

# ibaPDA Plugin

Пользовательские функции для ibaPDA



## Руководство

Версия 1.3

Measurement and Automation Systems



## **Производитель**

iba AG  
Koenigswarterstr. 44  
90762 Fuerth  
Germany

## **Контактная информация**

Центральный офис	+49 911 97282-0
Факс	+49 911 97282-33
Тех. поддержка	+49 911 97282-14
Технологич. отдел	+49 911 97282-13

E-Mail: [iba@iba-ag.com](mailto:iba@iba-ag.com)

Web: [www.iba-ag.com](http://www.iba-ag.com)

Распространение и размножение данного документа, использование и передача его содержания без согласия автора запрещены. Следствием нарушения данных положений является привлечение к ответственности с возмещением нанесенного ущерба.

© iba AG 2010, все права защищены.

Содержание данной публикации было проверено на предмет соответствия описанному аппаратному и программному обеспечению. Отклонения, однако, не могут быть исключены, поэтому гарантия полного совпадения не предоставляется. Информация, содержащаяся в данной брошюре, регулярно актуализируется. Необходимые исправления содержатся в последующих изданиях или могут быть загружены из Интернета.

Актуальную версию можно всегда найти на нашем веб-сайте: <http://www.iba-ag.com>.

## **Уведомление об авторском праве**

Windows® является названием и зарегистрированной торговой маркой компании Microsoft Corporation. Другие продукты и названия компаний, упомянутые в настоящем руководстве, также могут являться зарегистрированными торговыми марками и принадлежать соответствующим лицам.

Версия	Дата	Исправление	Глава	Стр.	Автор	Версия PDA
V 1.3	25 фев 2010	Макет, контактные данные			i	

## Содержание

<b>1</b>	<b>Об этом руководстве пользователя .....</b>	<b>4</b>
1.1	Целевая аудитория .....	4
1.2	Базовые знания .....	4
1.3	Условные обозначения .....	4
1.4	Используемые символы.....	5
<b>2</b>	<b>Введение.....</b>	<b>6</b>
<b>3</b>	<b>Как реализовать плагин .....</b>	<b>7</b>
3.1	Создание проекта плагина .....	7
3.2	Реализация плагина.....	8
3.2.1	Свойство "Имя" .....	8
3.2.2	Функция GetFunctions() .....	8
3.2.3	Функция CreateFunctionObject().....	8
3.2.4	Функция ClearAllFunctionObjects() .....	9
3.3	Реализация IPluginFunction .....	9
3.3.1	Функция Initialize().....	9
3.3.2	Функция Calculate().....	9
3.4	Установка плагина.....	9
<b>4</b>	<b>Как отладить плагин .....</b>	<b>10</b>
4.1	Отладка с использованием тестовой программы.....	10
4.2	Отладка с использованием ibaPDA .....	10
<b>5</b>	<b>Техническая поддержка и контактная информация .....</b>	<b>11</b>

# 1 Об этом руководстве пользователя

В этом руководстве пользователя содержится подробное описание продукта ibaPDA Plugin.

## 1.1 Целевая аудитория

Это руководство предназначено для специалистов, которые работают с электрическими и электронными модулями и обладают необходимыми знаниями в области коммуникационных и измерительных технологий. К вышеупомянутым специалистам относятся лица, которые соблюдают правила техники безопасности и могут оценить возможные последствия и риски, исходя из своей профессиональной подготовки, специальных знаний и опыта, а также знания соответствующих стандартных правил.

## 1.2 Базовые знания

Для работы с программным обеспечением ibaPDA Plugin требуется базовое знание следующих систем и программ:

- ☐ ОС Windows
- ☐ web-браузеров
- ☐ ibaPDA-V6

## 1.3 Условные обозначения

В настоящем руководстве используются следующие условные обозначения:

Действие	Обозначение
Команда меню	Меню "Логическая диаграмма"
Вызов команды меню	"Шаг 1 – Шаг 2 – Шаг 3 – Шаг x" Пример: Выбор меню "Логическая диаграмма – Добавить – Новая логическая диаграмма"
Клавиши	<Название клавиши> Пример: <Alt>; <F1>
Одновременное нажатие клавиш	<Название клавиши> + <Название клавиши> Пример: <Alt> + <Ctrl>
Кнопки	<Название кнопки> Пример: <OK>; <Cancel>
Имена файлов, пути	"Имя файла", "Путь" Пример: "Test.doc"

## 1.4 Используемые символы

При чтении этого руководства вам могут встретиться символы, которые имеют следующее значение:



Несоблюдение техники безопасности может привести к травме или смертельному исходу:

- От удара электрическим током!
- Из-за неправильного использования программных продуктов, которые связаны с процедурами ввода и вывода, имеющими функции управления!

При несоблюдении правил безопасности, относящихся к процессам, системам или управляемым устройствам, существует риск травм или смертельного исхода!

---



Несоблюдение этого правила безопасности может привести к травме или смертельному исходу!

---



Несоблюдение этого правила безопасности может привести к травме или причинить материальный ущерб!

---



### Примечание

В примечании указаны особые требования или действия, которые необходимо выполнить.

---



### Важно

Указывает на некоторые особенности, например исключения из правил.

---



### Совет

Советы, наглядные примеры и маленькие хитрости, позволяющие облегчить работу.

---



### Дополнительная документация

Ссылка на дополнительную документацию или специальную литературу.

---

## 2 Введение

Система ibaPDA plugin была создана для того, чтобы предоставить возможность пользователям ibaPDA создавать собственные функции для выполнения специфических вычислений на основе данных, измеренных системой ibaPDA. Эти функции исполняются системой ibaPDA в реальном времени. Функции могут использоваться в выражениях для виртуальных сигналов так же, как если бы они были встроенными функциями. Они также будут добавлены в построитель выражений.

Чтобы создавать пользовательские функции, сначала нужно создать .NET dll. Эта dll может быть написана на любом языке .NET (C#, C++, VB.NET ...). В следующей главе объясняется, как реализовать такую dll. В третьей главе приводятся различные способы отладки плагина.

Вместе с этой документацией предоставляется образец плагина, который можно найти в папке с образцами (sample). В этой папке содержится два проекта. SamplePlugin - это проект плагина, в котором реализованы две функции: AddMultiple и Integrate. MainProgram - это приложение windows forms, которое будет использовать плагин, а также может применяться для тестирования функций плагина.

## 3 Как реализовать плагин

### 3.1 Создание проекта плагина

Плагин состоит из .NET dll. Эту dll можно создать с помощью Visual Studio .NET 2003 или Visual Studio .NET 2005 (ibaPDA- v6.8.0 представляет собой приложение .NET v1.1, а **НЕ** приложение .NET v2.0).

Создайте новый проект для библиотеки классов на выбранном вами языке. Добавьте ссылку на dll **ibaPdaPluginInterface**. Эта dll содержит определения требуемых интерфейсов и некоторые вспомогательные классы и находится в папке bin дополнительных материалов, прилагающихся к данному руководству. В этой же папке находится документ ibaPdaPluginInterface.xml с дополнительной информацией, которая используется технологией intellisense программы Visual Studio. Этот XML-файл является опциональным.

## 3.2 Реализация плагина

Создайте новый класс, производный от IPlugin. Этот класс будет формировать список всех функций, реализованных в dll и создавать объекты функций для вычисления функций. В синтаксисе C# определение интерфейса выглядит следующим образом:

```
public interface IPlugin
{
    string Name {get;}
    PluginFunctionInfo[] GetFunctions();
    IPluginFunction CreateFunctionObject(string functionName);
    void ClearAllFunctionObjects();
}
```

А в синтаксисе VB.NET – таким образом:

```
Public Interface IPlugin
    ReadOnly Property Name() As String
    Function GetFunctions() As PluginFunctionInfo()
    Function CreateFunctionObject(ByVal functionName As String) As
        IPluginFunction
    Sub ClearAllFunctionObjects()
End Interface
```

### 3.2.1 Свойство "Имя"

Это свойство должно возвращать имя плагина, которое будет отображаться в дереве функций диалогового окна построителя выражений ibaPDA.

### 3.2.2 Функция GetFunctions()

Эта функция должна возвращать список всех функций, реализованных с помощью плагина. Список представляет собой массив объектов PluginFunctionInfo. Класс PluginFunctionInfo имеет следующие свойства:

- ☐ **string** Name: имя функции. Это имя будет отображаться в дереве функций.
- ☐ **int** MinArguments: минимальное количество аргументов для данной функции.
- ☐ **int** MaxArguments: максимальное количество аргументов для данной функции. Это значение может отличаться от значения MinArguments, если функция поддерживает аргументы по умолчанию.
- ☐ **ResultTypeEnum** ResultType: тип результата данной функции. Для результатов в аналоговых значениях: тип *Analog*, для результатов в цифровых значениях: *Digital*. Если тип данных результата совпадает с типом данных первого аргумента: *Invariant*.
- ☐ **string** Prototype: прототип функции. Это должна быть строка, которая выглядит следующим образом: *Add('expr1', 'expr2')*. Аргументы функции должны заключаться в одиночные кавычки (') и разделяться запятыми (,). Прототип должен начинаться с имени функции.
- ☐ **string** Description: описание того, что делает функция. Этот текст отображается в диалоговом окне построителя выражений ibaPDA.

### 3.2.3 Функция CreateFunctionObject()

Эта функция должна возвращать объект, который может вычислить требуемую функцию. Требуемая функция идентифицируется по ее имени. Пояснения



касательно интерфейса `IPluginFunction` приводятся в следующей главе. Эта функция вызывается системой `ibaPDA` для каждого выражения, в котором она используется.

### 3.2.4 Функция `ClearAllFunctionObjects()`

Эта функция вызывается, если сбор данных в `ibaPDA` остановлен. Она позволяет плагину очистить ресурсы, которые были задействованы для вычислений.

## 3.3 Реализация `IPluginFunction`

Функция плагина должна реализовываться классом, производным от `IPluginFunction`. Функция всегда имеет один результат типа `double` и может иметь несколько аргументов типа `double`. В синтаксисе C# определение интерфейса `IPluginFunction` выглядит следующим образом:

```
public interface IPluginFunction
{
    void Initialize(Int64 xBegin, Int64 xBase, double[] inputs);
    double Calculate(Int64 x, double[] inputs);
}
```

А в синтаксисе VB.NET – таким образом:

```
Public Interface IPluginFunction
    Sub Initialize(ByVal xBegin As Int64, ByVal xBase As Int64,
        ByVal inputs As Double())
    Function Calculate(ByVal x As Int64, ByVal inputs As Double())
        As Double
End Interface
```

### 3.3.1 Функция `Initialize()`

Эта функция вызывается системой `ibaPDA` один раз при запуске сбора данных. *xBegin* - это штамп времени первого измеренного значения, выраженного в виде количества 100-наносекундных тактов. *xBase* - это опорное время всех входов и выхода функции. Оно также выражается как количество 100-наносекундных тактов. Массив *входов* содержит первое измеренное значение каждого аргумента функции. Эта функция должна выполнять все инициализации, необходимые для вычисления.

### 3.3.2 Функция `Calculate()`

Эта функция вызывается для каждого измеренного значения. На входе она получает значения аргументов и вычисляет результат функции. Штамп времени *x* - это текущий штамп времени (всегда выражается в 100-наносекундных тактах). Массив *входов* содержит значения каждого аргумента функции на текущем штампе времени *x*. Эта функция будет часто вызываться в процессе сбора данных `ibaPDA`, поэтому она должна быть максимально оптимизирована.

## 3.4 Установка плагина

Установить плагин, чтобы система `ibaPDA` могла его использовать, - очень просто. Плагин нужно просто скопировать в правильную папку. Библиотеку `dll` нужно скопировать во вложенную папку `Plugins` в папке сервера `ibaPDA` (по умолчанию: `c:\Program Files\iba\ibaPda\Server\Plugins`). Все `dll`, находящиеся в папке `Plugins`, загружаются системой `ibaPDA` при запуске. Если вы копируете `dll` в папку во время, когда `ibaPDA` уже запущена, то необходимо выполнить перезапуск системы, чтобы новый плагин загрузился.

## 4 Как отладить плагин

Отладку плагина можно выполнить двумя способами. Можно написать маленькую тестовую программу, в которой размещается плагин и которая вызывает его функции. В решении, предоставляемом в качестве примера, содержится также пример такой тестовой программы. Для размещения плагина также можно использовать ibaPDA, отладка плагина в таком случае выполняется подключением его к процессу ibaPdaService.

### 4.1 Отладка с использованием тестовой программы

Вы можете добавить тестовую программу в свой проект. Эта тестовая программа может непосредственно ссылаться на плагин или может размещать его посредством PluginManager (см. пример). После этого в IPlugin можно вызывать функцию CreateFunctionObject. Далее следует вызвать функцию Initialize для возвращенного объекта IPluginFunction. После этого для каждого измеренного значения можно вызвать функцию Calculate.

Преимущество метода заключается в его скорости. Недостатком является то, что контекст вызова может отличаться от контекста вызова ibaPDA.

### 4.2 Отладка с использованием ibaPDA

Отладку плагина можно выполнить и в том случае, если он загружается системой ibaPDA. Сначала нужно остановить ibaPdaService. Затем следует скопировать dll и pdb-файл плагина в папку Plugins. После этого можно запустить ibaPdaService. Плагин будет загружен. Эту процедуру можно автоматизировать, создав командный файл. Пример командного файла вы найдете в примере решения для плагина (CopyPlugin.bat).

Если ibaPDA запущена, то можно подключиться к процессу с помощью Visual Studio. В Visual Studio откройте меню "Инструменты" (Tools) и выберите отладку (Debug processes). Найдите процесс ibaPdaService.exe и подключитесь к нему. Если вы не можете найти процесс ibaPdaService.exe в списке, поставьте галочку рядом с опцией "Показать системные процессы" ("Show system processes"). В диалоге "Подключиться к процессу" (Attach to Process) выберите только опцию "Общезыковая исполняющая среда" (Common Language Runtime). Не выбирайте Родной (Native). Теперь можно определить точки останова в плагине и приступить к отладке.

Преимущество данного метода в том, что вы отлаживаете непосредственно целевую среду плагина. Недостаток метода в том, что он занимает больше времени.

## 5 Техническая поддержка и контактная информация

### Техническая поддержка

Тел.: +49 911 97282-14

Факс: +49 911 97282-33

E-Mail: [support@iba-ag.com](mailto:support@iba-ag.com)



---

### Примечание

При обращении в службу техподдержки, сообщайте, пожалуйста, серийный номер (iba-S/N) продукта.

---

### Контактная информация

#### Центральный офис

iba AG

Koenigswarterstr. 44

90762 Fuerth

Germany

Тел.: +49 911 97282-0

Факс: +49 911 97282-33

E-Mail: [iba@iba-ag.com](mailto:iba@iba-ag.com)

Конт. лицо: г-н Harald Opel

### По всему миру и в регионах

Контактную информацию касательно вашего местного представителя или представительства компании iba вы можете найти на нашем сайте:

**[www.iba-ag.com](http://www.iba-ag.com)**.