

FLASH-I2C адаптер для LCD 1602/2004 - Datasheet

I2C - flash адаптер для LCD дисплеев 1602/2004.

Техническое описание: Данная страница содержит подробное техническое описание [I2C-flash адаптера для LCD дисплеев 1602/2004](#) и раскрывает работу с адаптером через его регистры. Ознакомиться с пользовательским описанием адаптера и примерами работы с ним, можно на странице [Wiki - I2C-flash адаптер для LCD дисплеев 1602/2004](#).

Назначение:

[I2C-flash адаптер для LCD дисплеев 1602/2004](#) - является преобразователем интерфейсов который обеспечивает обмен данными между двумя шинами: параллельной шиной LCD и шиной I2C. Наличие связи между этими шинами позволяет работать с LCD дисплеем по шине I2C.

- К одной шине LCD можно подключить один LCD дисплей 1602 или 2004.
- К одной шине I2C можно подключить более 100 адаптеров. Адрес адаптера на шине I2C (по умолчанию 0x09) назначается программно и хранится в его энергонезависимой памяти.

Описание:

Адаптер построен на базе микроконтроллера STM32F030F4, снабжен собственным стабилизатором напряжения, резистором настройки контрастности дисплея и переключателем режима работы: «ON» и «ADR».

- Если переключатель установлен в положение «ON», то адаптер находится в основном режиме работы, при котором осуществляется обмен данными между шинами LCD и I2C, что позволяет работать с LCD дисплеем по шине I2C. Работа с LCD дисплеем выполняется так же, как и при использовании [конвертера на базе чипа PCF8574](#), что делает адаптер полностью совместимым с [библиотеками разработанными для I2C LCD дисплеев 1602 и 2004](#).
- Если переключатель установлен в положение «ADR», то адаптер работает в режиме адресного доступа к своим регистрам, что позволяет управлять настройками адаптера. В этом режиме нельзя работать с LCD дисплеем, на его экране будет отображаться информация о текущем адресе адаптера на шине I2C и о установленной яркости подсветки дисплея в %. Доступ к

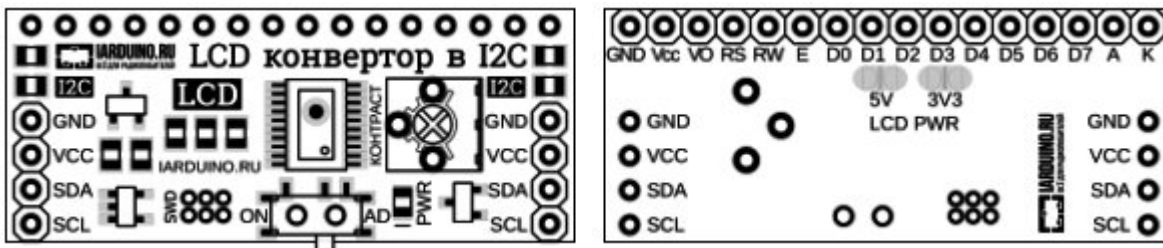
регистрам адаптера осуществляется по шине I2C.

В режиме «ON», передача данных по шине I2C осуществляется без указания адреса регистра, как при работе с чипом PCF8574. Каждый бит отправленного байта управляет одним из выводов шины LCD. Каждый бит полученного байта информирует о состоянии соответствующего вывода шины LCD.

В режиме «ADR», передача данных по шине I2C осуществляется с указанием адреса регистра. С помощью регистров, в данном режиме, можно выполнять следующие действия:

- Изменить адрес адаптера на шине I2C. При изменении адреса, можно указать, что новый адрес должен сохраниться в flash память модуля, а значит адрес сохранится и после отключения питания.
- Установить яркость свечения подсветки дисплея от 0 до 100%. Значение яркости автоматически сохраняется в flash память модуля, а значит яркость сохранится после отключения и включения питания.
- Включить / отключить внутреннюю подтяжку линий шины I2C (по умолчанию включена). Состояние подтяжки линий шины I2C автоматически сохраняется в flash память модуля, а значит состояние линий сохранится после отключения и включения питания.
- Узнать версию прошивки адаптера.

Выводы адаптера:



В верхней части платы расположена колодка шины LCD для подключения дисплея.

- **GND** - общий вывод питания.
- **Vcc** - вывод питания LCD дисплея 3,3 или 5 В (выбирается перемычками на плате).
- **VO** - (Voltage cOntrast) контрастность экрана.
- **RS** - (Register Select) линия выбора регистра команд / данных.
- **RW** - (Read/Write) линия выбора направления передачи.
- **E** - (Enable) линия тактирования по спаду.
- **D0-D7** - (Data) линии данных (адаптер использует только D4-D7).
- **A** - (Anode) положительный вывод питания подсветки LCD дисплея.
- **K** - (Cathode) отрицательный вывод питания подсветки LCD дисплея.

По бокам платы расположены разъемы для подключения адаптера к шине I2C. Шина подключается к любому разъему I2C, а второй разъем можно использовать для подключения следующего адаптера, или других устройств.

- **SCL** - вход/выход линии тактирования шины I2C.
- **SDA** - вход/выход линии данных шины I2C.
- **Vcc** - вход питания 3,3 или 5 В.
- **GND** - общий вывод питания.

По центру платы расположены две переключки выбора питания LCD дисплея.

- **5V** - установка данной переключки соединит вывод Vcc шины LCD с выводом Vcc шины I2C.
- **3V3** - установка данной переключки подключит вывод Vcc шины LCD к стабилизатору +3,3 В.
- Не устанавливайте обе переключки одновременно, это может повредить адаптер и дисплей.

Характеристики:

- Напряжение питания: 3,3 В или 5 В (постоянного тока).
- Потребляемый ток: до 6 мА.
- Интерфейс: I2C.
- Скорость шины I2C: 100 кбит/с.
- Адрес на шине I2C: устанавливается программно (по умолчанию 0x09).
- Уровень логической 1 на линиях шины I2C: 3,3 В (толерантны к 5 В).
- Рабочая температура: от -20 до +70 °С.
- Габариты: 41,91 x 17,15 мм (1650 x 675 mil).
- Вес: 6 г.

Установка адреса:

[I2C-flash адаптер для LCD дисплеев 1602/2004](#) относится к серии «Flash» модулей. Все модули данной серии позволяют назначать себе адрес для шины I2C, как временно (новый адрес действует пока есть питание), так и постоянно (новый адрес сохраняется в энергонезависимую память и действует даже после отключения питания). По умолчанию все модули серии «Flash» поставляются с адресом 0x09.

Допускается указывать адреса в диапазоне: $7 < \text{адрес} < 127$.

Для установки адреса необходимо перевести переключатель выбора режима работы адаптера в положение «ADR», при этом на экране должен появиться текущий адрес адаптера на шине I2C и установленная яркость подсветки экрана в %.

Установка адреса (без сохранения):

Если в регистр [0x06 «ADDRESS»](#) записать значение из 7 бит адреса и младшим битом «SAVE_FLASH» равным 0, то указанный адрес станет адресом модуля на шине I2C, но он не сохранится во FLASH памяти, а значит после отключения питания, установится прежний адрес модуля.

После записи на дисплее появятся два адреса: текущий адрес модуля (установленный без сохранения), а за ним постоянный адрес модуля (адрес хранимый в FLASH памяти).

Установка адреса (с сохранением):

Для установки адреса с его сохранением в FLASH память модуля необходимо выполнить два действия:

- Установить бит «SAVE_ADR_EN» в регистре [0x01 «BITS_0»](#) (при этом адрес модуля останется прежним).
- Записать в регистр [0x06 «ADDRESS»](#) значение из 7 бит адреса и младшим битом «SAVE_FLASH» равным 1.
- Если не выполнить первое действие (не установить бит «SAVE_ADR_EN»), то новый адрес будет проигнорирован и у модуля останется старый адрес.
- ВАЖНО: запись адреса занимает не менее 30 мс. После сохранения адреса бит «SAVE_ADR_EN» в регистре [0x01 «BITS_0»](#) самостоятельно сбросится в 0.

После записи адреса в FLASH память, он отобразится на дисплее.

Установка яркости подсветки:

Для установки яркости подсветки необходимо перевести переключатель выбора режима работы адаптера в положение «ADR», при этом на экране должен появиться текущий адрес адаптера на шине I2C и установленная яркость подсветки экрана в %.

Яркость устанавливается записью значения (от 0 до 255) в регистр [0x10 «LIGHT»](#), чем выше значение тем ярче подсветка. Регистр доступен как для записи, так и для чтения.

ВАЖНО: запись яркости подсветки занимает не менее 30 мс., т.к. значение регистра хранится в энергонезависимой памяти модуля.

После записи яркости подсветки в FLASH память, её значение отобразится на дисплее в %.

Регистры:

Карта регистров модуля:

адрес	7	6	5	4	3	2	1	0
HEТ	PIN_D7	PIN_D6	PIN_D5	PIN_D4	PIN_K	PIN_E	PIN_RW	PIN_RS
0x00	FLG RESET	FLG SELF TEST	-	-	-	FLG I2C UP	-	-
0x01	SET RESET	SET SELF TEST	-	-	-	SET I2C UP	SAVE ADR EN	-
0x02 0x03	RESERVED							
0x04	MODEL[7-0]							
0x05	VERSION[7-0]							
0x06	ADDRESS[6-0]							SAVE_FLASH
0x07	CHIP_ID[7-0]							
0x08 --- 0x0F	RESERVED							

адрес	7	6	5	4	3	2	1	0
0x10					LIGHT[7-0]			
0x11					SYMBOL[7-0]			

Регистры с адресами 0x02, 0x03, 0x08 - 0x0F зарезервированы, их биты сброшены в 0. Попытка записи данных в эти регистры будет проигнорирована модулем.

Доступ к регистру без адреса «PCF8574» (указан на карте регистров до регистра с адресом 0x00) возможен только при нахождении переключателя режима работы адаптера в положении «ON» (режим работы с дисплеем).

Доступ к остальным регистрам (с адресами) возможен только при нахождении переключателя режима работы адаптера в положении «ADR» (режим адресного доступа к регистрам).

Регистр без адреса «PCF8574» - позволяет управлять выводами шины LCD:

Регистр для записи и чтения, доступен только в режиме работы адаптера «ON».

- **PIN_D7** - Бит управляет выводом данных «D7» шины LCD.
- **PIN_D6** - Бит управляет выводом данных «D6» шины LCD.
- **PIN_D5** - Бит управляет выводом данных «D5» шины LCD.
- **PIN_D4** - Бит управляет выводом данных «D4» шины LCD.
- **PIN_K** - Бит управляет подсветкой экрана (0-отключена, 1-включена).
- **PIN_E** - Бит управляет выводом тактирования «E» шины LCD.
- **PIN_RW** - Бит управляет выводом выбора направления передачи «RW» шины LCD.
- **PIN_RS** - Бит управляет выводом выбора регистра «RS» шины LCD.

Регистр имитирует работу чипа PCF8574. Если переключатель выбора режима работы адаптера находится в положении «ON», то передача данных по шине I2C осуществляется без указания адреса регистра и все получаемые/отправляемые по шине I2C байты будут адресованы данному регистру.

Каждый бит регистра позволяет управлять одним из выводов шины LCD, так как это реализовано в чипе PCF8574.

- При чтении регистра, каждый его бит соответствует логическому уровню, физически установленному (адаптером или дисплеем) на соответствующем выводе шины LCD.
- При записи в регистр, каждый сброшенный в «0» бит приводит к установке адаптером уровня логического «0» на соответствующем выводе шины LCD.
- При записи в регистр, каждый установленный в «1» бит приводит к переводу соответствующего вывода шины LCD со стороны адаптера в режим входа с подтяжкой до уровня 3,3 В.

Регистр 0x00 «FLAGS_0» - содержит флаги чтения состояния модуля:

Регистр только для чтения.

- **FLG_RESET** - Флаг указывает на факт выполнения успешной перезагрузки модуля. Флаг самостоятельно сбрасывается после чтения регистра [0x00 «FLAGS_0»](#).
- **FLG_SELF_TEST** - Флаг указывает на результат выполнения самотестирования модуля (0-провал, 1-успех). Не поддерживается данным модулем.
- **FLG_I2C_UP** - Флаг указывает на то, что модуль позволяет управлять подтяжкой линий шины I2C при помощи бита «SET_I2C_UP» регистра [0x01 «BITS_0»](#).

Регистр [0x01 «BITS_0»](#) - содержит биты установки состояния модуля:

Регистр для записи и чтения.

- **SET_RESET** - Бит запускает программную перезагрузку модуля. О завершении перезагрузки свидетельствует установка флага «FLG_RESET» регистра [0x00 «FLAGS_0»](#).
- **SET_SELF_TEST** - Бит запускает самотестирование модуля. При успешном завершении самотестирования устанавливается флаг «FLG_SELF_TEST» регистра [0x00 «FLAGS_0»](#). Не поддерживается данным модулем.
- **SET_I2C_UP** - Бит управляет внутрисхемной подтяжкой линий шины I2C. Значение бита сохраняется в FLASH память модуля. Установка бита в «1» приведёт к подтяжке линий SDA и SCL до уровня 3,3 В. На линии I2C допускается устанавливать внешние подтягивающие резисторы и иные модули с подтяжкой до уровня 3,3 В или 5 В, вне зависимости от состояния текущего бита. Если флаг «FLG_I2C_UP» регистра [0x00 «FLAGS_0»](#) сброшен, значит управление подтяжкой не поддерживается модулем.
- **SAVE_ADR_EN** - Бит разрешает записать новый адрес модуля для шины I2C в FLASH память. Бит самостоятельно сбрасывается после сохранения адреса во FLASH память. Запись адреса выполняется следующим образом: нужно установить бит «SAVE_ADR_EN», после чего записать новый адрес в регистр [0x06 «ADDRESS»](#) с установленным битом «SAVE_FLASH».

Регистр [0x04 «MODEL»](#) - содержит идентификатор типа модуля:

Регистр только для чтения.

- **MODEL[7-0]** - Для адаптера LCD дисплеев, I2C-flash - идентификатор равен 0x12.

Регистр [0x05 «VERSION»](#) - содержит версию прошивки модуля:

Регистр только для чтения.

- **VERSION[7-0]** - Версия прошивки (от 0x01 до 0xFF).

Регистр [0x06 «ADDRESS»](#) - отвечает за чтение/установку адреса модуля на шине I2C:

Регистр для чтения и записи.

- **ADDRESS[6-0]** - 7 бит адреса модуля на шине I2C. При чтении возвращается текущий адрес модуля, при записи устанавливается указанный адрес модулю. Допускается указывать адреса в диапазоне: $7 < \text{адрес} < 127$.
- **SAVE_FLASH** - Флаг записи адреса в FLASH память модуля.

Флаг имеет значение только при записи данных в регистр.

Если флаг сброшен, то адрес в битах ADDRESS[6-0] будет установлен временно (до отключения питания, или сброса/записи нового адреса). Если флаг установлен, то адрес в битах ADDRESS[6-0] будет сохранён в FLASH память модуля (останется и после отключения питания), но только если в бите «SAVE_ADR_EN» регистра [0x01 «BITS_0»](#) установлена логическая 1. Если флаг «SAVE_FLASH» установлен, а бит «SAVE_ADR_EN» сброшен, то адрес в битах ADDRESS[6-0] не будет установлен ни временно, ни постоянно.

Регистр 0x07 «CHIP_ID» - содержит идентификатор общий для всех модулей серии «Flash»:

Регистр только для чтения.

У всех модулей серии «Flash» в регистре «CHIP_ID» содержится значение 0x3C. Если требуется отличить модули серии «Flash» на шине I2C от сторонних модулей, то достаточно прочитать значение регистров [0x06 «ADDRESS»](#) и 0x07 «CHIP_ID» всех модулей на шине I2C. Если 7 старших битов регистра [0x06 «ADDRESS»](#) хранят адрес совпадающий с адресом модуля, а в регистре 0x07 «CHIP_ID» хранится значение 0x3C, то можно с большой долей вероятности утверждать, что данный модуль является модулем серии «Flash».

Регистр 0x10 «LIGHT» - содержит значение яркости подсветки LCD дисплея:

Регистр для чтения и записи.

- **LIGHT[7-0]** - Значение яркости подсветки дисплея от 0 (выключена) до 255 (максимальная). Значение по умолчанию 255.

Регистр 0x11 «SYMBOL» - служит защитой в режиме «ADR»:

Регистр для чтения и записи.

- **SYMBOL[7-0]** - Значение не влияющее на работу адаптера. Если переключатель выбора режима работы адаптера (случайно) находится в положении «ADR» (режим адресного доступа к регистрам), а по шине I2C (случайно) был отправлен пакет данных (текста) для вывода на дисплей, то коды всех полученных символов, или команды дисплея, будут попадать в данный регистр. Каждый новый байт будет заменять предыдущее значение, без перехода в следующий регистр.

Важно: Есть малая доля вероятности что при инициализации или выводе текста на дисплей, в режиме «ADR», часть данных может попасть в регистры [0x01 «BITS_0»](#), [0x06 «ADDRESS»](#), [0x10 «LIGHT»](#).

Доступ к данным регистров:

Каждый регистр модуля хранит 1 байт данных. Так как модуль использует интерфейс передачи данных I2C, то и доступ к данным охарактеризован им.

Обмен данными по шине I2C происходит по одному биту за один такт, после каждых

переданных 8 бит (1 байта) принимающее устройство отвечает передающему одним битом: «ACK» в случае успешного приёма, или «NACK» в случае ошибки. Пакет приёма/передачи данных начинается сигналом «START» и завершается сигналом «STOP». Первый байт пакета всегда состоит из 7 бит адреса устройства и одного (младшего) бита R/W.

Сигналы интерфейса передачи данных I2C:

Для удобства восприятия сигналов они выполнены в следующих цветах:

- Зелёный - сигналы формируемые мастером.
- Красный - данные отправляемые мастером.
- Синий - данные отправляемые модулем.
- Фиолетовый - данные отправляемые мастером или модулем.
- «START» - отправляется мастером в начале пакета приёма/передачи данных. Сигнал представляет переход уровня линии «SDA» из «1» в «0» при наличии «1» на линии «SCL».
- «STOP» - отправляется мастером в конце пакета приёма/передачи данных. Сигнал представляет переход уровня линии «SDA» из «0» в «1» при наличии «1» на линии «SCL».
- БИТ - значение бита считывается с линии «SDA» по фронту импульса на линии «SCL».
- «ACK» - бит равный 0, отправляется после успешного приёма байта данных.
- «NACK» - бит равный 1, отправляется после байта данных в случае ошибки.
- ПЕРВЫЙ БАЙТ - отправляется мастером, состоит из 7 бит адреса и бита «RW».
- «R/W» - младший бит первого байта данных указывает направление передачи данных пакета, 1 - прием (от модуля к мастеру), 0 - передача (от мастера в модуль).
- «RESTART» - повторный старт, отправляется мастером внутри пакета. Сигнал представляет из себя «START» отправленный не на свободной шине, а внутри пакета.

ВАЖНО: Все изменения на линии «SDA» должны происходить только при наличии «0» на линии «SCL» за исключением сигналов «START», «STOP» и «RESTART».

Запись данных в регистры (режим «ADR»):

- Отправляем сигнал «START».
- Отправляем первый байт: 7 бит адреса модуля и бит «R/W» равный 0 (запись).
Получаем ответ от модуля в виде одного бита «ACK».
- Отправляем второй байт: адрес регистра в который будет произведена запись.
Получаем ответ от модуля в виде одного бита «ACK».
- Отправляем третий байт: данные для записи в регистр.
Получаем ответ от модуля в виде одного бита «ACK».
- Далее можно отправить четвёртый байт данных для записи в следующий по порядку регистр и т.д.
- Отправляем сигнал «STOP».

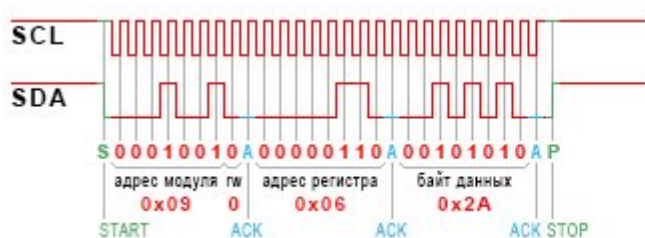
Запись данных в регистр (режим «ON» - режим работы с дисплеем):

Запись данных в режиме «ON» (режим работы с дисплеем) отличается тем, что после байта с адресом модуля сразу отправляется байт данных для записи в регистр (без отправки байта адреса

регистра). Все данные попадают в [регистр без адреса «PCF8574»](#).

Пример записи в один регистр (режим «ADR»):

Запись значения **0x2A** в регистр **0x06** модуля с адресом **0x09**:

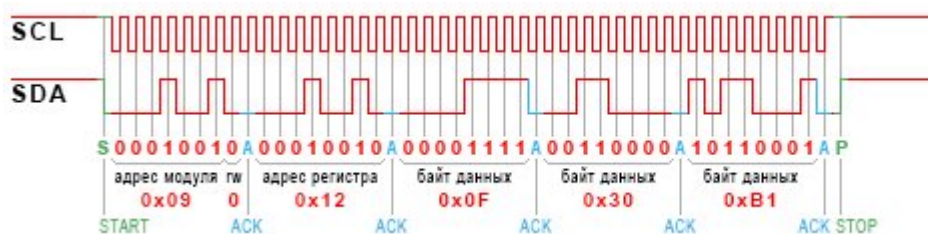


```
Wire.beginTransmission(0x09); // Запись в регистр методами библиотеки Wire.h
Wire.write(0x06); // Иницируем передачу данных в устройство с адресом 0
Wire.write(0x2A); // Записываем в буфер байт адреса регистра.
Wire.endTransmission(); // Записываем в буфер байт который будет записан в рег
// Выполняем передачу адреса и байтов из буфера. Функция
```

Пример записи в несколько регистров подряд (режим «ADR»):

Запись в модуль с адресом **0x09** нескольких значений начиная с регистра **0x12**:

В регистр **0x12** запишется значение **0x0F**, в следующий по порядку регистр (**0x13**) запишется значение **0x30** и в следующий по порядку регистр (**0x14**) запишется значение **0xB1**.



```
byte data[3] = {0x0F, 0x30, 0xB1}; // Запись в регистры методами библиотеки Wire.h
Wire.beginTransmission(0x09); // Определяем массив с данными для передачи.
Wire.write(0x12); // Иницируем передачу данных в устройство с адресом 0
Wire.write(data, 3); // Записываем в буфер байт адреса первого регистра.
Wire.endTransmission(); // Записываем в буфер 3 байта из массива data.
// Выполняем передачу адреса и байт из буфера. Функция
```

Чтение данных из регистров (режим «ADR»):

- При чтении пакет делится на 2 части: запись № регистра и чтение его данных.
- Отправляем сигнал «START».
- Отправляем **первый байт**: 7 бит адреса модуля и бит «R/W» равный 0 (запись).
Получаем ответ от модуля в виде одного бита «ACK».
- Отправляем **второй байт**: адрес регистра из которого нужно прочитать данные.
Получаем ответ от модуля в виде одного бита «ACK».

- Отправляем сигнал «RESTART».
- Отправляем **первый байт** после «RESTART»: 7 бит адреса и бит «R/W» равный 1 (чтение).
Получаем ответ от модуля в виде одного бита «ACK».
- Получаем **байт данных** из регистра модуля.
Отвечаем битом «ACK» если хотим прочитать следующий регистр, иначе отвечаем «NACK».
- Отправляем сигнал «STOP».

Чтение данных из регистра (режим «ON» - режим работы с дисплеем):

Чтение данных в режиме «ON» (режим работы с дисплеем) отличается тем, что после отправки сигнала «START» не отправляется байт адреса модуля (с битом R/W=0), не отправляется байт адреса регистра и не отправляется сигнал «RESTART». Все данные читаются из [регистра без адреса «PCF8574»](#).

Пример чтения одного регистра (режим «ADR»):

Чтение из модуля с адресом **0x09** байта данных регистра **0x05**:
(в примере модуль вернул значение **0x01**).



```

// Чтение регистра методами библиотеки Wire.h
byte data; // Объявляем переменную для чтения байта данных.
Wire.beginTransmission(0x09); // Инициуем передачу данных в устройство с адресом 0
Wire.write(0x05); // Записываем в буфер байт адреса регистра.
Wire.endTransmission(false); // Выполняем передачу без установки состояния STOP.
Wire.requestFrom(0x09, 1); // Читаем 1 байт из устройства с адресом 0x09. Функция
data=wire.read(); // Сохраняем прочитанный байт в переменную data.

```

Пример чтения нескольких регистров подряд (режим «ADR»):

Чтение из модуля с адресом **0x09** нескольких регистров начиная с регистра **0x05**:
(в примере модуль вернул значения: **0x01** из рег. 0x05, **0x13** из рег. 0x06, **0xC3** из рег. 0x07).



```

// Чтение регистров методами библиотеки Wire.h
byte data[3]; // Объявляем массив для чтения данных.
Wire.beginTransmission(0x09); // Иницируем передачу данных в устройство с адресом 0
Wire.write(0x05); // Записываем в буфер байт адреса регистра.
Wire.endTransmission(false); // Выполняем передачу без установки состояния STOP.
Wire.requestFrom(0x09, 3); // Читаем 3 байта из устройства с адресом 0x09. Функци
int i=0; // Определяем счётчик номера прочитанного байта.
while( Wire.available() ){ // Выполняем цикл while пока есть что читать из буфера
  if(i<3){ // Лучше делать такую проверку, чтоб не записать данны
    data[i] = wire.read(); i++; // Читаем очередной байт из буфера в массив data.
  } //
} //

```

Примечание:

- Если на линии I2C только один мастер, то сигнал «RESTART» можно заменить на сигналы «STOP» и «START».
- Рекомендуется не выполнять чтение или запись данных чаще 200 раз в секунду.

Обратите внимание на сигналы «RESTART» и «STOP» в пакетах чтения данных:

- Между фронтом и спадом сигнала «RESTART» проходит фронт импульса на линии «SCL», что расценивается как передача бита равного 1.
- Между сигналом «NACK» и сигналом «STOP» проходит фронт импульса на линии «SCL», что расценивается как передача бита равного 0.
- Эти биты не сохраняются в модулях и не расцениваются как ошибки.

Модуль не поддерживает горячее подключение: Подключайте модуль только при отсутствии питания и данных на шине I2C. В противном случае потребуется отключить питание при уже подключённом модуле.

Пример установки яркости подсветки экрана:

Следующий скетч демонстрирует пример установки 50% яркости подсветки экрана. До запуска скетча необходимо установить переключатель режима работы адаптера в положение «ADR» (режим адресного доступа к регистрам). После применения новой яркости, переключатель режима работы адаптера можно установить в положение «ON» (режим работы с дисплеем) и начать работу с дисплеем. Новое значение яркости сохранится и после отключения питания адаптера.

```

#include <Wire.h> // Подключаем библиотеку Wire для рабо
const int ADDRESS = 0x09; // Определяем адрес адаптера на шине I
const int REG_LIGHT = 0x10; // Определяем адрес регистра яркости п
int VAL_LIGHT = 127; // Определяем значение устанавливаемой
// (127 это 50% от максимального значе
void setup(){ //

```

```

// Иницируем работу с шиной I2C:
Wire.setClock(100000L);
Wire.begin();
// Устанавливаем яркость:
Wire.beginTransmission(ADDRESS);
Wire.write(REG_LIGHT);
Wire.write(VAL_LIGHT);
Wire.endTransmission();
}

void loop(){

```

Габариты:

Адаптер имеет размеры 41,91 x 17,15 мм = 1650 x 675 mil:

