

МНОГОКАНАЛЬНЫЕ СИСТЕМЫ СБОРА ДАННЫХ

LTR216

РУКОВОДСТВО ПРОГРАММИСТА

Ревизия 1.0.0
Апрель 2018

Автор руководства:

Борисов Алексей

ООО “Л Кард”

117105, г. Москва, Варшавское ш., д. 5, корп. 4, стр. 2

тел.: +7 (495) 785-95-25

факс: +7 (495) 785-95-14

Адреса в Интернет:

<http://www.lcard.ru>

E-Mail:

Отдел продаж: sale@lcard.ru

Техническая поддержка: support@lcard.ru

Отдел кадров: job@lcard.ru

Таблица 1: Ревизии текущего документа

Ревизия	Дата	Описание
1.0.0	17.04.2018	Первая ревизия данного документа

Оглавление

1	О чем этот документ	6
2	Установка и подключение библиотеки к проекту	7
3	Общий подход к работе с библиотекой	8
3.1	Общий алгоритм работы с модулем	8
3.2	Настройка модуля	9
3.2.1	Одноканальный и многоканальный режимы коммутации АЦП	10
3.2.1.1	Одноканальный режим	10
3.2.1.2	Многоканальный режим	10
3.2.2	Настройка фильтра АЦП	12
3.2.3	Настройка последовательности и параметров опроса каналов	13
3.2.4	Настройка тока питания датчиков	14
3.2.5	Настройка схемы подключения опорного датчика	15
3.2.6	Настройки используемых фоновых измерений	15
3.2.7	Настройка разрешения использования тарифовки	17
3.3	Измерение начальных параметров	17
3.4	Сбор и обработка данных	18
3.5	Проверка линий и условий измерения	19
3.6	Выполнение тарифовки	19
3.7	Работа с пользовательской областью энергонезависимой памяти модуля	20
4	Константы, типы данных и функции библиотеки	22
4.1	Константы и перечисления	22
4.1.1	Константы и макроопределения	22
4.1.2	Коды ошибок, специфичные для модуля LTR216	23
4.1.3	Диапазоны измерения модуля LTR216	24
4.1.4	Типы фильтров АЦП модуля LTR216	25
4.1.5	Режим коммутации каналов модуля	25
4.1.6	Флаги конфигурации	25
4.1.7	Фоновые измерения	26
4.1.8	Начальные измерения	28
4.1.9	Состояние вспомогательного измерения	28
4.1.10	Флаги обработки данных	29
4.1.11	Флаги состояния канала при обработке данных	30
4.1.12	Флаги обнаруженных ошибок канала при обработке данных	31
4.1.13	Флаги состояния модуля при обработке данных	31
4.1.14	Флаги обнаруженных ошибок модуля при обработке данных	32
4.1.15	Действительность информации в энергонезависимой памяти модуля	32
4.2	Типы данных	33

4.2.1	Калибровочные коэффициенты для одного канала.	33
4.2.2	Калибровочные коэффициенты источников тока питания датчиков	33
4.2.3	Заводские калибровочные коэффициенты	33
4.2.4	Настройки тарировки канала.	34
4.2.5	Информация о тарировке одного канала.	34
4.2.6	Информация о модуле.	34
4.2.7	Параметры текущего состояния модуля.	35
4.2.8	Настройки логического канала	36
4.2.9	Настройки модуля.	36
4.2.10	Низкоуровневые настройки модуля	40
4.2.11	Описатель модуля модуля.	41
4.2.12	Параметры фильтра	42
4.2.13	Состояние фонового измерения.	42
4.2.14	Состояние модуля во время сбора данных	43
4.2.15	Состояние канала во время сбора данных	44
4.3	Функции	45
4.3.1	Функции инициализации и работы с соединением с модулем.	45
4.3.1.1	Инициализация описателя модуля.	45
4.3.1.2	Открытие соединения с модулем.	45
4.3.1.3	Закрытие соединения с модулем.	46
4.3.1.4	Проверка, открыто ли соединение с модулем.	46
4.3.2	Функции для изменения настроек модуля	46
4.3.2.1	Запись настроек в модуль.	46
4.3.2.2	Заполнение делителя частоты АЦП	47
4.3.2.3	Вычисление делителя частоты АЦП	47
4.3.2.4	Заполнение параметров фильтра АЦП	48
4.3.2.5	Подбор параметров фильтра АЦП	49
4.3.2.6	Получение параметров фильтра по настройкам	49
4.3.2.7	Заполнение поля для задания силы тока питания датчиков	50
4.3.2.8	Подбор кода для задания силы тока питания датчиков	50
4.3.2.9	Расчет величины силы тока по коду	51
4.3.2.10	Расчет низкоуровневых параметров цикла опроса АЦП	51
4.3.3	Функции для управления сбором данных	52
4.3.3.1	Измерение начальных параметров перед запуском сбора	52
4.3.3.2	Запуск сбора данных	53
4.3.3.3	Останов сбора данных.	53
4.3.3.4	Инициализация структуры состояния модуля при обработке данных.	53
4.3.3.5	Инициализация структур состояния каналов модуля при обработке данных.	54
4.3.3.6	Прием данных от модуля.	55
4.3.3.7	Обработка принятых от модуля слов.	56
4.3.3.8	Проверка линий подключения и условий измерения	57
4.3.4	Функции для пользовательской тарировки каналов	58
4.3.4.1	Тарировка смещения нуля выбранных каналов.	58
4.3.4.2	Тарировка шкалы выбранных каналов.	59
4.3.4.3	Запись тарировочных коэффициентов во flash-память	60
4.3.4.4	Чтение тарировочных коэффициентов из flash-памяти	60
4.3.5	Функции для работы с flash-памятью модуля	61

4.3.5.1	Чтение блока данных из Flash-памяти.	61
4.3.5.2	Запись блока данных во Flash-память модуля.	61
4.3.5.3	Стирание блока во Flash-памяти.	62
4.3.5.4	Разрешение записи в пользовательскую область Flash-памяти.	62
4.3.5.5	Запрет записи в пользовательскую область Flash-памяти.	62
4.3.6	Функции вспомогательного характера	63
4.3.6.1	Разрешение или запрет источников тока питания датчиков.	63
4.3.6.2	Получение сообщения об ошибке.	63

Глава 1

О чем этот документ

Данный документ предполагает, что пользователь знаком с документами [‘Начиная работать с крейтовой системой LTR. Вопросы по программному обеспечению.’](#) и [‘Программное обеспечение для системы LTR’](#), в которых описываются основные принципы работы программного обеспечения для крейтов LTR.

Данный документ предназначен в первую очередь для программистов, которые собираются писать свои программы для работы с модулем LTR216 с использованием предоставляемой фирмой “Л Кард” библиотеки ltr216api.

В данном документе рассматривается вопрос подключения библиотеки к проекту пользователя, дается подробное описание интерфейсных функций, предоставляемых библиотекой, и используемых типов, а также дается описание основных подходов к использованию этих функций.

Сама библиотека написана на языке *C* и все объявления функций и типов приводятся на языке *C*. Однако все привязки к другим языкам программирования являются лишь обертками над библиотекой *C* и все функции, типы и параметры сохраняют свои значения и для других языков программирования. Поэтому этот документ полезен и пользователям, пишущим на других языках программирования.

В настоящем документе не рассматриваются какие-либо вопросы, касающиеся характеристик модуля и подключения сигналов, а также лишь в общем затрагиваются принципы работы самого модуля. Перечисленные вопросы рассматриваются в соответствующем разделе документа [‘Крейтовая система LTR. Руководство пользователя’](#), с которым необходимо ознакомиться перед прочтением данного документа.

Глава 2

Установка и подключение библиотеки к проекту

Использование библиотек для работы с крейтовой системой LTR описано в документе [‘Начиная работать с крейтовой системой LTR. Вопросы по программному обеспечению.’](#).

Глава 3

Общий подход к работе с библиотекой

3.1 Общий алгоритм работы с модулем

Данный раздел описывает типичную последовательность действий при работе с модулем LTR216. Более подробно каждый шаг будет описан в последующих разделах.

Типичная последовательность действий имеет следующий вид:

1. Создать экземпляр структуры `TLTR216`, представляющей собой описатель модуля. Описатель модуля содержит всю информацию о модуле и используется при вызове всех остальных функций.
2. Проинициализировать поля описателя с помощью `LTR216_Init()`
3. Установить соединение с интересующим модулем с помощью функции `LTR216_Open()`.
4. Заполнить необходимые поля с настройками модуля подструктуры `Cfg` описателя модуля и вызвать `LTR216_SetADC()` для записи настроек в модуль. Для заполнения части полей доступны вспомогательные функции.
5. Вызвать функцию `LTR216_InitMeasParams()` для определения начальных значений вспомогательных измерений.
6. Создание структуры `TLTR216_DATA_STATUS` и массива структур `TLTR216_DATA_CH_STATUS` для получения состояния модуля и каналов во время сбора. Проинициализировать их с помощью `LTR216_DataStatusInit()` и `LTR216_DataChannelsStatusInit()` соответственно.
7. Запуск сбора данных с помощью `LTR216_Start()`.
8. Прием заданного количества слов с помощью `LTR216_Recv()`. Количество слов должно быть кратно размеру кадра (поле `FrameWordsCount` подструктуры `State` описателя модуля)
9. Обработка принятых слов с помощью `LTR216_ProcessData()` для получения результатов измерения и признаков состояния модуля и каналов.
10. Переход к пункту 8, если нужны еще данные, или к пункту 11 при завершении сбора.

11. По завершению работы выполнить останов сбора данных с помощью [LTR216_Stop\(\)](#).
12. Закрывать соединение с модулем, вызвав функцию [LTR216_Close\(\)](#).

3.2 Настройка модуля

После того как связь с модулем установлена, необходимо выполнить настройку модуля. Она выполняется аналогично большинству других модулей LTR: вначале значения всех параметров записываются в соответствующие поля [структуры описателя модуля](#), затем вызывается функция [LTR216_SetADC\(\)](#), которая значения этих полей передает модулю.

Все поля, относящиеся к настройке модуля, объединены в структуру типа [TLTR216_CONFIG](#) (поле [Cfg описателя модуля](#)). Именно эти поля пользователь должен изменять для настройки модуля и именно эти поля влияют на параметры, записываемые [LTR216_SetADC\(\)](#). Часть полей пользователь может заполнять как явно, так и с помощью вспомогательных функций, которые позволяют рассчитать значение поля исходя из значения желаемого результирующего параметра, определяемого полем.

При настройке модуля задаются следующие параметры:

- Одноканальный или многоканальный режим коммутации АЦП (раздел [Одноканальный и многоканальный режимы коммутации АЦП](#))
- Настройки используемого фильтра АЦП, определяющие также его время преобразования (раздел [Настройка фильтра АЦП](#))
- Частоту АЦП в многоканальном режиме (с учетом времени преобразования и времени на коммутацию) (раздел [Многоканальный режим](#))
- Последовательность опроса каналов и диапазоны измерения (раздел [Настройка последовательности и параметров опроса каналов](#))
- Значение силы тока питания датчиков (раздел [Настройка тока питания датчиков](#))
- Используемую схему подключения опорного датчика (раздел [Настройка схемы подключения опорного датчика](#))
- Разрешение фоновых измерений (раздел [Настройки используемых фоновых измерений](#))
- Разрешение использования тарировки (раздел [Настройка разрешения использования тарировки](#))

При выполнении [LTR216_SetADC\(\)](#) функция проверит корректность заполнения полей конфигурации, а также рассчитает ряд результирующих параметров, соответствующих настроенной конфигурации, которые будут сохранены в поля структуры состояния модуля [TLTR216_STATE](#) (поле [State описателя модуля](#)).

3.2.1 Одноканальный и многоканальный режимы коммутации АЦП

Данный режим задается с помощью поля `AdcSwMode` структуры конфигурации и определяет, каким образом будет выполняться опрос каналов АЦП и задаваться временные параметры опроса. Этот режим влияет на значения части других параметров конфигурации и как правило заполняется в первую очередь. Ниже в подразделах будет подробно описан каждый из режимов.

3.2.1.1 Одноканальный режим

Для включения данного режима поле `AdcSwMode` должно иметь значение `LTR216_ADC_SWMODE_SIGNLECH_CONT`.

В одноканальном режиме АЦП коммутация каналов не выполняется. Коммутатор настраивается перед сбором на нужный канал, после чего запускается непрерывный сбор данных с АЦП с максимально возможной скоростью (для выбранных параметров фильтрации). Частота АЦП, соответствующая частоте получения отсчетов от АЦП, полностью определяется временем преобразования самой микросхемы АЦП и зависит только от установленных [настроек фильтра АЦП](#). В этом режиме не происходит сброс фильтра, т.е. время установки фильтра требуется только для первого отсчета АЦП, поэтому в этом режиме часто используется SINC3 фильтр, имеющий большее время установления.

Хотя технически в данном режиме настройки фильтра АЦП позволяют получить частоту сбора АЦП до 250 КГц, исходя из характеристик тракта модуля практически полезным пределом частоты АЦП является частота 50 КГц.

Данный режим имеет следующие особенности:

- Может быть настроен только один канал измерения. Размер логической таблицы (поле `LChCnt`) всегда должно быть равно 1.
- Частота АЦП определяется параметрами фильтра АЦП
- Невозможно выполнять фоновые измерения во время сбора. Соответственно, проверка целостности линий и измерение дополнительных параметров может выполняться только перед запуском пока сбор остановлен.

3.2.1.2 Многоканальный режим

Для включения данного режима поле `AdcSwMode` должно иметь значение `LTR216_ADC_SWMODE_MULTICH_SYNC`.

Данный режим является основным и позволяет использовать все каналы, а также выполнять фоновую проверку линий и других контрольных параметров, автокорректировку смещения нуля и напряжения на опорном тензодатчике во время сбора данных.

Частота АЦП в этом режиме соответствует частоте коммутации каналов и определяется частотой сигнала синхронизации АЦП, которая получается делением на целое число (делитель) частоты `LTR216_ADC_CLOCK` Гц. Данный делитель задается полем `SyncFreqDiv` структуры конфигурации, так что $f_{adc} = f_{adc_clock} / (SyncFreqDiv + 1)$. Для подбора делителя по желаемой частоте АЦП реализована вспомогательная функция `LTR216_FillSyncFreqDiv()`.

Измерения каналов как и в остальных АЦП с коммутацией каналов выполняются последовательно. Время опроса одного канала соответствует частоте АЦП и определяется частотой синхронизации АЦП, как описано выше. При этом цикл опроса одного канала выглядит следующим образом:

1. На АЦП подается сигнал синхронизации, по которому выполняется сброс фильтра и начало нового преобразования
2. АЦП выполняет преобразование в течение времени (далее называемом временем преобразования АЦП), которое определяется [настройками фильтра АЦП](#).
3. По завершению преобразования АЦП передает отсчет с результатом измерения и одновременно с этим происходит переключение коммутатора в соответствии с параметрами следующего измеряемого канала
4. По завершению периода частоты синхронизации АЦП на АЦП снова подается сигнал синхронизации и начинается выполнение следующего измерения в соответствии с пунктом 1.

Время выполнения стадии 4 цикла опроса одного канала называется временем коммутации, т.к. за него должны успеть завершиться все коммутационные процессы, чтобы избежать влияние предыдущего измерения на следующее. Это время зависит от параметров кабеля и подключенных датчиков. Способ его определения описан в документе [‘Крейтовая система LTR. Руководство пользователя’](#). Таким образом, параметры фильтра АЦП и частота АЦП должны быть выбраны таким образом, чтобы время выполнения преобразования было меньше времени опроса канала АЦП (обратная величина от частоты АЦП) не менее чем на требуемое время коммутации, соответствующее текущим параметрам подключения.

Настройка этих параметров может выполняться следующим образом:

- определяется желаемая частота АЦП и подбирается делитель частоты синхронизации с помощью [LTR216_FillSyncFreqDiv\(\)](#).
- определяется необходимое время коммутации в соответствии с параметрами подключения и записывается в поле [AdcMinSwTimeUs](#).
- определяются параметры фильтра АЦП, чтобы при полученном времени преобразования оставалось достаточно времени на выполнение коммутации. Для этого можно использовать функцию [LTR216_FillFilterParams\(\)](#), которая подбирает оптимальные параметры фильтра, чтобы время на коммутацию было не ниже заданного при заданной частоте АЦП. Если заданное значение времени коммутации не достижимо при выбранной частоте АЦП, то функция вернет меньшее значение результирующего времени коммутации, а в качестве результата будет возвращена ошибка [LTR216_ERR_UNSUFW_TIME](#). В этом случае частоту АЦП необходимо уменьшить. Минимальное время выполнения преобразования АЦП в многоканальном режиме составляет 20 мкс, т.е. частота АЦП должна быть не больше $1000000 / (20 + AdcMinSwTimeUs)$ Гц.

Возможен и обратный порядок, когда вручную подбираются параметры фильтра, исходя из нужных характеристик, затем определяется минимальное время коммутации и уже исходя из них рассчитывается и устанавливается частота АЦП.

3.2.2 Настройка фильтра АЦП

В модуле установлена микросхема АЦП AD7176-2, которая имеет встроенные настраиваемые фильтры. В данной главе будут дано общее описание возможностей настройки фильтров и способ задания этих настроек. Более развернутая информация с графиками и подробными таблицами для каждого фильтра может быть найдена в [документации на данную микросхему](#) в разделе “Digital Filters”.

Может быть выбран один из трех типов фильтра, а также для каждого типа фильтра по своей формуле может быть настроена частота выдачи данных АЦП (ODR - Output data rate) — величина, обратная времени выполнения преобразования (в одноканальном режиме эта частота соответствует частоте АЦП, а в многоканальном определяет максимальную частоту АЦП при заданном времени коммутации, как описано в разделе [Многоканальный режим](#)). Уменьшение значения ODR ограничивает максимально возможную частоту АЦП, но при этом улачивает соотношение сигнал-шум, а также от него зависят значения частот наибольшей режекции фильтра.

Тип фильтра задается полем [FilterType](#), а время выполнения преобразования определяется полем [AdcOdrCode](#) по своей формуле, в зависимости от типа фильтра и режима коммутации (одноканальный или многоканальный).

Всего доступно для настройки 3 типа фильтра (определяются перечислением [e_LTR216_FILTER_TYPE](#)):

- Каскад из SINC5 и SINC1 фильтра. Обладает наименьшим временем установления (один период преобразования для ODR 10 КГц и ниже), поэтому наиболее подходит для многоканального режима на больших частотах АЦП. Поле [AdcOdrCode](#) может принимать для этого типа фильтра значение от 0 до 20 и определяет одну из 21 предопределенных частот выдачи данных. Соответствие кодов поля и результирующей частоты выдачи данных (в обоих режимах), а также первой частоты режекции (Fnotch), приведены в таблице:

AdcOdrCode	ODR однокан., Гц	ODR многокан., Гц	Fnotch, Гц
0	250000	50000	250000
1	125000	41667	125000
2	62500	31250	62500
3	50000	27778	50000
4	31250	20833	31250
5	25000	17857	25000
6	15625	12500	15625
7	10000	10000	11905
8	5000	5000	5435
9	2500	2500	2604
10	1000	1000	1016
11	500	500	504
12	397.5	397.5	400
13	200	200	200.64
14	100.2	100.2	100.4
15	59.98	59.98	59.98
16	49.96	49.96	50.00
17	20	20	20.01
18	16.66	16.66	16.66

AdcOdrCode	ODR однокан., Гц	ODR многокан., Гц	Fnotch, Гц
19	10	10	10
20	5	5	5

- SINC3 фильтр. Имеет всегда время установления в 3 периода частоты ODR. Соответственно часто используется в одноканальном режиме, где нет необходимости сбрасывать фильтр при каждом измерении, в результате чего в этом режиме позволяет получить лучшее соотношение сигнал-шум по сравнению с SYNC1+SINC5 фильтрами при той же частоте ODR. Также особенностью данного режима является то, что он позволяет более точно настроить частоту выдачи данных и соответственно при необходимости подобрать точно частоту режекции фильтра (если в задаче известны частоты наибольших помех). В этом случае [AdcOdrCode](#) может принимать значение от 1 до $2^{14} - 1$, а частота выдачи данных в одноканальном режиме $ODR = 8000000 / (32 * AdcOdrCode)$ Гц, которая соответствует и частоте наибольшей режекции. В многоканальном режиме частота выдачи данных в 3 раза меньше, чем в одноканальном.
- Фильтр улучшенной фильтрации на частоте 50 Гц (и 60 Гц). Доступен только на низких частотах (ODR не выше 27.27 Гц). Чем ниже частота, тем сильнее подавление на частотах 50 и 60 Гц. Поле [AdcOdrCode](#) может принимать всего 4 значения, как указано в таблице:

AdcOdrCode	ODR, Гц	Уровень подавления на 50/60 Гц, дБ
2	27.27	47
3	25	62
5	20	85
6	16.667	90

Функция [LTR216_GetFilterOutParams\(\)](#) позволяет получить указанные параметры фильтра, исходя из выбранных настроек.

Настройки фильтра могут быть подобраны автоматически по заданному режиму коммутации АЦП, частоте АЦП и необходимому времени на коммутацию (для многоканального режим) с помощью [LTR216_FillFilterParams\(\)](#) или [LTR216_FindFilterParams\(\)](#).

3.2.3 Настройка последовательности и параметров опроса каналов

LTR216 является модулем с коммутацией каналов. Это означает, что на модуле присутствует одна микросхема АЦП и в один момент времени выполняется преобразование по одному каналу, а многоканальность достигается за счет последовательного опроса всех требуемых каналов. Как и в других АЦП “Л Кард” с коммутацией каналов последовательность опроса каналов и параметры измерения при каждом опросе задаются с помощью логической таблицы, которая представляет собой последовательность логических каналов. Количество логических каналов задается полем [LChCnt](#), а сама таблица определяется первыми [LChCnt](#) элементами массива [LChTbl](#). Параметры логического канала задаются структурой [TLTR216_LCHANNEL](#) и определяют номер физического

канала, который опрашивается за этот цикл, а также используемый диапазон измерения при данном цикле опроса. Соответственно, модуль сперва опрашивает физический канал, заданный в первом элементе логической таблицы, затем канал, определенный вторым элементом и так до конца таблицы, после чего начинает снова опрос с первого логического канала.

В отличие от большинства других АЦП с коммутацией каналов, конечный цикл опроса каналов модуля LTR216 может быть более сложным, чем задан в логической таблице, т.к. помимо основных измерений могут выполняться дополнительные фоновые измерения (их назначение описано в разделе [Настройки используемых фоновых измерений](#)). После выполнения заданных в логической таблице измерений, если разрешены какие-то фоновые измерения, то модуль переходит к их опросу и только после этого завершает цикл и переходит снова к опросу первого логического канала таблицы. Если же фоновые измерения не разрешены, то после последнего канала таблицы модуль переходит сразу к первому каналу.

Полный цикл опроса всех логических каналов, как и в остальных АЦП с коммутацией каналов, называется кадром. Величина, обратная времени одного опроса всех каналов, называется частотой кадра. Так как за кадр получается по одному отсчету каждого логического канала, то по сути частота кадра соответствует частоте следования отсчетов одного логического канала (частота на канал). В простейшем случае, когда не используются фоновые измерения, частота кадров в `LChCnt` раз меньше частоты АЦП, т.е. частота АЦП делится между всеми логическими каналами. В случае включения фоновых измерений, частота кадра будет ниже, т.к. необходимо время в кадре для дополнительных измерений. Величина снижения зависит от многих факторов: включенного набора фоновых измерений, частоты АЦП, используемых диапазонов и т.д.

Возможно уменьшить частоту кадров по сравнению с максимально возможной путем добавления холостых дополнительных холостых циклов измерения. Для этого можно установить значение поля `AdcReqFrameFreq`. Если оно установлено и больше нуля, то при выполнении настройки библиотека пытается подобрать количество холостых тактов так, чтобы результирующая частота кадра была как можно ближе к установленной.

3.2.4 Настройка тока питания датчиков

LTR216 имеет источники тока для питания подключаемых датчиков. Все источники тока согласованы, т.е. величины этих токов для разных каналов равны в пределах погрешности. Величина тока настраивается с помощью поля `ISrcCode`, которое может быть в пределе от 0 до `LTR216_ISRC_CODE_MAX` включительно. Значение тока задается как $2048 * (ISrcCode + 1) / (4096 * 33.2)$ мА. Кроме того, источник тока калибруется на заводе производителя и его калибровочные коэффициенты записываются во Flash-память модуля (после открытия, они доступны для чтения в информации о модуле в поле `ISrc`. При этом калибруется только опорный источник тока, а соответствие остальных опорному контролируется при производстве.

Функции `LTR216_FillISrcCode()` и `LTR216_FindISrcCode()` позволяют подобрать значение поля `ISrcCode` с использованием заводских калибровочных коэффициентов так, чтобы полученный ток был наиболее близок к заданному. Функция `LTR216_CalcISrcValue()` выполняет обратное преобразование — по коду вычисляет значение тока.

3.2.5 Настройка схемы подключения опорного датчика

Модуль позволяет подключить опорный датчик как по 2-х проводной, так и по 4-х проводной схеме подключения. 4-х проводная схема дает более точный результат, т.к. при ней на измерение напряжения на опорном датчике не влияет сопротивление проводов до этого датчика, однако для подключения второй пары проводов в ней используются выходы, соответствующие 16 каналу модуля. Соответственно, при 4-х проводной схеме подключения опорного датчика в модуле могут использоваться только 15, а не 16 каналов. Выбор режима осуществляется с помощью поля `Ch16ForUref` конфигурации модуля (если значение отлично от 0, то используется 4-х проводная схема подключения). Следует отметить, что все метрологические характеристики модуля получены именно для 4-х проводной схемы.

При использовании 2-х проводной схемы измерения и известном сопротивлении проводов его подключения можно учесть падение напряжения на проводах при вычислении значения разбаланса с помощью поля `RrefWireResistance` конфигурации модуля, установив его равным известному сопротивлению проводов. В этом случае падение только на датчике вычисляется как $U_{refR} = U_{ref} - 2 * R_{refWireResistance} * I_{SrcValue} / 1000$. Однако это позволяет учесть только постоянное значение сопротивления, для учета его тепловых уходов следует применять 4-х проводную схему подключения.

3.2.6 Настройки используемых фоновых измерений

Модуль LTR216 позволяет в кадре выполнять не только пользовательские измерения, но и дополнительные фоновые. Это позволяет производить автокорректировку нуля и значений вспомогательных измерений, необходимых для вычисления разбаланса, а также проверку исправности цепей подключения датчиков и режимов измерения, не останавливая сбор данных. Однако включение фоновых измерений приводит к уменьшению максимальной частоты кадра для заданной конфигурации каналов.

Для включения фоновых измерений достаточно заполнить значение поля `BgMeas` путем объединения флагов из `e_LTR216_BG_MEAS` по “ИЛИ”, где каждый флаг соответствует включенному измерению. При вызове `LTR216_SetADC()` функция сама распределит фоновые измерения в кадрах (при этом опрос всех фоновых измерений может быть растянут на несколько кадров для уменьшения влияния на частоту АЦП).

Следует отметить, что данное поле относится именно к фоновым измерениям, выполняемым во время сбора и влияющим на формирование основного цикла опроса каналов АЦП. Его значение не влияет на начальное измерение параметров (функция `LTR216_InitMeasParams()`) или отдельной проверке исправности линий и режима измерения (функция `LTR216_CheckMeasStatus()`). Указанные функции принимают свой набор измерений в качестве отдельного параметра.

В зависимости от типа фонового измерения, его значение может автоматически учитываться для корректировки результата и/или результаты его анализа могут возвращаться в структурах состояния модуля (`TLTR216_DATA_STATUS`) или каналов (`TLTR216_DATA_CH_STATUS`) при обработке данных с помощью `LTR216_ProcessData()`.

Для настройки доступны следующие фоновые измерения:

- Измерение собственного нуля каналов измерения разностного напряжения (`LTR216_BG_MEAS_OFFS`). Используется для автоматической корректировки смещения нуля основных измерительных каналов для учета его ухода во время сбора данных.

- Измерение падения напряжения на опорном датчике (`LTR216_BG_MEAS_UREF`). Используется для корректировки абсолютного значения U_{ref} , которое участвует в формуле расчета разбаланса моста. Влияет также на значения U_{ref} и U_{refR} (последнее только при 4-х проводном подключении опорного датчика) [структуры состояния модуля во время сбора](#).
- Измерение смещения нуля канала измерения напряжения на опорном датчике (`LTR216_BG_MEAS_UREF_OFFS`). Позволяет учитывать уход смещения нуля при включенном измерении `LTR216_BG_MEAS_UREF`.
- Измерение напряжения V_{adj} (`LTR216_BG_MEAS_VADJ`). Позволяет выполнять контроль данного напряжения нахождение в требуемых пределах. Результат проверки и измеренное значение сохраняются в поле V_{adj} [структуры состояния модуля во время сбора](#).
- Измерение падения напряжения в цепи U_- (`LTR216_BG_MEAS_UNEG`). Измерение учитывается в других фоновых измерениях, выполняющихся относительно земли. Результаты измерения сохраняются в поле U_{neg} [структуры состояния модуля во время сбора](#).
- Проверка цепи подключения опорного датчика из 2-х или 4-х проводов (в зависимости от схемы подключения) на короткое замыкание (`LTR216_BG_MEAS_UREF_SHORT`). Влияет на состояние измерения U_{ref} и U_{refR} (последнее только при 4-х проводном подключении опорного датчика) [структуры состояния модуля во время сбора](#).
- Проверка цепи подключения опорного датчика из 2-х или 4-х проводов (в зависимости от схемы подключения) на обрыв (`LTR216_BG_MEAS_UREF_OPEN`). Влияет на состояние измерения U_{ref} и U_{refR} (последнее только при 4-х проводном подключении опорного датчика) [структуры состояния модуля](#).
- Проверка цепи U_- на обрыв (`LTR216_BG_MEAS_UNEG_OPEN`). Влияет на состояние измерения U_{neg} [структуры состояния модуля во время сбора](#). Следует отметить, что при обрыве данной цепи все остальные измерения не могут быть корректно получены.
- Проверка цепей подключения датчиков на короткое замыкание (`LTR216_BG_MEAS_CH_SHORT`). Выполняется для каждого разрешенного в логической таблице канала. Влияет на состояние измерения U_x [структуры состояния каждого канала](#).
- Проверка цепей подключения датчиков на обрыв (`LTR216_BG_MEAS_CH_OPEN`). Выполняется для каждого разрешенного в логической таблице канала. Влияет на состояние измерения U_x [структуры состояния каждого канала](#).
- Измерение падения напряжения на тензодатчиках и проводах их подключения относительно U_- (`LTR216_BG_MEAS_CH_UX`). Выполняется для каждого разрешенного в логической таблице канала. Результат сохраняется в поле U_x [структуры состояния каждого канала](#). Данное измерение также необходимо для расчета синфазного напряжения.

- Измерение синфазного напряжения каналов измерения разностного напряжения (`LTR216_BG_MEAS_CH_CM`). Выполняется для каждого разрешенного в логической таблице канала. Результат сохраняется в поле `Ucm` структуры состояния каждого канала. Служит для контроля нахождения синфазного напряжения в требуемых для корректной работы границах. Разрешение данного значения также приводит к сохранению рассчитанных допустимых границ синфазного напряжения и сохранению их в полях `UcmMin` и `UcmMax` структуры состояния каждого канала для возможности оценки запаса по синфазному напряжению для каждого канал.

При проверке линий на короткое замыкание необходимо установить значение поля `ShortThresholdR`. Измеренное сопротивление линии сравнивается с данным порогом и в случае, если оно меньше него, считается, что обнаружено короткое замыкание. Таким образом этот порог должен быть ниже минимального значения сопротивления датчика сложенного с сопротивлением проводов его подключения для каждой проверяемой на короткое замыкание линии.

При проверке линий на обрыв, в линию подается тестовый ток (`burnout`), который приводит к зашкалу канала измерения при обрыве. Для установления этого состояния может потребоваться время, превышающее стандартный цикл опроса канала АЦП. Время это зависит от длины кабеля подключения датчиков и его погонной емкости. Значения не ниже максимальных для используемых каналов должны быть записаны в поля `CableLength` и `CableCapacityPerUnit`. По этим параметрам библиотека вычисляет необходимое время подачи тестового тока для обнаружения условия обрыва и определяет, сколько штатных циклов измерения должна занимать данная проверка в цикле опроса.

3.2.7 Настройка разрешения использования тарировки

Модуль позволяет выполнять пользователю тарировку каналов измерения, как описано в разделе [Выполнение тарировки](#). Разрешение применения тарировочных коэффициентов выполняется с помощью поля `TareEnabled` конфигурации модуля. Если данное поле не нулевое, то для всех каналов, для которых в [информации о модуле](#) присутствуют действительные тарировочные коэффициенты, эти коэффициенты будут использованы как для установки значения ЦАП для компенсации смещения нуля, так и при расчете возвращаемого значения разбаланса.

3.3 Измерение начальных параметров

После завершения конфигурации модуля и до запуска первого основного цикла сбора необходимо выполнить измерение дополнительных параметров, которые нужны для вычисления разбаланса или корректировки результатов измерений. Для этого необходимо вызвать функцию `LTR216_InitMeasParams()`, передав ей список измеряемых параметров.

Необходимость выполнения начальных измерений не зависит от разрешения выполнения аналогичных фоновых измерений, т.к. их значения нужны уже для расчета результатов начиная с первого кадра, когда фоновые измерения еще не выполнены. Таким образом, начальное измерение параметров позволяет получить начальные значения всех необходимых величин для расчета результата, начиная с первого кадра сбора, а опциональные соответствующие фоновые измерения позволяют корректировать

их значения во время сбора, компенсирую их возможный уход во время выполнения измерений.

Могут быть вычислены следующие параметры:

- Смещение нуля для каналов измерения разностных напряжений (`LTR216_INIT_MEAS_OFFS`). Данное значение может быть использовано для компенсации смещения нуля измерительных каналов. Оно не возвращает напрямую результат пользователю, но сохраняет внутри описателя код смещения нуля для каждого разрешенного канала, которые будут использованы при последующем сборе данных при обработке в `LTR216_ProcessData()`.
- Измерение напряжения на опорном датчике (`LTR216_INIT_MEAS_UREF`). Так как данное значение участвует в расчете значения разбаланса, необходимо выполнить его измерение до сбора для получения корректных результатов. Результат измерения сохраняется внутри описателя для дальнейшего использования, а также доступен через поля `Uref` и `UrefR` возвращаемой структуры состояния модуля во время сбора.

3.4 Сбор и обработка данных

В случае необходимости получения информации о результатах обработки фоновых измерений и иных проверок корректности условий измерений необходимо перед запуском создать и проинициализировать структуры, используемые для сохранения этих результатов:

- Для сохранения результатов, не связанных с конкретным каналом, нужно создать одну структуру типа `TLTR216_DATA_STATUS` и проинициализировать ее, вызвав `LTR216_DataStatusInit()`.
- Для сохранения результатов, связанных с каналами измерения, нужно создать массив из структур типа `TLTR216_DATA_CH_STATUS` размером, равным количеству логических каналов в таблице (настройка `LChCnt`) и проинициализировать его, вызвав `LTR216_DataChannelsStatusInit()`

Запуск сбора выполняется функцией `LTR216_Start()` после конфигурации модуля и измерения начальных параметров. После ее вызова модуль переходит в режим сбора данных и начинает опрос АЦП в соответствии с установленными настройками. При выполнении сбора данных недоступны никакие действия, кроме либо приема и обработки новых данных, либо останова сбора.

Для приема данных необходимо получать слова от модуля с помощью `LTR216_Recv()` со скоростью передачи их от модуля и выполнить обработку с помощью `LTR216_ProcessData()`. Обработку следует выполнять порциями, содержащими целое число кадров. Количество слов в кадре можно узнать по полю `FrameWordsCount` структуры состояния модуля (поле `State` описателя модуля), которое заполняется после вызова `LTR216_SetADC()`.

В `LTR216_ProcessData()` следует передавать созданные и проинициализированные ранее структуры для сохранения результатов фоновой обработки, если эта информация требуется. Отслеживать состояние можно двумя способами, либо анализируя состояния

измерений и флаги статуса каждой структуры, по которым можно узнать текущие результаты проверок. Либо можно проверять флаги в полях обнаружения ошибок, которые устанавливаются при их обнаружении и уже не сбрасываются без ручного обнуления пользователем, т.е. их можно использовать, чтобы проверить, что с момента начала сбора данных (или последнего ручного их сброса) не возникало ошибок измерений. Ошибки, сохраняемые в этих структурах, влияют на действительность выполняемых измерений, но не связаны с ошибками работ интерфейса модуля и не препятствуют дальнейшему сбору и обработке данных, поэтому они не влияют на код, возвращаемый самой функцией `LTR216_ProcessData()`.

По завершению обработки необходимо остановить сбор данных с помощью `LTR216_Stop()`.

3.5 Проверка линий и условий измерения

Проверку линий и условий измерений можно выполнить до сбора измерений, вызвав функцию `LTR216_CheckMeasStatus()`. Для ее вызова модуль должен быть уже сконфигурирован, но сбор данных не запущен. Функция принимает набор флагов, определяющих, какие измерения должны выполняться, аналогично тому, как задаются фоновые измерения с помощью поля `BgMeas` конфигурации модуля. При этом набор переданных измерений может отличаться от набора настроенных фоновых измерений. Например, можно не разрешать фоновые измерения, чтобы увеличить частоту кадров при сборе данных, а проверять нужные условия только перед сбором данных.

Назначение этих измерений аналогично фоновым измерениям и подробно описано в разделе [Настройки используемых фоновых измерений](#). Результаты проверок и обработки фоновых измерений возвращаются функцией `LTR216_CheckMeasStatus()` аналогично тому, как они возвращаются при включенных фоновых измерениях функцией `LTR216_ProcessData()`. Все структуры для сохранения результатов должны быть аналогично проинициализированны (можно использовать одни и те же, что при сборе, что при отдельной проверке).

3.6 Выполнение тарировки

Библиотека позволяет пользователю выполнять тарировку выбранных каналов модуля и сохранять полученные тарировочные коэффициенты в энергонезависимую память модуля. Тарировка позволяет подобрать коэффициенты всей измерительной системы в целом (датчики, кабели подключения, модуль АЦП) для получения результатов, приведенных к пользовательской шкале измерения. Кроме того, тарировка нуля позволяет скомпенсировать начальный разбаланс, превышающий используемый поддиапазон измерения в 2 раза за счет подстроечного ЦАП (± 70 мВ для диапазона ± 35 мВ и ± 140 мВ для диапазона ± 70 мВ).

При выполнении тарировки пользователь последовательно подает на вход системы величины, которые будут соответствовать сперва нулю, а затем максимальному значению шкалы в его измерительной системе. При этом уровень нуля должен быть ограничен двумя выставленными диапазонами измерения, а шкалы — в пределах диапазона от скорректированного нуля. Если требуется только скомпенсировать начальное смещение, то тарировка шкалы может не проводиться.

Тарировочные коэффициенты одного канала описываются структурой `TLTR216_TARE_CH_COEFS` и хранятся в массиве `Tare` в структуре `ModuleInfo`

[описателя модуля](#). Данная структура помимо самих значений коэффициентов имеет отдельные поля для проверки их действительности (указывающие, что была проведена успешная тарировка, в результате которой были рассчитаны эти коэффициенты), а также информацию о части конфигурационных параметров модуля, для которых она проводилась.

Полная тарировка выполняется следующим образом:

1. Заполняются настройки модуля. В логическую таблицу должны быть добавлены логические каналы, соответствующие всем физическим каналам, тарировку которых планируется провести.
2. Тарировка нуля. Для этого на каждый тарлируемый канал измерительной системы подаются уровни, соответствующие нулевому уровню пользовательской шкалы (начальный разбаланс), после чего вызывается функция [LTR216_TareOffset\(\)](#).
3. Тарировка шкалы (опционально). Для этого на каждый тарлируемый канал измерительной системы подаются уровни, соответствующие максимальному разбалансу в пользовательской шкале, после чего вызывается функция [LTR216_TareScale\(\)](#). Это значение должно быть в пределах выбранного диапазона измерения и при получении данных с использованием тарировочных коэффициентов при данном уровне сигнала будет возвращаться значение, соответствующее границе выбранного диапазона.

После завершения тарировки полученные коэффициенты сохраняются только в самой структуре описателя модуля. Они могут быть применены для последующих измерений. Для этого необходимо установить значение поля [TareEnabled](#) конфигурации модуля и вызвать [LTR216_SetADC\(\)](#), после чего следующий сбор данных будет проведен с учетом тарировочных коэффициентов. При включенной тарировке функция [LTR216_ProcessData\(\)](#) будет возвращать значения в шкале измерений, где 0 соответствует разбалансу, который был при тарировке нуля, а значение равно диапазону измерения соответствует разбалансу при тарировке шкалы.

Важно отметить, что после изменения любым способом тарировочных коэффициентов модуля необходимо повторно сконфигурировать модуль путем вызова [LTR216_SetADC\(\)](#), так как тарировочные коэффициенты влияют на конфигурацию ЦАП для компенсации смещения нуля.

Полученные тарировочные коэффициенты можно сохранить в энергонезависимую память модуля путем вызова функции [LTR216_WriteTareInfo\(\)](#). Коэффициенты автоматически считываются из энергонезависимой памяти при последующем открытии соединения с модулем через [LTR216_Open\(\)](#) и сохраняются в структуре [ModuleInfo](#) описателя модуля.

В случае, если во время работы коэффициенты в структуре модуля были изменены, но требуется вернуться снова к коэффициентам, которые были до этого записаны во flash-память, то можно прочитать коэффициенты явно, вызвав функцию [LTR216_ReadTareInfo\(\)](#).

3.7 Работа с пользовательской областью энергонезависимой памяти модуля

На модуле LTR216 присутствует энергонезависимая flash-память. Часть памяти используется для хранения стандартной информации, используемой библиотекой,

включая серийный номер модуля, заводские калибровки, а также результаты тарировки.

При этом часть этой памяти, начиная с адреса `LTR216_FLASH_USERDATA_ADDR`, размером `LTR216_FLASH_USERDATA_SIZE` отдана пользователю для возможности записи им любой своей информации.

Эта область всегда доступна на чтение с помощью функции `LTR216_FlashRead()`. При этом она изначально защищена от записи, чтобы невозможно было случайно изменить ее содержимое. Последовательность вызовов для изменения содержимого части информации в пользовательской области flash-памяти выглядит следующим образом:

1. Разрешение изменение содержимого пользовательской области flash-памяти с помощью `LTR216_FlashWriteEnable()`.
2. Стирание информации в блоке памяти, содержимое которого нужно изменить, с помощью `LTR216_FlashErase()`. Начальный адрес и размер блока должен быть кратен 256 байт.
3. Запись информации в стертую ранее область Flash-памяти с помощью `LTR216_FlashWrite()`.
4. При необходимости изменения еще одного блока можно повторить пункты 2 и 3 нужное количество раз.
5. По завершению изменения содержимого пользовательской области flash-памяти необходимо снова запретить возможность ее изменения, вызвав `LTR216_FlashWriteDisable()`.

Глава 4

Константы, типы данных и функции библиотеки

4.1 Константы и перечисления.

4.1.1 Константы и макроопределения.

Константа	Значение	Описание
LTR216_CHANNELS_CNT	16	Максимальное число измерительных каналов в одном модуле
LTR216_RANGES_CNT	2	Количество пользовательских диапазонов измерения
LTR216_CBR_RANGES_CNT	3	Количество калибруемых диапазонов, включая вспомогательные, не используемые для пользовательских измерений
LTR216_CHANNEL_MASK_ALL	0xFFFF	Маска, соответствующая всем каналам АЦП
LTR216_ADC_SCALE_CODE_MAX	8000000	Код АЦП, соответствующее максимальному пиковому значению диапазона
LTR216_ADC_CLOCK	32000000	Базовая частота, подающаяся на АЦП модуля в Гц, до делителя
LTR216_SYNC_FDIV_MAX	0xFFFFF	Максимальное значение делителя частоты синхронизации АЦП в многоканальном режиме
LTR216_ISRC_CODE_MAX	4095	Максимальное значение кода, задающего силу тока питания датчиков
LTR216_MAX_LCHANNEL	1024	Максимальное количество каналов в логической таблице
LTR216_NAME_SIZE	8	Размер поля с названием модуля
LTR216_SERIAL_SIZE	16	Размер поля с серийным номером модуля

LTR216_FLASH_USERDATA_ADDR	0x0	Адрес, с которого начинается пользовательская область flash-памяти
LTR216_FLASH_USERDATA_SIZE	0x100000	Размер пользовательской области flash-памяти
LTR216_VADJ_THRESHOLD	7	Порог для максимального значения V_{adj} , который служит для обнаружения неисправности
LTR216_SINC3_ODR_MAX_DIV	32767	Максимальное значение поля <code>AdcOdrCode</code> при использовании SINC3 фильтра
LTR216_DEFAULT_SHORT_THRESH_R	10	Значение поля <code>ShortThresholdR</code> по умолчанию, устанавливаемое при вызове <code>LTR216_Init()</code>
LTR216_CH_INTERNAL_CAPACITY	1450	Внутренняя емкость входов LTR216 в пФ. Значение используется при расчете времени проверки обрыва линии.
LTR216_DEFAULT_CABLE_CAPACITY_PER_UNIT	150	Значение поля <code>CableCapacityPerUnit</code> по умолчанию, устанавливаемое при вызове <code>LTR216_Init()</code>
LTR216_DEFAULT_CABLE_LENGTH	100	Значение поля <code>CableLength</code> по умолчанию, устанавливаемое при вызове <code>LTR216_Init()</code>
LTR216_I_BURNOUT	5e-6	Величина тестового источника тока (А) для реализации функции контроля обрыва линий

4.1.2 Коды ошибок, специфичные для модуля LTR216.

Тип: <code>e_LTR216_ERROR_CODES</code>		
Описание: Коды ошибок, которые определены и используются только в <code>ltr216api</code> . Остальные коды ошибок, которые используются разными модулями, определены в <code>ltrapi.h</code>		
Константа	Значение	Описание
LTR216_ERR_ADC_ID_CHECK	-10700	Не удалось обнаружить микросхему АЦП
LTR216_ERR_ADC_RECV_SYNC_OVERRATE	-10701	Частота синхронизации АЦП превысила частоту преобразования
LTR216_ERR_ADC_RECV_INT_CYCLE_ERROR	-10702	Ошибка внутреннего цикла чтения данных с АЦП
LTR216_ERR_ADC_REGS_INTEGRITY	-10703	Нарушена целостность регистров АЦП
LTR216_ERR_INVALID_ADC_SWMODE	-10704	Задано неверное значение режима опроса каналов АЦП

LTR216_ERR_INVALID_FILTER_TYPE	-10705	Задано неверное значение типа фильтра АЦП
LTR216_ERR_INVALID_ADC_ODR_CODE	-10706	Задано неверное значение кода, определяющего скорость преобразования АЦП
LTR216_ERR_INVALID_SYNC_FDIV	-10707	Задано неверное значение делителя частоты синхронизации АЦП
LTR216_ERR_INVALID_LCH_CNT	-10708	Задано неверное количество логических каналов
LTR216_ERR_INVALID_ISRC_CODE	-10709	Задан неверный код, определяющий силу тока питания датчиков
LTR216_ERR_CH_NOT_FOUND_IN_LTABLE	-10710	Указанный канал не был найден в логической таблице
LTR216_ERR_NO_CH_ENABLED	-10711	Ни один канал не был разрешен
LTR216_ERR_TARE_CHANNELS	-10712	Не удалось найти действительное значение при тарировке некоторых каналов
LTR216_ERR_TOO_MANY_LTABLE_CH	-10713	Превышено максимальное число каналов в основном цикле опроса АЦП
LTR216_ERR_TOO_MANY_LTABLE_BG_CH	-10714	Превышено максимальное число каналов в фоновом цикле опроса АЦП
LTR216_ERR_UNSUUF_SW_TIME	-10715	Полученное время на коммутацию меньше заданного предела
LTR216_ERR_BAD_INIT_MEAS_STATUS	-10716	Измеренное значение начального параметра недействительно
LTR216_ERR_INVALID_CH_RANGE	-10717	Задан неверный код диапазона канала АЦП
LTR216_ERR_INVALID_CH_NUM	-10718	Задан неверный номер физического канала АЦП

4.1.3 Диапазоны измерения модуля LTR216

Тип: e_LTR216_RANGE		
Описание: Диапазоны измерения модуля LTR216		
Константа	Значение	Описание
LTR216_RANGE_35	0	Диапазон ± 35 мВ
LTR216_RANGE_70	1	Диапазон ± 70 мВ

4.1.4 Типы фильтров АЦП модуля LTR216

Тип: e_LTR216_FILTER_TYPE		
Описание: Типы фильтров АЦП модуля LTR216		
Константа	Значение	Описание
LTR216_FILTER_SINC5_1	0	Каскад из SINC5 и SINC1 фильтра
LTR216_FILTER_SINC3	1	SINC3 фильтр
LTR216_FILTER_ENH_50HZ	2	Фильтр с усиленной фильтрацией частоты 50 Гц

4.1.5 Режим коммутации каналов модуля

Тип: e_LTR216_ADC_SWMODE		
Описание: Режим коммутации каналов модуля		
Константа	Значение	Описание
LTR216_ADC_SWMODE_MULTICH_SYNC	0	Многоканальный режим. В нем коммутация каналов происходит с частотой, определяемой настройкой SyncFreqDiv .
LTR216_ADC_SWMODE_SIGNLECH_CONT	1	Одноканальный режим без коммутации.

4.1.6 Флаги конфигурации

Тип: e_LTR216_CONFIG_FLAGS		
Описание: Флаги, которые могут быть установлены в поле Flags структуры конфигурации модуля, для определения специальных режимов конфигурации или поведения модуля.		
Константа	Значение	Описание
LTR216_CONFIG_FLAG_RAWTABLE	1 << 0	Признак, что нужно использовать низкоуровневые настройки из RawCfg , а не рассчитывать их на основе пользовательской конфигурации. Не используется в штатных режимах измерения.

4.1.7 Фоновые измерения

Тип: e_LTR216_BG_MEAS		
<p>Описание: Флаги, определяющие, какие вспомогательные измерения будут выполняться. Каждый флаг соответствует одному измерению. Для задания нескольких измерений флаги каждого измерения объединяются через логическое “ИЛИ”. Также в данном перечислении определены группы измерений, названия которых начинаются с LTR216_BG_MEASGROUP, включающие набор однотипных измерений. Данные флаги используются при настройке фоновых измерений (поле BgMeas), а также при выполнении отдельной проверки состояния линий и условий измерения с помощью LTR216_CheckMeasStatus().</p>		
Константа	Значение	Описание
LTR216_BG_MEAS_OFFS	$1 \ll 0$	Измерение смещения нуля каналов измерения разностного напряжения
LTR216_BG_MEAS_UREF	$1 \ll 1$	Измерение падения напряжения на опорном тензодатчике U_{ref}
LTR216_BG_MEAS_UREF_OFFS	$1 \ll 2$	Измерение смещения нуля для канала измерения U_{ref}
LTR216_BG_MEAS_VADJ	$1 \ll 3$	Измерение напряжения V_{adj}
LTR216_BG_MEAS_UNEG	$1 \ll 4$	Измерение напряжения U_- относительно земли
LTR216_BG_MEAS_UREF_SHORT	$1 \ll 5$	Проверка КЗ канала измерения U_{ref}
LTR216_BG_MEAS_UREF_OPEN	$1 \ll 6$	Проверка обрыва канала измерения U_{ref}
LTR216_BG_MEAS_UNEG_OPEN	$1 \ll 7$	Проверка обрыва линии U_-
LTR216_BG_MEAS_CH_SHORT	$1 \ll 8$	Проверка КЗ каналов измерения разностного напряжения
LTR216_BG_MEAS_CH_OPEN	$1 \ll 9$	Проверка обрыва каналов измерения разностного напряжения
LTR216_BG_MEAS_CH_UX	$1 \ll 10$	Измерение абсолютного значения падения напряжения на тензодатчиках в измерительных каналах
LTR216_BG_MEAS_CH_CM	$1 \ll 11$	Измерение синфазного напряжения каналов

LTR216_BG_MEASGROUP_OPEN	LTR216_BG_MEAS_CH_OPEN LTR216_BG_MEAS_UNEG_OPEN LTR216_BG_MEAS_UREF_OPEN	Группа из всех измерений для проверки обрыва различных цепей
LTR216_BG_MEASGROUP_SHORT	LTR216_BG_MEAS_CH_SHORT LTR216_BG_MEAS_UREF_SHORT	Группа из всех измерений для проверки КЗ различных цепей
LTR216_BG_MEASGROUP_SECTBL	LTR216_BG_MEAS_UREF LTR216_BG_MEAS_UREF_OFFS LTR216_BG_MEAS_VADJ LTR216_BG_MEAS_UNEG LTR216_BG_MEAS_CH_UX LTR216_BG_MEAS_CH_CM LTR216_BG_MEASGROUP_OPEN LTR216_BG_MEASGROUP_SHORT	Группа измерений, которые выполняются относительно земли в фоновом цикле опроса
LTR216_BG_MEASGROUP_UREF_CHECK	LTR216_BG_MEAS_UREF_SHORT LTR216_BG_MEAS_UREF_OPEN LTR216_BG_MEAS_UNEG_OPEN	Группа измерений для полной проверки целостности линий измерения U_{ref}
LTR216_BG_MEASGROUP_CH_CHECK	LTR216_BG_MEAS_CH_SHORT LTR216_BG_MEAS_CH_OPEN LTR216_BG_MEAS_CH_UX LTR216_BG_MEAS_CH_CM	Группа измерений для полной проверки целостности линий измерительных каналов
LTR216_BG_MEASGROUP_ALL	LTR216_BG_MEAS_OFFS LTR216_BG_MEASGROUP_SECTBL	Все фоновые измерения

4.1.8 Начальные измерения

Тип: e_LTR216_INIT_MEAS		
Описание: Флаги, определяющие, какие предварительные измерения будут выполнены при вызове <code>LTR216_InitMeasParams()</code> . Для указания всех возможных начальных измерений можно использовать константу <code>LTR216_INIT_MEAS_ALL</code> .		
Константа	Значение	Описание
LTR216_INIT_MEAS_OFFS	1 << 0	Измерение смещения нуля каналов измерения разностного напряжения
LTR216_INIT_MEAS_UREF	1 << 1	Измерение падения напряжения на опорном тензодатчике U_{ref}
LTR216_INIT_MEAS_ALL	LTR216_INIT_MEAS_OFFS LTR216_INIT_MEAS_UREF	Все возможные предварительные измерения

4.1.9 Состояние вспомогательного измерения

Тип: e_LTR216_MEAS_STATUS		
Описание: Константы определяют состояние соответствующего вспомогательного измерения. Проверка корректности работы и выполненных измерений может выполняться по анализу этого состояния для каждого разрешенного измерения. Если измерение не разрешено, то его состояние будет всегда соответствовать значению <code>LTR216_MEASSTATUS_NOT_USED</code> . Если же разрешено, то изначально оно будет в состоянии <code>LTR216_MEASSTATUS_NOT_INIT</code> . После получения необходимых данных для расчета значения измерения, его состояние изменится на <code>LTR216_MEASSTATUS_OK</code> , что соответствует состоянию при нормальной работе модуля, или в одно из состояний, соответствующих обнаруженной проблеме.		
Константа	Значение	Описание
LTR216_MEASSTATUS_NOT_USED	0	Измерение не используется в текущей конфигурации
LTR216_MEASSTATUS_OK	1	Получено действительное значение измерения
LTR216_MEASSTATUS_NOT_INIT	2	Измерение разрешено, но еще не были получены данные для расчета его значения
LTR216_MEASSTATUS_ADC_OVERRANGE	3	При выполнении данного измерения произошло переполнение разрядной сетки АЦП. Значение измерения не действительно
LTR216_MEASSTATUS_SHORT	4	Обнаружено КЗ цепи, соответствующей данному измерению. Значение измерения не действительно

LTR216_MEASSTATUS_OPEN	5	Обнаружен обрыв цепи, соответствующей данному измерению. Значение измерения не действительно
LTR216_MEASSTATUS_BAD_VALUE_RANGE	6	Фоновое измерение получено и его значение действительно, однако оно выходит за допустимый в нормальном режиме работы модуля диапазон
LTR216_MEASSTATUS_CANT_CALC	7	Фоновое измерение не может быть вычислено. Как правило это связано с тем, что не действительно другое измерение, которое необходимо для вычисления текущего

4.1.10 Флаги обработки данных

Тип: e_LTR216_PROC_FLAGS		
Описание: Набор флагов, которые влияют на работу функции LTR216_ProcessData() .		
Константа	Значение	Описание
LTR216_PROC_FLAG_MEAS_UNBALANCE	0 << 16	Флаг указывает, что в качестве результата будет возвращено значение разбаланса тензометрического полумоста, приведенного к номинальному опорному напряжению (2,5 В). Это основной режим измерения модуля и он используется по умолчанию, когда не передан ни один флаг, определяющий тип выходного измерения.
LTR216_PROC_FLAG_MEAS_UDIFF	1 << 16	Флаг указывает, что в качестве результата будет возвращено разностное напряжение между падением на измеряемом и опорном тензодатчиках
LTR216_PROC_FLAG_MEAS_UDIFF_CODE	2 << 16	Флаг указывает, что в качестве результата будет возвращено значение кода АЦП при измерении разностного напряжения между падением на измеряемом и опорном тензодатчиках (без перевода в Вольты)

LTR216_PROC_FLAG_NO_FABRIC_CBR	1 << 0	Флаг указывает, что не нужно применять заводские калибровочные коэффициенты. В штатном режиме работы они применяются всегда, независимо от того, используется ли пользовательская тарировка или нет
LTR216_PROC_FLAG_NONCONT_DATA	0x0100	По умолчанию LTR216_ProcessData() предполагает, что ей на обработку передаются все принятые данные и проверяет непрерывность счетчика не только внутри переданного блока данных, но и между вызовами. Если обрабатываются не все данные или одни и те же данные обрабатываются повторно, то нужно указать данный флаг, чтобы счетчик проверялся только внутри обрабатываемого блока

4.1.11 Флаги состояния канала при обработке данных

Тип: e_LTR216_CH_STATUS_FLAGS		
Описание: Дополнительные флаги, определяющие состояние канала за обрабатываемый функцией LTR216_ProcessData() блок данных, помимо состояний, определяемых фоновыми измерениями.		
Константа	Значение	Описание
LTR216_CH_STATUS_FLAG_OVERRANGE	1 << 0	Признак, что произошло переполнение разрядной сетки АЦП при измерении разностного напряжения данного канала

4.1.12 Флаги обнаруженных ошибок канала при обработке данных

Тип: e_LTR216_CH_DETECT_ERRS		
Описание: Данные флаги устанавливаются при обнаружении ошибок в статусе канала или фоновых измерений данного канала. В отличие от статусов, определяющих текущее состояние канала, данные флаги указывают, что отклонение было хотя бы раз за время обработки и сбрасываются только вручную пользователем.		
Константа	Значение	Описание
LTR216_CH_DETECT_ERR_OVERRANGE	1 << 0	Признак, что возникало переполнение разрядной сетки АЦП при измерении разностного напряжения для данного канала
LTR216_CH_DETECT_ERR_UCM_OUT_OF_RANGE	1 << 1	Признак, что возникал выход синфазного напряжения за пределы диапазона
LTR216_CH_DETECT_ERR_UX_OPEN	1 << 2	Признак, что возникал обрыв линии измерения разностного напряжения
LTR216_CH_DETECT_ERR_UX_SHORT	1 << 3	Признак, что возникало КЗ линии измерения разностного напряжения

4.1.13 Флаги состояния модуля при обработке данных

Тип: e_LTR216_STATUS_FLAGS		
Описание: Дополнительные флаги, определяющие состояние модуля за обрабатываемый функцией LTR216_ProcessData() блок данных, помимо состояний, определяемых фоновыми измерениями.		
Константа	Значение	Описание
LTR216_STATUS_FLAG_LOAD_POW_EXCEEDED	1 << 0	Признак наличия превышения допустимого предела мощности нагрузки

4.1.14 Флаги обнаруженных ошибок модуля при обработке данных

Тип: e_LTR216_DETECT_ERRS		
Описание: Данные флаги устанавливаются при обнаружении ошибок в статусе модуля или фоновых измерений не связанных с конкретным каналом. В отличие от статусов, определяющих текущее состояние модуля, данные флаги указывают, что отклонение было хотя бы раз за время обработки и сбрасываются только вручную пользователем.		
Константа	Значение	Описание
LTR216_DETECT_ERR_LOAD_POW_EXCEEDED	1 << 0	Признак, что было обнаружено превышение допустимого предела мощности нагрузки.
LTR216_DETECT_ERR_UREF_OVERRANGE	1 << 1	Признак, что был обнаружен выход за шкалу АЦП при измерении напряжения на опорном датчике.
LTR216_DETECT_ERR_UREF_OPEN	1 << 2	Признак, что был обнаружен обрыв канала измерения напряжения на опорном датчике.
LTR216_DETECT_ERR_UREF_SHORT	1 << 3	Признак, что было обнаружено КЗ канала измерения напряжения на опорном датчике.
LTR216_DETECT_ERR_VADJ_ERROR	1 << 4	Признак, что был обнаружен выход за допустимый диапазон измеренного значения V_{adj} .
LTR216_DETECT_ERR_UNEG_OPEN	1 << 5	Признак, что был обнаружен обрыв линии U_- .

4.1.15 Действительность информации в энергонезависимой памяти модуля

Тип: e_LTR216_FLASH_INFO_FLAGS		
Описание: Набор флагов, указывающий, какая информация была обнаружена в энергонезависимой flash-памяти и была прочитана из нее.		
Константа	Значение	Описание
LTR216_FLASH_INFO_FLAG_SERIAL_VALID	1 << 0	Признак действительности серийного номера
LTR216_FLASH_INFO_FLAG_ISRC_CBR_VALID	1 << 1	Признак действительности калибровки источников тока питания датчиков.
LTR216_FLASH_INFO_FLAG_DAC_CBR_VALID	0x7 << 8	Признак действительности калибровки ЦАП (бит на диапазон).
LTR216_FLASH_INFO_FLAG_ADC_CBR_VALID	0x7 << 12	Признак действительности калибровки АЦП (бит на диапазон).

4.2 Типы данных.

4.2.1 Калибровочные коэффициенты для одного канала.

Тип: TLTR216_CBR_VALUE		
Описание: Калибровочные коэффициенты для одного канала.		
Поле	Тип	Описание поля
Offset	double	Смещение нуля.
Scale	double	Множитель шкалы.

4.2.2 Калибровочные коэффициенты источников тока питания датчиков

Тип: TLTR216_ISRC_CBR		
Описание: Структура хранит заводские калибровочные коэффициенты для источников тока питания датчиков.		
Поле	Тип	Описание поля
Ref	TLTR216_CBR_VALUE	Калибровочные коэффициенты для источника тока питания опорного датчика
Ch	TLTR216_CBR_VALUE [LTR216_CHANNELS_CNT]	Калибровочные коэффициенты для остальных источников тока. Не калибруются, всегда смещение равно 0, а шкала 1. Резерв для возможного дальнейшего использования.

4.2.3 Заводские калибровочные коэффициенты

Тип: TLTR216_FABRIC_CBR		
Описание: Структура хранит все заводские калибровочные коэффициенты модуля, которые хранятся в энергонезависимой памяти.		
Поле	Тип	Описание поля
Adc	TLTR216_CBR_VALUE [LTR216_CBR_RANGES_CNT]	Калибровочные коэффициенты для каналов АЦП в разных диапазонах измерения
Dac	TLTR216_CBR_VALUE [LTR216_CBR_RANGES_CNT]	Калибровочные коэффициенты для каналов ЦАП в разных диапазонах измерения АЦП
ISrc	TLTR216_ISRC_CBR	Калибровочные коэффициенты для источников тока питания датчиков
Reserved	double [48]	Резерв

4.2.4 Настройки тарировки канала.

Тип: TLTR216_TARE_CH_CFG		
Описание: Структура содержит в себе часть настроек модуля, влияющих на тарировку, при которых выполнена тарировка канала.		
Поле	Тип	Описание поля
ISrcCode	DWORD	Код величины источника тока питания датчиков.
Range	BYTE	Диапазон измерения АЦП. Одно из значений e_LTR216_RANGE .
Reserved	DWORD [6]	Резерв.

4.2.5 Информация о тарировке одного канала.

Тип: TLTR216_TARE_CH_COEFS		
Описание: Структура содержит информацию о тарировке канала: сами коэффициенты, признаки их действительности, а также информацию о режиме, в котором тарировка была выполнена.		
Поле	Тип	Описание поля
DacValid	BOOLEAN	Признак, присутствует ли действительное значение ЦАП.
OffsetValid	BOOLEAN	Признак, присутствует ли действительное значение коэффициента смещения нуля.
ScaleValid	BOOLEAN	Признак, присутствует ли действительное значение коэффициента шкалы.
Reserved1	DWORD	Резерв.
DacValue	double	Напряжение для компенсации смещения нуля, выставяемое на ЦАП.
Offset	double	Коэффициент смещения нуля.
Scale	double	Коэффициент шкалы.
Reserved2	DWORD [8]	Резерв.
Cfg	TLTR216_TARE_CH_CFG	Настройки, при которых выполнена тарировка.

4.2.6 Информация о модуле.

Тип: TLTR216_MODULE_INFO		
Описание: Структура, содержащая информацию о версиях прошивок микросхем модуля и информацию, считанную из Flash-памяти модуля (серийный номер, калибровочные коэффициенты). Все поля заполняются при вызове LTR216_Open() . Значение тарировочных коэффициентов изменяется также при вызове функций тарировки.		
Поле	Тип	Описание поля
Name	CHAR [LTR216_NAME_SIZE]	Название модуля. Оканчивающуюся нулем ASCII-строка ("LTR216").
Serial	CHAR [LTR216_SERIAL_SIZE]	Серийный номер модуля. Оканчивающуюся нулем ASCII-строка.
VerFPGA	WORD	Версия прошивки ПЛИС.

Reserved	DWORD [4]	Резерв.
FlashInfoFlags	DWORD	Набор флагов из <code>e_LTR216_FLASH_INFO_FLAGS</code> , объединенных по ИЛИ, указывающих, какая информация содержалась и была прочитана из flash-памяти модуля
FabricCbr	<code>TLTR216_FABRIC_CBR</code>	Заводские калибровочные коэффициенты, считанные из flash-памяти модуля.
Tare	<code>TLTR216_TARE_CH_COEFS</code> [<code>LTR216_CHANNELS_CNT</code>]	Тарировочные коэффициенты, считанные из flash-памяти, или вычисленные при выполнении тарировки.

4.2.7 Параметры текущего состояния модуля.

Тип: <code>TLTR216_STATE</code>		
Описание: Структура, содержащая параметры модуля, которые пользователь должен использовать только для чтения, так как они изменяются только внутри функций <code>ltr216api</code> .		
Поле	Тип	Описание поля
Configured	BOOLEAN	Признак, что была выполнена конфигурация устройства через <code>LTR216_SetADC()</code> с момента установления соединения.
Run	BOOLEAN	Признак, запущен ли сбор данных.
FrameWordsCount	DWORD	Количество слов в одном кадре. В <code>LTR216_ProcessData()</code> нужно передавать количество слов, кратное этому значению. Обновляется после вызова <code>LTR216_SetADC()</code> .
AdcFreq	double	Установленная частота АЦП. Обновляется после вызова <code>LTR216_SetADC()</code> .
FrameFreq	double	Установленная частота следования кадров (соответствует частоте на логический канал). Обновляется после вызова <code>LTR216_SetADC()</code> .
ISrcValue	double	Установленное значение источника тока питания датчиков в мА. Обновляется после вызова <code>LTR216_SetADC()</code> .
Reserved	DWORD [16]	Резерв

4.2.8 Настройки логического канала

Тип: TLTR216_LCHANNEL		
Описание: Структура определяет настройки одного цикла опроса АЦП при пользовательских измерениях разностного напряжения (в цепи последовательных измерений, определяемых логической таблицей).		
Поле	Тип	Описание поля
PhyChannel	BYTE	Номер физического канала (входа на разъеме), для которого происходит измерение разностного напряжения
Range	BYTE	Используемый диапазон измерения напряжения. Значение из e_LTR216_RANGE
Reserved	WORD	Резерв

4.2.9 Настройки модуля.

Тип: TLTR216_CONFIG		
Описание: Структура содержит все настройки модуля, которые должен заполнить пользователь перед вызовом LTR216_SetADC() . Для заполнения части настроек существуют дополнительные вспомогательные функции подбора нужного значения.		
Поле	Тип	Описание поля
AdcSwMode	BYTE	Режим коммутации АЦП (одноканальный или многоканальный). Значение из e_LTR216_ADC_SWMODE .
SyncFreqDiv	DWORD	Делитель частоты синхронизации АЦП. В многоканальном режиме определяет частоту коммутации АЦП, которая получается делением частоты LTR216_ADC_CLOCK на (значение делителя + 1). Может заполняться функцией LTR216_FillSyncFreqDiv() . Может принимать значение до LTR216_SYNC_FDIV_MAX . Не имеет значения в одноканальном режиме.
FilterType	DWORD	Тип фильтра, который будет использоваться в АЦП. Значение из e_LTR216_FILTER_TYPE . Может заполняться с помощью LTR216_FillFilterParams() .

AdcOdrCode	DWORD	Код, определяющий время выполнения преобразования АЦП. Зависит от выбранного типа фильтра (поле FilterType). В одноканальном режиме определяет частоту АЦП, в многоканальном определяет, какое время цикла коммутации занимает преобразование, а какое — коммутация. Может заполняться с помощью LTR216_FillFilterParams() .
AdcMinSwTimeUs	double	Необходимое время в мкс, которое требуется для коммутации каналов в многоканальном режиме. Зависит от параметров кабеля подключения и датчиков. Частота АЦП и параметры фильтра должны быть выбраны таким образом, чтобы оставшееся время на коммутацию было не менее указанного параметра, что проверяется при вызове LTR216_SetADC() . Не имеет значения в одноканальном режиме.
AdcReqFrameFreq	double	Данное поле определяет желаемое значение частоты следования кадров (частоты на логический канал) в Гц. Если данное значение больше нуля, то при расчете настроек при вызове LTR216_SetADC() подбирается количество дополнительных холостых циклов таким образом, чтобы результирующая частота была максимально близкой к данному значению (но не был превышен максимальный размер цикла опроса). Если значение меньше или равно 0 (по умолчанию), то используется максимальная частота при заданных настройках. Не имеет значения в одноканальном режиме.
ISrcCode	DWORD	Код, задающий значение тока питания датчиков. Для заполнения этого поля можно использовать функцию LTR216_FillISrcCode() .

BgMeas	DWORD	Набор флагов из e_LTR216_BG_MEAS , определяющий, какие дополнительные измерения будут выполняться на фоне основных измерений во время сбора данных. Не имеет значения в одноканальном режиме.
Ch16ForUref	BOOLEAN	Признак использования 16-го канала для измерения падения напряжения на опорном тензодатчике по 4-х проводной схеме. Если истина, то используется основная схема измерения с подключением опорного тензодатчика по 4-х проводной схеме с использованием 16 канала. При этом для основных измерений доступны только первые 15 каналы. Если ложь — то опорный тензорезистор подключается по 2-х проводной схеме, а 16-ый канал может быть использован для подключения еще одного тензодатчика.
TareEnabled	BOOLEAN	Признак использования тарифовочных коэффициентов. Если истина, то для каналов, для которых действительны тарифовочные коэффициенты, их значения будут использованы для настройки ЦАП, а также для расчета результирующих значений в LTR216_ProcessData() .
Flags	DWORD	Дополнительные флаги, определяющие особые режимы конфигурации. Задается объединением по или констант из e_LTR216_CONFIG_FLAGS . В текущей версии в штатных режимах работы не используется.
LChCnt	DWORD	Количество используемых логических каналов в логической таблице LChTbl .
LChTbl	TLTR216_LCHANNEL [LTR216_MAX_LCHANNEL]	Логическая таблица. Определяет, какие каналы модуля должны быть опрошены в каком порядке и с какими настройками,

RrefWireResistance	double	<p>Сопротивление проводов подключения опорного датчика в Омах от модуля до датчика (в одну сторону).</p> <p>Может быть задано в 16-канальном режиме (запрещен Ch16ForUref) для компенсации падения напряжения на проводах при измерении напряжения на опорном датчике по 2-х проводной схеме.</p> <p>При 4-х проводной схеме подключения опорного датчика не используется.</p>
ShortThresholdR	double	<p>Порог сопротивления (Ом) для проверки КЗ линии. КЗ считается при измеренном сопротивлении линии ниже данного порога.</p>
CableLength	double	<p>Максимальная длина кабеля (м), использующегося для подключения датчиков. Используется только при проверке обрыва линий.</p>
CableCapacityPerUnit	double	<p>Максимальная погонная емкость кабеля (пФ/м) подключения датчиков. Используется только при проверке обрыва линий.</p>
Reserved	double [32]	Резерв

4.2.10 Низкоуровневые настройки модуля

Тип: TLTR216_RAWCONFIG		
<p>Описание: Данная структура описывает реальный цикл опроса АЦП, включая как пользовательские измерения, соответствующие логической таблицы пользовательской конфигурации, так и служебные фоновые измерения и холостые такты для их согласования.</p> <p>Включает в себя таблицы, описывающие два цикла опроса. Основной выполняется каждый раз полностью за каждый кадр, а фоновый цикл выполняется за несколько кадров.</p> <p>Как правило пользователю не требуется использовать эту структуру — она автоматически заполняется при вызове <code>LTR216_SetADC()</code>.</p> <p>Поля данной структуры можно использовать для информации о временных параметрах получившегося реального цикла опроса АЦП.</p>		
Поле	Тип	Описание поля
UserChCnt	DWORD	Количество логических каналов, соответствующих пользовательским измерениям из логической таблицы конфигурации модуля .
ProcChCnt	DWORD	Общее количество обрабатываемых каналов из основного цикла опроса (без холостых тактов).
MainLChCnt	DWORD	Количество действительных элементов в массиве <code>MainLChTbl</code> . Включает общее число измерений в основном цикле (без учета длительности шага фоновой таблицы).
BgStepSize	DWORD	Количество измерений, которое занимает выполнение одного шага фоновой таблицы.
BgStepsCnt	DWORD	Количество шагов, за которое будет пройдена вся фоновая таблица.
BgFrameStepCnt	DWORD	Количество шагов фоновой таблицы в одном кадре.
BgLChCnt	DWORD	Количество элементов в <code>BgLChTbl</code> .
BgBoutLChCnt	DWORD	Количество каналов с включенным током проверки обрыва в фоновой таблице.
FrameSize	DWORD	Общее количество измерений в одном кадре .
Reserved	DWORD [7]	Резерв.
MainLChTbl	DWORD [LTR216_MAX_LCHANNEL]	Массив, определяющий последовательность измерений в основном цикле опроса
BgLChTbl	DWORD [LTR216_MAX_LCHANNEL]	Массив, определяющий последовательность измерений в фоновом цикле опроса

4.2.11 Описатель модуля модуля.

Тип: TLTR216		
<p>Описание: Структура, которая хранит всю необходимую информацию о модуле и связи с ним: текущие настройки модуля, информацию о его состоянии, структуру канала связи и т.д.</p> <p>При работе с несколькими модулями одновременно для каждого модуля должен быть создан свой описатель.</p> <p>Передается в большинство функций библиотеки. Некоторые поля структуры доступны для изменения пользователем для настройки параметров модуля. Перед использованием требует инициализации с помощью функции LTR216_Init().</p>		
Поле	Тип	Описание поля
Size	INT	Размер структуры. Заполняется в LTR216_Init() .
Channel	TLTR	Структура, содержащая состояние клиентского соединения со службой ltrd. Не используется напрямую пользователем.
Internal	PVOID	Указатель на непрозрачную структуру с внутренними параметрами, используемыми исключительно библиотекой и недоступными для пользователя.
Cfg	TLTR216_CONFIG	Настройки модуля. Заполняются пользователем перед вызовом LTR216_SetADC() .
RawCfg	TLTR216_RAWCONFIG	Более детальные низкоуровневые настройки коммутации модуля. Рассчитываются из основных настроек и заполняются автоматически при вызове LTR216_SetADC() .
State	TLTR216_STATE	Состояние модуля и рассчитанные параметры. Поля изменяются функциями библиотеки. Пользовательской программой могут использоваться только для чтения.
ModuleInfo	TLTR216_MODULE_INFO	Информация о модуле.

4.2.12 Параметры фильтра

Тип: TLTR216_FILTER_OUT_PARAMS		
<p>Описание: Структура содержит расчетные параметры фильтра, полученные при заданных настройках.</p> <p>Поля <code>FilterType</code> и <code>FilterType</code> определяют настройки фильтра, расчетные параметры для которых содержит данная структура. Эти поля соответствуют одноименным полям структуры конфигурации модуля.</p> <p>Остальные поля содержат параметры фильтра, соответствующие указанным настройкам.</p>		
Поле	Тип	Описание поля
<code>FilterType</code>	DWORD	Тип фильтра, которому соответствуют рассчитанные параметры. Значение из <code>e_LTR216_FILTER_TYPE</code> .
<code>AdcOdrCode</code>	DWORD	Код, определяющий время выполнения преобразования АЦП, для которого получены рассчитанные параметры.
<code>Odr</code>	double	Частота выдачи данных АЦП в Гц при его непрерывной работе. Эта частота является обратной величиной времени преобразования АЦП.
<code>NotchFreq</code>	double	Первая частота режекции фильтра. На ней достигается наибольшее подавление шумов.
<code>NotchDB</code>	double	Уровень подавления на частоте режекции в децибелах, если известен. Возвращается только для фильтра типа <code>LTR216_FILTER_ENH_50HZ</code> .
<code>Reserved</code>	double [4]	Резерв.

4.2.13 Состояние фонового измерения.

Тип: TLTR216_MEASSTATE		
<p>Описание: Данная структура содержит в себе информацию о состоянии и значении одного фонового измерения.</p>		
Поле	Тип	Описание поля
<code>Status</code>	BYTE	Состояние измерения. Значение из <code>e_LTR216_MEAS_STATUS</code>
<code>ValueValid</code>	BOOLEAN	Признак, действительно ли значение в поле <code>Value</code>
<code>Reserved</code>	BYTE [6]	Резерв
<code>Value</code>	double	Значение фонового измерения. Действительно только при установленном флаге <code>ValueValid</code> .

4.2.14 Состояние модуля во время сбора данных

Тип: TLTR216_DATA_STATUS		
Описание: В данной структуре сохраняется информация об обнаруженных проблемах и результаты анализа фоновых измерений, относящиеся ко всему модулю. Данная структура должна быть изначально проинициализирована с помощью <code>LTR216_DataStatusInit()</code> , после чего она может быть использована в функциях <code>LTR216_ProcessData()</code> и <code>LTR216_CheckMeasStatus()</code> для возвращения результатов анализа фоновых измерений.		
Поле	Тип	Описание поля
DetectedErrors	DWORD	При обнаружении проблем, не относящихся к отдельным каналам, в данном поле устанавливаются соответствующие флаги из <code>e_LTR216_DETECT_ERRS</code> . Данные флаги могут быть сброшены только вручную пользователем, т.е. они описывают возникла ли проблема с момента инициализации структуры или ручного сброса данного поля.
StatusFlags	DWORD	Объединение по “ИЛИ” флагов <code>e_LTR216_STATUS_FLAGS</code> , описывающих состояние модуля на текущий момент, которое не описывается состоянием фоновых измерений.
Uref	<code>TLTR216_MEASSTATE</code>	Текущее состояние измерения падения напряжения на опорном датчике и проводах до него.
UrefR	<code>TLTR216_MEASSTATE</code>	Текущее состояние измерения падения напряжения только на опорном датчике. Данное измерение в 4-х проводной схеме подключения опорного датчика измеряется явно, а признаки обрыва и КЗ характеризуют состояние линий 16-го канала, которые используются в этом режиме именно для измерения данного значения. При 2-х проводной схеме подключения опорного датчика данное измерение действительно только если задано значение <code>RrefWireResistance</code> больше нуля и вычисляется на основе <code>Uref</code> .
Vadj	<code>TLTR216_MEASSTATE</code>	Текущее состояние измерения напряжения V_{adj} с учетом проверки порога.
Uneg	<code>TLTR216_MEASSTATE</code>	Измерение падения напряжения на линии U_-
Reserver	DWORD [16]	Резерв

4.2.15 Состояние канала во время сбора данных

Тип: TLTR216_DATA_CH_STATUS		
<p>Описание: В данной структуре сохраняется информация об обнаруженных проблемах и результаты анализа фоновых измерений, относящиеся к отдельному измерительному каналу.</p> <p>Используется массив данных структур (по структуре на канал), количество элементов которого должно соответствовать количеству каналов в логической таблице. Данный массив должен быть изначально проинициализирован с помощью <code>LTR216_DataChannelsStatusInit()</code>, после чего может быть использована в функциях <code>LTR216_ProcessData()</code> и <code>LTR216_CheckMeasStatus()</code> для возвращения результатов анализа фоновых измерений.</p>		
Поле	Тип	Описание поля
DetectedErrors	DWORD	При обнаружении проблем, относящихся к отдельным каналам, в данном поле устанавливаются соответствующие флаги из <code>e_LTR216_CH_DETECT_ERRS</code> . Данные флаги могут быть сброшены только вручную пользователем, т.е. они описывают возникла ли проблема с момента инициализации структуры или ручного сброса данного поля.
StatusFlags	DWORD	Объединение по “ИЛИ” флагов <code>e_LTR216_CH_STATUS_FLAGS</code> , описывающих состояние канала на текущий момент, которое не описывается состоянием фоновых измерений.
Ures	<code>TLTR216_MEASSTATE</code>	Состояние результирующего измерения (в штатном режиме — разбаланса). Действительность зависит от действительности измерений, на основе которых производится вычисление, а значение, при его действительности, соответствует последнему результату по данному каналу
Ux	<code>TLTR216_MEASSTATE</code>	Состояние измерения абсолютного напряжения на тензодатчике, подключенному к каналу, и сопротивлению проводов до датчика.
Ucm	<code>TLTR216_MEASSTATE</code>	Рассчитанное значение синфазного напряжения для данного канала
UcmMin	double	Минимально допустимое значение синфазного напряжения для данного канала. Действительно только при действительном значении <code>Ucm</code> .
UcmMax	double	Максимально допустимое значение синфазного напряжения для данного канала. Действительно только при действительном значении <code>Ucm</code> .

Reserver	DWORD [16]	Резерв.
----------	------------	---------

4.3 Функции

4.3.1 Функции инициализации и работы с соединением с модулем.

4.3.1.1 Инициализация описателя модуля.

Формат: INT LTR216_Init (TLTR216 *hnd)
Описание: Функция инициализирует поля структуры описателя модуля значениями по умолчанию. Эта функция должна вызываться для каждой структуры TLTR216 перед вызовом остальных функций.
Параметры: hnd — Описатель модуля
Возвращаемое значение: Код ошибки .

4.3.1.2 Открытие соединения с модулем.

Формат: INT LTR216_Open (TLTR216 *hnd, DWORD ltrd_addr, WORD ltrd_port, const CHAR *csn, INT slot)
Описание: Функция устанавливает соединение с модулем в соответствии с переданными параметрами, проверяет наличие модуля и считывает информацию о нем. Должна быть вызвана перед работой с модулем. После завершения работы необходимо закрыть соединение с помощью LTR216_Close() .
Параметры: hnd — Описатель модуля ltrd_addr — IP-адрес машины, на которой запущена служба ltrd, в 32-битном формате (описан в разделе “Формат задания IP-адресов” руководства для библиотеки ltrapi). Если служба ltrd запущена на той же машине, что и программа, вызывающая данную функцию, то в качестве адреса можно передать LTRD_ADDR_DEFAULT. ltrd_port — TCP-порт для подключения к службе ltrd. По умолчанию используется LTRD_PORT_DEFAULT. csn — Серийный номер крейта, в котором находится интересующий модуль. Представляет собой оканчивающуюся нулем ASCII-строку. Для соединения с первым найденным крейтом можно передать пустую строку или нулевой указатель. slot — Номер слота крейта, в котором установлен интересующий модуль. Значение от LTR_CC_CHNUM_MODULE1 до LTR_CC_CHNUM_MODULE16.
Возвращаемое значение: Код ошибки .

4.3.1.3 Закрытие соединения с модулем.

Формат: INT LTR216_Close (TLTR216 *hnd)
Описание: Функция закрывает ранее открытое с помощью LTR216_Open() соединение. Должна вызываться после завершения работы с модулем. При любом возвращенном значении после вызова этой функции соответствующий описатель уже нельзя использовать для работы с модулем без открытия нового соединения.
Параметры: hnd — Описатель модуля
Возвращаемое значение: Код ошибки .

4.3.1.4 Проверка, открыто ли соединение с модулем.

Формат: INT LTR216_IsOpened (TLTR216 *hnd)
Описание: Функция проверяет, открыто ли в данный момент соединение с модулем. Если соединение открыто, функция возвращает LTR_OK, если закрыто — код ошибки LTR_ERROR_CHANNEL_CLOSED.
Параметры: hnd — Описатель модуля
Возвращаемое значение: Код ошибки (LTR_OK, если соединение установлено)

4.3.2 Функции для изменения настроек модуля

4.3.2.1 Запись настроек в модуль.

Формат: INT LTR216_SetADC (TLTR216 *hnd)
Описание: Функция передает настройки, соответствующие значениям полей структуры конфигурации описателя модуля, в модуль. Должна вызываться хотя бы раз до первого запуска сбора, а также после любых изменений настроек модуля, чтобы они вступили в силу.
Параметры: hnd — Описатель модуля
Возвращаемое значение: Код ошибки .

4.3.2.2 Заполнение делителя частоты АЦП

Формат: INT LTR216_FillSyncFreqDiv (TLTR216 *hnd, double adcFreq, double *resultAdcFreq)
Описание: <p>Функция по заданному значению частоты АЦП подбирает делитель частоты таким образом, чтобы результирующая частота оказалась наиболее близка к заданной, и записывает полученное значение в поле SyncFreqDiv в структуре конфигурации описателя модуля.</p> <p>Также может возвращать в выходном параметре результирующее значение частоты АЦП.</p> <p>Используется только для многоканального режима сбора.</p>
Параметры: <p>hnd — Описатель модуля.</p> <p>adcFreq — Требуемая частота АЦП в Гц.</p> <p>resultAdcFreq — В данной переменной возвращается полученная в результате подбора частота АЦП в Гц. Может быть передан нулевой указатель, если значение не требуется.</p>
Возвращаемое значение: <p>Код ошибки.</p>

4.3.2.3 Вычисление делителя частоты АЦП

Формат: INT LTR216_FindSyncFreqDiv (double adcFreq, DWORD *div, double *resultAdcFreq)
Описание: <p>Функция аналогична LTR216_FillSyncFreqDiv(), но возвращает рассчитанный делитель явно в переменной. Позволяет производить вычисление без действительного описателя модуля.</p>
Параметры: <p>adcFreq — Требуемая частота АЦП.</p> <p>div — В данной переменной возвращается рассчитанное значение делителя частоты АЦП в Гц.</p> <p>resultAdcFreq — В данной переменной возвращается полученная в результате подбора частота АЦП в Гц. Может быть передан нулевой указатель, если значение не требуется.</p>
Возвращаемое значение: <p>Код ошибки.</p>

4.3.2.4 Заполнение параметров фильтра АЦП

Формат: INT LTR216_FillFilterParams (TLTR216 *hnd, double adcFreq, TLTR216_FILTER_OUT_PARAMS *filterParams, double *swTimeRes)
Описание: <p>Функция по режиму коммутации АЦП, частоте АЦП и требуемому времени на коммутацию (последнее только в многоканальном режиме) подбирает оптимальные параметры фильтра и заполняет полученными значениями поля FilterType и AdcOdrCode в структуре конфигурации описателя модуля.</p> <p>Способ подбора зависит от того, настроен одноканальный или многоканальный режим сбора (перед вызовом данной функции должно быть заполнено поле AdcSwMode в структуре конфигурации описателя модуля.</p> <p>В одноканальном режиме сбора функция подбирает параметры фильтра так, чтобы результирующая частота преобразования АЦП (которая соответствует частоте АЦП в данном режиме) была наиболее близка к переданному значению adcFreq. Подбранную частоту преобразования можно узнать по значению поля Odr полученных параметров фильтра.</p> <p>В многоканальном режиме функция подбирает значения фильтра так, чтобы время преобразования АЦП было максимальным, но при этом для заданной частоты АЦП, определяемой по полю SyncFreqDiv конфигурации, оставшееся время коммутации было не меньше AdcMinSwTimeUs мкс. Соответственно, оба поля в структуре конфигурации должны быть заполнены перед вызовом. Если нужного времени коммутации достичь не удалось, то функция вернет ошибку LTR216_ERR_UNSUUF_SW_TIME.</p>
Параметры: <p>hnd — Описатель модуля.</p> <p>adcFreq — Требуемая частота АЦП для одноканального режима. Для многоканального это значение не используется, а установленная частота определяется по значению поля SyncFreqDiv конфигурации модуля.</p> <p>filterParams — В данной структуре возвращаются параметры, описывающие подобранные настройки фильтра. Может быть передан нулевой указатель, если данная информация не требуется.</p> <p>swTimeRes — В многоканальном режиме в данной переменной возвращается полученное время на коммутацию в мкс. В одноканальном режиме не используется. Может быть передан нулевой указатель, если данная информация не требуется.</p>
Возвращаемое значение: <p>Код ошибки.</p>

4.3.2.5 Подбор параметров фильтра АЦП

Формат: INT LTR216_FindFilterParams (DWORD adcSwMode, double adcFreq, double swTimeUs, TLTR216_FILTER_OUT_PARAMS *filterParams, double *swTimeRes)
Описание: Функция аналогична LTR216_FillFilterParams() , но только рассчитывает параметры без заполнения полей описателя. Позволяет выполнять расчет без создания описателя модуля.
Параметры: adcSwMode — Режим коммутации АЦП. Значение из e_LTR216_ADC_SWMODE . adcFreq — Требуемая частота АЦП в Гц. swTimeUs — Требуемое время на коммутацию в мкс в многоканальном режиме. Не имеет значения для одноканального. filterParams — В данной структуре возвращаются параметры, описывающие подобранные настройки фильтра. swTimeRes — В многоканальном режиме в данной переменной возвращается полученное время на коммутацию в мкс. В одноканальном режиме не используется. Может быть передан нулевой указатель, если данная информация не требуется.
Возвращаемое значение: Код ошибки .

4.3.2.6 Получение параметров фильтра по настройкам

Формат: INT LTR216_GetFilterOutParams (DWORD adcSwMode, DWORD filterType, DWORD adcOdrCode, TLTR216_FILTER_OUT_PARAMS *filterParams)
Описание: Функция возвращает параметры фильтра по его настройкам в зависимости от режима коммутации АЦП.
Параметры: adcSwMode — Режим коммутации АЦП. Значение из e_LTR216_ADC_SWMODE . filterType — Тип фильтра. Значение из e_LTR216_FILTER_TYPE . adcOdrCode — Код, определяющий время выполнения преобразования АЦП. Значение аналогично полю AdcOdrCode структуры конфигурации модуля. filterParams — В данной структуре возвращаются параметры, описывающие фильтр, соответствующий указанным настройкам.
Возвращаемое значение: Код ошибки .

4.3.2.7 Заполнение поля для задания силы тока питания датчиков

Формат: INT LTR216_FillISrcCode (TLTR216 *hnd, double isrc, double *resultValue)
Описание: Функция по заданному значению силы тока питания датчиков подбирает и устанавливает значение поля ISrcCode конфигурации так, чтобы полученный ток питания датчиков был наиболее близок к заданному. Функция использует калибровочные коэффициенты источника тока для более точной установки требуемой величины.
Параметры: hnd — Описатель модуля. isrc — Требуемое значение источника тока в мА resultValue — В данной переменной возвращается полученное значение тока питания датчиков в мА, соответствующее подобранному коду. Может быть передан нулевой указатель, если данная информация не требуется.
Возвращаемое значение: Код ошибки .

4.3.2.8 Подбор кода для задания силы тока питания датчиков

Формат: INT LTR216_FindISrcCode (const TLTR216_ISRC_CBR *cbr, double isrc, DWORD *code, double *resultValue)
Описание: Функция аналогична LTR216_FillISrcCode() , но только подбирает нужное значение кода, не заполняя конфигурацию описателя модуля. Функция позволяет выполнять расчет без создания описателя модуля.
Параметры: cbr — Калибровочные коэффициенты источника тока. Может быть передан нулевой указатель, если не требуется применение калибровочных коэффициентов. isrc — Требуемое значение источника тока в мА code — Полученное значение кода задания величины тока (которое может быть записано в поле ISrcCode конфигурации модуля). Может быть передан нулевой указатель, если данная информация не требуется. resultValue — В данной переменной возвращается полученное значение тока питания датчиков в мА, соответствующее подобранному коду. Может быть передан нулевой указатель, если данная информация не требуется.
Возвращаемое значение: Код ошибки .

4.3.2.9 Расчет величины силы тока по коду

Формат: INT LTR216_CalcISrcValue (const TLTR216_ISRC_CBR *cbr, DWORD code, double *resultValue)
Описание: Функция позволяет по заданному значению поля ISrcCode получить результирующее значение силы тока питания датчиков. Функция может использовать калибровочные коэффициенты для более точного расчета силы тока.
Параметры: cbr — Калибровочные коэффициенты источника тока. Может быть передан нулевой указатель, если не требуется применение калибровочных коэффициентов. code — Значение поля ISrcCode , для которого рассчитывается сила тока. resultValue — В данной переменной возвращается значение силы тока в мА, соответствующее заданному коду.
Возвращаемое значение: Код ошибки.

4.3.2.10 Расчет низкоуровневых параметров цикла опроса АЦП

Формат: INT LTR216_FillRawTables (const TLTR216_MODULE_INFO *minfo, const TLTR216_CONFIG *cfg, TLTR216_RAWCONFIG *rawCfg)
Описание: Функция по заданной пользовательской конфигурации рассчитывает низкоуровневые настройки, соответствующие результирующему циклу опроса АЦП, включая фоновые измерения, холостые такты и т.п. Данная функция как правило не требуется, т.к. действие этой функции выполняется при записи конфигурации в модуль вызовом LTR216_SetADC() . Данная функция позволяет сделать этот расчет без записи настроек в модуль, в частности если связь не установлена или описатель не создан.
Параметры: minfo — Структура с информацией о модуле, включая его калибровочные коэффициенты. Может быть передан нулевой указатель, если не требуется их учет. cfg — Пользовательская конфигурация модуля rawCfg — В данной структуре сохраняются полученные низкоуровневые параметры цикла опроса АЦП.
Возвращаемое значение: Код ошибки.

4.3.3 Функции для управления сбором данных

4.3.3.1 Измерение начальных параметров перед запуском сбора

Формат: INT LTR216_InitMeasParams (TLTR216 *hnd, DWORD meas, DWORD bgMeas, TLTR216_DATA_STATUS *status)
Описание: <p>Функция позволяет выполнить начальные измерения, необходимые для расчета результатов при дальнейшем сборе, перед запуском самого сбора.</p> <p>Какие именно измерения будут выполняться, зависит от переданных флагов в качестве поля meas. Для выполнения всех требуемых измерений можно передать значение LTR216_INIT_MEAS_ALL.</p> <p>Помимо измерений начальных параметров на фоне могут быть использованы дополнительные фоновые измерения для одновременной проверки линий, связанных с выполняемым измерением. Эти измерения задаются с помощью флагов, аналогично полю BgMeas конфигурации модуля. Для измерения смещения нуля не используются, а для измерения U_{ref} могут использоваться набор измерений из группы LTR216_BG_MEASGROUP_UREF_CHECK. Если измерение начального значения не может быть выполнено из-за обнаруженных ошибок, то функция вернет ошибку LTR216_ERR_BAD_INIT_MEAS_STATUS и более подробную информацию можно будет получить по полям структуры TLTR216_DATA_STATUS, возвращаемой в параметре status.</p> <p>При завершении выполнения функция обновляет внутренние поля, явно не доступные пользователю, необходимые для расчетов при запущенном сборе данных. Также результаты измерения напряжения на опорном датчике можно получить через соответствующие поля структуры TLTR216_DATA_STATUS, возвращаемой в параметре status.</p>
Параметры: <p>hnd — Описатель модуля</p> <p>meas — Флаги, определяющие, какие начальные измерения должны быть выполнены. Может быть использован флаг LTR216_INIT_MEAS_ALL для выполнения всех измерений.</p> <p>bgMeas — Дополнительные фоновые измерения, которые могут использоваться для проверки целостности линий, связанных с выбранными начальными измерениями.</p> <p>status — В данной структуре могут быть возвращены результаты выполненных измерений (если они доступны пользователю), а также их состояние. Если данная информация не требуется, то может быть передан нулевой указатель.</p>
Возвращаемое значение: <p>Код ошибки.</p>

4.3.3.2 Запуск сбора данных

Формат: INT LTR216_Start (TLTR216 *hnd)
Описание: При вызове данной функции запускается сбор данных с АЦП и модуль начинает передавать в ПК полученные отсчеты, которые необходимо вычитывать с помощью LTR216_Recv() . При завершении измерений для остановки сбора данных необходимо вызвать LTR216_Stop() .
Параметры: hnd — Описатель модуля
Возвращаемое значение: Код ошибки .

4.3.3.3 Останов сбора данных.

Формат: INT LTR216_Stop (TLTR216 *hnd)
Описание: При вызове данной функции модуль останавливает сбор и выдачу данных. При этом вычитываются и отбрасываются все переданные, но непрочитанные данные от модуля.
Параметры: hnd — Описатель модуля
Возвращаемое значение: Код ошибки .

4.3.3.4 Инициализация структуры состояния модуля при обработке данных.

Формат: INT LTR216_DataStatusInit (TLTR216_DATA_STATUS *status)
Описание: Данная функция заполняет начальными значениями поля структуры TLTR216_DATA_STATUS . Необходимо вызвать данную функцию после создания структуры до передачи ее в другие функции.
Параметры: status — Структура состояния модуля, которая заполняется начальными значениями
Возвращаемое значение: Код ошибки .

4.3.3.5 Инициализация структур состояния каналов модуля при обработке данных.

Формат: INT LTR216_DataChannelsStatusInit (TLTR216_DATA_CH_STATUS *chStatusList, DWORD chCnt)
Описание: Данная функция заполняет начальными значениями поля в массиве структур TLTR216_DATA_STATUS . Может быть проинициализирована и одна структура, если передан указатель на нее и количество элементов указано равным 1. Необходимо вызвать данную функцию после создания данных структур до передачи их в другие функции.
Параметры: chStatusList — Массив из chCnt структур состояния каналов, которые необходимо проинициализировать. chCnt — Количество элементов в массиве chStatusList
Возвращаемое значение: Код ошибки .

4.3.3.6 Прием данных от модуля.

Формат: INT LTR216_Recv (TLTR216 *hnd, DWORD *data, DWORD *tmark, DWORD size, DWORD timeout)
Описание: <p>Функция принимает запрашиваемое число слов от модуля. Возвращаемые слова находятся в специальном формате, который включает в себя служебную информацию. Возвращаемые слова соответствуют как основным пользовательским измерениям, так и фоновым измерениям в соответствии с настройками модуля.</p> <p>Для обработки принятых слов, получения измеренных значений и проверки состояния модуля и каналов используется функция LTR216_ProcessData().</p> <p>Функция возвращает управление либо когда примет запрошенное количество слов, либо по истечению таймаута. При этом реально принятое количество слов можно узнать по возвращенному значению.</p>
Параметры: <p>hnd — Описатель модуля.</p> <p>data — Массив, в который будут сохранены принятые слова. Должен быть размером на size 32-битных слов.</p> <p>tmark — Указатель на массив размером на size 32-битных слов, в который будут сохранены значения счетчиков синхрометок, соответствующие принятым данным. Генерация меток настраивается для крейта или специального модуля отдельно. Синхрометки подробнее описаны в разделе “Синхрометки” руководства для библиотеки ltrapi. Если синхрометки не используются, то можно передать в качестве параметра нулевой указатель.</p> <p>size — Запрашиваемое количество 32-битных слов на прием.</p> <p>timeout — Таймаут на выполнение операции в миллисекундах. Если в течение заданного времени не будет принято запрашиваемое количество слов, то функция все равно вернет управление, возвратив в качестве результата реально принятое количество слов</p>
Возвращаемое значение: <p>Значение меньше нуля соответствует коду ошибки. Значение больше или равное нулю соответствует количеству реально принятых и сохраненных в массив data слов.</p>

4.3.3.7 Обработка принятых от модуля слов.

<p>Формат: INT LTR216_ProcessData (TLTR216 *hnd, const DWORD *src, double *dest, INT *size, DWORD flags, TLTR216_DATA_STATUS *status, TLTR216_DATA_CH_STATUS *chStatusList)</p>
<p>Описание:</p> <p>Функция используется для обработки слов, принятых от модуля с помощью LTR216_Recv(). Функция проверяет служебные поля принятых слов, извлекает полезную информацию с отсчетами и выполняет расчет результирующих измерений с применением при необходимости калибровочных и тарифовочных коэффициентов.</p> <p>Кроме того, данная функция выполняет обработку фоновых измерений и использует их результаты либо для учета в расчете результатов, либо для проверки линий подключения и условий измерений, в зависимости от назначения фонового измерения. Результаты указанных проверок и значения этих фоновых измерений сохраняются в структуры, переданные в качестве параметров status и chStatusList.</p> <p>В функцию необходимо передавать слова, выравненные на начало кадра и в количестве кратном количеству слов в кадре (можно узнать по значению поля FrameWordsCount).</p> <p>Функция изначально считает, что все данные, принятые LTR216_Recv() полностью последовательно обрабатываются LTR216_ProcessData(). Если это не так, то необходимо передать флаг LTR216_PROC_FLAG_NONCONT_DATA.</p>
<p>Параметры:</p> <p>hnd — Описатель модуля.</p> <p>src — Указатель на массив, содержащий слова, принятые от модуля с помощью LTR216_Recv(), которые нужно обработать.</p> <p>dest — Указатель на массив, в который будут сохранены результирующие значения измерений. Порядок следования соответствует порядку во входном массиве: сперва первый отсчет первого логического канала, затем первый отсчет второго логического канала и т.д.</p> <p>size — На входе принимает размер массива src для обработки. На выходе, при успешном завершении, возвращает количество сохраненных отсчетов в массиве dest.</p> <p>flags — Флаги из e_LTR216_PROC_FLAGS, управляющие работой функции. Может быть объединено несколько флагов через логическое ИЛИ.</p> <p>status — В данной структуре сохраняются результаты обработки фоновых измерений и проверки корректности условий измерения, не связанные с конкретным каналом. Может быть передан нулевой указатель, если информация не требуется.</p> <p>chStatusList — Массив структур, количество элементов в котором должно быть равно количеству элементов в логической таблице (LChCnt). В нем сохраняются результаты обработки фоновых измерений и проверки корректности условий измерения, связанные с соответствующим логическим каналом. Может быть передан нулевой указатель, если информация не требуется.</p>
<p>Возвращаемое значение: Код ошибки.</p>

4.3.3.8 Проверка линий подключения и условий измерения

Формат: INT LTR216_CheckMeasStatus (TLTR216 *hnd, DWORD measList, TLTR216_DATA_STATUS *status, TLTR216_DATA_CH_STATUS *chStatusList)
Описание: <p>Функция позволяет выполнить вспомогательные измерения, для проверки целостности линий и корректности условий измерения, не запуская основной цикл сбора данных.</p> <p>Функция принимает в качестве параметра свой набор из флагов, определяющих, какие вспомогательные измерения должны быть проведены. Назначение флагов аналогично фоновым измерениям, но сам набор может отличаться от того, на который настроен модуль.</p> <p>Результаты проверок и значения этих фоновых измерений сохраняются в структуры, переданные в качестве параметров status и chStatusList, аналогично LTR216_ProcessData().</p>
Параметры: <p>hnd — Описатель модуля.</p> <p>measList — Набор флагов из e_LTR216_BG_MEAS, определяющий, какие вспомогательные измерения нужно провести.</p> <p>status — В данной структуре сохраняются результаты обработки фоновых измерений и проверки корректности условий измерения, не связанные с конкретным каналом. Может быть передан нулевой указатель, если информация не требуется.</p> <p>chStatusList — Массив структур, количество элементов в котором должно быть равно количеству элементов в логической таблице (LChCnt). В нем сохраняются результаты обработки фоновых измерений и проверки корректности условий измерения, связанные с соответствующим логическим каналом. Может быть передан нулевой указатель, если информация не требуется.</p>
Возвращаемое значение: <p>Код ошибки.</p>

4.3.4 Функции для пользовательской тарировки каналов

4.3.4.1 Тарировка смещения нуля выбранных каналов.

<p>Формат: INT LTR216_TareOffset (TLTR216 *hnd, DWORD chsMask, DWORD *chsOk)</p>
<p>Описание:</p> <p>Функция выполняет тарировку смещения нуля для выбранных с помощью параметра chsMask каналов.</p> <p>Для каждого выбранного канала должен присутствовать элемент логической таблицы с соответствующим номером физического канала. Соответствующий логический канал и определяет диапазон измерения, на котором будет выполняться тарировка соответствующего физического канала.</p> <p>Также должны быть предварительно заполнены все остальные настройки модуля.</p> <p>Перед вызовом этой функции на каждый тарлируемый канал необходимо подать сигнал, который должен соответствовать нулевому значению после выполнения тарировки.</p> <p>При выполнении тарировки сперва подбирается значение ЦАП для компенсации смещения нуля, величиной до двух диапазонов измерения. После этого выполняется подбор дополнительного калибровочного коэффициента для тонкой компенсации нуля.</p> <p>В случае успешного завершения тарировки данная функция обновляет информацию в полях DacValue и Offset (а также признаки их действительности) структуры TLTR216_TARE_CH_COEFS с информацией о тарировке соответствующего канала в поле ModuleInfo описателя модуля.</p> <p>В случае, если не по всем заданным каналам удалось измерить смещение, то функция вернет ошибку LTR216_ERR_TARE_CHANNELS и по возвращаемой маске в параметре chsOk можно определить, тарировка каких каналов выполнена успешно.</p> <p>Успешное выполнение этой функции очищает результаты тарировки шкалы для соответствующих каналов, т.к. тарировка шкалы зависит от тарировки нуля и должна быть выполнена после тарировки нуля заново.</p>
<p>Параметры:</p> <p>hnd — Описатель модуля.</p> <p>chsMask — Маска каналов, для которых должна быть выполнена тарировка. Каждому физическому каналу соответствует свой бит, начиная с младшего. Если бит установлен, то по соответствующему каналу будет выполнена тарировка.</p> <p>chsOk — В данной переменной возвращается битовая маска, указывающая, по каким каналам успешно выполнена тарировка (бит установлен в 1, если тарировка по соответствующему каналу выполнена успешно).</p>
<p>Возвращаемое значение:</p> <p>Код ошибки.</p>

4.3.4.2 Тарировка шкалы выбранных каналов.

Формат: INT LTR216_TareScale (TLTR216 *hnd, DWORD chsMask, DWORD *chsOk)
Описание: <p>Функция выполняет тарировку шкалы для выбранных с помощью параметра chsMask каналов.</p> <p>Тарировка шкалы выполняется после тарировки нуля, если необходимо выполнить обе тарировки.</p> <p>Для каждого выбранного канала должен присутствовать элемент логической таблицы с соответствующим номером физического канала. Соответствующий логический канал и определяет диапазон измерения, на котором будет выполняться тарировка соответствующего физического канала.</p> <p>Также должны быть предварительно заполнены все остальные настройки модуля.</p> <p>Перед вызовом этой функции на каждый тарируемый канал необходимо подать сигнал, который должен соответствовать максимальному измеряемому значению пользовательской шкалы.</p> <p>В случае успешного завершения тарировки данная функция обновляет информацию в поле Scale (а также признак действительности этого поля) в соответствующем каналу элементе массива Tare в структуре ModuleInfo описателя модуля.</p> <p>В случае, если не по всем заданным каналам удалось измерить смещение, то функция вернет ошибку LTR216_ERR_TARE_CHANNELS и по возвращаемой маске в параметре chsOk можно определить, тарировка каких каналов выполнена успешно.</p>
Параметры: <p>hnd — Описатель модуля.</p> <p>chsMask — Маска каналов, для которых должна быть выполнена тарировка. Каждому физическому каналу соответствует свой бит, начиная с младшего. Если бит установлен, то по соответствующему каналу будет выполнена тарировка.</p> <p>chsOk — В данной переменной возвращается битовая маска, указывающая, по каким каналам успешно выполнена тарировка (бит установлен в 1, если тарировка по соответствующему каналу выполнена успешно).</p>
Возвращаемое значение: <p>Код ошибки.</p>

4.3.4.3 Запись тарифовочных коэффициентов во flash-память

Формат: INT LTR216_WriteTareInfo (TLTR216 *hnd, DWORD chsMask)
Описание: <p>Функция выполняет запись полученных тарифовочных коэффициентов для заданных каналов из соответствующих элементов массива Tare в структуре ModuleInfo описателя модуля в энергонезависимую flash-память модуля.</p> <p>Информация во flash-памяти по неуказанным каналам остается без изменений.</p> <p>Запись коэффициентов во flash-память позволяет их автоматически восстановить при последующих сеансах работы с модулем (при вызове LTR216_Open()) или вручную через LTR216_ReadTareInfo().</p>
Параметры: <p>hnd — Описатель модуля.</p> <p>chsMask — Маска каналов, для которых должна быть выполнена запись информации во flash-память.</p>
Возвращаемое значение: <p>Код ошибки.</p>

4.3.4.4 Чтение тарифовочных коэффициентов из flash-памяти

Формат: INT LTR216_ReadTareInfo (TLTR216 *hnd, DWORD chsMask)
Описание: <p>Функция выполняет чтение тарифовочных коэффициентов для заданных каналов из энергонезависимой flash-памяти модуля и сохранение их в соответствующих элементах массива Tare в структуре ModuleInfo описателя модуля.</p> <p>Информация в элементах массива Tare, соответствующим неуказанным каналам, остается без изменений.</p> <p>Чтение тарифовочных коэффициентов происходит автоматически по всем каналам после открытия соединения с модулем функцией LTR216_Open(). Данная функция позволяет их повторно прочитать для восстановления, в случае если они были изменены вручную или при выполнении тарифовки без записи во flash-память.</p>
Параметры: <p>hnd — Описатель модуля.</p> <p>chsMask — Маска каналов, для которых должно быть выполнено чтение информации из flash-памяти.</p>
Возвращаемое значение: <p>Код ошибки.</p>

4.3.5 Функции для работы с flash-памятью модуля

4.3.5.1 Чтение блока данных из Flash-памяти.

Формат: INT LTR216_FlashRead (TLTR216 *hnd, DWORD addr, BYTE *data, DWORD size)
Описание: Функция считывает массив данных из Flash-памяти модуля в массив, переданный пользователем. Для считывания не нужно специальное разрешение — данная операция всегда доступна.
Параметры: hnd — Описатель модуля. addr — Адрес начала блока во flash-памяти. data — Массив, куда будут сохранены считанные данные (должен быть не менее size байт). size — Количество байт для чтения.
Возвращаемое значение: Код ошибки.

4.3.5.2 Запись блока данных во Flash-память модуля.

Формат: INT LTR216_FlashWrite (TLTR216 *hnd, DWORD addr, const BYTE *data, DWORD size)
Описание: Функция записывает переданный массив данных во Flash-память модуля. Эта область должна быть предварительно стерта с помощью LTR216_FlashErase() и изменения в пользовательской области должны быть ранее разрешены через LTR216_FlashWriteEnable() . Пользователю для записи доступна только область памяти размером LTR216_FLASH_USERDATA_SIZE с адреса LTR216_FLASH_USERDATA_ADDR .
Параметры: hnd — Описатель модуля. addr — Адрес начала блока. data — Массив с записываемыми данными (должен быть не меньше size байт). size — Количество байт для записи.
Возвращаемое значение: Код ошибки.

4.3.5.3 Стирание блока во Flash-памяти.

Формат:	INT LTR216_FlashErase (TLTR216 *hnd, DWORD addr, DWORD size)
Описание:	<p>Функция стирает блок во Flash-памяти модуля (все ячейки будут читаться как 0xFF). Адрес и размер должны быть кратны 256 байт.</p> <p>Стирание необходимо перед записью данных с помощью LTR216_FlashWrite().</p> <p>Для работы данной функции должны быть разрешены изменения в пользовательской области flash-памяти с помощью LTR216_FlashWriteEnable().</p>
Параметры:	<p>hnd — Описатель модуля.</p> <p>addr — Адрес начала блока (должен быть кратен 256).</p> <p>size — Количество байт для стирания (кратно 256).</p>
Возвращаемое значение:	Код ошибки.

4.3.5.4 Разрешение записи в пользовательскую область Flash-памяти.

Формат:	INT LTR216_FlashWriteEnable (TLTR216 *hnd)
Описание:	<p>Функция разрешает запись в пользовательскую область Flash-памяти. Пользовательская область начинается с адреса LTR216_FLASH_USERDATA_ADDR и занимает LTR216_FLASH_USERDATA_SIZE байт.</p> <p>Должна быть вызвана перед использованием LTR216_FlashErase() и LTR216_FlashWrite() для изменения содержимого пользовательской области памяти.</p> <p>После завершения изменений следует вызвать LTR216_FlashWriteDisable().</p>
Параметры:	<p>hnd — Описатель модуля.</p>
Возвращаемое значение:	Код ошибки .

4.3.5.5 Запрет записи в пользовательскую область Flash-памяти.

Формат:	INT LTR216_FlashWriteDisable (TLTR216 *hnd)
Описание:	<p>Функция запрещает запись в пользовательскую область Flash-памяти модуля. Должна быть вызвана после того, как нужные данные в пользовательской области были изменены с помощью LTR216_FlashErase() и LTR216_FlashWrite(), чтобы защитить пользовательскую область от случайного изменения в дальнейшем.</p>
Параметры:	<p>hnd — Описатель модуля.</p>
Возвращаемое значение:	Код ошибки .

4.3.6 Функции вспомогательного характера

4.3.6.1 Разрешение или запрет источников тока питания датчиков.

Формат: INT LTR216_SetISrcEnabled (LTR216 *hnd, BOOLEAN enabled)
Описание: <p>Данная функция позволяет вручную управлять разрешением источников тока питания датчиков.</p> <p>Источники тока включены, если они разрешены и сконфигурированы.</p> <p>По умолчанию источники тока разрешены и будут включены после первого вызова <code>LTR216_SetADC()</code> и до последующего сброса модуля путем закрытия и открытия нового соединения с ним. В большинстве случаев это поведение соответствует ожидаемому и явное управление разрешением не требуется.</p> <p>Но, если по какой-то причине это необходимо, пользователь может явно запретить источники тока, вызвав данную функцию со значением параметра <code>enabled</code> равным ЛОЖЬ, в результате чего они будут выключены до следующего включения этой же функцией с <code>enabled</code> равным ИСТИНА.</p> <p>При этом при разрешении источников тока будет установлен код, соответствующий значению поля <code>ISrcCode</code> конфигурации модуля.</p>
Параметры: <p><code>hnd</code> — Описатель модуля.</p> <p><code>enabled</code> — Параметр указывает, должны быть источники тока разрешены или запрещены.</p>
Возвращаемое значение: <p>Код ошибки.</p>

4.3.6.2 Получение сообщения об ошибке.

Формат: LPCSTR LTR216_GetErrorString (INT err)
Описание: <p>Функция возвращает строку, соответствующую переданному коду ошибки, в кодировке CP1251 для ОС Windows или UTF-8 для ОС Linux. Функция может обработать как ошибки из <code>ltr216api</code>, так и общие коды ошибок из <code>ltrapi</code>.</p>
Параметры: <p><code>err</code> — Код ошибки</p>
Возвращаемое значение: <p>Указатель на строку, содержащую сообщение об ошибке.</p>