

Применение MMC-карт в микроконтроллерных системах

Олег Пушкарёв (г. Омск)

В статье описывается протокол управления, внутренняя структура и система команд MMC-карт, приводится схема и программа для микроконтроллера, реализующего обмен данными с MMC-картой.

ВВЕДЕНИЕ

MMC-карты (MultiMedia Card) широко применяются в разнообразной цифровой технике в качестве носителя информации большой ёмкости. Внешний вид такой карты показан на рис. 1. Основная сфера применения – цифровые фотоаппараты, сотовые телефоны, видеокамеры, карманные компьютеры. Ёмкость MMC-карт достигает 4 Гб. Наиболее широко на рынке представлены карты ёмкостью 128, 256 и 512 Мб. MMC-карты ёмкостью 64 Мб ещё встречаются в прайс-листах, но их цена лишь на пару долларов ниже карт ёмкостью 128 Мб. Цены на MMC-карты постоянно снижаются, и сейчас розничная цена карты ёмкостью 128 Мб менее 400 руб. Основные параметры MMC-карт приведены в табл. 1.

MMC-карты являются привлекательным устройством энергонезависимой памяти также для проектировщиков микроконтроллерных встраиваемых систем. Большая ёмкость, низкое напряжение питания, энергонезависимость памяти и простой интерфейс делают MMC-карту оптимальным носителем в тех приборах, где требуется хранить большие объёмы информации, например, в устройствах сбора данных, речевых информаторах, справочных системах. Физически па-

мять MMC-карты разбита на сектора по 512 байт. MMC-карты, как правило, имеют файловую систему FAT16, хотя разработчик сам волен выбирать структуру хранения своей информации. Подробное описание протоколов обмена и всех команд MMC-карт можно найти в Интернете [1, 2].

ОПИСАНИЕ ПРОТОКОЛА ОБМЕНА И СИСТЕМЫ КОМАНД

MMC-карта представляет собой функционально и конструктивно законченный модуль, содержащий массив памяти и управляющий микроконтроллер (см. рис. 2). Обмен данными возможен по двум протоколам: MMC и SPI. Протокол MMC обеспечивает большую скорость и возможность параллельного включения нескольких карт и является основным. Однако для встраиваемых применений более подходящим является протокол SPI, аппаратная реализация которого встречается во многих микроконтроллерах. Даже без наличия аппаратного модуля SPI этот протокол элементарно реализуется на программном уровне. SPI-протокол в MMC-картах является вторичным протоколом обмена, который, однако, позволяет выполнять все операции по записи и чтению данных. SPI-протокол является последователь-



Рис. 1. Внешний вид MMC-карты

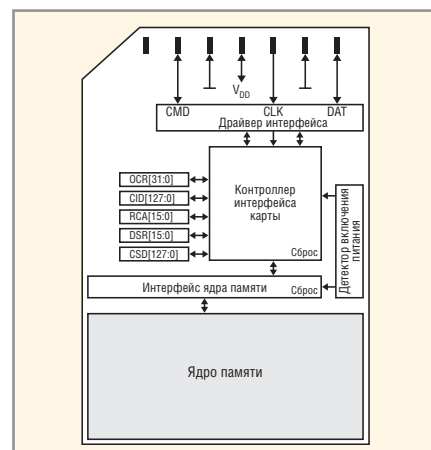


Рис. 2. Структурная схема MMC-карты

ным протоколом обмена данными по схеме ведущий/ведомый. Как и любое SPI-устройство, MMC-карта содержит четыре управляющие линии:

- CS – линия выбора MMC-карты (от микроконтроллера к MMC-карте);
- CLK – линия тактового сигнала (от микроконтроллера к MMC-карте);
- DataIn – линия принимаемых данных (от микроконтроллера к MMC-карте);
- DataOut – линия передачи данных (от MMC-карты к микроконтроллеру).

SPI-протокол является байт-ориентированным протоколом, т.е. передача данных идёт байтами и всегда синхронизирована с сигналом CS. Микроконтроллер выступает ведущим по отношению к MMC-карте, т.е. он является инициатором обмена и формирует тактовый сигнал. Передача байта идёт последовательно, по битам, начиная со старшего бита. Очередной бит при-

Таблица 1. Основные параметры MMC-карт

Параметры	Значение	Примечание
Размеры, мм (вес, г)	32 × 24 × 1,4 (1,5)	Есть уменьшенного размера (RS-MMC)
Ёмкость, Мб*	16, 32, 64, 128, 256, 512	
Скорость записи/чтения, Мб/с*	~3	Использовано оборудование: USB 2.0, карт-ридер, P4 – 2 ГГц
Тактовая частота, МГц	0...20	
Напряжение питания, В	2,7...3,6	В дальнейшем ожидается 1,8 В
Ток потребления, мА*	~40	При 3,6 В, в процессе записи/чтения
Размер сектора, байт	512	Файловая система – FAT16
Надёжность, циклов*	100 000	Цикл – запись и чтение
Рабочая температура, °С*	-25...+85	Температура хранения: -40...+85°С

* Данные для конкретного производителя/модели.



Рис. 3. Цоколёвка MMC-карты

нимается MMC-картой (и передаётся из MMC-карты) по положительному фронту тактового сигнала CLK («0» → «1»). Приём и передача байта происходит одновременно, т.е. когда микроконтроллер передаёт данные по линии DataIn, одновременно с этим принимаются данные по линии DataOut. Однако следует понимать, что не при каждой передаче принимаемые данные несут какую-либо информацию. Это свойство SPI-протокола, и микроконтроллер никак не использует поступающие в этот момент байты. В то же время, чтобы принять действительно актуальные данные (этот момент всегда известен, т.к. микроконтроллер полностью управляет обменом), микроконтроллер должен передать «пустой» байт (как правило, 0xFF). Ответом на передачу этого «пустого» байта и будут ожидаемые данные от MMC-карты.

Цоколёвка MMC-карты показана на рис. 3, а назначение выводов в режиме SPI приведено в табл. 2. Регистры MMC-карты (см. табл. 3) содержат различную информацию о параметрах карты – от рабочего напряжения питания до кодов ошибок текущей операции.

С точки зрения обмена данными MMC-карта принимает от микроконтроллера ряд команд, на которые она выдаёт либо ответы определённого типа, либо блоки данных. Ответ – R1, R2 или R3 – может состоять из одного, двух или пяти байтов. Формат обмена представлен на рис. 4. Блоки данных могут быть различной длины и состоят из стартового байта (0xFE), собственно данных (их длительность 1...N байт, где N определяется размером физического сектора, в большинстве случаев – 512 байт) и двух байт контрольной суммы. Контрольная сумма является опциональной в SPI-интерфейсе и, как правило, не исполь-

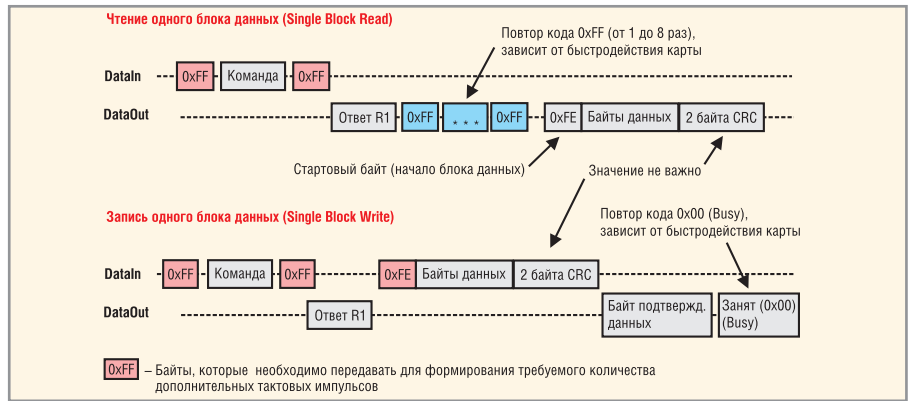


Рис. 4. Форматы обмена данными для разных команд

зуется для упрощения процедуры обмена. Значения двух байт контрольной суммы можно игнорировать, но сами эти байты должны обязательно передаваться/приниматься для соблюдения протокола обмена. Перед

передачей команды или после этого микроконтроллер должен выдавать не менее 8 тактовых импульсов по линии CLK, т.е. просто передавать «лишний» байт 0xFF. Значение сигнала CS в этот момент может быть как лог. 0, так и

Таблица 2. Назначение выводов MMC-карты в режиме SPI

Вывод	Наименование	Тип	Описание
1	CS	Вход	Выбор карты, активный «0» (Chip Select)
2	DataIn	Вход	Данные и команды от микроконтроллера
3	VSS1	Питание	Общий
4	VDD	Питание	Питание 2,7...3,6 В
5	CLK	Вход	Тактовый сигнал (Clock)
6	VSS2	Питание	Питание 2,7...3,6 В
7	DataOut	Выход	Данные и состояние (статус) от MMC-карты

Таблица 3. Регистры MMC-карты, доступные в режиме SPI

Имя регистра	Ширина, байт	Описание
CID	16	Идентификационные данные карты: серийный номер, ID производителя, дата производства, имя карты в виде ASCII-символов и т.п.
CSD	16	Информация об условиях работы: максимальная скорость доступа, максимальная длина блока данных, допустимые классы команд, ёмкость карты, максимальный ток потребления, возможность записи/чтения неполного блока данных и т.п.
OCR	4	Рабочее напряжение карты
STATUS	2	Причины ошибок (ошибки операции или неверно заданы параметры в команде)

Таблица 4. Некоторые команды MMC-карты, доступные в режиме SPI

Индекс	Аргумент	Ответ	Аббревиатура	Описание
CMD0	Нет	R1	GO_IDLE_STATE	Сброс
CMD1	Нет	R1	SEND_OP_COND	Запуск процесса инициализации карты
CMD9	Нет	R1	SEND_CSD	Запрос на чтение регистра CSD
CMD10	Нет	R1	SEND_CID	Запрос на чтение регистра CID
CMD12	Нет	R1	STOP_TRANSMISSION	Прекратить передачу (в процессе чтения нескольких блоков данных)
CMD13	Нет	R2	SEND_STATUS	Запрос на передачу регистра STATUS
CMD16	[31:0] длина блока	R1	SET_BLOCKLEN	Выбрать длину блока (в байтах) для последующих команд по записи/чтению блоков данных
CMD17	[31:0] адрес начала данных	R1	READ_SINGLE_BLOCK	Прочитать один блок данных (длина предварительно может быть установлена CMD16)
CMD18	[31:0] адрес начала данных	R1	READ_MULTIPLE_BLOCK	Последовательная передача блоков данных от MMC-карты (чтение) до момента получения команды СТОП
CMD24	[31:0] адрес начала данных	R1	WRITE_BLOCK	Записать блок данных (длина предварительно может быть установлена CMD16)
CMD25	[31:0] адрес начала данных	R1	WRITE_MULTIPLE_BLOCK	Последовательная передача блоков данных к MMC-карте (запись) до момента получения команды СТОП
CMD27	Нет	R1	PROGRAM_CSD	Программирование регистра CSD (только допустимые биты)
CMD58	Нет	R3	READ_OCR	Запрос на чтение регистра OCR

Таблица 5. Структура ответа R1

Бит	Имя	Описание
7	–	Всегда «0»
6	Ошибка параметра команды (Parameter error)	Аргумент команды (например, адрес, длина блока) выходит за пределы, допустимые для данной карты
5	Ошибка адреса (Address error)	Адрес в команде не соответствует длине блока (например, не выровнен относительно начала сектора)
4	Ошибка стирания (Erase Sequence error)	Ошибка в команде стирания данных
3	Ошибка контрольной суммы (Com CRC error)	Ошибка CRC при передаче команды
2	Неверная команда (Illegal command)	Неверный код команды
1	Ошибка стирания (Erase reset)	Ошибка операции стирания (прервана до исполнения)
0	Режим простоя (Idle State)	Карта в режиме простоя, выполняется инициализация

Таблица 6. Структура второго байта ответа R2

Бит	Имя	Описание
7	Вне диапазона/ошибка перезаписи (Out of range/CSD_overwrite)	Аргумент команды выходит за пределы, допустимые для данной карты, или попытка записи бита регистра CSD, не предназначенного для записи
6	Ошибка параметра (Erase param)	Неправильное задание сектора или группы при стирании
5	Защита записи (Write protect violation)	Попытка записи блока, защищённого от записи
4	Ошибка коррекции (Card ECC failed)	Ошибка внутреннего контроллера карты при попытке коррекции ошибки
3	Внутренняя ошибка (CC error)	Ошибка внутреннего контроллера карты
2	Ошибка (Error)	Общая или неизвестная ошибка в процессе операции
1	Защита от записи/ошибка пароля (Write protect erase skip/lock-unlock command failed)	Попытка стирания сектора, защищённого от записи, или ошибка пароля в процессе блокировки/разблокировки карты
0	Блокировка (Card is locked)	Карта заблокирована пользователем

Таблица 7. Формат байта подтверждения данных (data response)

Номер бита	7	6	5	4	3	2	1	0
Значение	x	x	x	0	Статус*		1	1

* Значение битов [3:1] статуса (Status):

010 – данные приняты;

101 – данные не приняты из-за ошибки CRC;

110 – данные не приняты из-за ошибки записи.

лог. 1. При чтении блока данных после передачи команды ЧТЕНИЕ микроконтроллер принимает байты 0xFF до тех пор, пока не встретится байт 0xFE (стартовый байт блока данных). Любой иной байт (отличный от 0xFF), полученный в этот момент, будет

означать ошибку – это может быть, например, байт ошибки данных.

Все команды, воспринимаемые MMC-картой, имеют длину 6 байт. Индекс команды (порядковый номер) находится в битах [5:0] первого байта команды; биты [7:6] всегда содержат «0» и «1» соответственно. Следующие 4 байта содержат аргумент команды, например, 32-битный адрес первого байта данных. Последний байт команды содержит в битах [7:1] контрольную сумму, бит [0] всегда равен «1». В табл. 4 приведён ряд команд для MMC-карты, работающей в режиме SPI. Подробное описание протоколов и всех команд MMC-карт можно найти в [3, 4].

Таблица 8. Формат стартового байта при обмене данными

Тип	Операция	Номер бита							
		7	6	5	4	3	2	1	0
Start Block	Чтение одиночного блока (Single Block Read)	1	1	1	1	1	1	1	0
	Чтение множества блоков (Multiple Block Read)	1	1	1	1	1	1	1	0
	Запись одиночного блока (Single Block Write)	1	1	1	1	1	1	1	0
	Запись множества блоков (Multiple Block Write)	1	1	1	1	1	1	1	0
Stop Tran	Запись множества блоков (Multiple Block Write)	1	1	1	1	1	1	0	1

Таблица 9. Формат байта ошибки данных (data error token)

Номер бита	7	6	5	4	3	2	1	0
Значение	0	0	0	Card is locked*	Out of range*	Card ECC failed*	CC error*	Error*

* Соответствуют битам второго байта ответа R2.

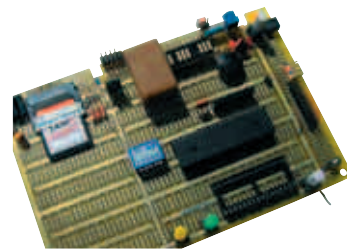


Рис. 5. Внешний вид макета устройства

MMC-карта после приёма команды выдаёт ответ, содержащий один, два или пять байт. Первым передаётся старший байт. Ответ формата R1 содержит один байт. Структура ответа приведена в табл. 5. Лог. 1 в бите означает наличие соответствующей ошибки или указанного состояния. Ответ R2 состоит из двух байт, причём первый байт ответа идентичен структуре ответа R1. Структура второго байта в ответе R2 приведена в табл. 6. Ответ R3 состоит из 5 байт. Первый байт идентичен ответу R1, остальные 4 байта представляют собой содержимое регистра OCR.

При записи данных в MMC-карту после получения блока данных карта отвечает байтом подтверждения данных. Формат этого байта представлен в табл. 7. Команды записи и чтения сопровождаются пересылкой блоков данных. Каждый блок данных начинается со стартового байта. Следующий за ним байт – это фактические данные двумя байтами контрольной суммы (16 бит CRC). Так как в режиме SPI контрольную сумму можно не вычислять, значения этих двух байтов не имеют значения, но сами байты контрольной суммы обязательны для передачи или чтения. Формат стартового байта представлен в табл. 8. Если операция чтения данных завершилась неудачно и карта не может предоставить запрашиваемые данные, то она будет посылать байт ошибки данных. Формат этого байта представлен в табл. 9.

После подачи напряжения питания MMC-карта находится в режиме MMC, а не в режиме SPI. Для перевода карты в режим SPI и инициализации карты необходимо выполнить определённую последовательность действий:

- не выбирая MMC-карту (CS = 1) послать 80 импульсов по линии CLK (передать 10 байт 0xFF);
- выбрать MMC-карту (CS = 0);
- послать команду CMD0 (сброс);
- послать шесть байт команды: 0x40,0,0,0,0,0x95 (в этой команде контрольная сумма должна иметь

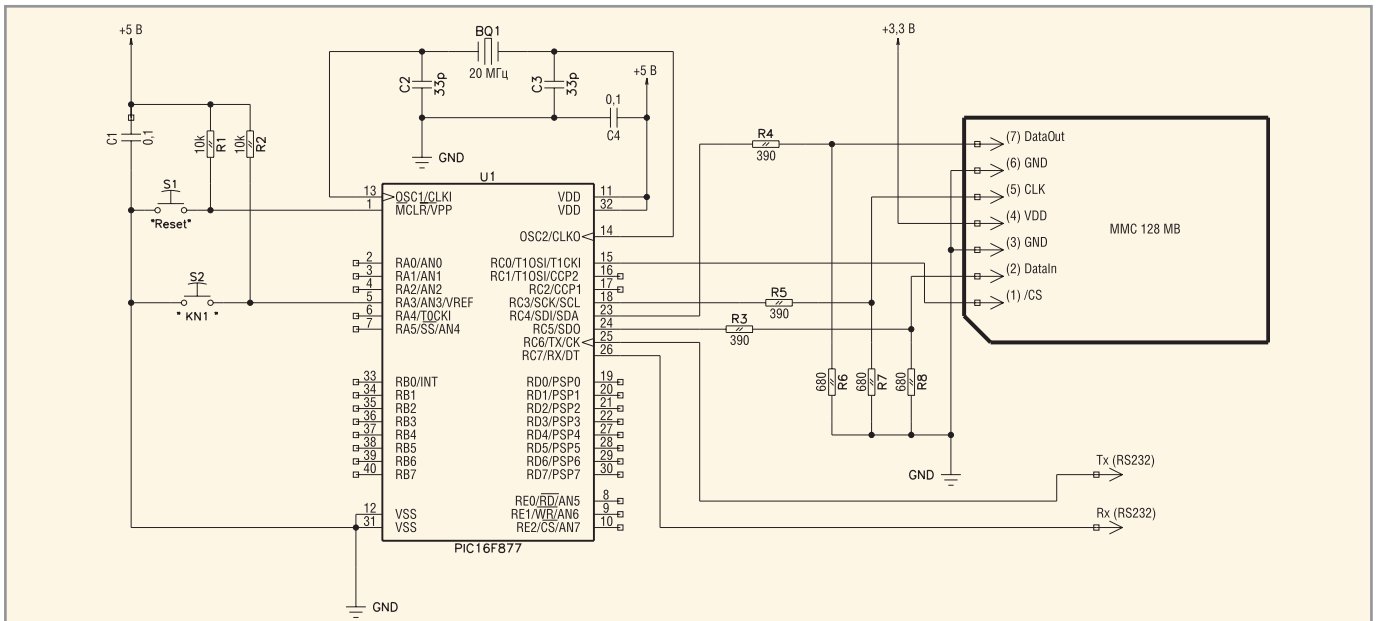


Рис. 6. Электрическая схема устройства

реальное значение (0x95), т.к. данная команда посылается в тот момент, когда MMC-карта находится в режиме MMC, а не SPI);

- дождаться правильного ответа 0x01;
- в цикле посылать команду CMD1 (инициализация) и ждать, когда будет получен ответ 0x00 (этот ответ означает, что карта инициализирована в режиме SPI и готова принимать команды).

ПРАКТИЧЕСКОЕ ПРИМЕНЕНИЕ

В качестве примера рассмотрим работу устройства (см. рис. 5), в котором организован обмен данными между микроконтроллером PIC16F877 и MMC-картой Kingston 128 Мб. Электрическая схема устройства приведена на рис. 6.

На процессоре PIC16F877 реализована тестовая программа (файл MMC128.ASM в архиве MMC128.rar), демонстрирующая процесс обмена информацией с картой MMC. Тактовая частота работы устройства 20 МГц, частота шины SPI – 5 МГц. На резисторах R3 – R8 собрана схема, обеспечивающая уровни 5 и 3,3 В. Обратите внима-

ние, что MMC-карта питается от отдельного источника напряжением 3,3 В. Информация о процессе обмена выдётся на внешний терминал по протоколу RS-232 на скорости 115 Кбод. Линия приёма данных не используется. Для подключения к СОМ-порту компьютера необходимо поставить преобразователь уровня на микросхеме MAX232 или её аналоге. Можно также применить интерфейсный СОМ-кабель от любого сотового телефона. В качестве программы-приёмника используется программа Telemax либо терминал, встроенный в РС-часть используемого в кристалле загрузчика PIC-Tiny-Bootloader [5]. После нажатия на кнопку Reset программа инициализирует MMC-карту, выводит на экран содержимое её внутренних регистров и отображает содержимое сектора 0. Далее во все 512 байт сектора 1 производится запись символа «*» и содержимое этого сектора выводится на экран. Затем в сектор 1 записываются байты с кодами от 0 до 255, и содержимое этого сектора вновь отображается на экране. После этого программа останавливается. По нажатию на кнопку KN1 начинается последовательное отображение на экране содержимого всех секторов MMC-карты. Приостановить вывод на экран можно кнопкой KN1. Пример работы программы приведён на рис. 7.

Программа построена таким образом, что ёмкость применяемой карты (в данном случае 128 Мб) не влияет на её работу. При последовательном отображении содержимого всех секторов при выходе за адрес последне-

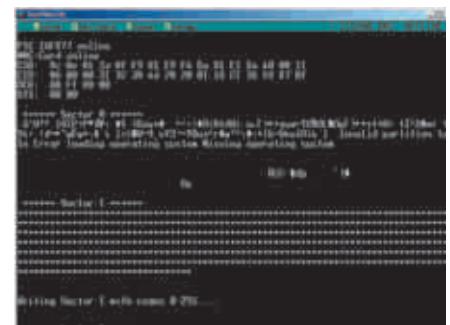


Рис. 7. Отображение результатов работы в программе Telemax

го физического сектора произойдёт ошибка чтения и на экран будет выведено соответствующее сообщение. Размер карты и количество секторов по 512 байт приведены в табл. 10.

Спецификация команд протокола MMC непрерывно развивается. Например, в версии 3.1 появились команды записи/чтения фиксированного количества секторов. Из MMC «выросли» такие стандарты, как SD (Secure Digital) и MMC-secure. Последние изменения в спецификациях MMC можно посмотреть на сайте [6].

ЛИТЕРАТУРА

1. www.sandisk.com/pdf/oem/ProdManualMMCv5.2.pdf.
2. www.vvm.com/~darkwing/PDF/mmcsversion_3-1.pdf.
3. www.sandisk.com/pdf/oem/ProdManualMMCv5.2.pdf.
4. www.vvm.com/~darkwing/PDF/mmcsversion_3-1.pdf.
5. www.etc.ugal.ro/cchiculita/software/pic-bootloader.htm.
6. www.mmca.org.

Таблица 10. Ёмкость MMC-карты и соответствующее число секторов

Ёмкость, Мб	Физическое число секторов	Адрес последнего сектора
16	31360	00 F4 FE 00
32	62720	01 E9 FE 00
64	125440	03 D3 FE 00
128	250880	07 A7 FE 00
256	501760	0F 4F FE 00
512	1003520	1E 9F FE 00