

Протокол обмена данными с УСПД LiGO-7188

Обмен данными осуществляется в рамках стандарта Modbus RTU. Поддерживаются стандартный и расширенный набор функций. Их стандартных функций поддерживаются следующие:

Код	Название	Действие
01	READ COIL STATUS	Получение текущего состояния (ON/OFF) дискретных выходов
02	READ INPUT STATUS	Получение текущего состояния (ON/OFF) дискретных входов.
03	READ HOLDING REGISTERS	Получение текущего состояния счетных входов (накопительные параметры)
04	READ INPUT REGISTERS	Получение текущего значения одного или нескольких аналоговых входов
05	FORCE SINGLE COIL	Изменение дискретного выхода в состояние ON или OFF.

Контроллер не отвечает на запросы с неверным CRC или не ему адресованные (поле «Номер» не совпадает с заданным в его конфигурации адресом).

Отрицательный ответ (инвертированный старший бит номера функции) может быть выдан контроллером в случае неизвестной функции, либо ошибочного адреса (для стандартного протокола Modbus).

Пакет стандартного протокола Modbus выглядит следующим образом:

Номер	Функция	Старший байт адреса	Младший байт адреса	Старший байт числа ячеек	Младший байт числа ячеек	Младший байт CRC	Старший байт CRC
-------	---------	---------------------	---------------------	--------------------------	--------------------------	------------------	------------------

Адресация начинается с 0. Для функций READ HOLDING REGISTERS и READ INPUT REGISTERS при передаче значений типа FLOAT и LONG используется два регистра. Например запрос 8 параметров типа FLOAT выглядит следующим образом.

1	4	0	0	0	16	Младший байт CRC	Старший байт CRC
---	---	---	---	---	----	------------------	------------------

При конфигурации контроллера можно указать тип величины. Возможны варианты WORD (целое 2 байта), LONG (целое 4 байта) и FLOAT (4 байта с плавающей запятой). Так же можно для вариантов WORD и LONG указать число десятичных знаков. В результате можно передавать дробные значения целыми числами, разбирая значения на «верхнем» уровне.

Расширенный протокол Modbus:

71. Накопительные итоги по нескольким каналам

Запрос:

Номер	71	Тип	Nbegin	Nend	LCRC	HCRC
-------	----	-----	--------	------	------	------

Тип – байт "С"

Nbegin (2 байта), Nend (2 байта) – с какого номера канала по какой включительно.

Ответ:

Номер	71	Тип	Nbegin	Nend	S1 _{Nbegin}	S2 _{Nbegin}	R _{Nbegin}	...	S1 _{Nend}	S2 _{Nend}	R _{Nend}	LCRC	HCRC
-------	----	-----	--------	------	----------------------	----------------------	---------------------	-----	--------------------	--------------------	-------------------	------	------

$S1_n$ – первый байт статуса результата по каналу n
 $S2_n$ – второй байт статуса результата по каналу n
 R_n – значение по каналу n в формате real (6 байт)

72. Чтение привязанных ко времени с признаком сезона накопительных итогов

Запрос:

Номер	72	Тип	Nbegin	Nend	LCRC	HCRC
-------	----	-----	--------	------	------	------

Тип – байт “С”

Nbegin (2 байта), Nend (2 байта) – с какого номера канала по какой включительно

Ответ:

Номер	72	Тип	Nbegin	Nend	T_{Nbegin}	R_{Nbegin}	...	T_{Nend}	R_{Nend}	LCRC	HCRC
-------	----	-----	--------	------	--------------	--------------	-----	------------	------------	------	------

T_n – время получения значения по каналу n в формате T3 (10 байт)

R_n – значение по каналу n в формате real (6 байт)

Если контроллер не в состоянии выдать запрошенное значение по одному из каналов, по нему выставляются нули и в R , и в T

76. Версия ПО

Запрос:

Номер	76	LCRC	HCRC
-------	----	------	------

Ответ:

Номер	76	H	L	Время (7байт)	LCRC	HCRC
-------	----	---	---	---------------	------	------

H – major номер версии

L – minor номер версии

Время – время записи этой версии в прибор в формате T1

88. Инициализация системы

Запрос:

Номер	88	Подфункция (1 байт)	LCRC	HCRC
-------	----	---------------------	------	------

Ответ:

Номер	88	LCRC	HCRC
-------	----	------	------

Номер подфункции задает действия перед инициализацией УСПД:

0 - только перезагрузка

1 - очистка всех архивов

95. Состояние дискретных входов и выходов

Запрос:

Номер	95	Тип	Nbeg	Nend	LCRC	HCRC
-------	----	-----	------	------	------	------

Тип – тип канала (“R”, “K”, “L”, “E”)

Nbeg, Nend – с какого по какой номер канала (1-255) включительно

Ответ:

Номер	95	Sbeg	...	Send	LCRC	HCRC
-------	----	------	-----	------	------	------

S_n – байты состояния по каналам. Интерпретация байта зависит от типа канала и описана в приложении

110. Выдать текущее время УСПД с признаком текущего сезона

Запрос:

Номер	110	LCRC	HCRC
-------	-----	------	------

Ответ:

Номер	110	Время в формате T3 (10 байт)	LCRC	HCRC
-------	-----	------------------------------	------	------

Точность выдаваемого в ответе времени определяется конфигурацией контроллера. На малых скоростях обмена, а также при наличии в линии связи источников дополнительных задержек (например, асинхронных модемов) рекомендуется также учитывать, что выдаваемое время действительно на момент начала ответа контроллера.

111. Установка времени УСПД

Запрос:

Номер	111	Время в формате T3 (10байт)	Len	Psw	LCRC	HCRC
-------	-----	-----------------------------	-----	-----	------	------

Команда не позволяет изменить текущий сезон – байт season формата времени T3 определяет лишь способ интерпретации контроллером параметров запроса, даты же сезонных переходов определяются конфигурацией контроллера.

Len — длина пароля

Psw — пароль

Ответ:

Номер	111	LCRC	HCRC
-------	-----	------	------

Команда может быть отвергнута (в ответе номер функции возвратится с инвертированным старшим битом), если уровень доступа, определяемый паролем, недостаточен для запрошенной коррекции времени или параметры запроса неверны.

118. Мгновенные значения по нескольким каналам

Запрос:

Номер	118	Тип	Nbegin	Nend	интервал	LCRC	HCRC
-------	-----	-----	--------	------	----------	------	------

Тип – байт “G” для аналоговых входов УСО

Nbegin (1 байт), Nend (1 байт) – с какого номера канала по какой включительно

интервал – 0 для короткого, 1 для основного интервала хранения, 2 для суток, 3 для месяца

Ответ:

Номер	118	Тип	Nbegin	Nend	интервал	S1 _{Nbegin}	S2 _{Nbegin}	R _{Nbegin}	...	S1 _{Nend}	S2 _{Nend}	R _{Nend}	LCRC	HCRC
-------	-----	-----	--------	------	----------	----------------------	----------------------	---------------------	-----	--------------------	--------------------	-------------------	------	------

S1_n – первый байт статуса результата по каналу n

S2_n – второй байт статуса результата по каналу n

R_n – значение по каналу n в формате real (6 байт)

125. Архивные значения с полным статусом по нескольким каналам за интервал времени

Запрос:

Номер	125	Тип	Nbeg	Nend	Время (10 байт)	Интервал	ID_device	LCRC	HCRC
-------	-----	-----	------	------	-----------------	----------	-----------	------	------

Тип – байт “G”, “B” аналоговых входов УСО, КВНА соответственно

Nbeg (1 байт), Nend (1 байт) – с какого номера канала по какой включительно.

Время – любая временная точка внутри выбранного интервала в формате T3 (10 байт)

интервал – 0 для короткого, 1 для основного интервала хранения, 2 для суток, 3 для месяца, 4 для года

ID_Device — порядковый номер устройства в конфигурации (нумерация с 0)

Ответ:

Номер	125	Тип	Nbeg	Nend	Время (10 байт)	Интервал	S1 _{Nbeg}	S2 _{Nbeg}	R _{Nbeg}	...	S1 _{Nend}	S2 _{Nend}	R _{Nend}	LCRC	HCRC
-------	-----	-----	------	------	-----------------	----------	--------------------	--------------------	-------------------	-----	--------------------	--------------------	-------------------	------	------

R_n – значение по каналу n в формате real (6 байт)

S1_n – первый байт статуса результата по каналу n.

S2_n – второй байт статуса результата по каналу n. Значения отдельных битов статуса даны в приложении 1.

Архивные значения по каналам (средние или накопленные за интервал – в зависимости от типа

канала) сопровождаются двухбайтовым (в ответ на некоторые запросы передается только первый байт) статусом. Статус представляет собой набор битовых флагов, каждый из которых может оказаться ненулевым при определенных условиях. Ниже перечислены эти условия

Первый байт статуса:

Маска	Условие ненулевого значения бита
1	выключенное питание, отказ платы преобразователя
16 (10h)	первый интервал после инициализации архивов
64 (40h)	данные пока не готовы, опрос следует повторить позже
128 (80h)	канал не описан (некорректно описан) в конфигурации

Второй байт статуса не используется.

Дискретные входы и выходы

Бит 0 – состояние входного канала после переключения (при правильном конфигурировании 1 отвечает замкнутому контакту, 0 – разомкнутому).

Бит 7 – канал отсутствует (1) / канал имеется (0)

Передаваемое в запросах и ответах время в общем случае представляет собой набор следующих байтовых полей

ms1	ms2	sec	min	hour	day	month	year1	year2	season
-----	-----	-----	-----	------	-----	-------	-------	-------	--------

Байт ms1 – младший байт значения миллисекунд

Байт ms2 – два младших бита являются старшими битами значения миллисекунд, шесть старших бит зарезервированы.

Байт sec – шесть младших бит дают значение секунд (0-59)

Байт min – шесть младших бит дают значение минут (0-59)

Байт hour – пять младших бит дают значение часов (0-23)

Байт day – пять младших бит дают день месяца (1-31)

Байт month – четыре младших бита дают значение месяца (1-12)

Байт year1 – младший байт значения года от РХ

Байт year2 – четыре младших бита являются старшими битами значения года

Байт season – младший бит определяет текущий сезон (1-«летнее» время)

Формат времени T3 использует все десять байтовых полей.

Формат T2 – 9 байт без поля season (подразумевается астрономическое время без учета «летнего» сдвига)

Формат T1 – 7 байт без полей ms1, ms2 и season

Процедура вычисления CRC на языке C

```

/* CRC16 Table High byte */
static unsigned char CRC16Hi[] = {
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
}

```



```
unsigned char CRCHi ; /* high byte of CRC initialized */
unsigned char CRCLo ; /* low byte of CRC initialized */
unsigned GetCRC16(unsigned char *puchMsg, int DataLen)
{
    unsigned Index ; /* will index into CRC16 lookup table */

    CRCHi = 0xFF ; /* high byte of CRC16 initialized */
    CRCLo = 0xFF ; /* low byte of CRC16 initialized */
    while (DataLen--){
        Index = CRCHi ^ *puchMsg++ ; /* calculate the CRC16 */

        CRCHi = CRCLo ^ CRC16Hi[Index] ;
        CRCLo = CRC16Lo[Index] ;
    }
    return ((unsigned)CRCHi << 8 | CRCLo) ;
}
```