

МНОГОКАНАЛЬНЫЕ СИСТЕМЫ СБОРА ДАННЫХ

LTR210

РУКОВОДСТВО ПРОГРАММИСТА

*Ревизия 1.0.5
Сентябрь 2014*

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	23.04.2013	Первая ревизия данного документа (предварительные данные)
1.0.1	14.06.2013	Добавлено описание нового варианта установки порогов синхронизации, настройка режимов дополнительного бита, скорости выдачи в интерфейс. Все названия полей приведены к финальной версии библиотеки
1.0.2	25.06.2013	Дополнено описание процедуры коррекции АЧХ
1.0.3	19.08.2013	Добавлено описание возможности измерения и коррекции смещения нуля
1.0.4	02.06.2014	Добавлено описание подключения библиотеки для проектов с использованием 64-битных компиляторов MSVC, Embarcadero C++ Builder и MinGW
1.0.5	10.09.2014	Вопрос подключения библиотеки к проектам вынесен в отдельный документ

Оглавление

1	О чем этот документ	5
2	Установка и подключение библиотеки к проекту	6
3	Общий подход к работе с библиотекой	7
3.1	Общий алгоритм работы с модулем	7
3.1.1	Работа с модулем в режиме покадрового сбора	7
3.1.2	Работа с модулем при непрерывном потоковом вводе	8
3.2	Загрузка прошивки ПЛИС модуля	8
3.3	Настройка модуля	9
3.3.1	Настройка частоты сбора АЦП	10
3.3.2	Настройка каналов АЦП	10
3.3.3	Настройка параметров принимаемого кадра	11
3.3.4	Настройка события синхронизации	11
3.3.5	Работа нескольких модулей LTR210 в группе	12
3.3.6	Настройка скорости выдачи данных в интерфейс	13
3.3.7	Настройка режима дополнительного бита в потоке данных от АЦП	13
3.4	Принцип записи и выдачи кадра	14
3.5	Прием данных в режиме покадрового сбора	16
3.6	Анализ статуса принятого кадра	16
3.7	Контроль работы модуля с помощью периодических статусов (сигнала жизни)	17
3.8	Режим непрерывного сбора данных	18
3.9	Особенности калибровки данных модулем	19
3.10	Коррекция АЧХ	19
3.11	Измерение и коррекция смещения нуля	20
3.12	Вызов функций библиотеки из разных потоков	20
4	Константы, типы данных и функции библиотеки	21
4.1	Константы и перечисления	21
4.1.1	Константы и макроопределения	21
4.1.2	Коды ошибок, специфичные для LTR210	22
4.1.3	Диапазоны канала АЦП	23
4.1.4	Режим измерения канала АЦП	24
4.1.5	Режим работы и события синхронизации	24
4.1.6	Режим работы модуля в группе	25
4.1.7	Коды асинхронных событий	25
4.1.8	Коды, определяющие правильность принятого кадра	26
4.1.9	Флаги статуса	26
4.1.10	Дополнительные флаги настроек	27

4.1.11	Флаги обработки данных.	28
4.1.12	Скорость выдачи данных в интерфейс.	28
4.1.13	Режим работы дополнительного бита во входном потоке.	29
4.2	Типы данных.	29
4.2.1	Калибровочные коэффициенты.	29
4.2.2	Параметры БИХ-фильтра.	29
4.2.3	Информация о модуле.	30
4.2.4	Настройки канала АЦП.	31
4.2.5	Настройки модуля.	31
4.2.6	Параметры состояния модуля.	32
4.2.7	Описатель модуля.	33
4.2.8	Дополнительная информация о принятом отсчете.	34
4.2.9	Информация о статусе обработанного кадра	34
4.2.10	Тип функции для индикации процесса загрузки ПЛИС.	35
4.3	Функции	35
4.3.1	Функции инициализации и работы с соединением с модулем.	35
4.3.1.1	Инициализация описателя модуля.	35
4.3.1.2	Открытие соединения с модулем.	36
4.3.1.3	Закрытие соединения с модулем.	36
4.3.1.4	Проверка, открыто ли соединение с модулем.	37
4.3.1.5	Проверка, загружена ли прошивка ПЛИС модуля.	37
4.3.1.6	Загрузка прошивки ПЛИС модуля.	37
4.3.2	Функции для изменения настроек модуля	38
4.3.2.1	Запись настроек в модуль.	38
4.3.2.2	Установка заданной частоты сбора АЦП.	38
4.3.2.3	Установка заданной частоты следования кадров.	39
4.3.3	Функции для управления сбором данных	39
4.3.3.1	Запуск сбора данных.	39
4.3.3.2	Останов сбора данных.	40
4.3.3.3	Программный запуск сбора кадра.	40
4.3.3.4	Ожидание асинхронного события от модуля.	41
4.3.3.5	Прием данных от модуля.	42
4.3.3.6	Обработка принятых от модуля слов.	43
4.3.4	Функции вспомогательного характера	44
4.3.4.1	Измерение смещения нуля	44
4.3.4.2	Получение прошедшего интервала с момента приема послед- него слова.	44
4.3.4.3	Получение сообщения об ошибке.	45
4.3.4.4	Загрузка коэффициентов в ПЛИС.	45

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 LTR210 с использованием предоставляемой фирмой “Л Кард” библиотеки ltr210api.

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 свое 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 LTR210. Более подробно каждый шаг будет описан в последующих разделах.

Типичная последовательность действий имеет следующий вид:

1. Создать экземпляр структуры [TLTR210](#), представляющей собой описатель модуля. Описатель модуля содержит всю информацию о модуле и используется при вызове всех остальных функций.
2. Проинициализировать поля описателя с помощью [LTR210_Init\(\)](#)
3. Установить соединение с интересующим модулем с помощью функции [LTR210_Open\(\)](#).
4. Проверить, загружена ли прошивка ПЛИС модуля, с помощью функции [LTR210_FPGAIsLoaded\(\)](#). Если прошивка не загружена, то загрузить ее, вызвав функцию [LTR210_LoadFPGA\(\)](#).
5. Заполнить необходимые поля с настройками модуля подструктуры [Cfg описателя модуля](#) и вызвать [LTR210_SetADC\(\)](#), для записи настроек в модуль.
6. При желании можно выполнить измерение собственного нуля для дополнительной коррекции его ухода с помощью [LTR210_MeasAdcZeroOffset\(\)](#).
1. Запуск сбора данных с помощью [LTR210_Start\(\)](#).
2. Прием и обработка данных в зависимости от режима, как описано ниже
3. По завершению работы выполнить останов сбора данных с помощью [LTR210_Stop\(\)](#).
4. Закрыть соединение с модулем, вызвав функцию [LTR210_Close\(\)](#).

3.1.1 Работа с модулем в режиме покадрового сбора

Типичный цикл приема и обработки данных в режиме покадрового сбора, выглядит следующим образом:

1. Ожидание прихода данных от модуля с помощью [LTR210_WaitEvent\(\)](#).

2. По типу события (параметр event) определить, что за данные пришли:
 - Если не пришло никаких данных за указанное время, то возвращаемся к пункту 1 для ожидания дальнейших данных. При включенной [периодической посылке статуса](#), можно проверить время прихода последнего статуса для выявления фактов нарушения работы модуля.
 - Если принят [периодический статус модуля \(сигнал жизни\)](#), проанализировать статус (при необходимости) и перейти к пункту 1.
 - Если принято начало записанного кадра, то переход к пункту 3
3. Прием отсчетов кадра в количестве [State.RecvFrameSize](#) с помощью [LTR210_Recv\(\)](#).
4. Обработка принятого кадра с помощью [LTR210_ProcessData\(\)](#).
5. Анализ [статуса принятого кадра](#), для определения, действителен ли принятый кадр. Пользовательская обработка и переход к пункту 1.

3.1.2 Работа с модулем при непрерывном потоковом вводе

Типичный цикл приема и обработки данных аналогичен большинству модулей АЦП и выглядит следующим образом:

1. Прием заданного количества отсчетов с помощью [LTR210_Recv\(\)](#).
2. Обработка принятых отсчетов с помощью [LTR210_ProcessData\(\)](#).

3.2 Загрузка прошивки ПЛИС модуля

В модуле LTR210 не реализована возможность хранения прошивки ПЛИС во внутренней энергонезависимой памяти, поэтому для работы с модулем необходимо предварительно загрузить прошивку в ПЛИС.

Загрузка прошивки в модуль является, как правило, самым первым действием, которое необходимо выполнить, после установления соединения с модулем через [LTR210_Open\(\)](#). Однако, так как после загрузки прошивки ПЛИС она сохраняется в модуле до сброса питания, прошивку не обязательно повторно загружать, если она уже была загружена при предыдущем сеансе работы с модулем. Для проверки, загружена ли прошивка, служит функция [LTR210_FPGAIsLoaded\(\)](#). Если эта функция вернула LTR_OK, то прошивка уже загружена и можно переходить к настройке модуля. В противном случае, нужно загрузить прошивку с помощью функции [LTR210_LoadFPGA\(\)](#).

Функция [LTR210_LoadFPGA\(\)](#) принимает одним из параметров имя файла прошивки. Файл `ltr210_fpga.rbf`, содержащий прошивку, поставляется вместе с библиотекой `ltr210api`. Кроме того, в ОС Windows последняя версия прошивки на момент создания библиотеки встраивается в библиотеку как ресурс, что позволяет не хранить отдельно файл прошивки. Загрузка встроенной в библиотеку прошивки выполняется при передаче пустой строки или нулевого указателя в качестве имени файла. Для ОС Linux при установке из предоставляемых пакетов, файл прошивки устанавливается в директорию `/usr/share/ltrapi/ltr210` и при указании пустого имени берется файл, расположенный по этому стандартному пути.

Следует отметить, что в `LTR210_LoadFPGA()` можно передать указатель на функцию обратного вызова (callback), которая будет вызываться каждый раз, когда был успешно записан блок данных из файла прошивки в ПЛИС. Это позволяет при необходимости реализовать в программе индикатор прогресса загрузки. В функцию обратного вызова каждый раз передаются полный размер файла прошивки в байтах, количество успешно записанных байт и указатель на пользовательские данные, который можно передать в `LTR210_LoadFPGA()`. В первый раз функция обратного вызова будет вызвана сразу после успешного открытия файла (или загрузки ресурса) с нулевым размером записанных байт, а последний — после успешного завершения загрузки — с размером записанных байт равным размеру прошивки. В случае возникновения ошибки `LTR210_LoadFPGA()` немедленно возвращает код произошедшей ошибки в качестве результата, не вызывая при этом дополнительно функцию обратного вызова. Если индикатор загрузки не требуется, то в качестве указателя на функцию можно передать нулевой указатель.

3.3 Настройка модуля

Если прошивка ПЛИС была загружена, то следующим этапом необходимо установить настройки модуля.

Настройка модуля выполняется аналогично большинству других модулей LTR: вначале значения всех параметров модуля записываются в соответствующие поля [структуры описателя модуля](#), затем вызывается функция `LTR210_SetADC()`, которая значения этих полей передает модулю.

Следует отметить, что все поля, относящиеся к настройке модуля, объединены в структуру типа `TLTR210_CONFIG` (поле [Cfg описателя модуля](#)). Только эти поля пользователь должен изменять вручную в [описателе модуля](#) при штатной работе и только эти поля влияют на параметры, записываемые `LTR210_SetADC()`.

При настройке модуля задаются следующие параметры:

- Частота сбора АЦП (раздел [Настройка частоты сбора АЦП](#))
- Настройки каналов АЦП (раздел [Настройка каналов АЦП](#))
- Размер кадра и предыстории для режима покадрового сбора (раздел [Настройка параметров принимаемого кадра](#))
- Событие синхронизации, по которому происходит сбор кадра (раздел [Настройка события синхронизации](#))
- Настройки для работы цепочки модулей LTR210 в группе (раздел [Работа нескольких модулей LTR210 в группе](#))

После выполнения `LTR210_SetADC()` рассчитываются некоторые параметры, которые являются производными от настроек из `TLTR210_CONFIG`. Эти параметры записываются в соответствующие поля структуры состояния модуля типа `TLTR210_STATE` (поле [State описателя модуля](#)). Например, на основе делителей частоты АЦП и коэффициента прореживания рассчитывается результирующая частота сбора АЦП в Гц и записывается в поле `AdcFreq`.

Настройка модуля должна быть выполнена хотя бы один раз перед началом записи данных.

Кроме того, часть параметров модуля может быть изменена при запущенном сборе:

- Диапазоны измерения каналов АЦП
- Режим измерения для каналов АЦП
- Уровень аналоговой синхронизации
- Частота кадров при периодическом сборе

Для изменения этих параметров на лету, нужно аналогичным образом изменить нужные поля [структуры конфигурации](#) и вызвать `LTR210_SetADC()`.

3.3.1 Настройка частоты сбора АЦП

Оба канала АЦП модуля всегда работают параллельно с одной и той же частотой преобразования, которая может достигать до 10 МГц (на каждый канал).

Частота сбора АЦП задается двумя параметрами:

- Делителем частоты преобразования АЦП. Частота 10 МГц делится на делитель, равный значению поля `AdcFreqDiv`+1. Значение `AdcFreqDiv` может быть от 0 до `LTR210_ADC_FREQ_DIV_MAX`-1.
- Коэффициентом прореживания данных. Для получения более низких частот сбора в ПЛИС сделано прореживание данных от АЦП. Этот коэффициент задается полем `AdcDcmCnt`. Из `AdcDcmCnt`+1 отсчетов будет записан в буфер модуля только один отсчет. Значение `AdcDcmCnt` может изменяться в диапазоне от 0 до `LTR210_ADC_DCM_CNT_MAX`-1.

Таким образом, результирующая частота сбора для каждого канала АЦП будет равна

$$f_{acq} = \frac{10000000}{(AdcFreqDiv + 1) * (AdcDcmCnt + 1)} \text{ Гц}$$

Для установки нужных значений полей `AdcFreqDiv` и `AdcDcmCnt` можно также воспользоваться функцией `LTR210_FillAdcFreq()`, которая подбирает эти значения так, чтобы частота сбора АЦП была наиболее близка к указанной в соответствующем параметре функции.

После вызова `LTR210_SetADC()` значение, соответствующее реально установленной частоте сбора АЦП, записывается в поле `AdcFreq` структуры [состояния модуля](#).

Установленная частота сбора не зависит от количества разрешенных каналов АЦП.

3.3.2 Настройка каналов АЦП

Каждый модуль LTR210 имеет два канала, которые выполняют преобразования параллельно. Можно разрешить запись как по любому из двух, так и по обоим каналам одновременно. Настройки каждого канала объединены в структуре `TLTR210_CHANNEL_CONFIG`. Массив структур из 2-х элементов, каждый элемент которого соответствует нужному каналу, является полем `Ch` структуры с [настройками модуля](#).

Для каждого канала можно независимо задать:

- Разрешена ли запись по этому каналу. Задается полем `Enabled`.
- Диапазон для данного канала (поле `Range`).

- Режим работы канала (поле [Mode](#)): открытый вход (без отсечки постоянной составляющей), закрытый вход (с отсечкой постоянной составляющей) или измерение собственного нуля.
- Верхний и нижний уровень синхронизации при использовании соответствующего режима синхронизации (раздел [Настройка события синхронизации](#))
- Режим работы специального бита, вставляемого в поток отсчетов, соответствующих данному каналу (раздел [Настройка режима дополнительного бита в потоке данных от АЦП](#)).

3.3.3 Настройка параметров принимаемого кадра

В режиме покадрового сбора, который является основным для модуля LTR210, выдача отсчетов осуществляется блоками, называемыми в данном документе кадрами. Выдача кадра осуществляется только при возникновении события синхронизации.

Количество точек в кадре на канал задается с помощью поля [FrameSize](#). При этом часть отсчетов кадра соответствует измерениям непосредственно перед возникновением события синхронизации, а часть — отсчетам АЦП непосредственно после его возникновения. Отсчеты кадра, соответствующие времени до события, приведшего к выдаче этого кадра, называются предысторией события. Размер предыстории в количестве точек на канал задается с помощью параметра [HistSize](#) (может быть равен 0).

Ограничения на размер кадра в зависимости от режима записи описаны в разделе [Принцип записи и выдачи кадра](#).

После последнего слова кадра всегда высылается [Статус кадра](#), который содержит информацию о действительности кадра, а также о различных других событиях, произошедших во время записи и выдачи кадра. Статус кадра всегда должен быть проверен, чтобы можно было сделать вывод о правильности принятых данных.

Таким образом, сам кадр состоит из $N_{ch} * FrameSize + 1$ слов, где N_{ch} — количество разрешенных каналов (1 или 2). После записи настроек модуля с помощью [LTR210_SetADC\(\)](#) вычисленный размер кадра сохраняется в поле [RecvFrameSize](#) структуры [состояния модуля](#). Значение этого поля можно использовать в дальнейшем при приеме данных от модуля.

3.3.4 Настройка события синхронизации

В режиме покадрового сбора при вызове функции [LTR210_Start\(\)](#) модуль переходит в режим ожидания заданного условия, и только при возникновении этого условия будет выдан кадр данных в интерфейс модуля. Это условие, называемое событием синхронизации, настраивается с помощью поля [SyncMode](#). Само условие должно быть настроено до вызова [LTR210_Start\(\)](#) на этапе конфигурации модуля.

Возможны следующие варианты события синхронизации:

- По команде, пришедшей от ПК. Эта команда посылается с помощью функции [LTR210_FrameStart\(\)](#). По приходу команды будет осуществлена выдача одного кадра данных.
- По фронту или спаду аналогового сигнала на одном из двух каналов относительно заданного уровня синхронизации. Уровень синхронизации задается отдельно для каждого канала с помощью двух уровней — верхнего уровня гистерезиса (поле

`SyncLevelH`) и нижнего (поле `SyncLevelL`). Два уровня позволяют избежать ложного срабатывания даже при зашумленном сигнале. Фронтом считается переход сигнала через верхний уровень, при условии, что до этого он был ниже нижнего уровня. Спадом же — переход ниже нижнего уровня, если до этого сигнал превышал верхний. Установленные уровни должны быть в пределах, соответствующих установленному диапазону для данного канала. Безусловно, верхний уровень должен быть не ниже нижнего. При этом запись по каналу, используемому для аналоговой синхронизации, может быть запрещена, если интересуют лишь данные другого канала, но диапазон и режим сбора должен быть настроен верно.

- По фронту или спаду цифрового сигнала на входе SYNC. Важно отметить, что этот режим предназначен для синхронизации от внешнего источника цифрового сигнала, отличного от другого модуля LTR210. Для работы нескольких модулей LTR210 в группе по принципу “Ведущий - ведомые” существует специальная [настройка для работы модуля в группе](#).
- Периодический сбор кадров. В этом режиме событие синхронизации генерируется аппаратурой самого модуля с заданной частотой. Эта частота определяется полем `FrameFreqDiv` и равняется $f_{frame} = \frac{10^6}{FrameFreqDiv+1}$ Гц. После вызова `LTR210_SetADC()` рассчитанное значение этой частоты в Герцах записывается в поле `FrameFreq` структуры [состояния модуля](#). Для заполнения поля `FrameFreqDiv` можно также использовать функцию `LTR210_FillFrameFreq()`.

Следует отметить, что если после возникновения события синхронизации и до завершения записи и последующей выдачи в интерфейс кадра произойдет еще одно событие синхронизации, то это событие будет отброшено. Предыдущий кадр же будет успешно передан, но для оповещения о пропущенном событии в флагах его статуса будет установлен флаг `LTR210_STATUS_FLAG_SYNC_SKIP`.

Отдельно следует отметить, что `SyncMode=LTR210_SYNC_MODE_CONTINUOUS` является специальным случаем. При данном значении модуль настраивается на режим непрерывного сбора, который более подробно описан в разделе [Режим непрерывного сбора данных](#).

3.3.5 Работа нескольких модулей LTR210 в группе

Для модуля LTR210 имеется возможность объединять несколько модулей в цепочку по принципу “Ведущий-ведомые”. Для настройки данного режима предназначено поле `GroupMode`. По-умолчанию это поле равно `LTR210_GROUP_MODE_INDIVIDUAL`, что соответствует работе модуля независимо от остальных.

При работе модулей в группе необходимо один модуль настроить ведущим (мастером), установив `GroupMode = LTR210_GROUP_MODE_MASTER`. Ведущий модуль при запуске записи отслеживает события синхронизации в соответствии с установленным значением поля `SyncMode`. При обнаружении события синхронизации ведущий модуль не только выдает кадр в интерфейс с крейтом, но и генерирует импульс на линии SYNC. Для ведущего модуля может быть установлено любое значение `SyncMode`, кроме `LTR210_SYNC_MODE_CONTINUOUS`.

Ведомые модули должны по цепочке быть соединены с ведущим с помощью входов SYNC (подробнее описано в документе [“Крейтовая система LTR. Руководство пользователя”](#)). Т.е. если имеется ведущий модуль *M* и два ведомых *S1* и *S2*, то *M* может быть соединен с *S1*, а *S1* с *S2*. Для всех ведомых модулей должно быть

установлено поле `GroupMode = LTR210_GROUP_MODE_SLAVE`. При этом поле `SyncMode` не имеет значения — ведомый модуль всегда запускает сбор кадра по сигналу от ведущего.

Модули могут быть настроены в произвольном порядке. Однако запуск записи данных через `LTR210_Start()` следует сперва выполнить для всех ведомых модулей, а уже только после этого для ведущего, чтобы гарантировать, что первое событие синхронизации не будет потеряно ни одним ведомым.

3.3.6 Настройка скорости выдачи данных в интерфейс

При возникновении события синхронизации записанные данные вычитываются из кольцевого буфера и передаются в интерфейс модуль → крейт с заданной скоростью. Максимальная скорость определяется типом крейта, в который вставлен модуль. Для LTR-U-1 она равна 200 КСлов/с, для остальных крейтов — 500 КСлов/с. Эта скорость определяет время выдачи записываемого кадра и влияет на минимальный интервал следования событий синхронизации, которые не будут пропущены, а также на максимальный размер записываемого кадра в случае, если автоматическая приостановка записи не включена (подробнее в разделе [Принцип записи и выдачи кадра](#)).

Скорость может быть задана явно с помощью поля `IntfTransfRate`. По-умолчанию установлена скорость 500 КСлов/с. При этом, если установленное значение превышает максимально допустимое значение скорости для крейта (в случае скорости 500 КСлов/с и LTR-U-1), то библиотека автоматически запишет корректное значение. Т.е. по-умолчанию используется максимально возможная для используемого крейта скорость.

Явное указание скорости может использоваться в случае большого количества модулей LTR210, для которых время выдачи кадра может пересекаться во времени, чтобы не превысить максимальную пропускную способность интерфейса крейт → компьютер.

Пример: Пусть используется LTR-EU-16 с 16-ю модулями LTR210, подключенный по интерфейсу Ethernet. Если все модули будут одновременно передавать записанный кадр со скоростью 500 КСлов/с, то суммарная скорость составит 8 МСлов/с (24 МБайт/с), что превышает максимальную скорость передачи крейта по Ethernet (порядка 10 МБайт/с или 2.5 МСлов/с). То есть, крейт не сможет передавать данные в ПК с той же скоростью, что принимает, и при определенных размерах кадров может произойти переполнение внутреннего буфера крейта. В этом случае можно для модулей установить скорость выдачи в интерфейс, равную 100 КСлов/с, что снизит скорость общего потока до 1.6 МСлов/с.

Таким образом, при большом количестве модулей следует всегда оценивать общую скорость передачи данных всех модулей крейта (а также нескольких крейтов, если они подключены к одной сети), учитывая и модули других типов (не LTR210), и, при необходимости, явно уменьшать скорость выдачи данных в интерфейс для избежания переполнения внутреннего буфера крейта.

3.3.7 Настройка режима дополнительного бита в потоке данных от АЦП

В модуле LTR210 в месте с отсчетами АЦП передается дополнительный бит данных. Что конкретно будет отображать этот бит, можно настроить независимо для каждого канала АЦП с помощью поля `DigBitMode`. Например, этот бит может отображать уровень на входе SYNC, что позволяет, при необходимости, использовать этот вход для

синхронного ввода одного цифрового бита. Другие возможности приведены в описании типа `e_LTR210_DIG_BIT_MODE`, задающего режим данного бита.

При обработке данных с помощью `LTR210_ProcessData()` значение данного бита сохраняется в массив структур с дополнительной информацией (если в качестве параметра `data_info` был передан ненулевой указатель). Каждая структура соответствует своему отчету из обработанных данных, что позволяет выполнить однозначное соответствие.

3.4 Принцип записи и выдачи кадра

В отличие от большинства других АЦП от “Л Кард”, основной режим использования модуля LTR210 не предполагает непрерывного сбора данных. Вместо этого, при вызове `LTR210_Start()` модуль включает запись данных во внутренний кольцевой буфер в SRAM самого модуля размером на 16 МСлов (определено константой `LTR210_INTERNAL_BUFFER_SIZE`) и ожидает наступление указанного события синхронизации.

Только при возникновении события синхронизации модуль выдает кадр данных в интерфейс модуль → крейт (далее просто интерфейс) в соответствии с параметрами кадра, описанными в разделе [Настройка параметров принимаемого кадра](#).

Сразу после завершения передачи кадра по интерфейсу модуль готов к выдаче следующего кадра по событию синхронизации. Все события синхронизации, которые произошли от момента начала выдачи кадра в интерфейс (момента возникновения соответствующего синхрособытия) до момента завершения выдачи будут потеряны, при этом в [статусе кадра](#) будет указан флаг `LTR210_STATUS_FLAG_SYNC_SKIP`.

Следует учесть, что скорость интерфейса между модулем и крейтом ограничена (200 КСлов/с для LTR-U-1 и 500 КСлов/с для остальных крейтов). Кроме того, ее можно, при необходимости, дополнительно ограничить с помощью специальной настройки (см. раздел [Настройка скорости выдачи данных в интерфейс](#)). Таким образом, время выдачи кадра в интерфейс может существенно превышать время записи кадра во внутренний буфер модуля.

По-умолчанию модуль выполняет непрерывную запись во внутренний буфер. В таком режиме при определенных размерах кадра и частотах сбора АЦП может возникнуть ситуация, что процесс записи отсчетов в циклический буфер обгонит процесс чтения из буфера на круг и часть передаваемого кадра будет перетёрта новыми данными. Соответственно, в этом случае часть данных переданного кадра будет недействительна. Этот факт отражается в [статусе кадра](#) установкой флага `LTR210_STATUS_FLAG_OVERLAP`.

Скорость записи данных в буфер равна $N_{ch} * f_{acq}$, где f_{acq} — частота сбора (см. раздел [Настройка частоты сбора АЦП](#)), а N_{ch} — количество каналов, по которым разрешена запись. Если скорость записи ниже скорости чтения, то размер `FrameSize` может быть любым, вплоть до $\frac{LTR210_FRAME_SIZE_MAX}{N_{ch}}$. Константа `LTR210_FRAME_SIZE_MAX` задает максимальный размер кадра, который можно установить при одном разрешенном канале, что соответствует размеру внутреннего буфера за вычетом 512 слов — размера блочной записи в SDRAM модуля.

В случае, если скорость записи больше скорости чтения, для гарантии, что не произойдет перетирание, время передачи кадра по интерфейсу должно быть меньше, чем время записи всего циклического буфера вместе с размером кадра за вычетом предыдущего кадра.

тории. Т.е. должно выполняться неравенство

$$\frac{S + N_{ch} * (FrameSize - HistSize)}{f_{acq} * N_{ch}} \geq \frac{N_{ch} * FrameSize}{f_{intf}}$$

где $S = \text{LTR210_FRAME_SIZE_MAX}$, f_{intf} – скорость выдачи данных в интерфейс.

Если задать в формуле размер предыстории как долю от размера всего кадра n : $HistSize = n * FrameSize$, где n от 0 до 1, то можно можно переписать формулу в более удобный вид. Если необходимо узнать максимальный размер кадра при заданных частоте сбора АЦП, скорости выдачи данных в интерфейс и процентной доли предыстории, то формула выглядит следующим образом:

$$FrameSize \leq \frac{S * f_{intf}}{N_{ch} * (f_{acq} * N_{ch} - f_{intf} * (1 - n))}$$

Таблица с результатами данной формулы для некоторых типовых случаев приведена в документе ‘Крейтовая система LTR. Руководство пользователя’.

Для случая, если для заданных размеров кадра нужно определить максимальную частоту сбора, формула выглядит следующим образом:

$$f_{acq} \leq \frac{f_{intf} * (S/N_{ch} + FrameSize * (1 - n))}{N_{ch} * FrameSize}$$

Для того, чтобы иметь возможность установки полного размера буфера и при частотах сбора вплоть до максимальной, введен режим автоматической приостановки записи в кольцевой буфер. Этот режим устанавливается при конфигурации модуля с помощью установки флага `LTR210_CFG_FLAGS_WRITE_AUTO_SUSP` в поле `Flags` структуры конфигурации. В этом режиме при возникновении события синхронизации выполняется запись во внутренний буфер до конца кадра, после чего процесс записи будет автоматически остановлен. При этом, запись будут возобновлена только после завершения выдачи всего кадра в интерфейс и перетирание данных невозможно.

Плюсом режима автоматической приостановки записи является возможность записать кадр размером до $\frac{\text{LTR210_FRAME_SIZE_MAX}}{N_{ch}}$, независимо от частоты отсчетов АЦП. Минусом является то, что при ненулевом размере предыстории необходимо, чтобы после завершения выдачи кадра в интерфейс до возникновения следующего события синхронизации прошло время, необходимое для записи предыстории. В режиме же непрерывной записи при событии, произошедшем сразу после завершения передачи предыдущего кадра в интерфейс, кадр будет всегда записан успешно, независимо от размера предыстории.

Таким образом, при непрерывной записи интервал между соседними событиями синхронизации, чтобы ни одно событие не было пропущено, должен быть:

$$T_{sync_cont} > \max\left(\frac{N_{ch} * FrameSize}{f_{intf}}, \frac{FrameSize - HistSize}{f_{acq}}\right)$$

а в режиме с автоматической приостановкой записи:

$$T_{sync_susp} > T_{sync_cont} + \frac{HistSize}{f_{acq}}$$

3.5 Прием данных в режиме покадрового сбора

Как было уже описано, в штатном режиме работы от модуля нет непрерывного потока данных, а данные передаются асинхронно по кадрам. Кроме того, помимо кадров данных модуль может периодически передавать свое слово состояния (подробнее описано в разделе [Контроль работы модуля с помощью периодических статусов \(сигнала жизни\)](#)). В связи с этим операция приема данных несколько отличается от других модулей LTR. Вначале необходимо дождаться события прихода каких-либо данных от модуля. Для этого служит функция `LTR210_WaitEvent()`, которая ожидает в течение таймаута прихода слова данных от модуля, анализирует это слово и возвращает в параметре *event*, что за данные пришли.

Если это начало кадра (`LTR210_RECV_EVENT_SOF`), то далее следует прочитать кадр данных аналогично тому, как это делается и для остальных модулей LTR: сперва почитать `RecvFrameSize` слов, принятых от модуля, с помощью `LTR210_Recv()`, а затем обработать их с помощью `LTR210_ProcessData()`. При желании кадр можно принимать и/или обрабатывать не целиком, а блоками меньшего размера, однако следует учитывать, что только после обработки последнего блока будет известен статус кадра и можно будет сделать вывод о действительности принятых данных.

Важно отметить, что первое слово кадра при передаче от модуля в крейт/ПК отмечается специальным образом и `LTR210_WaitEvent()` возвращает `LTR210_RECV_EVENT_SOF` именно при нахождении начала кадра. Если `LTR210_WaitEvent()` будет вызвано в середине не до конца принятого кадра, то слова неполного кадра будут отброшены.

Также `LTR210_Recv()`, в отличие от аналогичных функций остальных модулей, при обнаружении конца кадра немедленно вернет данные до конца кадра, даже если запрашивалось больше данных и таймаут еще не истек.

3.6 Анализ статуса принятого кадра

Как уже отмечалось, в кадре помимо слов с отсчетами АЦП последним словом передается статус кадра, по которому и можно сделать вывод, верны ли данные в кадре. Информация о статусе кадра возвращается функцией `LTR210_ProcessData()` в виде структуры типа `TLTR210_FRAME_STATUS`.

По значению поля `Result` можно сделать общий вывод, был ли найден конец кадра и, если найден, действительны ли данные в принятом кадре. Если слово со статусом кадра не было найдено, то поле `Result` равно `LTR210_FRAME_RESULT_PENDING` (например, в случае, если пользователь обрабатывает кадр по частям). Если же слово статуса кадра найдено, то `Result = LTR210_FRAME_RESULT_OK`, если все данные действительны, либо `LTR210_FRAME_RESULT_ERROR`, если часть из данных повреждена.

Поле `Flags` содержит полную информацию о статусе кадра в виде побитовой комбинации набора флагов и может служить для более детального анализа состояния кадра. В частности, по этим флагам можно узнать причину повреждения данных, если было возвращено значение `Result = LTR210_FRAME_RESULT_ERROR`.

Следующие флаги относятся непосредственно к принятому кадру:

- `LTR210_STATUS_FLAG_OVERLAP` — признак, что процесс записи обогнал процесс чтения и часть данных в кадре была перетерта. Нельзя утверждать, какие

данные в кадре являются действительными, а какие нет. Поле `Result` устанавливается в `LTR210_FRAME_RESULT_ERROR`. Подробнее о причине возникновения данного события описано в разделе [Настройка параметров принимаемого кадра](#).

- `LTR210_STATUS_FLAG_INVALID_HIST` — признак, что событие синхронизации произошло раньше, чем было записано `HistSize` точек для каждого канала и предыстория в принятом кадре не действительна. Поле `Result` устанавливается в `LTR210_FRAME_RESULT_ERROR`, однако при отсутствии других флагов ошибки можно предположить, что часть кадра, относящаяся к времени после возникновения синхрособытия, верна. При непрерывной записи в буфер возникновение такой ситуации возможно только в случае, если событие синхронизации произошло непосредственно после начала записи с помощью `LTR210_Start()` и `HistSize` $\neq 0$ и только в первом принятом кадре после разрешения записи. При включенной автоматической приостановке записи возникновение этого флага возможно в случае прихода следующего события синхронизации раньше, чем было записано `HistSize` точек для каждого канала с момента завершения выдачи в интерфейс предыдущего кадра.
- `LTR210_STATUS_FLAG_SYNC_SKIP` — признак, что во время записи и выдачи кадра в интерфейс произошло одно или несколько событий синхронизации, которые были отброшены, так как выдача предыдущего кадра не была закончена. Этот флаг не влияет на действительность данных и, соответственно, на значение поля `Result`.

Кроме этого, в флагах передается информация:

- По каким каналам разрешена запись.
- Состояние захвата PLL модуля. При этом при отсутствии признака захвата PLL `LTR210_ProcessData()` вернет ошибку `LTR210_ERR_PLL_NOT_LOCKED`, так как данный факт свидетельствует о неисправности модуля.

Все значения флагов приведены в описании типа `e_LTR210_STATUS_FLAGS`.

3.7 Контроль работы модуля с помощью периодических статусов (сигнала жизни)

Так как в режиме покадрового сбора при отсутствие событий синхронизации модуль ведет запись лишь во внутренний кольцевой буфер и не передает никаких данных в крейт/ПК, то в такой ситуации нет информации, по которой можно судить о исправности модуля. Если необходима дополнительная проверка того, что модуль исправно выполняет запись во внутренний буфер и проверку условий синхронизации, то можно разрешить периодическую посылку модулем своего слова состояния. Для этого необходимо при настройке конфигурации модуля установить флаг `LTR210_CFG_FLAGS_KEEPLIVE_EN` в поле `Flags` структуры конфигурации до вызова `LTR210_SetADC()`. В этом случае, после разрешения записи вызовом `LTR210_Start()` модуль будет раз в 500 мс передавать свое слово состояния вплоть до запрета записи вызовом `LTR210_Stop()`. При этом слова состояния передаются только в момент, когда не идет передача кадра (т.е. они могут быть лишь между кадрами, но не внутри самого кадра), благодаря чему размер слов в кадре остается неизменным

и процедура его приема не отличается для случаев включенной или выключенной периодической передаче статуса модуля.

Периодическая передача статуса может быть использована в качестве сигнала жизни модуля — отсутствие слов статуса от модуля в течение заданного интервала служит признаком неисправности модуля. Следует иметь ввиду, что этот интервал должен быть больше 500мс (например, несколько секунд), так как должен учитывать все возможные задержки как передачи данных по интерфейсу между крейтом и ПК, так и возможные программные задержки получения данных в сервере и программе верхнего уровня.

При обнаружении прихода статуса от модуля, `LTR210_WaitEvent()` вернет событие `LTR210_RECV_EVENT_KEEPLIVE`. Кроме того, в параметре *status* (если не был передан нулевой указатель) будет сохранена дополнительная информация из пришедшего статуса. Она аналогична информации в [статусе кадра](#), за исключением отсутствия флагов, которые относятся непосредственно к принятому кадру. Таким образом, в периодическом статусе доступна следующая информация:

- По каким каналам разрешена запись
- Состояние захвата PLL модуля. При этом при отсутствии признака захвата PLL `LTR210_WaitEvent()` вернет ошибку `LTR210_ERR_PLL_NOT_LOCKED`, так как данный факт свидетельствует о неисправности модуля

Проверку наличия сигнала жизни (статуса) в течение заданного интервала можно выполнять одним из следующих способов:

1. Можно в цикле приема вызывать `LTR210_WaitEvent()` с таймаутом, равным интервалу в течение которого считается, что статус должен обязательно прийти. Если по истечению таймаута не произошло никакого события (`LTR210_WaitEvent()` вернула событие `LTR210_RECV_EVENT_TIMEOUT`), то это свидетельствует о неисправности модуля.
2. Можно вызывать `LTR210_WaitEvent()` с меньшим таймаутом, и в случае, если никакого события не произошло, проверить, сколько времени прошло с момента прихода последних данных от модуля, вызвав `LTR210_GetLastWordInterval()`. Если этот интервал превышает порог, то можно считать, что модуль неисправен. Этот способ позволяет, в частности, интерактивно остановить сбор данных, так как не требует ожидания в `LTR210_WaitEvent()` больших таймаутов.

3.8 Режим непрерывного сбора данных

Для настройки непрерывного сбора данных нужно установить `SyncMode = LTR210_SYNC_MODE_CONTINUOUS`. В этом режиме параметры кадра (`FrameSize` и `HistSize`) не имеют значение, а `GroupMode` должен быть равен `LTR210_GROUP_MODE_INDIVIDUAL` (значение по-умолчанию).

Кроме того, в режиме непрерывного сбора есть ограничение на частоту сбора АЦП. Это ограничение связано с ограничением скорости передачи данных по интерфейсу между модулем и крейтом. Для LTR-U-1 должно выполняться неравенство $N_{ch} * f_{acq} \leq 200$ КГц, а для остальных крейтов $N_{ch} * f_{acq} \leq 500$ КГц. Данные условия проверяются при вызове `LTR210_SetADC()`.

Также для непрерывного сбора данных не используется посылка статусов жизни, т.е. флаг `LTR210_CFG_FLAGS_KEEPLIVE_EN` не имеет значение.

В режиме непрерывного сбора по вызову `LTR210_Start()` начинается сбор данных с АЦП и выдача их в интерфейс. Прием данных осуществляется аналогично остальным модулям LTR с АЦП с помощью `LTR210_Recv()` и `LTR210_ProcessData()`. Модуль не передает никаких статусных слов, так как поток отсчетов не разбит на кадры. Таким образом, в качестве результата обработки кадра `LTR210_ProcessData()` всегда возвращает `LTR210_FRAME_RESULT_PENDING` и эти данные не несут какой-либо дополнительной информации, поэтому как правило в качестве `frame_status` можно передавать нулевой указатель.

3.9 Особенности калибровки данных модулем

Следует отметить, что в отличие от большинства остальных модулей LTR, калибровка данных происходит аппаратно внутри модуля, а не программно. В связи с этим в `LTR210_ProcessData()` нет никаких указаний о выполнении калибровки. Самой библиотекой выполняется чтение калибровочных коэффициентов из Flash-памяти модуля, сохранение их в полях массива `CbrCoef` в структуре с информацией о модуле и запись этих коэффициентов в ПЛИС.

ПЛИС налету выполняет калибровку по формуле $Y = (X + Offset) * Gain$, где X — отсчет с АЦП, Y — калиброванные данные, $Offset$ — смещение шкалы, а $Gain$ — коэффициент шкалы. При этом перед выполнением калибровки, входное значение АЦП расширяется на 1 бит. Уже откалиброванные 15-битные отсчеты передаются в крейт. При этом код `LTR210_ADC_SCALE_CODE_MAX` соответствует напряжению, равному максимальному напряжению для заданного диапазона.

3.10 Коррекция АЧХ

В библиотеке реализована возможность выполнения коррекции амплитудно-частотной характеристики (АЧХ) модуля.

Для всех диапазонов может выполняться корректировка наклона АЧХ с помощью КИХ-фильтра второго порядка в соответствии с методом, описанным в статье [Метод тонкой коррекции наклона АЧХ с помощью простого цифрового фильтра](#). Во Flash-памяти модуля сохранено значение отношения измеренной амплитуды сигнала заданной частоты, выполненное при максимальной частоте сбора АЦП (10 МГц), к амплитуде сигнала с наименьшей частотой для каждого канала и диапазона, а также значение частоты сигнала в Гц.

Кроме того, для диапазонов ± 10 В и ± 5 В может выполняться дополнительная коррекция излома АЧХ в низкочастотной области с помощью БИХ-фильтра в соответствии с методом, описанным в статье [Простой БИХ-фильтр коррекции излома АЧХ в низкочастотной области полосы частот пропускания](#). Во Flash-памяти хранятся параметры **R2** и **C** (**R1** всегда равен 1) эквивалентной схемы фильтра для корректировки АЧХ на максимальной частоте дискретизации (10 МГц). БИХ-фильтр применяется до КИХ-фильтра (при этом в записанном во Flash-память коэффициенте КИХ-фильтра учитывается вносимые БИХ-фильтром изменения АЧХ).

Все параметры для коррекции АЧХ считываются из Flash-памяти модуля при его открытии вместе с калибровочными коэффициентами и сохраняется в соответствующие поля структуры с информацией о модуле.

При настройке модуля с помощью `LTR210_SetADC()` библиотека, используя упомянутые выше параметры, выполняет расчет коэффициентов фильтра для заданной ча-

стоты дискретизации, если они уже не были рассчитаны при предыдущем запуске. При обработке принятых данных в функцию `LTR210_ProcessData()` можно передать флаг `LTR210_PROC_FLAG_AFC_COR`, который указывает, что для полученных данных необходимо применить рассчитанный фильтр/фильтры для корректировки АЧХ модуля.

3.11 Измерение и коррекция смещения нуля

В LTR210 есть возможность выполнять измерение собственного нуля. Это позволяет при необходимости учесть смещение нуля, которое может быть связано либо с его временным уходом, либо с изменением окружающих условий. Для этого необходимо выполнить измерение собственного нуля непосредственно перед началом записи и при дальнейшем сборе, вычтя из полученных отсчетов измеренное значение нуля, получить скорректированные измерения. Смещение нуля зависит от использованных диапазонов и канала, поэтому измерение нуля нужно выполнять для тех же настроек, что и дальнейшие измерения.

Хотя пользователь может запрограммировать этот алгоритм при необходимости и сам, библиотека предоставляет функции для облегчения данной операции. Функция `LTR210_MeasAdcZeroOffset()` выполняет измерение собственного нуля для заданных в конфигурации модуля диапазонов. Ее необходимо вызывать после настройки АЦП перед разрешением записи данных. Рассчитанные значения сохраняются в поле `AdcZeroOffset` структуры *состояния модуля*.

В дальнейшем, при обработке данных, в `LTR210_ProcessData()` можно передать флаг `LTR210_PROC_FLAG_ZERO_OFFS_COR`, в результате чего `LTR210_ProcessData()` вычтет сохраненные измеренные значения смещения нуля из обрабатываемых отсчетов.

Следует при этом отметить, что с момента измерения смещения нуля с помощью `LTR210_MeasAdcZeroOffset()` до обработки данных диапазоны каналов не должны меняться. При смене диапазонов требуется повторное измерение смещения нуля.

3.12 Вызов функций библиотеки из разных потоков

Функции библиотеки `ltr210api`, как и библиотек для работы с другими модулями LTR, не являются потокобезопасными. То есть пользователь сам должен гарантировать, что вызовы функций для работы с одним и тем же модулем выполняются последовательно (естественно, возможна параллельная работа с разными модулями из разных потоков).

Однако, для удобства использования из этого правила есть исключения. На фоне приема данных в отдельном потоке с помощью `LTR210_WaitEvent()`, `LTR210_Recv()` и `LTR210_ProcessData()` можно вызывать из другого потока следующие функции:

- `LTR210_FrameStart()` для вызова программного события синхронизации
- `LTR210_SetADC()` для обновления на лету части настроек

Это позволяет вызывать `LTR210_FrameStart()` и `LTR210_SetADC()` из потока пользовательского интерфейса даже в случае, если сам сбор выполняется в отдельном потоке.

Глава 4

Константы, типы данных и функции библиотеки

4.1 Константы и перечисления.

4.1.1 Константы и макроопределения.

Константа	Значение	Описание
LTR210_NAME_SIZE	8	Размер строки с именем модуля в структуре TINFO_LTR210
LTR210_SERIAL_SIZE	16	Размер строки с серийным номером модуля в структуре TINFO_LTR210
LTR210_CHANNEL_CNT	2	Количество каналов АЦП в одном модуле
LTR210_RANGE_CNT	5	Количество диапазонов измерения АЦП
LTR210_AFC_IIR_COR_RANGE_CNT	2	Количество диапазонов, для которых нужно выполнять дополнительную коррекцию АЧХ с помощью IIR фильтра
LTR210_ADC_SCALE_CODE_MAX	13000	Код принятого отсчета АЦП, соответствующий максимальному напряжению заданного диапазона
LTR210_ADC_FREQ_DIV_MAX	10	Максимальное значение делителя частоты АЦП
LTR210_ADC_DCM_CNT_MAX	256	Максимальное значение коэффициента прореживания данных от АЦП
LTR210_ADC_FREQ_HZ	10000000	Частота в Герцах, относительно которой задается частота отсчетов АЦП

LTR210_FRAME_FREQ_HZ	1000000	Частота в Герцах, относительно которой задается частота следования кадров в режиме LTR210_SYNC_MODE_PERIODIC
LTR210_INTERNAL_BUFFER_SIZE	(16777216)	Размер внутреннего циклического буфера модуля в отсчетах АЦП
LTR210_FRAME_SIZE_MAX	(16777216 - 512)	Максимальный размер кадра, который можно установить в одноканальном режиме

4.1.2 Коды ошибок, специфичные для LTR210.

Тип: e_LTR210_ERROR_CODES		
Описание: Коды ошибок, которые определены и используются только в ltr210api. Остальные коды ошибок, которые используются разными модулями, определены в ltrapi.h		
Константа	Значение	Описание
LTR210_ERR_INVALID_SYNC_MODE	-10500	Задан неверный код условия сбора кадра
LTR210_ERR_INVALID_GROUP_MODE	-10501	Задан неверный код режима работы модуля в составе группы
LTR210_ERR_INVALID_ADC_FREQ_DIV	-10502	Задано неверное значение делителя частоты АЦП
LTR210_ERR_INVALID_CH_RANGE	-10503	Задан неверный код диапазона канала АЦП
LTR210_ERR_INVALID_CH_MODE	-10504	Задан неверный режим измерения канала
LTR210_ERR_SYNC_LEVEL_EXCEED_RANGE	-10505	Установленный уровень аналоговой синхронизации выходит за границы установленного диапазона
LTR210_ERR_NO_ENABLED_CHANNEL	-10506	Ни один канал АЦП не был разрешен
LTR210_ERR_PLL_NOT_LOCKED	-10507	Ошибка захвата PLL
LTR210_ERR_INVALID_RECV_DATA_CNTR	-10508	Неверное значение счетчика в принятых данных
LTR210_ERR_RECV_UNEXPECTED_CMD	-10509	Прием неподдерживаемой команды в потоке данных
LTR210_ERR_FLASH_INFO_SIGN	-10510	Неверный признак информации о модуле во Flash-памяти
LTR210_ERR_FLASH_INFO_SIZE	-10511	Неверный размер прочитанной из Flash-памяти информации о модуле
LTR210_ERR_FLASH_INFO_UNSUP_FORMAT	-10512	Неподдерживаемый формат информации о модуле из Flash-памяти

LTR210_ERR_FLASH_INFO_CRC	-10513	Ошибка проверки CRC информации о модуле из Flash-памяти
LTR210_ERR_FLASH_INFO_VERIFY	-10514	Ошибка проверки записи информации о модуле во Flash-память
LTR210_ERR_CHANGE_PAR_ON_THE_FLY	-10515	Часть измененных параметров нельзя изменять на лету
LTR210_ERR_INVALID_ADC_DCM_CNT	-10516	Задан неверный коэффициент прореживания данных АЦП
LTR210_ERR_MODE_UNSUP_ADC_FREQ	-10517	Установленный режим не поддерживает заданную частоту АЦП
LTR210_ERR_INVALID_FRAME_SIZE	-10518	Неверно задан размер кадра
LTR210_ERR_INVALID_HIST_SIZE	-10519	Неверно задан размер предыстории
LTR210_ERR_INVALID_INTF_TRANSF_RATE	-10520	Неверно задано значение скорости выдачи данных в интерфейс
LTR210_ERR_INVALID_DIG_BIT_MODE	-10521	Неверно задан режим работы дополнительного бита
LTR210_ERR_SYNC_LEVEL_LOW_EXCEED_HIGH	-10522	Нижний порог аналоговой синхронизации превышает верхний
LTR210_ERR_KEEPLIVE_TOUT_EXCEEDED	-10523	Не пришло ни одного статуса от модуля за заданный интервал
LTR210_ERR_WAIT_FRAME_TIMEOUT	-10524	Не удалось дождаться прихода кадра за заданное время
LTR210_ERR_FRAME_STATUS	-10525	Слово статуса в принятом кадре указывает на ошибку данных

4.1.3 Диапазоны канала АЦП

Тип: e_LTR210_ADC_RANGE		
Описание: Диапазоны канала АЦП		
Константа	Значение	Описание
LTR210_ADC_RANGE_10	0	Диапазон ± 10 В
LTR210_ADC_RANGE_5	1	Диапазон ± 5 В
LTR210_ADC_RANGE_2	2	Диапазон ± 2 В
LTR210_ADC_RANGE_1	3	Диапазон ± 1 В
LTR210_ADC_RANGE_0_5	4	Диапазон ± 0.5 В

4.1.4 Режим измерения канала АЦП

Тип: e_LTR210_CH_MODE		
Описание: Режим измерения канала АЦП		
Константа	Значение	Описание
LTR210_CH_MODE_ACDC	0	Измерение переменной и постоянной составляющей (открытый вход)
LTR210_CH_MODE_AC	1	Отсечка постоянной составляющей (закрытый вход)
LTR210_CH_MODE_ZERO	2	Режим измерения собственного нуля

4.1.5 Режим работы и события синхронизации.

Тип: e_LTR210_SYNC_MODE		
Описание: Определяет режим работы модуля и условие запуска сбора кадра (события синхронизации)		
Константа	Значение	Описание
LTR210_SYNC_MODE_INTERNAL	0	Режим сбора кадра по программной команде, передаваемой вызовом LTR210_FrameStart()
LTR210_SYNC_MODE_CH1_RISE	1	Режим сбора кадра по фронту сигнала относительно уровня синхронизации на первом аналоговом канале
LTR210_SYNC_MODE_CH1_FALL	2	Режим сбора кадра по спаду сигнала относительно уровня синхронизации на первом аналоговом канале
LTR210_SYNC_MODE_CH2_RISE	3	Режим сбора кадра по фронту сигнала относительно уровня синхронизации на втором аналоговом канале
LTR210_SYNC_MODE_CH2_FALL	4	Режим сбора кадра по спаду сигнала относительно уровня синхронизации на втором аналоговом канале
LTR210_SYNC_MODE_SYNC_IN_RISE	5	Режим сбора кадра по фронту цифрового сигнала на входе SYNC (не от другого модуля!)
LTR210_SYNC_MODE_SYNC_IN_FALL	6	Режим сбора кадра по спаду цифрового сигнала на входе SYNC (не от другого модуля!)
LTR210_SYNC_MODE_PERIODIC	7	Режим периодического сбора кадров с установленной частотой следования кадров
LTR210_SYNC_MODE_CONTINUOUS	8	Режим непрерывного сбора данных

4.1.6 Режим работы модуля в группе.

Тип: e_LTR210_GROUP_MODE		
Описание: Определяет режим работы модуля в составе группы. Используется при организации цепочки из модулей LTR210, записывающих кадры по одному событию		
Константа	Значение	Описание
LTR210_GROUP_MODE_INDIVIDUAL	0	Модуль работает независимо от остальных
LTR210_GROUP_MODE_MASTER	1	Режим мастера — при возникновении заданного события синхронизации модуль выдает сигнал на выход SYNC. Этот сигнал могут использовать подчиненные модули для запуска преобразования одновременно с мастером. Может использоваться совместно с любым значением SyncMode, кроме LTR210_SYNC_MODE_CONTINUOUS
LTR210_GROUP_MODE_SLAVE	2	Режим подчиненного модуля — модуль запускает сбор кадра от сигнала на входе SYNC, который должен сгенерировать другой LTR210, настроенный на режим мастера. Значение SyncMode при этом не учитывается

4.1.7 Коды асинхронных событий.

Тип: e_LTR210_RECV_EVENT		
Описание: Коды, определяющие, что за асинхронные данные были приняты от модуля и какому событию они соответствуют. Возвращаются функцией LTR210_WaitEvent() в параметре event		
Константа	Значение	Описание
LTR210_RECV_EVENT_TIMEOUT	0	Не пришло никакого события от модуля за указанное время
LTR210_RECV_EVENT_KEEPLIVE	1	Пришел корректный сигнал статуса от модуля (сигнал жизни)
LTR210_RECV_EVENT_SOF	2	Пришло начало записанного кадра

4.1.8 Коды, определяющие правильность принятого кадра

Тип: e_LTR210_FRAME_RESULT		
Описание:		
Константа	Значение	Описание
LTR210_FRAME_RESULT_OK	0	Кадр принят без ошибок. Данные кадра действительны
LTR210_FRAME_RESULT_PENDING	1	В обрабатываемых данных не было признака конца кадра.
LTR210_FRAME_RESULT_ERROR	2	Кадр принят с ошибкой. Данные кадра недействительны. Причину ошибки можно узнать по флагам статуса

4.1.9 Флаги статуса

Тип: e_LTR210_STATUS_FLAGS		
Описание: Флаги, указывающие как текущее состояние самого модуля LTR210, так и информацию о последнем принятом кадре. В виде комбинации данных флагов представляется информация как из периодических слов статуса модуля, так и из слов статуса кадра		
Константа	Значение	Описание
LTR210_STATUS_FLAG_PLL_LOCK	0x0001	Признак захвата PLL в момент передачи статуса. Если равен нулю, то модуль неработоспособен.
LTR210_STATUS_FLAG_PLL_LOCK_HOLD	0x0002	Признак, что захват PLL не пропал с момента предыдущей передачи статуса. Должен быть установлен во всех статусах, кроме первого
LTR210_STATUS_FLAG_OVERLAP	0x0004	Признак, что процесс записи обогнал процесс чтения. Часть данных в кадре может быть недействительна
LTR210_STATUS_FLAG_SYNC_SKIP	0x0008	Признак, что во время записи кадра возникло хотя бы одно событие синхронизации, которое было пропущено. Не влияет на правильность самого кадра.
LTR210_STATUS_FLAG_INVALID_HIST	0x0010	Признак того, что предыстория принятого кадра недействительна (событие наступило меньше чем через HistSize отсчетов после разрешения записи)
LTR210_STATUS_FLAG_CH1_EN	0x0040	Признак, что разрешена запись по первому каналу
LTR210_STATUS_FLAG_CH2_EN	0x0080	Признак, что разрешена запись по второму каналу

4.1.10 Дополнительные флаги настроек.

Тип: e_LTR210_CFG_FLAGS		
Описание: Набор флагов, управляющих конфигурацией модуля при его настройке. Комбинация этих флагов может быть записана в поле Flags структуры TLTR210_CONFIG		
Константа	Значение	Описание
LTR210_CFG_FLAGS_KEEPLIVE_EN	0x001	Разрешение периодической передачи статуса модуля при запущенном сборе
LTR210_CFG_FLAGS_WRITE_AUTO_SUSP	0x002	Разрешение режима автоматической приостановки записи на время, пока кадр выдается по интерфейсу в крейт. Данный режим позволяет установить максимальный размер кадра независимо от частоты сбора АЦП
LTR210_CFG_FLAGS_TEST_CNTR_MODE	0x100	Включение тестового режима, в котором вместо данных передается счетчик

4.1.11 Флаги обработки данных.

Тип: e_LTR210_PROC_FLAGS		
Описание: Флаги, управляющие работой функции LTR210_ProcessData()		
Константа	Значение	Описание
LTR210_PROC_FLAG_VOLT	0x0001	Признак, что нужно перевести коды АЦП в Вольты. Если данный флаг не указан, то будут возвращены коды АЦП. При этом код LTR210_ADC_SCALE_CODE_MAX соответствует максимальному напряжению для установленного диапазона.
LTR210_PROC_FLAG_AFC_COR	0x0002	Признак, что необходимо выполнить коррекцию АЧХ на основании записанных во Flash-памяти модуля коэффициентов
LTR210_PROC_FLAG_ZERO_OFFS_COR	0x0004	Признак, что необходимо выполнить дополнительную коррекцию нуля с помощью значений из State.AdcZeroOffset, которые могут быть измерены с помощью функции LTR210_MeasAdcZeroOffset()
LTR210_PROC_FLAG_NONCONT_DATA	0x0100	По умолчанию LTR210_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

4.1.12 Скорость выдачи данных в интерфейс.

Тип: e_LTR210_INTF_TRANSF_RATE		
Описание: Набор констант, которые задают скорость чтения данных из буфера модуля LTR210 и выдачи их в интерфейс крейта.		
Константа	Значение	Описание
LTR210_INTF_TRANSF_RATE_500K	0	500 КСлов/с
LTR210_INTF_TRANSF_RATE_200K	1	200 КСлов/с
LTR210_INTF_TRANSF_RATE_100K	2	100 КСлов/с
LTR210_INTF_TRANSF_RATE_50K	3	50 КСлов/с
LTR210_INTF_TRANSF_RATE_25K	4	25 КСлов/с

LTR210_INTF_TRANSF_RATE_10K	5	10 КСлов/с
-----------------------------	---	------------

4.1.13 Режим работы дополнительного бита во входном потоке.

Тип: e_LTR210_DIG_BIT_MODE		
Описание: Определяет, какое значение будут передаваться в дополнительном бите в отсчетах, принятых от модуля.		
Константа	Значение	Описание
LTR210_DIG_BIT_MODE_ZERO	0	Всегда нулевое значение бита
LTR210_DIG_BIT_MODE_SYNC_IN	1	Бит отражает состояние цифрового входа SYNC модуля
LTR210_DIG_BIT_MODE_CH1_LVL	2	Бит равен "1", если уровень сигнала для 1-го канала АЦП превысил верхний уровень синхронизации и не опустился ниже нижнего
LTR210_DIG_BIT_MODE_CH2_LVL	3	Бит равен "1", если уровень сигнала для 2-го канала АЦП превысил верхний уровень синхронизации и не опустился ниже нижнего
LTR210_DIG_BIT_MODE_INTERNAL_SYNC	4	Бит равен "1" для одного отсчета в момент срабатывания программной или периодической синхронизации

4.2 Типы данных.

4.2.1 Калибровочные коэффициенты.

Тип: TLTR210_CBR_COEF		
Описание: Структура, хранящая калибровочные коэффициенты для одного канала и диапазона.		
Поле	Тип	Описание поля
Offset	float	15-битный код смещения
Scale	float	Коэффициент шкалы

4.2.2 Параметры БИХ-фильтра.

Тип: TLTR210_AFC_IIR_COEF		
Описание: Параметры для расчета коэффициентов БИХ-фильтра, который используется для коррекции АЧХ диапазонов 10 В и 5 В.		
Поле	Тип	Описание поля
R	double	Сопротивление эквивалентной цепи фильтра
C	double	Емкость эквивалентной цепи фильтра

4.2.3 Информация о модуле.

Тип: TINFO_LTR210		
<p>Описание: Структура, содержащая информацию о версиях прошивок микросхем модуля и информацию, считанную из Flash-памяти модуля (серийный номер, калибровочные коэффициенты).</p> <p>За исключением VerFPGA все поля действительны после вызова LTR210_Open(). Поле VerFPGA, если ПЛИС не загружен на момент установления соединения с модулем, будет действительно только после успешного вызова LTR210_LoadFPGA().</p>		
Поле	Тип	Описание поля
Name	CHAR [LTR210_NAME_SIZE]	Название модуля (оканчивающаяся нулем ASCII-строка)
Serial	CHAR [LTR210_SERIAL_SIZE]	Серийный номер модуля (оканчивающаяся нулем ASCII-строка)
VerFPGA	WORD	Версия прошивки ПЛИС модуля (действительна только после ее загрузки)
VerPLD	BYTE	Версия прошивки PLD
CbrCoef	TLTR210_CBR_COEF [LTR210_CHANNEL_CNT] [8]	Заводские калибровочные коэффициенты (на канал действительны первые LTR210_RANGE_CNT , остальные - резерв)
AfcCoefFreq	double	Частота в Гц, которой соответствуют корректировочные коэффициенты АЧХ
AfcCoef	double [LTR210_CHANNEL_CNT] [8]	Коэффициенты, задающие спад АЧХ модуля на частоте AfcCoefFreq. Представляют собой отношение амплитуды измеренного синусоидального сигнала на указанной частоте к амплитуде реально выставленного сигнала. Коэффициенты загружаются из Flash-памяти модуля при открытии связи с ним. Могут быть использованы для корректировки АЧХ при необходимости. На канал действительны первые LTR210_RANGE_CNT коэффициентов, остальные - резерв.
AfcIirParam	TLTR210_AFC_IIR_COEF [LTR210_CHANNEL_CNT] [8]	Параметры для расчета коэффициентов ИР-фильтра, который используется для корректировки АЧХ на диапазонах 10 В и 5 В. Данные параметры хранятся во Flash-памяти модуля. На канал действительны первые LTR210_AFC_IIR_COR_RANGE_CNT коэффициентов, остальные - резерв.
Reserved	DWORD [32]	Резервные поля

4.2.4 Настройки канала АЦП.

Тип: TLTR210_CHANNEL_CONFIG		
Описание: Структура, содержащая настройки одного канала АЦП.		
Поле	Тип	Описание поля
Enabled	BOOLEAN	Признак, разрешен ли сбор по данному каналу
Range	BYTE	Установленный диапазон — константа из e_LTR210_ADC_RANGE
Mode	BYTE	Режим измерения — константа из e_LTR210_CH_MODE
DigBitMode	BYTE	Режим работы дополнительного бита во входном потоке данных данного канала. Константа из e_LTR210_DIG_BIT_MODE
Reserved	BYTE [4]	Резервные поля (не должны изменяться пользователем)
SyncLevelL	double	Нижний порог гистерезиса при аналоговой синхронизации в Вольтах. Спадом считается уменьшение сигнала ниже уровня SyncLevelL, если до этого сигнал превышал SyncLevelH. Фронтом — превышение SyncLevelH, если до этого сигнал был ниже SyncLevelL. Должен быть в пределах установленного диапазона.
SyncLevelH	double	Верхний порог гистерезиса при аналоговой синхронизации в Вольтах. Должен быть в пределах установленного диапазона и не меньше SyncLevelL.
Reserved2	DWORD [10]	Резервные поля (не должны изменяться пользователем)

4.2.5 Настройки модуля.

Тип: TLTR210_CONFIG		
Описание: Структура содержит все настройки модуля, которые должен заполнить пользователь перед вызовом LTR210_SetADC() .		
Поле	Тип	Описание поля
Ch	TLTR210_CHANNEL_CONFIG [LTR210_CHANNEL_CNT]	Настройки каналов АЦП
FrameSize	DWORD	Размер точек на канал в кадре при покадровом сборе
HistSize	DWORD	Размер предыстории (количество точек в кадре на канал, измеренных до возникновения события синхронизации)
SyncMode	BYTE	Условие сбора кадра (событие синхронизации). Одно из значений e_LTR210_SYNC_MODE
GroupMode	BYTE	Режим работы в составе группы модулей. Одно из значений e_LTR210_GROUP_MODE
AdcFreqDiv	WORD	Значение делителя частоты АЦП минус 1. Может быть в диапазоне от 0 до LTR210_ADC_FREQ_DIV_MAX-1

AdcDcmCnt	DWORD	Значение коэффициент прореживания данных АЦП минус 1. Может быть в диапазоне от 0 до LTR210_ADC_DCM_CNT_MAX -1.
FrameFreqDiv	DWORD	Делитель частоты запуска сбора кадров для SyncMode = LTR210_SYNC_MODE_PERIODIC . Частота кадров равна $\frac{10^6}{FrameFreqDiv+1}$ Гц
Flags	DWORD	Флаги конфигурации (комбинация из e_LTR210_CFG_FLAGS)
IntfTransfRate	BYTE	Скорость выдачи данных в интерфейс (одно из значений из e_LTR210_INTF_TRANSF_RATE). По умолчанию устанавливается максимальная скорость (500 КСлов/с). Если установленная скорость превышает максимальную скорость интерфейса для крейта, в который установлен модуль, то будет установлена максимальная скорость, поддерживаемая данным крейтом
Reserved	DWORD [39]	Резервные поля (не должны изменяться пользователем)

4.2.6 Параметры состояния модуля.

Тип: TLTR210_STATE		
Описание: Структура, содержащая параметры модуля, которые пользователь должен использовать только для чтения, так как они изменяются только внутри функций ltr210api.		
Поле	Тип	Описание поля
Run	BOOLEAN	Признак, запущен ли сбор данных
RecvFrameSize	DWORD	Количество слов в принимаемом кадре, включая статус. (устанавливается после вызова LTR210_SetADC())
AdcFreq	double	Рассчитанная частота сбора АЦП в Гц (устанавливается после вызова LTR210_SetADC())
FrameFreq	double	Рассчитанная частота следования кадров для режима синхронизации LTR210_SYNC_MODE_PERIODIC (устанавливается после вызова LTR210_SetADC())

AdcZeroOffset	double [LTR210_CHANNEL_CNT]	Измеренные значения смещения нуля АЦП в кодах (приведенных к значению LTR210_ADC_SCALE_CODE_MAX) для текущих настроек. Устанавливаются при вызове функции LTR210_MeasAdcZeroOffset() . Могут быть использованы для дополнительной коррекции смещения нуля АЦП.
Reserved	DWORD [4]	Резервные поля

4.2.7 Описатель модуля.

Тип: TLTR210		
Описание: Структура содержит всю информацию о настройках модуля и текущем состоянии соединения с модулем. Используется всеми функциями для работы с модулем.		
Поле	Тип	Описание поля
Size	INT	Размер структуры. Заполняется в LTR210_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	TLTR210_CONFIG	Настройки модуля. Заполняются пользователем перед вызовом LTR210_SetADC() .
State	TLTR210_STATE	Состояние модуля и рассчитанные параметры. Поля изменяются функциями библиотеки. Пользовательской программой могут использоваться только для чтения.
ModuleInfo	TINFO_LTR210	Информация о модуле

4.2.8 Дополнительная информация о принятом отсчете.

Тип: TLTR210_DATA_INFO		
Описание: Структура содержит дополнительную информацию о принятом и обработанном слове данных, которая извлечена из информационных полей принятого слова.		
Поле	Тип	Описание поля
DigBitState	BYTE	Младший бит соответствует значению дополнительного бита, передаваемого вместе с потоком данных. Что означает данный бит задается одной из констант из <code>e_LTR210_DIG_BIT_MODE</code> в поле DigBitMode настроек каждого канала на этапе конфигурации. Остальные биты могут быть использованы в будущем, поэтому при анализе нужно проверять значение DigBitState & 1
Ch	BYTE	Номер канала, которому соответствует принятое слово (0 — первый, 1 — второй)
Range	BYTE	Диапазон канала, установленный во время измерения данного отсчета
Reserved	BYTE	Резервное поле (всегда равно 0)

4.2.9 Информация о статусе обработанного кадра

Тип: TLTR210_FRAME_STATUS		
Описание: Данная структура содержит поля, сообщающие о статусе обработанного кадра. Ее необходимо проверять, чтобы убедиться в том, что данные принятого кадра верны и при его записи не произошло ошибок. Для этого можно использовать поле Result, а поля Flags предоставляет дополнительную информацию, по которой, в частности, можно узнать причину ошибки.		
Поле	Тип	Описание поля
Result	BYTE	Код результата обработки кадра (одно из значений <code>e_LTR210_FRAME_RESULT</code>). Позволяет определить, найден ли был конец кадра и действительны ли данные в кадре
Reserved	BYTE	Резервное поле (всегда равно 0)
Flags	WORD	Дополнительные флаги из <code>e_LTR210_STATUS_FLAGS</code> , представляющие собой информацию о статусе самого модуля и принятого кадра. Может быть несколько флагов, объединенных через логическое ИЛИ

4.2.10 Тип функции для индикации процесса загрузки ПЛИС.

Тип: TLTR210_LOAD_PROGR_CB
Определение: <code>typedef void(APIENTRY * TLTR210_LOAD_PROGR_CB) (void *cb_data, TLTR210 *hnd, DWORD done_size, DWORD full_size)</code>
Описание: Тип для callback-функции, вызываемой при выполнении загрузки прошивки ПЛИС модуля, которая может быть использована для индикации хода выполнения данного процесса. Указатель на функцию данного типа может быть передан в <code>LTR210_LoadFPGA()</code> .
Параметры: cb_data — Указатель, переданный в <code>LTR210_LoadFPGA()</code> в качестве параметра <code>cb_data</code> hnd — Указатель на дескриптор модуля, для которого выполняется загрузка прошивки ПЛИС done_size — Количество байт, которое было успешно записано full_size — Полный размер файла прошивки в байтах

4.3 Функции

4.3.1 Функции инициализации и работы с соединением с модулем.

4.3.1.1 Инициализация дескриптора модуля.

Формат: <code>INT LTR210_Init (TLTR210 *hnd)</code>
Описание: Функция инициализирует поля структуры дескриптора модуля значениями по умолчанию. Эта функция должна вызываться для каждой структуры <code>TLTR210</code> перед вызовом остальных функций.
Параметры: hnd — Дескриптор модуля
Возвращаемое значение: Код ошибки

4.3.1.2 Открытие соединения с модулем.

Формат: INT LTR210_Open (TLTR210 *hnd, DWORD ltrd_addr, WORD ltrd_port, const CHAR *csn, WORD 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 LTR210_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 LTR210_Close (TLTR210 *hnd)
Описание: Функция закрывает ранее открытое с помощью LTR210_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 LTR210_IsOpened (TLTR210 *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)

4.3.1.5 Проверка, загружена ли прошивка ПЛИС модуля.

Формат: INT LTR210_FPGAIsLoaded (TLTR210 *hnd)
Описание: Функция проверяет, загружена ли в данный момент прошивка ПЛИС модуля. Если прошивка загружена, то функция возвращает LTR_OK, иначе — LTR_ERROR_FPGA_IS_NOT_LOADED. Если ПЛИС не загружен, то необходимо выполнить загрузку с помощью LTR210_LoadFPGA() .
Параметры: hnd — Описатель модуля
Возвращаемое значение: Код ошибки (LTR_OK, если прошивка загружена)

4.3.1.6 Загрузка прошивки ПЛИС модуля.

Формат: INT LTR210_LoadFPGA (TLTR210 *hnd, const char *filename, TLTR210_LOAD_PROGR_CB progr_cb, void *cb_data)
Описание: Функция загружает прошивку ПЛИС модуля из указанного файла. Если в качестве имени файла указать нулевой указатель или пустую строку, то для ОС Windows загружается версия прошивки, встроенная в библиотеку как ресурс, а для ОС Linux в этом случае берется файл по стандартному пути (путь установки при сборке библиотеки). Данная функция должна выполняться после открытия связи с модулем перед дальнейшей работой, если прошивка не была загружена до этого. Проверить, загружена ли прошивка ПЛИС можно с помощью LTR210_FPGAIsLoaded() .
Параметры: hnd — Описатель модуля. filename — Имя файла с прошивкой ПЛИС модуля или пустая строка. progr_cb — Указатель на функцию, которая будет вызываться в процессе хода загрузки и может быть использована для индикации прогресса загрузки. Если не используется, то может быть передан нулевой указатель. cb_data — Если указана функция progr_cb, то данный указатель будет передаваться в нее в качестве одноименного параметра.
Возвращаемое значение: Код ошибки

4.3.2 Функции для изменения настроек модуля

4.3.2.1 Запись настроек в модуль.

Формат: INT LTR210_SetADC (TLTR210 *hnd)
Описание: Функция передает настройки, соответствующие значением полей поля Cfg описателя модуля, в модуль. Должна вызываться перед запуском модуля. Возможно вызывать данную функцию и при запущенном сборе данных для изменения ограниченного числа параметров, которые можно менять на лету.
Параметры: hnd — Описатель модуля
Возвращаемое значение: Код ошибки

4.3.2.2 Установка заданной частоты сбора АЦП.

Формат: INT LTR210_FillAdcFreq (TLTR210_CONFIG *cfg, double freq, DWORD flags, double *set_freq)
Описание: Функция заполняет поля AdcFreqDiv и AdcDcmCnt значениями, для которых частота сбора АЦП будет максимально близка к заданной. Может быть вызвана перед LTR210_SetADC() вместо ручного заполнения указанных полей.
Параметры: cfg — Указатель на структуру с конфигурацией модуля freq — Частота сбора АЦП в Гц, которую нужно установить flags — Флаги (резерв — всегда должен передаваться 0) set_freq — В данной переменной сохраняется реальное значение частоты, соответствующее подобранным параметрам. Может быть передан нулевой указатель, если это значение не нужно.
Возвращаемое значение: Код ошибки

4.3.2.3 Установка заданной частоты следования кадров.

Формат: INT LTR210_FillFrameFreq (TLTR210_CONFIG *cfg, double freq, double *set_freq)
Описание: <p>Функция устанавливает значение поля FrameFreqDiv таким, чтобы частота следования кадров в режиме <code>LTR210_SYNC_MODE_PERIODIC</code> была максимально близка к заданной.</p> <p>Может быть вызвана перед <code>LTR210_SetADC()</code> вместо ручного заполнения FrameFreqDiv.</p>
Параметры: <p>cfg — Указатель на структуру с конфигурацией модуля</p> <p>freq — Частота следования кадров в Гц, которую нужно установить</p> <p>set_freq — В данной переменной сохраняется реальное значение частоты, соответствующее подобранным параметрам. Может быть передан нулевой указатель, если это значение не нужно.</p>
Возвращаемое значение: <p>Код ошибки</p>

4.3.3 Функции для управления сбором данных

4.3.3.1 Запуск сбора данных.

Формат: INT LTR210_Start (TLTR210 *hnd)
Описание: <p>При вызове данной функции запускается запись данных АЦП модулем.</p> <p>При непрерывном сборе данных (SyncMode = <code>LTR210_SYNC_MODE_CONTINUOUS</code>), по этой функции запускается сбор данных с АЦП и выдача их в крейт.</p> <p>При покадровом сборе при вызове функции модуль переходит в режим записи во внутренний буфер и ожидания событий синхронизации для выдачи кадров в крейт. Также после вызова данной функции модуль начинает передавать периодически свой статус, если эта возможность разрешена.</p> <p>Хотя бы один из каналов АЦП должен быть до этого разрешен и модуль должен быть сконфигурирован с помощью <code>LTR210_SetADC()</code>.</p>
Параметры: <p>hnd — Описатель модуля</p>
Возвращаемое значение: <p>Код ошибки</p>

4.3.3.2 Останов сбора данных.

Формат: INT LTR210_Stop (TLTR210 *hnd)
Описание: При вызове данной функции модуль прекращает запись данных, ожидание событий синхронизации и выдачу периодических статусов. При этом вычитываются и выбрасываются все переданные, но непрочитанные данные от модуля.
Параметры: hnd — Описатель модуля
Возвращаемое значение: Код ошибки

4.3.3.3 Программный запуск сбора кадра.

Формат: INT LTR210_FrameStart (TLTR210 *hnd)
Описание: Функция посылает команду модулю, по которой возникает программное событие синхронизации. По этому событию модуль начинает сбор кадра (и в режиме мастера вырабатывает импульс на выходе SYNC для ведомых модулей). Может использоваться исключительно в режиме LTR210_SYNC_MODE_INTERNAL . При вызове этой функции запись данных с АЦП уже должна быть разрешена с помощью LTR210_Start() .
Параметры: hnd — Описатель модуля
Возвращаемое значение: Код ошибки

4.3.3.4 Ожидание асинхронного события от модуля.

Формат: INT LTR210_WaitEvent (LTR210 *hnd, DWORD *event, DWORD *status, DWORD tout)
Описание: <p>Функция пытается принять от модуля слово, соответствующее одному из возможных асинхронных событий, в течение заданного таймаута. Функция возвращает управление по приему первого слова, соответствующего какому-либо событию.</p> <p>Если за заданный таймаут события не произошло, то это считается нормальным завершением (функция возвращает LTR_OK), но в качестве кода события возвращается LTR210_RECV_EVENT_TIMEOUT.</p> <p>Для непрерывного сбора данных функция всегда возвращает событие LTR210_RECV_EVENT_SOF и не принимает никаких данных.</p>
Параметры: <p>hnd — Описатель модуля.</p> <p>event — В данной переменной возвращается код произошедшего события (LTR210_RECV_EVENT_TIMEOUT, если никакого события за заданный интервал не произошло).</p> <p>status — Если указан не NULL, то при событии LTR210_RECV_EVENT_KEEPALIVE в данную переменную сохраняется информация о статусе модуля в виде комбинации флагов e_LTR210_STATUS_FLAGS. При остальных событиях данный параметр не изменяется.</p> <p>tout — Время ожидания события в миллисекундах.</p>
Возвращаемое значение: <p>Код ошибки</p>

4.3.3.5 Прием данных от модуля.

Формат: INT LTR210_Recv (TLTR210 *hnd, DWORD *data, DWORD *tmark, DWORD size, DWORD timeout)
Описание: <p>Функция принимает запрашиваемое число слов от модуля. Возвращаемые слова находятся в специальном формате и включают в себя служебную информацию. Для обработки принятых слов и получения значений АЦП используется функция LTR210_ProcessData().</p> <p>При покадровом сборе функция используется для приема кадра после обнаружения начала кадра помощью LTR210_WaitEvent(), а при непрерывном сборе может вызываться без дополнительных вызовов, как и для остальных модулей LTR.</p> <p>Функция возвращает управление либо когда примет запрошенное количество слов, либо по истечению таймаута, а также по обнаружению конца кадра, если используется покадровый сбор.</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. При покадровом режиме можно использовать размер кадра из поля RecvFrameSize описателя модуля.</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>

4.3.3.6 Обработка принятых от модуля слов.

Формат:	INT LTR210_ProcessData (TLTR210 *hnd, const DWORD *src, double *dest, INT *size, DWORD flags, TLTR210_FRAME_STATUS *frame_status, TLTR210_DATA_INFO *data_info)
Описание:	<p>Функция используется для обработки слов, принятых от модуля с помощью LTR210_Recv(). Функция проверяет служебные поля принятых слов, извлекает полезную информацию с отсчетами и, при указании флага LTR210_PROC_FLAG_VOLT, переводит отсчеты в Вольты. Также функция анализирует слово состояние кадра при его наличии и возвращает результат в параметре <code>frame_status</code>.</p> <p>Для получения дополнительной служебной информации можно передать указатель на массив <code>data_info</code>.</p> <p>Функция проверяет также целостность данных с помощью счетчика из служебной информации. По-умолчанию функция предполагает, что все принятые данные обрабатываются и только один раз, проверяя непрерывность счетчика и между вызовами функции. Если это условие не выполняется, то нужно передать флаг LTR210_PROC_FLAG_NONCONT_DATA.</p>
Параметры:	<p>hnd — Описатель модуля.</p> <p>src — Указатель на массив, содержащий слова, принятые от модуля с помощью LTR210_Recv(), которые нужно обработать.</p> <p>dest — Указатель на массив, в который будут сохранены обработанные данные. Порядок следования соответствует порядку во входном массиве (т.е. если разрешены оба канала то отсчеты каналов будут чередоваться).</p> <p>size — На входе принимает размер массива <code>src</code> для обработки. На выходе, при успешном завершении, возвращает количество сохраненных отсчетов в массиве <code>dest</code>.</p> <p>flags — Флаги из e_LTR210_PROC_FLAGS, управляющие работой функции. Может быть объединено несколько флагов через логическое ИЛИ.</p> <p>frame_status — В данной структуре сохраняется состояние принятого кадра, которое формируется на основе слова статуса, переданного за последним словом кадра. По статусу можно судить о правильности данных в кадре. Если слова состояния в обрабатываемых данных не было (т.е. данные соответствуют лишь части кадра, не включая конец), то будет возвращен статус с кодом LTR210_FRAME_RESULT_PENDING.</p> <p>data_info — Массив, в который будет сохранена информация, извлеченная из служебных полей обработанных слов. В частности, в него сохраняется состояние дополнительного служебного бита. Если эта информация не нужна, то можно передать нулевой указатель.</p>
Возвращаемое значение:	Код ошибки.

4.3.4 Функции вспомогательного характера

4.3.4.1 Измерение смещения нуля

Формат: INT LTR210_MeasAdcZeroOffset (TLTR210 *hnd, DWORD flags)
Описание: <p>Функция выполняет сбор одного кадра в режиме измерения собственного нуля, используя при этом установленные в конфигурации значения диапазонов, и сохраняет результат в поля State.AdcZeroOffset. После измерения восстанавливаются прежние настройки АЦП.</p> <p>Измеренные значения могут быть использованы для корректировки принятых данных при передаче флага LTR210_PROC_FLAG_ZERO_OFFS_COR в LTR210_ProcessData(), что позволяет учесть уход нуля для конкретных условий.</p> <p>Эта функция вызывается после конфигурации модуля непосредственно перед запуском записи данных АЦП с помощью LTR210_Start().</p> <p>При изменении диапазонов необходимо заново проводить измерение смещения нуля.</p>
Параметры: <p>hnd — Описатель модуля. flags — Флаги (резерв - должен передаваться 0)</p>
Возвращаемое значение: <p>Код ошибки.</p>

4.3.4.2 Получение прошедшего интервала с момента приема последнего слова.

Формат: INT LTR210_GetLastWordInterval (TLTR210 *hnd, DWORD *interval)
Описание: <p>Функция используется для получения интервала времени, которое прошло с момента успешного приема последнего слова данных от модуля (с помощью LTR210_WaitEvent() или LTR210_Recv()). Может использоваться при разрешенной периодической посылке статуса (сигнала жизни) для проверки исправной работы модуля.</p> <p>Если во время сбора при разрешенной посылке статуса это время превысит допустимый интервал (который должен включать допустимые задержки по интерфейсу передачи крейт->ПК), то это событие можно считать неисправностью модуля.</p>
Параметры: <p>hnd — Описатель модуля. interval — Время в миллисекундах с момента успешного приема последнего слова.</p>
Возвращаемое значение: <p>Код ошибки.</p>

4.3.4.3 Получение сообщения об ошибке.

Формат: LPCSTR LTR210_GetErrorString (INT err)
Описание: 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 ltr210api, так и общие коды ошибок из ltrapi.
Параметры: err — Код ошибки
Возвращаемое значение: Указатель на строку, содержащую сообщение об ошибке.

4.3.4.4 Загрузка коэффициентов в ПЛИС.

Формат: INT LTR210_LoadCbrCoef (TLTR210 *hnd)
Описание: Функция выполняет загрузку коэффициентов из ModuleInfo.CbrCoef в ПЛИС модуля для последующей автокоррекции. Так как в API функциях загрузка коэффициентов из Flash-памяти уже выполняется автоматически после загрузки ПЛИС, то данная функция введена только для того, чтобы можно было использовать свои коэффициенты. Для этого нужно заполнить поля в ModuleInfo.CbrCoef и вызвать данную функцию. Функция может вызываться только в случае, если ПЛИС уже загружен.
Параметры: hnd — Описатель модуля.
Возвращаемое значение: Код ошибки.