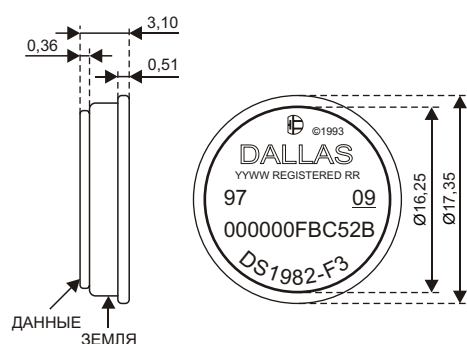
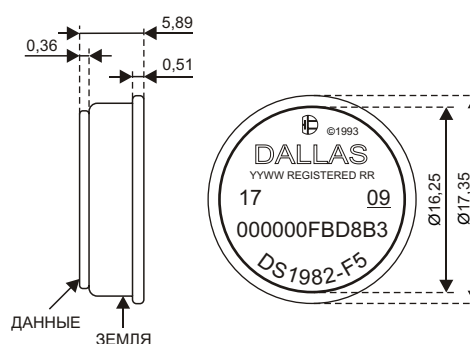


СПЕЦИАЛЬНЫЕ ВОЗМОЖНОСТИ

- Электрически программируемая память (EPROM) объемом 1024 бита с доступом по одному сигнальному проводу плюс земля
- EPROM разбита на четыре страницы по 256 бит для произвольного доступа к данным, организованным в виде пакетов
- Каждая страница памяти может быть полностью защищена от записи для предотвращения несанкционированного изменения данных
- Дополнительные данные записываются в EPROM без разрушения имеющихся данных
- Архитектура позволяет программе обновлять данные путем замены старой страницы вновь запрограммированной страницей
- Сигналы управления, адреса, данных, питания и программирования сведены к единственному выводу данных
- 8-битный код семейства определяет для считывателя требования к обмену с DS1982
- Считывание возможно в диапазоне напряжений питания от 2,8В до 6,0В и температур от -40°C до +85°C; программирование возможно в диапазоне напряжений от 11,5В до 12,0В и температур от -40°C до +50°C
- Встроенный контроллер многоточечной сети MicroLAN
- Цифровая идентификация и получение информации в одно касание
- Компактный носитель информации на кристалле микросхемы
- Данные могут быть доступными при касании объекта
- Экономичный обмен с мастером шины с помощью единственного цифрового сигнала на скорости 16,3 Кбит в секунду
- Стандартный диаметр 16 мм и 1-проводный протокол (1-Wire®) гарантируют совместимость с семейством iButton
- Форма в виде таблетки обеспечивает автоматическое центрирование в считывающем устройстве
- Долговечный корпус из нержавеющей стали с гравированным регистрационным номером устойчив к внешним воздействиям
- Легко прикрепляется с помощью самоклеющейся подложки, фиксируется собственным фланцем или напрессовываемым кольцом
- Детектор присутствия выдает ответ, когда считыватель в первый раз подает напряжение питания
- Соответствует UL#913 (4-я редакция); взрывобезопасное исполнение, утверждено для использования в классе I, отделение 1, группы A, B, C и D (в зависимости от приложения)

ОБЩИЕ ВОЗМОЖНОСТИ iButton

- Уникальный, занесенный лазером на этапе изготовления и проверенный 64-битный регистрационный номер (8-битный код семейства + 48-битный серийный номер + 8-битная контрольная сумма CRC) гарантирует абсолютный контроль, так как не существует двух устройств с одинаковыми номерами

F3 MICROCAN**F5 MICROCAN**

Все размеры даны в миллиметрах

ИНФОРМАЦИЯ ДЛЯ ЗАКАЗА

DS1982-F3 корпус F3 MicroCan
 DS1982-F5 корпус F5 MicroCan

ПРИМЕРЫ АКСЕССУАРОВ

DS9096P Самоклеющаяся подложка
 DS9101 Универсальный зажим
 DS9093RA Крепежное кольцо
 DS9093F Держатель с защелкой
 DS9092 Панелька для iButton

ОПИСАНИЕ iButton

Обладая возможностью записи/чтения, 1-Кбит iButton типа DS1982 является надежным носителем данных, который идентифицирует и хранит информацию, относящуюся к предмету или человеку, за которым он закреплен. Эта информация может быть доступна с помощью минимального аппаратного обеспечения, например, с помощью единственного вывода порта микроконтроллера. DS1982 содержит занесенный с помощью лазера при изготовлении регистрационный номер, включающий уникальный 48-битный серийный номер, 8-битную CRC и 8-битный код семейства (09h), а также EPROM объемом 1 Кбит, которая может программироваться пользователем. Питание для программирования и чтения поступает непосредственно с 1-проводной линии обмена. Данные передаются последовательно с помощью 1-проводного протокола, для которого требуется только один вывод данных и земля. Устройство может быть полностью запрограммировано, а затем, если требуется, защищено от записи. Как вариант, устройство может программироваться много раз вновь поступающими данными, которые можно добавлять при каждом последующем программировании, но нельзя перезаписывать уже существующие данные. Заметьте: состояние отдельных бит можно изменять с логической 1 в логический 0, но не наоборот. Существует возможность указать, что данные в некоей странице или в нескольких страницах уже не являются действительными и заменены новыми или обновленными данными, которые находятся в странице с другим адресом. Такое перенаправление адреса страницы позволяет программному обеспечению обновлять данные и увеличивает гибкость устройства как автономной базы данных. Серийный 48-битный номер, записанный лазером в каждый экземпляр DS1982 при изготовлении, обеспечивает гарантированную идентификацию, что позволяет осуществлять абсолютный контроль. Долговечный корпус MicroCan исключительно устойчив к агрессивным внешним условиям, таким как грязь, влажность и удары. Его компактный профиль в форме таблетки автоматически центрирует прибор в считывающем устройстве, позволяя пользователям или автоматизированному оборудованию легко им оперировать. Аксессуары позволяют монтировать DS1982 на печатную плату, в пластиковый держатель, идентификационные бэджи, браслеты и на множество других предметов. Применения включают

контроль стадий производственных процессов, электронный контроль перевозок, ограничение доступа, хранение калибровочных коэффициентов и осуществление электронных платежей.

ОБЗОР

Блок-схема, показанная на рис. 1, демонстрирует связи между блоками управления и блоками памяти DS1982. Всего в DS1982 имеются три основных компонента хранения данных: 1) 64-битное записанное лазером ПЗУ, 2) 1024-битная EPROM и 3) байты состояния EPROM. При осуществлении чтения устройство питается непосредственно от 1-проводной линии обмена, сохраняя энергию во внутреннем конденсаторе в те периоды времени, когда линия обмена находится в состоянии высокого логического уровня, и продолжая работать на этом «паразитном» источнике энергии, когда линия находится в состоянии низкого уровня. Возвращаясь в состояние высокого уровня, линия пополняет паразитный источник энергии (конденсатор). Во время программирования 1-проводная линия обмена на некоторое время переходит из состояния нормального напряжения в состояние напряжения программирования, что вызывает запись выбранных бит EPROM. Для программирования EPROM требуется, чтобы 1-проводная линия обеспечивала напряжение 12В и ток 10мА. Когда на линии присутствует напряжение программирования, специальная схема детектора высокого напряжения вырабатывает внутренний логический сигнал.

Иерархическая структура 1-проводного протокола показана на рис. 2. Мастер шины должен вначале послать одну из четырех команд функций ПЗУ: 1) Чтение ПЗУ, 2) Сравнение ПЗУ, 3) Поиск ПЗУ, 4) Пропуск ПЗУ. Эти команды работают с 64-битным ПЗУ, записанным лазером, которое есть в каждом устройстве. Они позволяют выделить отдельное устройство, когда на 1-проводной линии представлено несколько устройств. Они также дают возможность мастеру узнать, сколько устройств подключено к 1-проводной линии, и какого они типа. Протокол, который требуется для передачи этих команд функций ПЗУ, показан на рис. 9. После того, как команда функции ПЗУ успешно выполнена, могут подаваться команды функций памяти, которые работают с EPROM DS1982. Мастер шины может передать одну из пяти команд функций памяти, которые специфичны для DS1982 и позволяют считывать или записывать различные поля данных. Протокол для команд функций памяти показан на рис. 6. При считывании и записи всех данных первым передается младший бит.

64-БИТНОЕ ПЗУ, ЗАПИСАННОЕ ЛАЗЕРОМ

Каждый экземпляр DS1982 содержит в ПЗУ уникальный код длиной 64 бита. Первые 8 бит являются кодом семейства. Следующие 48 бит являются уникальным серийным номером. Последние 8 бит являются контрольной суммой (CRC) первых 56 бит (см. рис. 3). 64-битное ПЗУ и блок управления ПЗУ позволяют DS1982 работать как 1-проводному устройству и следовать 1-проводному протоколу, подробно описанному в разделе «1-проводная шина». Функции памяти, требующиеся для чтения и программирования EPROM, недоступны, пока не будет выполнен протокол функции ПЗУ. Этот протокол показан на блок-схеме функций ПЗУ (рис. 9). Мастер 1-проводной шины в первую очередь должен подать одну из четырех команд функций ПЗУ: 1) Чтение ПЗУ, 2) Сравнение ПЗУ, 3) Поиск ПЗУ, 4) Пропуск ПЗУ. После того, как команда функции ПЗУ будет успешно выполнена, мастер шины может подавать одну из команд функций памяти, специфичных для DS1982 (рис. 6).

Контрольная сумма (CRC) ПЗУ, записанного лазером, получена с помощью полинома $X^8 + X^5 + X^4 + 1$. Дополнительную информацию о контрольной сумме, используемой фирмой Dallas Semiconductor, можно найти в книге «Book of DS19xx iButton Standards». Сдвиговый регистр, работающий как аккумулятор CRC, инициализируется нулем. Затем, начиная с младшего бита кода семейства, по одному биту задвигаются данные. После ввода 8-го бита кода семейства вводятся биты серийного номера. После ввода 48-го бита серийного номера сдвиговый регистр содержит значение CRC. Если ввести еще 8 бит CRC, то содержимое регистра вновь станет равным нулю.

Рис. 1. БЛОК-СХЕМА DS1982.

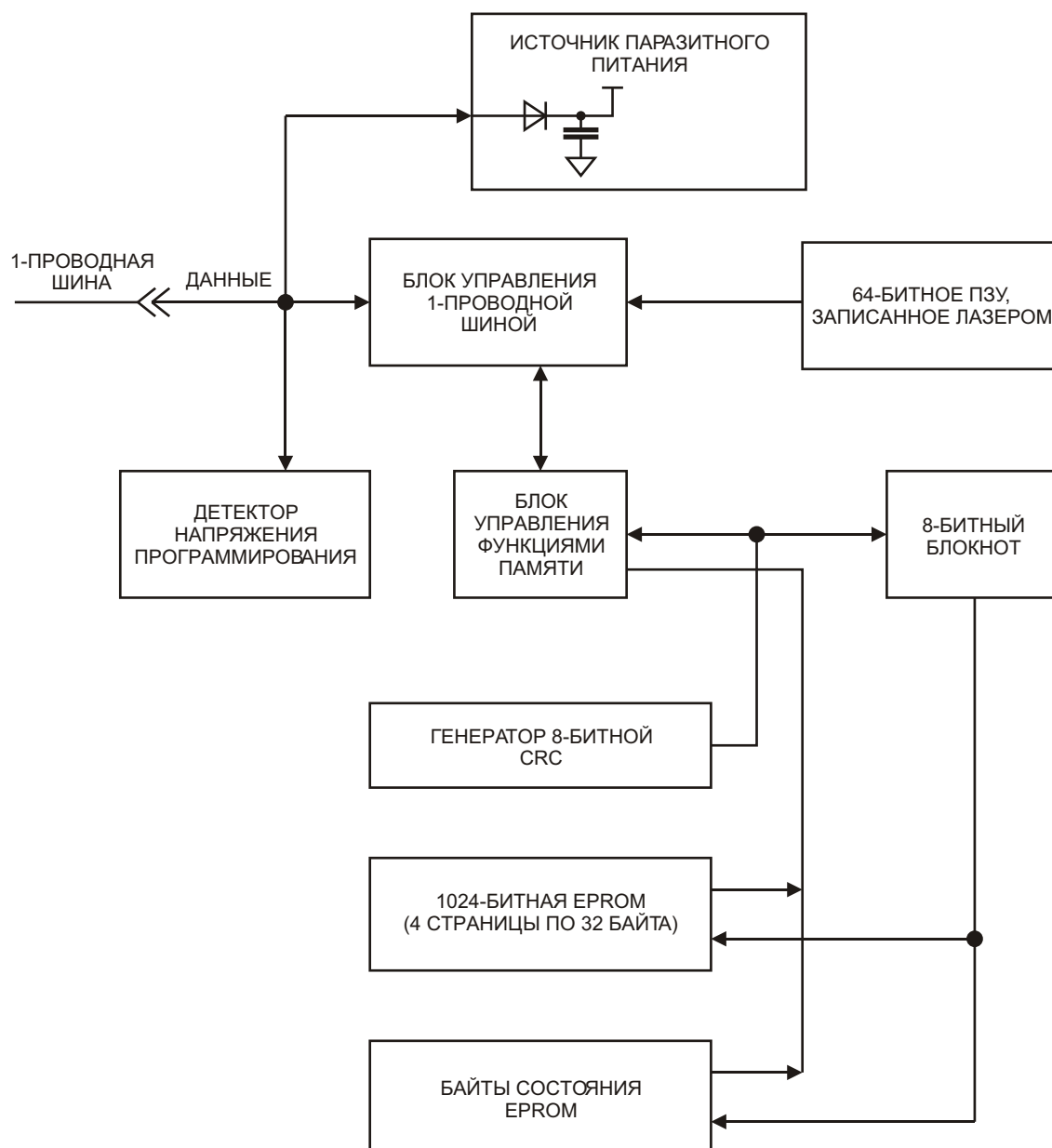


Рис. 2. ИЕРАРХИЧЕСКАЯ СТРУКТУРА 1-ПРОВОДНОГО ПРОТОКОЛА.

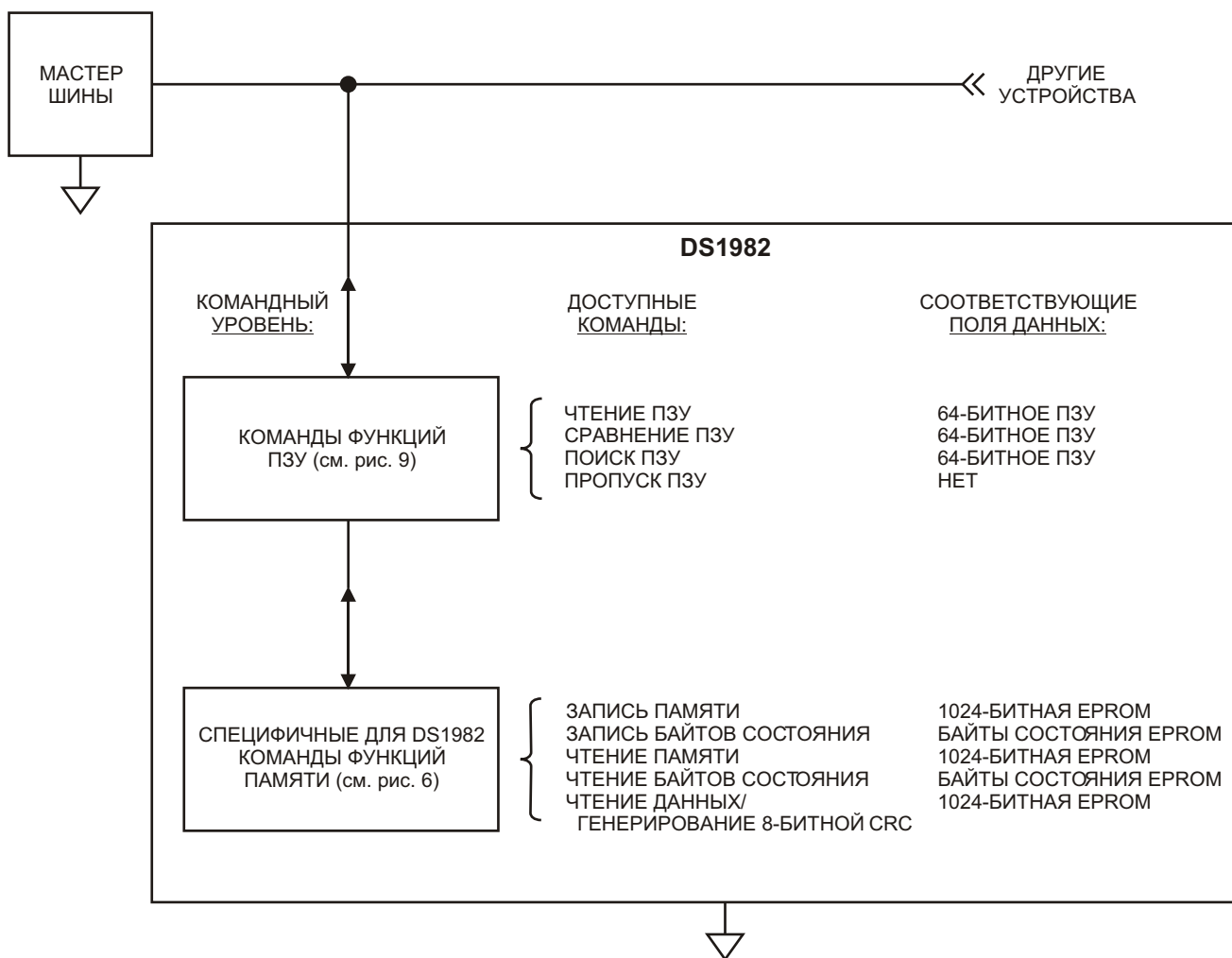


Рис. 3. 64-БИТНОЕ ПЗУ, ЗАПИСАННОЕ ЛАЗЕРОМ.

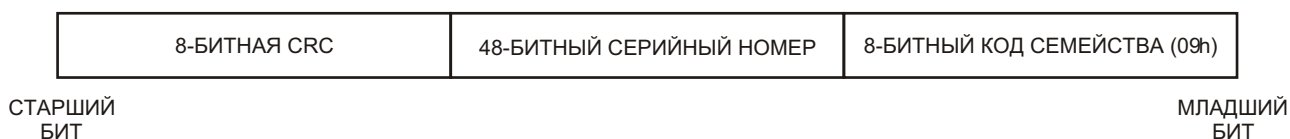
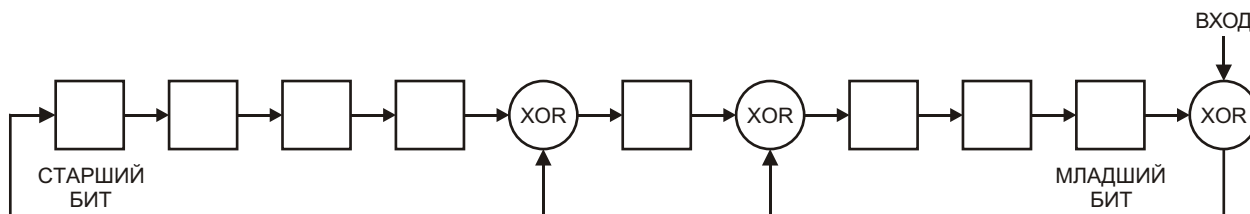


Рис. 4. ГЕНЕРАТОР CRC.



1024-БИТНАЯ EPROM

Карта памяти, показанная на рис. 5, изображает 1024-битную EPROM DS1982, которая сконфигурирована как четыре страницы по 32 байта в каждой. 8-битный блокнот является дополнительным регистром, который работает как буфер при программировании памяти. Данные вначале записываются в блокнот, затем проверяются путем считывания 8-битной CRC из DS1982, что служит подтверждением их правильного приема. Если содержимое буфера правильное, подается напряжение программирования, и байт данных записывается в память по выбранному адресу. Этот процесс гарантирует целостность данных при программировании памяти. Подробности чтения и программирования 1024-битной EPROM DS1982 приведены в разделе «Команды функций памяти».

БАЙТЫ СОСТОЯНИЯ EPROM

В дополнение 1024 битам памяти данных, DS1982 содержит 64 бита состояния, которые доступны с помощью отдельных команд.

Байты состояния EPROM могут быть считаны или записаны для индикации различных состояний DS1982, которые затем могут быть определены программно. Первый байт памяти состояния EPROM содержит биты защиты страниц от записи. Если некоторые биты защиты от записи запрограммированы, запрещается программирование соответствующих страниц в 1024-битной основной области памяти. Будучи однажды запрограммированным, один бит защиты от записи делает 32-байтную страницу, соответствующую этому биту, недоступной для дальнейших изменений, но ее по-прежнему можно считывать.

Следующие 4 байта в памяти состояния EPROM представляют собой байты перенаправления адреса страницы, которые показывают, что одна (или более) страница данных в 1024-битной EPROM была объявлена недействительной и перенаправлена на адрес страницы, который содержится в соответствующем байте перенаправления. Аппаратно DS1982 не производит никаких действий, зависящих от содержимого байтов перенаправления адресов страниц. Эти дополнительные байты состояния EPROM позволяют программе перенаправить страницу целиком на другой адрес страницы, показывая, что данные в оригинальной странице больше не являются действительными. Из-за особенностей технологии EPROM состояние битов внутри страницы при программировании может быть изменено с логической 1 на логический 0, но не может быть изменено обратно. Поэтому невозможно просто перезаписать страницу, если данные требуют изменения или обновления. Но если позволяет место, страница целиком может быть перенаправлена на другую страницу DS1982 путем записи инверсного значения нового адреса страницы в байт перенаправления адреса, который соответствует оригинальной (замещаемой) странице.

Данная архитектура позволяет пользовательскому программному обеспечению производить обновление данных в EPROM путем указания того, что отдельная страница или страницы должны быть замещены другими, указанными в байтах перенаправления адреса.

Если байт перенаправления адреса содержит значение FFh, то данные в соответствующей странице основной памяти являются действительными. Если байт перенаправления адреса содержит любое другое значение, то данные в соответствующей странице являются недействительными, а действительные данные теперь можно найти по адресу, инверсному записанному в байт перенаправления адреса. Значение FDh байта перенаправления адреса для страницы 1, например, будет показывать, что обновленные данные теперь находятся в странице 2. Подробности считывания и программирования памяти состояния EPROM DS1982 приведены в разделе описания команд функций памяти.

КОМАНДЫ ФУНКЦИЙ ПАМЯТИ

Блок-схема функций памяти (рис. 6) описывает протоколы, необходимые для доступа к различным полям данных DS1982. Блок управления памятью, 8-битный блокнот и схема детектора

напряжения программирования совместно интерпретируют команды, получаемые от мастера шины, и вырабатывают необходимые сигналы управления внутри устройства. Мастер использует 3-байтный протокол. Он включает байт команды, который определяет тип операции, и 2 байта адреса, которые определяют начальное положение байта в поле данных. Байт команды показывает, требуется чтение устройства или запись. Записываемые данные должны включать не только корректную последовательность команд, но и в нужный момент времени сопровождаться напряжением программирования 12В. Для выполнения процесса записи вначале байт данных загружается в блокнот, и лишь затем записывается по выбранному адресу. Запись всегда производится по одному байту. При выполнении процесса чтения мастер передает начальный адрес, и данные считываются из устройства с начального адреса до конца выбранного поля данных или до появления импульса сброса. Во всех пересылках, передаваемых в DS1982 и принимаемых мастером шины, вначале передается младший бит.

Рис. 5. КАРТА ПАМЯТИ DS1982.

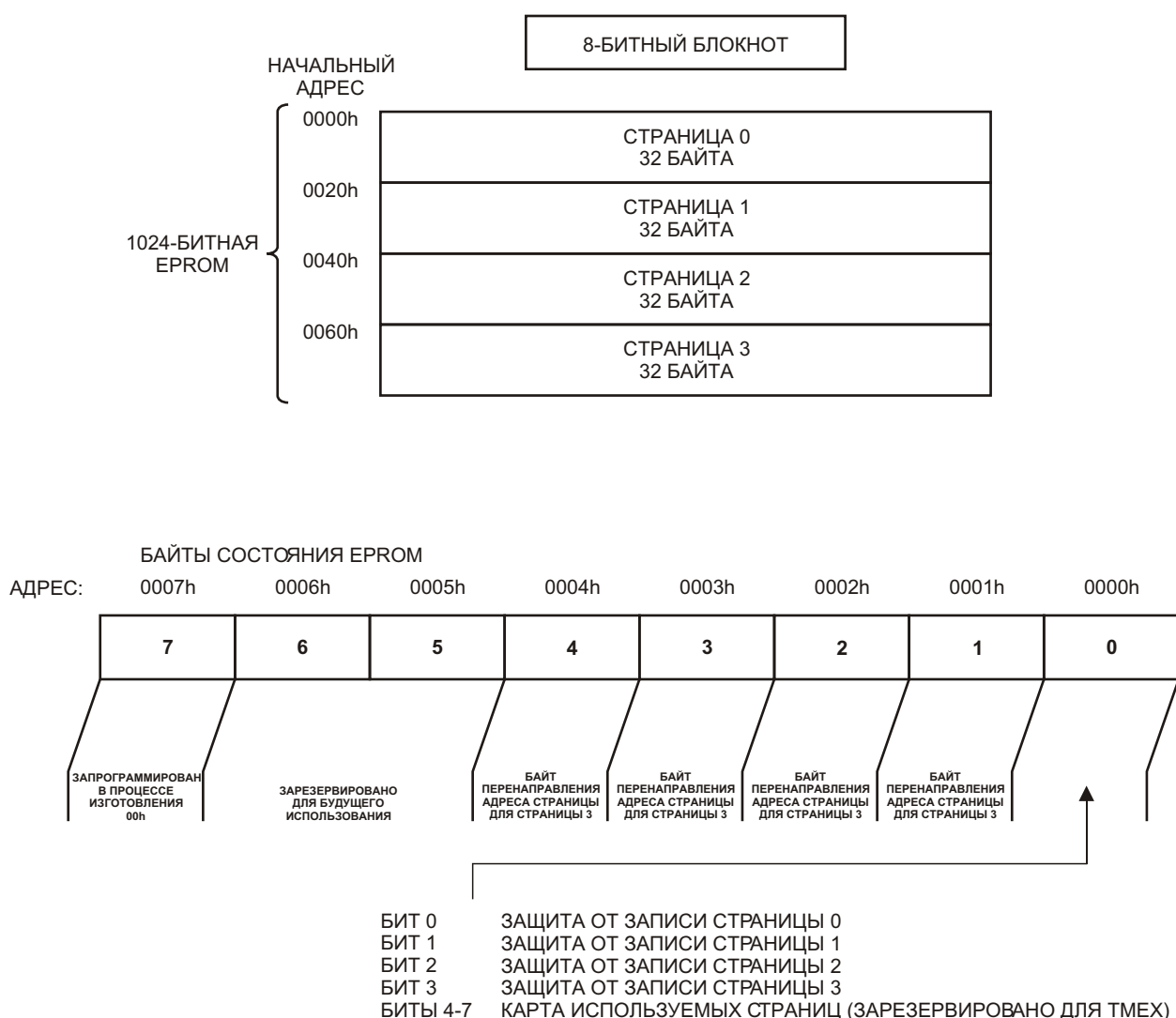


Рис. 6. БЛОК-СХЕМА ФУНКЦИЙ ПАМЯТИ.

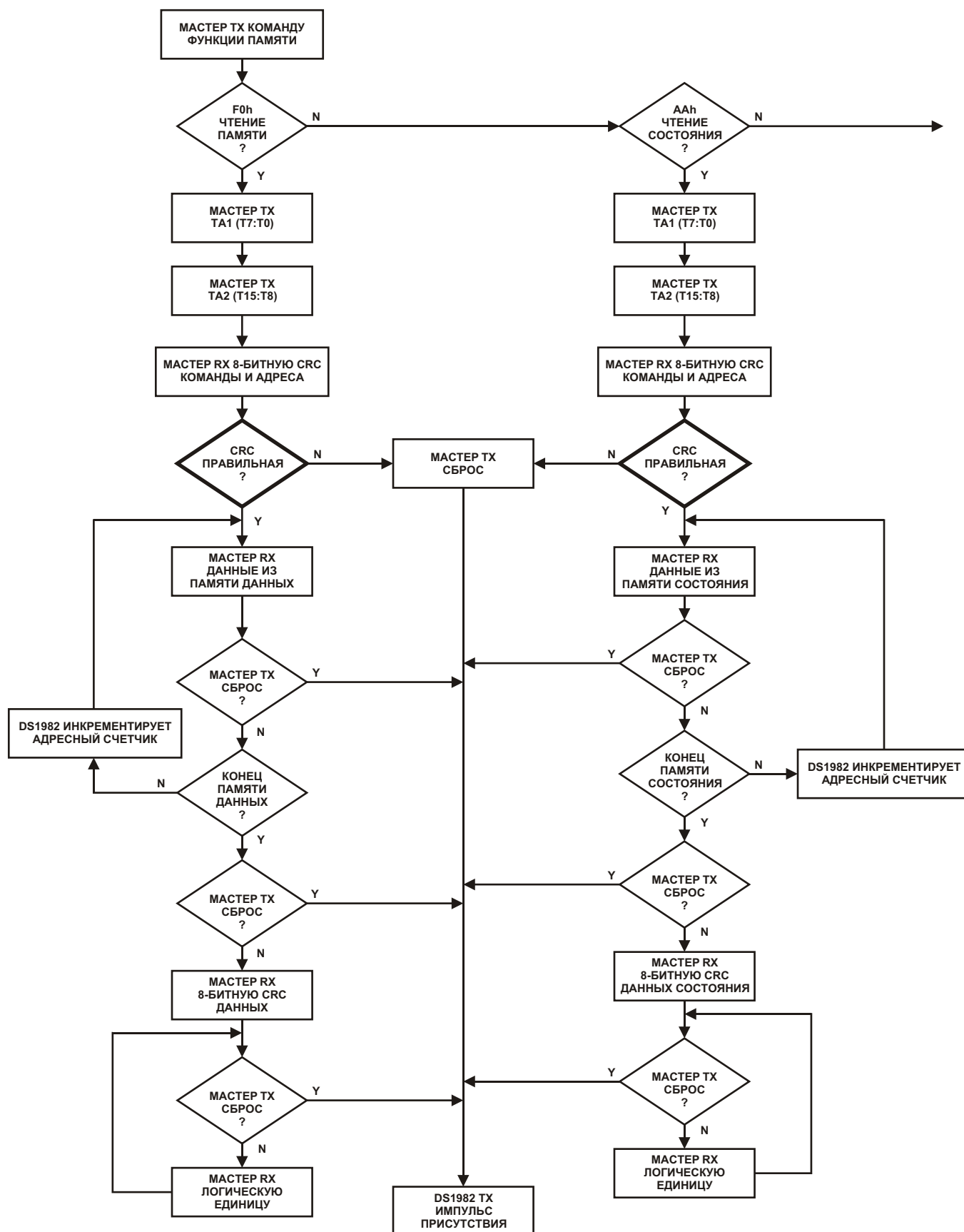


Рис. 6. БЛОК-СХЕМА ФУНКЦИЙ ПАМЯТИ (продолжение).

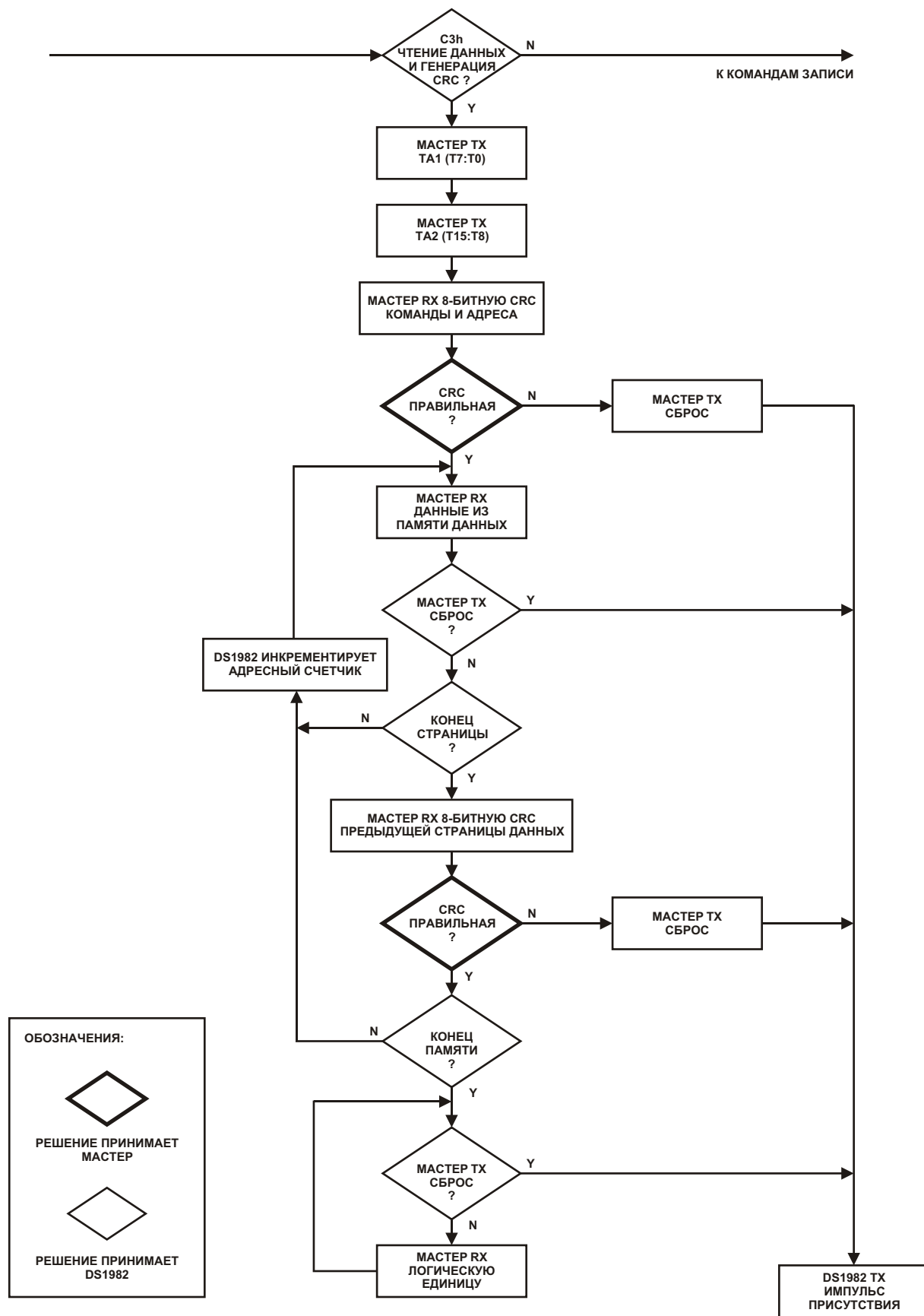
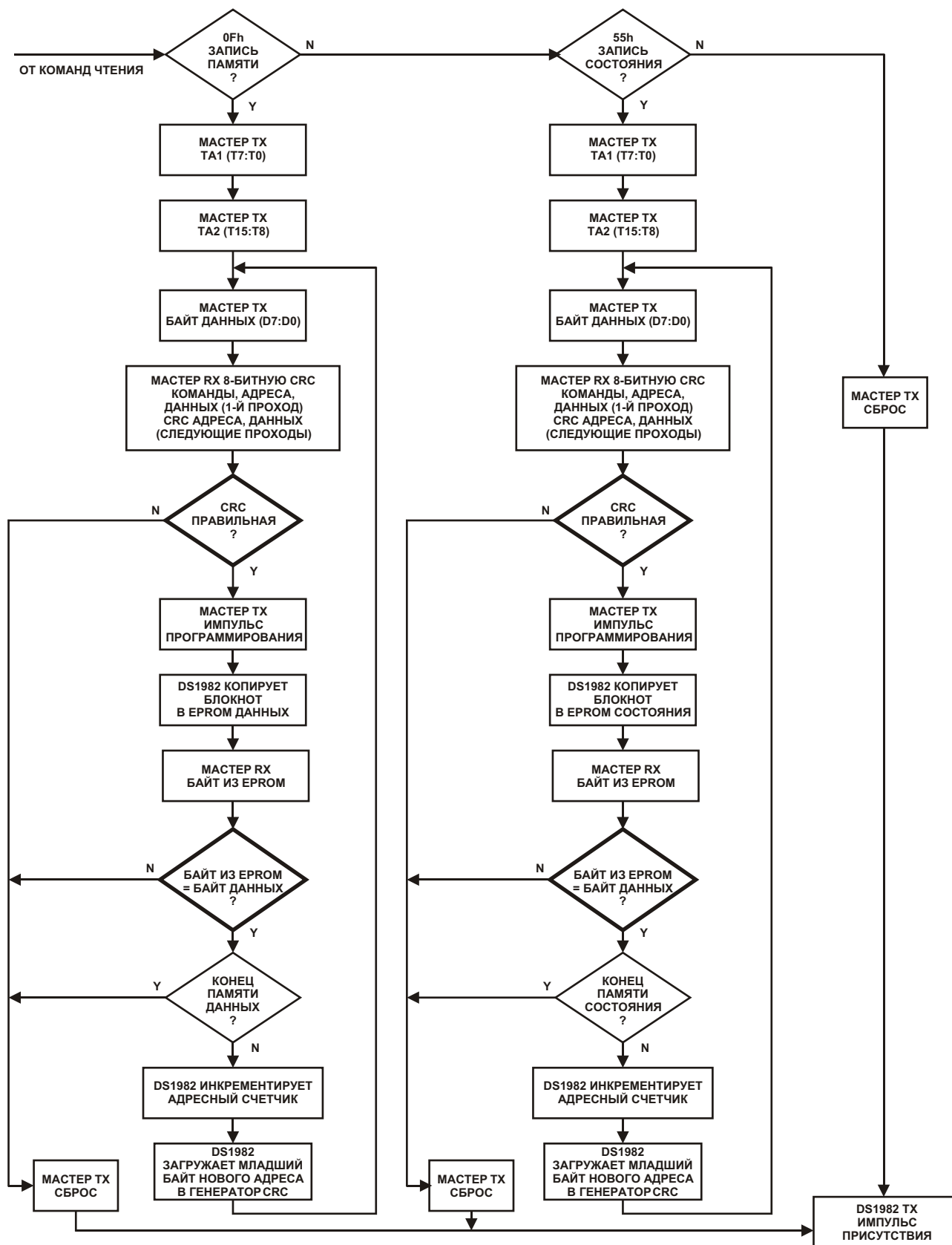


Рис. 6. БЛОК-СХЕМА ФУНКЦИЙ ПАМЯТИ (продолжение).



ЧТЕНИЕ ПАМЯТИ [F0h]

Команда чтения памяти используется для чтения данных из 1024-битного поля данных EPROM. Вслед за байтом команды мастер шины передает 2-байтный адрес ($TA1=(T7:T0)$, $TA2=(T15:T8)$), который указывает положение начального байта в поле данных. В DS1982 вычисляется 8-битная CRC байта команды и байтов адреса, которая затем считывается мастером для подтверждения правильности принятых команды и адреса. Если CRC, прочитанная мастером, является неправильной, должен быть выдан импульс сброса и вся последовательность должна быть повторена. Если прочитанная мастером CRC является правильной, мастер шины выдает интервалы чтения и принимает данные из DS1982, с начального адреса и до конца 1024-битного поля данных или до выдачи импульса сброса. Если в процессе чтения достигнут конец адресного пространства, мастер шины может выдать восемь дополнительных интервалов чтения, и DS1982 передаст 8-битную CRC для всех байтов данных, с начального байта до последнего байта памяти. После того, как CRC принята мастером шины, любые дополнительные интервалы чтения приводят к считыванию логической 1 вплоть до выдачи импульса сброса. Любое чтение, остановленное импульсом сброса ранее достижения границы памяти, не позволяет получить 8-битную CRC.

В типичном случае, в каждой странице памяти вместе с данными должна сохраняться 16-битная CRC для обеспечения скоростных и безошибочных пересылок данных. Это избавляет от необходимости многократного чтения страницы для определения того, являются ли считанные данные корректными, или нет (см. книгу «Book of DS19xx iButton Standards», глава 7, где приведена рекомендованная файловая структура, которая может быть использована с 1-проводными приборами). Если значения CRC встроены в данные, то во время выполнения команды чтения в конце адресного пространства может выдаваться импульс сброса, так как получение 8-битной CRC, вычисляемой DS1982, не требуется.

ЧТЕНИЕ СОСТОЯНИЯ [AAh]

Команда чтения состояния используется для чтения данных из поля данных состояния EPROM. Вслед за байтом команды мастер шины передает 2-байтный адрес ($TA1=(T7:T0)$, $TA2=(T15:T8)$), который указывает положение начального байта в поле данных. В DS1982 вычисляется 8-битная CRC байта команды и байтов адреса, которая затем считывается мастером для подтверждения правильности принятых команды и адреса. Если CRC, прочитанная мастером, является неправильной, должен быть выдан импульс сброса и вся последовательность должна быть повторена. Если прочитанная мастером CRC является правильной, мастер шины выдает интервалы чтения и принимает данные из DS1982, с начального адреса и до конца поля данных состояния EPROM. Затем мастер шины принимает 8-битную CRC, которая является результатом сдвига в генератор CRC всех байтов данных, с начального байта и до последнего байта, который запрограммирован в процессе изготовления устройства и содержит значение 00h.

Данная возможность нужна ввиду того, что информация о состоянии EPROM может со временем изменяться, а это делает невозможным однократное программирование данных вместе с сопутствующей CRC, которая всегда должна быть правильной. Поэтому команда чтения состояния всегда сопровождается 8-битной CRC, которая вычисляется на основе текущих данных, сохраненных в поле данных состояния EPROM.

После чтения 8-битной CRC мастер шины будет принимать логические 1 вплоть до выдачи импульса сброса. Выполнение команды чтения состояния EPROM может быть в любой момент прервано выдачей импульса сброса.

ЧТЕНИЕ ДАННЫХ/ГЕНЕРАЦИЯ 8-БИТНОЙ CRC [C3h]

Команда чтения данных/генерации 8-битной CRC используется для чтения данных из 1024-битного поля данных EPROM. Вслед за байтом команды мастер шины передает 2-байтный адрес ($TA1=(T7:T0)$, $TA2=(T15:T8)$), который указывает положение начального байта в поле данных. В DS1982 вычисляется 8-битная CRC байта команды и байтов адреса, которая затем считывается

мастером для подтверждения правильности принятых команды и адреса. Если CRC, прочитанная мастером, является неправильной, должен быть выдан импульс сброса и вся последовательность должна быть повторена. Если прочитанная мастером CRC является правильной, мастер шины выдает интервалы чтения и принимает данные из DS1982, с начального адреса и до конца 32-байтной страницы. Затем мастер шины выдает восемь дополнительных интервалов чтения и принимает 8-битную CRC для всех байтов данных, с начального байта и до последнего байта текущей страницы памяти. После того, как 8-битная CRC принята, продолжается чтение данных с начала следующей страницы 1024-битного поля данных EPROM. Эта последовательность будет повторяться, пока мастером шины не будет считана последняя страница вместе с CRC. Таким образом, каждая страница данных может рассматриваться как 33-байтная, где 32 байта являются данными EPROM, программируемыми пользователем, а последний байт представляет собой 8-битную CRC, которая генерируется автоматически в конце каждой страницы.

Этот тип чтения отличается от команды чтения памяти, которая просто считывает каждую страницу, пока не будет достигнута граница памяти. Команда чтения памяти генерирует 8-битную CRC только в конце адресного пространства, а это во многих приложениях может быть игнорировано, так как пользователь обычно сохраняет 16-битную CRC вместе с данными в каждой странице 1024-битного поля данных EPROM во время программирования страниц. Команда чтения данных/генерации 8-битной CRC обеспечивает альтернативную возможность чтения для приложений, которые «битово-ориентированы», а не «странично-ориентированы». В них информация в 1024-битной области EPROM может изменяться со временем без учета границы страниц, делая невозможным программирование страницы целиком с включением сопутствующей CRC, которая всегда должна быть правильной. Поэтому команда чтения данных/генерации 8-битной CRC снабжает каждую страницу 8-битной CRC, которая генерируется в DS1982 на основе текущих данных, сохраненных в каждой странице 1024-битного поля данных EPROM. После считывания 8-битной CRC последней страницы, мастер шины будет принимать из DS1982 логические 1 вплоть до выдачи импульса сброса. Выполнение команды чтения данных/генерации 8-битной CRC может быть в любой момент прервано выдачей импульса сброса.

ЗАПИСЬ ПАМЯТИ [0Fh]

Команда записи памяти используется для программирования 1024-битного поля данных EPROM. Вслед за байтом команды мастер шины передает 2-байтный начальный адрес ($TA1=(T7:T0)$, $TA2=(T15:T8)$) и байт данных ($D7:D0$). В DS1982 вычисляется 8-битная CRC для байта команды, байтов адреса и байта данных, которая затем считывается мастером для проверки правильности принятых DS1982 команды, адреса и данных.

Самым большим начальным адресом для DS1982 является адрес 007Fh. Если мастер шины передаст начальный адрес с большим значением, девять старших битов адреса будут сброшены внутренней схемой в 0. Это приведет к несоответствию между CRC, которую вычисляет DS1982, и CRC, вычисляемой мастером, что будет указывать на ошибку.

Если CRC, прочитанная мастером, является неправильной, должен быть выдан импульс сброса и вся последовательность должна быть повторена. Если прочитанная мастером CRC является правильной, мастер выдает импульс программирования (12В на 1-проводной шине в течении 480 мкс). До программирования все незапрограммированное 1024-битное поле данных EPROM содержит логические 1. Во время подачи импульса программирования биты в выбранном байте EPROM, которые соответствуют нулевым битам в переданном мастером байте данных, будут запрограммированы в состояние логического 0.

После подачи импульса программирования длительностью 480 мкс, когда линия данных возвратится к уровню 5В, мастер шины выдает восемь интервалов чтения для проверки того, что соответствующие биты запрограммировались. DS1982 отвечает байтом данных, считанным по выбранному адресу EPROM, передавая младший бит первым. Этот байт содержит результат логической операции «И» над всеми байтами, которые записывались по этому адресу EPROM. Если байт данных EPROM содержит 1 в тех позициях, где должны быть 0, должен быть выдан

импульс сброса и программирование байта по этому адресу должно быть повторено. Если байт данных EPROM содержит 0 в нужных позициях, значит программирование прошло успешно и DS1982 автоматически инкрементирует счетчик адреса для выбора следующего байта в 1024-битном поле данных. Младший байт нового 2-байтного адреса загружается в генератор 8-битной CRC в качестве начального значения. Затем мастер шины с помощью восьми интервалов записи передает следующий байт данных.

Как только DS1982 принимает этот байт данных в блокнот, данные вдвигаются в генератор CRC, который загружен младшим байтом текущего адреса. Результатом является 8-битная CRC нового байта данных и младшего байта нового адреса. После передачи байта данных мастер считывает эту 8-битную CRC из DS1982 с помощью восьми интервалов чтения, чтобы убедиться в том, что адрес инкрементирован правильно и данные переданы без ошибок. Если CRC, прочитанная мастером, является неправильной, должен быть выдан импульс сброса и команда записи памяти должна быть повторена. Если прочитанная мастером CRC является правильной, мастер выдает импульс программирования, и выбранный байт памяти записывается.

Заметьте, что при первоначальном проходе по блок-схеме команды записи памяти, 8-битная CRC является результатом сдвига в генератор CRC байта команды, 2-байтного адреса и байта данных. При последующих проходах по блок-схеме, когда DS1982 должен автоматически инкрементировать свой адресный счетчик, генерируется 8-битная CRC, которая является результатом загрузки (а не сдвига) младшего байта нового (инкрементированного) адреса в генератор CRC, а затем сдвига нового байта данных.

В обоих случаях решение продолжать запись (выдавать импульс программирования на DS1982) принимает исключительно мастер, так как DS1982 не может определить, совпадает ли 8-битная CRC, вычисленная мастером, с 8-битной CRC, вычисленной в DS1982. Если неправильная CRC будет игнорирована и мастер выдаст импульс программирования, произойдет неверное программирование DS1982. Заметьте также, что DS1982 всегда инкрементирует свой внутренний счетчик адреса после приема восьми интервалов чтения, используемых для подтверждения программирования выбранного байта EPROM. Решение продолжать запись снова принимает исключительно мастер; поэтому если байт данных EPROM не совпадает с переданным мастером байтом данных, но мастер продолжит запись, произойдет неверное программирование DS1982. Процесс записи может быть прерван в любой момент выдачей импульса сброса.

ЗАПИСЬ СОСТОЯНИЯ [55h]

Команда записи состояния используется для программирования поля данных состояния EPROM. Вслед за байтом команды мастер шины передает 2-байтный начальный адрес ($TA1=(T7:T0)$, $TA2=(T15:T8)$) и байт данных (D7:D0). В DS1982 вычисляется 8-битная CRC для байта команды, байтов адреса и байта данных, которая затем считывается мастером для проверки правильности принятых DS1982 команды, адреса и данных.

Если CRC, прочитанная мастером, является неправильной, должен быть выдан импульс сброса и вся последовательность должна быть повторена. Если прочитанная мастером CRC является правильной, мастер выдает импульс программирования (12В на 1-проводной шине в течении 480 мкс). До программирования первые 7 байт поля данных состояния EPROM содержит логические 1. Во время подачи импульса программирования биты в выбранном байте поля данных состояния EPROM, которые соответствуют нулевым битам в переданном мастером байте данных, будут запрограммированы в состояние логического 0. 8-й байт поля данных состояния EPROM запрограммирован в процессе изготовления устройства и содержит код 00h.

После подачи импульса программирования длительностью 480 мкс, когда линия данных возвратится к уровню 5В, мастер шины выдает восемь интервалов чтения для проверки того, что соответствующие биты запрограммировались. DS1982 отвечает байтом данных, считанным по выбранному адресу поля состояния EPROM, передавая младший бит первым. Этот байт содержит результат логической операции «И» над всеми байтами, которые записывались по этому адресу поля состояния EPROM. Если байт состояния EPROM содержит 1 в тех позициях, где должны

быть 0, должен быть выдан импульс сброса и программирование байта по этому адресу должно быть повторено. Если байт состояния EPROM содержит 0 в нужных позициях, значит программирование прошло успешно и DS1982 автоматически инкрементирует свой счетчик адреса для выбора следующего байта в поле состояния EPROM. Младший байт нового 2-байтного адреса загружается в генератор 8-битной CRC в качестве начального значения. Затем мастер шины с помощью восьми интервалов записи передает следующий байт данных.

Как только DS1982 принимает этот байт данных в блокнот, данные вдвигаются в генератор CRC, который загружен младшим байтом текущего адреса. Результатом является 8-битная CRC нового байта данных и младшего байта нового адреса. После передачи байта данных мастер считывает эту 8-битную CRC из DS1982 с помощью восьми интервалов чтения, чтобы убедиться в том, что адрес инкрементирован правильно и данные переданы без ошибок. Если CRC, прочитанная мастером, является неправильной, должен быть выдан импульс сброса и команда записи состояния должна быть повторена. Если прочитанная мастером CRC является правильной, мастер выдает импульс программирования и выбранный байт состояния записывается.

Заметьте, что при первоначальном проходе по блок-схеме команды записи состояния 8-битная CRC является результатом сдвига в генератор CRC байта команды, 2-байтного адреса и байта данных. При последующих проходах по блок-схеме, когда DS1982 должен автоматически инкрементировать свой адресный счетчик, генерируется 8-битная CRC, которая является результатом загрузки (а не сдвига) младшего байта нового (инкрементированного) адреса в генератор CRC, а затем сдвига нового байта данных.

В обоих случаях решение продолжать запись (выдавать импульс программирования на DS1982) принимает исключительно мастер, так как DS1982 не может определить, совпадает ли 8-битная CRC, вычисленная мастером, с 8-битной CRC, вычисленной в DS1982. Если неправильная CRC будет игнорирована и мастер выдаст импульс программирования, произойдет неверное программирование DS1982. Заметьте также, что DS1982 всегда инкрементирует свой внутренний счетчик адреса после приема восьми интервалов чтения, используемых для подтверждения программирования выбранного байта состояния EPROM. Решение продолжать запись снова принимает исключительно мастер; поэтому если байт состояния EPROM не совпадает с переданным мастером байтом данных, но мастер продолжит запись, произойдет неверное программирование DS1982. Процесс записи байтов состояния может быть прерван в любой момент выдачей импульса сброса.

1-ПРОВОДНАЯ ШИНА

1-проводная шина представляет собой систему, в которой имеется один мастер шины и одно или несколько подчиненных устройств. Во всех случаях DS1982 является подчиненным устройством. Мастером шины обычно является микроконтроллер. Обсуждение данной системы можно разбить на три части: аппаратная конфигурация, последовательность пересылки и 1-проводные сигналы (типы сигналов и их временные параметры). Протокол 1-проводной шины определяет пересылки с помощью понятия специальных временных интервалов, которые начинаются спадом импульса синхронизации, выдаваемого мастером. Более детальное описание протокола приведено в главе 4 книги «Book of DS19xx iButton Standards».

АППАРАТНАЯ КОНФИГУРАЦИЯ

1-проводная шина по определению имеет только одну линию; важным является тот факт, что каждое устройство, подключенное к шине, может в соответствующие моменты времени ею управлять. Для обеспечения такой возможности каждое устройство, подключенное к 1-проводной шине, должно иметь выход с открытым стоком или с тремя состояниями. DS1982 имеет выход с открытым стоком, его внутренняя схема эквивалентна показанной на рис. 7. Мастер шины может иметь такую же эквивалентную схему. Если двунаправленные выводы отсутствуют, могут быть соединены отдельные входной и выходной выводы.

Мастер шины требует наличия подтягивающего резистора на шине со стороны мастера, если его схема эквивалентна одной из схем, показанных на рис. 8а и 8б. Номинал подтягивающего резистора должен составлять примерно 5 Ком для линий небольшой длины.

Многоточечная шина представляет собой 1-проводную шину, к которой подключено несколько подчиненных устройств. Максимальная скорость передачи данных по 1-проводной шине составляет 16,3 Кбит в секунду. Если мастер шины должен иметь возможность программировать EPROM в DS1982, требуется источник питания для программирования, способный обеспечить ток до 10 мА при напряжении 12В в течении 480 мкс. В свободном состоянии линия 1-проводной шины находится в состоянии высокого уровня. Если по каким-либо причинам пересылка должна быть приостановлена, линия ДОЛЖНА быть оставлена в свободном состоянии, чтобы пересылка в последствии могла быть продолжена. Если этого не сделать и оставить линию в состоянии низкого уровня дольше, чем на 120 мкс, одно или несколько устройств на шине могут быть сброшены.

ПОСЛЕДОВАТЕЛЬНОСТЬ ПЕРЕСЫЛКИ

Последовательность действий для доступа к DS1982 через 1-проводный порт должна быть следующей:

- Инициализация
- Команда функций ПЗУ
- Команда функций памяти
- Чтение/запись памяти/состояния

ИНИЦИАЛИЗАЦИЯ

Все пересылки по 1-проводной шине начинаются с последовательности инициализации. Последовательность инициализации содержит импульс сброса, выдаваемый мастером шины, за которым следует импульс (импульсы) присутствия, передаваемый подчиненным устройством (устройствами).

Импульс присутствия говорит мастеру шины о том, что DS1982 имеется на шине и устройство готово к работе. Для более подробной информации см. раздел «Сигналы 1-проводной шины».

КОМАНДЫ ФУНКЦИЙ ПЗУ

Когда мастер шины обнаруживает импульс присутствия, он может подать одну из четырех команд функций ПЗУ. Все команды функций ПЗУ имеют длину 8 бит. Список этих команд приведен ниже (см. блок-схему на рис. 9):

ЧТЕНИЕ ПЗУ [33h]

Эта команда позволяет мастеру шины считать из DS1982 8-битный код семейства, уникальный 48-битный серийный номер и 8-битную CRC. Эта команда может быть использована только в том случае, когда на шине присутствует всего одно подчиненное устройство. Если имеется несколько подчиненных устройств, то произойдет искажение данных, так как все они попытаются одновременно передать данные (открытые стоки реализуют функцию «монтажное И»).

СРАВНЕНИЕ ПЗУ [55h]

Команда сравнения ПЗУ, за которой следует 64-битная последовательность содержимого ПЗУ, позволяет мастеру шины адресовать отдельное устройство на многоточечной шине. Только тот экземпляр DS1982, содержимое ПЗУ которого полностью совпадет с переданной мастером последовательностью, будет отвечать на последующие команды функций памяти. Все подчиненные устройства, содержимое ПЗУ которых не совпало с требуемым, будут ожидать

импульса сброса. Эта команда может использоваться при наличии на шине как одного, так и нескольких устройств.

Рис. 7. ЭКВИВАЛЕНТНАЯ СХЕМА DS1982.

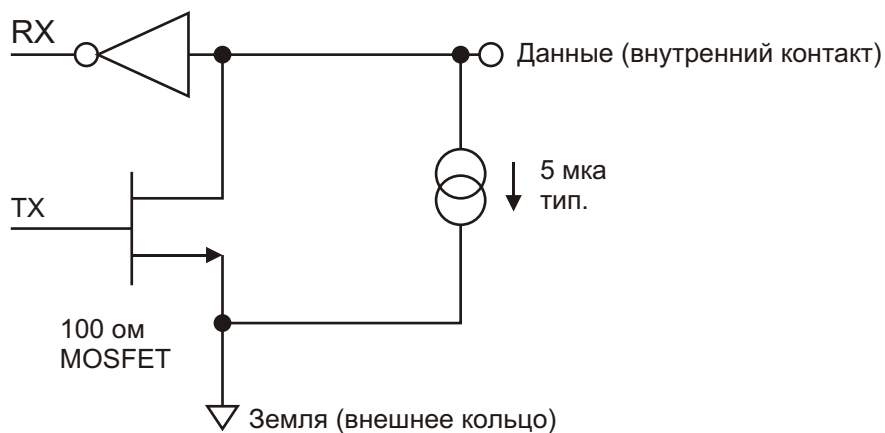


Рис. 8. СХЕМА МАСТЕРА ШИНЫ.

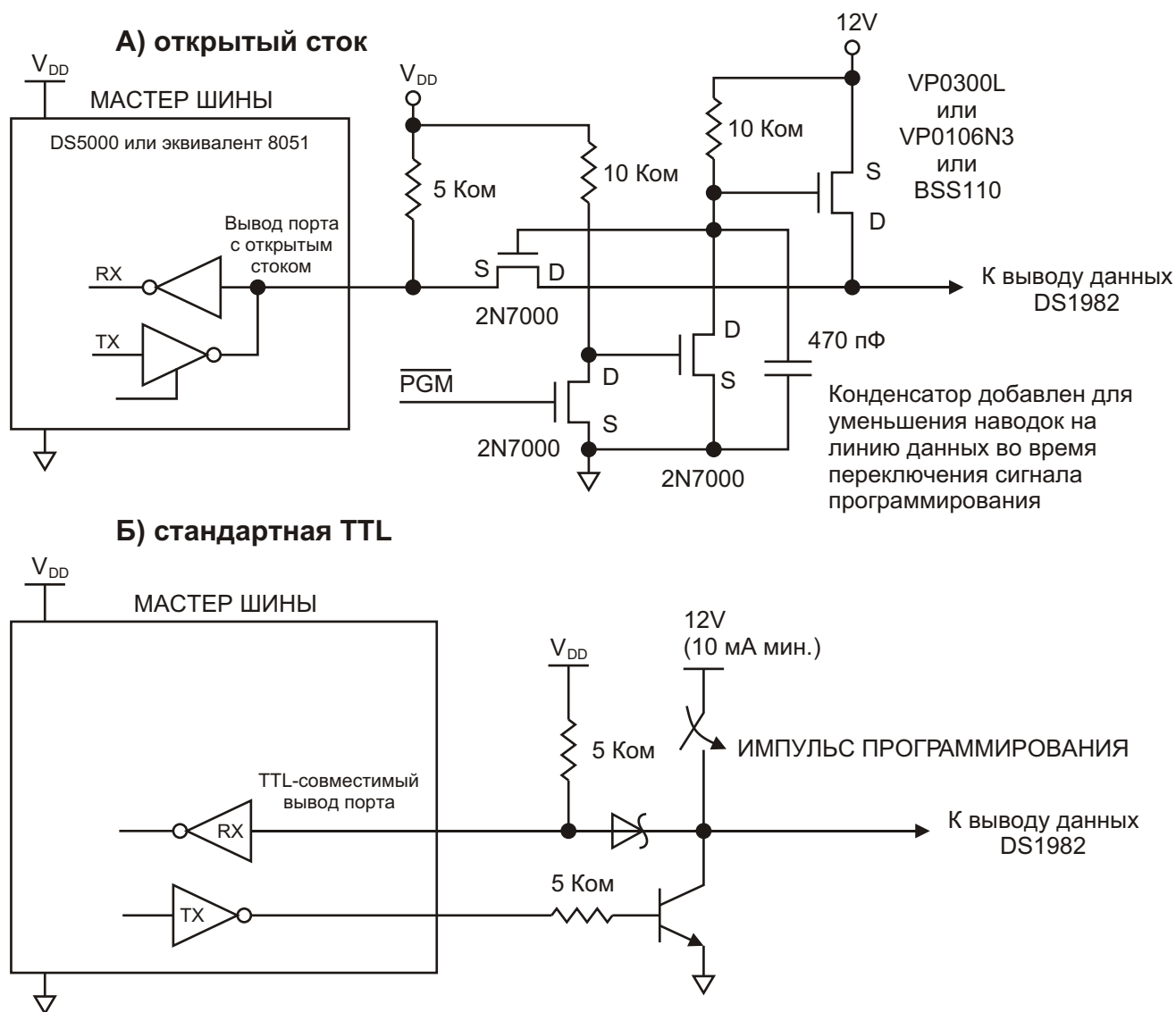
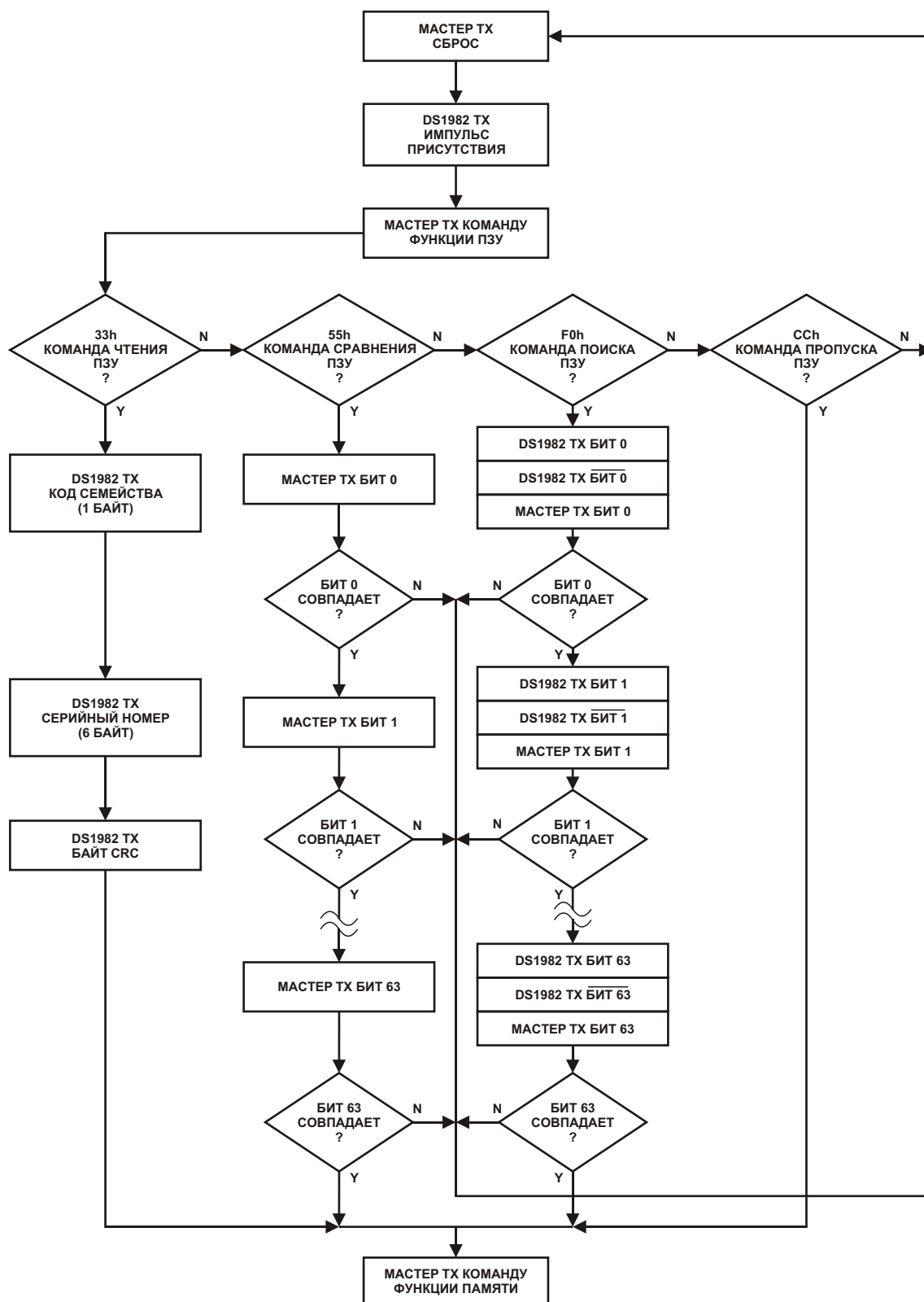


Рис. 9. БЛОК-СХЕМА ФУНКЦИЙ ПЗУ.



ПРОПУСК ПЗУ [CCh]

Эта команда позволяет экономить время в случае наличия на шине всего одного устройства, позволяя мастеру шины обращаться к функциям памяти без привлечения 64-битного кода ПЗУ. Если на шине присутствует более одного подчиненного устройства, и команда чтения посылается вслед за командой пропуска ПЗУ, произойдет искажение данных, так как несколько подчиненных устройств попытаются передать данные одновременно (открытые стоки реализуют функцию «монтажное И»).

ПОИСК ПЗУ [F0h]

Когда система включается в первый раз, мастер шины может не знать количества присутствующих на шине устройств или их 64-битных кодов ПЗУ. Команда поиска ПЗУ позволяет мастеру шины воспользоваться процессом идентификации 64-битных кодов ПЗУ всех подчиненных устройств, подключенных к шине. Процесс поиска ПЗУ представляет собой повторение простых, выполняющихся в три приема процедур: чтение бита, чтение инверсии бита, затем записи желаемого значения этого бита. Мастер шины выполняет эту процедуру для каждого бита ПЗУ. После одного полного прохода мастер шины определяет содержимое ПЗУ одного устройства. Коды ПЗУ остальных устройств можно определить с помощью дополнительных проходов. См. главу 5 книги «Book of DS19xx iButton Standards», где приведено исчерпывающее описание процесса поиска ПЗУ, включая конкретный пример.

СИГНАЛЫ 1-ПРОВОДНОЙ ШИНЫ

DS1982 требует строгого соблюдения протокола для гарантии целостности данных. Протокол содержит пять типов сигналов: последовательность сброса с импульсом сброса и импульсом присутствия, запись нуля, запись единицы, чтение данных и импульс программирования. Все эти сигналы, за исключением импульса присутствия, инициируются мастером шины. Последовательность инициализации, которая требуется для начала любого обмена с DS1982, показана на рис. 10. За импульсом сброса следует импульс присутствия, который говорит о готовности DS1982 принять команду функций ПЗУ. Мастер шины передает (TX) импульс сброса (t_{RSTL} , минимум 480 мкс). После этого мастер освобождает линию и переходит в режим приема (RX). 1-проводная шина находится в состоянии высокого уровня, что обеспечивается подтягивающим резистором. После обнаружения перехода линии в состояние высокого уровня, DS1982 формирует задержку (t_{PDH} , 15 – 60 мкс), а затем посылает импульс присутствия (t_{PDL} , 60 – 240 мкс).

ИНТЕРВАЛЫ ЗАПИСИ/ЧТЕНИЯ

Определение интервалов записи и чтения проиллюстрировано на рис. 11. Все интервалы инициируются мастером, который переводит линию в состояние низкого уровня. Этот переход линии данных синхронизирует DS1982 с мастером путем запуска в DS1982 схемы задержки. Для интервала чтения, когда передается 0, схема задержки определяет, как долго DS1982 будет удерживать линию в состоянии низкого уровня. Если передается 1, то подчиненное устройство не вмешивается в интервал чтения.

ИМПУЛЬС ПРОГРАММИРОВАНИЯ

Для копирования данных из 8-битного блокнота в EPROM или память состояния, после проверки мастером правильности CRC для текущего байта, на линию данных подается 12-вольтный импульс программирования. В процессе программирования переходом линии данных из свободного состояния, когда высокий уровень на ней обеспечивается подтягивающим резистором, в состояние, когда на линию подается напряжение 12В и обеспечивается ток как минимум 10 мА для DS1982, управляет мастер шины. Напряжение программирования (см. рис. 12) должно быть

подано на 480 мкс, затем мастер шины возвращает линию данных в свободное состояние с высоким уровнем, обеспечиваемым подтягивающим резистором. Заметьте, что из-за требований высоковольтного программирования для 1-проводных устройств с EPROM, невозможно во время программирования вместе с DS1982 подключать на шину 1-проводные устройства без EPROM. Внутренние диоды 1-проводных устройств без EPROM ограничат напряжение на линии данных на уровне примерно 8В, к тому же потенциально возможно повреждение этих устройств.

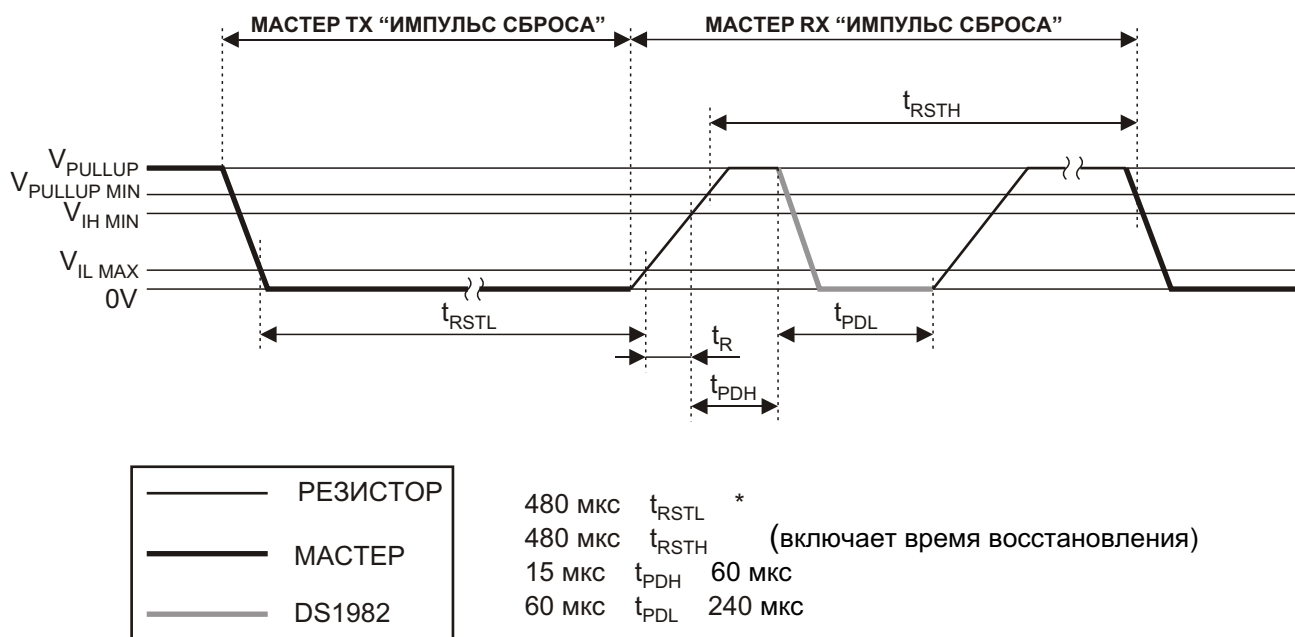
ВЫЧИСЛЕНИЕ CRC

DS1982 содержит 8-битную CRC, которая сохранена в старшем байте 64-битного ПЗУ. Мастер шины может вычислить значение CRC для первых 56 бит 64-битного ПЗУ и сравнить его со значением, сохраненным в DS1982 для проверки правильности считывания данных ПЗУ. Эквивалентный полином для этой CRC имеет следующий вид: $X^8 + X^5 + X^4 + 1$.

В некоторых условиях DS1982 генерирует 8-битную CRC, используя полином, приведенный выше, и передает ее значение мастеру шины для проверки правильности приема команды, адреса и данных от мастера. Блок-схема функций памяти на рис. 6 показывает, что DS1982 вычисляет 8-битную CRC для команды, адреса и данных, принятых для записи в память или в байты состояния, затем передает ее значение мастеру шины для подтверждения правильности пересылки. Подобным образом DS1982 вычисляет 8-битную CRC для команды и адреса, принятых от мастера шины для чтения памяти, чтения состояния и чтения данных/генерации 8-битной CRC для проверки того, что эти байты были приняты без ошибок. Генератор CRC в DS1982 также используется для проверки правильности пересылки, когда мастеру шины передается каждая страница 1024-битной EPROM по команде чтения данных/генерации 8-битной CRC, а также для 8 байт информации памяти состояния.

Во всех случаях, когда используется CRC для проверки правильности пересылки, мастер шины должен вычислять значение CRC, используя полином, приведенный выше, а затем сравнивать вычисленное значение с 8-битной CRC, сохраненной в 64-битном ПЗУ DS1982 (при чтении ПЗУ) или с 8-битным значением CRC, вычисленным DS1982. Сравнение значений CRC и принятие решения продолжения операции выполняется исключительно мастером. В DS1982 отсутствуют схемы, которые способны прерывать выполнение команды в случае отличия сохраненной или вычисленной CRC от значения, генерируемого мастером. Результатом правильного использования CRC, которое показано на блок-схеме (рис. 6), будет высокая надежность передачи данных в канале. Более подробное описание процесса вычисления CRC, включая пример аппаратной и программной реализации, приведено в книге «Book of DS19xx iButton Standards».

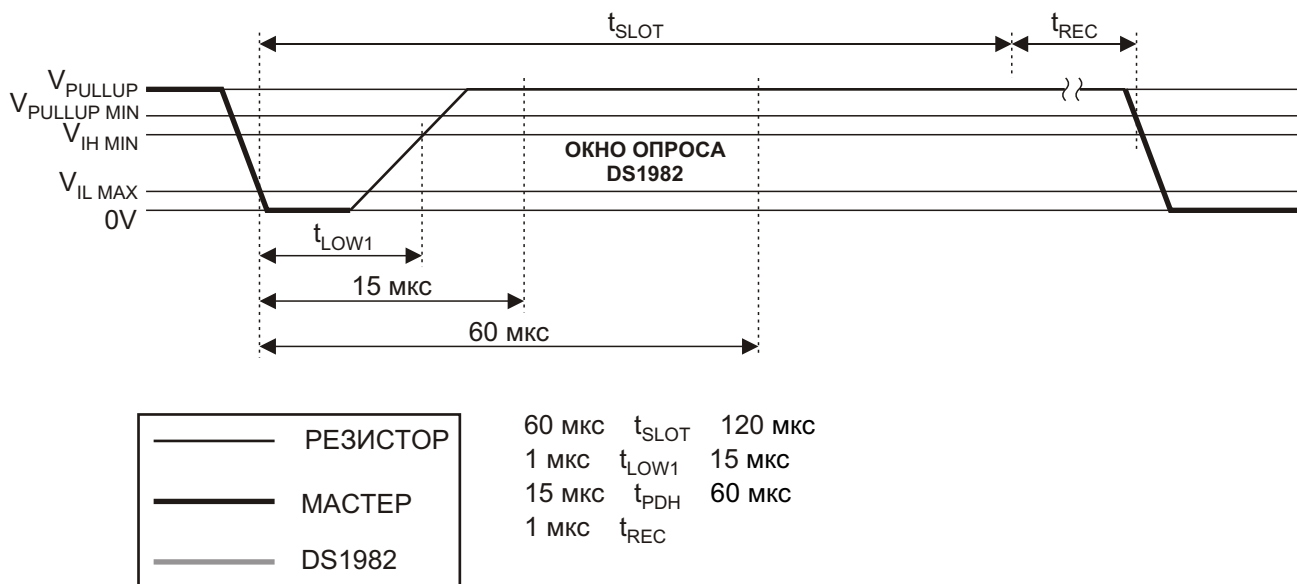
Рис. 10. ПРОЦЕДУРА ИНИЦИАЛИЗАЦИИ «ИМПУЛЬСЫ СБРОСА И ПРИСУТСТВИЯ».



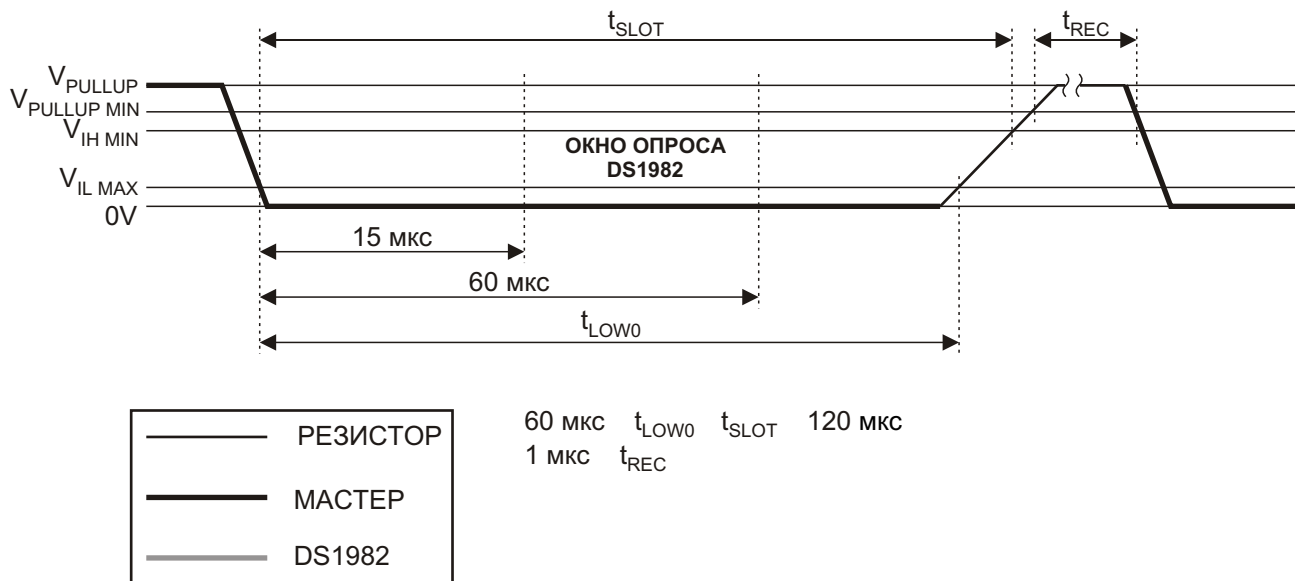
* Для того, чтобы не маскировать прерывания, посылаемые по 1-проводной шине другими устройствами, длительность $t_{RSTL} + t_R$ должна быть всегда меньше 960 мкс.

Рис. 11. ВРЕМЕННАЯ ДИАГРАММА ЗАПИСИ/ЧТЕНИЯ.

Временной интервал записи единицы



Временной интервал записи нуля



Временной интервал чтения данных

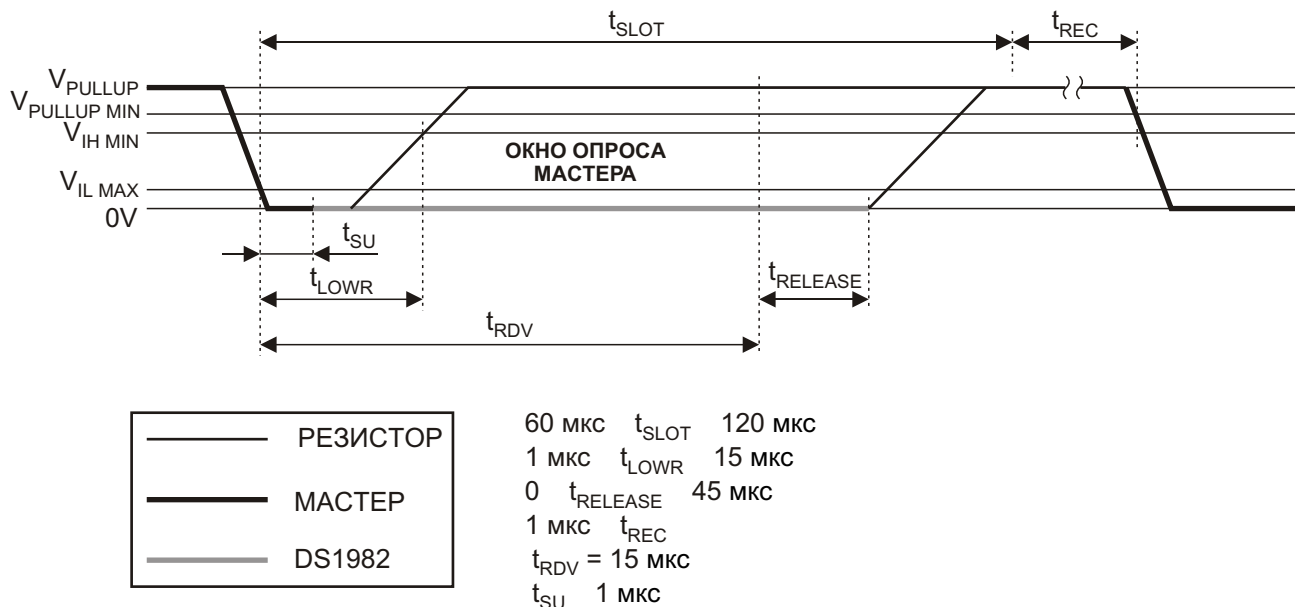
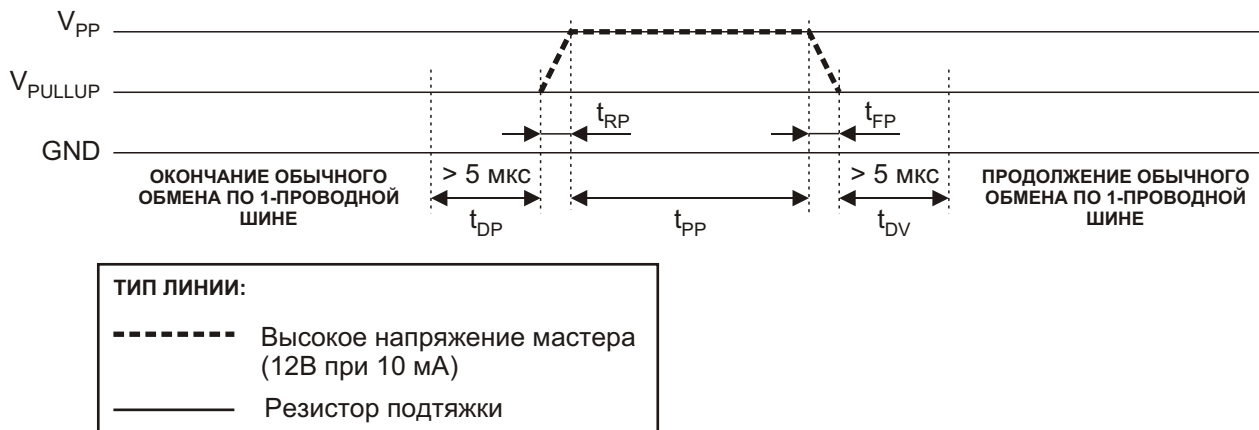


Рис. 12. ВРЕМЕННАЯ ДИАГРАММА ИМПУЛЬСА ПРОГРАММИРОВАНИЯ.



МАКСИМАЛЬНО ДОПУСТИМЫЕ УСЛОВИЯ*

Напряжение на любом выводе относительно земли	-0.5В .. +12В
Рабочая температура	-40°C .. +85°C
Температура хранения	-55°C .. +125°C

*Данные условия являются напряженными, и функционирование устройства при этих или любых других условиях, выходящих за приведенные в спецификации рамки, не предполагается. Работа при максимально допустимых условиях в течение длительного периода времени может привести к снижению надежности.

СТАТИЧЕСКИЕ ХАРАКТЕРИСТИКИ(V_{PUP}=2,8В .. 6,0В; -40°C .. +85°C)

ПАРАМЕТР	СИМВ.	МИН	ТИП	МАКС	ЕДИН.	ПРИМ.
Логическая 1	V _{IN}	2,2			В	1, 6
Логический 0	V _{IL}	-0,3		+0,8	В	1, 11
Выход логич. 0 при 4 мА	V _{OL}			0,4	В	1
Выход логич. 1	V _{OH}		V _{PUP}	6,0	В	1, 2
Входной ток нагрузки	I _L		5		мкА	3
Рабочий заряд	Q _{OP}			30	нКл	7, 8
Напряжение программирования при 10 мА	V _{PP}	11,5		12,0	В	

ЕМКОСТЬ(T_{ОКР. СР.}=25°C)

ПАРАМЕТР	СИМВ.	МИН	ТИП	МАКС	ЕДИН.	ПРИМ.
Вывод данных	C _{IN/OUT}			800	пФ	9

ДИНАМИЧЕСКИЕ ХАРАКТЕРИСТИКИ(V_{PUP}=2,8В .. 6,0В; -40°C .. +85°C)

ПАРАМЕТР	СИМВ.	МИН	ТИП	МАКС	ЕДИН.	ПРИМ.
Временной интервал	t _{SLOT}	60		120	мкс	
Длительность низкого уровня при записи 1	t _{LOW1}	1		15	мкс	
Длительность низкого уровня при записи 0	t _{LOW0}	60		120	мкс	
Действительные данные при чтении	t _{RDV}	точно 15			мкс	
Время освобождения шины	t _{RELEASE}	0	15	45	мкс	
Установление данных при чтении	t _{SU}			1	мкс	5
Время восстановления	t _{REC}	1			мкс	
Длительность высокого уровня при сбросе	t _{RSTH}	480			мкс	4
Длительность низкого уровня при сбросе	t _{RSTL}	480			мкс	
Высокий уровень импульса присутствия	t _{PDHIGH}	15		60	мкс	
Низкий уровень импульса присутствия	t _{PDLow}	60		240	мкс	
Задержка перед программированием	t _{DP}	5			мкс	10
Задержка перед проверкой	t _{DV}	5			мкс	10
Длительность импульса записи	t _{PP}	480		5000	мкс	10, 12
Время нарастания напряжения программир.	t _{RP}	0,5		5,0	мкс	10
Время спада напряжения программир.	t _{FP}	0,5		5,0	мкс	10

ПРИМЕЧАНИЯ:

1. Все напряжения указаны относительно земли.
2. V_{PUP} = напряжение внешней подтяжки.
3. Нагрузка на землю, представленная входом.
4. Дополнительный сброс или последовательность обмена не может начинаться во время высокого уровня сброса.
5. Время установления данных при чтении определяет длительность низкого уровня, который мастер должен установить на шине для чтения бита. Данные гарантированно установлены через 1 мкс после спада и остаются действительными минимум 14 мкс (в сумме 15 мкс после спада на 1-проводной шине).
6. V_{IN} является функцией внешнего подтягивающего резистора и напряжения, к которому он подключен.
7. 30 нанокулон за 72 временных интервала при 5В.
8. При $V_{CC}=5,0В$ с подтягивающим резистором 5 Ком на V_{CC} и максимальным временным интервалом 120 мкс.
9. При первом включении емкость вывода данных может быть 800 пФ. Если используется подтягивающий резистор 5 Ком с линии данных на V_{CC} , то достаточно 5 мс после включения питания, чтобы паразитная емкость перестала влиять на нормальный обмен.
10. Максимальное напряжение программирования на шине для параметров программирования составляет от 11,5В до 12,0В; температурный диапазон от $-40^{\circ}C$ до $+50^{\circ}C$.
11. При некоторых условиях для малых напряжений питания может потребоваться уменьшение $V_{IL,MAX}$ вплоть до 0,5В для того, чтобы всегда гарантировать наличие импульса присутствия.
12. Суммарная длительность импульсов программирования для каждого адреса не должна превышать 5 мс.