Новосибирский государственный университет

|  |  |
| --- | --- |
| СОГЛАСОВАНО | УТВЕРЖДАЮ |
| Начальник 628 ВП МО РФ  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ О.Э. Тимаков  «\_\_\_» \_\_\_\_\_\_\_\_\_ 2020 г. | Проректор НГУ по научно- исследовательской деятельности  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Д. В. Чуркин  «\_\_\_» \_\_\_\_\_\_\_\_\_ 2020 г. |
|  | СОГЛАСОВАНО |
|  | Главный конструктор  электрического проектирования и испытаний КА АО «ИСС»  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ С.И. Опенько  «\_\_\_» \_\_\_\_\_\_\_\_\_ 2020 г. |

ОАИ КПА

Описание цифрового модуля

|  |  |
| --- | --- |
| От 628 ВП МО: | От НГУ: |
| Ведущий специалист  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Н.В. Сапожникова  «\_\_\_» \_\_\_\_\_\_\_\_\_ 2020 г. | Инженер  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.С. Козлов  «\_\_\_» \_\_\_\_\_\_\_\_\_ 2020 г. |
|  |  |

2020 г..

Оглавление

[1 Назначение 3](#_Toc58341818)

[2 Технические данные 3](#_Toc58341819)

[3 Регистры управления UART1 9](#_Toc58341820)

[3.1 Регистры передатчика UART1 9](#_Toc58341821)

[3.2 Регистры настроек UART1 11](#_Toc58341822)

[3.3 Регистры приемника UART1 13](#_Toc58341823)

[4 Регистры управления GPIO 14](#_Toc58341824)

[4.1 Регистры настройки GPIO. 14](#_Toc58341825)

[4.2 Регистры управления GPIO настроенных на выход. 16](#_Toc58341826)

[4.3 Регистры чтения GPIO 18](#_Toc58341827)

[5 Регистры SPI 23](#_Toc58341828)

[5.1 SPI Settings. 23](#_Toc58341829)

[5.2 SPI Transmit registers. 26](#_Toc58341830)

[5.3 SPI Receive 28](#_Toc58341831)

# Назначение

Цифровой модуль ОАИ КПА предназначен для использования в КПА изготавливаемых ОАИ НГУ.

# Технические данные

Цифровой модуль является стандартной частью модулей используемых в КПА.

Цифровой модуль обеспечивает:

Взаимодействие функциональной части модуля КПА с УВУ КПА.

На GND\_IN подать GND, На POWER\_IN подать +20-48В.

После подачи питания, должен загореться нижний красный светодиод, сигнализирующий о том, что есть питание по 3.3 В.

Модуль подключается к ПК посредством USB разъема, находящегося в верхней части слева.

Модуль определяется как Virtual Com Port, у каждого модуля есть свой идентификатор который можно узнать при помощи базового ПО.

В модуле реализована система команд по протоколу ModbusRTU, подробно можно ознакомиться здесь <https://ipc2u.ru/articles/prostye-resheniya/modbus-rtu/>

В модуле реализованы команды для чтения и записи аналоговых входов и аналоговых выходов. Дискретные входы/выходы не используются.

Управление модулем и получение данных с модуля осуществляется посредством записи и чтения соответствующих регистров.

Регистр представляет собой 16 битное слово, хранящееся по некому адресу.

Регистры в памяти структурированы карта регистров представлена ниже.

Необходимо помнить что данные в модуле хранятся в формате младший байт впереди. это учтено в базовом ПО верхнего уровня. Необходимо выставить флаг byte order revers.

В модуле представлены следующие регистры:  
Регистры АЦП. Позволяют вычитывать данные АЦП, Запускать и останавливать АЦП.

Регистры ЦАП. Позволяют генерировать сигнал сложной формы, запускать и останавливать ЦАП.

Регистры GPIO. Позволяют конфигурировать GPIO как входы или выходы. Получить текущее состояние GPIO

Регистры UART1, UART2. Позволяют настроить UART1/2 независимо друг от друга (Baudrate, Бит четности и пр). Позволяют производить отправку пакетов длиной до 128 байт. Прием осуществляется в кольцевой буфер размером 2048 байт.

# Структура регистров модуля

Все реализованные функции модуля управляются или опрашиваются при помощи регистров. Регистры находящиеся в области Analog inputs доступны для чтения при помощи функций 0x04 протокола Modbus RTU. Регистры находящиеся в области Analog outputs доступны как для записи, так и для чтения при помощи функций 0x03, 0x06 и 0x10.

Данные функции реализованы в python модуле OAI\_Modbus.

Адреса регистров представленных в таблице 1.

Таблица 1.

|  |  |  |  |
| --- | --- | --- | --- |
| Analog\_inputs | Адресс | Название | Краткое описание |
| 0 | Входы АЦП | Сюда записываются результаты измерений аналоговых входов 1-8. Частота обновления 2Гц |
| 8 | INA226 3V3 | Содержат данные о собственном напряжении и токе потребления модуля (по 3.3 и 5В) |
| 11 | INA226 5V |
| 14 | UART1 данные приемника | Содержат данные которые принимает модуль UART |
| 1042 | UART2 данные приемника |
| 2070 | GPIO in | Содержат информацию о текущем состоянии GPIO |
| 2074 | SPI receive data | Данные приемника SPI |
| Analog outputs | 0 | DAC1 | Регистры настройки ЦАП |
| 528 | DAC2 |
| 1056 | ADC | Регистры настройки АЦП (вкл/выкл) |
| 1059 | GPIO\_config | Настройки GPIO. (Маска вход выход) |
| 1064 | GPIO\_out | Управление GPIO настроенными на выход |
| 1072 | UART1 transmitter | Управление UART1 на передачу |
| 1140 | UART1 settings | Настройки UART1 |
| 1150 | UART2 transmitter | Управление UART2 на передачу |
| 1218 | UART2 settings | Настройки UART2 |
| 1228 | GPIO\_alternative | Управление альтернативным состоянием GPIO настроенных на выход. Позволяет сформировать импульс необходимой длительности |
| 1246 | SPI setings | Настройки SPI |
| 1266 | SPI transmit | Управление SPI на передачу |
| 1308 | SPI receive | Управление SPI на прием |

Регистры данных АЦП.   
Обновляются 1 раз в 500мс. Находятся в области analog\_inputs.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 0 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| N/U | | | | Data[0] | | | | | | | | | | | |
| 0/1 | | | | | | | | | | | | | | | |
| Reg num 7 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| N/U | | | | Data[7] | | | | | | | | | | | |
| 0/1 | | | | | | | | | | | | | | | |

Данные в 12 битном формате. 0 – 0В, 0x0FFF – 2048В.

Data[0] – ADC\_IN1

Data[8] – ADC\_IN8

Регистры управления АЦП. Находятся в области analog\_outputs

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1056 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Scaler | | | | | | | | | | | | | | | |
| 0/1 | | | | | | | | | | | | | | | |

Scaler – регистр запуска обновления команды.  
При записи 1 – обновляется и исполняется команда. После завершения команды регистр перезаписывается 0.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1057 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Start | | | | | | | | | | | | | | | |
| 0/1 | | | | | | | | | | | | | | | |

Start – если 1 то данные АЦП обновляются. Если 0 – то данные не обновляются. По умолчанию выставлена 1.

Данные мониторов питания цифровой части

Находятся в области analog\_inputs

3.3 В

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 8 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Voltage 3,3V | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 9 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Current 3,3V | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | |

5В.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 11 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Current 5V | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 12 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Current 5V | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | |

Немного позже здесь появится формула пересчета.

Регистры управления ЦАП. Здесь будет описание на регистры ЦАП

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 12 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Current 5V | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | |

# Регистры управления UART1

## Регистры передатчика UART1

Регистры передатчика UART1 transmit находятся в области analog outputs.

|  |  |
| --- | --- |
| Addr | Name |
| 1072 | Scaler |
| 1073 | Start |
| 1074 | Len |
| 1075 | Error + Transmit flag |
| 1076 | Data[0]/Data[1] |
| 1139 | Data[126]/Data[127] |

Scaler – регистр запуска обновления команды.  
При записи 1 – обновляется и исполняется команда. После завершения команды регистр перезаписывается 0. Не тоже самое что TransmitFlag

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1072 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Scaler | | | | | | | | | | | | | | | |

Start – если равен 0x01 будет отправлен пакет длиной Len

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1073 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Start | | | | | | | | | | | | | | | |

Len – длина пакета на отправку в байтах. Если превысить 128, ничего не будет отправлено. В регистр Transmit Flag в будет записано 0xFF00;

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1074 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Len | | | | | | | | | | | | | | | |

Transmit\_flag – по завершению отправки будет выставлен в 0x01

Error flag –

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1075 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Error flag | | | | | | | | Transmit flag | | | | | | | |
| 0x00/0xFF | | | | | | | | 0/1 | | | | | | | |

Data[] данные на отправку. Отправляются с нулевого по len.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1076 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Data[0] | | | | | | | | Data[1] | | | | | | | |
| Reg num 1139 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Data[126] | | | | | | | | Data[127] | | | | | | | |

## Регистры настроек UART1

Регистры настроек UART1 settings находятся в области analog outputs

|  |  |
| --- | --- |
| Addr | Name |
| 1140 | Scaler |
| 1141 | Low\_Baud |
| 1142 | High\_Baud |
| 1143 | Uart\_stop\_bits |
| 1144 | Uart\_pairity |
| 1145 | Uart\_word\_lengh |
| 1146 | Set Default |
| 1149 | Init Flag |

Scaler – регистр запуска обновления команды.  
При записи 1 – обновляется и исполняется команда. После завершения команды регистр перезаписывается 0.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1140 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Scaler | | | | | | | | | | | | | | | |
| 0/1 | | | | | | | | | | | | | | | |

BaudRate – 32 битный регистр выставления скорости общения по интерфейсу UART1. Получается записью двух регистров.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1141 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Low\_Baud | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1142 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| High\_Baud | | | | | | | | | | | | | | | |

Стоп бит. 0 – 1 стоп бит. 1 – 2 стоп бита.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1143 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Uart\_stop\_bits | | | | | | | | | | | | | | | |
| 0/1 | | | | | | | | | | | | | | | |

Бит четности. 0 – без проверки. 1 – Проверка на четность. 2 – проверка на нечетность.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1144 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Uart\_pairity | | | | | | | | | | | | | | | |
| 0-2 | | | | | | | | | | | | | | | |

Длина данных. 0 – 8 бит. 1 – 9 бит. 9 битный прием/передача данных на данный момент не реализованы.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1145 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Uart\_word\_lengh | | | | | | | | | | | | | | | |
| 0-2 | | | | | | | | | | | | | | | |

Настройки по умолчанию. Если 0 будут использоваться значения записанные в предыдущие регистры. Если 1 то будут использоваться настройки по умолчанию. 115200, 8 бит,.1 стоп бит. Без проверки четности.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1146 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Set Default | | | | | | | | | | | | | | | |
| 0-2 | | | | | | | | | | | | | | | |

Флаг инициализации. После успешной инициализации Init Flag = 1. Перед инициализацией сбрасывается в 0.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1146 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Init Flag | | | | | | | | | | | | | | | |
| 0/1 | | | | | | | | | | | | | | | |

## Регистры приемника UART1

Регистры UART1 receive находятся в области analog inputs

Прием данных по UART начинается сразу после инициализации микроконтроллера с настройками UART по умолчанию.

|  |  |
| --- | --- |
| Addr | Name |
| 17 | Wr\_ptr |
| 18 | Data[0] | Data[1] |
| 1042 | Data[2046] | Data[2047] |

Указатель записи. Данный указатель в каждый момент времени указывает в какое место будет записан следующий принятый байт. При достижении 2047 автоматически изменяется на 0. Не вычитанные данные перезаписываются.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 17 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Wr\_ptr | | | | | | | | | | | | | | | |
| 0-2047 | | | | | | | | | | | | | | | |

Data[] принятые данные.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 18 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Data[0] | | | | | | | | Data[1] | | | | | | | |
| Reg num 1042 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Data[2046] | | | | | | | | Data[2047] | | | | | | | |

# Регистры управления GPIO

На плате цифрового модуля доступны 60 GPIO. Их можно конфигурировать по своему усмотрению как входы или как выходы. Выставлять значения 0/1 на выходах и считывать значения как со входов, так и текущее состояние выходов.

## Регистры настройки GPIO.

Регистры настройки находятся в области analog outputs.

По умолчанию все GPIO настроены на вход с подтяжкой к 0.

|  |  |
| --- | --- |
| Addr | Name |
| 1059 | Init Flag |
| 1060 | Scaler, Mask GPIO1-12 |
| 1061 | Mask GPIO 13-28 |
| 1062 | Mask GPIO 29-44 |
| 1063 | Mask GPIO 45-60 |

Init Flag. Выставляется в 1 после успешной инициализации. Перед инициализацией сбрасывается принудительно в 0.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1059 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Init Flag | | | | | | | | | | | | | | | |
| 0x01/0x00 | | | | | | | | | | | | | | | |

Scaler – регистр запуска обновления команды.  
При записи 1 – обновляется и исполняется команда. После завершения команды регистр перезаписывается 0.

Mask GPIO. Битовый регистр. Если в маску записано 1 – GPIO конфигурируется как выход. Если в маску записано 0 – то GPIO конфигурируется как вход.

Пример 1:

Чтобы сконфигурировать GPIO 1,3 и 12 на выход, остальные на вход нужно записать в регистр 1060 следующее значение 0b0001101000000001 иначе 0x1A01. После того как будет выполнена команда, можно прочитать регистр 1059. Там будет записано 0x0A01. В регистре Init Flag (1059) будет записано 0x0001.

При необходимости переинициализировать GPIO.

Пример 2:

Для использования GPIO 1,12, 15, 49, 50 как выход нужно записать следующие значения в регистры.  
Reg num 1060 – 0x1801, 1061 – 0x8000, 1063 – 0x0С00. После инициализации в регистре 1059 будет выставлено 0x0001. В регистре 1060 – 0x0801.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1060 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Scaler | | | | Mask GPIO | | | | | | | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| 0/1 | | | | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1061 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Mask GPIO | | | | | | | | | | | | | | | |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 24 | 22 | 23 | 24 | 25 | 26 | 27 | 28 |
| 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1062 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Mask GPIO | | | | | | | | | | | | | | | |
| 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 |
| 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1063 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Mask GPIO | | | | | | | | | | | | | | | |
| 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 |
| 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 |

## Регистры управления GPIO настроенных на выход.

|  |  |
| --- | --- |
| Addr | Name |
| 1064 | Scaler, SET\_RESET GPIO1-12 |
| 1065 | SET\_RESET GPIO 13-28 |
| 1066 | SET\_RESET GPIO 29-44 |
| 1067 | SET\_RESET GPIO 45-60 |

Данные регистры можно использовать только после выполнения процедуры инициализации.

Scaler – регистр запуска обновления команды.  
При записи 1 – обновляется и исполняется команда. После завершения команды регистр перезаписывается 0.

SET\_RESET GPIO. Битовые регистры. Управляют каждый своей GPIO. Чтобы выставить высокий уровень на GPIO необходимо записать в соответствующий регистр 1 и выставить Scaler в 1.

Пример 1:

Для выставления в высокое состояние регистров которые были ранее проинициализированы в примере 2 инициализации нужно записать следующие значения в регистры. Reg num 1060 – 0x1801, 1061 – 0x8000, 1063 – 0x0С00. После выполнения команды в регистрах будет выставлено следующее - 1060 – 0x0801, 1061 – 0x8000, 1063 – 0x0С00. Изменится только Scaler.

Пример 2:

Для выставления в низкое состояние тех же самых регистров, необходимо записать в них 0. 1060 – 0x1000, 1061 – 0x0000, 1063 – 0x0000. После завершения выполнения команды состояние изменится на 1060 – 0x0000, 1061 – 0x0000, 1063 – 0x0000

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1064 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Scaler | | | | SET\_RESET GPIO | | | | | | | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| 0/1 | | | | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1065 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| SET\_RESET GPIO | | | | | | | | | | | | | | | |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 24 | 22 | 23 | 24 | 25 | 26 | 27 | 28 |
| 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1066 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| SET\_RESET GPIO | | | | | | | | | | | | | | | |
| 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 |
| 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1067 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| SET\_RESET GPIO | | | | | | | | | | | | | | | |
| 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 |
| 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 |

## Регистры чтения GPIO

Регистры чтения находятся в области analog inputs.

Регистры обновляются автоматически 1 раз в 500 мс. В регистрах хранится последнее считанное состояние. Данные регистры можно использовать для контроля выставленных значений GPIO.

|  |  |
| --- | --- |
| Addr | Name |
| 2070 | GET GPIO1-12 |
| 2071 | GET GPIO 13-28 |
| 2072 | GET GPIO 29-44 |
| 2073 | GET GPIO 45-60 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1064 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Reserved | | | | SET\_RESET GPIO | | | | | | | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| 0 | | | | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1065 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| SET\_RESET GPIO | | | | | | | | | | | | | | | |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 24 | 22 | 23 | 24 | 25 | 26 | 27 | 28 |
| 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1066 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| SET\_RESET GPIO | | | | | | | | | | | | | | | |
| 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 |
| 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1067 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| SET\_RESET GPIO | | | | | | | | | | | | | | | |
| 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 |
| 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 | 0/1 |

Регистры альтернативного (временного) состояния GPIO

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1228 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Scaler | | | | | | | | | | | | | | | |
| 0/1 | | | | | | | | | | | | | | | |

Scaler – регистр запуска обновления команды.  
При записи 1 – обновляется и исполняется команда. После завершения команды регистр перезаписывается 0.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1229 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| LOW\_Time | | | | | | | | | | | | | | | |
| Reg num 1230 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| HIGH\_Time | | | | | | | | | | | | | | | |

Регистры Low и High time вместе составляют 32 битную переменную определяющую время выставления альтернативного состояния. Время в микросекундах.

(uint32\_t) TIME = (HIGH\_time << 16 )| (LOW\_time)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1231 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Start | | | | | | | | | | | | | | | |
| 0/1 | | | | | | | | | | | | | | | |
| Reg num 1232 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| End\_flag | | | | | | | | | | | | | | | |
| 0/1 | | | | | | | | | | | | | | | |
| Reg num 1233 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Stop | | | | | | | | | | | | | | | |
| 0/1 | | | | | | | | | | | | | | | |

Регистры Start и Stop – обеспечивают запуск процесса выставления альтернативного состояния и принудительную остановку процесса соответственно. При записи 1 в Start запускается альтернативное состояние. При записи 1 в Stop процесс останавливается (даже если записана 1 в Start.

Регистр End\_Flag принимает значение 1 при штатном завершении процесса (прошло заданное время)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1234 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Process\_flag | | | | | | | | | | | | | | | |
| 0/1 | | | | | | | | | | | | | | | |

Регистр Proces\_flag = 1 на протяжении времени выставления альтернативного состояния GPIO.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1233 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| LOW\_Time\_left | | | | | | | | | | | | | | | |
| Reg num 1234 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| HIGH\_Time\_left | | | | | | | | | | | | | | | |

Регистры Low и High time left вместе составляют 32 битную переменную определяющую время которое осталось до конца выставления альтернативного состояния. Время в микросекундах. Обновляется 1 раз в 500 мс. Time = (HIGH\_Time\_left <<16)|(LOW\_time\_left)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1242 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 15 | 14 | | 13 | | 12 | | 11 | | 10 | | 9 | | 8 | | 7 | | | 6 | | 5 | 4 | 3 | | 2 | | 1 | | 0 | |
| reserved | | | | | | | GPIO | | | | | | | | | | | | | | | | | | | | | | |
| GPIO1 | | 2 | | 3 | | 4 | | 5 | | | 6 | | 7 | 8 | 9 | | 10 | | 11 | | GPIO12 | |
| 0-16 | | | | | | | 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 | | | 0/1 | | 0/1 | 0/1 | 0/1 | | 0/1 | | 0/1 | | 0/1 | |
| Reg num 1243 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 15 | | 14 | | 13 | | 12 | | 11 | | 10 | | 9 | | 8 | | 7 | 6 | | 5 | | 4 | | 3 | | 2 | | 1 | | 0 |
| GPIO | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| GPIO13 | | 14 | | 15 | | 16 | | 17 | | 18 | | 19 | | 20 | | 21 | 22 | | 23 | | 24 | | 25 | | 26 | | 27 | | GPIO28 |
| 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 | 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 |
| Reg num 1244 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 15 | | 14 | | 13 | | 12 | | 11 | | 10 | | 9 | | 8 | | 7 | 6 | | 5 | | 4 | | 3 | | 2 | | 1 | | 0 |
| GPIO | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| GPIO29 | | 30 | | 31 | | 32 | | 33 | | 34 | | 35 | | 36 | | 37 | 38 | | 39 | | 40 | | 41 | | 42 | | 43 | | GPIO44 |
| 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 | 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 |
| Reg num 1245 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 15 | | 14 | | 13 | | 12 | | 11 | | 10 | | 9 | | 8 | | 7 | 6 | | 5 | | 4 | | 3 | | 2 | | 1 | | 0 |
| GPIO | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| GPIO45 | | 46 | | 47 | | 48 | | 49 | | 50 | | 51 | | 52 | | 53 | 54 | | 55 | | 56 | | 57 | | 58 | | 59 | | GPIO60 |
| 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 | 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 | | 0/1 |

Регистры определяющие состояние GPIO на заданное время, аналогично предыдущим.

# Регистры SPI

## SPI Settings.

Находятся в области analog\_outputs

|  |  |
| --- | --- |
| Addres | Name |
| 1246 | Scaler |
| 1247 | Mode |
| 1248 | Direction |
| 1249 | Data\_size |
| 1250 | Polarity |
| 1251 | Phase |
| 1252 | Slave |
| 1253 | Baud |
| 1254 | First\_bit |
| 1255 | Ti\_mode |
| 1256 | Init\_flag |
| 1257 | Set\_default |

Scaler – регистр запуска обновления команды.  
При записи 1 – обновляется и исполняется команда. После завершения команды регистр перезаписывается 0.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1246 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Scaler | | | | | | | | | | | | | | | |
| 0x0/0x1 | | | | | | | | | | | | | | | |

Mode. Master – 0, Slave – 1. Slave – не используется.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1247 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Mode | | | | | | | | | | | | | | | |
| 0x0/0x1 | | | | | | | | | | | | | | | |

Direction. 0 – 2 Line, 1 – 2 Line Rx only, 2 – 1 Line.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1248 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Direction | | | | | | | | | | | | | | | |
| 0x0/0x1/0x2 | | | | | | | | | | | | | | | |

Data\_size. 0 – 8 Bit, 1 – 16 Bit.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1249 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Data\_size | | | | | | | | | | | | | | | |
| 0x0/0x1 | | | | | | | | | | | | | | | |

Polarity. 0 – Low, 1 – High.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1250 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Polarity | | | | | | | | | | | | | | | |
| 0x0/0x1 | | | | | | | | | | | | | | | |

Phase. 0 – One Edge, 1 – Two edge.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1251 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Phase | | | | | | | | | | | | | | | |
| 0x0/0x1 | | | | | | | | | | | | | | | |

Baud. 0 – 21MB/s, 1 – 10,5 MB/s, 2 – 5,25 MB/s, 3 – 2,625 MB/s, 4 – 1,3125 MB/s, 5 – 656,25 kB/s, 6 – 328,125 kB/s, 7 – 164,062 kB/s.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1253 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Baud | | | | | | | | | | | | | | | |
| 0x0-0x7 | | | | | | | | | | | | | | | |

First bit. 0 – Msb\_first, 1 – Lsb\_first.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1254 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| First\_bit | | | | | | | | | | | | | | | |
| 0x0/0x1 | | | | | | | | | | | | | | | |

Ti Mode. 0 – disable, 1 – enable.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1255 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Ti Mode | | | | | | | | | | | | | | | |
| 0x0/0x1 | | | | | | | | | | | | | | | |

Enable flag. Перед инициализацией сбрасывается принудительно мк в 0. После успешной инициализации выставляется в 1.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1256 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Enable flag | | | | | | | | | | | | | | | |
| 0x0/0x1 | | | | | | | | | | | | | | | |

Set default. Если выставлен в 1, то принудительно выставляются настройки по умолчанию. Master, 2Lines, 8bit, Polarity\_low, Phase\_1Edge, Baud\_rate – 21MB/s. TiMode – disable.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1257 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Set default | | | | | | | | | | | | | | | |
| 0x0/0x1 | | | | | | | | | | | | | | | |

## SPI Transmit registers.

Находятся в области analog outputs

|  |  |
| --- | --- |
| Addr | Name |
| 1266 | Scaler |
| 1268 | Len |
| 1269 | Start |
| 1270 | Transaction\_end |
| 1271 | Rx\_tx\_flag |
| 1276 | Data[0], Data[1] – 8 Bit format.  Data[0] – 16 Bit format |
| 1307 | Data [62], Data[63] – 8 Bit format  Data[31] – 16 Bit format |

Scaler – регистр запуска обновления команды.  
При записи 1 – обновляется и исполняется команда. После завершения команды регистр перезаписывается 0.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1266 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Scaler | | | | | | | | | | | | | | | |
| 0x0/0x1 | | | | | | | | | | | | | | | |

Len. Количество данных на отправку. Не более 64 для 8 битного режима. Не более 32 для 16 битного режима.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1268 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Len | | | | | | | | | | | | | | | |
| 0 – 64 / 0 – 32 | | | | | | | | | | | | | | | |

Start. Если установлен в 1. То выполняется транзакция.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1269 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Start | | | | | | | | | | | | | | | |
| 0x0/0x1 | | | | | | | | | | | | | | | |

Transaction end. Перед началом транзакции выставляется в 0. По завершению транзакции выставляется в 1.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1270 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Transaction end | | | | | | | | | | | | | | | |
| 0x0/0x1 | | | | | | | | | | | | | | | |

Rx tx flag. 0 – режим работы только передатчик. 1 – режим работы передатчик/приемник. Данные приема складываются в регистры данных приемника.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1271 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Rx\_tx\_flag | | | | | | | | | | | | | | | |
| 0x0/0x1 | | | | | | | | | | | | | | | |

Data. Регистры данных на отправку.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1276 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Data[0] 8 bit format | | | | | | | | Data[1] 8 bit format | | | | | | | |
| Data[0] 16 bit format | | | | | | | | | | | | | | | |
| Reg num 1307 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Data[62] 8 bit format | | | | | | | | Data[63] 8 bit format | | | | | | | |
| Data[31] 16 bit format | | | | | | | | | | | | | | | |

## SPI Receive

Регистры приемника разделены на две части. Управление и данные.   
Регистры управления находятся в области analog outputs. Регистры данных находятся в области analog inputs.

Регистры управления приемником

|  |  |
| --- | --- |
| Addres | Name |
| 1308 | Scaler |
| 1310 | Len |
| 1311 | Start |
| 1312 | Transaction\_end |

Scaler – регистр запуска обновления команды.  
При записи 1 – обновляется и исполняется команда. После завершения команды регистр перезаписывается 0.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1308 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Scaler | | | | | | | | | | | | | | | |
| 0x0/0x1 | | | | | | | | | | | | | | | |

Len – длина данных на прием. В 16 битном режиме максимальное значение 64. В 8 битном режиме максимальное значение 128.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1310 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Len | | | | | | | | | | | | | | | |

Start. Если установлен в 1. То выполняется транзакция.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1311 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Start | | | | | | | | | | | | | | | |
| 0x0/0x1 | | | | | | | | | | | | | | | |

Transaction\_end – перед началом транзакции выставляется в 0. По завершению транзакции выставляется в 1.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 1312 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Transaction\_end | | | | | | | | | | | | | | | |
| 0x0/0x1 | | | | | | | | | | | | | | | |

Регистры данных приемника SPI находятся в области analog\_inputs

Data. Регистры данных принятых.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reg num 2074 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Data[0] 8 bit format | | | | | | | | Data[1] 8 bit format | | | | | | | |
| Data[0] 16 bit format | | | | | | | | | | | | | | | |
| Reg num 2137 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Data[62] 8 bit format | | | | | | | | Data[63] 8 bit format | | | | | | | |
| Data[31] 16 bit format | | | | | | | | | | | | | | | |