

---

ФЕДЕРАЛЬНОЕ АГЕНТСТВО  
ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

---



НАЦИОНАЛЬНЫЙ  
СТАНДАРТ  
РОССИЙСКОЙ  
ФЕДЕРАЦИИ

ГОСТ Р ИСО/МЭК  
19784-2—  
2010

---

**Автоматическая идентификация**  
**ИДЕНТИФИКАЦИЯ БИОМЕТРИЧЕСКАЯ**  
**Биометрический программный интерфейс**  
**Часть 2**  
**Интерфейс поставщика**  
**биометрической функции архива**

ISO/IEC 19784-2:2007  
Information technology — Biometric application programming interface —  
Part 2: Biometric archive function provider interface  
(IDT)

Издание официальное



Москва  
Стандартинформ  
2011

## Предисловие

Цели и принципы стандартизации в Российской Федерации установлены Федеральным законом от 27 декабря 2002 г. № 184-ФЗ «О техническом регулировании», а правила применения национальных стандартов Российской Федерации — ГОСТ Р 1.0—2004 «Стандартизация в Российской Федерации. Основные положения»

### Сведения о стандарте

1 РАЗРАБОТАН Научно-исследовательским и испытательным центром биометрической техники Московского государственного технического университета имени Н.Э. Баумана (НИИЦ БТ МГТУ им. Н.Э. Баумана) на основе собственного аутентичного перевода на русский язык стандарта, указанного в пункте 4, при консультационной поддержке ассоциации автоматической идентификации «ЮНИСКАН ГС1 РУС»

2 ВНЕСЕН Управлением технического регулирования и стандартизации Федерального агентства по техническому регулированию и метрологии

3 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Приказом Федерального агентства по техническому регулированию и метрологии от 28 сентября 2010 г. № 269-ст

4 Настоящий стандарт идентичен международному стандарту ИСО/МЭК 19784-2:2007 «Информационные технологии. Биометрический программный интерфейс. Часть 2: Интерфейс поставщика биометрической функции архива» (ISO/IEC 19784-2: 2007 «Information technology — Biometric application programming interface — Part 2: Biometric archive function provider interface»), за исключением приложения ДА. Наименование настоящего стандарта изменено относительно наименования указанного международного стандарта для приведения в соответствие с ГОСТ Р 1.5 (раздел 3) и учета его принадлежности к группе стандартов «Автоматическая идентификация».

При применении настоящего стандарта рекомендуется использовать вместо ссылочных международных (региональных) стандартов соответствующие им национальные стандарты, сведения о которых приведены в приложении ДА

### 5 ВВЕДЕН ВПЕРВЫЕ

*Информация об изменениях к настоящему стандарту публикуется в ежегодно издаваемом информационном указателе «Национальные стандарты», а текст изменений и поправок — в ежемесячно издаваемых информационных указателях «Национальные стандарты». В случае пересмотра (замены) или отмены настоящего стандарта соответствующее уведомление будет опубликовано в ежемесячно издаваемом информационном указателе «Национальные стандарты». Соответствующая информация, уведомление и тексты размещаются также в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет*

© Стандартинформ, 2011

Настоящий стандарт не может быть полностью или частично воспроизведен, тиражирован и распространен в качестве официального издания без разрешения Федерального агентства по техническому регулированию и метрологии

## Содержание

1 Область применения . . . . .	1
2 Соответствие . . . . .	1
3 Термины, определения и сокращения . . . . .	1
4 Архитектура интерфейса . . . . .	2
5 Определение интерфейса поставщика биометрической функции архива . . . . .	3
5.1 Структуры данных интерфейса поставщика биометрической функции архива . . . . .	3
5.1.1 BioAFPI_EventHandler . . . . .	3
5.1.2 BioAFPI_BAFPPPropertyID . . . . .	3
5.1.3 BioAFPI_BAFPPPropertySchema . . . . .	3
5.1.4 BioAFPI_UnitPropertyID . . . . .	4
5.1.5 BioAFPI_UnitPropertySchema . . . . .	4
5.2 Функции поставщика биометрической функции архива . . . . .	4
5.2.1 BioAFPI_BAFPLoad . . . . .	4
5.2.2 BioAFPI_BAFPUndload . . . . .	5
5.2.3 BioAFPI_UnitAttach . . . . .	5
5.2.4 BioAFPI_UnitDetach . . . . .	6
5.2.5 BioAFPI_QueryUnits . . . . .	6
5.2.6 BioAFPI_Free . . . . .	7
5.2.7 BioAFPI_ControlUnit . . . . .	7
5.2.8 BioAFPI_Cancel . . . . .	8
5.2.9 BioAFPI_SetPowerMode . . . . .	8
5.2.10 BioAFPI_DbOpen . . . . .	9
5.2.11 BioAFPI_DbClose . . . . .	9
5.2.12 BioAFPI_DbCreate . . . . .	10
5.2.13 BioAFPI_DbDelete . . . . .	10
5.2.14 BioAFPI_DbSetMarker . . . . .	11
5.2.15 BioAFPI_DbFreeMarker . . . . .	11
5.2.16 BioAFPI_DbStoreBIR . . . . .	11
5.2.17 BioAFPI_DbGetBIR . . . . .	12
5.2.18 BioAFPI_DbGetNextBIR . . . . .	13
5.2.19 BioAFPI_DbDeleteBIR . . . . .	13
Приложение ДА (справочное) Сведения о соответствии ссылочных международных стандартов национальным стандартам Российской Федерации . . . . .	14

## Введение

Настоящий стандарт ГОСТ Р ИСО/МЭК 19784-2 определяет интерфейс поставщика биометрической функции архива. Поставщик биометрической функции архива используется для хранения записей биометрической информации и управления ими. Интерфейс поставщика биометрической функции архива включает в свой состав функции для присоединения, отсоединения связанных функциональных модулей БиоАПИ и управления ими со стороны поставщика биометрической услуги, а также функции для хранения записей биометрической информации, получения информации о хранящихся записях биометрической информации и извлечения ранее добавленных записей биометрической информации для их дальнейшей обработки поставщиком биометрической услуги.

Стандарт ИСО/МЭК 19784-1 определяет высокоуровневую общую модель биометрического распознавания. Настоящий стандарт определяет интерфейс поставщика биометрической функции архива. Модуль архива БиоАПИ управляет хранением биометрических данных. Биометрические приложения требуют хранения биометрических данных, поэтому интерфейс модуля архива должен быть стандартизован.

Модуль архива БиоАПИ, в котором хранятся записи биометрической информации, полностью контролируется поставщиком биометрической функции архива. В соответствии с общей моделью записи биометрической информации обрабатываются как записи в базе данных.

## Автоматическая идентификация

## ИДЕНТИФИКАЦИЯ БИОМЕТРИЧЕСКАЯ

## Биометрический программный интерфейс

## Часть 2

## Интерфейс поставщика биометрической функции архива

Automatic identification. Biometric identification. Biometric application programming interface. Part 2. Biometric archive function provider interface

Дата введения — 2011—07—01

## 1 Область применения

Настоящий стандарт определяет интерфейс поставщика биометрической функции архива.

**Примечание** — Наличие интерфейса предполагает, что архив будет управляться как база данных, вне зависимости от его физической реализации. (Смарт-карты, персональные жетоны аутентификации, карты памяти, файлы на жестких дисках и любой другой вид памяти могут обрабатываться с помощью уровня абстракции, обеспечивающего интерфейс базы данных.)

Настоящий стандарт позволяет сторонним производителям создавать поставщики биометрической функции архива, которые могут использоваться любым поставщиком биометрической услуги, поддерживающим интерфейс поставщика биометрической функции архива.

Настоящий стандарт не устанавливает требования по обеспечению безопасности и конфиденциальности при хранении записей биометрической информации и управлении ими.

## 2 Соответствие

Для обеспечения соответствия настоящему стандарту поставщик биометрической функции архива должен поддерживать все функции и параметры, определенные в настоящем стандарте.

Настоящий стандарт не определяет подмножества функций и параметров, необходимых для обеспечения соответствия.

**Примечание** — Поставщику биометрической услуги может потребоваться любая из описанных ниже функций или любой параметр.

## 3 Термины, определения и сокращения

В настоящем стандарте применены следующие термины с соответствующими определениями.

**Примечание** — Названия функций и элементов данных не включены в данный раздел. Их определения приведены в тексте настоящего стандарта.

**3.1 поставщик биометрической функции архива; ПБФА:** Компонент БиоАПИ, связанный с поставщиком биометрической услуги с помощью интерфейса поставщика функции БиоАПИ и предназначенный для хранения записей биометрической информации и обмена ими.

**3.2 интерфейс поставщика биометрической функции архива;** ИПБФА: Интерфейс взаимодействия поставщика биометрической функции и поставщика биометрической услуги, который поддерживает функции управления собственным поставщиком биометрической функции архива, его базами данных и записями биометрической информации.

**3.3 поставщик биометрической услуги;** ПБУ: Компонент, осуществляющий для приложения определенные действия либо с помощью определенного интерфейса путем непосредственного управления одним или несколькими модулями БиоАПИ, либо через поставщиков функции БиоАПИ также с помощью определенного интерфейса.

**3.4 поставщик функции БиоАПИ;** ПБФ: Компонент, управляющий одним или более модулями БиоАПИ определенной категории.

**3.5 интерфейс поставщика услуги;** ИПУ.

**3.6 интерфейс поставщика функции;** ИПФ.

**3.7 универсальный уникальный идентификатор;** УУИД.

**3.8 запись биометрической информации;** ЗБИ.

## 4 Архитектура интерфейса

ИСО/МЭК 19784-1—2006 определяет интерфейс инфраструктуры БиоАПИ, который используется биометрическим приложением для взаимодействия с ПБУ. ПБУ обеспечивает взаимодействие биометрического приложения с биометрическим модулем непосредственно (прямое управление модулем БиоАПИ) или с помощью ПБФ (непрямое управление модулем БиоАПИ, см. рисунки 1 и 2 ИСО/МЭК 19784-1—2006).

ИПБФА содержит следующие типы функций:

а) общие функции управления ПБФ, которые вызываются ПБУ для управления ПБФ и связанными с ним модулями БиоАПИ. Общие функции не имеют непосредственной связи с функциями интерфейса поставщика услуги:

- 1) BioAFPI\_BAFPLoad;
- 2) BioAFPI\_BAFPLoad;
- 3) BioAFPI\_UnitAttach;
- 4) BioAFPI\_UnitDetach;
- 5) BioAFPI\_QueryUnits;

6) BioAFPI\_Free (взаимосвязь данной функции с вызовом соответствующего ИПУ зависит от режима работы ПБУ — если ПБУ копирует данные, для которых ПБФ выделил память, то ИПУ может немедленно вызвать функцию BioAFPI\_Free для ПБФ, в противном случае ПБФ вызовет функцию BioAFPI\_Free, когда им будет получен запрос на очистку памяти, выделенной для этих данных);

б) функции управления модулем БиоАПИ:

- 1) BioAFPI\_ControlUnit (данная функция непосредственно связана с функцией ИПУ);
- 2) BioAFPI\_Cancel (данная функция может быть вызвана в связи с соответствующими вызовами ИПУ или приложения, но может быть вызвана ПБУ и по другим причинам, в том числе и при отсутствии соответствующих запросов от инфраструктуры БиоАПИ или приложения);
- 3) BioAFPI\_SetPowerMode (данная функция непосредственно связана с функцией ИПУ);

с) функции управления базами данных. Функции непосредственно связаны с соответствующими функциями ИПУ:

- 1) BioAFPI\_DbOpen;
- 2) BioAFPI\_DbClose;
- 3) BioAFPI\_DbCreate;
- 4) BioAFPI\_DbDelete;

д) функции управления записями. Функции непосредственно связаны с соответствующими функциями ИПУ:

- 1) BioAFPI\_DbSetMarker;
- 2) BioAFPI\_DbFreeMarker;

е) функции управления данными. Функции непосредственно связаны с соответствующими функциями ИПУ:

- 1) BioAFPI\_DbStoreBIR;
- 2) BioAFPI\_DbGetBIR;
- 3) BioAFPI\_DbGetNextBIR;
- 4) BioAFPI\_DbQueryBIR;
- 5) BioAFPI\_DbDeleteBIR.

## 5 Определение интерфейса поставщика биометрической функции архива

### 5.1 Структуры данных интерфейса поставщика биометрической функции архива

#### 5.1.1 BioAFPI\_EventHandler

Данная структура данных определяет интерфейс обработчика событий для получения асинхронного уведомления о событиях типа BioAPI\_EVENT от модуля БиоАПИ. Примерами событий могут быть подключение или отключение модуля БиоАПИ (например, вставка или извлечение смарт-карты, содержащей биометрические шаблоны для сопоставления на карте или вне карты).

Данный обработчик событий передается поставщику биометрической функции архива во время выполнения функции BioAFPI\_BFPLoad. Он является единственным обработчиком событий, который должны использовать все модули БиоАПИ, управляемые данным ПБФА, для уведомления ПБУ о типе событий, произошедших в загруженном ПБФА.

```
typedef BioAPI_RETURN (BioAPI *BioAFPI_EventHandler)
(const BioAPI_UUID *BAFPUuid,
 BioAPI_UNIT_ID UnitID,
 BioAPI_UNIT_SCHEMA *UnitSchema,
 BioAPI_EVENT EventType);
```

где *BAFPUuid* (входной параметр) — универсальный уникальный идентификатор (УУИД) ПБФА, вызывающего событие;

*UnitID* (входной параметр) — идентификатор модуля БиоАПИ (модуля архива), связанного с событием;

*UnitSchema* (входной параметр) — схема модуля БиоАПИ, вызывающего событие;

*EventType* (входной параметр) — произошедшее событие BioAPI\_EVENT.

#### 5.1.2 BioAFPI\_BAFPPPropertyID

Определяет УУИД формата BioAFPI\_BAFPPPropertySchema. Для соответствия настоящему стандарту ПБФА должен использовать PropertySchema в соответствии со следующим форматом:

```
#define (BioAFPI_BAFPPPropertyID, 0x561403e5, 0x7d2d, 0x489a,
 0x89, 0x41, 0xe7, 0xa9, 0x35, 0x99, 0xcf, 0x9c),
```

#### 5.1.3 BioAFPI\_BAFPPPropertySchema

BioAFPI\_BAFPPPropertySchema содержит информацию о свойствах ПБФ, которые приводятся в реестре компонента и реализуются при вызове соответствующих функций. Свойства ПБФ могут отличаться от свойств модуля архива. Свойства, используемые приложением БиоАПИ или ПБУ, представляют собой конъюнкцию свойств ПБФ и модуля архива.

```
typedef struct _bioafpi_bafp_property_schema {
  uint32_t MaxRecordLength;
  uint32_t MaxRecordNumber;
  uint32_t MaxDatabases;
  uint32_t MaxResponseTime;
  BioAPI_POWER_MODE SupportedPowerModes;
} BioAFPI_BAFPPROPERTY_SCHEMA,
* BioAFPI_BAFPPROPERTY_SCHEMA_PTR;
```

где *MaxRecordLength* — значение, определяющее максимальную длину записи в байтах. Запись представляет собой ЗБИ. В UnitSchema модуля БиоАПИ должно использоваться значение, не превышающее указанное. Значение «0» соответствует отсутствию данных о максимально допустимой длине записи;

*MaxRecordNumber* — значение, определяющее максимальное количество записей. В UnitSchema модуля БиоАПИ должно использоваться значение, не превышающее указанное. Значение «0» соответствует отсутствию данных о максимально допустимом количестве записей;

*MaxDatabases* — значение, определяющее максимальное количество баз данных, которыми может одновременно управлять ПБФА. Значение «0» соответствует отсутствию данных о максимально допустимом количестве баз данных;



- MaxResponseTime* — значение, определяющее максимальное время ожидания возврата из вызванной функции ПБФА. В *UnitSchema* модуля БиоАПИ должно использоваться значение, не превышающее указанное. Значение «0» соответствует отсутствию данных о максимально допустимом времени ожидания;
- SupportedPowerModes* — битовая маска, определяющая режимы энергопотребления, которые могут поддерживаться ПБФ. Поддержка режима энергопотребления означает возможность отправки вызова функции в модуль БиоАПИ, поддерживающий данный режим, и возможность возвращения кода ошибки, связанного с результатом реакции модуля БиоАПИ на отправленный вызов.

#### 5.1.4 BioAFPI\_UnitPropertyID

Определяет УИД формата *BioAFPI\_BAFPPPropertySchema*. Для соответствия настоящему стандарту ПБФА должен использовать *PropertySchema* в соответствии со следующим форматом:

```
#define (BioAFPI_UnitPropertyID, 0x94cae58f, 0x6b26, 0x41fc,  
0x9d, 0xdb, 0xd9, 0x8e, 0x9a, 0xd5, 0x6f, 0xa0).
```

#### 5.1.5 BioAFPI\_UnitPropertySchema

*BioAFPI\_UnitPropertySchema* содержит данные, которые должны присутствовать в связанных вызовах функции. Представляя конкретный модуль архива, данная схема содержит информацию, необходимую для идентификации модуля (*DbUuid* и *DbName*), а также данные о свойствах.

```
typedef struct bioafpi_unit_property_schema {  
    BioAPI_UUID      DbUuid;  
    BioAPI_STRING     DbName;  
    uint32_t          MaxRecordLength;  
    uint32_t          MaxRecordNumber;  
    uint32_t          MaxDatabases;  
    uint32_t          MaxResponseTime;  
    BioAPI_POWER_MODE SupportedPowerModes;  
}BioAFPI_UNIT_PROPERTY_SCHEMA,  
*BioAFPI_UNIT_PROPERTY_SCHEMA_PTR,
```

- где *DbUuid* — УИД базы данных, связанной с модулем архива;
- DbName* — имя базы данных;
- MaxRecordLength* — значение, определяющее максимальную длину записи в байтах. Запись представляет собой ЗБИ. Значение «0» соответствует отсутствию данных о максимально допустимой длине записи;
- MaxRecordNumber* — значение, определяющее максимальное количество записей. Значение «0» соответствует отсутствию данных о максимально допустимом количестве записей;
- MaxDatabases* — значение, определяющее максимальное количество баз данных, которыми может одновременно управлять ПБФА. Значение «0» соответствует отсутствию данных о максимально допустимом количестве баз данных;
- MaxResponseTime* — значение, определяющее максимальное время в миллисекундах, которое процесс, обращающийся к модулю БиоАПИ с помощью функции, должен ожидать возврата из вызываемой функции. Значение «0» соответствует отсутствию данных о максимально допустимом времени ожидания возврата функции из модуля БиоАПИ;
- SupportedPowerModes* — битовая маска, определяющая режимы энергопотребления, которые могут поддерживаться ПБФ. Поддержка режима энергопотребления означает возможность отправки вызова функции в модуль БиоАПИ, поддерживающий данный режим, и возможность возвращения кода ошибки, связанного с результатом реакции модуля БиоАПИ на отправленный вызов.

## 5.2 Функции поставщика биометрической функции архива

### 5.2.1 BioAFPI\_BAFPLoad

```
BioAPI_RETURN BioAPI BioAFPI_BAFPLoad  
(const BioAPI_UUID *BAFPUuid,  
BioAFPI_EventHandler BioAFPINotifyCallback).
```



**Описание**

Данная функция загружает ПБФА.

*BAFPUuid* является идентификатором вызванного ПБФА.

*BioAFPINotifyCallback* определяет обратный вызов, использующийся для уведомления ПБУ о событиях с типом *BioAPI\_EVENT*. ПБФА должен сохранить данную информацию для дальнейшего использования.

**Параметры**

*BAFPUuid* (входной параметр) — УИД вызываемого ПБФА. Используется для обнаружения входа в каталог компонента.

*BioAFPINotifyCallback* (входной параметр) — указатель функции для обработчика события, который управляет событиями с типом *BioAPI\_EVENT*.

**Возвращаемое значение**

Значение *BioAPI\_RETURN* указывает на успешное выполнение функции или определяет тип ошибки. Значение *BioAPI\_OK* соответствует успешному выполнению функции. Остальные значения определяют тип ошибки.

**Ошибки**

*BioAPIERR\_MODULE\_LOAD\_FAIL*

*BioAPIERR\_INVALID\_UUID*

**5.2.2 BioAFPI\_BAFPUndload**

*BioAPI\_RETURN BioAPI BioAFPI\_BAFPUndload*  
(const *BioAPI\_UUID* \**BAFPUuid*,  
*BioAFPI\_EventHandler BioAFPINotifyCallback*).

**Описание**

Данная функция блокирует события и отменяет регистрацию функции уведомления о событиях ПБУ. ПБФА должен выполнить операции по очистке, отменяя инициализацию, выполненную функцией *BioAFPI\_BAFPLoad*.

**Параметры**

*BAFPUuid* (входной параметр) — УИД вызванного ПБФА.

*BioAFPINotifyCallback* (входной параметр) — указатель функции для обработчика события, который управляет событиями с типом *BioAPI\_EVENT*.

**Возвращаемое значение**

Значение *BioAPI\_RETURN* указывает на успешное выполнение функции или определяет тип ошибки. Значение *BioAPI\_OK* соответствует успешному выполнению функции. Остальные значения определяют тип ошибки.

**Ошибки**

*BioAPIERR\_INVALID\_UUID*

**5.2.3 BioAFPI\_UnitAttach**

*BioAPI\_RETURN BioAPI BioAFPI\_UnitAttach*  
(*BioAPI\_UNIT\_ID UnitID*).

**Описание**

Данная функция создает сеанс соединения.

Значение идентификатора модуля, предоставленное ПБУ в качестве входного параметра данной функции, и значение идентификатора модуля, предоставленное инфраструктурой БиоАПИ ПБУ при вызове функции *BioSPI\_BSPAttach* (в случае его наличия), могут не совпадать, если оба значения идентификатора соответствуют одному и тому же модулю БиоАПИ (см. 5.2.5).

**Параметры**

*UnitID* (входной параметр) — значение идентификатора модуля БиоАПИ, определенное в результате предварительного вызова функции *BioAFPI\_QueryUnits*. Значение идентификатора *UnitID* было внесено в поле *UnitId* каждого элемента *BioAPI\_UNIT\_SCHEMA\_ARRAY* ПБФА.

**Возвращаемое значение**

Значение *BioAPI\_RETURN* указывает на успешное выполнение функции или определяет тип ошибки. Значение *BioAPI\_OK* соответствует успешному выполнению функции. Остальные значения определяют тип ошибки.

**Ошибки**

*BioAPIERR\_INVALID\_UNIT\_ID*

*BioAPIERR\_UNIT\_IN\_USE*

### 5.2.4 BioAFPI\_UnitDetach

BioAPI\_RETURN BioAPI BioAFPI\_UnitDetach  
(BioAPI\_UNIT\_ID UnitID).

#### Описание

Данная функция завершает сеанс соединения.

Значение идентификатора модуля, предоставленное ПБУ в качестве входного параметра данной функции, и значение идентификатора модуля, предоставленное инфраструктурой БиоАПИ ПБУ при вызове функции BioSPI\_BSPAttach (в случае его наличия), могут не совпадать, если оба значения идентификатора соответствуют одному и тому же модулю БиоАПИ (см. 5.2.5).

#### Параметры

*UnitID (входной параметр)* — значение идентификатора модуля БиоАПИ, определенное в результате предварительного вызова функции BioAFPI\_QueryUnits. Значение идентификатора UnitID было внесено в поле UnitId каждого элемента BioAPI\_UNIT\_SCHEMA\_ARRAY ПБФА.

#### Возвращаемое значение

Значение BioAPI\_RETURN указывает на успешное выполнение функции или определяет тип ошибки. Значение BioAPI\_OK соответствует успешному выполнению функции. Остальные значения определяют тип ошибки.

#### Ошибки

BioAPIERR\_INVALID\_UNIT\_ID

### 5.2.5 BioAFPI\_QueryUnits

BioAPI\_RETURN BioAPI BioAFPI\_QueryUnits  
(const BioAPI\_UUID \*BafpUuid,  
BioAPI\_UNIT\_SCHEMA \*\*UnitSchemaArray,  
uint32\_t \*NumberOfElements).

#### Описание

Данная функция возвращает массив схем модулей БиоАПИ (см. подраздел 7.55 ИСО/МЭК 19784-1—2006), управляемых данным ПБФА и находящихся в настоящий момент в подключенном состоянии.

**Примечание** — Когда ПБУ вызывает функцию ПБФ *BioAFPI\_QueryUnits*, ПБФ выделяет память для данных, которые должны быть возвращены ПБУ. В некоторых архитектурах реализации ПБУ передает инфраструктуре БиоАПИ данные и адреса в том виде, в котором они были возвращены ПБФ, поскольку инфраструктура БиоАПИ интерпретирует эти адреса так же, как их интерпретирует ПБФ, и поэтому может получить доступ к данным, которые размещены по этим адресам. В других архитектурах реализации ПБУ необходимо переместить все данные, возвращенные ПБФ, во вновь выделенные блоки памяти, к которым имеет доступ инфраструктура БиоАПИ, и вызвать функцию *BioAFPI\_Free* после копирования каждого блока памяти, но до возврата из вызова функции *BioSPI\_QueryUnits*. В первом случае, когда инфраструктура БиоАПИ вызывает функцию *BioSPI\_Free*, ПБУ осуществляет соответствующий вызов функции *BioAFPI\_Free*. Во втором случае вызов функции *BioSPI\_Free* будет осуществляться непосредственно ПБУ. Тем не менее подобные различия в поведении ПБУ не отражаются в инфраструктуре БиоАПИ.

Блок памяти, содержащий массив, должен быть освобожден ПБУ посредством вызова функции *BioAFPI\_Free*, если он больше не используется ПБУ. Блок памяти, на который указывают элементы *UnitProperty* в пределах каждого элемента массива, также должен быть освобожден приложением посредством вызова функции *BioAFPI\_Free*, если он больше не используется ПБУ.

Данная функция может быть вызвана только после вызова функции *BioAFPI\_Load* для определенного ПБУ и не может быть вызвана после вызова функции *BioAFPI\_Unload* для данного ПБУ.

Значение идентификатора модуля, возвращаемое данной функцией для данного модуля БиоАПИ, может не совпадать со значением идентификатора модуля, возвращаемым ПБУ инфраструктуре БиоАПИ. ПБУ может преобразовать любое значение идентификатора модуля (полученное от ПБФ) в другое значение идентификатора модуля перед передачей его инфраструктуре БиоАПИ. Данное преобразование может выполняться во избежание существования повторяющихся идентификаторов модулей в пределах ПБУ, что может иметь место в случае использования ПБУ двух или более ПБФ одной и той же категории или в случае одновременного использования ПБУ ПБФА и непосредственного управления модулями архива.

#### Параметры

*BafpUuid (входной параметр)* — уникальный идентификатор ПБФА, для которого должны быть возвращены данные.

*UnitSchemaArray* (выходной параметр) — указатель на адрес массива элементов типа `BioAPI_UNIT_SCHEMA` (размещенного ПБУ, см. также примечание к описанию данной функции), содержащего информацию о схеме модуля БиоАПИ.

*NumberOfElements* (выходной параметр) — указатель на количество элементов массива, которые управляются данным ПБФА и в настоящий момент находятся в подключенном состоянии.

#### **Возвращаемое значение**

Значение `BioAPI_RETURN` указывает на успешное выполнение функции или определяет тип ошибки. Значение `BioAPI_OK` соответствует успешному выполнению функции. Остальные значения определяют тип ошибки.

#### **Ошибки**

`BioAPIERR_INVALID_UUID`

`BioAPIERR_MEMORY_ERROR`

#### **5.2.6 BioAFPI\_Free**

`BioAPI_RETURN BioAPI BioAFPI_Free`

(`void *Ptr`).

#### **Описание**

Данная функция очищает область памяти, на которую указывает параметр *Ptr*. Если *Ptr* имеет `NULL`-значение, никакие действия не выполняются. Если *Ptr* не соответствует указателю, ранее возвращенному одной из функций `BioAFPI`, или область памяти ранее уже была очищена с помощью вызова функции `BioAFPI_Free`, последствия вызова данной функции не определены.

#### **Параметры**

*Ptr* (входной параметр) — указатель на очищаемую область памяти.

#### **Возвращаемое значение**

Значение `BioAPI_RETURN` указывает на успешное выполнение функции или определяет тип ошибки. Значение `BioAPI_OK` соответствует успешному выполнению функции. Остальные значения определяют тип ошибки.

#### **Ошибки**

#### **5.2.7 BioAFPI\_ControlUnit**

`BioAPI_RETURN BioAPI BioAFPI_ControlUnit`

(`BioAPI_UNIT_ID UnitID`,

`uint32_t ControlCode`,

`const BioAPI_DATA *InputData`,

`BioAPI_DATA *OutputData`).

#### **Описание**

Данная функция отправляет управляющие данные модулю БиоАПИ и получает от модуля БиоАПИ данные о состоянии или функционировании. Содержание элемента *ControlCode*, а также отправляемые данные (входной параметр для функции `BioAFPI_ControlUnit`) и получаемые данные (выходной параметр для функции `BioAFPI_ControlUnit`) будут подробно рассмотрены в спецификации интерфейса, связанного с данным модулем БиоАПИ.

Данная функция выделяет область памяти, достаточную для размещения выходных данных, которые должны быть возвращены приложению, заполняет выделенную область данными и записывает в поля *Length* и *Data* структуры *OutputData* размер и адрес области памяти.

Область памяти, возвращенная в результате вызова одной из функций ИПБФА, будет освобождена приложением с помощью вызова функции `BioAFPI_Free`.

Данная функция является открытым каналом к модулю БиоАПИ и может быть использована для любых целей, которые не могут быть достигнуты стандартизированными функциями (например, ведение архива, восстановление индекса и прочее).

Значение идентификатора модуля, предоставленное ПБУ в качестве входного параметра данной функции, и значение идентификатора модуля, предоставленное инфраструктурой БиоАПИ ПБУ при вызове функции `BioSPI_BSPAttach` (в случае его наличия), могут не совпадать, если оба значения идентификатора соответствуют одному и тому же модулю БиоАПИ (см. 5.2.5).

#### **Параметры**

*UnitID* (входной параметр) — значение идентификатора модуля БиоАПИ, определенное в результате предварительного вызова функции `BioAFPI`.

*ControlCode* (входной параметр) — код вызываемой функции в модуле БиоАПИ.

*InputData (входной параметр)* — адрес и длина буфера, содержащего данные, которые необходимо отправить модулю БиоАПИ в соответствии со значением параметра *ControlCode*.

*OutputData (выходной параметр)* — указатель на структуру *BioAPI\_DATA*. После выполнения функции данный параметр должен содержать адрес и длину буфера, содержащего данные, полученные от модуля БиоАПИ после вызова функции, определенной с помощью *ControlCode*. Если функция не выделила область памяти, то указателю должно быть присвоено *NULL*-значение, а длине буфера данных — значение, равное нулю.

#### **Возвращаемое значение**

Значение *BioAPI\_RETURN* указывает на успешное выполнение функции или определяет тип ошибки. Значение *BioAPI\_OK* соответствует успешному выполнению функции. Остальные значения определяют тип ошибки.

#### **Ошибки**

*BioAPIERR\_MEMORY\_ERROR*  
*BioAPIERR\_INVALID\_POINTER*  
*BioAPIERR\_FUNCTION\_FAILED*

### **5.2.8 BioAFPI\_Cancel**

*BioAPI\_RETURN BioAPI BioAFPI\_Cancel*  
 (*BioAPI\_UNIT\_ID UnitID*).

#### **Описание**

Данная функция отменяет заблокированное в данный момент обращение к модулю БиоАПИ. Возврат из данной функции не должен осуществляться до тех пор, пока заблокированный вызов (обращение) не будет отменен.

Значение идентификатора модуля, предоставленное ПБУ в качестве входного параметра данной функции, и значение идентификатора модуля, предоставленное инфраструктурой БиоАПИ ПБУ при вызове функции *BioSPI\_BSPAttach* (в случае его наличия), могут не совпадать, если оба значения идентификатора соответствуют одному и тому же модулю БиоАПИ (см. 5.2.5).

#### **Параметры**

*UnitID (входной параметр)* — идентификатор модуля БиоАПИ, в котором должен быть отменен заблокированный вызов.

#### **Возвращаемое значение**

Значение *BioAPI\_RETURN* указывает на успешное выполнение функции или определяет тип ошибки. Значение *BioAPI\_OK* соответствует успешному выполнению функции. Остальные значения определяют тип ошибки.

#### **Ошибки**

### **5.2.9 BioAFPI\_SetPowerMode**

*BioAPI\_RETURN BioAPI BioAFPI\_SetPowerMode*  
 (*BioAPI\_UNIT\_ID UnitID*,  
*BioAPI\_POWER\_MODE PowerMode*).

#### **Описание**

Данная функция переводит подключенный модуль БиоАПИ в требуемый режим энергопотребления в том случае, если модуль БиоАПИ поддерживает этот режим.

Значение идентификатора модуля, предоставленное ПБУ в качестве входного параметра данной функции, и значение идентификатора модуля, предоставленное инфраструктурой БиоАПИ ПБУ при вызове функции *BioSPI\_BSPAttach* (в случае его наличия), могут не совпадать, если оба значения идентификатора соответствуют одному и тому же модулю БиоАПИ (см. пункт 5.2.5).

#### **Параметры**

*UnitID (входной параметр)* — идентификатор модуля БиоАПИ, для которого должен быть установлен режим энергопотребления. Параметр *BioAPI\_DONT\_CARE* для данной функции не используется.

*PowerMode (входной параметр)* — 32-битовое значение, определяющее режим энергопотребления, в который следует перевести модуль БиоАПИ.

#### **Возвращаемое значение**

Значение *BioAPI\_RETURN* указывает на успешное выполнение функции или определяет тип ошибки. Значение *BioAPI\_OK* соответствует успешному выполнению функции. Остальные значения определяют тип ошибки.

#### **Ошибки**

*BioAPIERR\_FUNCTION\_NOT\_SUPPORTED*

### 5.2.10 BioAFPI\_DbOpen

BioAPI\_RETURN BioAPI BioAFPI\_DbOpen  
(BioAPI\_UNIT\_ID UnitID,  
BioAPI\_DB\_ACCESS\_TYPE AccessRequest,  
BioAPI\_DB\_MARKER \*Marker).

#### Описание

Данная функция открывает базу данных ЗБИ, поддерживаемую подключенным в настоящее время архивом вызванного модуля БиоАПИ, использованием режима доступа, определяемого параметром *AccessRequest*. Указатель *Marker* базы данных устанавливается на первую запись в базе данных ЗБИ.

**Примечание** — Каждый вызов данной функции должен сопровождаться вызовом функции *BioAFPI\_DbFreeMarker* для освобождения ресурсов, выделенных в функции открытия базы данных. В ПБУ должно быть учтено, что каждый вызов функции *BioAFPI\_DbOpen* приводит к выделению ресурсов, которые не могут быть автоматически освобождены (в противном случае станет невозможным повторное обращение к базе данных с использованием возвращенного дескриптора). Если в ПБУ не планируется повторное обращение к базе данных, следует немедленно вызвать функцию *BioAFPI\_DbFreeMarker*, в противном случае вызов данной функции должен быть выполнен после завершения повторного обращения к базе данных.

#### Параметры

*UnitID* (входной параметр) — идентификатор модуля БиоАПИ, предварительно определенный ПБФА.

*AccessRequest* (входной параметр) — признак требуемого режима доступа к базе данных, возможные значения признака **read** и **write**. В отдельной системе в каждый момент времени только одно приложение может открыть адресуемую базу данных в режиме *BioAPI\_DB\_ACCESS\_WRITE*. Выполнение данного требования контролируется ПБУ.

*Marker* (выходной параметр) — указатель, который используется для повторного обращения к базе данных.

#### Возвращаемое значение

Значение *BioAPI\_RETURN* указывает на успешное выполнение функции или определяет тип ошибки. Значение *BioAPI\_OK* соответствует успешному выполнению функции. Остальные значения определяют тип ошибки.

#### Ошибки

BioAPIERR\_INVALID\_UNIT\_ID  
BioAPIERR\_UNABLE\_TO\_OPEN\_DATABASE  
BioAPIERR\_DATABASE\_IS\_LOCKED  
BioAPIERR\_DATABASE\_DOES\_NOT\_EXIST  
BioAPIERR\_INVALID\_ACCESS\_REQUEST  
BioAPIERR\_DATABASE\_IS\_CORRUPT

### 5.2.11 BioAFPI\_DbClose

BioAPI\_RETURN BioAPI BioAFPI\_DbClose  
(BioAPI\_UNIT\_ID UnitID).

#### Описание

Данная функция закрывает базу данных ЗБИ, поддерживаемую вызовом идентифицируемого модуля БиоАПИ. Все указатели, установленные на записи в базе данных, освобождаются.

**Примечание** — Если база данных, открытая в режиме *DB\_ACCESS\_WRITE*, не будет закрыта, она может быть повреждена.

Значение идентификатора модуля, предоставленное ПБУ в качестве входного параметра данной функции, и значение идентификатора модуля, предоставленное инфраструктурой БиоАПИ ПБУ при вызове функции *BioSPI\_BSPAttach* (в случае его наличия), могут не совпадать, если оба значения идентификатора соответствуют одному и тому же модулю БиоАПИ (см. 5.2.5).

#### Параметры

*UnitID* (входной параметр) — идентификатор модуля БиоАПИ, предварительно определенный ПБФА.

**Возвращаемое значение**

Значение `BioAPI_RETURN` указывает на успешное выполнение функции или определяет тип ошибки. Значение `BioAPI_OK` соответствует успешному выполнению функции. Остальные значения определяют тип ошибки.

**Ошибки**

`BioAPIERR_INVALID_UNIT_ID`  
`BioAPIERR_UNABLE_TO_CLOSE_DATABASE`  
`BioAPIERR_DATABASE_IS_CORRUPT`

**5.2.12 BioAFPI\_DbCreate**

`BioAPI_RETURN BioAPI BioAFPI_DbCreate`  
 (const `BioAPI_UUID *DbUuid`,  
`uint32_t NbrOfRecords`).

**Описание**

Данная функция создает новую базу данных ЗБИ, поддерживаемую присоединенным в данный момент ПБФА. Идентификатор новой базы данных определяется входным параметром `DbUuid`, который должен быть создан биометрическим приложением и должен отличаться от УУИД всех баз данных, с которыми в настоящий момент работает приложение. ПБФ не откроет эту базу данных, поэтому для управления этой базой не будет создан ни один модуль БиоАПИ. После успешного создания базы данных ПБФ архива сначала внесет в реестр компонентов информацию о максимальной длине записи и максимальном количестве записей, затем вызовет событие подключения к ПБУ и, таким образом, представит системе новый модуль архива БиоАПИ во включенном состоянии.

**Параметры**

`DbUuid` (входной параметр) — указатель на УУИД, который будет идентифицировать создаваемую базу данных ЗБИ.

`NbrOfRecords` (входной параметр) — максимальное число записей, которые будут сохранены в базе данных ЗБИ.

**Возвращаемое значение**

Значение `BioAPI_RETURN` указывает на успешное выполнение функции или определяет тип ошибки. Значение `BioAPI_OK` соответствует успешному выполнению функции. Остальные значения определяют тип ошибки.

**Ошибки**

`BioAPIERR_UNABLE_TO_CREATE_DATABASE`  
`BioAPIERR_DATABASE_ALREADY_EXISTS`  
`BioAPIERR_INVALID_UUID`  
`BioAPIERR_INVALID_UNIT_ID`

**5.2.13 BioAFPI\_DbDelete**

`BioAPI_RETURN BioAPI BioAFPI_DbDelete`  
 (`BioAPI_UNIT_ID UnitID`).

**Описание**

Данная функция удаляет все записи из указанной базы данных ЗБИ, а также всю информацию о состоянии этой базы данных. ПБФА также удаляет всю информацию из реестра компонентов. После успешного удаления базы данных ПБФА вызывает событие отключения.

Значение идентификатора модуля, предоставленное ПБУ в качестве входного параметра данной функции, и значение идентификатора модуля, предоставленное инфраструктурой БиоАПИ ПБУ при вызове функции `BioSPI_BSPAttach` (в случае его наличия), могут не совпадать, если оба значения идентификатора соответствуют одному и тому же модулю БиоАПИ (см. 5.2.5).

**Параметры**

`UnitID` (входной параметр) — идентификатор модуля архива, который управляет удаляемой базой данных.

**Возвращаемое значение**

Значение `BioAPI_RETURN` указывает на успешное выполнение функции или определяет тип ошибки. Значение `BioAPI_OK` соответствует успешному выполнению функции. Остальные значения определяют тип ошибки.

**Ошибки**

`BioAPIERR_DATABASE_IS_OPEN`  
`BioAPIERR_INVALID_UNIT_ID`



### 5.2.14 BioAFPI\_DbSetMarker

BioAPI\_RETURN BioAPI BioAFPI\_DbSetMarker  
(BioAPI\_UNIT\_ID UnitID,  
const BioAPI\_UUID \*KeyValue,  
BioAPI\_DB\_MARKER \*Marker).

#### Описание

С помощью данной функции указателю *Marker* присваивается значение для указания на запись, обозначенную параметром *KeyValue* в базе данных ЗБИ, идентифицируемой с помощью параметра *UnitID*. Если параметр *KeyValue* имеет NULL-значение, указатель устанавливается на первую запись в базе данных ЗБИ.

**Примечание** — При возникновении ошибки положение указателя не изменяется.

Значение идентификатора модуля, предоставленное ПБУ в качестве входного параметра данной функции, и значение идентификатора модуля, предоставленное инфраструктурой БиоАПИ ПБУ при вызове функции BioSPI\_BSPAttach (в случае его наличия), могут не совпадать, если оба значения идентификатора соответствуют одному и тому же модулю БиоАПИ (см. 5.2.5).

#### Параметры

*UnitID* (входной параметр) — идентификатор модуля архива, управляющего базой данных.

*KeyValue* (входной параметр) — значение первичного ключа записи в базе данных ЗБИ, на которую должен указывать *Marker*.

*Marker* (входной/выходной параметр) — указатель, который используется для повторного обращения к базе данных, начиная с позиции извлекаемой записи.

#### Возвращаемое значение

Значение BioAPI\_RETURN указывает на успешное выполнение функции или определяет тип ошибки. Значение BioAPI\_OK соответствует успешному выполнению функции. Остальные значения определяют тип ошибки.

#### Ошибки

BioAPIERR\_RECORD\_NOT\_FOUND  
BioAPIERR\_INVALID\_UNIT\_ID

### 5.2.15 BioAFPI\_DbFreeMarker

BioAPI\_RETURN BioAPI BioAFPI\_DbFreeMarker  
(BioAPI\_UNIT\_ID UnitID,  
BioAPI\_DB\_MARKER \*Marker).

#### Описание

Данная функция освобождает память и ресурсы, связанные с определенным указателем *Marker*.

Значение идентификатора модуля, предоставленное ПБУ в качестве входного параметра данной функции, и значение идентификатора модуля, предоставленное инфраструктурой БиоАПИ ПБУ при вызове функции BioSPI\_BSPAttach (в случае его наличия), могут не совпадать, если оба значения идентификатора соответствуют одному и тому же модулю БиоАПИ (см. 5.2.5).

#### Параметры

*UnitID* (входной параметр) — идентификатор модуля архива, управляющего базой данных.

*Marker* (входной параметр) — указатель базы данных ЗБИ, который должен быть освобожден.

#### Возвращаемое значение

Значение BioAPI\_RETURN указывает на успешное выполнение функции или определяет тип ошибки. Значение BioAPI\_OK соответствует успешному выполнению функции. Остальные значения определяют тип ошибки.

#### Ошибки

BioAPIERR\_MARKER\_IS\_INVALID  
BioAPIERR\_INVALID\_UNIT\_ID

### 5.2.16 BioAFPI\_DbStoreBIR

BioAPI\_RETURN BioAPI BioAFPI\_DbStoreBIR  
(BioAPI\_UNIT\_ID UnitID,  
const BioAPI\_INPUT\_BIR \*BIRToStore,  
BioAPI\_UUID \*Uuid).



**Описание**

Данная функция сохраняет ЗБИ, идентифицируемую параметром *BIRToStore*, в открытой базе данных ЗБИ, идентифицируемой параметром *UnitID*. Значение идентификатора модуля, предоставленное ПБУ в качестве входного параметра данной функции, и значение идентификатора модуля, предоставленное инфраструктурой БиоАПИ ПБУ при вызове функции *BioSPI\_BSPAttach* (в случае его наличия), могут не совпадать, если оба значения идентификатора соответствуют одному и тому же модулю БиоАПИ (см. 5.2.5).

Если значение параметра *BIRToStore* определяется дескриптором ЗБИ, то входной дескриптор ЗБИ освобождается. Если значение параметра *BIRToStore* определяется значением первичного ключа базы данных, то ЗБИ копируется в открытую базу данных. Новой ЗБИ в базе данных присваивается новый УУИД, который может использоваться в качестве значения первичного ключа для последующего доступа к ЗБИ.

**Параметры**

*UnitID* (входной параметр) — идентификатор модуля архива, управляющего базой данных.

*BIRToStore* (входной параметр) — ЗБИ, сохраняемая в открытой базе данных (или ЗБИ, или ее дескриптор, или указатель на ЗБИ в другой открытой базе данных).

*Uuid* (выходной параметр) — УУИД, однозначно определяющий новую ЗБИ в базе данных. Данный УУИД не может быть изменен. ЗБИ добавляется в конец базы данных.

**Возвращаемое значение**

Значение *BioAPI\_RETURN* указывает на успешное выполнение функции или определяет тип ошибки. Значение *BioAPI\_OK* соответствует успешному выполнению функции. Остальные значения определяют тип ошибки.

**Ошибки**

*BioAPIERR\_INVALID\_UNIT\_ID*

**5.2.17 BioAFPI\_DbGetBIR**

*BioAPI\_RETURN BioAPI BioAFPI\_DbGetBIR*  
(*BioAPI\_UNIT\_ID UnitID*,  
*const BioAPI\_UUID \*KeyValue*,  
*BioAPI\_BIR\_HANDLE \*RetrievedBIR*,  
*BioAPI\_DB\_MARKER \*Marker*).

**Описание**

Данная функция извлекает ЗБИ, определенную параметром *KeyValue*, в открытой базе данных ЗБИ, определенной параметром *UnitID*. Функция копирует ЗБИ в память ПБУ и возвращает дескриптор ЗБИ. Указатель *Marker* устанавливается на извлекаемую ЗБИ и может быть впоследствии использован для повторного обращения к базе данных, начиная с записи, следующей за извлекаемой.

Блок памяти, возвращенный с помощью функции ИПБФА, должен быть освобожден ПБУ.

Значение идентификатора модуля, предоставленное ПБУ в качестве входного параметра данной функции, и значение идентификатора модуля, предоставленное инфраструктурой БиоАПИ ПБУ при вызове функции *BioSPI\_BSPAttach* (в случае его наличия), могут не совпадать, если оба значения идентификатора соответствуют одному и тому же модулю БиоАПИ (см. 5.2.5).

**Параметры**

*UnitID* (входной параметр) — идентификатор модуля архива, управляющего базой данных.

*KeyValue* (входной параметр) — значение первичного ключа извлекаемой ЗБИ в базе данных ЗБИ.

*RetrievedBIR* (выходной параметр) — дескриптор извлекаемой ЗБИ.

*Marker* (выходной параметр) — указатель на запись, который может быть впоследствии использован для повторного обращения к базе данных, начиная с записи, следующей за извлекаемой.

**Возвращаемое значение**

Значение *BioAPI\_RETURN* указывает на успешное выполнение функции или определяет тип ошибки. Значение *BioAPI\_OK* соответствует успешному выполнению функции. Остальные значения определяют тип ошибки.

**Ошибки**

*BioAPIERR\_INVALID\_UNIT\_ID*

*BioAPIERR\_RECORD\_NOT\_FOUND*

**5.2.18 BioAFPI\_DbGetNextBIR**

BioAPI\_RETURN BioAPI BioAFPI\_DbGetNextBIR  
 (BioAPI\_UNIT\_ID UnitID,  
 BioAPI\_DB\_MARKER \*Marker,  
 BioAPI\_BIR\_HANDLE \*RetrievedBIR,  
 BioAPI\_UUID \*Uuid).

**Описание**

Данная функция извлекает ЗБИ, определенную параметром *Marker*. Функция копирует ЗБИ в память ПБУ и возвращает ее дескриптор, а также указатель на УУИД, который однозначно определяет ЗБИ в базе данных, после чего указатель *Marker* устанавливается на следующую запись в базе данных.

**Примечание** — Если в базе данных больше нет записей, указатель устанавливается на недействительное местоположение.

Значение идентификатора модуля, предоставленное ПБУ в качестве входного параметра данной функции, и значение идентификатора модуля, предоставленное инфраструктурой БиоАПИ ПБУ при вызове функции BioSPI\_BSPAttach (в случае его наличия), могут не совпадать, если оба значения идентификатора соответствуют одному и тому же модулю БиоАПИ (см. 5.2.5).

**Параметры**

*UnitID* (входной параметр) — идентификатор модуля архива, управляющего базой данных.

*Marker* (входной/выходной параметр) — указатель на извлекаемую запись.

*RetrievedBIR* (выходной параметр) — дескриптор извлекаемой ЗБИ.

*Uuid* (выходной параметр) — УУИД, однозначно определяющий извлекаемую ЗБИ в базе данных ЗБИ.

**Возвращаемое значение**

Значение BioAPI\_RETURN указывает на успешное выполнение функции или определяет тип ошибки. Значение BioAPI\_OK соответствует успешному выполнению функции. Остальные значения определяют тип ошибки.

**Ошибки**

BioAPIERR\_INVALID\_UNIT\_ID

BioAPIERR\_END\_OF\_DATABASE

BioAPIERR\_MARKER\_IS\_INVALID

**5.2.19 BioAFPI\_DbDeleteBIR**

BioAPI\_RETURN BioAPI BioAFPI\_DbDeleteBIR  
 (BioAPI\_UNIT\_ID UnitID,  
 const BioAPI\_UUID \*KeyValue).

**Описание**

Данная функция удаляет ЗБИ, определенную параметром *KeyValue*, в открытой базе данных ЗБИ, определенной параметром *UnitID*. Если на удаляемую ЗБИ установлен указатель, то:

a) если удаляемая ЗБИ не является последней ЗБИ в базе данных, указатель устанавливается на следующую ЗБИ;

b) в противном случае указатель устанавливается на недействительное местоположение. Однако дескриптор указателя остается действительным и может быть использован в последующем вызове функции BioAFPI\_DBSetMarker для установки указателя на другую запись.

Значение идентификатора модуля, предоставленное ПБУ в качестве входного параметра данной функции, и значение идентификатора модуля, предоставленное инфраструктурой БиоАПИ ПБУ при вызове функции BioSPI\_BSPAttach (в случае его наличия), могут не совпадать, если оба значения идентификатора соответствуют одному и тому же модулю БиоАПИ (см. 5.2.5).

**Параметры**

*UnitID* (входной параметр) — идентификатор модуля архива, управляющего базой данных.

*KeyValue* (входной параметр) — УУИД удаляемой ЗБИ.

**Возвращаемое значение**

Значение BioAPI\_RETURN указывает на успешное выполнение функции или определяет тип ошибки. Значение BioAPI\_OK соответствует успешному выполнению функции. Остальные значения определяют тип ошибки.

**Ошибки**

BioAPIERR\_INVALID\_UNIT\_ID

BioAPIERR\_RECORD\_NOT\_FOUND

Приложение ДА  
(справочное)Сведения о соответствии ссылочных международных стандартов  
национальным стандартам Российской Федерации

Таблица ДА.1

Обозначение ссылочного международного стандарта	Степень соответствия	Обозначение и наименование соответствующего национального стандарта
ИСО/МЭК 19784-1—2006	IDT	ГОСТ Р 19784-1—2007 (ИСО/МЭК 19784-1—2006) «Спецификация биометрического программного интерфейса»
Примечание — В настоящей таблице использовано следующее условное обозначение степени соответствия стандарта: - IDT — идентичный стандарт.		

---

УДК 004.93'1:006.89

ОКС 35.040

П85

Ключевые слова: автоматическая идентификация, биометрическая идентификация, биометрический архив, поставщик биометрической функции архива, интерфейс поставщика биометрической функции архива

---

Редактор *М.Р. Холодкова*  
Технический редактор *В.Н. Прусакова*  
Корректор *В.И. Варенцова*  
Компьютерная верстка *И.А. Налейкиной*

Сдано в набор 08.04.2011. Подписано в печать 26.07.2011. Формат 60 × 84  $\frac{1}{8}$ . Бумага офсетная. Гарнитура Ариал.  
Печать офсетная. Усл. печ. л. 2,32. Уч.-изд. л. 1,75. Тираж 109 экз. Зак. 662.

---

ФГУП «СТАНДАРТИНФОРМ», 123995 Москва, Гранатный пер., 4.  
[www.gostinfo.ru](http://www.gostinfo.ru) [info@gostinfo.ru](mailto:info@gostinfo.ru)

Набрано во ФГУП «СТАНДАРТИНФОРМ» на ПЗВМ.

Отпечатано в филиале ФГУП «СТАНДАРТИНФОРМ» — тип. «Московский печатник», 105062 Москва, Лялин пер., 6.