

ООО «Технологии Радиосвязи»



**Технологии
Радиосвязи**

УТВЕРЖДЕН

ТИШЖ.468157.168-01 Д01-ЛУ

Преобразователь частоты «вверх» 70МГц/L
Протокол информационно-логического взаимодействия
ТИШЖ.468157.168-01 Д01

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Справ. №	Перв. примен.
	ТИШЖ.468157.168-01

Содержание

Лист

1. Описание протокола.....	3
2. Структура посылки.....	3
3. Типы и структура запросов.....	4
3.1. Команда на чтение регистра.....	4
3.2. Ответ на команду чтения регистра.....	4
3.3. Команда на запись регистра.....	5
3.4. Ответ на команду записи.....	5
4. Сообщения об ошибках обмена.....	6
5. Регистры БПЧ.....	7
6. Расчет контрольной суммы.....	12

Инв. № подл.	Взам. инв. №	Инв. № дубл.	Подп. и дата

ТИШЖ.468157.168-01 Д01										
Изм	Лист	№ докум.	Подп.	Дата	Преобразователь частоты «вверх» 70МГц/L Протокол информационно-логического взаимодействия			Лит.	Лист	Листов
Разраб.		Орлов								
Пров.		Харченко								
Т.контр.		Званцугов								
Н.контр.		Фадеев								
Утв.		-								

Данный документ определяет протокол обмена данными по интерфейсу RS-485 между блоком преобразователя частоты «вверх» 70МГц/L (БПЧ) и устройством управления (УУ).

1. Описание протокола

Физический интерфейс: RS-485 двухпроводной.

Организация сети: ведущий – УУ, ведомый – БПЧ.

Инициировать передачу может только ведущий. Ведомый отвечает на запрос (если команда в запросе предполагает выдачу ответа).

Битовая структура данных: 8N2 (8 бит данных, без бита четности, два стоповых бита).

Скорость обмена: программируется. Возможные значения скорости передачи (бит/сек): 1200, 1800, 2400, 4800, 9600, 19200, 38400, 57600, 115200, 230400, 460800, 576000, 921600.

Скорость обмена 115200 бит/сек является скоростью по умолчанию (заводские установки).

Адресация:

Адреса БПЧ программируются. Допустимые значения адреса 0x01-0xFF.

Адрес 0xFF является циркулярным и может применяться только в пакете от УУ. Пакеты с адресом 0xFF, воспринимаются всеми БПЧ.

Адрес 0 является запрещенным для БПЧ.

2. Структура посылки

Структура посылки передаваемой в прибор или принимаемой из прибора содержит следующие поля:

START	ADR_1	ADR_2	DATA	CRC	STOP
2 байта	1 байт	1 байт	N байт	2 байта	2 байта

Описание полей:

Поле START - флаг начала пакета. Содержит два байта 0xFE 0xFE.

Поле ADR_1 – адрес отправителя. Содержит 1 байт.

Поле ADR_2 – адрес получателя. Содержит 1 байт.

Поле DATA – данные пакета. Размер поля определяется типом запроса.

Инов.№подл.	Подп. и дата	Взам. инв.№	Инов.№дубл.	Подп. и дата	Инов.№докум.	Изм.	Лист	№ докум.	Подпись	Дата	ТИШЖ.468157.168-01 Д01		Лист
													3

Поле CRC – контрольная сумма по полям START, ADR_1, ADR_2, DATA пакета.

Алгоритм вычисления контрольной суммы приведен в разделе 6.

Поле STOP - флаг конца пакета. Содержит два байта 0xFC 0xFC.

Примечание 1: Если в полях ADR_1, ADR_2, DATA, CRC встречается байт 0xFE или 0xFC, то после него добавляется байт со значением равным 0x00. Соответственно, при приеме пакета этот байт из пакета изымается (байт-стаффинг).

Примечание 2: При передаче байт-стаффинг используется после расчета контрольной суммы. При приеме – сначала байт-стаффинг, потом расчет контрольной суммы.

3.ТИПЫ И СТРУКТУРА ЗАПРОСОВ (поле DATA)

3.1. Команда на чтение регистра

Команда «Чтение регистра»	Номер регистра
0x03	0xНННН
1 байт	2 байта

Где: 0x03 – код команды на чтение регистра

0xНННН – номер регистра (адресуемое пространство регистров 0x0000-0xFFFF)

3.2. Ответ на команду чтения регистра

Команда «Ответ на чтение регистра»	Номер регистра	Данные из регистра
0x04	0xНННН	Data_from_Registr
1 байт	2 байта	N байт

Где: 0x04 – код команды ответ на чтение регистра

0xНННН – номер регистра

Data_from_Registr - данные, считанные из регистра. Размер данных определяется номером регистра и может составлять до 255 байт.

Изн.№подл.	Подп. и дата
Взам. инв.№	Изн.№дубл.
Подп. и дата	Подп. и дата

Изн.№подл.	Изн.№дубл.	Взам. инв.№	Подп. и дата	Изн.№подл.
Изм.	Лист	№ докум.	Подпись	Дата

ТИШЖ.468157.168-01 Д01

Лист
4

3.3. Команда на запись регистра

Команда «Запись регистра»	Номер регистра	Данные в регистр
0x05	0xНННН	Data_In_Registr
1 байт	2 байта	N байт

Где: 0x05 – код команды на запись регистра

0xНННН – номер регистра

Data_In_Registr – данные на запись в регистр (до 255 байт)

3.4. Ответ на команду записи

Команда «Ответ на запись регистра»	Номер регистра	Данные из регистра
0x06	0xНННН	Data_from_Registr
1 байт	2 байта	N байт

Где: 0x06 – код команды ответ на запись регистра

0xНННН – номер регистра

Data_from_Registr - данные считанные из регистра после его записи (до 255 байт).

Примечание: Порядок следования байтов – младший бат передается первым.

Инд. № подл.	Подп. и дата	Инд. № дубл.	Подп. и дата	Взам. инв. №	Инд. № докум.	ТИШЖ.468157.168-01 Д01					Лист				
											5				
						Изм.	Лист	№ докум.	Подпись	Дата					

4. Сообщения об ошибках обмена

При ошибках обмена БПЧ высылает пакет со следующей структурой поля DATA

Команда «Признак ошибки»	Код ошибки
0x0A	0xНННН
1 байт	2 байта

Где: 0x0A – признак ошибки

0xНННН – код ошибки

Перечень кодов ошибок

Код ошибки	Что означает
0x02	Чтение регистра невозможно, либо регистр не найден
0x03	Запись в регистр невозможна, либо регистр не найден
0x04	Неудачная попытка чтения регистра
0x05	Неудачная попытка записи регистра
0x06	Неверное кол-во байтов в запросе в поле DATA при записи регистра

Инв.№подл.	Подп. и дата	Взам. инв.№	Инв.№дубл.	Подп. и дата	Инв.№докум.	Подпись	Дата	ТИШЖ.468157.168-01 Д01	Лист
									6

5. Регистры БПЧ

Номер, дес	Признак	Описание регистра	Длина, байт
СТАТУСНЫЕ ПАРАМЕТРЫ			
0	R	<p>Регистр состояния БПЧ</p> <p>Байт 0 – общие аварии БПЧ (тип unsigned char) Бит 0 – Флаг суммарной аварии 0 – нет аварии 1 – авария Бит 1 – Авария Flash-памяти Бит 2 – Невалидный пользовательский ключ Бит 3 – Тип преобразователя 0-BDC L->70 1- BUC 70->L Бит 4-7 – зарезервировано</p> <p>Байт 1 – общий статус модуля ПЧ (тип unsigned char) Бит 0 – Флаг суммарной аварии в модуле ПЧ 0–нет аварии 1–авария Бит 1 – Авария «UNLOCK PLL» в модуле ПЧ Бит 2 – Авария «UNLOCK 10MHz» в модуле ПЧ Бит 3 – Авария превышение тока в модуле ПЧ Бит 4 – Авария превышение температуры в модуле ПЧ Бит 5 – Авария отказ датчиков тока и температуры в модуле ПЧ Бит 6 – Источник опорного сигнала 10 МГц 0-Внутренняя 1-Внешняя Бит 7 – Питание ВЧ-модуля в МПЧ 0-выключено 1-включено</p> <p>Байт 2-5 – Температура модуля ПЧ, градусы Цельсия (тип float) NaN-ошибка датчика</p> <p>Байт 6-9 – Ток потребления модуля ПЧ, мА (тип float) NaN-ошибка датчика</p> <p>Байт 10 – Инверсия спектра в БПЧ 0-нет инверсии 1- инверсия (тип unsigned char)</p>	17

Инва.№подл.	Подп. и дата
Взам. инв.№	Инва.№дубл.
Подп. и дата	Подп. и дата

Изм.	Лист	№ докум.	Подпись	Дата
------	------	----------	---------	------

ТИШЖ.468157.168-01 Д01

Лист

7

			<p>Байт 11 – Атенюатор БПЧ Значения 0-60 дБ (тип unsigned char)</p> <p>Байт 12-15 – Входная частота БПЧ, кГц Значения 950000-2150000 (тип unsigned int 4 байта)</p> <p>Байт 16 – Значение аттенюатора тракта демодулятора L-band (тип unsigned char) Значения 0-30 дБ</p>	
	1	R	<p><u>Регистр индикатора БПЧ</u></p> <p>Содержит 48 байтов индикатора БПЧ</p>	48
	2	R	<p><u>Регистр состояния БПЧ+Регистр индикатора БПЧ</u></p> <p>Содержит байты регистра состояния R0 и 48 байтов индикатора БПЧ</p>	R0+48
	3	R/W	<p><u>Регистр кнопок БПЧ</u> (тип unsigned char)</p> <p>0 – кнопка ButtonNULL 1 – кнопка ButtonLeft 2 – кнопка ButtonUP 3 – кнопка ButtonRight 4 – кнопка ButtonDown 5 – кнопка ButtonOK 6 – кнопка ButtonRedit 7 – кнопка ButtonALARM 8 – кнопка ButtonKrest 9 – кнопка ButtonESCAPE 10 – кнопка ButtonAR 11-255 - зарезервировано</p>	1
ПАРАМЕТРЫ УПРАВЛЕНИЯ БПЧ				
	4	R/W	<p>Значение аттенюатора БПЧ</p> <p>Диапазон значений 0-60 дБ (тип unsigned char)</p>	1

Инь.№подл.	Подп. и дата
Взам. инв.№	Инь.№дубл.
Подп. и дата	

Изм.	Лист	№ докум.	Подпись	Дата
------	------	----------	---------	------

ТИШЖ.468157.168-01 Д01

Лист
8

			8 - 500000 9 - 576000 10 – 921600 Тип unsigned char (0-255)	
	44-62	R/W	Зарезервировано	-
	63	R/W	Адрес БПЧ Допустимые значения адреса 0x01-0xFF. Адрес 0xFF является циркулярным. Адрес 0 является запрещенным для БПЧ Тип unsigned char (0-255)	1
	64-78	R/W	Зарезервировано	-
	79	R/W	Байты 0-3 Журнал аварий БПЧ При чтении содержит битовую структуру журнала аварий БПЧ Бит 0- Авария «UNLOCK PLL» ПЧ Бит 1- Общая авария ПЧ Бит 2-Ошибка FLASH-памяти Бит 3-Невалидный ключ При записи в этот регистр любого значения сбрасывает журнал текущих аварии БПЧ Тип unsigned long (4 байта)	4
КОМПЛЕКСНЫЕ РЕГИСТРЫ КОМАНД				
	80 ... 65529	...	Зарезервировано	
	65530	W	Выставить параметры по умолчанию (запись 1 приводит к активации заводских настроек) Тип unsigned char (0-255)	1
	65531	R	Версия ПО Тип string[48]	48
	65532	R	ID-номер контроллера Тип unsigned long	4
	65533	R	Признак валидности пользовательского ключа 0-валиден 1-невалиден Тип unsigned char	1

Инва.№подл.	Подп. и дата
	Инва.№дубл.
Взам. инв.№	Подп. и дата
	Инва.№дубл.
Инва.№подл.	Подп. и дата
	Инва.№дубл.

Изм.	Лист	№ докум.	Подпись	Дата
------	------	----------	---------	------

ТИШЖ.468157.168-01 Д01

Лист

10

	65534	R/W	Пользовательский ключ 0XXXXXXXXX Тип unsigned long	4
	65535	R/W	Регистр перезагрузки БПЧ (запись в этот регистр вызывает перезагрузку БПЧ) Тип unsigned char (0-255)	1

Признак: **R** – только чтение, **W** – только запись, **W/R** – чтение и запись

Инв.№подл.	Подп. и дата	Взам. инв.№	Инв.№дубл.	Подп. и дата						Лист
					ТИШЖ.468157.168-01 Д01					11
					Изм.	Лист	№ докум.	Подпись	Дата	

6. Расчет контрольной суммы

Примеры процедур расчета контрольной суммы по пакету на языке ANSI C приведены ниже.

```

unsigned int crc_chk(unsigned char* data, unsigned char length)
{//расчет контрольной суммы
int j;
unsigned int reg_crc=0xFFFF;
while(length--)
{
reg_crc ^= *data++;
for(j=0;j<8;j++)
{
if(reg_crc & 0x01) reg_crc=(reg_crc>>1) ^ 0xA001;
else reg_crc=reg_crc>>1;
}
}
return reg_crc;
}
    
```

Где: data – принятые данные, length – размер (длина) данных

Примеры процедур расчета контрольной суммы на языке Pascal по пакету приведены ниже.

```

function C485Modbus(unCRC_temp,unData:integer):integer;
//вспомогательная функция
Var LSB:integer;
i:integer;
begin
unCRC_temp:=((unCRC_temp xor unData) or $FF00) and (unCRC_temp or
$FF);
for i:=1 to 8 do begin
LSB:=unCRC_temp and $1;
unCRC_temp:=unCRC_temp shr 1;
if (LSB<>0) then unCRC_temp:=unCRC_temp xor $A001;
end;//for i
C485Modbus:=unCRC_temp;
end;
//=====
function CRC_Modbus(LenDat:integer;DATAsend: array[1..100] of
integer):integer;
//расчет контрольной суммы
Var CRC:word;
i:integer;
begin
CRC:=$FFFF;
for i:=1 to LenDat do CRC:=C485Modbus(CRC,DATAsend[i]);
CRC_Modbus:=CRC;
end;
    
```

Изм.	Лист	№ докум.	Подпись	Дата	Подп. и дата	Инва.№дубл.	Взам. инв.№	Подп. и дата	Инва.№подл.	Итого	Лист

