

Датчики уровня топлива

**СТРЕЛА**



**ИНСТРУКЦИЯ ПО  
НАСТРОЙКЕ ДУТ СТРЕЛА 232 и 485**

**Группа компаний Сапсан**

**[www.skontrol.ru](http://www.skontrol.ru)**

**Челябинск**

10.09.2011

1.	Подключение датчика Стрела D485 к ПК .....	3
2.	Подключение датчика Стрела D232 к ПК .....	4
3.	Установка драйвера .....	6
4.	Подключение датчика к программе DUT CONFIG 2.xx .....	8
5.	Калибровка датчика.....	9
6.	Задание формата выдачи данных .....	11
7.	Специальный случай 1. задание таблицы тарировки для бака прямоугольной формы внутри датчика 12	
8.	Специальный случай 2. задание формулы тарировки бака цилиндрической формы внутри датчика..	13
9.	Описание протокола обмена данных датчиков Стрела D232 (D485) .....	14

## 1. Подключение датчика Стрела D485 к ПК

### **Вариант 1: используя универсальный сервисный адаптер**

Данный способ более предпочтителен – он не требует внешнего источника питания.

1. Подключить датчик (1) в сервисный кабель (2) адаптера USB-RS485 (рис. 1).
2. Подключить сервисный кабель к адаптеру (3).
3. Подключить к адаптеру к usb порту компьютера (5) через USB-кабель (4).

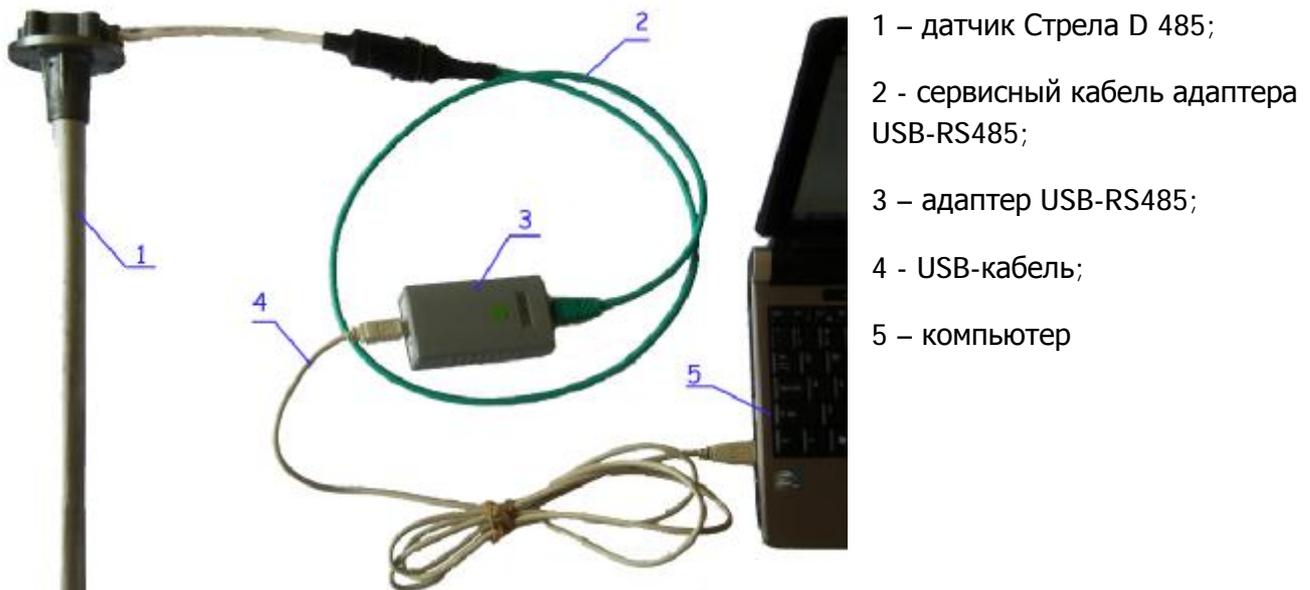


рис. 1 Схема подключения

### **Вариант 2: используя адаптер USB-RS485**

#### **Для работы необходим внешний источник напряжения**



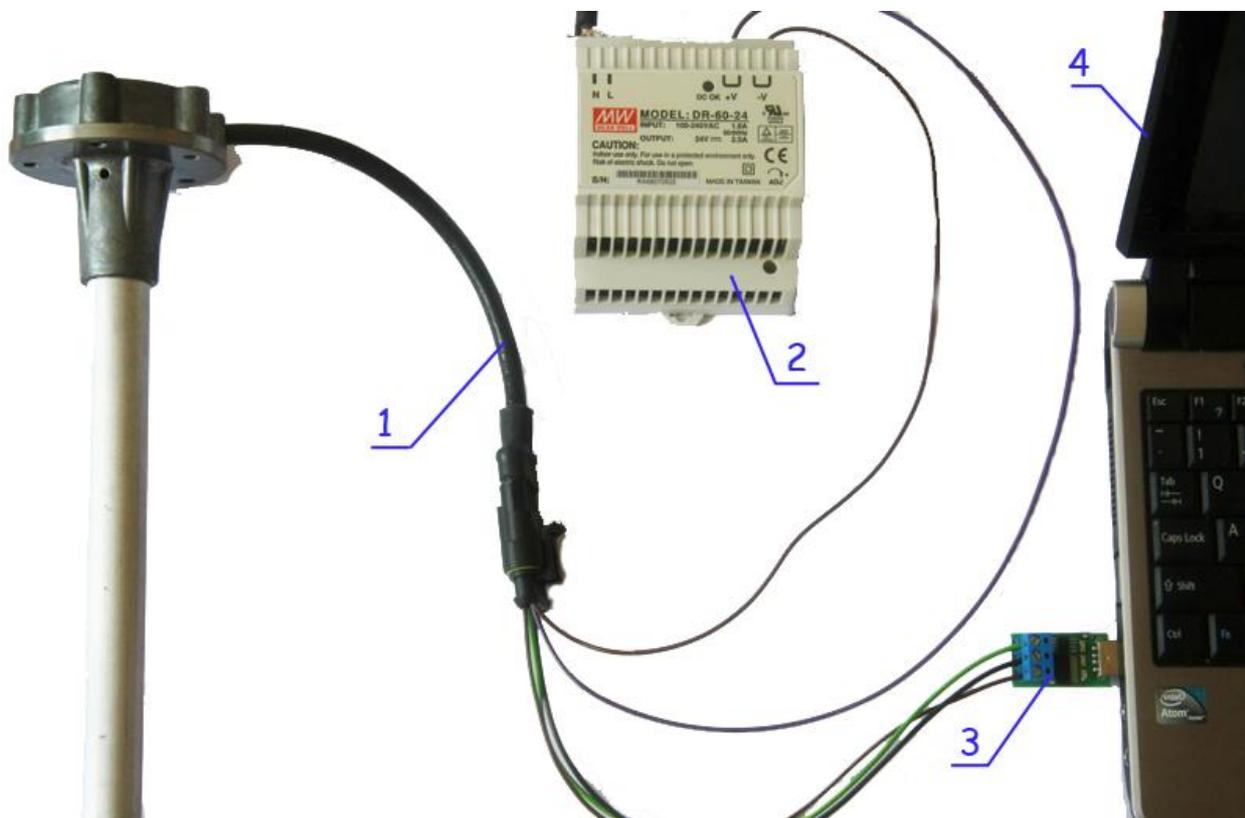
Устройство сопряжения USB-RS485 служит для подключения аппаратов с интерфейсом 485 к USB порту компьютеров. Оно использует стандартные драйверы FTDI с использованием виртуального COM-порта. При подключении адаптера в порт USB будет обнаружено новое устройство. Драйверы для него находятся здесь: [http:// ftdichip.com/Drivers/](http://ftdichip.com/Drivers/)

В случае возникновения конфликтов при установке драйвера смотрите рекомендации Приложения 3.

рис. 2 Внешний вид адаптера USB-RS485

Порядок подключения датчика (см. рис 3):

1. Соединить “минус” питания датчика с “землей” устройства сопряжения USB-RS85 и минусом источника питания.
2. Соединить “плюс” питания датчика к питанию 12 (24) вольт).
3. Соединить выход RS-A датчика с клеммой 1 адаптера.
4. Соединить выход RS-B датчика с клеммой 2 адаптера.



1 – кабель датчика стрела D 485; 2 – внешний источник питания 12 (24) В;  
3 – устройство сопряжения USB-RS485; 4 – компьютер

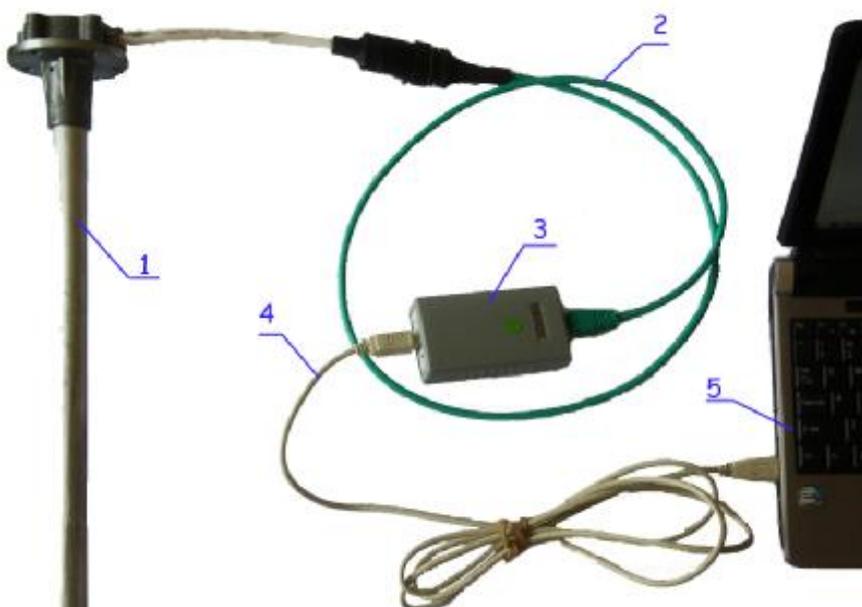
**рис. 3 схема подключения датчика Стрела 485 к компьютеру при помощи устройства сопряжения USB-RS485**

## 2. Подключение датчика Стрела D232 к ПК

### Вариант 1: используя универсальный адаптер USB-R232

Данный способ более предпочтителен – он не требует внешнего источника питания

1. Подключить датчик (1) в сервисный кабель (2) универсального адаптера (рис.11).
2. Подключить сервисный кабель к адаптеру (3).
3. Подключить адаптер к usb порту компьютера (5) через USB-кабель (4).



1 – Датчик стрела D232;  
2 – сервисный кабель универсального адаптера;  
3 – универсальный адаптер;  
4 - USB-кабель;  
5 – компьютер.

**рис. 4 схема подключения датчика Стрела D232 к компьютеру через универсальный сервисный адаптер**

## Вариант 2: используя сервисный кабель USB-RS232

### **Для работы необходим внешний источник напряжения**

1. Подключить датчик (1) (рис.14) в сервисный кабель(2).
2. Подключить сервисный кабель к COM порту ПК либо через дополнительное устройство сопряжения USB-RS232 (5) (рис.15) к USB порту.
3. Соединить “минус” питания кабеля датчика (см. **Ошибка! Источник ссылки не найден.** и) с минусом питания источника напряжения 12(24) В.
4. Подключить “плюс” питания кабеля датчика (см. **Ошибка! Источник ссылки не найден.**) с плюсом питания источника напряжения 12 (24) В.



рис. 5 сервисный кабель RS232

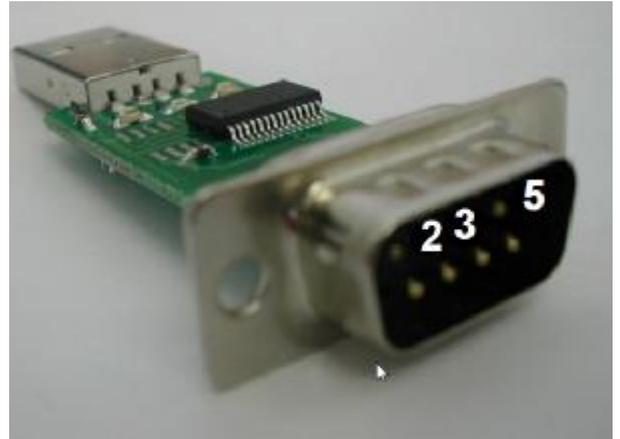
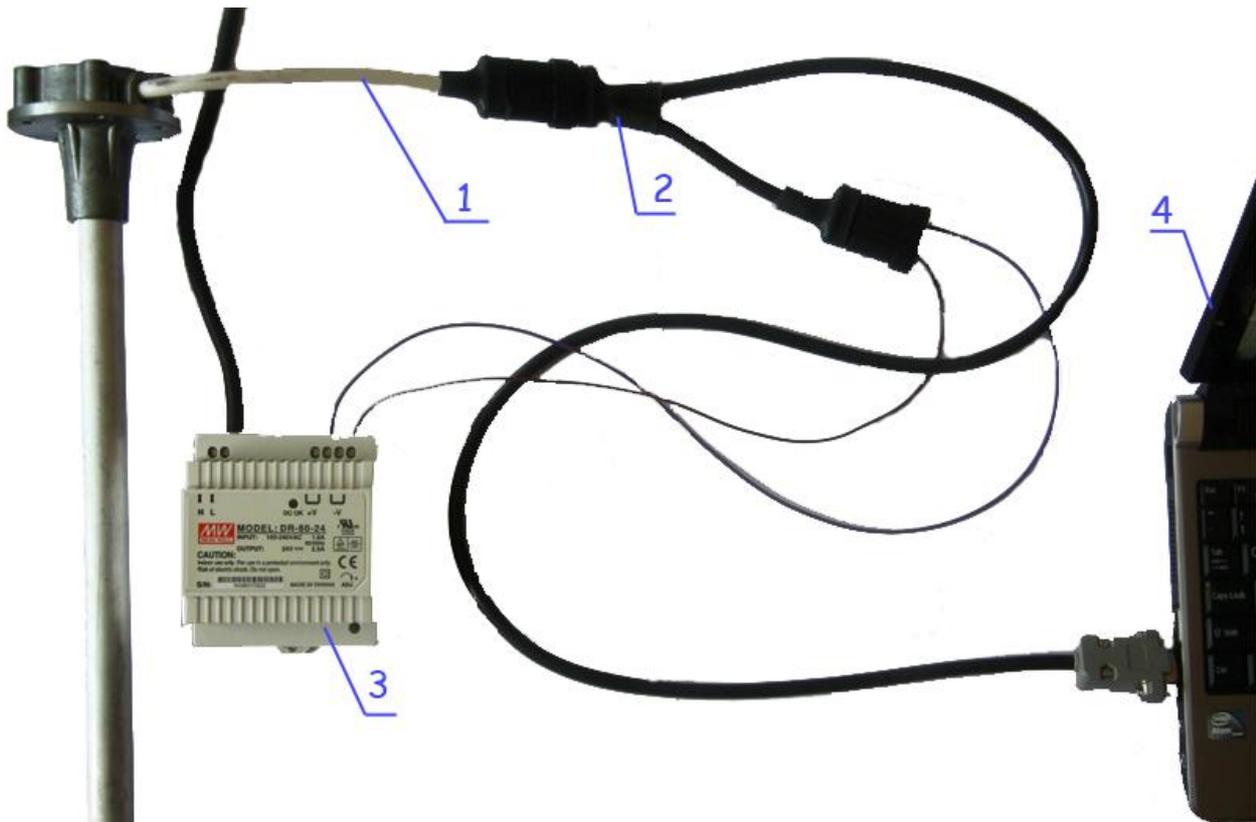


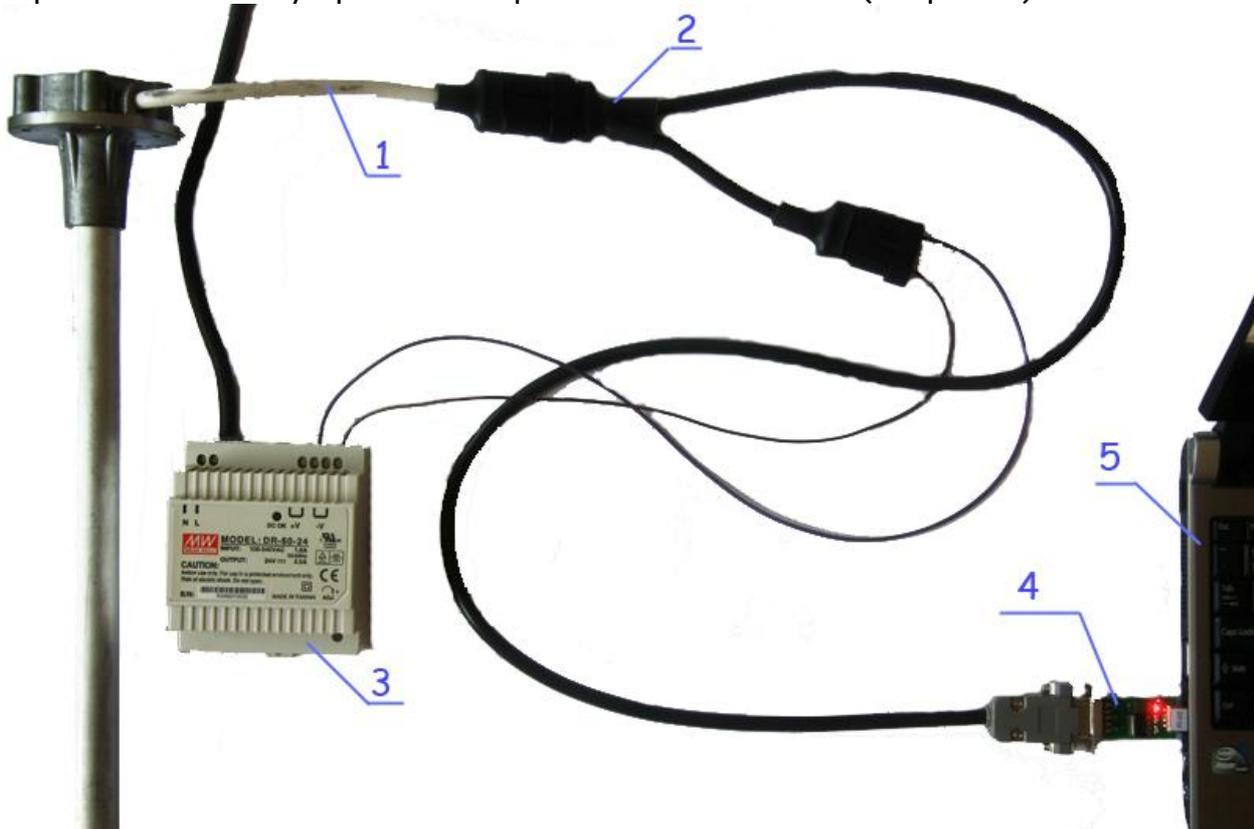
рис. 6 Устройство сопряжения USB-RS232



- 1 – Датчик стрела D 232 2 – сервисный кабель RS-232 3 – внешний источник питания 12 (24) В;  
4 – компьютер;

**рис. 7 схема подключения датчика Стрела 232 к компьютеру при помощи устройства сопряжения USB-RS232**

В случае отсутствия на компьютере com-порта можно использовать для подключения сервисного кабеля устройство сопряжения USB – RS 232 (см. рис 15):



- 1 – Датчик стрела D 232
- 2 – сервисный кабель RS-232
- 3 – внешний источник питания 12 (24) В;
- 4 - Устройство сопряжения USB-RS232
- 5 – компьютер;

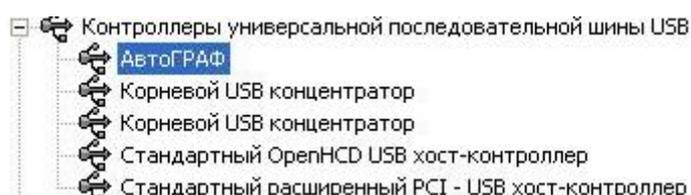
**рис. 8 схема подключения датчика Стрела 232 к компьютеру при помощи устройства сопряжения USB-RS232**

### 3. Установка драйвера

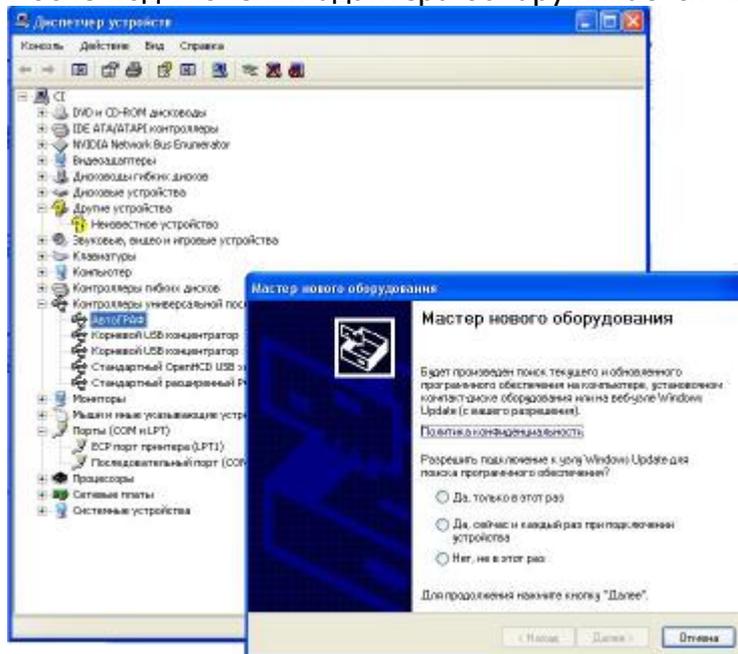
Драйвер USB-RS485 использует стандартные драйверы FTDI, которые могут также использоваться другим оборудованием, которое подключается к ПК. При этом работа с оборудованием через USB-порт может осуществляться в двух режимах: прямом и через виртуальный COM-порт. Если другое оборудование установщика работает через виртуальный COM-порт, то никаких проблем не возникнет.

Рассмотрим случай работы с терминалами Автограф GSM, которые так же настраиваются через USB-интерфейс, но в режиме прямого доступа.

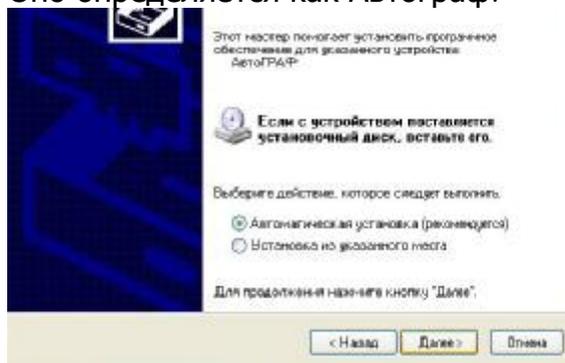
В этом случае перед первым подключением адаптера у установщика уже стоит драйвер FTDI, а устройство называется Автограф:



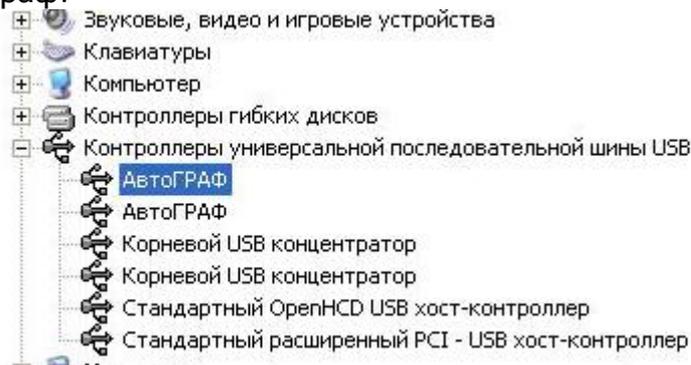
После подключения адаптера обнаруживается новое устройство:



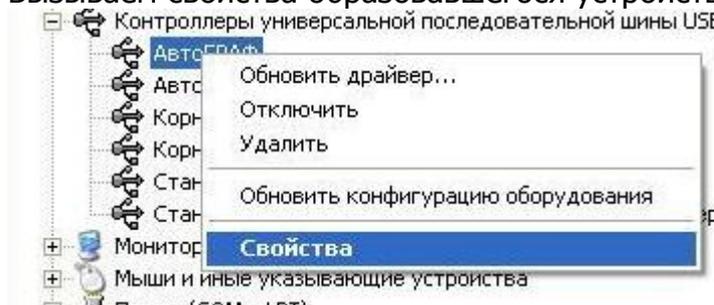
Оно определяется как Автограф.



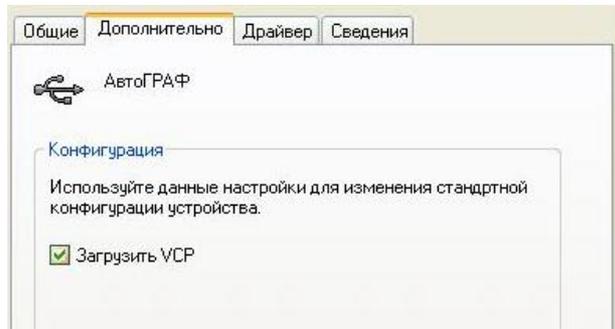
Продолжить установку. В диспетчере устройств появится второе устройство Автограф:



Вызываем свойства образовавшегося устройства



В разделе "Дополнительно" указываем режим работы "загрузить VCP".



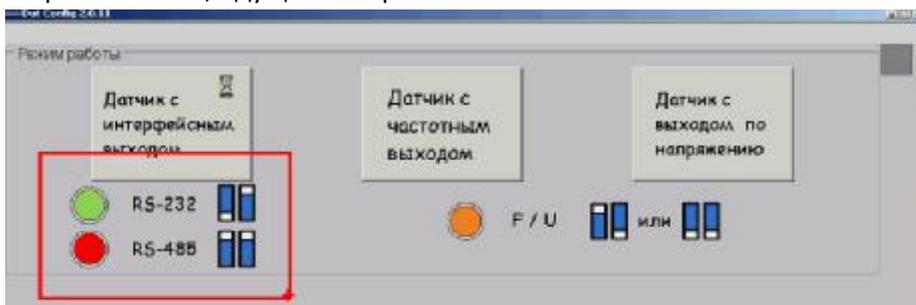
Теперь с этого ПК возможно как настраивать приборы Автограф, так и работать с нашим адаптером – второе устройство Автограф работает с прибором напрямую.

Обращаем внимание, что возможность переключения режима работы имеется только в самой новой версии драйвера, которую можно скачать с сайта [FTDICHIP.COM](http://FTDICHIP.COM)

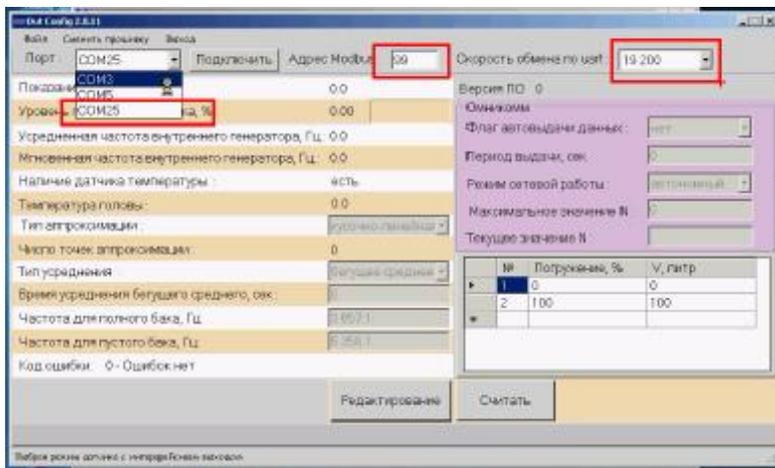
## 4. Подключение датчика к программе DUT CONFIG 2.xx

Для датчиков, выпущенных после декабря 2010 года для работы с полным функционалом настройки датчиков Вам необходима последняя версия программы DUTCONFIG.exe 2.0. В случае использования программы dut config версии 1.x вы сможете откалибровать датчики, но не сможете воспользоваться описанными ниже дополнительными функциями. Датчики, выпущенные до 2011 года необходимо настраивать через программу dut config версии 1.x.

1. Подсоединяем универсальный сервисный адаптер (УСА) к компьютеру. На коробочке УСА со стороны usb-кабеля должна загореться зелёная лампочка.
2. Проверяем, верно ли установлены переключатели на 232 и 485 стрелы и верный ли горит огонёк со стороны кабеля, идущего к Стреле

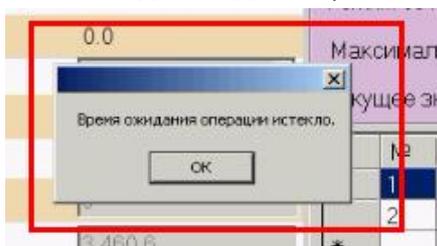


3. Запускаем программу Dut Config 2.exe
4. Нажимает кнопку датчики с интерфейсным выходом
5. а) Выбираем COM-порт. Его номер можно посмотреть в ДИСПЕТЧЕРЕ УСТРОЙСТВ windows, где он назначен драйверу УСА.  
б) Выбираем адрес MODBUS. для 232 датчика – этот параметр неважно чему равен. Для 485 выставляйте 0 - будет идти поиск по всем адресам. Или, если знаете, выставляйте точный адрес (02, 03). По умолчанию с завода датчики выходят с адресом 99.  
в) Стандартная скорость работы с датчиком – 19200.



Нажимаем кнопку ПОДКЛЮЧИТЬ

Если вы видите надпись время ОЖИДАНИЯ ОПЕРАЦИИ ИСТЕКЛО – проверьте порт и адрес.



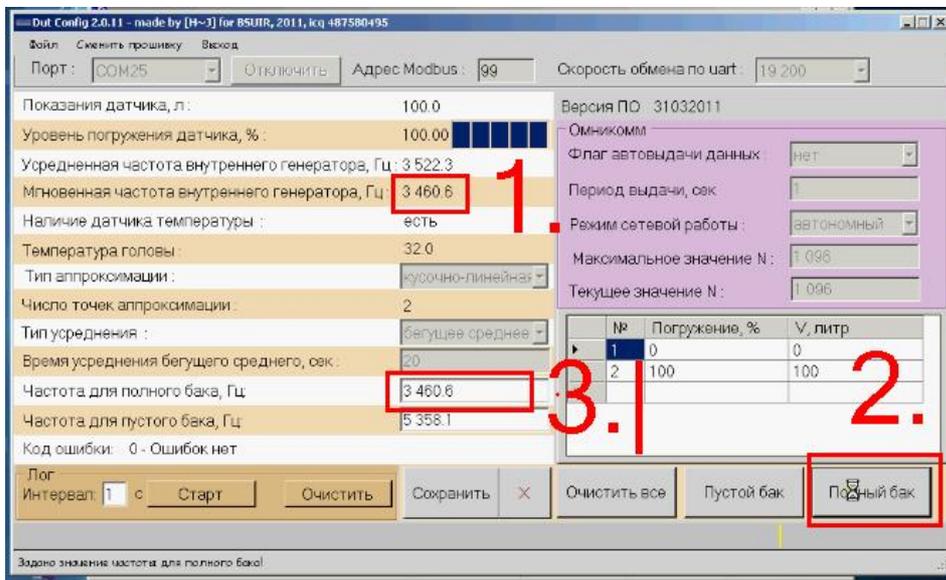
## 5. Калибровка датчика

Калибровка датчика проводится после его окончательной обрезки. Цель калибровки – установить минимум и максимум работы датчика, разделить высоту на равные интервалы, и задать каждому уровню заполненности соответствующее значения выходного сигнала.

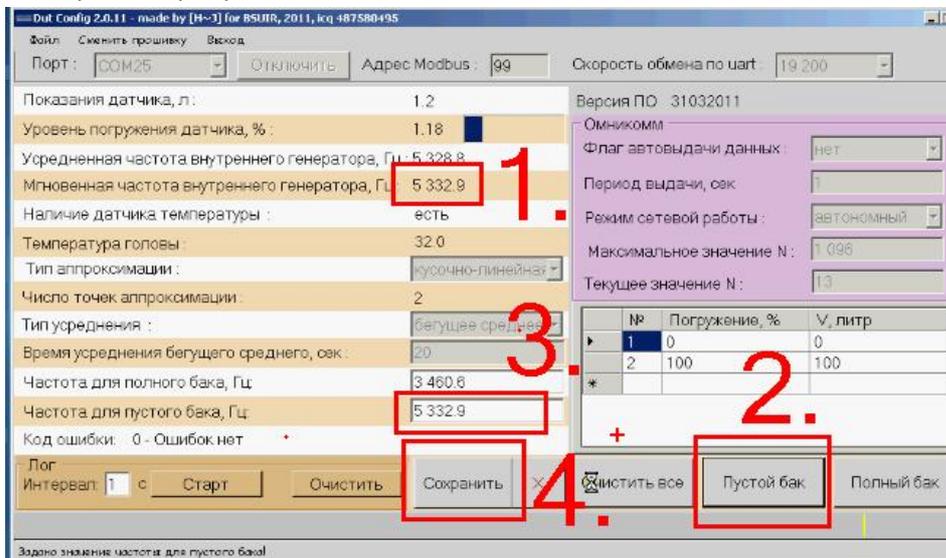
1. Опускаем датчик целиком в ёмкость с топливом.
2. Подключаем его к компьютеру (см. выше раздел 1 ПОДКЛЮЧЕНИЕ).
3. Нажимаем кнопку ТАРИРОВКА.



4. Ожидаем момент, когда в ячейке МГНОВЕННАЯ ЧАСТОТА показания топлива станут неизменными (1). Нажимаем кнопку ПОЛНЫЙ БАК (2). При этом показания занесутся в строку ЧАСТОТА ДЛЯ ПОЛНОГО БАКА.



5. Достаем датчик из топлива и даём ему стечь. Ожидаем момент, когда в ячейке МГНОВЕННАЯ ЧАСТОТА показания топлива станут неизменными (1). Нажимаем кнопку ПУСТОЙ БАК (2). При этом показания занесутся в строку ЧАСТОТА ДЛЯ ПУСТОГО БАКА (3).



Задание **таблицы соответствия**. Вам необходимо проконтролировать (добавить при необходимости) наличие 2 строк в таблице:

№	Погружение, %	V, литр
1	0	0
2	100	100

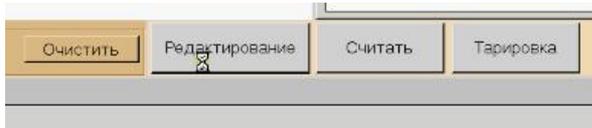
В таком случае, выходное значение датчика будет кодировать % от высоты погружения. Например, при разрядности 4096 выходное значение 2000 будет соответствовать высоте уровня топлива  $2000/4096$  высоты датчика.

6. Сохраняем изменения, нажимая кнопку СОХРАНИТЬ (4).

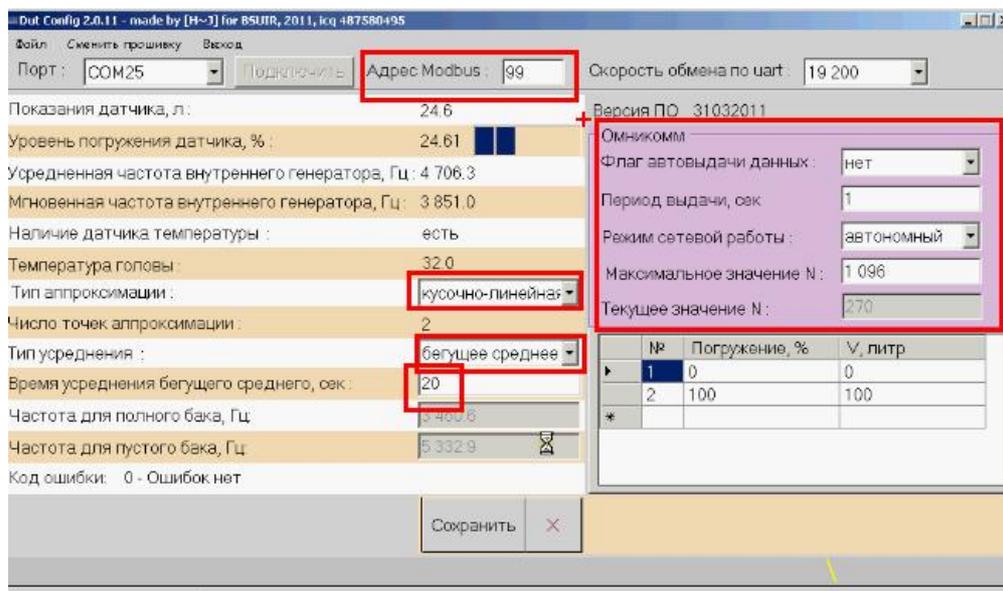
## 6. Задание формата выдачи данных

Задание формата выдачи данных проводится в меню РЕДАКТИРОВАТЬ

1. Нажимаем кнопку РЕДАКТИРОВАТЬ



2. Проводим настройку следующих параметров:



**Флаг выдачи данных.** Необходимо выставить ДА, если терминал не умеет сам опрашивать датчик. После этого задать ПЕРИОД ВЫДАЧИ ДАННЫХ.

**Режим сетевой работы.** Если у вас к регистратору подключается один 232 датчик или один 485 датчика то надо выставить АВТОНОМНЫЙ. Если вы используете несколько 485 датчиков, подключённых к 1 терминалу, то необходимо выставить СЕТЕВОЙ и назначить АДРЕС MODBUS. Адреса MODBUSу них должны быть разными (01, 02 и др).

**Максимальное значение N** – произвольное число до 65536, которое поддерживает терминал. Например, 1024, 4096, 9999 и др. Чем выше значение – чем выше дискретность (то есть выше точность кодирования данных).

**Усреднение показаний датчика:** Усреднить можно и функционалом самого терминала, но внутреннее усреднение датчика позволяем передавать в терминал более точное значение. Стандартные настройки усреднения:

**Тип аппроксимации** – выставить кусочно-линейную. Полиномиальная аппроксимация и случай её использования описан ниже.

**Тип усреднения** – оставить БЕГУЩЕЕ СРЕДНЕЕ для баков прямоугольной формы или выставить ЭКСПОНЕНЦИАЛЬНОЕ для баков цилиндрической формы. Максимальное значение бегущего среднего усреднения – 90 секунд (большее значение не сохраняется, а “сбрасывается” в 90. Рекомендуем ставить от 30 до 60 секунд для обычного транспорта. Чем хуже условия движения, чем больше время усреднения. При необходимости использовать большее усреднение – выставляйте режим ЭКСПОНЕНЦИАЛЬНОЕ и задавайте желаемое усреднение коэффициентом  $\alpha$ , равным:  $\alpha=1/(\text{время усреднения в секундах})$ .

## 7. Специальный случай 1. задание таблицы тарировки для бака прямоугольной формы внутри датчика

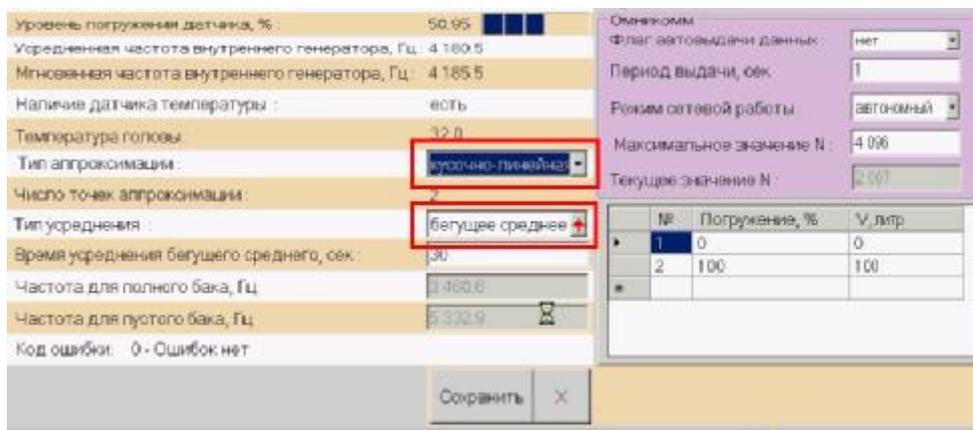
Для гибкости работы мы рекомендуем использовать эту функцию только в тех случаях, когда регистратор или диспетчерское ПО не умеют хранить внутри себя тарировочную таблицу.

В датчике есть возможность хранить внутри себя тарировочную таблицу. В таком случае датчик будет вам выдавать на выходе значение в % от его объёма, а не % заполнения по высоте. Например, при разрядности 4096 выходное значение 2000 будет соответствовать высоте уровня топлива 2000/4096 объёма датчика.

1. ПРОВОДИМ КАЛИБРОВКУ ДАТЧИКА И ЗАДАНИЕ ФОРМАТА ВЫДАЧИ ДАННЫХ (см выше). При задании формата выдачи данных проверяем, что у нас выставлены

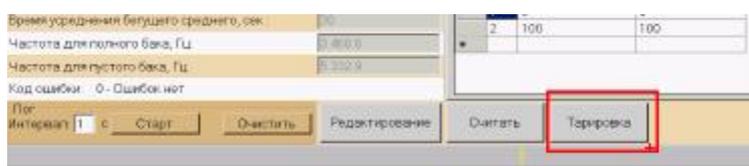
тип аппроксимации – кусочно-линейная

тип усреднения – бегущее среднее

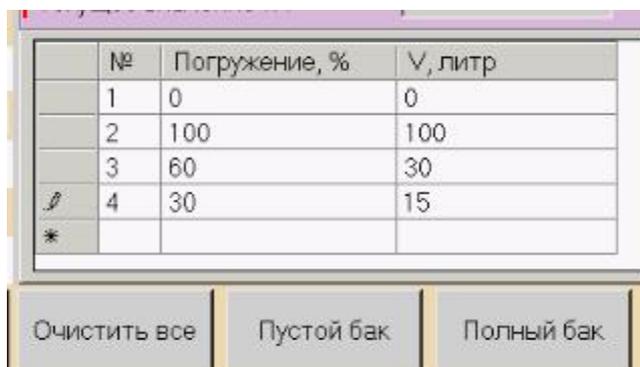


2. Внесение тарировочной таблицы проводится в режиме тарировки:

Нажимаем кнопку ТАРИРОВКА.



Вносим в таблицу дополнительные строки соответствия процента заполнения бака литрам:



Порядок следования строк может быть любым (не обязательно строго по возрастанию или убыванию). Ошибочно введённые строки можно удалять.

## 8. Специальный случай 2. задание формулы тарировки бака цилиндрической формы внутри датчика

Для гибкости работы мы рекомендуем использовать эту функцию только в тех случаях, когда регистратор или диспетчерское ПО не умеют хранить внутри себя тарировочную таблицу.

Полиномная аппроксимация разработана для баков цилиндрической и эллиптической формы (бензовозы, цистерны). В этом случае, в процессе проведения тарировки вы, как и ранее, создаёте таблицу соответствия заливаемого топлива выходному значению датчика. Но в датчик вы заносите математически определённые при тарировке коэффициенты A,B,C,D, которые и описывают это соответствие функцией. После этого вам только остаётся в регистраторе внести размер бака и датчик будет вам выдавать на выходе значение в % от его объёма, а не % заполнения по высоте. То есть, при значении 511 при разрядности 1023 будет соответствовать  $511/1023=50\%$  объёма бака.

1. ПРОВОДИМ КАЛИБРОВКУ ДАТЧИКА И ЗАДАНИЕ ФОРМАТА ВЫДАЧИ ДАННЫХ (см выше). При задании формата выдачи данных проверям, что у нас выставлены

**тип аппроксимации** – полиномиальная

**тип усреднения** – экспоненциальная(предпочтительно) или бегущее среднее

Переменная	Значение
A	0
B	100
C	45
D	45

$$V = A \cdot L^3 + B \cdot L^2 + C \cdot L + D$$

2. Внесение тарировочной функции проводится в режиме тарировки:

Нажимаем кнопку ТАРИРОВКА и вносим коэффициенты, описывающие функцию:

Переменная	Значение
A	0
B	100
C	45
D	45

$$V = A \cdot L^3 + B \cdot L^2 + C \cdot L + D$$

## 9. Описание протокола обмена данными датчиков Стрела D232 (D485)

Протокол обмена датчика уровня топлива с внешним устройством состоит из двух частей: открытая часть для общего использования и закрытая часть.

Закрытая часть предназначена для настройки параметров датчика с помощью программы DYTConfig.exe. Команды закрытой части могут меняться без дополнительного оповещения и не поддерживаться отдельными типами датчиков уровня топлива Стрела, однако поддерживаться программой DYTConfig.exe.

Команды открытой части поддерживаются всеми типами датчиков Стрела, имеющих цифровой интерфейс. Команды не могут быть изменены и будут поддерживаться в новых типах датчиков Стрела.

Открытая часть протокола поддерживает два типа протокола обмена: в бинарном (HEX) виде или в символьном виде (передачей ASCII-последовательностей). Для работы рекомендуется использовать бинарный протокол обмена.

После подачи питания на датчик уровня топлива Стрела и перед подачей первой команды запроса необходимо выждать время не менее 100 мс. Команда, отправленная в течение 100 мс после включения, датчиком Стрела воспринята не будет, обмен возобновится только после 100 мс "тишины" в канале передачи данных.

При работе с датчиками Стрела, находящимися в режиме slave, после отправки команды запроса необходимо дождаться ответа от датчика. Время задержки ответа зависит от скорости обмена и типа протокола обмена, но составляет не более 100 мс. В случае если через 100 мс ответ не получен, возможна повторная отправка запроса.

Данные, между датчиком и внешним устройством передаются в виде сообщений стандартного формата. Сообщение передаётся пакетами байт. Передача каждого байта начинается СТАРТ битом, а завершается СТОП битом (Рисунок 11). Данные передаются младшим байтом вперёд.

Интервал между последующими байтами в пакете (Тт) должен быть либо меньше длительности передачи 35 бит, либо меньше 1 мс, если

$$\frac{35}{\text{скорость (бод)}} < 1 \text{ мс}$$

Окончанием пакета байт считается ситуация, когда последующий байт не приходит в течение времени (Тп), превышающего максимальный интервал (Тт) + 1 мс (Рисунок 12).

Тбайт - время передачи одного байта информации;

Тт - таймаут между последовательными байтами в пакете.

Данный порядок распространяется на работу датчика Стрела и на работу внешнего устройства.

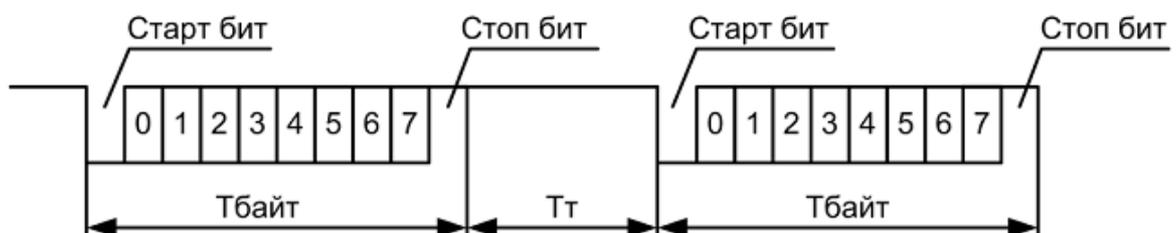


рис. 9 Сигнал передачи данных

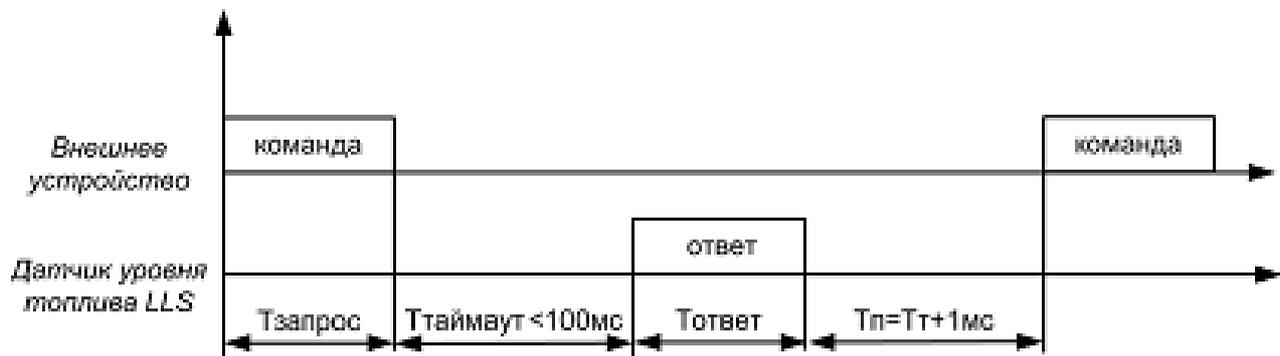


рис. 10 Сигнал передачи данных

## Описание команд для бинарного протокола обмена

### Формат сообщений для бинарного протокола обмена

Все команды бинарного протокола обмена имеют одинаковый стандартизованный вид, представленный в табл. 10:

табл. 1

Порядковый номер поля	Название поля	Размер поля, байт	Описание
1	Префикс	1	Поле является маркером начала сообщения. Входящие сообщения должны иметь префикс 31h, а исходящие сообщения выдаваться программой с префиксом 3Eh.
2	Сетевой адрес	1	Поле содержит: - для префикса 31h сетевой адрес получателя сообщения; - для префикса 3Eh сетевой адрес отправителя сообщения.
3	Код операции	1	Поле содержит: - для префикса 31h код операции, которую программа должна выполнить; - для префикса 3Eh код операции, на которую выдаётся ответ.
4	Данные	Зависит от кода операции	Состав данных и формат поля зависит от кода операции.
5	Контрольная сумма	1	Поле используется для контроля целостности данных. Алгоритм вычисления приведён в Приложении А.

### Однократное считывание данных (команда 06h)

Команда предназначена для чтения текущих данных: относительный уровень, температура, частота. Относительный уровень выдается датчиком в интервале, заданном параметрами "Минимальное показание", "Максимальное показание".

Данные передаются младшим байтом вперёд.

Формат команды:

Смещение, байт	Размер поля, байт	Значение	Описание
0	1	31h	Префикс.
+1	1	00h...FFh	Сетевой адрес получателя.
+2	1	06h	Код операции.
+3	1	00h...FFh	Контрольная сумма.

Формат ответа:

Смещение, байт	Размер поля, байт	Значение	Описание
0	1	3Eh	Префикс.
+1	1	00h...FFh	Сетевой адрес отправителя.
+2	1	06h	Код операции.
+3	1	-128...127	Температура в градусах Цельсия.
+4	2	0000h...FFFFh	Относительный уровень.
+6	2	0000h...FFFFh	Значение частоты.
+8	1	00h...FFh	Контрольная сумма.

После включения датчика и до момента установки устойчивого результата измерения (время для различных моделей датчиков может составлять единицы секунд) результаты измерения уровня не являются достоверными. При этом датчик возвращает значение уровня LVL, превышающее число 0FFFh (или 4095d). Получив пакет с таким значением уровня, рекомендуется прекратить его дальнейшую обработку и выждать паузу 1-2 сек. и повторить запрос к датчику.

ПЕРИОДИЧЕСКАЯ ВЫДАЧА ДАННЫХ (КОМАНДА 07h)

Команда предназначена для включения периодической выдачи данных.

После обработки команды датчик начнёт производить периодическую выдачу данных: уровня, температуры, частоты, с интервалом времени, заданным командой 13h. При нулевом значении интервала выдачи данные не выдаются.

Отключение периодической выдачи данных производится после получения любой достоверной команды, сброса процессора или отключения напряжения питания, если не установлен режим выдачи данных по умолчанию (п.6.1.4).

Формат сообщения с данными представлен в таблице (Таблица 7). Данные передаются младшим байтом вперёд.

Формат команды:

Смещение, байт	Размер поля, байт	Значение	Описание
0	1	31h	Префикс.
+1	1	00h...FFh	Сетевой адрес получателя.
+2	1	07h	Код операции.
+3	1	00h...FFh	Контрольная сумма.

Формат ответа на команду:

Смещение, байт	Размер поля, байт	Значение	Описание
0	1	3Eh	Префикс.
+1	1	00h...FFh	Сетевой адрес отправителя.
+2	1	07h	Код операции.
+3	1	00h	Команда выполнена успешно.
		01h	Команда не может быть выполнена.
+4	1	00h...FFh	Контрольная сумма.

Формат периодически выдаваемых данных:

Смещение, байт	Размер поля, байт	Значение	Описание
0	1	3Eh	Префикс.
+1	1	00h...FFh	Сетевой адрес отправителя.
+2	1	07h	Код операции.
+3	1	-128...127	Температура в градусах Цельсия T.
+4	2	0000h...FFFFh	Относительный уровень LVL.
+6	2	0000h...FFFFh	Значение частоты F.
+8	1	00h...FFh	Контрольная сумма.

РЕГУЛИРОВКА ИНТЕРВАЛА ПЕРИОДИЧЕСКОЙ ВЫДАЧИ (КОМАНДА 13H)

Команда предназначена для установки интервала периодической выдачи данных.

По команде датчик сохраняет новое значение интервала выдачи данных в энергонезависимую память. При нулевом значении интервала данные не выдаются.

Формат команды:

Смещение, байт	Размер поля, байт	Значение	Описание
0	1	31h	Префикс.
+1	1	00h...FFh	Сетевой адрес получателя.
+2	1	13h	Код операции.
+3	1	0...255	Интервал выдачи данных в секундах.
+4	1	00h...FFh	Контрольная сумма.

Формат ответа:

Смещение, байт	Размер поля, байт	Значение	Описание
0	1	3Eh	Префикс.
+1	1	00h...FFh	Сетевой адрес отправителя.
+2	1	13h	Код операции.
+3	1	00h	Команда выполнена успешно.
		01h	Команда не может быть выполнена.
+4	1	00h...FFh	Контрольная сумма.

РЕЖИМ ВЫДАЧИ ДАННЫХ ПО УМОЛЧАНИЮ (КОМАНДА 17H)

Команда определяет порядок выдачи данных после включения питания или сброса процессора.

По команде (Таблица 12) программа сохраняет параметр в энергонезависимую память и отправляет ответ (Таблица 13) с результатами выполнения команды.

После включения питания или сброса программа будет отправлять по интерфейсу данные периодически с интервалом времени, задаваемым командой 13h. При нулевом значении интервала данные выдаваться не будут.

#### Формат команды:

Смещение, байт	Размер поля, байт	Значение	Описание
0	1	31h	Префикс
+1	1	00h...FFh	Сетевой адрес получателя.
+2	1	17h	Код операции.
+3	1	00h	Данные не выдаются.
		01h	Данные выдаются в бинарном виде.
		02h	Данные выдаются в символьном виде.
+4	1	00h...FFh	Контрольная сумма.

#### Формат ответа:

Смещение, байт	Размер поля, байт	Значение	Описание
0	1	3Eh	Префикс
+1	1	00h...FFh	Сетевой адрес отправителя.
+2	1	17h	Код операции.
+3	1	00h	Команда выполнена успешно.
		01h	Команда не может быть выполнена.
+4	1	00h...FFh	Контрольная сумма.

## Описание команд для символьного протокола обмена

Обмен по символьному протоколу состоит в приеме и посылке последовательности ASCII символов, воспринимаемых как команды запроса и ответа.

### Чтение данных

Команда предназначена для чтения текущих данных: относительный уровень, температура, частота.

Команда представляет собой последовательность ASCII символов «D» и «O». После получения команды «DO» программа выдаст ответ в виде последовательности ASCII символов.

Например, F=0AF9 t=1A N=03FF.0 <CR><LF> ,

где F - текущее значение частоты, t - текущее значение температуры в градусах по Цельсию, N - значение уровня. Все значения в шестнадцатеричном виде.

В случае если значение частоты больше FFFh, данные считаются невалидными.

### Периодическая выдача данных

Команда предназначена для включения периодической выдачи данных. После обработки команды датчик производит периодическую выдачу в символьном виде (ASCII коды) следующих данных: относительный уровень, температура, частота.

Данные выдаются периодически с интервалом, заданным при настройке датчика (программой DYT Config.exe). В случае установки интервала выдачи данных равным нулю выдача данных производиться не будет.

Включение периодической выдачи данных производится отправкой строки символов «DP». После обработки команды будет получена строка символов.

Например, F=0AF9 t=1A N=03FF.0 <CR><LF> ,

где F - текущее значение частоты, t - текущее значение температуры в градусах по Цельсию, N - значение уровня.

Выключение периодической выдачи данных производится после получения любой достоверной команды, сброса процессора или отключения напряжения питания.

### АЛГОРИТМ CRC

Для расчета CRC с полином  $a^8 + a^5 + a^4 + 1$  применяются следующий алгоритм (язык C):

```
U8 CRC8(U8 data, U8 crc)
{ U8 i = data ^ crc;
  crc = 0;
  if(i & 0x01) crc ^= 0x5e;
  if(i & 0x02) crc ^= 0xbc;
  if(i & 0x04) crc ^= 0x61;
  if(i & 0x08) crc ^= 0xc2;
  if(i & 0x10) crc ^= 0x9d;
  if(i & 0x20) crc ^= 0x23;
  if(i & 0x40) crc ^= 0x46;
  if(i & 0x80) crc ^= 0x8c;
  return crc; }
```