

CAN-контроллер фирмы Microchip

В сфере автоматизации производственных процессов сейчас происходят значительные перемены: пользователи постепенно отходят от практики применения закрытых архитектур и протоколов обмена собственной разработки, все больше ориентируясь на стандартные и открытые промышленные шины.

Одним из видов промышленной шины, проверенной временем, является CAN (Controller Area Network). Популярность данного протокола доказывается тем, что все большее число фирм начали внедрять контроллеры CAN в свои изделия. Среди них Motorola, Philips, Infineon technologies, Microchip, ST Microelectronics и другие. В данной статье пойдет речь об автономном CAN-контроллере MCP2510, выпущенном недавно фирмой Microchip. Отличительные черты данного изделия – управление по последовательному SPI интерфейсу и, как следствие этого, корпус с малым числом выводов. Учитывая низкую стоимость этой микросхемы (порядка \$3,5), можно рекомендовать данное изделие для новых разрабо-

ток. Читателям, не знакомым с самим протоколом CAN, рекомендуется вначале обратиться к литературе [1–5].

Основные возможности MCP2510

- Полная реализация CAN V2.0A и V2.0B при скорости 1 Мбит/с;
- длина сообщения до 8 байт;
- стандартный и расширенный фреймы;
- программируемая скорость передачи до 1 Мбит/с;
- поддержка удаленных фреймов;
- два буфера приема с приоритетным доступом;
- шесть фиксированных фильтров для приема сообщений;
- два фильтра-маски для приема сообщений;
- три буфера приема с функциями приоритетности и возможности прерывания передачи;
- петлевой режим для самотестирования.
- Аппаратные возможности:
 - высокоскоростной SPI интерфейс (5 МГц при напряжении питания 4,5 В);
 - поддержка режимов 0 и 3 SPI;
 - выход тактовой частоты с предварительным делителем;
 - выход прерывания с разрешением (маской) по разным источникам прерываний;
 - выводы “Буфер заполнен”, конфигурируемые как выходы прерывания либо как цифровые выходы общего назначения;
 - выводы “Запрос передачи”, конфигурируемые как входы, управляющие запросом на посылку сообщения

индивидуально для каждого буфера передачи, либо как цифровые входы общего назначения;

- режим пониженного энергопотребления (Sleep).
- Малопотребляющая КМОП-технология:
- рабочее напряжение питания 3...5,5 В;
- ток потребления в режиме Standby 10 мкА при напряжении питания 5,5 В.
- 18-выводной PDIP/SOIC и 20-выводной TSSOP корпуса.
- Температурный диапазон: расширенный (E): –40°C... +125°C; промышленный (I): –40°C... +85°C.

Описание

MCP2510 является автономным CAN-контроллером, разработанным с целью упростить приложения, требующие интер-

фейса с промышленной шиной CAN. Типичная реализация системы, использующей эту микросхему, приведена на рис. 1.

На рис. 2 представлена блок-схема данного устройства. В него входят следующие основные узлы:

- автомат CAN-протокола;
- логика управления и регистровое ОЗУ, используемые для конфигурации устройства и задания режимов его работы;
- блок SPI-протокола.

Автомат CAN-протокола обрабатывает все функции, связанные с приемом и передачей сообщений по шине. Для передачи сообщения требуется вначале загрузить соответствующие буферный и

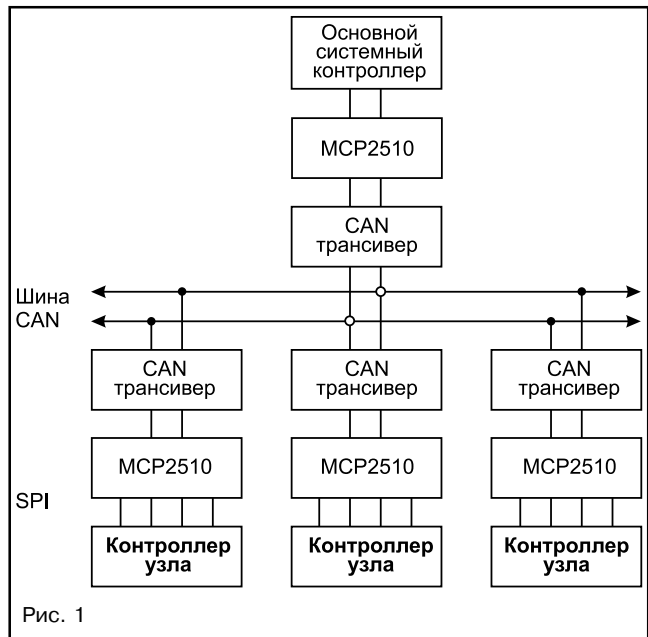


Рис. 1

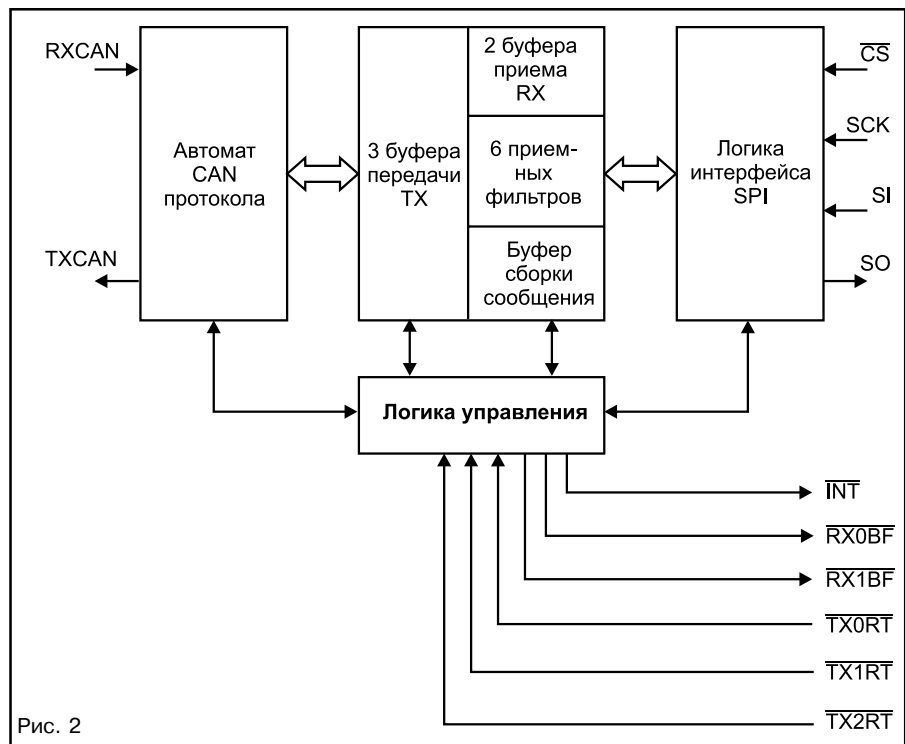


Рис. 2

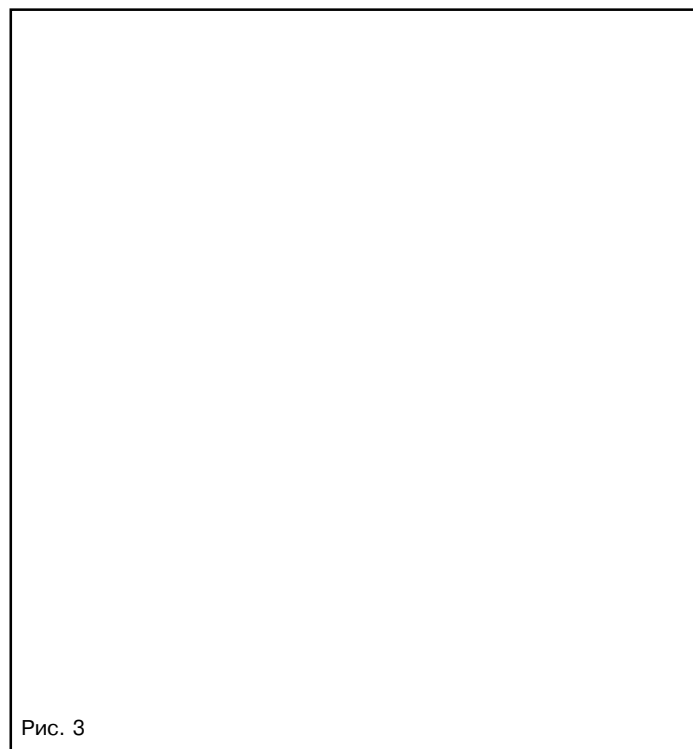


Рис. 3

управляющий регистры. Передача инициируется либо установкой бит в соответствующих регистрах управления через интерфейс SPI, либо подачей логического 0 на выходы микросхемы TX0RTS, TX1RTS, TX2RTS, которые в этом случае должны быть сконфигурированы для реализации данной функции, а не как цифровые входы общего назначения. Состояние микросхемы и ошибки могут

ответствующих регистров. Любое передаваемое по шине сообщение проверяется на ошибки и затем сравнивается с заданными пользователем фильтрами для определения возможности загрузки в один из двух буферов приема. Для большей гибкости при построении системы имеются выходы прерывания. Один вывод прерывания (INT) – множественного назначения, отражающий

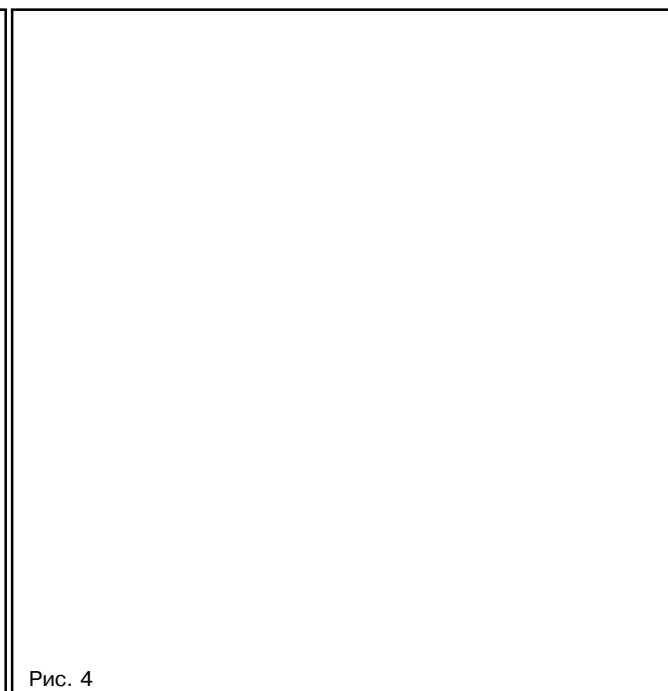


Рис. 4

быть проверены чтением со-

сразу несколько возможных ситуаций. Два других вывода прерывания (RX0BF, RX1BF) могут быть использованы для индикации факта принятия действительного сообщения и помещения его в соответствующий буфер приема. Данная функция выводов также должна быть задана при конфигурировании. Иначе их можно использовать как цифровые выходы общего назначения, а факт принятия действительного сообщения определять чтением регистра

Таблица 1

Имя вывода	Номер вывода DIP/SOIC	Номер вывода TSSOP	Тип вывода	Функция
TXCAN	1	1	O	Выход передатчика к трансиверу CAN
RXCAN	2	2	I	Вход приемника от трансивера CAN
CLKOUT	3	3	O	Выход тактовой частоты после предварительного делителя
TX0RTS	4	4	I	Вход «Запрос передачи буфера 0» (TXB0), либо цифровой вход общего назначения
TX1RTS	5	5	I	Вход «Запрос передачи буфера 1» (TXB1), либо цифровой вход общего назначения
TX2RTS	6	7	I	Вход «Запрос передачи буфера 2» (TXB0), либо цифровой вход общего назначения
OSC2	7	8	O	Выход генератора частоты (подключение кварцевого резонатора)
OSC1	8	9	I	Вход генератора частоты (подключение кварцевого резонатора)
VSS	9	10	P	Общий
RX1BF	10	11	O	Выход прерывания от буфера приема 1 (RXB1) или цифровой выход общего назначения
RX0BF	11	12	O	Выход прерывания от буфера приема 2 (RXB2) или цифровой выход общего назначения
INT	12	13	O	Выход прерывания от различных источников
SCK	13	14	I	Вход тактовой частоты интерфейса SPI
SI	14	16	I	Вход данных интерфейса SPI
SO	15	17	O	Выход данных интерфейса SPI
CS	16	18	I	Вход «Выбор кристалла» интерфейса SPI
RESET	17	19	I	Вход сброса (активный низкий уровень)
VDD	18	20	P	Напряжение питания
NC	–	6, 15	–	Не подсоединены

Обозначения: I = Вход; O = Выход; P = Питание

состояния по SPI. В таблице 1 представлен полный список выводов CAN-контроллера.

Буферы приема/передачи

MCP2510 имеет три передающих и два приемных буфера, два фильтра-маски (по одному на каждый буфер приема) и шесть фиксированных фильтров по приему сообщений. На рис. 3 изображены эти буферы и их связь с автоматом CAN-протокола.

Автомат CAN-протокола (CAN Protocol Engine)

Автомат CAN-протокола объединяет несколько функциональных блоков, показанных на рис. 4. Эти блоки и их функции описываются ниже.

Конечный автомат протокола (Protocol Finite State Machine)

Центральной частью (сердцем) автомата CAN-протокола является конечный автомат. Он производит обработку сообщений на побитовом уровне, изменяя свои состояния в соответствии с полями различных типов фреймов. Конечный автомат управляет последовательным потоком данных между сдвиговым регистром TX/RX, регистром контрольной суммы CRC и шиной. Он также управляет логикой

обработки ошибок (EML) и параллельным потоком данных между сдвиговыми регистрами TX/RX и буферами. Реализация CAN в качестве конечного автомата гарантирует, что процессы приема, арбитража, передачи, сигнализации об ошибках осуществляются в точном соответствии с протоколом. Автоматическая повторная передача сообщений по шине также возлагается именно на него.

Проверка контрольной суммы (Cyclic Redundancy Check)

Регистр контрольной суммы генерирует CRC-код, который передается либо после управляющего поля (при отсутствии данных в сообщении), либо после поля данных, а также используется при проверке поля контрольной суммы входящих сообщений.

Логика управления ошибками (Error Management Logic)

Данный блок отвечает за ограничение неполадок CAN-устройства. Его два счетчика ошибок (приема – Receive Error Counter – и передачи – Transmit Error Counter) инкрементируются и декрементируются по командам процессора битового потока. В соответствии со значениями счетчиков ошибок CAN-контроллер будет на-

ходиться в следующих состояниях: ошибка активная, ошибка пассивная, отключен от шины.

Логика бит-тайминга (Bit Timing Logic)

Данный блок контроллера постоянно отслеживает вход CAN-шины, осуществляет синхронизацию старта фрейма по перепаду с рецессивного уровня к доминантному (жесткая синхронизация) и синхронизируется по любому дальнейшему перепаду с рецессивного уровня к доминантному, если, конечно, сам CAN-контроллер не передает доминантный бит (ресинхронизация). Имеется возможность программировать временные сегменты, составляющие битовый промежуток, для компенсации задержки распространения сигнала и фазовых сдвигов. Также возможно программирование позиции (и количества) точек выборки значения бита внутри его битового промежутка.

В заключение хотелось бы отметить, что в рамках статьи не имеет смысла давать детальное описание регистров, вплоть до побитового уровня. Полное фирменное руководство занимает 76 страниц и заинтересованные читатели смогут ознакомиться с ним самостоятельно.

Игорь Лапшин,
gkb_luch@mail.ru

Литература

1. Журнал “Современные технологии автоматизации” №4/98, с. 16.
2. Журнал “Современные технологии автоматизации” №3/99, с. 6.
3. Журнал “Инженерная микроэлектроника” №2/98, с. 35.
4. Журнал “Chip News” №5/99, с. 6.