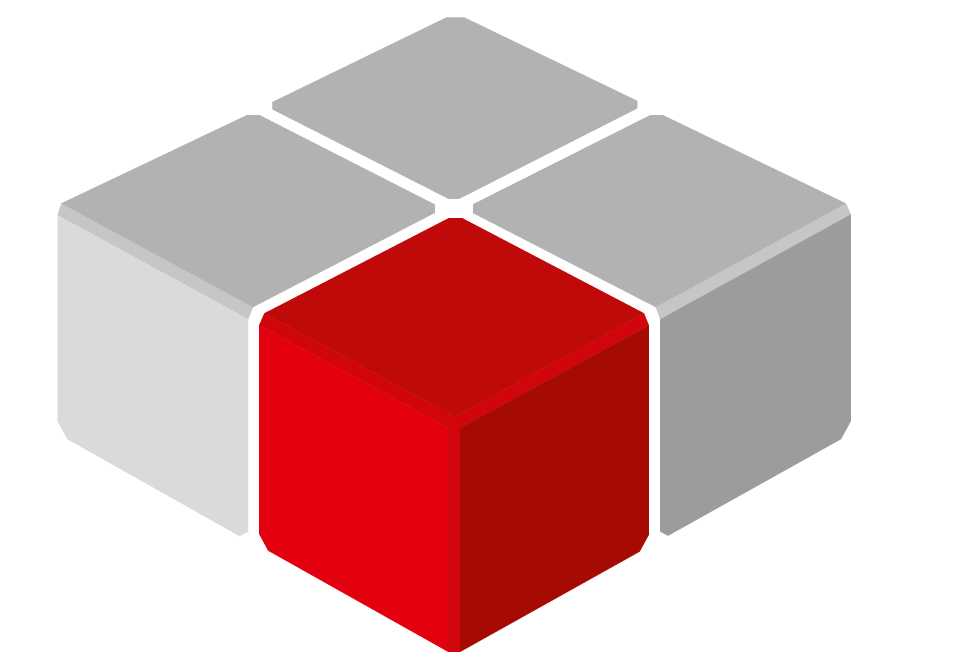


**CODESYS V3.5**

**Описание библиотеки CmpOwenUsbHid**



Руководство пользователя

20.06.2020

версия 2.0

Оглавление

[1 Цель документа 3](#_Toc38975199)

[2 Описание библиотеки CmpOwenUsbHid 4](#_Toc38975200)

[2.1 Установка библиотеки 4](#_Toc38975201)

[2.2 Добавление библиотеки в проект CODESYS 5](#_Toc38975202)

[2.3 Описание библиотеки 6](#_Toc38975203)

[2.3.1 ФБ HID\_EVENT 6](#_Toc38975204)

# Цель документа

Настоящее руководство представляет собой описание библиотеки **CmpOwenUsbHid**, которая позволяет получать данные от USB HID устройств (например, клавиатуры, сканера штрихкодов и т. д.), подключенных к порту контроллера **USB A**. Значение максимального тока потребления подключаемых устройств приведено в руководстве по эксплуатации на соответствующий контроллер. Реализация библиотеки находится в **Linux** (библиотека **CODESYS** представляет собой только интерфейс), поэтому может меняться в зависимости от версии прошивки контроллера. В данном документе описана версия библиотеки **3.5.14.1**.

Библиотека поддерживается следующими контроллерами:

* СПК1хх [М01] начиная с прошивки **1.2.0623.1009**;
* ПЛК2xx начиная с прошивки **1.2.0623.0953**.

Пример работы с библиотекой: [Example\_UsbHidExample\_3514v1.projectarchive](https://owen.ua/uploads/121/example_usbhidexample_3514v1.projectarchive)

Пример создан в среде **CODESYS V3.5 SP14 Patch 3** и подразумевает запуск на **СПК1хх [М01]** с таргет-файлом **3.5.14.34.** В случае необходимости запуска проекта на другом устройстве следует изменить таргет-файл в проекте (**ПКМ** на узел **Device** – **Обновить устройство).**

|  |  |
| --- | --- |
|  | **ПРИМЕЧАНИЕ**  Библиотека может использоваться только в контроллерах ОВЕН, программируемых в среде **CODESYS V3.5.** |

|  |  |
| --- | --- |
|  | **ПРИМЕЧАНИЕ**  Библиотека является внешней (external) и не может быть использована в симуляции или на виртуальном контроллере. |

# Описание библиотеки CmpOwenUsbHid

## Установка библиотеки

Библиотека **CmpOwenUsbHid** доступна на сайте компании [ОВЕН](http://www.owen.ua/) в разделе **CODESYS V3/Библиотеки**.

Для установки библиотеки в **CODESYS** в меню **Инструменты** следует выбрать пункт **Репозиторий библиотек**, после чего нажат кнопку **Установить** и указать путь к файлу библиотеки:

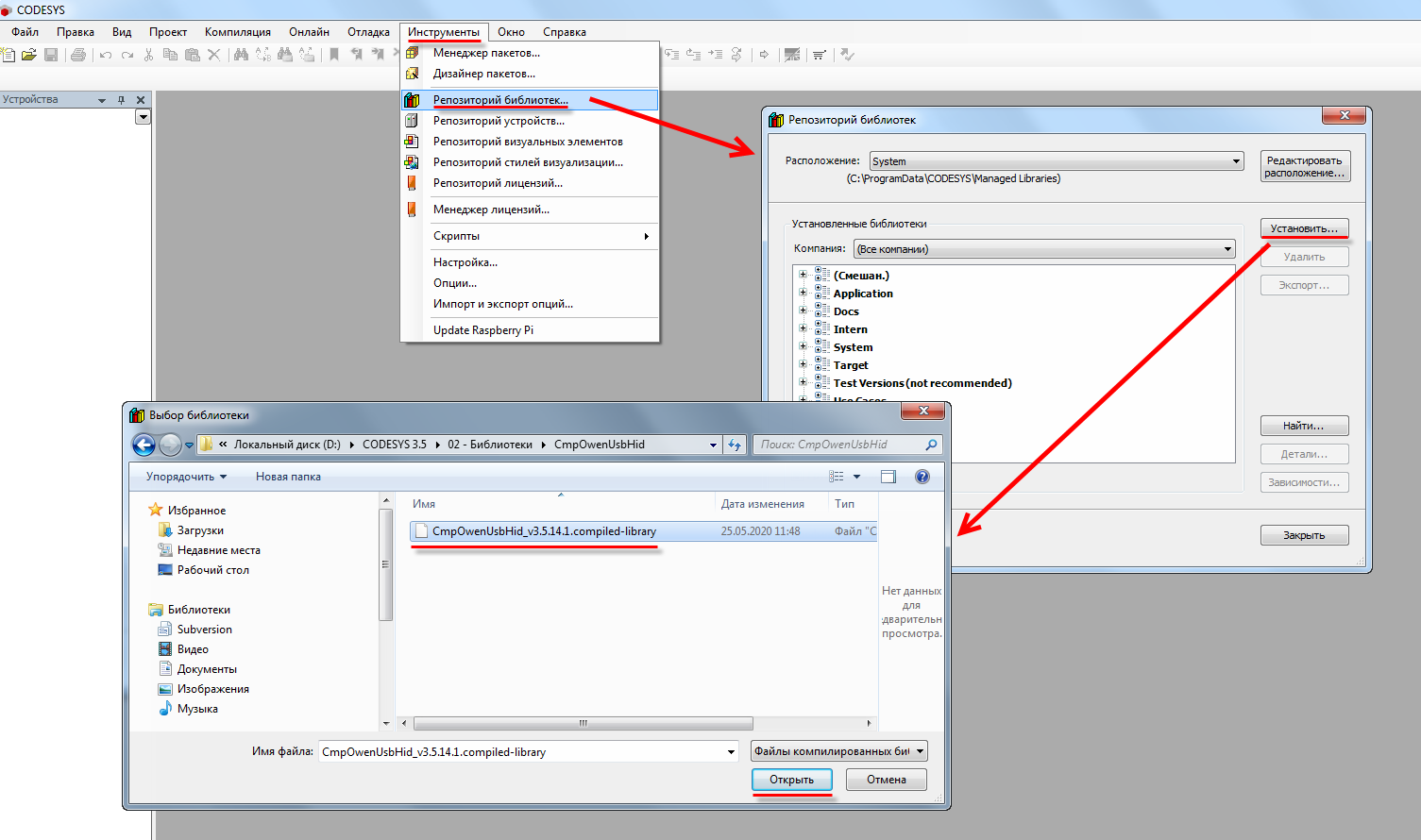


Рисунок 2.1 – Установка библиотеки

## Добавление библиотеки в проект CODESYS

Для добавления библиотеки **CmpOwenUsbHid** в проект **CODESYS** в **Менеджере библиотек** следует нажать кнопку **Добавить библиотеку**, в появившемся списке выбрать библиотеку **CmpOwenUsbHid** и нажать **ОК**.

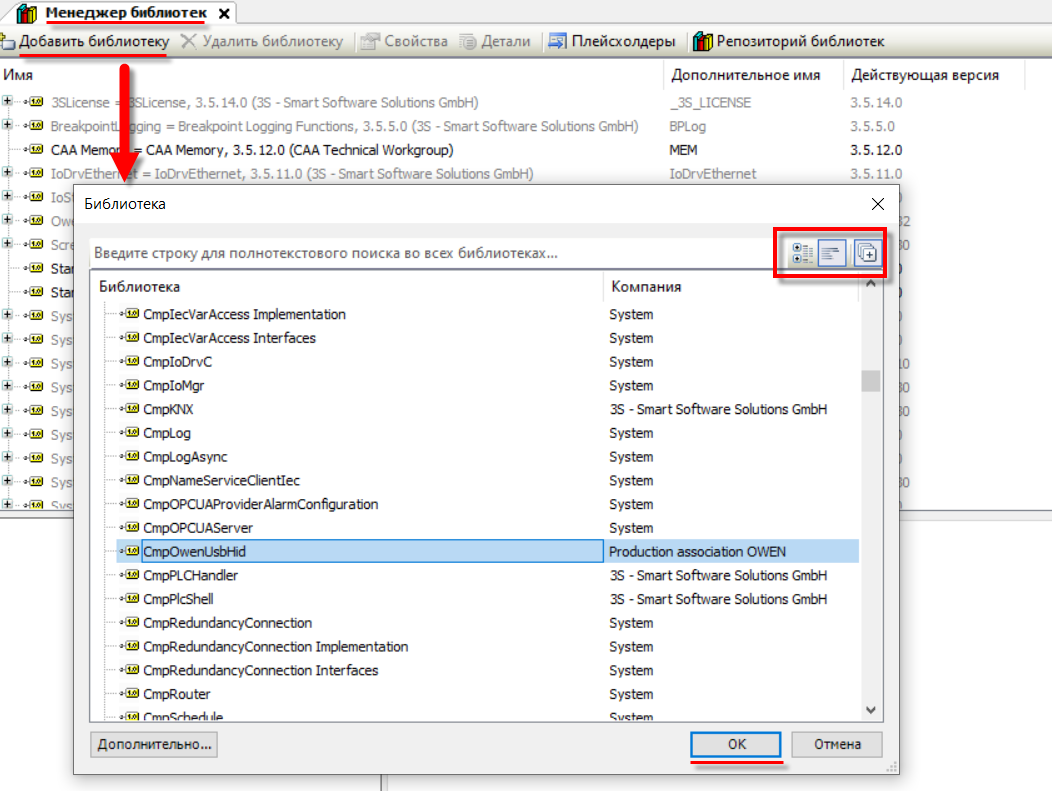
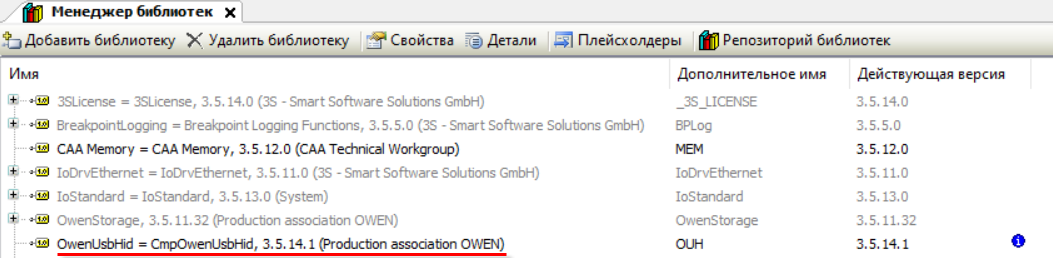


Рисунок 2.2 – Добавление библиотеки CmpOwenUsbHid

После добавления библиотека появится в списке **Менеджера библиотек**:

 Рисунок 2.3 – Список библиотек проекта

|  |  |
| --- | --- |
|  | **ПРИМЕЧАНИЕ**  При обращении к функциям и ФБ библиотеки следует перед их названием указывать префикс **OUH** (пример: **OUH.HID\_EVENT**). |

## Описание библиотеки

### ФБ HID\_EVENT

Функциональный блок **HID\_EVENT** используется для получения данных от USB HID устройства. Пока вход **xEnable** имеет значение **TRUE** – блок находится в работе. Если выход **xActive** имеет значение **TRUE** – блок готов к работе и ожидает получения новых данных. Для устройств, передающих символьные данные (клавиатура, сканер штрихкодов) окончание строки определяются по отсутствию новых символов в течение времени **tCharTimeout**. При подключении таких устройств выход **xIsKeyboard** будет иметь значение **TRUE**.

При получении новых данных выход **xDone** принимает значение **TRUE** на один цикл ПЛК и в течение этого цикла выходы **sResult**, **diX**, **diY**, **diWheel**, **dwButtons** имеют актуальные значения (первый выход используется для получения символьных данных, остальные – для координат и обработки нажатий устройств типа «мышь»). См. более подробное описание в таблице 2.1.

Вход **byDevID** определяет идентификатор USB-устройства. К контроллеру может быть подключено до 5 UDB HID-устройств (через USB Hub с активным питанием), которые будут обрабатываться отдельными экземплярами данного блока.

Для **СПК1хх** идентификатор лежит в диапазоне **1…5**, для **ПЛК210** – в диапазоне **0…4**. Подключенное устройство определяется в Linux как **/dev/input/eventX** (где X – идентификатор, который надо присвоить на вход **byDevId**). Просмотреть список устройств можно в терминале Linux с помощью команды **ls -lah /dev/input**. Некоторые HID-устройства могут быть составными (т. е. при подключении одного устройства могут появится устройства **event1** и **event2**). В этом случае идентификатор устройства должен быть определен опытным путем из имеющегося списка.

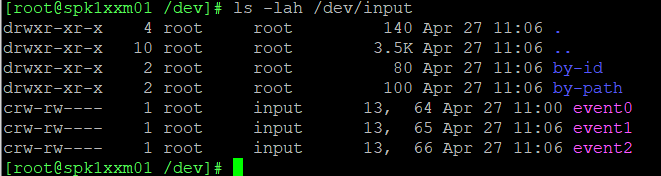


Рисунок 2.4 – Определение идентификаторов подключенных устройство в терминале Linux

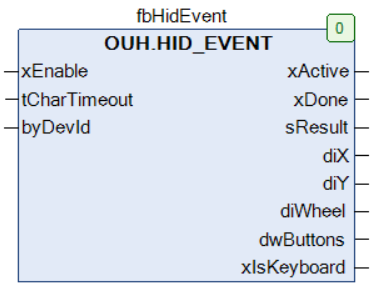


Рисунок 2.5 – Внешний вид ФБ HID\_EVENT на языке CFC

**Таблица 2.1 – Описание входов и выходов ФБ HID\_EVENT**

|  |  |  |
| --- | --- | --- |
| **Имя переменной** | **Тип** | **Описание** |
| **Входные переменные** | | |
| xEnable | BOOL | Пока вход имеет значение **TRUE** – блок находится в работе |
| tCharTimeout | TIME | Если в течение заданного времени в порт не приходит новых символов, то прием строки данных считается завершенным |
| byDevId | BYTE | Идентификатор подключенного устройства (см. выше).  Для **СПК1хх** идентификатор лежит в диапазоне **1…5**, для **ПЛК210** – в диапазоне **0…4** |
| **Выходные переменные** | | |
| xActive | BOOL | **TRUE** – блок готов к работе и ожидает получения данных |
| xDone | BOOL | Принимает значение **TRUE** на один цикл ПЛК при получении новых данных. В течение этого цикла выходы **sResult**, **diX**, **diY**, **diWheel**, **dwButtons** имеют актуальные значения |
| sResult | STRING(255) | Строка данных, полученная от символьного USB HID устройства (клавиатуры, сканера штрихкодов) |
| diX | DINT | Смещение координаты курсора мыши по оси X (разность между текущим и предыдущим значением) |
| diY | DINT | Смещение координаты курсора мыши по оси Y (разность между текущим и предыдущим значением) |
| diWheel | DINT | Направление прокрутки колесика мыши (**1** - вверх, **-1** – вниз, **0** – прокрутка не осуществляется) |
| dwButtons | DWORD | Битовая маска нажатых кнопок мыши (соответствие бит и кнопок зависит от модели мыши) |
| xIsKeyboard | BOOL | Тип устройства. **TRUE** – символьное устройство (клавиатура, сканер штрихкодов), **FALSE** – другое устройство (например, мышь) |

|  |  |
| --- | --- |
|  | **ПРИМЕЧАНИЕ**  Для определения подключенного устройства может потребоваться перезагрузка контроллера. |

|  |  |
| --- | --- |
|  | **ПРИМЕЧАНИЕ**  При работе с символьными устройствами с помощью библиотеки поддерживается только англоязычная раскладка клавиатуры. Символы принимаются в верхнем регистре. Обработка одновременного нажатия нескольких клавиш не поддерживается. |