Министерство образования и науки Украины
Национальный технический университет
«Харьковский политехнический институт»

Методические указания
к лабораторной работе № 5

«Связь микроконтроллера с компьютером по каналу I^2C»

по курсу «*Интерфейсы компьютерных систем управления*»

Харьков 2006

Министерство образования и науки Украины

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ЛАБОРАТОРНОЙ РАБОТЕ № 5

«СВЯЗЬ МИКРОКОНТРОЛЛЕРА С КОМПЬЮТЕРОМ ПО КАНАЛУ I^2C»

ПО КУРСУ «ИНТЕРФЕЙСЫ КОМПЬЮТЕРНЫХ СИСТЕМ УПРАВЛЕНИЯ»

для студентов специальности 7.0910002

Утверждено
редакционно-издательским
советом университета,
протокол № 2 от 02.06.04

Харьков НТУ «ХПИ» 2006

Методические указания к лабораторной работе «Связь микроконтроллера с компьютером по каналу I^2C» по курсу «Интерфейсы компьютерных систем управления» для студентов специальности 7.0910002 / Сост. Д.А. Лунин, М.В. Гунбин. – Харьков: НТУ «ХПИ» 2006. – 13 с.

Составители: Д.А. Лунин, М.В. Гунбин

Рецензент А.В. Ивашко

Кафедра автоматики и управления в технических системах

Цель работы: изучить принципы работы интерфейса I^2C.

Краткий обзор интерфейса I^2C

I^2C - двухпроводный последовательный интерфейс, разработанный корпорацией Philips. В Первоначальном техническом требовании к интерфейсу максимальная скорость передачи данных составляла 100 Кбит/с. Однако позже появились стандартные более скоростные режимы работы шины I^2C (400Кбит/с и 1 Мбит/с). К одной шине I^2C могут быть подключены устройства с различными скоростями доступа, если скорость передачи данных будет удовлетворять требованиям самого низкоскоростного устройства.

Протокол передачи данных по шине I^2C разработан таким образом, чтобы гарантировать надежный качественный прием/передачу данных. При передаче данных одно устройство является «Ведущим», которое инициирует передачу данных и формирует сигналы синхронизации. Другое устройство «Ведомое», которое может начать передачу данных только по команде ведущего шины.

Каждое устройство на шине I^2C имеет уникальный адрес. Когда «ведущий» инициирует передачу данных, то сначала передается адрес устройства, к которому выполняется обращение. Остальные устройства проверяют переданный «ведущим» адрес. В состав байта адреса устройства входит бит направления передачи данных (выполняется чтение из «ведомого» или запись). «Ведомый» и ведущий шины всегда находятся в противоположном режиме работы, что можно представить в виде двух состояний:

- «Ведущий» передатчик – ведомый приемник.
- «Ведомый» передатчик – ведущий приемник.

В обоих случаях «ведущий» формирует тактовый сигнал.

Вывод тактового сигнала (SCL) и данных (SDA) должны иметь выход с открытым коллектором, чтобы выполнять требования «монтажного И» на шине. Для формирования высокого уровня сигнала на линиях к ним подключаются подтягивающие резисторы. Число устройств, которые могут быть подключены к шине I^2C, ограничивается только максимальной емкостью шины (400пФ) и способностью адресации этих устройств.

Инициализация и завершение передачи данных

В то время, когда передача данных на шине I^2C отсутствует, сигнал синхронизации (SCL) и данных (SDA) имеют высокий логический уровень за счет подтягивающих резисторов. Биты START и STOP формируются «ведущим» для определения начала и окончания передачи данных соответственно. Бит START формируется переходом сигнала SDA из высокого уровня в низкий при высоком уровне сигнала SCL. Бит STOP формируется переходом SDA из низкого уровня в высокий при высоком уровне SCL. На рисунке 1 показано формирование битов START и STOP. «Ведущий» шины формирует биты START и STOP для указания начала и завершения передачи данных. При передаче данных сигнал SDA может изменяться только, когда SCL имеет низкий логический уровень.

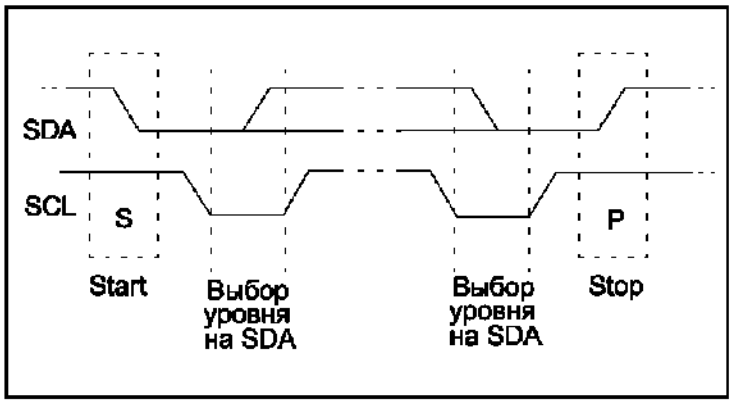

Рис. 1 Биты START и STOP

Адресация устройств на шине I^2C

Для адресации устройств используется два формата адреса: простой 7-разрядный формат с битом чтения/записи R/W (см. рис. 2); 10-разрядный формат, передается два байта. В первом байте передается: пять битов, определяющих, что это 10-разрядный адрес; два старших бита адреса; бит записи/чтения. Во втором байте передается 8 младших бит адреса (см. рис. 3).

Рис. 2 7-разрядная адресация

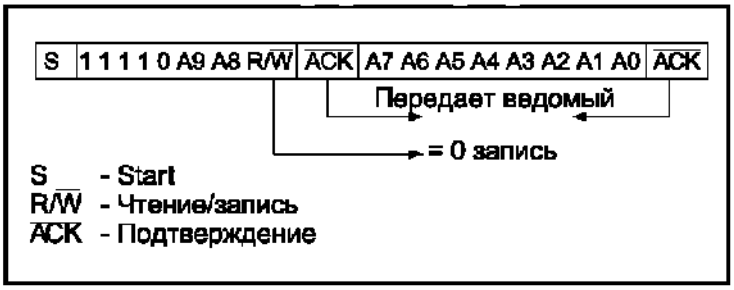

Рис. 3 10-разрядная адресация

Подтверждение приема

При передаче данных после каждого переданного байта приемник должен подтвердить получения байта сигналом АСК. (см. рис. 4). Если «ведомый» не подтверждает получение байта адреса или данных, «ведущий» должен прервать передачу сформировав сигнал STOP (ведомый должен отпустить SDA для формирования STOP «ведущим»).

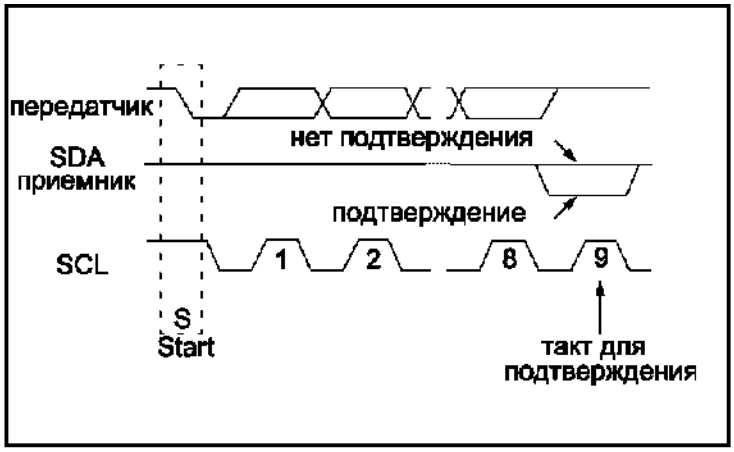

Рис. 4 Подтверждение приема ведомым

Когда ведущий шины принимает данные, то на каждый принимаемый байт формируется бит подтверждения, если принятый байт не последний. Для сообщения «ведомому» о том, что «ведущий» прекращает принимать данные по приему последнего байта – АСК не формируется. «Ведомый» отпускает SDA, чтобы «ведущий» смог передать бит STOP. «Ведущий» может формировать бит STOP на месте бита подтверждения.

Если «ведомому» необходимо задержать передачу данных, то он может удерживать SCL в низком логическом уровне. Передача данных продолжится, когда «ведомый» отпустит SCL. Это позволяет «ведомому» подготовить новые данные для передачи. Методика задержки передачи данных может использоваться и при передаче отдельных битов (см. рис. А-5).

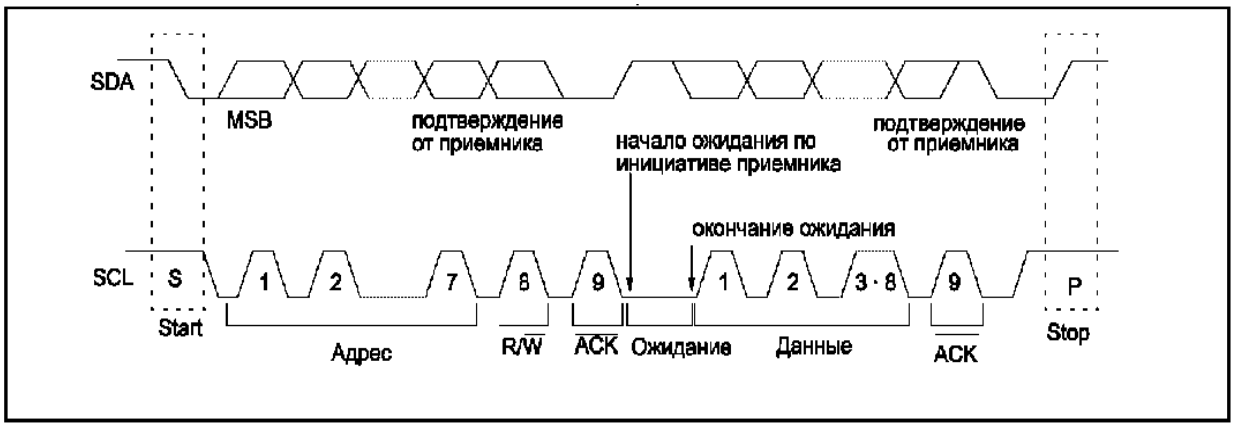

Рис. 5 Ожидание передачи данных

На рисунках 6 и 7 показаны последовательности приема и передачи данных «ведущим» шины.

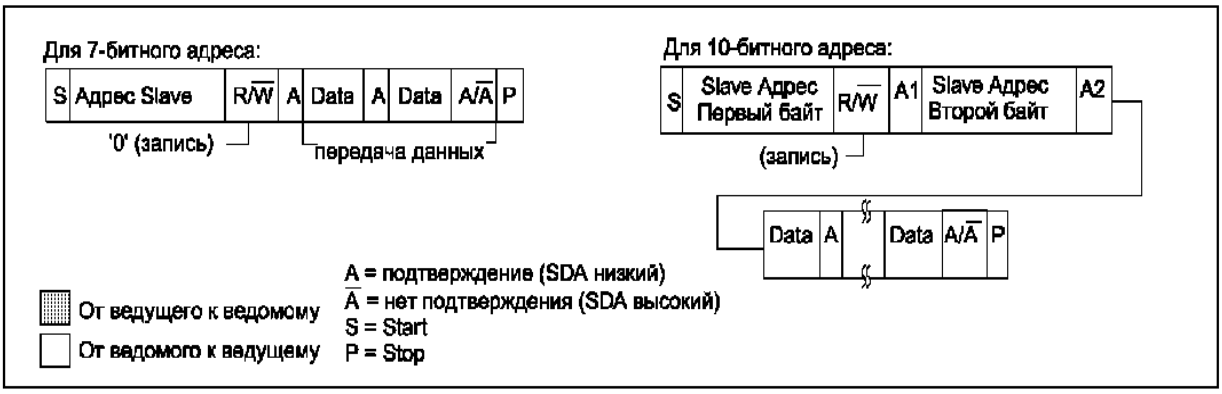

Рис. 6 Последовательность передачи данных «ведущим»

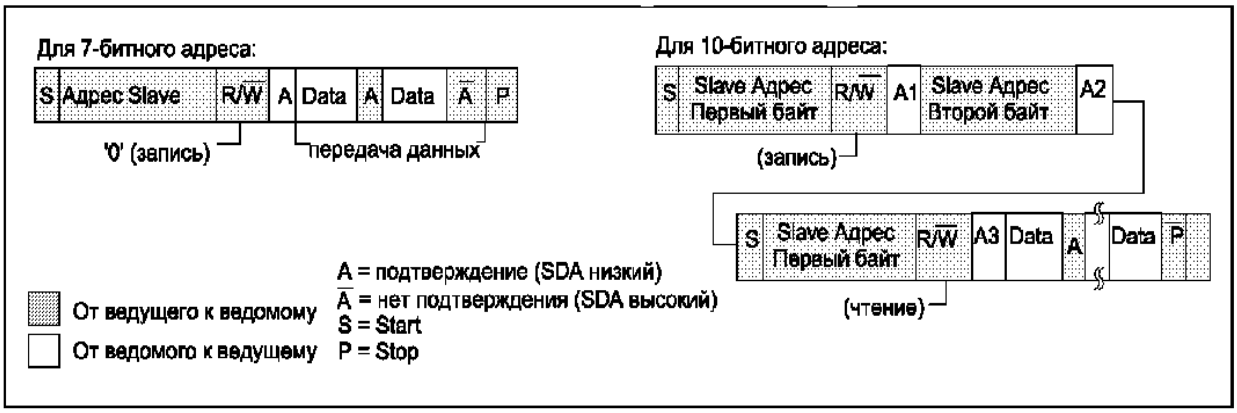

Рис. 7 Последовательность приема данных «ведущем»

Когда «ведущему» шины необходимо продолжить обмен данными (при формировании бита STOP управление шиной теряется) может быть передан бит повторный START. Условие повторный START идентично условию START (на SDA формируется переход с высокого логического уровня в низкий при высоком уровне сигнала на SCL), но формируется после передачи бита подтверждения. Это позволяет «ведущему» продолжить обмен с текущим устройством или адресовать новое (см. рис. 8).

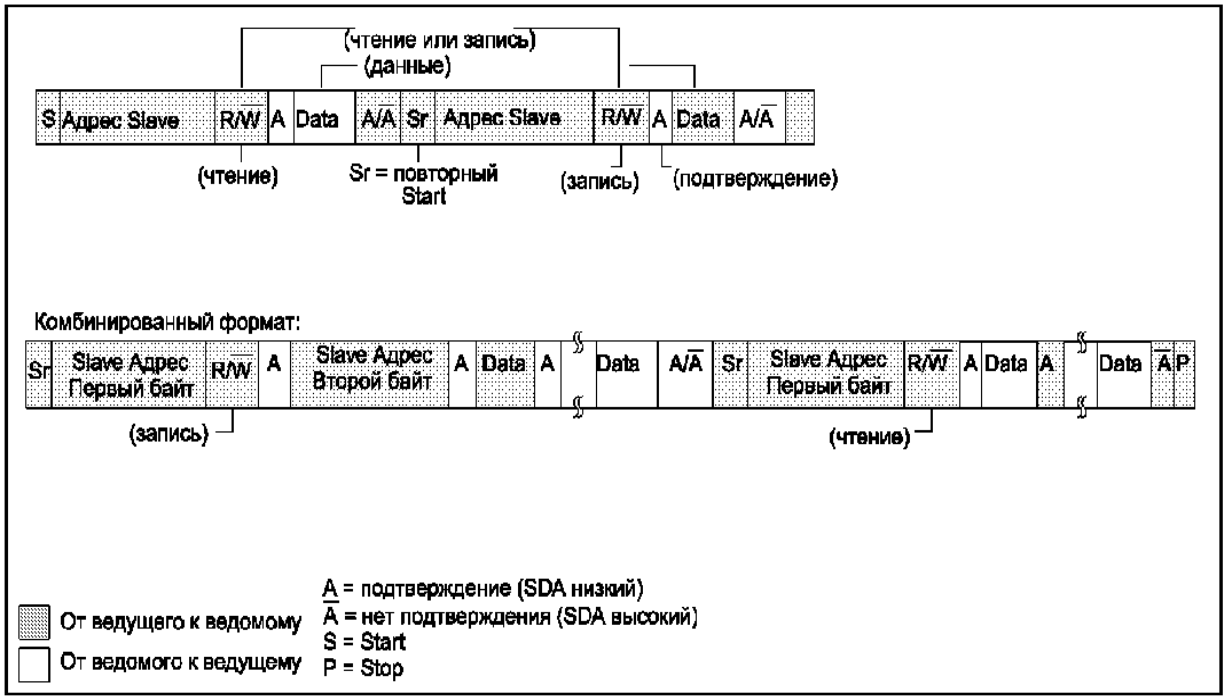

Рис. 8 Комбинированный формат

Режим конкуренции

Протокол передачи данных I^2C позволяет иметь более одного «ведущего» на шине. Для разрешения конфликтов на шине при инициализации передачи используются функции арбитража и синхронизации.

Арбитраж

Арбитраж выполняется на линии SDA при высоком уровне сигнала на SCL. Устройство, которое формирует на линии SDA высокий уровень, когда другое устройство передает низкий, теряет право быть «ведущим» и должно перейти в режим «ведомого». «Ведущий», потерявший инициативу на шине I^2C, может формировать тактовые импульсы до конца байта, в котором потерял управление шиной. Когда несколько «ведущих» адресуют одно и тоже устройство, то арбитраж выполняется при передаче данных.

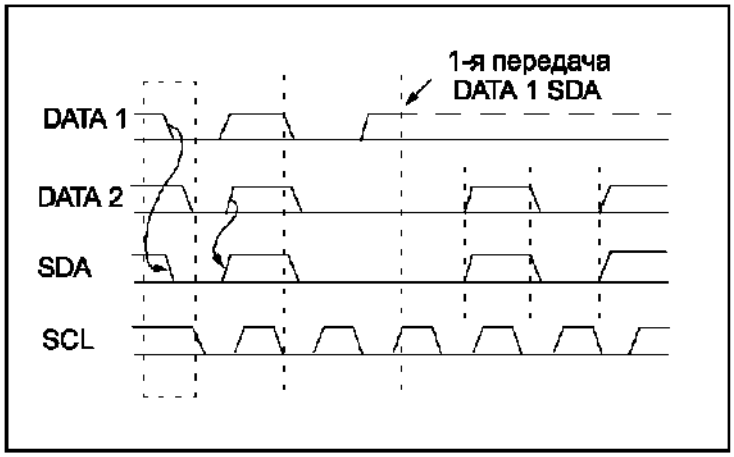

Рис. 9 Арбитраж на шине I C (два «ведущих»)

«Ведущий», потерявший арбитраж, должен немедленно перейти в режим

«ведомого», поскольку он может быть адресован текущим «ведущим».

Арбитраж не допускается между:

- Битами повторный START;
- Битом STOP и битом данных;
- Битами повторный START и STOP.

«Ведущий» шины должен гарантировать отсутствие указанных условий.

Синхронизация

Синхронизация тактового сигнала выполняется, когда устройства начинают арбитраж. Синхронизация реализуется за счет включение линии SCL по схеме "монтажное И". Переход сигнала на SCL с высокого логического уровня в низкий заставляет устройства, выполняющие арбитраж, начать отсчет длительности низкого логического уровня. После того, как тактовый сигнал устройства перешел в низкий уровень, оно будет удерживать этот уровень на SCL до тех пор, пока тактовый сигнал не перейдет в высокий уровень, но на SCL может бить по прежнему низкий уровень, если другое устройство формирует низкий логический уровень. Низкий уровень на SCL удерживается устройством с минимальной частотой тактового сигнала передачи данных. Устройства с меньшей длительностью низкого уровня на SCL переходят в состояние ожидания, пока на SCL не появится высокий логический уровень сигнала. Затем все устройства начинают отсчет длительность высокого уровня сигнала. Устройство, с минимальной длительность высокого уровня сигнала, первым переведет SCL в низкий уровень (см. рис. 10).

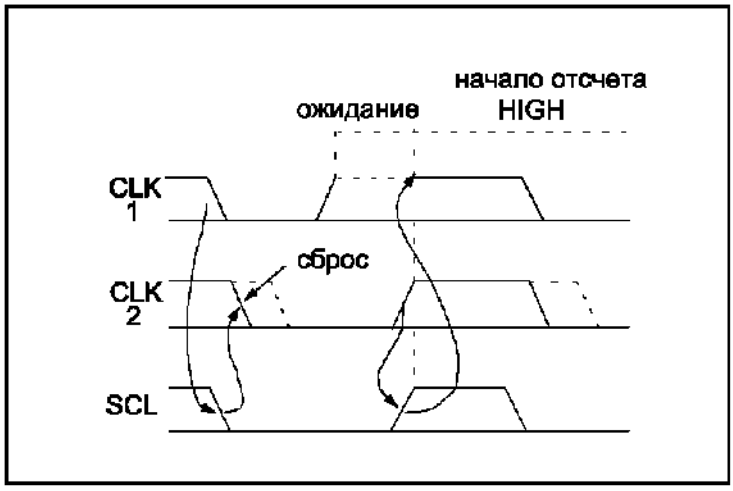

Рис. 10 Синхронизация тактового сигнала на шине I^2C

Задание на лабораторную работу.

1. На основании вышеприведенного теоретического материала разработать программу связи по интерфейсу I^2C микроконтроллера с компьютером. Для описания протокола обмена микроконтроллера использовать язык ассемблер. Для описания протокола обмена COM- порта в персональном компьютере, использовать среду разработки приложений Delphi.

2. Проверить наличие связи компьютера с контроллером посредством интерфейса I^2C.

3. Передать массив из компьютера в ОЗУ микроконтроллера, очистить ячейки

ОЗУ, хранящие массив, заполнить эти ячейки значением, указанным в команде компьютера, и считать массив из ОЗУ микроконтроллера в компьютер. Все запросы должен генерировать компьютер, на каждый запрос микроконтроллер должен дать ответ.

Контрольные вопросы.

1. Для чего предназначен интерфейс I^2C?
2. Какие режимы работы интерфейса I^2C?
3. Электрические параметры интерфейса I^2C.
4. Структура сообщения по интерфейсу I^2C.
5. Требования интерфейса I^2C.