В.Д. Ролдугин

Национальный исследовательский ядерный университет «МИФИ», Москва

РАЗРАБОТКА МОДУЛЯ ОБРАБОТКИ ГРУППОВЫХ ЗАПРОСОВ MODBUS В СРЕДЕ OWEN LOGIC

Доклад посвящен вопросам повышения эффективности обмена данными в системах промышленной автоматизации. Рассмотрены архитектура и алгоритмы модуля групповой обработки запросов для протокола Modbus, интегрируемого в среду разработки Owen Logic. Приведены результаты тестирования, подтверждающие повышение производительности и надежности взаимодействия с устройствами.

**Введение**

Актуальность темы обусловлена растущими требованиями к скорости и надежности обмена данными в распределенных системах управления технологическими процессами (АСУ ТП). Протокол Modbus остается одним из наиболее распространенных стандартов в промышленной автоматизации. Однако его базовая реализация, особенно в средах конфигурирования ПЛК, зачастую не использует потенциал групповых операций чтения и записи, что приводит к избыточному сетевому трафику и увеличению времени отклика системы.

Среда программирования Owen Logic, предназначенная для контроллеров компании «ОВЕН», предоставляет средства для создания алгоритмов управления, но обладает ограниченными возможностями по оптимизации Modbus-обмена. Разработка модуля групповой обработки запросов направлена на устранение этого недостатка и актуальна по следующим причинам:

– необходимость снижения нагрузки на каналы связи в реальном времени;

– требование повышения общей производительности системы за счет сокращения количества транзакций;

– потребность в гарантированной целостности данных при групповой записи связанных параметров.

Архитектура модуля групповых запросов

Принципы группировки переменных

Модуль предназначен для автоматического объединения одиночных запросов к переменным Modbus в групповые на основе анализа их атрибутов. Для корректной группировки должны быть выполнены все условия, приведенные в таблице 1.

Таблица 1. Условия формирования группового запроса

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Условие формирования группового запроса** | **Пример валидного условия** | **Пример невалидного условия** |
| 1 | Группировка только для переменных одного устройства | устройство адрес 16, var1, регистр 0; var2, регистр 2 | разные адреса устройств (16 и 17) |
| 2 | Тип данных переменных должен быть одинаковым | var1:real (0), var2:real (2), var3:real (4) | var1:real, var2:int16 |
| 3 | Адреса регистров/битов должны идти подряд без разрывов | var1:рег.0, var2:рег.2, var3:рег.4 | var1:рег.0, var2:рег.3 |
| 4 | Функции чтения/записи должны совпадать | все var1,var2: чтение 0x03 | var1: чтение 0x03, var2: чтение 0x04 |
| 5 | Условия опроса (период, команды) должны совпадать | var1,var2: период 100мс | var1: 100мс, var2: 200мс |
| 6 | Для опроса по команде «Запуск чтение» – одна командная переменная | var1,var2: чтение по команде var\_b1 | var1: var\_b1, var2: var\_b2 |
| 7 | Для опроса по команде «Запуск записи» – одна командная переменная | var1,var2: запись по команде var\_b1 | var1: var\_b1, var2: var\_b2 |
| 8 | Переменные статуса должны быть одинаковы или не заданы | var1,var2: статус var\_i1 | var1: статус var\_i1, var2: не выбран |

Учет ограничений протокола Modbus

Важным аспектом является соблюдение ограничений протоколов Modbus RTU и ASCII на максимальный размер кадра. Для устройств КС1 размер буфера составляет 256 байт. Модуль динамически рассчитывает максимально допустимое количество регистров в одном запросе в зависимости от выбранного протокола и функции.

Таблица 2. Ограничения на количество регистров в запросе

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№ п/п** | **Тип протокола** | **Номер функции** | **макс. кол-во регистров** | **Примечание** |
| 1 | RTU | 0x03, 0x04 | 125 | Ограничение на ввод параметра в UI - 125. |
| 2 | RTU | 0x10 | 123 | При пользовательской настройке в 125 регистров модуль формирует 2 посылки: на 123 и на 2 регистра. |
| 3 | RTU | 0x01, 0x02 | 125 | Максимальное количество бит/койлов. В UI ограничено 125. |
| 4 | RTU | 0x0F | 125 | Максимальное количество бит/койлов. В UI ограничено 125. |
| 5 | ASCII | 0x03, 0x04 | 61 | При пользовательской настройке в 125 регистров модуль формирует 3 посылки: две по 61 и одну на 3 регистра. |
| 6 | ASCII | 0x10 | 59 | При пользовательской настройке в 125 регистров модуль формирует 3 посылки: две по 59 и одну на 7 регистров. |
| 7 | ASCII | 0x01, 0x02 | 125 | Максимальное количество бит/койлов. В UI ограничено 125. |
| 8 | ASCII | 0x0F | 125 | Максимальное количество бит/койлов. В UI ограничено 125. |

При превышении лимита модуль автоматически разбивает группу переменных на несколько физических запросов (MB\_Request).

Реализация алгоритмов и пользовательский интерфейс

Алгоритм группировки

Основная логика реализована в классе `ModbusRequestGroupBuilder`. Алгоритм последовательно обходит отсортированный список переменных и формирует группы, проверяя условия совместимости и ограничения протокола. Ключевые проверки включают `CanAddVariable()`, которая оценивает:

– превышение лимита регистров;

– наличие разрыва в адресах;

– изменение функции Modbus;

– совместимость типов данных.

. Интеграция с Owen Logic

Модуль тесно интегрирован с существующей архитектурой Owen Logic через сервис `CreateModbusRequestService` и взаимодействует с глобальным словарем переменных (`GlobalVariableDictionary`). Это обеспечивает согласованность данных и позволяет использовать модуль в рамках стандартного жизненного цикла проекта.

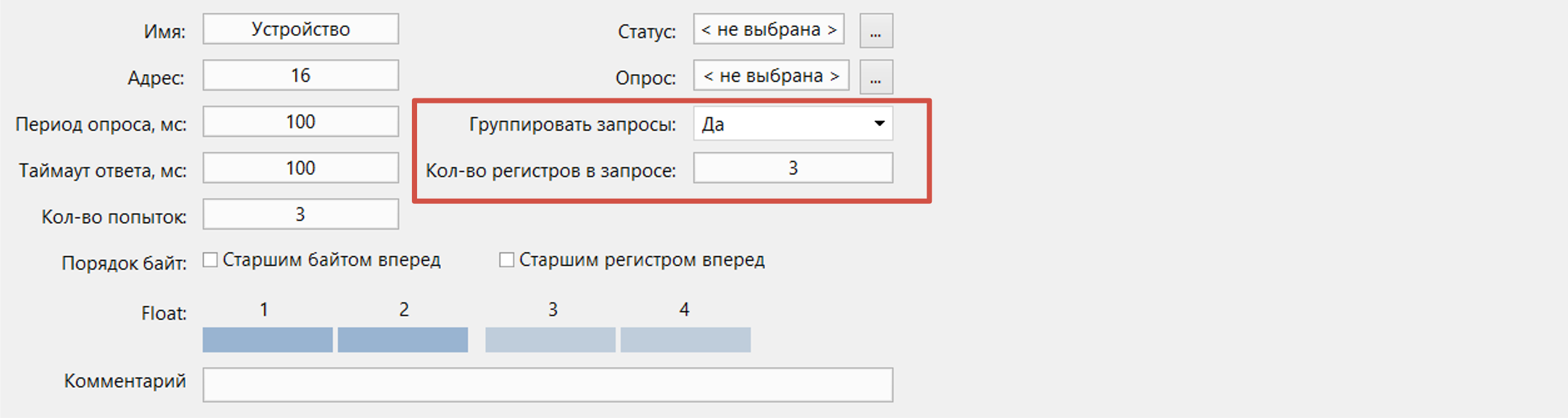
\*\*2.3. Пользовательский интерфейс\*\*

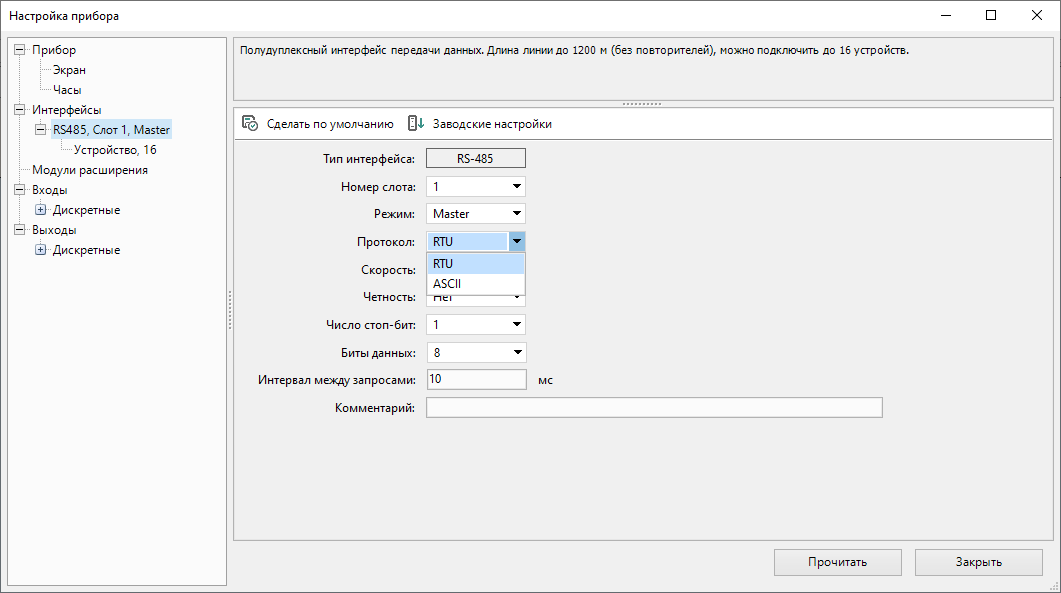
Для настройки модуля в интерфейс окна конфигурации устройства добавлены элементы управления:

– \*\*Выбор протокола\*\* (Modbus RTU / Modbus ASCII).

– \*\*Переключатель «Группировать запросы»\*\* (Да/Нет).

– \*\*Поле «Количество регистров в запросе»\*\* с валидацией вводимого значения в зависимости от протокола.

****



\*Рис. 2. Окно настройки групповых запросов в Owen Logic\*

Валидация в реальном времени предотвращает ввод некорректных данных и информирует пользователя о допустимых диапазонах.

\*\*3. Тестирование и результаты\*\*

Для верификации корректности работы модуля разработан комплекс тестов:

– \*\*Юнит-тесты\*\* для алгоритмов группировки и валидации.

– \*\*Интеграционные тесты\*\*, проверяющие взаимодействие модуля с ядром Owen Logic.

– \*\*Функциональные тесты\*\* на реальном оборудовании КС1.

Результаты тестирования подтвердили:

– Сокращение количества Modbus-запросов на 60-80% для типовых конфигураций с большим количеством переменных.

– Корректную обработку граничных случаев (разрывы в адресах, разные типы данных).

– Соблюдение ограничений протоколов RTU и ASCII.

– Стабильную работу при длительной эксплуатации.

\*\*Выводы\*\*

Разработанный модуль групповой обработки запросов Modbus решает актуальную задачу оптимизации обмена данными в системах на базе контроллеров «ОВЕН». Реализованный функционал позволяет:

1. Существенно повысить производительность за счет сокращения сетевого трафика.

2. Обеспечить целостность данных при групповых операциях записи.

3. Предоставить пользователю интуитивно понятный интерфейс для настройки без необходимости глубокого погружения в спецификации Modbus.

Внедрение модуля в среду Owen Logic расширяет ее возможности, делая ее более конкурентоспособным решением для построения сложных и высоконагруженных систем промышленной автоматизации, где требования к времени отклика и надежности являются критичными.

\*\*Список литературы\*\*

1. Официальная документация MAP – Modbus Application Protocol. URL: https://modbus.org/docs/Modbus\_Application\_Protocol\_V1\_1b3.pdf

2. Ролдугин В.Д. Разработка модуля обработки групповых запросов Modbus в среде Owen Logic: Дипломный проект. НИЯУ МИФИ, 2026.

3. Смит С.И. Разработка программного обеспечения с использованием микросервисов. – М.: Диалектика, 2021. – 350 с.

4. Иванов О.П. Основы программирования с OWEN Logic. – М.: Наука, 2020. – 280 с.

5. Документация на программное обеспечение Owen Logic. Компания «ОВЕН», 2024.

---