

***NaftaVision***



## **Руководство по настройке библиотеки DLL**

2023

# Содержание

<b>1. Условные обозначения и термины.....</b>	<b>3</b>
1.1. Условные обозначения.....	3
1.2. Перечень терминов и сокращений.....	3
<b>2. Введение.....</b>	<b>4</b>
<b>3. Операции с библиотекой DLL.....</b>	<b>5</b>
3.1. Добавление библиотеки DLL в проект.....	5
3.2. Переименование библиотеки DLL.....	7
3.3. Удаление библиотеки DLL из проекта.....	8
<b>4. Операции с картой DLL функций.....</b>	<b>9</b>
4.1. Создание карты DLL функций.....	9
4.2. Переименование карты DLL функций.....	10
4.3. Удаление карты DLL функций.....	10
<b>5. Операции с картой DLL тегов.....</b>	<b>11</b>
5.1. Создание карты DLL тегов.....	11
5.2. Переименование карты DLL тегов.....	12
5.3. Удаление карты DLL тегов.....	12
<b>6. Настройка карт DLL.....</b>	<b>13</b>
6.1. Настройка карты DLL функций.....	13
6.2. Настройка карты DLL тегов.....	15

# 1. Условные обозначения и термины

## 1.1. Условные обозначения



### **Внимание:**

Помечает информацию, с которой необходимо ознакомиться, чтобы учесть особенности работы какого-либо элемента программного обеспечения.



### **ОСТОРОЖНО:**

Помечает информацию, с которой необходимо ознакомиться, чтобы предотвратить нарушения в работе программного обеспечения либо предотвратить потерю данных.



### **ОПАСНО:**

Помечает информацию, с которой необходимо ознакомиться, чтобы избежать потери контроля над технологическим процессом.

## 1.2. Перечень терминов и сокращений

### **ПО**

Программное обеспечение.

### **Проект**

Набор данных, который представляет конфигурацию SCADA.

### **SCADA**

Supervisory Control And Data Acquisition - диспетчерское управление и сбор данных.

## 2. Введение

Документ "Руководство по настройке библиотеки DLL" (далее Руководство) относится к комплексу эксплуатационных документов программного обеспечения (ПО).

DLL - динамическая библиотека, то есть файл, содержащий машинный код, загружаемый по запросу сервера ввода-вывода.

В зависимости от операционной системы, расширения имён файлов динамических библиотек:

1. .dll (dynamic-link library) - Windows;
2. .so (shared object) - Linux.

Назначение руководства - обеспечить пользователя информацией об алгоритме настройки обмена данными с библиотекой DLL. Данный алгоритм включает в себя следующие действия:

- добавление в проект библиотеки DLL;
- добавление и настройка карт DLL функций и DLL тегов.



**Внимание:** Справочная информация доступна:

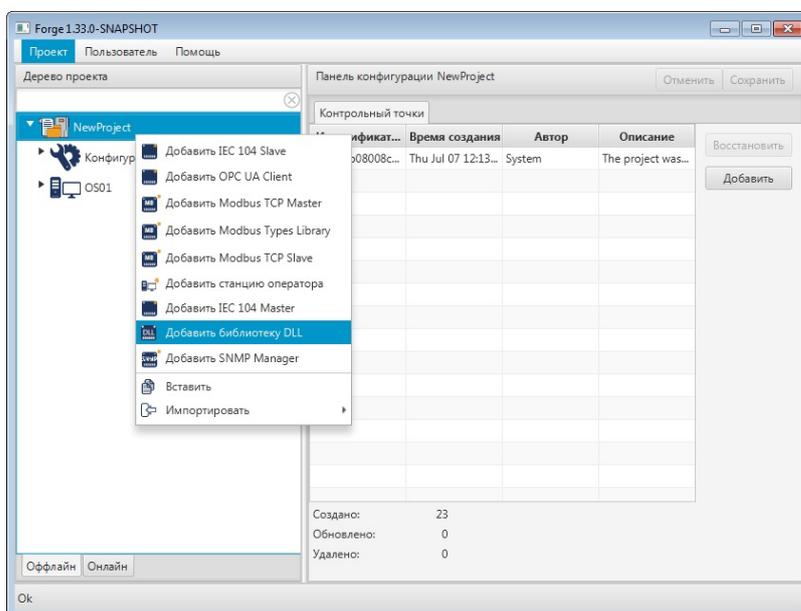
- из главного меню командой **Помощь > Справка**;
- по клавише **“F1”**;
- выбором пункта **Справка** из контекстного меню дерева проекта.

## 3. Операции с библиотекой DLL

### 3.1. Добавление библиотеки DLL в проект

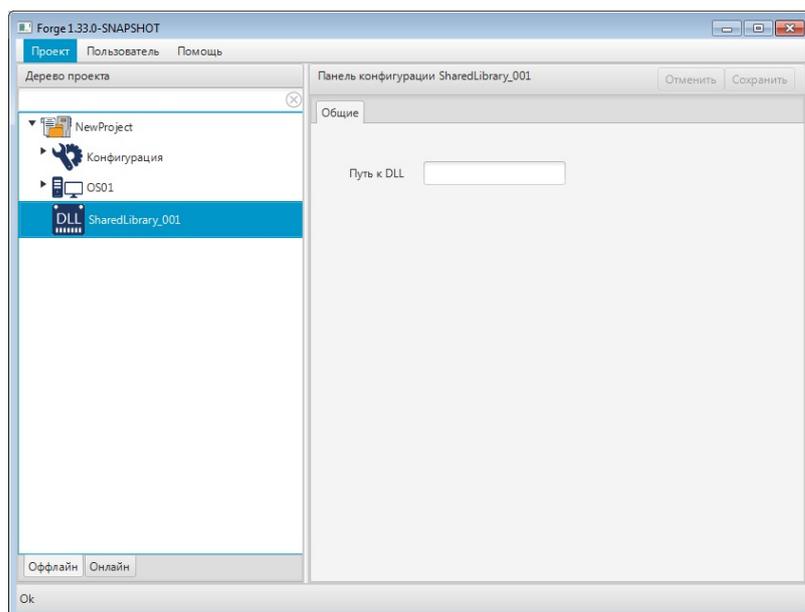
Для того чтобы добавить библиотеку DLL в проект, необходимо выполнить следующие действия:

1. В дереве проекта правой кнопкой мыши выберите имя проекта.
2. В открывшемся контекстном меню выберите **Добавить библиотеку DLL**:



**Рисунок 1. Добавление библиотеки DLL**

Новая библиотека DLL будет отображена в дереве проекта:



**Рисунок 2. Библиотека DLL в дереве проекта**

Указать путь к библиотеке DLL можно в соответствующем поле вкладки **Общие** Панели конфигурации.

После редактирования поля ввода сохраните/отмените настройки, нажав на соответствующие кнопки Сохранить/Отменить в правом верхнем углу. Результат настройки библиотеки DLL будет сохранен в панели конфигурации.

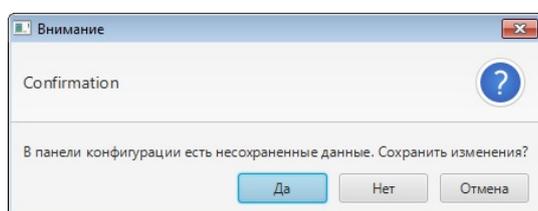


**Внимание:** При выборе другой ноды, если в окне **Панель конфигурации** имеются несохраненные изменения, Forge предложит сохранить их через окно подтверждения, где

**Да** - сохранение измененных данных;

**Нет** - восстановление данных из базы данных;

**Отмена** - остаться на измененной ноды для дальнейшего редактирования.



**Рисунок 3. Окно подтверждения**

## 3.2. Переименование библиотеки DLL

Для того чтобы переименовать библиотеку DLL, необходимо выполнить следующие действия:

1. В дереве проекта правой кнопкой мыши выберите имя нужной библиотеки DLL.
2. В открывшемся контекстном меню выберите **Переименовать**:

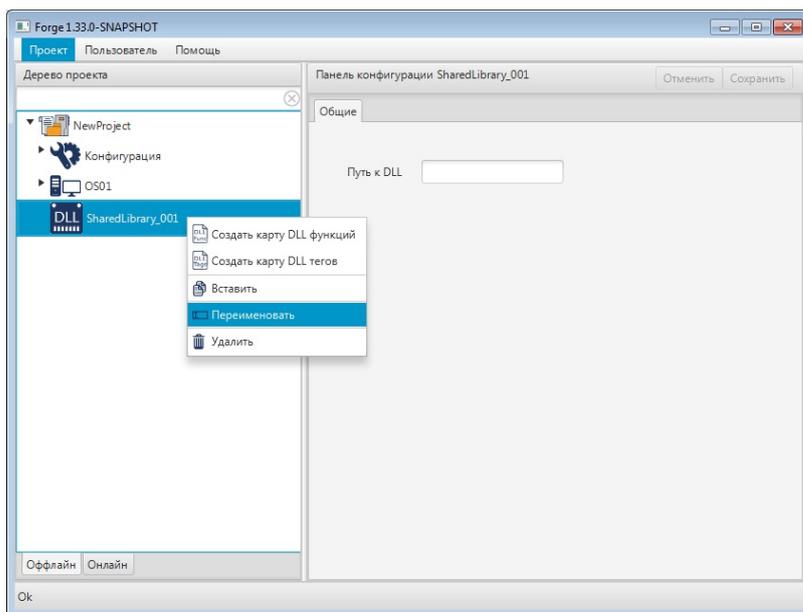


Рисунок 4. Переименование библиотеки DLL

3. В открывшемся диалоговом окне введите новое имя библиотеки DLL и нажмите **ОК**:

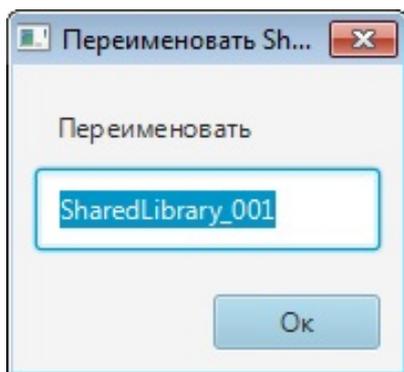


Рисунок 5. Новое имя библиотеки DLL

В дереве проекта библиотека DLL будет отображена с новым именем.

### 3.3. Удаление библиотеки DLL из проекта

Для того чтобы удалить библиотеку DLL из проекта, необходимо выполнить следующие действия:

1. В дереве проекта правой кнопкой мыши выберите имя нужной библиотеки DLL.
2. В открывшемся контекстном меню выберите **Удалить**.
3. В открывшемся диалоговом окне выберите **ОК**.  
Библиотека DLL будет удалена из дерева проекта.

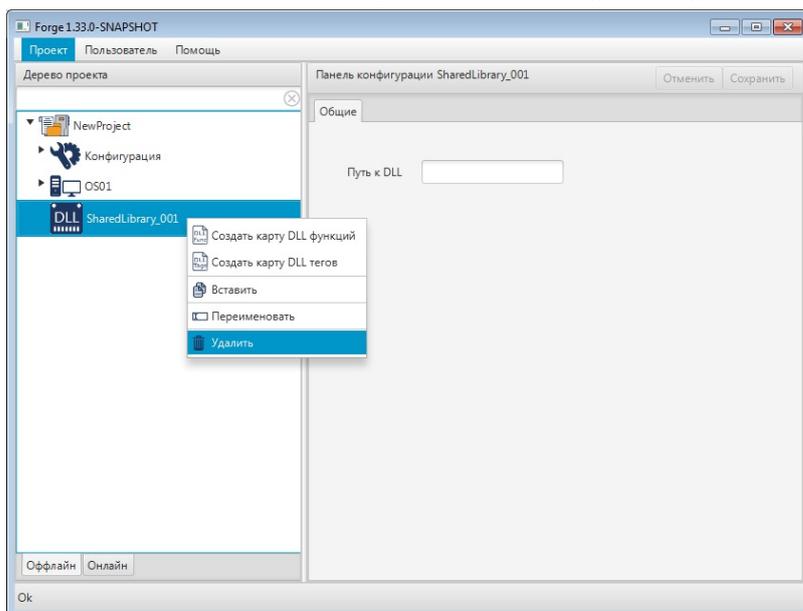


Рисунок 6. Удаление библиотеки DLL

## 4. Операции с картой DLL функций

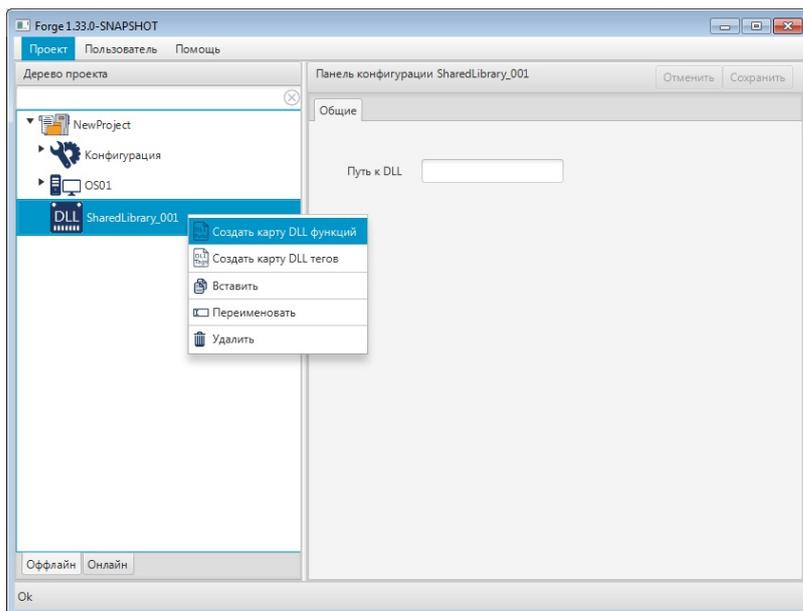
Работа с картой DLL функций в проекте предусматривает следующие операции:

- создание;
- переименование;
- редактирование;
- удаление.

### 4.1. Создание карты DLL функций

Для того чтобы создать карту DLL функций в проекте, необходимо выполнить следующие действия:

1. В дереве проекта правой кнопкой мыши выберите библиотеку DLL, в которой необходимо создать карту DLL функций.
2. В открывшемся контекстном меню выберите **Создать карту DLL функций**.



**Рисунок 7. Создание карты DLL функций**

Новая карта DLL функций появится в дереве проекта во вкладке выбранной библиотеки DLL:

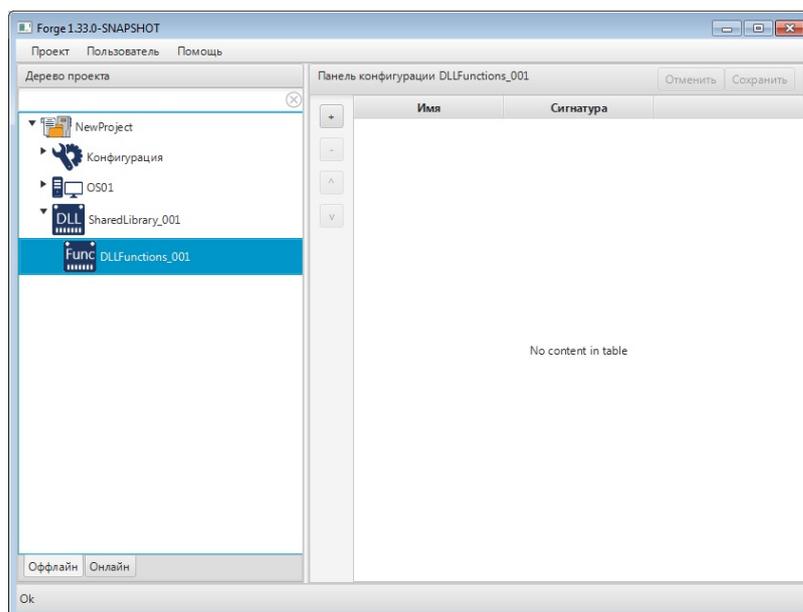


Рисунок 8. Карта DLL функций в дереве проекта

## 4.2. Переименование карты DLL функций

Операция по переименованию карты DLL функций аналогична операции по переименованию библиотеки DLL, описанной в п. 3.2 Руководства.

## 4.3. Удаление карты DLL функций

Операция удаления карты DLL функций аналогична операции удаления библиотеки DLL, описанной в п. 3.3 Руководства.

## 5. Операции с картой DLL тегов

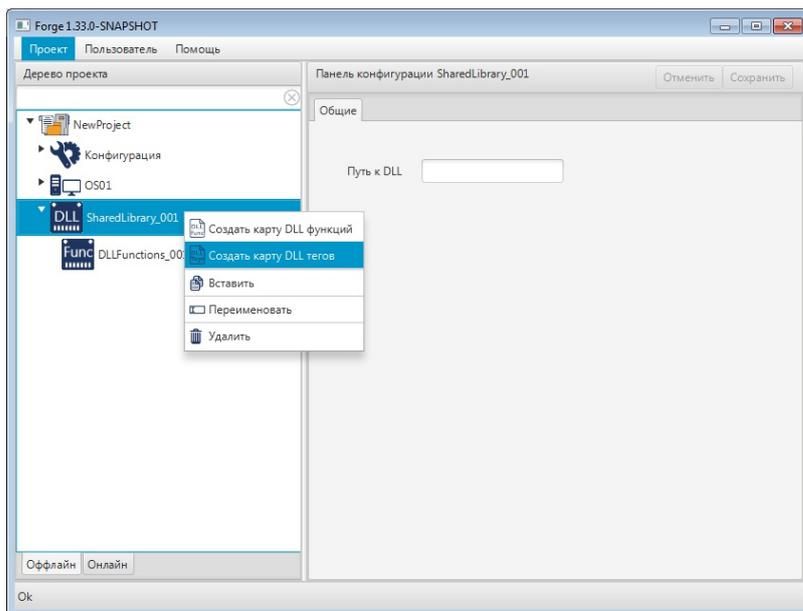
Работа с картой DLL тегов в проекте предусматривает следующие операции:

- создание;
- переименование;
- редактирование;
- удаление.

### 5.1. Создание карты DLL тегов

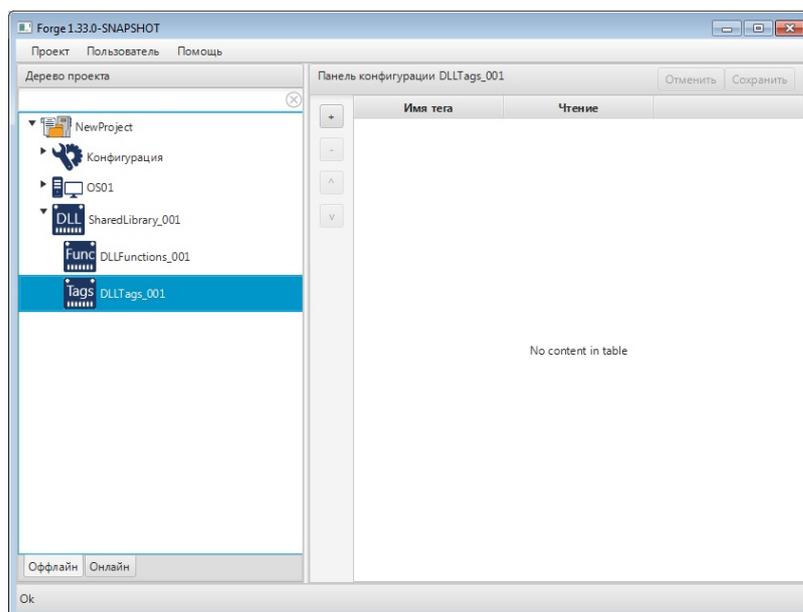
Для того чтобы создать карту DLL тегов в проекте, необходимо выполнить следующие действия:

1. В дереве проекта правой кнопкой мыши выберите библиотеку DLL, в которой необходимо создать карту DLL тегов.
2. В открывшемся контекстном меню выберите **Создать карту DLL тегов**.



**Рисунок 9. Создание карты DLL тегов**

Новая карта DLL тегов появится в дереве проекта во вкладке выбранной библиотеке DLL:



**Рисунок 10. Карта DLL тегов в дереве проекта**

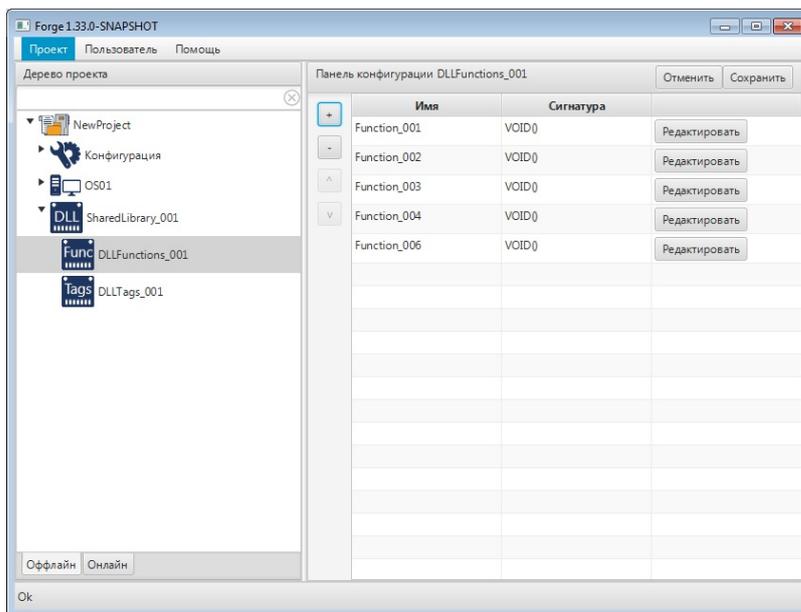
## 5.2. Переименование карты DLL тегов

Операция по переименованию карты DLL тегов аналогична операции по переименованию библиотеки DLL, описанной в п. 3.2 Руководства.

## 5.3. Удаление карты DLL тегов

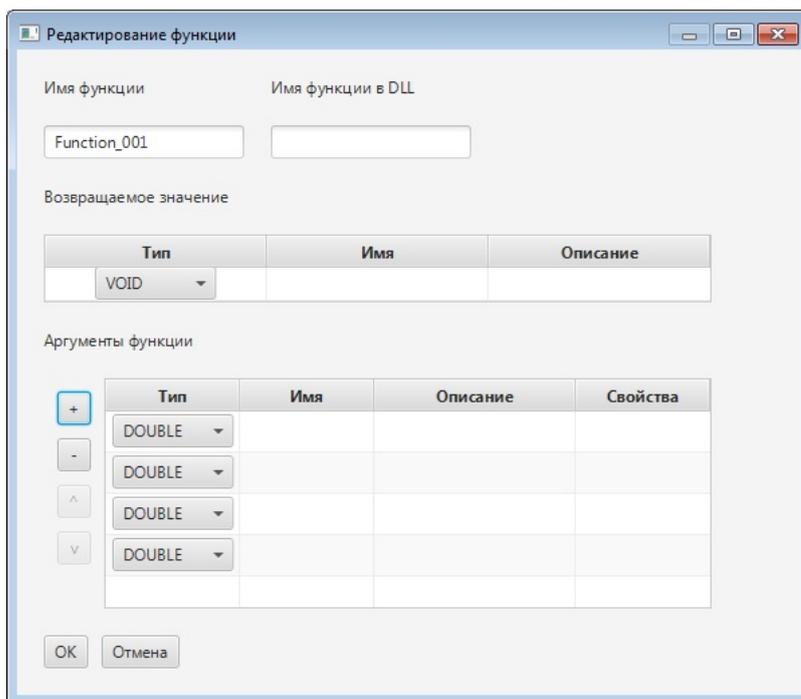
Операция удаления карты DLL тегов аналогична операции удаления библиотеки DLL, описанной в п. 3.3 Руководства.





**Рисунок 13. Добавление функции**

3. Заполните данные добавленной строки в окне редактирования:



**Рисунок 14. Редактирование функции**

- **Имя функции** - имя функции в системе NaftaVision;
- **Имя функции в DLL** - имя функции в файле DLL;
- **Возвращаемое значение:**
  - **Тип** - тип возвращаемого значения. Поддерживаемые типы: VOID, BOOL, UINT8, UINT16, UINT32, UINT64, INT8, INT16, INT32, INT64, DOUBLE;
  - **Имя** - имя возвращаемого значения в модели OPC;

- **Описание** - описание возвращаемого значения в модели OPC.
- **Аргументы функции** - добавьте строку в таблицу аргументов функции при помощи кнопки :
- **Тип** - тип аргумента функции. Поддерживаемые типы: BOOL, UINT8, UINT16, UINT32, UINT64, INT8, INT16, INT32, INT64, DOUBLE, STRING;
- **Имя** - имя аргумента функции в модели OPC;
- **Описание** - описание аргумента функции в модели OPC;
- **Свойства** - описание свойств аргумента функции. Свойство используется только для данных типа STRING: buffer(SIZE), где SIZE - положительное целое число, измеряется в байтах; задаёт использование данного аргумента функции как строкового буфера, куда вызываемая функция будет записывать данные. Для этого введите свойство в таблицу.

#### 4. Добавьте и заполните строки для остальных функций.

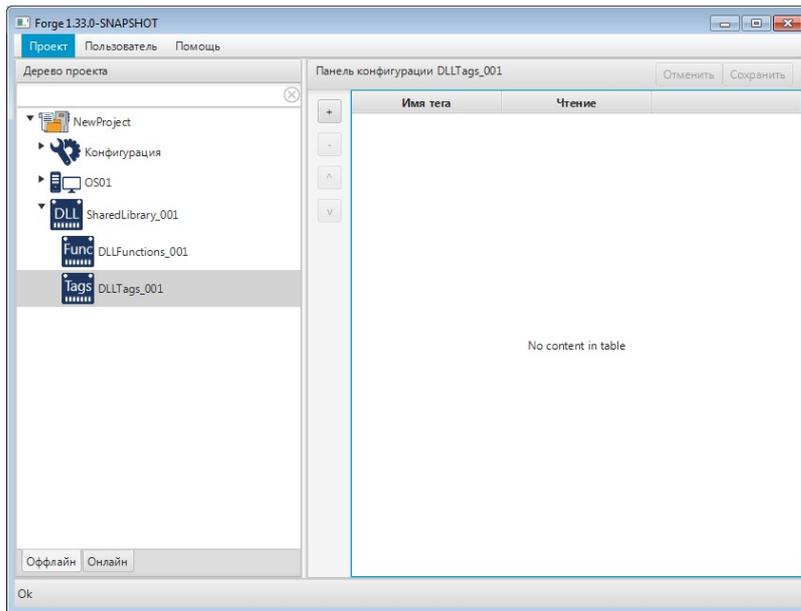
**Примечание:** строки добавляются/удаляются при помощи кнопок  / , расположенных слева от таблицы. Для того чтобы удалить строку, выделите ее левой кнопкой мыши.

Результат настройки карты DLL функций будет сохранен в панели конфигурации.

## 6.2. Настройка карты DLL тегов

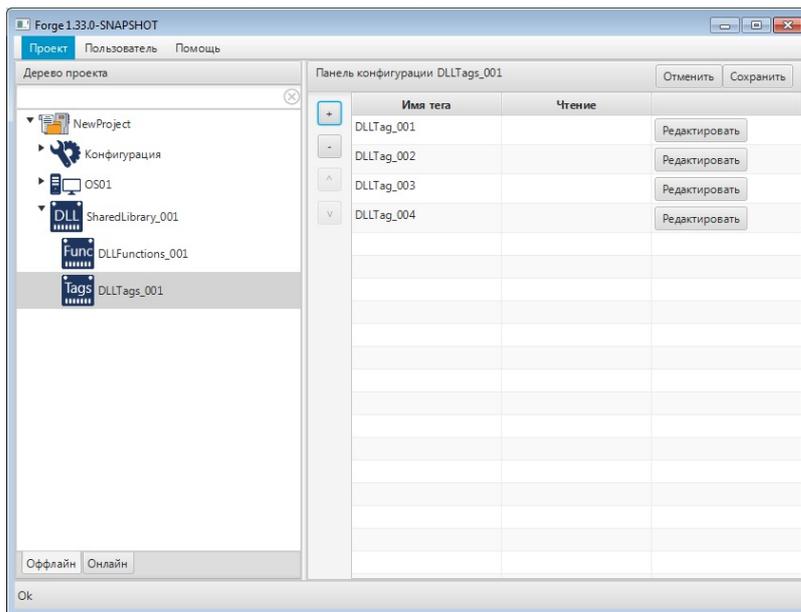
Карта DLL тегов позволяет связать теги сервера ввода-вывода с функциями получения их значений из DLL. Для того чтобы настроить карту DLL тегов, выполните следующие действия:

1. Выберите карту DLL тегов левой кнопкой мыши в дереве проекта. Отобразится панель конфигурации карты DLL тегов:



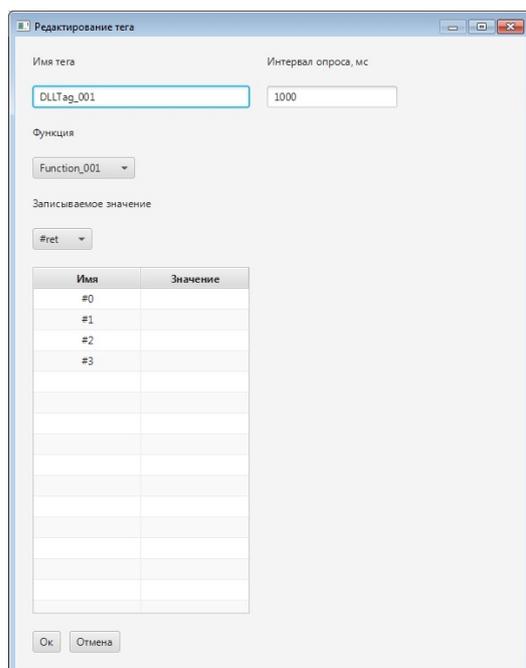
**Рисунок 15. Панель конфигурации карты DLL тегов**

2. Добавьте строку в таблицу тегов при помощи кнопки :



**Рисунок 16. Добавление тега**

3. Заполните данные добавленной строки в окне редактирования:



**Рисунок 17. Редактирование тега**

- **Имя тега** - наименование тега;
- **Интервал опроса** - интервал между двумя последовательными запросами значения данного тега с помощью связанной функции из DLL, измеряется в миллисекундах (мс);
- **Функция** - функция, с помощью которой будет запрашиваться значение из DLL. Выберите функцию из выпадающего списка;
- **Записываемое значение** - выберите значение из выпадающего списка <sup>1</sup>;
- Таблица аргументов:
  - **Имя** - заполняется автоматически при выборе функции;
  - **Значение** - в поле введите соответствующее значение;
- левой кнопкой мыши кликните по кнопке **ОК** для сохранения настроек.

**4.** Добавьте и заполните строки для остальных тегов.

**Примечание:** строки добавляются/удаляются при помощи кнопок  / , расположенных слева от таблицы. Для того чтобы удалить строку, выделите ее левой кнопкой мыши.

Результат настройки карты DLL тегов будет сохранен в панели конфигурации.

<sup>1</sup> Если из функции есть несколько возвращаемых значений (например, один или несколько аргументов имеют свойство `buffer(..)`), выбирается, какое из них будет использоваться в качестве значения тега.