

L-CARD

Устройства для мобильных систем

E20-10

Внешний быстродействующий модуль на шину **USB 2.0**

Потоковый ввод с АЦП: 4 канала, 10 МГц, 14 бит

Асинхронный цифровой ввод-вывод: по 16 линий

Асинхронный вывод на ЦАП: 2 канала, 12 бит (опция)

Библиотека *Lusbapi*

Windows'98/Me/2000/XP/Vista/7

История дополнений и изменений

Москва. Март 2011 г.

Ревизия документа АЗ

ООО «Л КАРД»

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

тел. (495) 785-95-25

факс (495) 785-95-14

Адреса в Интернет:

WWW: www.lcard.ru

FTP: [ftp.lcard.ru](ftp://ftp.lcard.ru)

E-Mail:

Общие вопросы: lcard@lcard.ru

Отдел продаж: sale@lcard.ru

Техническая поддержка: support@lcard.ru

Отдел кадров: job@lcard.ru

Представители в регионах:

Украина:	“ХОЛИТ Дэйта Системс, Лтд”	www.holit.com.ua	
Санкт-Петербург:	ЗАО “АВТЭКС Санкт-Петербург”	www.autex.spb.ru	ф
Санкт-Петербург:	Компания "Ниеншанц-Автоматика"	www.nnz-ipc.ru	
Новосибирск:	ООО “Сектор Т”	www.sector-t.ru	ф
Екатеринбург:	ООО “Авеон”	www.aveon.ru	
Казань:	ООО “Шатл”	www.shuttle.kazan.ru	ф

E20-10. Внешний быстродействующий модуль общего назначения на шину **USB 2.0**.

© Copyright 1989–2011, **ООО “Л Кард”**. Все права защищены.

Оглавление

1. Введение	5
2. Версия 3.3 (Октябрь 2009 г.)	5
3. Версия 3.2 (Октябрь 2008 г.)	5
3.1. Общие замечания	5
3.2. Константы	8
3.2.1. Режимы аналоговой синхронизации ввода данных.....	8
3.2.2. Управление входным током смещения.....	8
3.2.3. Переполнение внутреннего буфера модуля	8
3.2.4. Признаки локального переполнения разрядной сетки модуля	8
3.2.5. Признак глобального переполнения разрядной сетки модуля	9
3.3. Структуры	9
3.3.1. Новая структура SYNCHRO_PARS_E2010.....	9
3.3.2. Новая структура DATA_STATE_E2010	9
3.3.3. Автоматическая корректировка данных	10
3.3.4. Управление входным током смещения.....	10
3.3.5. Режимы синхронизации ввода данных	10
3.3.6. Корректировочные коэффициенты	10
3.3.7. Изменения в поле InterKadrDelay	10
3.3.8. Изменения в поле NumberOfWordsToPass	11
3.4. Функции	11
3.4.1. Текущее состояние процесса сбора данных.....	11
4. Версия 3.1 (Апрель 2007 г.)	12
4.1. Общие замечания	12
4.2. Константы	13
4.2.1. Максимальное кол-во виртуальных слотов.....	13
4.2.2. Режимы фиксации перегрузки каналов	13
4.2.3. Типы целостности собираемых данных.....	13
4.2.4. Доступные индексы ревизий модуля	13
4.2.5. Текущее кол-во ревизий модуля.....	13
4.2.6. Доступные ревизии модуля.....	13
4.2.7. Маркеры перегрузки каналов	13
4.3. Структуры	14
4.3.1. Режим фиксации перегрузки каналов	14
4.4. Функции	14
4.4.1. Проверка целостности данных АЦП.....	14
5. Версия 3.0 (Ноябрь 2006 г.)	14

1. Введение

Данный документ предназначен для предоставления пользователю всех дополнений и изменений в развитии штатного программного обеспечения Lusbapi в части работы с быстродействующим модулем E20-10 от фирмы ООО "А Кард".

Последнюю версию библиотеки Lusbapi можно скачать с нашего сайта www.lcard.ru из раздела "Библиотека файлов". Там из подраздела "ПО для внешних модулей" следует выбрать самораспаковывающийся архив lusbapiXY.exe, где X.Y обозначает номер версии программного обеспечения. На момент написания данного руководства последняя библиотека Lusbapi имеет версию 3.3, а содержащий её архив называется lusbapi33.exe.

Последнюю версию основной программы (Firmware) MCU модуля всегда можно скачать с нашего сайта www.lcard.ru из раздела "Библиотека файлов". Там из подраздела "Firmware и BIOS" следует выбрать архив e2010fw_WXa_YZb.zip, где W.X означает номер версии основной программы MCU для модуля E20-10 (Rev.'A'), а Y.Z – для модуля E20-10 (Rev.'B'). На момент написания данного руководства этот архив имеет имя e2010fw_18a_21b.zip.

2. Версия 3.3 (Октябрь 2009 г.)

В библиотеке Lusbapi версии 3.3 было сделано всего два небольших изменения, а именно:

- На данный момент модуль E20-10 стал доступен в двух модификациях (исполнениях):
 - ✓ с полосой пропускания входного сигнала равной 1.25 МГц (базовое исполнение);
 - ✓ с полосой пропускания входного сигнала равной 5.0 МГц (заказное исполнение);С целью информирования пользователя о текущем исполнении модуля в структуру MODULE_INFO_LUSBAPI было введено новое числовое поле Modification. Это поле может принимать два значения: 0 (предопределённая константа BASE_MODIFICATION_E2010) или 1 (предопределённая константа F5_MODIFICATION_E2010)
- Для модуля E20-10 (Rev.'A') в функции ReadData() подправлена нижнее ограничение и кратность величины запроса NumberOfWordsToPass структуры IO_REQUEST_LUSBAPI. Раньше эти величины были равны 128 отсчётам, теперь – 256.

Внесённые изменения могут приводить к небольшой корректировке исходных текстов ранее созданных пользовательских приложений. Кроме того, потребуется полная перекомпиляция ранее созданных приложений.

3. Версия 3.2 (Октябрь 2008 г.)

3.1. Общие замечания

В начале 2008 г. начато производство новой ревизии модуля E20-10 (Rev.'B'). Данная модификация представляет собой продукт весьма основательной модернизации E20-10 (Rev.'A'). Библиотека Lusbapi версии 3.2 призвана осуществлять полную поддержку всех новых функций и свойств, появившихся у модуля E20-10 (Rev.'B'). Подробнее обо всех аппаратных новшествах данного модуля можно прочитать в "E20-10. Руководство пользователя".

В связи с тем, что были внесены довольно значительные дополнения в аппаратную часть модуля, библиотека Lusbapi также подверглась заметным изменениям. Внесённые изменения могут приводить к небольшой корректировке исходных текстов ранее созданных пользовательских приложений. Кроме того, потребуется полная перекомпиляция ранее созданных приложений.

Стоит особо подчеркнуть, что, начиная с версии 3.2, в библиотеке Lusbapi изменился основной файл USB драйвера. Теперь он называется ldevusb.sys вместо бывшего ранее ldevusb.sys. Т.о. при переходе со старых версий Lusbapi на более новую версию 3.2 и выше, конечному пользователю следует через "Device Manager" ("Диспетчер устройств") переключить модуль E20-10 на работу с новым USB драйвером. Подробности смотри в "E20-10. Руководство пользователя. § 4.4. Различия в USB драйверах библиотеки Lusbapi".

Основные изменения библиотеки *Lusbar1* приведены в таблице ниже:

Новое название	Старое название	Смысл
<u>К О Н С Т А Н Т Ы</u>		
NO_ANALOG_SYNCHRO_E2010, ANALOG_SYNCHRO_ON_ RISING_CROSSING_E2010, ANALOG_SYNCHRO_ON_ FALLING_CROSSING_E2010, ANALOG_SYNCHRO_ON_HIGH_ LEVEL_E2010, ANALOG_SYNCHRO_ON_LOW_ LEVEL_E2010, INVALID_ANALOG_SYNCHRO_ E2010	_____	Константы возможных режимов аналоговой синхронизации ввода данных для модуля <i>E20-10 Rev.B</i> и выше.
INPUT_CURRENT_OFF_E2010, INPUT_CURRENT_ON_E2010, INVALID_INPUT_CURRENT_E2010	_____	Константы для управления входным током смещения для модуля <i>E20-10 Rev.B</i> и выше.
BUFFER_OVERRUN_E2010	_____	В битовом поле <i>BufferOverrun</i> структуры <i>DATA_STATE_E2010</i> определяет номер бита, отвечающего за состояние переполнения внутреннего буфера модуля.
OVERFLOW_OF_CHANNEL_1_ E2010, OVERFLOW_OF_CHANNEL_2_ E2010, OVERFLOW_OF_CHANNEL_3_ E2010, OVERFLOW_OF_CHANNEL_4_ E2010	_____	Номера битов признаков локального переполнения разрядной сетки канала АЦП.
OVERFLOW_E2010	_____	Номер бита признака глобального переполнения разрядной сетки модуля за всё время сбора данных от момента <i>START_ADC()</i> до <i>STOP_ADC()</i> .

_____	DATA_INTEGRITY_OK_E2010, DATA_INTEGRITY_BAD_E2010, INVALID_DATA_INTEGRITY_E2010	Константы целостности собираемых с АЦП данных.
<u>С Т Р У К Т У Р Ы</u>		
Структура SYNCHRO_PARS_E2010	_____	В новой структуре сгруппированы все параметры, относящиеся к синхронизации сбора данных. Используется в поле SynchroPars в структуре <i>ADC_PARS_E2010</i> .
Структура DATA_STATE_E2010	_____	В новой структуре содержится текущее состояние процесса сбора данных.
поле IsAdcCorrectionEnabled	_____	Введено новое поле IsAdcCorrectionEnabled структуры <i>ADC_PARS_E2010</i> .
поле InputCurrentControl	_____	Введено новое поле InputCurrentControl структуры <i>ADC_PARS_E2010</i> .
поле SynchroPars	_____	Введено новое поле SynchroPars структуры <i>ADC_PARS_E2010</i> .
поле AdcOffsetCoefs	_____	Введено новое поле AdcOffsetCoefs структуры <i>ADC_PARS_E2010</i> .
поле AdcScaleCoefs	_____	Введено новое поле AdcScaleCoefs структуры <i>ADC_PARS_E2010</i> .
поле InterKadrDelay	_____	Изменился диапазон возможных значений для этого поля структуры <i>ADC_PARS_E2010</i> .
поле NumberOfWordsToPass	_____	Изменился диапазон возможных значений для этого поля структуры <i>IO_REQUEST_LUSBAPI</i> .
<u>Ф У Н К Ц И И</u>		
GET_DATA_STATE()	CHECK_DATA_INTEGRITY()	Изменена функция проверки текущего состояния процесса сбора данных.

3.2. Константы

3.2.1. Режимы аналоговой синхронизации ввода данных

Эти константы объявлены следующим образом:

enum

```
{  
    NO_ANALOG_SYNCHRO_E2010,           // отсутствие аналоговой синхронизации  
    ANALOG_SYNCHRO_ON_RISING_CROSSING_E2010,  
    ANALOG_SYNCHRO_ON_FALLING_CROSSING_E2010,  
    ANALOG_SYNCHRO_ON_HIGH_LEVEL_E2010,  
    LOG_SYNCHRO_ON_LOW_LEVEL_E2010,  
    INVALID_ANALOG_SYNCHRO_E2010,  
};
```

Они задают различные режимы аналоговой синхронизации ввода данных для *E20-10 Rev.B* и выше. Местом использования этих констант, как правило, является поле *SynchroAdMode* структуры *SynchroPars* типа *SYNCHRO_PARS_E2010*, которая является вложенной по отношению к структуре *ADC_PARS_E2010*.

3.2.2. Управление входным током смещения

Эти константы объявлены следующим образом:

enum

```
{  
    INPUT_CURRENT_OFF_E2010, INPUT_CURRENT_ON_E2010,  
    INVALID_INPUT_CURRENT_E2010  
};
```

Они позволяют включать или выключать входной ток смещения для *E20-10 Rev.B* и выше. Местом использования этих констант, как правило, является поле *InputCurrentControl* структуры *ADC_PARS_E2010*.

3.2.3. Переполнение внутреннего буфера модуля

Эта константа объявлена следующим образом:

enum

```
{  
    BUFFER_OVERRUN_E2010 = 0x0,       // переполнение внутреннего буфера модуля  
};
```

Она определяет номер бита в поле *BufferOverrun* структуры *DATA_STATE_E2010*. Этот бит отражает состояние внутреннего буфера модуля.

3.2.4. Признаки локального переполнения разрядной сетки модуля

Эти константы объявлены следующим образом:

enum

```
{  
    OVERFLOW_OF_CHANNEL_1_E2010 = 0x0, OVERFLOW_OF_CHANNEL_2_E2010,  
    OVERFLOW_OF_CHANNEL_3_E2010, OVERFLOW_OF_CHANNEL_4_E2010,  
};
```

Они определяют номера битов в поле *ChannelsOverflow* структуры *DATA_STATE_E2010*. Эти биты отражают признаки локального (за время одного запроса) переполнения разрядной сетки у каждого из физического канала АЦП.

3.2.5. Признак глобального переполнения разрядной сетки модуля

Эта константа объявлена следующим образом:

```
enum
{
    OVERFLOW_E2010 = 0x7
};
```

Она определяет номер бита в поле *ChannelsOverFlow* структуры *DATA_STATE_E2010*. Этот бит отражает признак глобального переполнения разрядной сетки модуля за всё время сбора данных от момента *START_ADC()* до *STOP_ADC()*.

3.3. Структуры

3.3.1. Новая структура SYNCHRO_PARS_E2010

Эта структура объявлена следующим образом:

```
struct SYNCHRO_PARS_E2010
{
    WORD    StartSource;           // источник импульса начала сбора данных с АЦП
    DWORD   StartDelay;           // задержка старта сбора данных в кадрах отсчётов с
                                // АЦП (для модуля Rev. 'B' и выше)
    WORD    SynhroSource;         // источник тактовых импульсов запуска АЦП
    DWORD   StopAfterNKadrs;      // останов сбора данных после задаваемого здесь
                                // кол-ва собранных кадров отсчётов АЦП (для
                                // модуля Rev. 'B' и выше)
    WORD    SynchroAdMode;        // режим аналоговой сихронизации: по переходу
                                // или по уровню (для модуля Rev. 'B' и выше)
    WORD    SynchroAdChannel;     // физический канал АЦП для аналоговой
                                // синхронизации (для модуля Rev. 'B' и выше)
    SHORT   SynchroAdPorog;       // порог срабатывания при аналоговой
                                // синхронизации (для модуля Rev. 'B' и выше)
    BYTE    IsBlockDataMarkerEnabled; // маркирование начала блока данных, что
                                // весьма удобно, например, при аналоговой
                                // синхронизации ввода данных по уровню
                                // (для модуля Rev. 'B' и выше)
};
```

Данная структура сгруппировала в себя все параметры необходимые для задания режима синхронизации ввода данных. В эту структуру были перемещены поля *StartSource* и *SynhroSource* из структуры *ADC_PARS_E2010*. Структура *SYNCHRO_PARS_E2010* входит составной частью в структуру *ADC_PARS_E2010* в виде поля *SynchroPars*.

3.3.2. Новая структура DATA_STATE_E2010

Эта структура объявлена следующим образом:

```
struct DATA_STATE_E2010
{
    BYTE ChannelsOverFlow;        // битовые признаки перегрузки входных каналов
                                // для модуля Rev. 'B' и выше
    BYTE BufferOverrun;           // битовые признаки переполнения внутреннего
                                // аппаратного буфера модуля
    DWORD CurBufferFilling;       // текущая заполненность внутреннего буфера
                                // модуля Rev. 'B' и выше, в отсчётах
    DWORD MaxOfBufferFilling;     // за всё время сбора максимальная заполненность
                                // внутреннего буфера модуля Rev. 'B' и выше, в отсчётах
};
```

```

DWORD BufferSize; // размер внутреннего буфера модуля Rev.'B' и выше, в отсчётах
double CurBufferFillingPercent; // текущая степень заполнения внутреннего
// буфера модуля Rev.'B' и выше, в %
double MaxOfBufferFillingPercent; // за всё время сбора максимальная степень
// заполнения внутреннего буфера модуля Rev.'B' и выше, в %
};

```

Данная структура используется в функции *GET_DATA_STATE()*, позволяя получать всю доступную информацию о текущем состоянии сбора данных (сбои, переполнение и т.д.).

3.3.3. Автоматическая корректировка данных

В структуру *ADC_PARS_E2010* было введено новое поле *IsAdcCorrectionEnabled*. Оно объявлено следующим образом:

```

BOOL IsAdcCorrectionEnabled; // управление автоматической корректировкой
// на уровне ПЛИС модуля получаемых с АЦП
// данных (для модуля Rev.'B' и выше)

```

Данное поле позволяет управлять на уровне ПЛИС модуля автоматической корректировкой получаемых с АЦП данных.

3.3.4. Управление входным током смещения

В структуру *ADC_PARS_E2010* было введено новое поле *InputCurrentControl*. Оно объявлено следующим образом:

```

WORD InputCurrentControl; // управление входным током смещения
// (для модуля Rev.'B' и выше)

```

Данное поле позволяет управлять на уровне ПЛИС модуля автоматической корректировкой получаемых с АЦП данных.

3.3.5. Режимы синхронизации ввода данных

В структуру *ADC_PARS_E2010* было введено новое поле *SynchroPars*, которое само является структурой типа *SYNCHRO_PARS_E2010*. Это поле объявлено следующим образом:

```

SYNCHRO_PARS_E2010 SynchroPars; // параметры синхронизации ввода
// данных с АЦП

```

Данное поле позволяет задавать самые разнообразные режимы синхронизации ввода данных.

3.3.6. Корректировочные коэффициенты

В структуру *ADC_PARS_E2010* было введено два массива *AdcOffsetCoefs[[]]* и *AdcScaleCoefs[[]]*. Эти массивы объявлены следующим образом:

```

double AdcOffsetCoefs[ADC_INPUT_RANGES_QUANTITY_E2010
[ADC_CHANNELS_QUANTITY_E2010];
// массив коэффициентов для корректировки смещение отсчётов АЦП:
// (3 диапазона)*(4 канала) (для модуля Rev.'B' и выше)
double AdcScaleCoefs[ADC_INPUT_RANGES_QUANTITY_E2010
[ADC_CHANNELS_QUANTITY_E2010];
// массив коэффициентов для корректировки масштаба отсчётов АЦП:
// (3 диапазона)*(4 канала) (для модуля Rev.'B' и выше)

```

Данные поля позволяют передавать в модуль коэффициенты для выполнения автоматической корректировки.

3.3.7. Изменения в поле *InterKadrDelay*

Для модуля *E20-10* (*Rev.B* и выше) у поля *InterKadrDelay* структуры *ADC_PARS_E2010* изменилась сетка возможных значений, т.е. теперь диапазон будет от $1/AdcRate$ до $65535/AdcRate$.

3.3.8. Изменения в поле **NumberOfWordsToPass**

Для модуля *E20-10* (*Rev.B* и выше) у поля *NumberOfWordsToPass* структуры *IO_REQUEST_LUSBAPI* изменился диапазон допустимых значений: от **1** до (1024*1024).

3.4. Функции

3.4.1. Текущее состояние процесса сбора данных

В библиотеке заменена функция *CHECK_DATA_INTEGRITY()*. Теперь эта функция объявлена следующим образом:

```
BOOL WINAPI GET_DATA_STATE(DATA_STATE_E2010 * const DataState);
```

Данная функция позволяет получать в структуре типа *DATA_STATE_E2010* всю доступную информацию о текущем состоянии сбора данных (сбои, переполнение и т.д.).

4. Версия 3.1 (Апрель 2007 г.)

4.1. Общие замечания

В этой версии библиотеки Lusbapi для удобства работы программиста были сделаны некоторые изменения и дополнения. Внесённые изменения не являются такими уж существенными и не приводят к корректировке исходных текстов ранее созданных пользовательских приложений. Однако перекомпиляцию проектов таких пользовательских приложений всё-таки придётся выполнить.

Всё привнесённые в библиотеку Lusbapi изменения и дополнения приведены в таблице:

Новое название	Старое название	Смысл
<u>К О Н С Т А Н Т Ы</u>		
MAX_VIRTUAL_SLOTS_ QUANTITY_LUSBAPI	_____	Максимально возможное кол-во обрабатываемых виртуальных слотов.
CLIPPING_OVERLOAD_E2010, MARKER_OVERLOAD_E2010, INVALID_OVERLOAD_E2010	_____	Константы возможных режимов фиксации перегрузки входных каналов модуля.
DATA_INTEGRITY_OK_E2010, DATA_INTEGRITY_BAD_E2010, INVALID_DATA_INTEGRITY_ E2010	_____	Константы целостности собираемых с АЦП данных.
REVISIONS_QUANTITY_E2010	_____	Константа кол-ва ревизий (модификаций) модуля.
REVISIONS_E2010	_____	Константный массив с ревизиями (модификациями) модуля.
ADC_PLUS_OVERLOAD_MARKER, ADC_MINUS_OVERLOAD_MARKER	_____	Константы-маркеры перегрузки входных каналов модуля.
REVISION_A_E2010 INVALID_REVISION_E2010	REV_A_E2010 NO_REV_E2010	Константы индексов ревизий модуля.
<u>С Т Р У К Т У Р Ы</u>		
поле OverloadType	_____	Введено новое поле OverloadType структуры <i>ADC_PARS_E2010</i>
<u>Ф У Н К Ц И И</u>		
CHECK_DATA_INTERGRITY()	_____	Введена новая функция проверки целостности данных АЦП.

4.2. Константы

4.2.1. Максимальное кол-во виртуальных слотов

Эта константа объявлена следующим образом:

```
const WORD MAX_VIRTUAL_SLOTS_QUANTITY_LUSBAPI = 127;
```

Она задаёт максимально возможное кол-во опрашиваемых виртуальных слотов и может быть использована совместно с функцией *OpenLDevice()*.

4.2.2. Режимы фиксации перегрузки каналов

Эти константы объявлены следующим образом:

```
enum { CLIPPING_OVERLOAD_E2010, MARKER_OVERLOAD_E2010,  
        INVALID_OVERLOAD_E2010 };
```

Они задают возможные режимы, которые позволяют по-разному фиксировать факт перегрузки входных каналов модуля. Используются полем *OverloadType* структуры *ADC_PARS_E2010*.

4.2.3. Типы целостности собираемых данных

Эти константы объявлены следующим образом:

```
enum { DATA_INTEGRITY_OK_E2010, DATA_INTEGRITY_BAD_E2010,  
        INVALID_DATA_INTEGRITY_E2010 };
```

Они определяют возможные типы нарушений целостности собираемых с АЦП данных и могут использоваться совместно с функцией *CHECK_DATA_INTEGRITY()*.

4.2.4. Доступные индексы ревизий модуля

Эти константы объявлены следующим образом:

```
enum { REVISION_A_E2010, INVALID_REVISION_E2010 };
```

Они определяют индексы доступных ревизий (модификаций) модуля.

4.2.5. Текущее кол-во ревизий модуля

Эта константа объявлена следующим образом:

```
enum { REVISIONS_QUANTITY_E2010 = INVALID_REVISION_E2010 };
```

Она определяет текущее кол-во ревизий (модификаций) модуля.

4.2.6. Доступные ревизии модуля

Этот константный массив объявлен следующим образом:

```
const BYTE REVISIONS_E2010[REVISIONS_QUANTITY_E2010] = { 'A' };
```

Он определяет названия всех доступных ревизий (модификаций) модуля.

4.2.7. Маркеры перегрузки каналов

Эти константы объявлены следующим образом:

```
enum  
{  
    ADC_PLUS_OVERLOAD_MARKER = 0x5FFF, // маркер 'плюс' перегрузки отсчёта с АЦП  
    ADC_MINUS_OVERLOAD_MARKER = 0xA000 // маркер 'минус' перегрузки отсчёта с АЦП  
};
```

В *маркерном режиме* фиксации перегрузки каналов они определяют значения маркеров перегрузки.

4.3. Структуры

4.3.1. Режим фиксации перегрузки каналов

В структуру *ADC_PARS_E2010* было введено новое поле *OverloadType*. Оно объявлено следующим образом:

```
WORD OverloadMode; // режим фиксации факта перегрузки входных каналов модуля
```

При подаче на канал модуля напряжения, выходящего за пределы установленного входного диапазона, происходит перегрузка каналов либо в 'плюс', либо в 'минус'. Факт наличия перегрузки входных каналов по-разному отражается в каждом отсчёте с АЦП в зависимости от значения поля *OverloadMode*.

4.4. Функции

4.4.1. Проверка целостности данных АЦП

В библиотеку введена новая функция, которая была объявлена следующим образом:

```
BOOL WINAPI CHECK_DATA_INTEGRITY(BYTE * const DataIntegrity);
```

Данная функция возвращает в своём параметре *DataIntegrity* признак целостности передаваемых в компьютер данных АЦП. Для надлежащей работы этой функции необходимо, чтобы версия *основной программы MCU* модуля была **1.7** или выше.

5. Версия 3.0 (Ноябрь 2006 г.)

Эта была первая версия библиотеки *Lusbapi*, с которой началась осуществляться полноценная поддержка быстродействующего модуля *E20-10* от фирмы ООО "ЛКард".