курсора (0..39)
[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:

На экране дисплея модуля с номером 01 установить курсор на нулевую строку дисплея в позицию 4:

команда	-	\$01P004(cr)
ответ	-	!01(cr)

5.4. КОМАНДЫ ОБЩЕГО НАЗНАЧЕНИЯ

5.4.1. \$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	-	!01B1.7-RUS(cr)

5.4.2. \$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:
 - !AAMICLCD[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 | - | !01HMICLCD(cr) |

5.5. КОМАНДЫ ПОВЫШЕНИЯ НАДЕЖНОСТИ СИСТЕМЫ

5.5.1. \sim^{**}

- Назначение: Управляющее устройство в норме
- Описание: Сообщает всем модулям системы о том, что управляющее устройство (контроллер, компьютер) работает в штатном режиме. Дополнительная информация содержится в разделе 4 настоящего руководства.
- 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

5.5.2. ~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. При таком значении статуса все команды установки состояния выходных каналов модуля будут игнорироваться. Дополнительная информация содержится в пункте 4.2 настоящего руководства.
- 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:

Считать статус модуля с номером 01.

команда	-	~010(cr)
ответ	-	!0100(cr)

5.5.3. ~AA1

- Назначение: Сброс статуса модуля.
- Описание: Производит сброс адресуемого модуля. Если статус модуля равен 04, то все команды установки состояния выходных каналов будут игнорироваться, поэтому прежде чем продолжить работу с модулем пользователь должен осуществить сброс статуса модуля и убедиться, что статус модуля равен 00. Дополнительная информация содержится в пункте 4.2 настоящего руководства.
- 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)

5.5.4. ~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 ~AA3ETT. Дополнительная информация содержится в пункте 4.2 настоящего руководства.

▪ 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

▪ <u>Ответ:</u>	!AATT[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)

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

5.5.5. ~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. Дополнительная информация содержится в пункте 4.2 настоящего руководства.

- **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:**

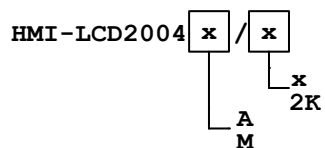
!AA[chk](cr)	⇒	команда выполнена успешно
?AA[chk](cr)	⇒	ошибка в команде
нет ответа	⇒	синтаксическая или коммуникационная ошибка либо неверный адрес модуля

!	-	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**Базовая версия пульта оператора**

HMI-LCD2004A	Индикатор 4 строки по 20 символов с светодиодной подсветкой, Н симв=9,6мм, синий фон подсветки, белые символы, RS-485/232, ASCII, щитовое исполнение корпуса, защита передней панели IP54
---------------------	---

Опции для заказа при отклонении от базовой версии

A	поставка с системой команд в ASCII формате (протокол DCON)
M	поставка с протоколом обмена MODBUS RTU SLAVE
/2K	поставка с мембранной клавиатурой 2 клавиши с "рисунком" заказчика