

# API

## Общие сведения

Управление прибором USB-SA44B возможно из программы, разработанной конечным пользователем самостоятельно с учетом конкретных требований. Для этого разработана специальная библиотека SH\_API.dll, предоставляющая соответствующий функционал по управлению устройством.

Взаимодействие пользовательской программы с устройством через данную библиотеку схематично отобразено на рис. 1.

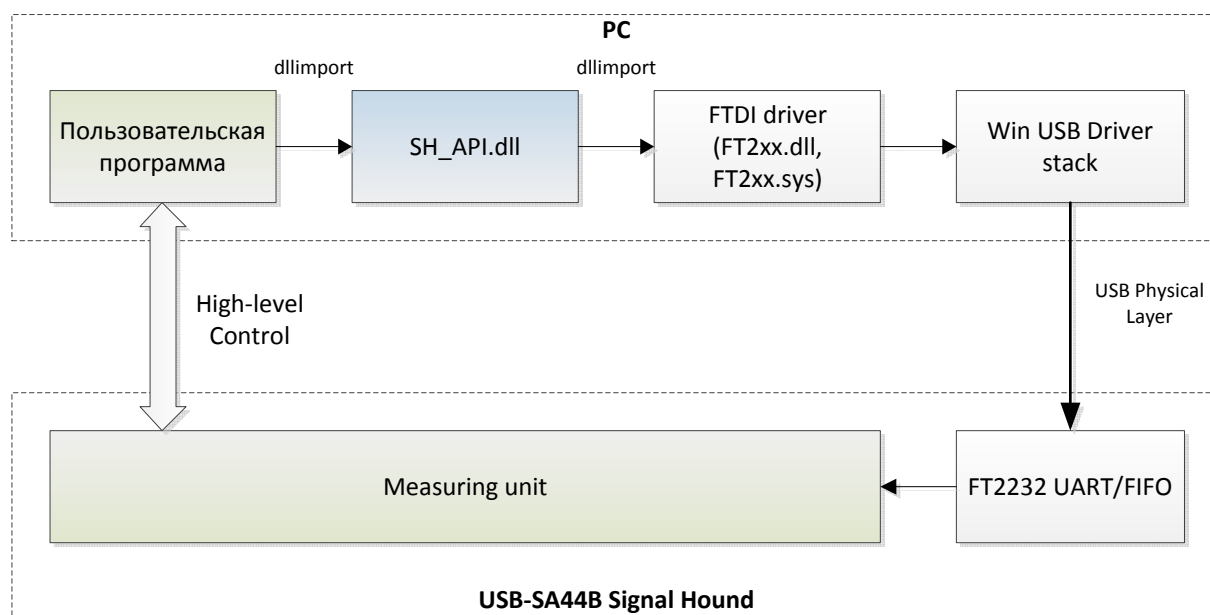


Рис. 1

Как видно из представленного выше рисунка, высокоуровневое управление устройством осуществляется непосредственно через библиотеку SH\_API.dll путем вызова ее функций.

В состав библиотеки SH\_API помимо непосредственно исполняемого файла SH\_API.dll включены:

- заголовочный файл SHAPI.h для языков программирования C/C++;
- библиотека импорта SH\_API.lib, подходящая для связывания импортируемых библиотекой SH\_API.dll символов (функций) посредством сборщика (линкера) объектных файлов среды проектирования VisualStudio, начиная с версии 6.

Для получения возможности вызывать функции библиотеки SH\_API.dll из других сред проектирования (Builder, LabView и др.) обратитесь к документации соответствующего продукта. Возможно, потребуется использовать специальные утилиты, позволяющие импортировать из библиотеки SH\_API.dll список импортируемых символов в формате, подходящем для конкретной среды проектирования.

## Функции библиотеки SH\_API.dll

Все функции, предоставляемые библиотекой конечному пользователю, можно разделить на несколько групп: инициализация и настройка; формирование и получение частотного портрета сигнала в различных режимах качания частоты; передача в клиентскую программу массива квадратурных составляющих (т.н. режим “Zero Span”); другие сервисные функции.

## Инициализация и настройка

В данную группу входит набор функций, выполняющих как однократные действия, инициализирующие прибор, так и задания параметров, отвечающих за текущий режим работы.

Так, функции `SHAPI_Initialize()` и `SHAPI_InitializeNext()` выполняют начальную инициализацию устройства и должны быть вызваны перед первым вызовом какой-либо другой функции. При первом подключении устройства к компьютеру и вызову одной из этих функций производится загрузка калибровочной таблицы из прибора, занимающая продолжительное время (до 20 с), которая сохраняется в файле и последующие вызовы данных функций выполняются существенно быстрее.

Первая из указанных функций выполняет инициализацию текущего активного устройства, вторая — следующего по порядку в случае подключения к компьютеру более чем одного устройства. Большинство управляющих и формирующих функций работают применительно к *текущему* устройству, при этом прибор может быть назначен текущим путем вызова функции `SHAPI_SelectDevice()` с указанием соответствующего номера.

Конфигурирование прибора осуществляется путем вызова функции `SHAPI_Configure()`, которая получает и задает следующие параметры: значение аттенюатора; полосу смесителя (из двух возможных); чувствительность приемника; коэффициент деления, задающий частоту дискретизации; промежуточную частоту (ПЧ; из двух возможных с изменением селективности и чувствительности); частоту дискретизации АЦП. Вызов этой функции следует производить каждый раз, когда изменяются параметры исследуемого сигнала, либо способ оценки полученных результатов. Выполнение функции занимает время порядка 0,5 с.

Функция `SHAPI_SelectExt10MHz()` включает внешний источник опорной частоты 10 МГц, если он физически присутствует.

Функция `SHAPI_SetPreamp()` включает или выключает предусилитель.

Функции `SHAPI_GetTemperature()` и `SHAPI_LoadTemperatureCorrections()` используются для выполнения температурной коррекции.

Сброс устройства к исходным параметрам осуществляется путем вызова функции `SHAPI_CyclePort()`. Выполнение функции занимает время порядка 2,5 с.

## Частотный анализ

В эту группу входят две основные функции: `SHAPI_GetSlowSweep()` и `SHAPI_GetFastSweep()`. Обе эти функции выполняют частотный анализ в заданной полосе частот с использованием быстрого преобразования Фурье (БПФ), возвращая вызывающей процедуре готовый набор отсчетов частотного портрета исследуемого сигнала для отображения, анализа, сохранения и т.д. Различие между функциями заключается в том, что функция `SHAPI_GetSlowSweep()` выполняет усреднение полученных результатов (задается), поэтому выполняется медленнее.

Число возвращаемых отсчетов задается параметром окна БПФ и также возвращается функцией. Этот параметр в сочетании с коэффициентом деления (параметр функции `SHAPI_Configure()`) определяет ширину полосы пропускания приемника.

Возвращаемые отсчеты представляют собой амплитуды (дБм) спектра в задаваемой полосе частот.

Также существует три сервисные функции, которые можно отнести к данному разделу. Это функция `SHAPI_GetRBW()` расчета полосы пропускания приемника по заданному размеру окна БПФ и коэффициенту деления и функции `SHAPI_GetSlowSweepCount()` и `SHAPI_GetFastSweepCount()` расчета числа формируемых отсчетов по начальной и конечной частотам и размеру окна БПФ для варианта с усреднением и без усреднения соответственно.

### **Получение массива квадратурных составляющих**

Функция `SHAPI_GetIQDataPacket()` используется для получения двух массивов квадратурных составляющих сигнала (синфазной и квадратурной) в режиме “Zero Span” для заданной центральной частоты.

Функции `SHAPI_StartStreamingData()` и `SHAPI_StopStreamingData()` соответственно включают и выключают режим непрерывной передачи указанных массивов из прибора в компьютер. Вычитывание очередной пары полученных пакетов осуществляется функцией `SHAPI_GetStreamingPacket()` и должно осуществляться достаточно быстро для избежания потери пакетов, которые, будучи не вычитанными, перезаписываются новыми. Задание центральной частоты выполняется функцией `SHAPI_GetIQDataPacket()`. Управление частотой сэмплирования и ПМ выполняется изменением параметров функции `SHAPI_Configure()`.