

Пример: ПЛК210 (Modbus RTU Slave) + Owen OPC Server

В качестве примера будет рассмотрена настройка обмена с *Owen OPC Server*, который будет использоваться в режиме **Modbus RTU Master**.

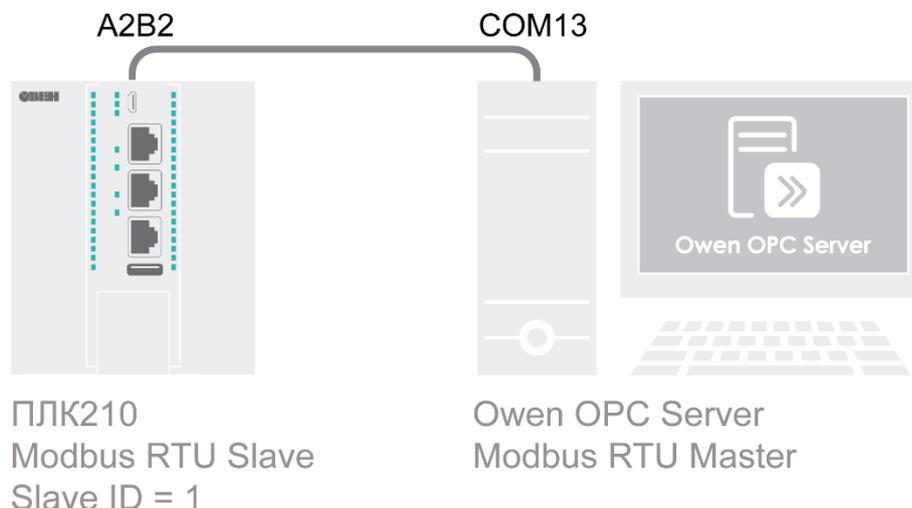


Рисунок 1 – Структурная схема примера

Таблица 1 – Сетевые параметры устройств в примере

| Параметр | ПЛК210 | Owen OPC Server |
|-----------------------|--------|-----------------|
| Режим работы | Slave | Master |
| COM-порт | A2B2 | COM13 |
| Slave ID | 1 | - |
| Скорость обмена | 115200 | |
| Количество бит данных | 8 | |
| Контроль четности | Нет | |
| Количество стоп-бит | 1 | |

Таблица 2 – Регистры/флаги ПЛК в примере

| Адрес регистра/флага | Тип в устройстве | Область памяти |
|----------------------|------------------|-------------------|
| 0 | WORD | Holding Registers |
| 0 | BOOL | Coils |
| 1,2 | REAL | Holding Registers |

Для настройки обмена следует:

1. Подключить контроллер и ПК в соответствии с [рис. 1](#).
2. Создать новый проект Полигон (в примере с именем *MB_RTU_slave_Owen OPC_server*). Добавить в проект библиотеку *paModbus*.
3. Добавить в место работы **Фон** программу с именем *Modbus_RTU_Slave*.
4. Внутри программы добавить 2 **Страницы**, в свойстве **Комментарии** которых указать, соответственно, *Modbus RTU Slave* и *Регистры Modbus*.

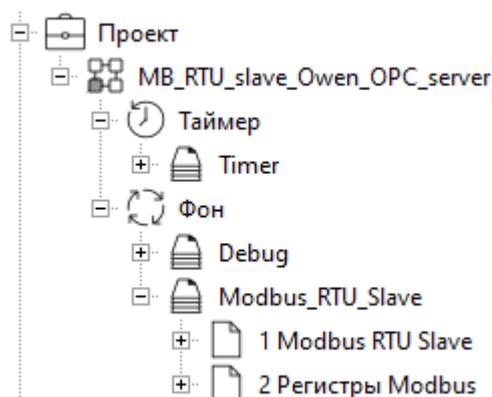


Рисунок 2 – Дерево проекта

5. Внутри страницы *Modbus RTU Slave* создать блок **210-RS485** из библиотеки *paOwenIO*. Во входы блока прописать значения в соответствии с [табл. 1](#).

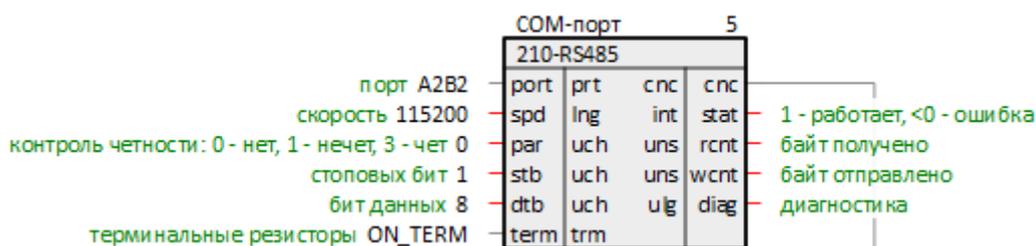


Рисунок 3 – Настройка блока COM-порта

6. Далее создать блок *Modbus RTU Slave*. Соединить вход *cnc* с соответствующим выходом блока **210-RS485**.

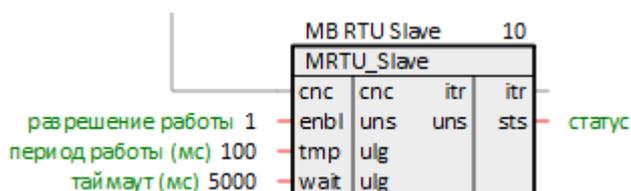


Рисунок 4 – Настройка блока Modbus RTU Slave

7. Затем создать блоки записи регистров в ПЛК (в соответствии с [табл. 2](#)). Соединить входы блоков *itr* с соответствующим выходом блока *Modbus RTU Slave*.

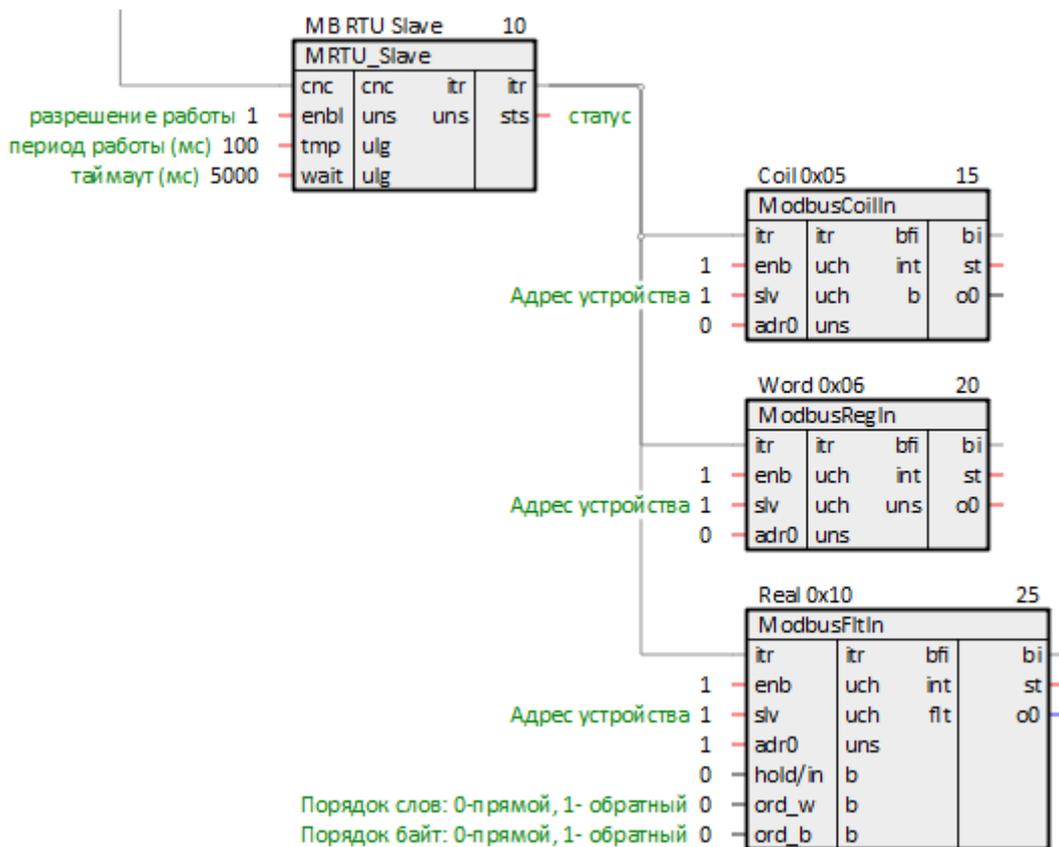


Рисунок 5 – Настройка блоков записи

8. Создать блоки чтения регистров из ПЛК (в соответствии с [табл. 2](#)). Создать три входа **bo** у блока **Modbus RTU Slave**. Соединить их с соответствующими выходами блоков чтения.

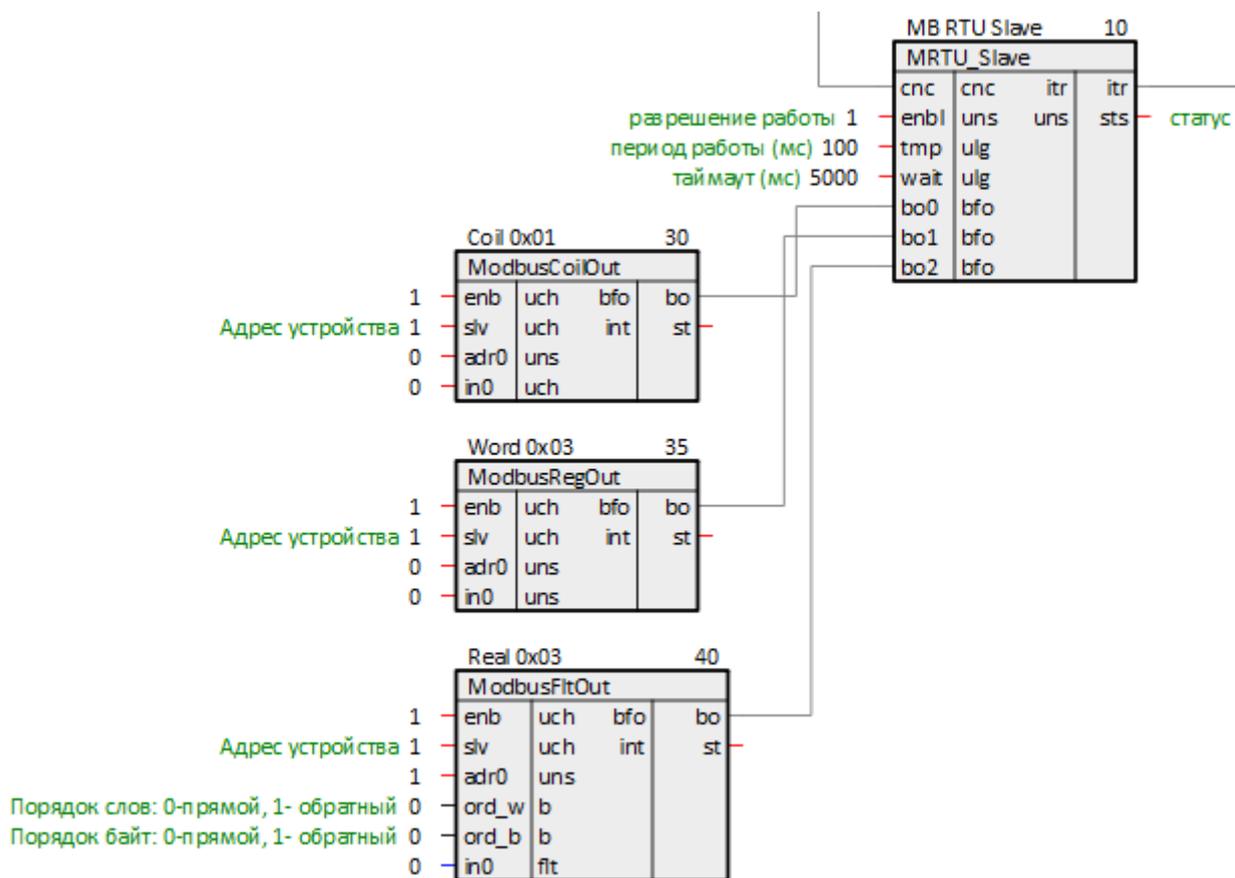


Рисунок 6 – Настройка блоков чтения

9. Для того чтобы одновременно читать и записывать одни и те же значения мастером сети следует соединить выходы блоков записи **o** со входами блоков чтения **in**. Для удобства в примере используются скрытые связи.

Для создания скрытой связи следует в свойствах выхода **o** добавить свойства **Полный алиас** и **Глобальная константа**. В свойстве **Полный алиас** задать имя новой константы.

Повторить те же действия для всех блоков записи на странице.

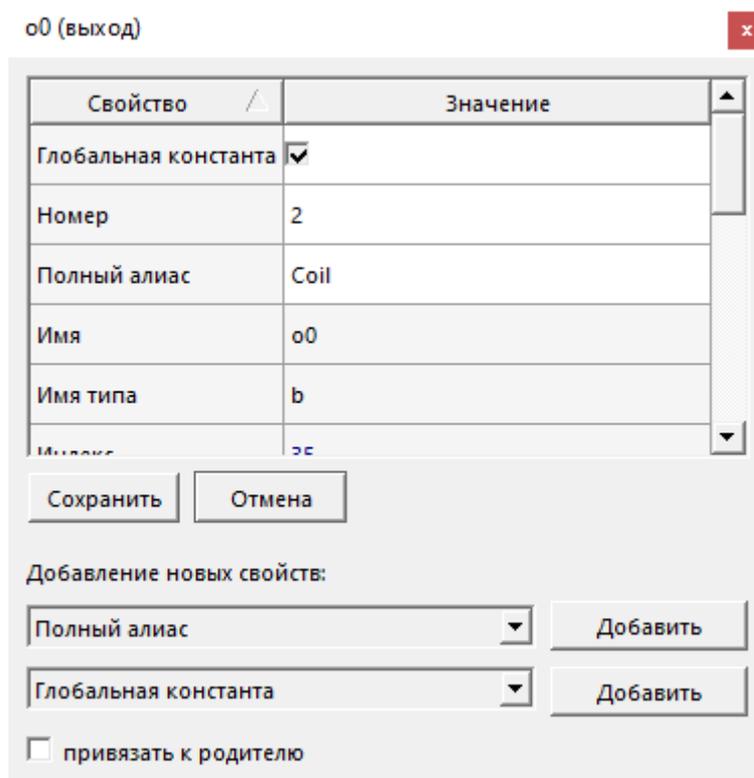


Рисунок 7 – Создание скрытой связи

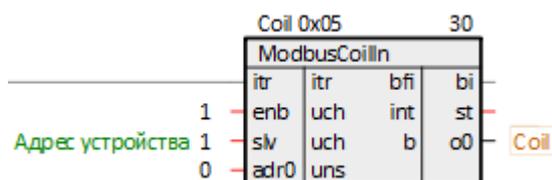


Рисунок 8 – Блок записи с константным выходом

10. У соответствующих входов блоков чтения **in** правой кнопкой мыши задать созданные глобальные константы.

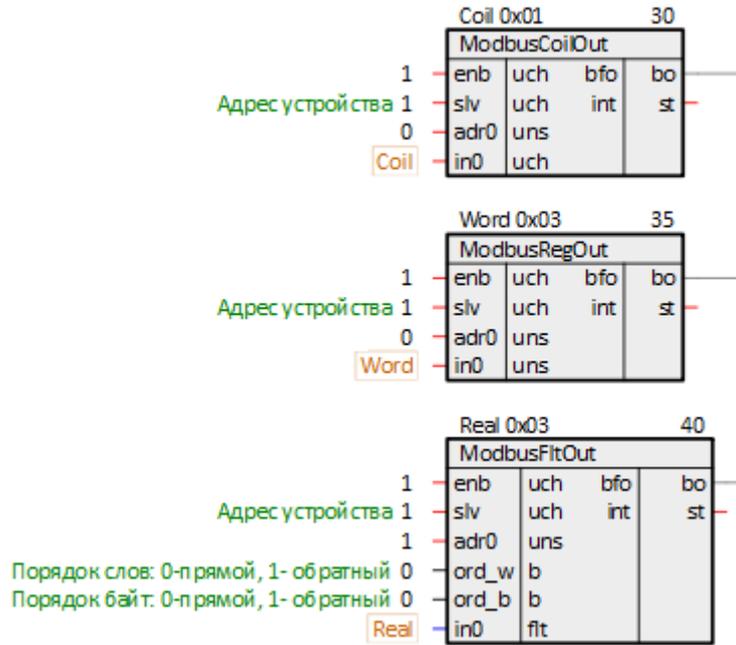


Рисунок 9 – Создание скрытой связи

11. Поставить на странице порядки *По потоку данных*.

Таким образом, вид страницы *Modbus RTU Slave*:

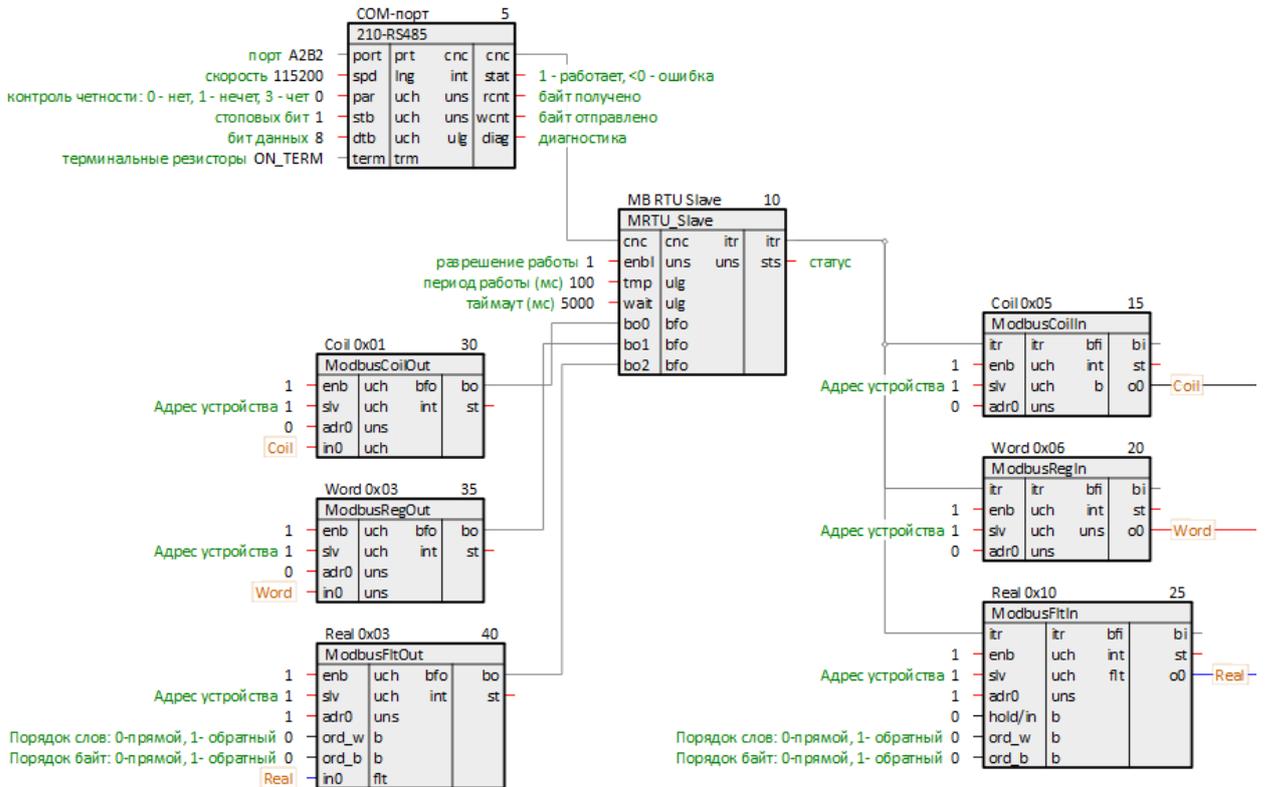


Рисунок 10 – Вид страницы Modbus TCP Server

12. Для удобства можно вынести значения созданных регистров ПЛК на отдельную страницу *Регистры Modbus*.

Для этого следует на странице *Регистры Modbus* создать блоки **TransBit**, **TransInt** и **TransFlt** из библиотеки **paCore**.

На входы созданных блоков **I** задать созданные ранее константы. Выходы блоков **O** при необходимости соединить с другими блоками в проекте.

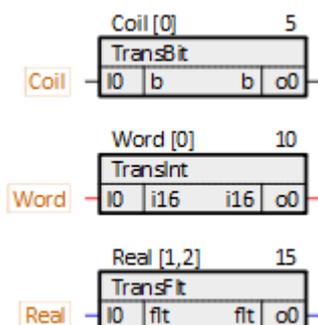


Рисунок 11 – Терминальные блоки

13. Установить и запустить *Owen OPC Server*.

14. Правой кнопкой мыши по компоненту **Сервер** добавить узел.

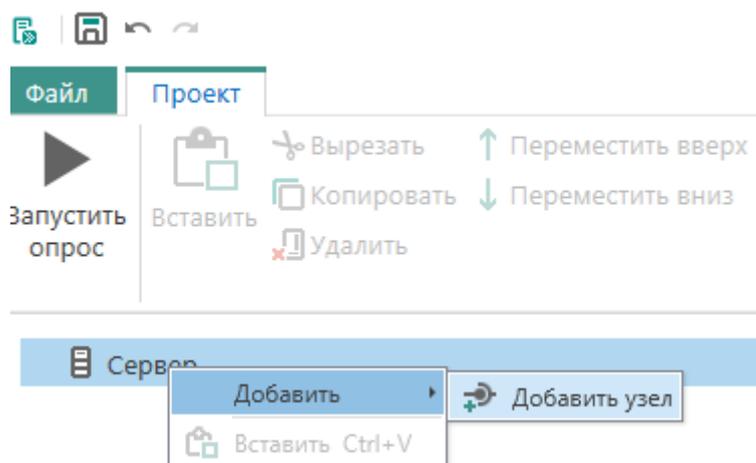


Рисунок 12 – Добавление узла

15. В свойствах добавленного узла задать протокол **Modbus RTU** и настройки интерфейса в соответствии с [табл. 1](#).

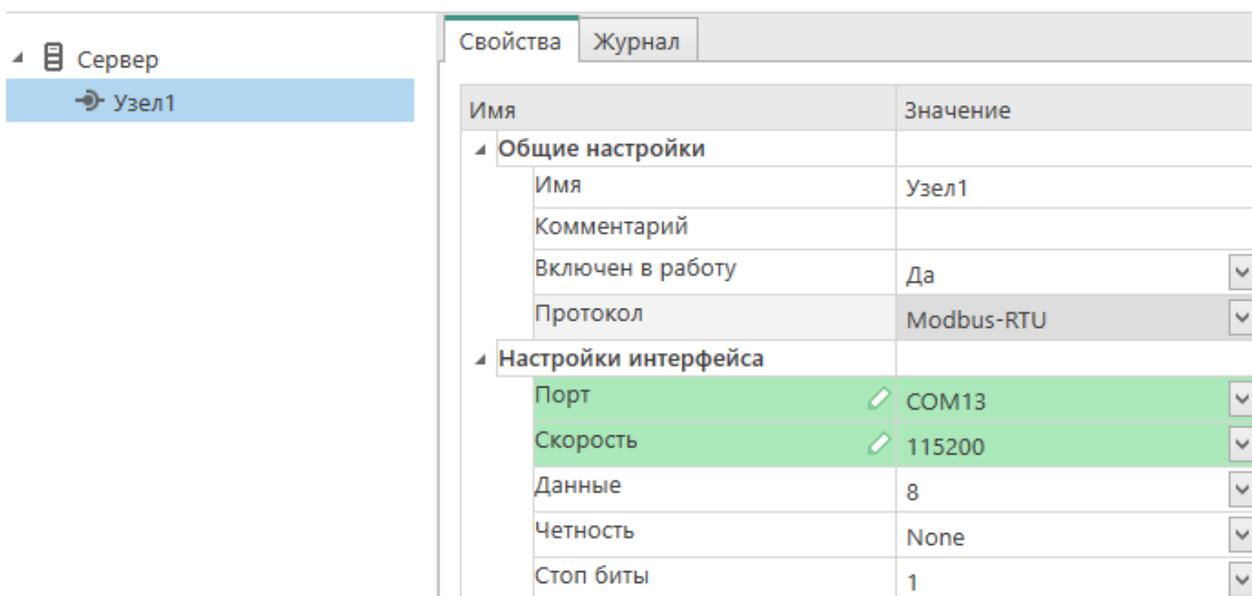


Рисунок 13 – Свойства узла

16. Добавить в узел *Устройство*.

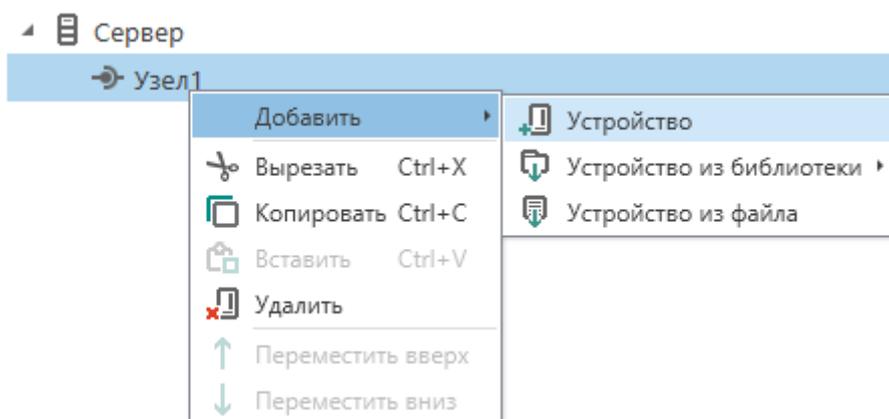


Рисунок 14 – Добавление устройства

17. Задать в устройстве свойства в соответствии с [табл. 1](#).

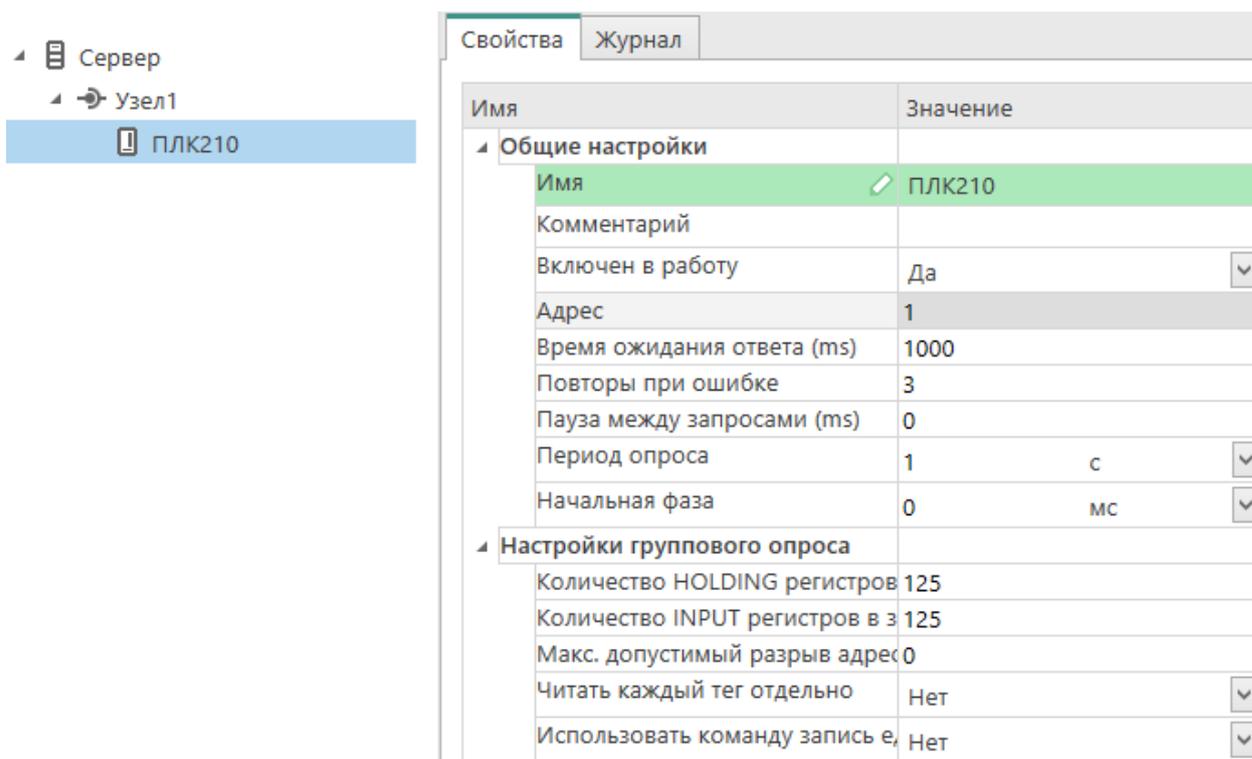


Рисунок 15 – Свойства устройства

18. Добавить в устройстве три **Тега**.

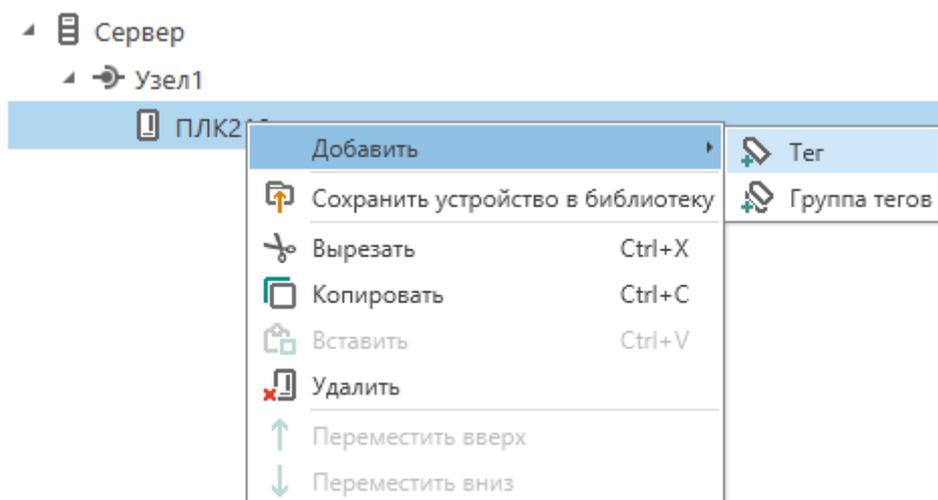


Рисунок 16 – Добавление тега

19. Задать созданным тегам свойства в соответствии с [табл. 2](#).

| Имя | Значение |
|---------------------------------|---------------|
| Общие настройки | |
| Имя | Coil |
| Комментарий | |
| Включен в работу | Да |
| Тип доступа | Чтение/Запись |
| Разовое чтение | Нет |
| Тип данных | Boolean |
| Индивидуальные настройки команд | Да |
| Настройки адресации | |
| Регион | Coils |
| Функция чтения | 0x01 |
| Функция записи | 0x05 |
| Адрес | 0 |
| Дополнительные параметры | |

Рисунок 17 – Ter Coil

| Имя | Значение |
|---------------------------------|-------------------|
| Общие настройки | |
| Имя | Word |
| Комментарий | |
| Включен в работу | Да |
| Тип доступа | Чтение/Запись |
| Разовое чтение | Нет |
| Тип данных | Word |
| Индивидуальные настройки команд | Да |
| Настройки адресации | |
| Регион | Holding Registers |
| Функция чтения | 0x03 |
| Функция записи | 0x06 |
| Адрес | 0 |
| Младшим байтом вперед | Нет |
| Дополнительные параметры | |

Рисунок 18 – Ter Word

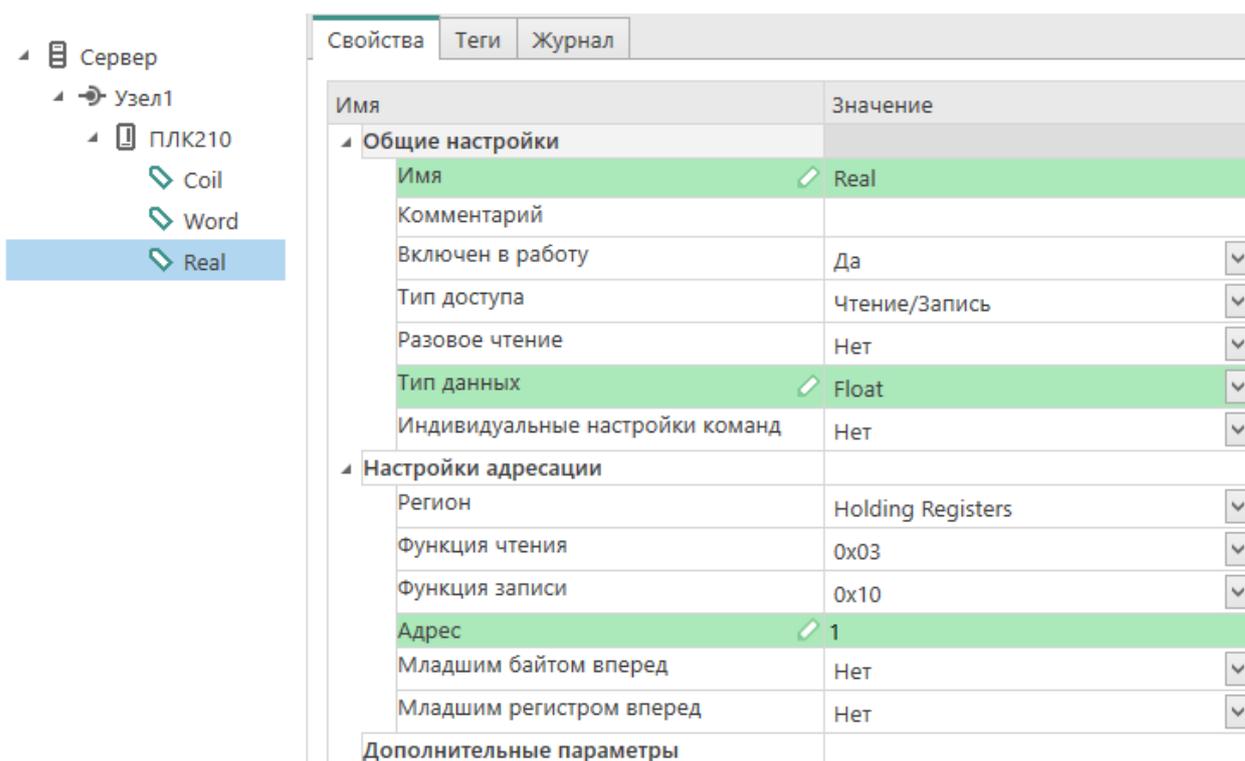


Рисунок 19 – Ter Real

20. Загрузить и запустить программу в ПЛК, подключиться отладчиком. Запустить опрос Owen OPC Server.

21. Наблюдать корректный обмен данными.

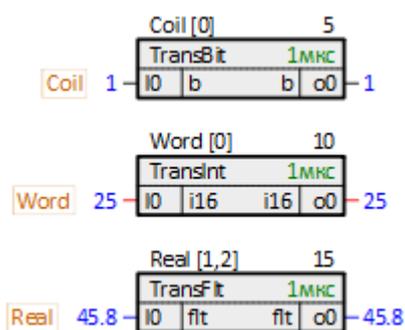


Рисунок 20 – Работа программы ПЛК

| Теги | | Устройства | | | |
|---------------|-----------------------|------------|------------|----------|-------------|
| Имя | Адрес | Значение | Тип данных | Качество | Комментарий |
| ПЛК210.Coil | Coils [0] | True | Boolean | GOOD | |
| ПЛК210.Word | Holding Registers [0] | 25 | Word | GOOD | |
| ▶ ПЛК210.Real | Holding Registers [1] | 45,8 | Float | GOOD | |

Рисунок 21 – Работа Owen OPC Server