|  |  |
| --- | --- |
| **Работа** | **Выполнение** |
| 1. Добавить адрес устройства модуля в Sensor\_array. Добавить в ModBus.cpp - Sensor\_array. ЕСТЬ | ЕСТЬ |
| 1. Описание адресов регистров добавить адреса регистров модуля. ModBus.hpp - MB\_Reg\_t; ЕСТЬ | ЕСТЬ |
| 1. Добавить в страницу программирования в TouchGFX тип модуля ввода-вывода. В TouchGFX добавить кол-во items в колесе прокрутки типов датчиков. ЕСТЬ | ЕСТЬ |
| 1. Добиться чтение-запись в/из модуля в режиме программирования. ЕСТЬ | ЕСТЬ |
| 1. Сделать программирование скорости и адреса модуля. | ЕСТЬ |
| 1. Добавить в начальное сканирование датчиков модуль ввода-вывода | ЕСТЬ |
| 1. Переделать функцию Master\_Request() для передачи датчику заданного количества байт | ЕСТЬ |
| 1. Переделать функцию Master\_RW() для обработки команды записи в несколько регистров массива данных | ЕСТЬ |
| 1. Добавить запись в регистр реле модуля в основном цикле. | ЕСТЬ |
| 1. Проверить совместную работу всех устройств на основной шине. | ЕСТЬ |
| 1. Проверить программирование: | ЕСТЬ |
| * 1. Датчик типа 1. | ЕСТЬ |
| * 1. Датчик типа 2. | ЕСТЬ |
| * 1. Датчик типа 4 | ЕСТЬ |
| 1. Проверить коррекцию: | ЕСТЬ |
| * 1. Датчик типа 1. | ЕСТЬ |
| * 1. Датчик типа 2. | ЕСТЬ |
| 1. Проверить считывание входных данных модуля ввода-вывода | ЕСТЬ |
| 1. Добавить запись регистра управления периферийными устройствами в класс Sensor. Data.hpp – class Sensor. | ЕСТЬ |
| 1. Записывать считанные данные состояния периферийных устройств в класс Sensor. Data.hpp – class Sensor. | ЕСТЬ |
| 1. Завести регистр аппаратного управления периферийными устройствами. | ЕСТЬ |

Команды управления устройством:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Код функции** | **Что делает функция** | | **Тип значения** | **Тип доступа** |
| 01 (0x01) | Чтение DO | Read Coil Status | Дискретное | Чтение |
| 02 (0x02) | Чтение DI | Read Input Status | Дискретное | Чтение |
| 03 (0x03) | Чтение AO | Read Holding Registers | 16 битное | Чтение |
| 04 (0x04) | Чтение AI | Read Input Registers | 16 битное | Чтение |
| 05 (0x05) | Запись одного DO | Force Single Coil | Дискретное | Запись |
| 06 (0x06) | Запись одного AO | Preset Single Register | 16 битное | Запись |
| 15 (0x0F) | Запись нескольких DO | Force Multiple Coils | Дискретное | Запись |
| 16 (0x10) | Запись нескольких AO | Preset Multiple Registers | 16 битное | Запись |

Расшифровка отправки команд в устройство:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0х01 | 0x02 | 0x03 | 0х05 | 0x06 | 0х0F | 0x10 |
| № | Чтение DO | Чтение  DI | Чтение  AO | Запись  DO | Запись  AO | Запись mult DO | Запись mult AO |
| 1 | Адрес | Адрес | Адрес | Адрес | Адрес | Адрес | Адрес |
| 2 | Код | Код | Код | Код | Код | Код | Код |
| 3 | Рг1 Hi | Рг1 Hi | Рг1 Hi | Рг1 Hi | Рг1 Hi | Рг1 Hi | Рг1 Hi |
| 4 | Рг1 Lo | Рг1 Lo | Рг1 Lo | Рг1 Lo | Рг1 Lo | Рг1 Lo | Рг1 Lo |
| 5 | Кол-во Рг Hi | Кол-во Рг Hi | Кол-во Рг Hi | Значение Hi байта | Значение Hi байта | Кол-во Рг Hi | Кол-во Рг Hi |
| 6 | Кол-во Рг Lo | Кол-во Рг Lo | Кол-во Рг Lo | Значение Lo байта | Значение Lo байта | Кол-во Рг Lo | Кол-во Рг Lo |
| 7 | CRC Lo | CRC Lo | CRC Lo | CRC Lo | CRC Lo | Количество байт далее | Количество байт далее |
| 8 | CRC Hi | CRC Hi | CRC Hi | CRC Hi | CRC Hi | Значение байт DO Lo 27-20 (1100 1101) | Значение Hi 40002 |
| 9 |  |  |  |  |  | Значение байт DO Hi 29-28 (0000 0001) | Значение Lo 40002 |
| 10 |  |  |  |  |  | CRC Lo | Значение Hi 40003 |
| 11 |  |  |  |  |  | CRC Hi | Значение Lo 40003 |
| 12 |  |  |  |  |  |  | CRC Lo |
| 13 |  |  |  |  |  |  | CRC Hi |

Результаты анализа команд:

1. Для всех команд, кроме команд записи mult, одинаковая структура отправки – 8 байт. Есть различие только в назначении данных 5 и 6 байтов.
2. Командам mult требуется информация о количестве байт и сами байты для записи в устройство.
3. Можно создать массив, в котором элемент 0 – это количество байт для отправки, последующие элементы массива – байты для отправки.
4. Подготовку массива для отправки поручить специализированному обработчику перед вызовом команды записи.
5. 0xfc15, 0x15 – 0xFC

|  |
| --- |
| CRC value Lo |
| CRC value Hi |

1. Изображение выглядит как текст, снимок экрана

   Автоматически созданное описание
2. Изображение выглядит как текст, снимок экрана, Шрифт, Мультимедийное программное обеспечение

   Автоматически созданное описание
3. После корректировки выводим 0х5522
4. Изображение выглядит как текст, снимок экрана, Шрифт, Мультимедийное программное обеспечение

   Автоматически созданное описание
5. И считываем назад из регистра вывода:
6. Изображение выглядит как текст, снимок экрана, линия, Мультимедийное программное обеспечение

   Автоматически созданное описание