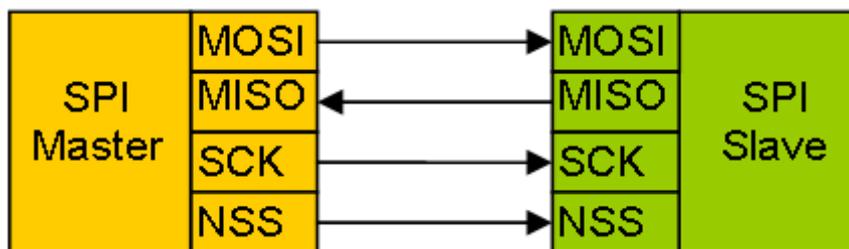


SPI в микроконтроллерах STM32. Основы

SPI (serial peripheral interface), последовательный интерфейс периферийных устройств - интерфейс, обеспечивающий полнодуплексную синхронную последовательную связь между устройствами; для связи используются четыре однонаправленные сигнальные линии (две линии данных - для передачи и приёма данных, тактовый сигнал и сигнал выбора "кристалла").



Типичный вариант подключения устройства по SPI.

Что это означает?

- Полнодуплексная связь - устройство может одновременно передавать и получать данные через интерфейс SPI.
- Синхронная связь - передача данных по SPI интерфейсу синхронизируется отдельным тактовым сигналом (используется общий тактовый сигнал как для принимаемых, так и для отправляемых данных). Наличие сигнала синхронизации намного упрощает реализацию интерфейса, решает многие проблемы (например, проблему согласования скоростей).
- Последовательная связь - информация по линиям данных передаётся в последовательной форме, по одному биту.

Первоначально разработанный компанией Motorola, интерфейс SPI впоследствии получил очень широкое распространение. Он поддерживается многими периферийными устройствами, выпускаемыми как в виде модулей, так и в виде отдельных микросхем. В частности, это всевозможные средства отображения информации (текстовые и графические дисплеи); датчики; модули беспроводной связи и многое другое. Следует упомянуть микросхемы энергонезависимой памяти с интерфейсом SPI, применяемые для хранения данных, а также для хранения прошивок устройств (загружаемых в RAM память при каждом запуске; такой способ хранения довольно удобен с точки зрения выполнения обновления прошивки). Кроме того, поддержку SPI имеют SD/microSD карты - они способны работать в SPI-режиме и могут использоваться устройствами как накопители данных большого объёма.

Многие микроконтроллеры имеют аппаратную поддержку SPI, что существенно упрощает использование внешней периферии, подключаемой по данному интерфейсу. Благодаря интерфейсу, может быть существенно упрощена разработка устройства, если пойти по пути объединения готовых компонентов, обменивающихся данными по SPI.

Интерфейс SPI достаточно прост с точки зрения реализации. При наличии такого желания, интерфейс может быть без труда реализован как программно, так и аппаратно, в том числе на базе простейших логических схем.

Обобщённо, без привязки к конкретной реализации, данный интерфейс рассматривается в статье "SPI".

Здесь уделим внимание особенностям реализации SPI в микроконтроллерах STM, в том числе рассмотрим специфичные, редко используемые режимы работы. Для быстрого ознакомления с вопросами практического использования SPI, смотрите ["Примеры использования SPI в STM32"](#).

Оглавление

SPI в микроконтроллерах STM32. Основы

Введение

Основные характеристики

Устройство SPI

Управление выбором ведомого устройства

Полярность и фаза тактового сигнала

Источники и дополнительная информация

SPI в STM32. Работа в ведомом и ведущем режимах

SPI в STM32. Управление передачей данных

Регистры SPI

Использование SPI при работе с микроконтроллерами STM32F100xx

Примеры программ

Введение

В микроконтроллерах STM32 имеется поддержка SPI, разработчику предоставлена масса возможностей по гибкому конфигурированию интерфейса.

Естественно, что SPI может быть сконфигурирован для работы в качестве ведущего или ведомого устройства (в режиме ведущего устройства SPI является источником тактового сигнала SCK для ведомого устройства; соответственно, ведомое устройство получает тактовый сигнал от другого устройства, работающего в режиме ведущего). Интерфейс допускает конфигурации с несколькими ведущими устройствами. Кроме обычной полнодуплексной синхронной передачи по двум линиям, предусмотрена возможность передачи данных в одну сторону по одной линии; поддерживается полудуплексная связь по одной двунаправленной линии. Имеются средства для поддержки надёжной связи (CRC контроль целостности данных).

Рассмотрим подробнее эти и другие возможности SPI в микроконтроллерах STM32.

Предупреждение. Некоторые выводы устройств SPI1, SPI3 и выводы интерфейса JTAG могут иметь привязку к общим внешним выводам в некоторых моделях микроконтроллеров. Тогда для использования такого SPI можно:

- отключить JTAG, а для отладки использовать интерфейс SWD;
- отключить как JTAG, так и SWD интерфейс для автономных, не требующих отладки приложений.

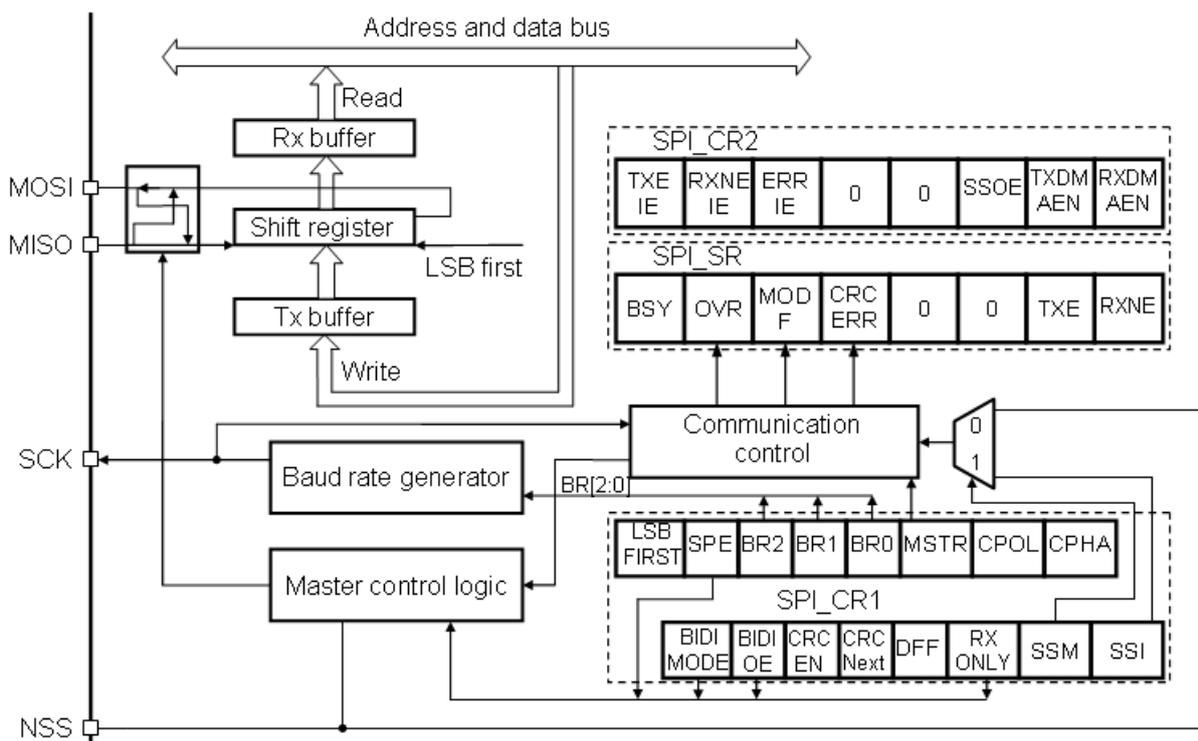
Основные характеристики

Интерфейс SPI в микроконтроллерах STM32F100xx имеет следующие основные возможности.

- Поддерживается полдуплексная (двусторонняя) синхронная связь по трём линиям: MOSI, MISO, SCK (без учёта линии NSS). Это наиболее типичный вариант использования SPI.
- Поддерживается симплексная (односторонняя) синхронная связь или полдуплексная связь по двум линиям (без учёта линии NSS); полдуплексная связь требует одной двунаправленной линии данных.
- Имеется возможность выбора длины фрейма для передачи данных: 8 или 16 битов. Обычно размер фрейма составляет 8 бит.
- Имеется возможность работы в качестве ведущего или ведомого устройства.
- Поддерживается режим работы с несколькими ведущими устройствами.
- Имеется предделитель на 8 скоростей передачи данных ($f_{PCLK}/2$ максимум, $f_{PCLK}/256$ минимум; PCLK - тактовый сигнал на шине микроконтроллера для периферийных устройств, APB1 или APB2, на той из них, на которой находится данное устройство SPI).
- Возможно аппаратное или программное управление функциями линии NSS как в ведущем, так и ведомом режиме; допускается динамическое переключение между ведущим и ведомым режимом.
- Настраивается полярность и фаза тактового сигнала SPI.
- Есть возможность задать порядок битов при передаче данных: MSB (старший бит) первый или LSB (младший бит) первый. Чаще для SPI выбирается вариант MSB.
- Имеются отдельные флаги, сигнализирующие о передаче и получении данных с возможностью генерации прерываний в ответ на установку флага.
- Имеется флаг занятого состояния SPI шины.
- Имеется возможность аппаратного вычисления CRC для контроля достоверности передаваемых данных:
 - CRC код может автоматически передаваться в качестве последнего байта (фрейма) в пакете данных из нескольких фреймов в Tx режиме;
 - возможен автоматический контроль ошибок CRC при получении последнего байта (фрейма) в пакете данных из нескольких фреймов.
- Наличие флагов ошибок: переполнение, ошибка CRC, ошибка при работе в ведущем режиме (конфликт в среде с несколькими ведущими устройствами). Возможность генерации прерывания при установке любого из указанных флагов.
- Наличие буферов для передачи и получения данных объёмом в один фрейм с поддержкой механизмов DMA для отправки и приёма данных.

Устройство SPI

На рисунке изображена блок-схема SPI в микроконтроллерах STM32F100xx, а с учётом высокой степени совместимости разных семейств микроконтроллеров STM между собой, она применима и ко многим другим устройствам.



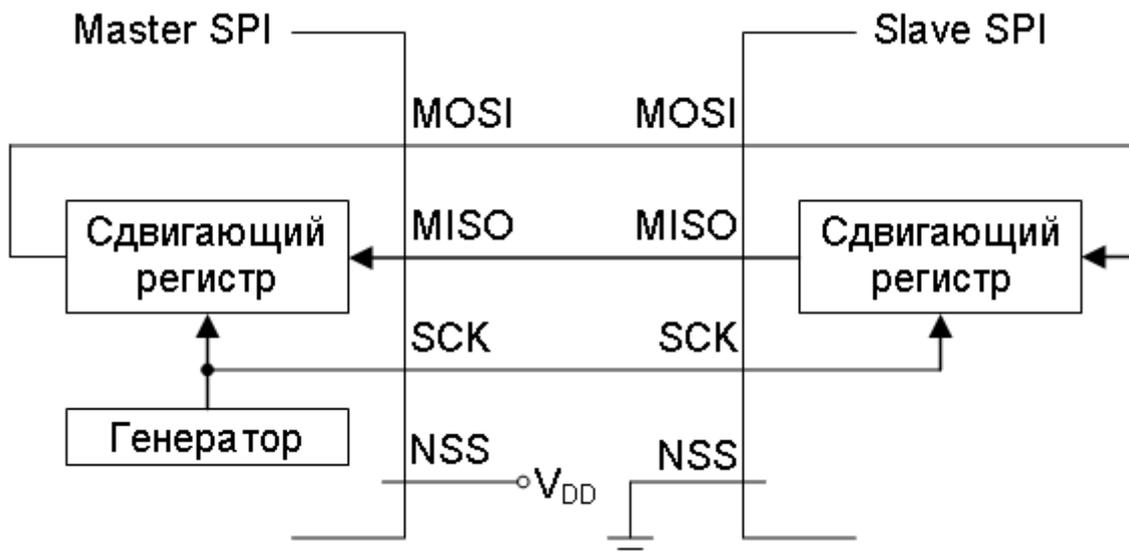
Обычно для подключения внешнего устройства к SPI используется 4 вывода.

- MISO: Master In / Slave Out data - вход ведущего устройства, выход ведомого. Этот вывод используется для передачи данных при работе в режиме ведомого устройства и для получения данных при работе в режиме ведущего. Если устройство является ведомым, но не выбрано (в зависимости от настроек - сигналом NSS или программно), его выход находится в высокоимпедансном состоянии (Z-состоянии).
- MOSI: Master Out / Slave In data - выход ведущего устройства, вход ведомого. Этот вывод используется для передачи данных при работе в режиме ведущего устройства и для получения данных при работе в режиме ведомого.
- SCK: Serial Clock - тактовый сигнал. Является выходом для ведущего устройства и входом для ведомого.
- NSS: Slave select - выбор ведомого устройства. Это необязательный вывод, служит для выбора ведомого устройства (с которым ведущее устройство намерено производить обмен данными). Этот вывод действует как сигнал "выбора кристалла" (chip select) и позволяет ведущему устройству взаимодействовать с каждым из ведомых устройств в индивидуальном порядке, избегая конфликтов в линии MISO.

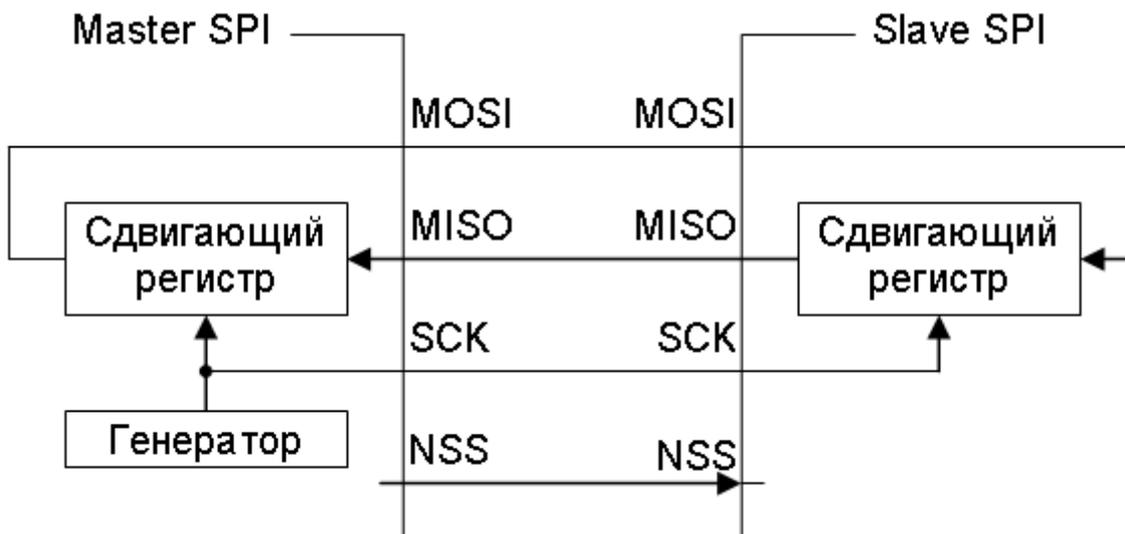
У ведомого устройства вывод NSS - это вход, если выбран аппаратный режим NSS. Ведомое устройство может отказаться от использования внешнего сигнала NSS и управлять своим выбором самостоятельно, программно (установкой бита SSI в регистре SPI_CR1 при установленном бите SSM). Это может быть хорошим решением в случае единственного ведомого устройства.

Вывод NSS ведущего устройства может быть входом (обычно используется при наличии нескольких ведущих устройств на шине) или выходом (только если на шине только одно ведущее устройство). Режим выхода для ведущего устройства включается установкой бита SSOE (регистр SPI_CR2). В таком случае, на выходе NSS ведущего устройства устанавливается низкий уровень, когда начинается обмен данными и остаётся на низком уровне до отключения SPI. NSS как выход довольно бесполезен: если ведомое устройство одно, зачастую можно обойтись вообще без сигнала NSS; если ведомых устройств несколько, в любом случае необходимо наличие у ведущего устройства нескольких выводов NSS. А поскольку аппаратный вывод NSS только один, то всё равно потребуется формирование нужного количества сигналов NSS программными средствами с использованием стандартных портов ввода-вывода.

Пример варианта соединения между единственным ведущим и единственным ведомым устройством изображён на рисунке. Здесь у обоих устройств вывод NSS сконфигурирован как вход. Ещё проще настроить NSS ведущего устройства как выход и оставить его неподключённым.



Типичный вариант подключения, когда задействованы все линии интерфейса, изображён на рисунке. Здесь NSS является выходом у ведущего устройства и входом у ведомого, выбор ведомого осуществляется низким уровнем сигнала NSS.



Как видно из рисунков, для подключения по SPI, одноимённые выводы объединяемых устройств подключаются вместе (выводы MOSI устройств соединяются вместе, выводы MISO соединяются вместе, выводы SCK соединяются вместе). Использование NSS зависит от того, каким образом сконфигурированы устройства.

Данные передаются между ведущим и ведомым устройством в последовательной форме, по одному биту. Обычно старший бит передаётся первым, но порядок можно программно настроить. Пересылку данных всегда инициализирует ведущее устройство. Когда ведущее устройство передаёт данные к ведомому через вывод MOSI, оно одновременно получает данные от ведомого устройства через вывод MISO. Таким образом реализуется полнодуплексная связь с синхронизацией и исходящих, и входящих данных одним и тем же тактовым сигналом, который формируется ведущим устройством на выводе SCK.

Преобразование данных из параллельной формы, в которой данные пересылаются по шине микроконтроллера, в последовательную форму для передачи через SPI, осуществляется с помощью регистра сдвига, Shift register на [рисунок](#). Данные, предназначенные для передачи через SPI, загружаются в этот регистр из буферного регистра Tx. Когда завершается передача-приём очередного фрейма по SPI и полученный фрейм оказывается в регистре сдвига, содержимое регистра сдвига помещается в буферный регистр Rx. Программный доступ к регистрам осуществляется посредством регистра SPI_DR: при записи в него, данные помещаются в Tx; при чтении из регистра, получаем данные из Rx.

Прежде чем выполнять запись в SPI_DR, следует убедиться в том, что установлен флаг TXE в регистре SPI_SR, который свидетельствует, что буфер Tx в данный момент пуст (помещённые в него ранее данные уже перемещены в регистр сдвига, либо никакие данные в него ещё не записывались). О том, что в регистре Rx содержатся новые данные, полученные по SPI, сигнализирует флаг SPI_RXNE.

Управление выбором ведомого устройства

Выбор ведомого устройства может осуществляться аппаратно или программно, в зависимости от значения бита SSM в регистре SPI_CR1.

- Программное управление (SSM = 1).
 - Сигнал выбора для ведомого устройства формируется внутренне на основе состояния бита SSI в регистре SPI_CR1. Внешний вывод NSS остаётся свободным и может использоваться в других целях.
 - Аппаратное управление (SSM = 0).

Возможны две конфигурации, в зависимости от бита SSOE в регистре SPI_CR2, который переключает вывод NSS в режим выхода.

- NSS вывод сконфигурирован как выход (SSM = 0, SSOE = 1).

Эта конфигурация используется только при работе в ведущем режиме (вывод NSS ведомого устройства не может быть выходом). На выходе NSS устанавливается низкий уровень, когда ведущее устройство начинает обмен данными и сохраняется на этом уровне до отключения SPI.
- NSS вывод сконфигурирован как вход (SSM = 0, SSOE = 0).

В режиме ведомого устройства, NSS используется обычным образом: низкий уровень на входе делает ведомое устройство выбранным, высокий - невыбранным. В ведущем режиме данная конфигурация позволяет реализовать поддержку множественных ведущих устройств на одной шине. Ведущее устройство активно только пока уровень на входе NSS остаётся высоким. Если уровень на входе становится низким, ведущее устройство фиксирует ошибку, автоматически переходит в режим ведомого и отключается.

Полярность и фаза тактового сигнала

Имеется четыре возможных варианта тактирования передаваемых данных, которые выбираются с помощью битов CPOL и CPHA в регистре SPI_CR1. Биты влияют на работу SPI как в ведущем, так и в ведомом режиме.

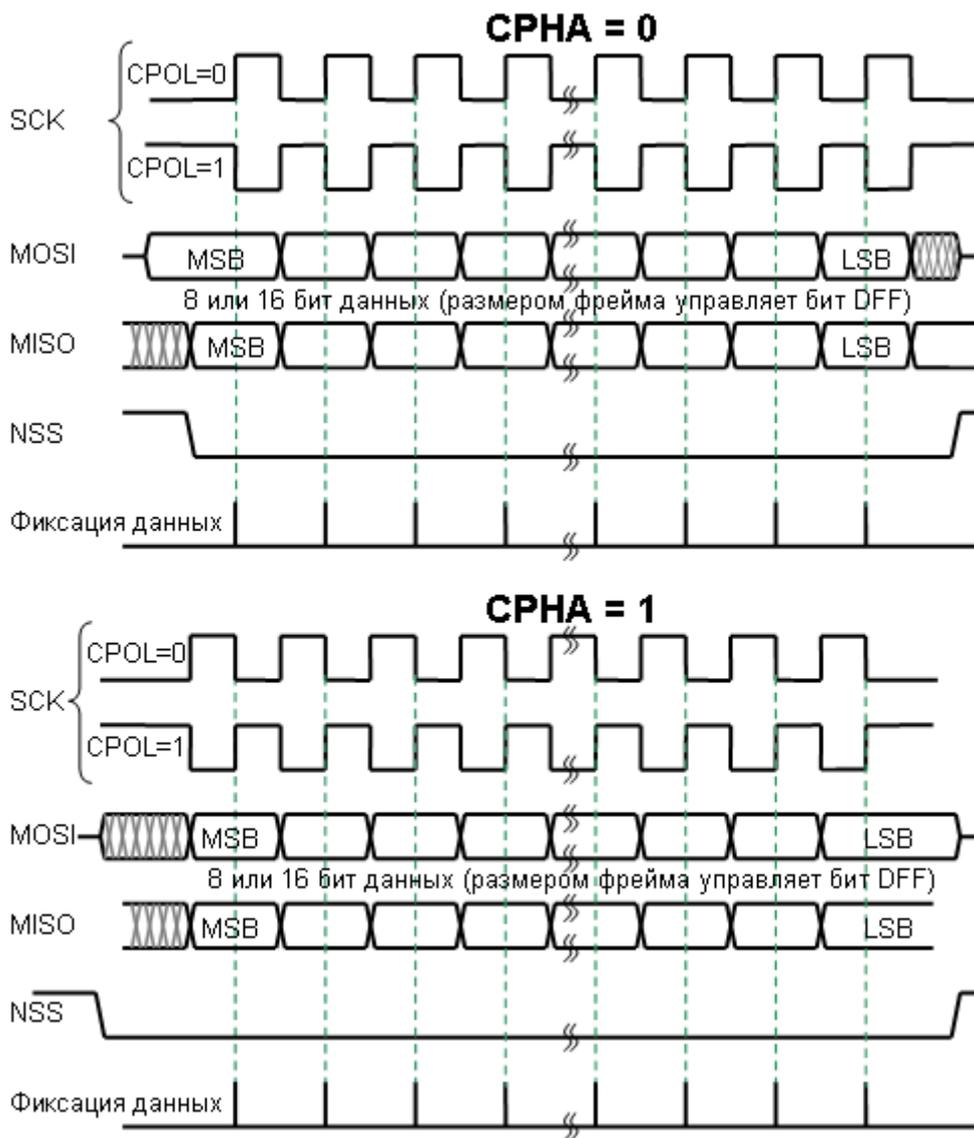
Бит CPOL контролирует полярность сигнала SCK, т.е. начальный (пассивный) уровень тактового сигнала, который он имеет до начала пересылки данных и в паузах между передачами. Этот уровень совпадает со значением бита: если бит сброшен, низкий уровень является начальным для тактового сигнала; если бит CPOL установлен, начальным является высокий уровень.

Бит CPHA контролирует фазу тактового сигнала. Если бит CPHA сброшен, то первый фронт сигнала на выводе SCK (нарастающий, если CPOL = 0; спадающий, если CPOL = 1) является первым стробирующим импульсом, по которому фиксируется первый передаваемый бит фрейма. Если бит CPHA установлен, то второй фронт на выводе SCK (спадающий, если CPOL = 0; нарастающий, если CPOL = 1) является первым стробирующим импульсом и фиксирует первый передаваемый бит данных.

Последующие биты фрейма также будут фиксироваться по нарастающим или спадающим фронтам тактового сигнала, в зависимости от комбинации битов CPOL и CPHA.

Для того чтобы была возможна связь между двумя устройствами, полярность и фаза тактового сигнала у них должны быть настроены одинаковым образом.

На рисунке приведены диаграммы процесса передачи данных при четырёх возможных комбинациях битов CPOL, CPHA. Диаграммы можно рассматривать как изображающие процессы на выводах как ведущего, так и ведомого устройства - одноимённые выводы устройств непосредственно соединены вместе и, естественно, сигналы на выводах совпадают.



Примечание. Изменять биты CPOL/CPHA допускается только при отключённом SPI (т.е. при сброшенном бите SPE). Поэтому, если необходимо изменить настройки тактового сигнала в процессе работы, сначала следует отключить SPI (о правильном отключении в разных режимах, без риска потери данных, смотрите: "Правильное отключение SPI"). Изменение настроек в процессе работы может потребоваться, например, если к одной SPI шине подключено несколько ведомых устройств, для которых необходимы разные варианты тактирования.

Хотя на рисунке показан случай, когда первым передаётся старший бит фрейма, установка бита LSBFIRST в регистре SPI_CR1 позволяет изменить порядок передачи битов (младший бит первым).

Имеется возможность выбора длины фрейма (8 или 16 бит) с помощью бита DFF в регистре SPI_CR1.

Понятно, что размер фрейма, порядок следования битов, а также параметры тактирования должны быть одинаковыми для ведомого и ведущего устройств, обменивающихся информацией. Скорость передачи настраивается только для ведущего устройства, для ведомого она определяется получаемым тактовым сигналом.

Смотрите далее:

[SPI в STM32. Работа в ведомом и ведущем режимах](#)

Источники и дополнительная информация

1. "SPI Block Guide V04.01"; Motorola, Inc. (Freescale Semiconductor, Inc., <http://www.freescale.com>); 21 JAN 2000, Revised: 14 JUL 2004.
2. "RM0041. Reference manual. STM32F100xx advanced ARM^(R)-based 32-bit MCUs"; STMicroelectronics; Doc ID16188 Rev 5; June 2016

author: hamper; date: 2020-09-30