

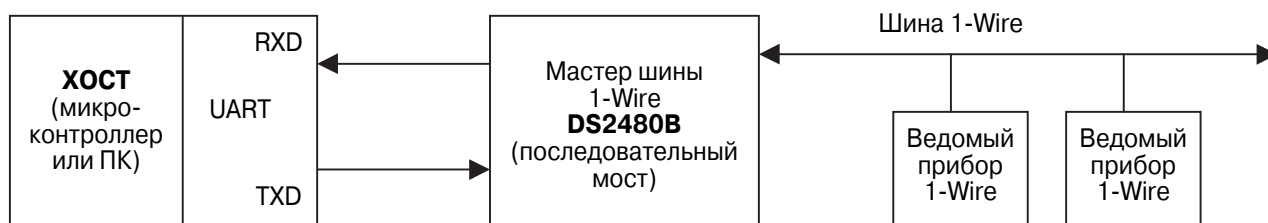
ВВЕДЕНИЕ

Протокол обмена данными для шины 1-Wire® может быть реализован с использованием порта ввода/вывода микропроцессора. Однако для создания надежно функционирующей сети 1-Wire необходимо также обеспечить корректные временные соотношения и соответствующие скорости нарастания/спада сигналов. Генерируемые мастером шины 1-Wire сигналы с некорректными временными параметрами могут быть причиной неустойчивого обмена данными с ведомыми приборами 1-Wire или даже его полного срыва. Отсутствие возможности управления скоростью нарастания/спада сигналов может существенно ограничить длину сети и привести к непредсказуемому поведению системы. При наличии последовательного коммуникационного порта UART все эти проблемы можно решить путем использования моста «последовательный порт — шина 1-Wire» (DS2480B).

Прибор DS2480B является последовательным мостом для сети, использующей протокол 1-Wire. Этот мост позволяет любому хосту, снабженному самым примитивным последовательным интерфейсом UART, генерировать соответствующие временные сигналы 1-Wire с управляемой скоростью нарастания/спада. Прибор DS2480B получает от хоста команды и данные, выполняет операции на шине 1-Wire и возвращает результат обратно хосту. Упрощенная схема применения DS2480B показана на Рис. 1. Описание реализации этого протокола и подробное рассмотрение всех возможных команд DS2480B может занять много времени и оказаться довольно трудным для понимания. Поэтому в настоящем руководстве описываются только общие операции 1-Wire и разъясняется формирование входящих и интерпретация исходящих пакетов последовательных данных для микросхемы DS2480B.

Этот документ является дополнением к спецификации на DS2480B, но не заменяет ее. Спецификацию на DS2480B можно найти по адресу <http://www.maxim-ic.com/>.

Рис. 1. Применение DS2480B (упрощенная схема)



Модуль UART хоста для работы с этим мостом должен как минимум поддерживать следующий протокол обмена данными: 8 битов данных, отсутствие контроля по четности, скорость передачи — 9600 бод (бит/с). Могут использоваться и более высокие скорости, вплоть до 115 200 бод, но при включении питания мост всегда начинает работать на скорости 9600 бод. Вопросы электрического подключения, например, к интерфейсу RS232, рассмотрены в спецификации на DS2480B.

ИНТЕРФЕЙС 1-WIRE

Все команды и режимы микросхемы DS2480B должны быть преобразованы в программный интерфейс обмена данными 1-Wire, на основе которого строятся приложения и который может ими использоваться. Существует несколько базовых функций (операций) 1-Wire, которые должно поддерживать приложение для выполнения любых действий на шине 1-Wire. Первой такой операцией является сброс всех ведомых приборов, подключенных к шине 1-Wire, подготавливающий их к приему команды от мастера шины. Второй операцией является запись бита, передаваемого от мастера шины 1-Wire ведомым приборам, а третьей — чтение бита из ведомых приборов 1-Wire. Поскольку все процессы обмена данными на шине 1-Wire инициируются мастером, технически «чтение» представляет собой «запись» единицы со считыванием (выборкой) результата. Практически все остальные действия на шине 1-Wire могут быть реализованы с помощью этих трех операций. Например, запись (выдача) байта на шину 1-Wire представляет собой просто восемь операций записи одиночных битов. Даже алгоритм поиска приборов на шине 1-Wire (см. Руководство по применению AN187, <http://pdfserv.maxim-ic.com/arpdf/AppNotes/app187.pdf>) может быть реализован при помощи этих примитивов (элементарных функций). Однако такой метод реализации не обязательно будет наиболее эффективным.

В микросхеме DS2480B имеется режим ускорения поиска, который позволяет значительно сократить последовательный обмен данными, требуемый для выполнения поиска. Для повышения эффективности следует также объединять последовательности передаваемых битов в байты и даже в блоки байтов. Везде, где это возможно, для достижения максимальной эффективности в приложении необходимо использовать наибольшее группирование команд (пакеты самого большого размера).

В Табл. 1 приведен минимальный набор операций для реализации интерфейса 1-Wire. Их наименования будут использоваться далее в настоящем документе в качестве обозначений конкретной операции.

Таблица 1. Базовые операции 1-Wire

Операция	Описание
OWReset	Посылает на шину 1-Wire импульс сброса и проверяет наличие импульсов присутствия от ведомых приборов 1-Wire
OWWriteBit/OWReadBit	Посылает или принимает один бит данных по шине 1-Wire
OWWriteByte/OWReadByte	Посылает или принимает один байт данных по шине 1-Wire
OWBlock	Посылает и принимает несколько байтов данных по шине 1-Wire
OWSearch	Выполняет алгоритм поиска на шине 1-Wire (см. Руководство по применению AN187)

Кроме того, имеются расширенные функции 1-Wire, которые не входят в состав базовых операций. Некоторые ведомые приборы 1-Wire могут работать на двух различных скоростях обмена данными: обычной (стандартной) и повышенной. Все устройства поддерживают как минимум обычную скорость. Повышенная скорость примерно в 10 раз больше обычной. Микросхема DS2480B поддерживает обе скорости обмена данными по шине 1-Wire.

Обычно приборы 1-Wire «заимствуют» часть или всю энергию, необходимую для своей работы, с шины 1-Wire. Однако некоторым приборам в отдельные моменты времени требуется дополнительная энергия. Например, прибору необходимо выполнить преобразование температуры или вычислить хэш-функцию по алгоритму SHA-1. Эта энергия обеспечивается за счет подключения более мощной схемы подтяжки (pull-up) шины 1-Wire. Во время такой подачи питания обычный обмен данными осуществляться не может. Микросхема DS2480B имеет несколько дополнительных возможностей для обеспечения подачи питания.

Однократно программируемые приборы памяти ППЗУ (EPROM) 1-Wire требуют для осуществления записи специального импульса 12 В. Если к микросхеме DS2480B подключено напряжение питания 12 В, то ей может быть передана команда послать на шину 1-Wire импульс для программирования ППЗУ.

В Табл. 2 приведены расширенные операции 1-Wire, касающиеся управления скоростью обмена данными, подачи питания и импульса программирования.

Таблица 2. Расширенные операции 1-Wire

Операция	Описание
OWSpeed	Устанавливает скорость обмена данными по шине 1-Wire. Может быть выбрана обычная или повышенная скорость. Заметим, что эта операция изменяет только скорость работы мастера шины 1-Wire; команда переключения режима работы ведомых приборов 1-Wire должна передаваться при переходе от обычной скорости к повышенной. Ведомые приборы 1-Wire всегда будут возвращаться к обычной скорости при обнаружении на шине импульса сброса, передаваемого на обычной скорости
OWLevel	Устанавливает режим питания шины 1-Wire (обычный или подачи питания)
OWProgramPulse	Посылает импульс программирования 12 В заданной длительности для записи приборов ППЗУ 1-Wire
OWReadBitPower	Считывает один бит данных с шины 1-Wire и, как опция, осуществляет подачу питания сразу же после завершения считывания бита
OWWriteBytePower	Посылает один байт данных на шину 1-Wire и осуществляет подачу питания сразу же после завершения передачи байта

В настоящем документе представлена реализация базовых и расширенных операций 1-Wire с использованием DS2480B. Эти операции обеспечивают выполнение всех функций как существующих, так и будущих приборов 1-Wire. Абстрактное представление операций 1-Wire позволяет получать приложения 1-Wire, не зависящие от типа мастера шины (см. Пример 1). Заметим, что данная реализация не является единственно возможной и не обязательно использует все функции микросхемы DS2480B. Другие примеры использования операций 1-Wire приведены в конце настоящего документа.

Пример 1. Псевдокод чтения памяти

trans_block – временный буфер передачи, значения представлены в шестнадцатеричной системе

```
// сброс шины 1-Wire
If OWReset = TRUE

    // послать команду Сравнение ПЗУ прибору, память которого будет
    // считываться; идентификатор ПЗУ содержится в байтах R0...R7
    trans_block = 55,R0,R1,R2,R3,R4,R5,R6,R7
    OWBlock(trans_block)

    // послать команду Чтение памяти, адрес (0),
    // и 32 байта для считывания страницы данных
    trans_block = F0,00,00,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,
        FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF
    OWBlock(trans_block)

    // данные страницы 0 теперь находятся в последних 32 байтах
    // буфера trans_block
    ...
Else
    // прибор отсутствует
    ...
EndIf
```

КОНФИГУРАЦИЯ ХОСТА

Хост прибора DS2480B должен иметь модуль UART, способный как минимум поддерживать 8-битный обмен данными без контроля по четности при скорости 9600 бод. Конфигурация модуля UART хоста зависит от используемой платформы и поэтому не рассматривается в данном документе. Тем не менее, он должен обеспечивать стандартные операции интерфейса, включающие в себя чтение/запись, завершение любых незавершенных операций чтения/записи с последующим сбросом буфера, генерирование сигнала прерывания и, как опция, изменение скорости передачи данных. В Табл. 3 приведен список терминов, используемых при описании интерфейса для типичного модуля UART хоста. Эти термины будут использоваться далее при описании операций, выполняемых модулем UART.

Таблица 3. Операции, которые должен выполнять модуль UART хоста

Операция	Описание
Break (Прерывание)	Посылает сигнал прерывания (BREAK) на коммуникационный порт в течение не менее 2 мс
Flush (Очистка буфера)	Позволяет завершить любую незавершенную операцию записи и очистить входной буфер (чтения)
Read (Чтение)	Считывает определенное число байтов из последовательного порта. Обеспечивает достаточно длинный тайм-аут, чтобы при работе в обычном режиме гарантировать прием всех байтов
Write (Запись)	Записывает определенное число байтов в последовательный порт. Возвращает результат после записи всех байтов
SetBaud (Установка скорости передачи данных)	Изменяет скорость передачи данных последовательного порта на заданное значение скорости (эта опция используется, если требуется работа в ускоренном режиме)
Delay (Задержка)	Осуществляет задержку не менее чем на заданное число миллисекунд

Пример реализации на языке «Си» функций, описанных в настоящем руководстве, где в качестве хоста используется последовательный порт RS232 (под управлением 32-битной операционной системы Windows) с адаптером DS9097U, можно загрузить по следующему адресу:

ftp://ftp.dalsemi.com.pub.auto_id/public/an192.zip.

Данный пример представляет собой упрощенную версию кода, который входит в состав общедоступного комплекта ПО для шины 1-Wire (1-Wire Public Domain Kit). Этот комплект, который можно найти по адресу <http://www.ibutton.com/software/1wire/wirekit.html>, содержит также модули и примеры, относящиеся к конкретным приборам.

КОНФИГУРАЦИЯ DS2480B

Перед тем, как выполнять какие-либо операции на шине 1-Wire, хост должен установить параметры последовательного драйвера линии 1-Wire DS2480B и синхронизироваться с ним. Эта процедура установки и синхронизации выполняется также всякий раз при обнаружении каких-либо проблем во время обмена данными между хостом и мостом. Во время установки параметров DS2480B скорость передачи данных должна составлять 9600 бод. После завершения процедуры установки скорость передачи может быть увеличена до 115 200 бод. Однако, поскольку прибор DS2480B имеет однобайтный входной буфер, необходимо быть очень внимательным. Заданная команда 1-Wire должна завершиться до начала ввода следующей команды. Чтобы узнать, на каких скоростях будет выполняться та или иная команда, следует обратиться к Табл. 7 спецификации на DS2480B.

DS2480B_Detect

Поскольку микросхема DS2480B не имеет кварцевого резонатора, она должна осуществлять подстройку своего задающего генератора, используя сигналы последовательного обмена данными, посылаемые хостом. Последовательность установки инициируется сбросом микросхемы DS2480B и последующей посылкой специального байта синхронизации. В результате сброса прибора все параметры конфигурации 1-Wire примут значения, устанавливаемые по умолчанию. Для улучшения работы с сетя-

ми малой и средней длины, при осуществлении обмена данными на обычной скорости рекомендуется использовать микросхему DS2480B в режиме «гибкой» (flex) скорости. Для формирования сигналов 1-Wire в режиме гибкой скорости используются параметры конфигурации 1-Wire. Следовательно, после каждого сброса микросхемы DS2480B необходимо перезагружать параметры конфигурации. Предпочтительными значениями параметров при гибкой скорости являются: PDSRC = 1.37 В/мкс, WILT = 10 мкс, DSO/WORT = 8 мкс (условные обозначения параметров приведены в Приложении к настоящему документу). Описанная последовательность сброса и конфигурирования реализована в функции *DS2480B_Detect*.

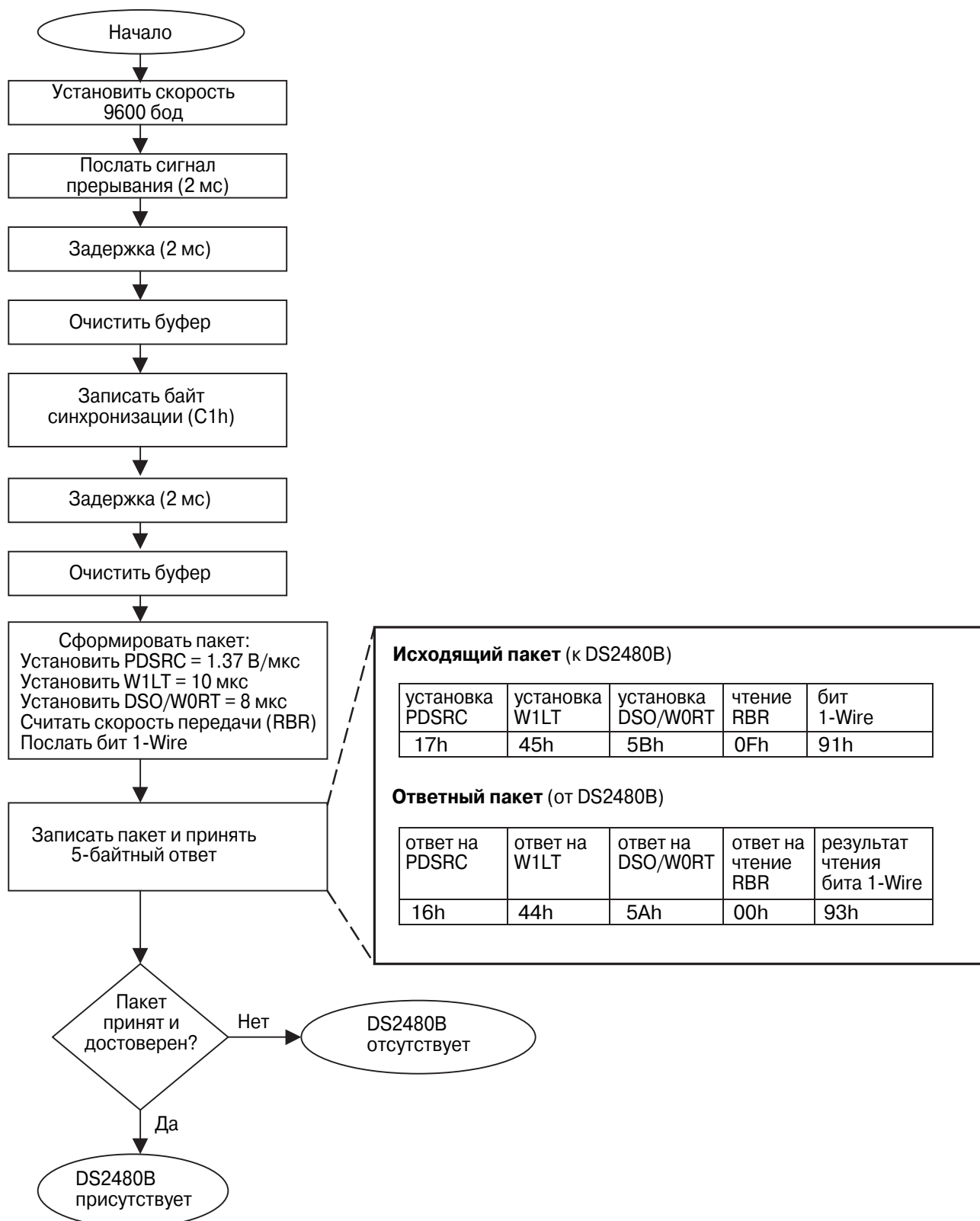
Микросхема DS2480B сбрасывается, если она обнаруживает вместо стоп-бита уровень паузы. Самым простым способом генерирования паузы является прерывание последовательного обмена данными на время, превышающее длительность 8-битного слова, передаваемого на скорости 9600 бод. Если модуль UART хоста не может сгенерировать сигнал прерывания, то прерывание можно имитировать переключением на более низкую скорость передачи данных и посылкой нулевого байта. Того же результата можно достичь переключением в режим контроля по нулевому биту четности или использованием 9-битного слова, в котором старший бит установлен в 0.

Некоторые из значений задержки в последовательности конфигурации (см. Рис. 2) произвольно взяты большими для того, чтобы удовлетворить требованиям большинства модулей UART. Эти значения могут быть уменьшены.

Считывание регистра скорости передачи данных и запись бита 1-Wire в конце последовательности установки предназначены для контроля корректности выполнения установки микросхемы DS2480B. Если ответ на любую из этих операций будет недостоверным, то установка считается неудачной.

Заметим, что при данной реализации не производится проверка незапрашиваемого байта от микросхемы DS2480B, извещающего о наличии импульса присутствия на шине. Поэтому любая из операций 1-Wire может получить байт ответа неверного формата, что приводит к вызову функции *DS2480B_Detect*. Поскольку любое приложение 1-Wire, которое используется в среде с непостоянными подключениями, где могут возникнуть эти незапрашиваемые байты извещения об импульсе присутствия, должно предусматривать повторное выполнение операций, это не является проблемой.

Рис. 2. Блок-схема функции DS2480B_Detect



DS2480B_ChangeBaud

Для изменения скорости обмена данными между ХОСТОМ и драйвером DS2480B необходимо произвести запись в регистр RBR (RS232 Baud Register — регистр скорости передачи данных порта RS232). После этого микросхема DS2480B сразу же посылает ответ на новой установленной скорости передачи данных. Как правило, этот ответ не обрабатывается хостом (пропускается). Соответственно, в рекомендованной блок-схеме (см. Рис. 3) байт ответа подавляется и игнорируется.

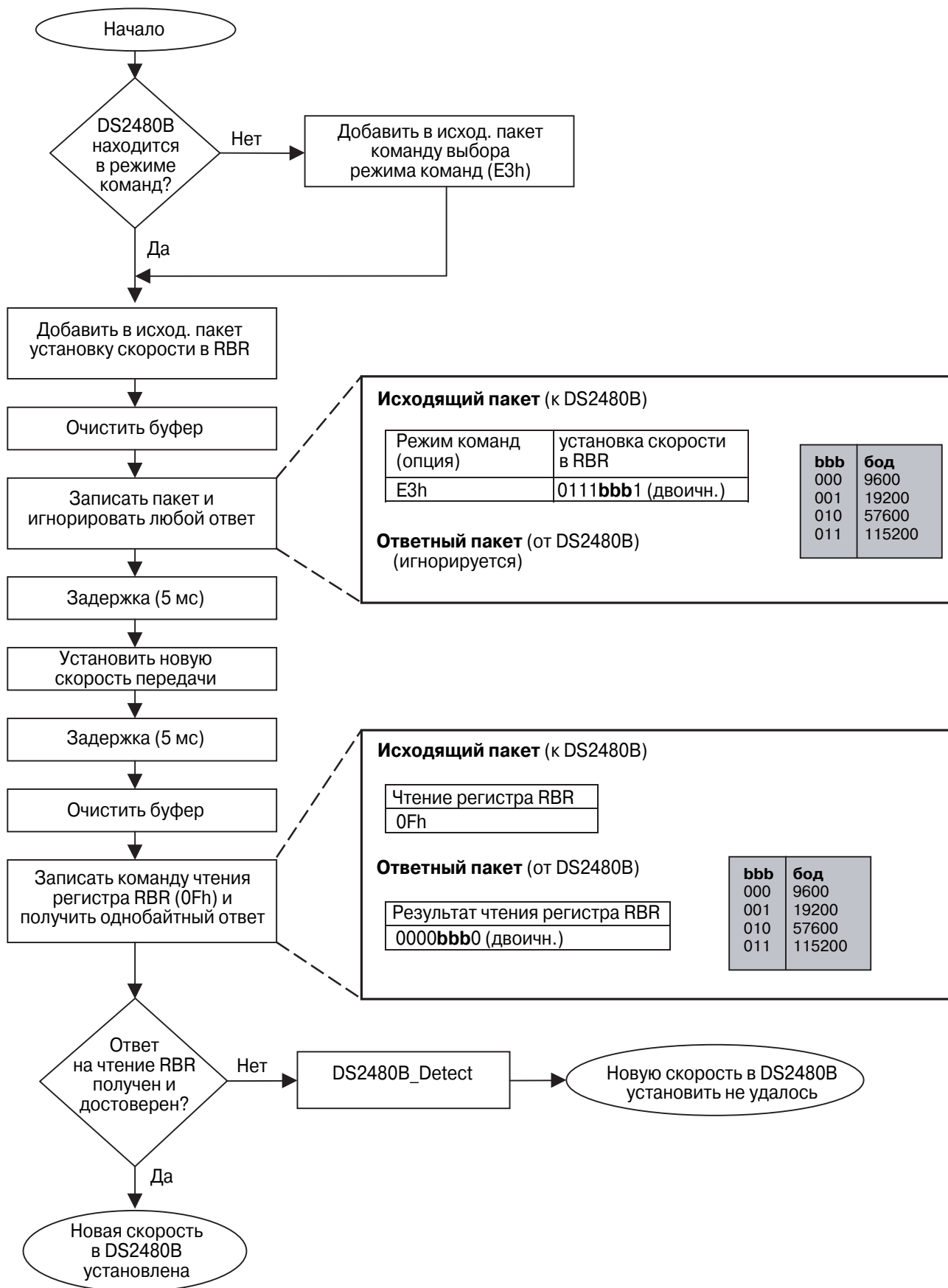
После того как и хост, и DS2480B установили новую скорость передачи данных, производится чтение регистра RBR для проверки завершения операции изменения скорости. Если скорость передачи данных в DS2480B установлена неверно, то осуществить чтение регистра не удастся и будет вызвана последовательность инициализации *DS2480B_Detect*.

Микросхема DS2480B может работать на четырех различных скоростях передачи данных: 9600, 19 200, 57 600 и 115 200 бод. Поскольку DS2480B имеет однобайтный входной буфер, выполнение посланной ей команды должно быть завершено до получения следующей команды. В Табл. 7 спецификации на микросхему DS2480B показано, какие команды на каких скоростях могут выполняться без риска наложения команд. Рекомендуется всегда принимать и обрабатывать результат сброса шины 1-Wire перед переходом к выполнению следующей команды. Поэтому не следует включать операцию сброса в непрерывный поток вместе с остальными байтами. В описываемой реализации сброс шины 1-Wire вынесен в отдельную функцию *OWReset*, так что данная проблема решена. Подобным образом в отдельные функции вынесены также операции с одиночными битами и байтами: *OWReadBit*, *OWWriteBit*, *OWReadByte* и *OWWriteByte*. Поскольку они не объединяются в общий поток с другими командами, для их выполнения можно использовать максимальную скорость. Заметим также, что в описываемой реализации при осуществлении обмена данными по шине 1-Wire с обычной скоростью используется исключительно режим гибкой скорости с расширенной битовой синхронизацией. Это влияет на допустимые скорости передачи данных. Рекомендуемые скорости передачи приведены в Табл. 4. Для упрощения в описываемой реализации будут использоваться только две скорости обмена данными: 9600 бод для операций, выполняемых на обычной (гибкой) скорости, и 115 200 бод — на повышенной скорости (кроме функций поиска).

Таблица 4. Максимальные потоковые скорости передачи данных

Тип функции	Скорость (бод)	
	Обычная (гибкая)	Повышенная
Поиск (OWSearch)	9600	57 600
Команда (все операции, не связанные с шиной 1-Wire)	115 200	115 200
Данные (OWBlock)	9600	115 200

Рис. 3. Блок-схема функции DS2480B_ChangeBaud



ОПЕРАЦИИ 1-WIRE

Базовые и расширенные операции 1-Wire создают общий интерфейс 1-Wire, который обеспечивает выполнение любых операций с любыми приборами 1-Wire.

Реализации для каждой из этих операций 1-Wire имеют некоторые общие характеристики. Везде, где это возможно, команды и данные группируются вместе, чтобы сократить число пакетов, пересылаемых между хостом и DS2480B. Текущий режим работы микросхемы DS2480B запоминается как состояние, поэтому пакет может начинаться с команды изменения режима. Если ответный пакет имеет некорректную длину или несоответствующий формат, осуществляется вызов функции *DS2480B_Detect*.

Изменение скорости последовательного обмена данными между хостом и микросхемой DS2480B осуществляется только в функции *OWSpeed* при изменении скорости обмена данными по шине 1-Wire. В приводимой реализации функция *OWSearch* не может выполняться в ускоренном режиме. Поэтому при выполнении данной функции может быть добавлена простая операция проверки скорости обмена данными и ее снижения до 57 600 бод, если была установлена более высокая скорость обмена данными.

Перед выполнением любых операций 1-Wire необходимо убедиться в том, что на шине установлен обычный режим питания (схема мощной подтяжки отключена). Поэтому подразумевается, что выполнение каждой операции 1-Wire начинается с вызова функции *OWLevel(normal)*, не указанной в приводимых блок-схемах.

OWReset

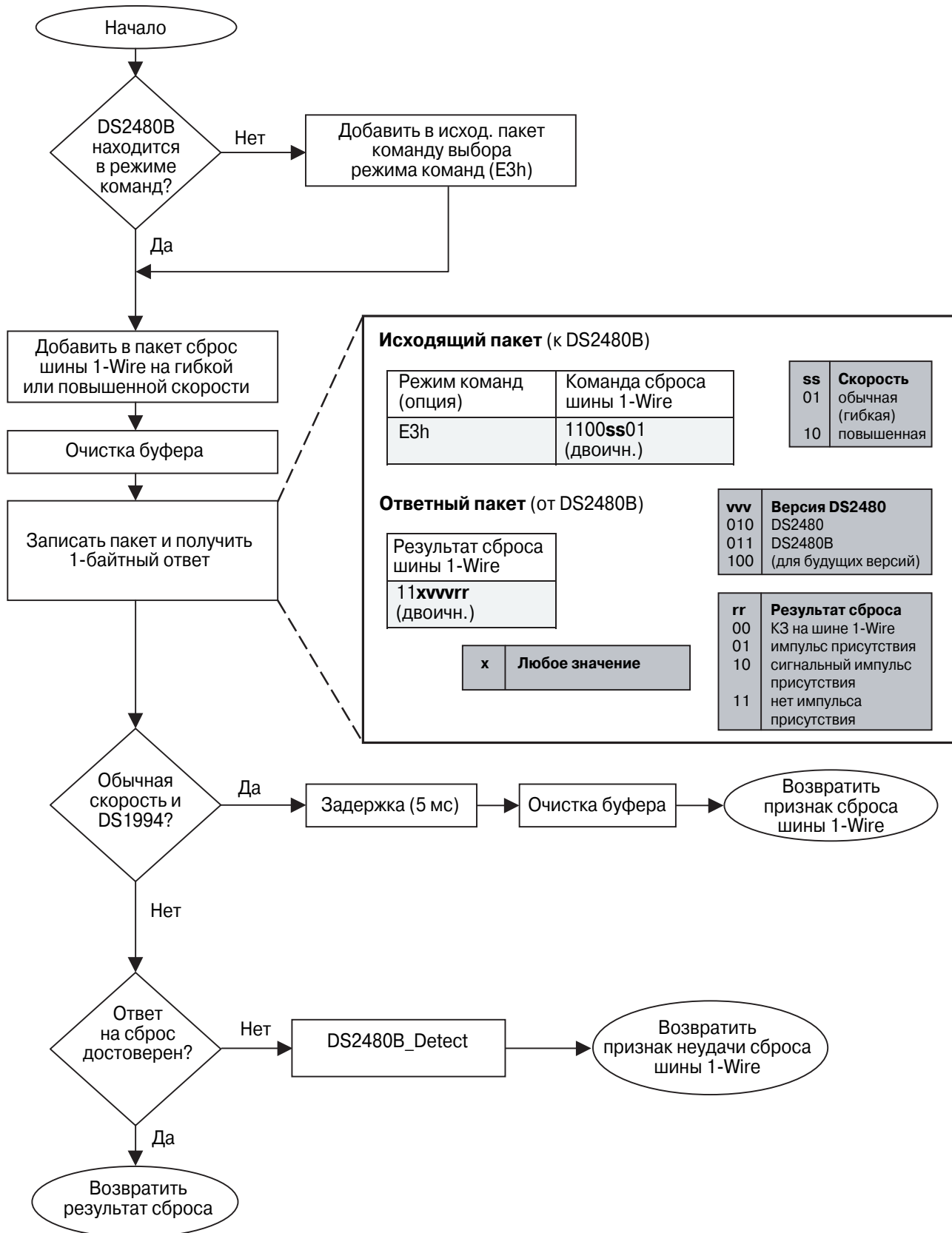
Функция *OWReset* дает команду микросхеме DS2480B послать на шину 1-Wire импульс сброса и осуществить выборку (считывание состояния) шины для обнаружения импульсов присутствия от ведомых приборов 1-Wire. Хотя основным назначением этой функции является выполнение операции сброса, она также возвращает различную полезную информацию. Например, в байте ответа имеется трехбитное поле, где указывается код версии микросхемы. Содержимое этого поля для всех микросхем типа DS2480B будет одинаковым, однако оно может использоваться для опознавания предыдущей версии данного моста — микросхемы DS2480. (Приведенная в настоящем документе реализация совместима с микросхемой DS2480.) Кроме того, это поле позволит сделать программное или аппаратное обеспечение хоста, по крайней мере, частично совместимым с будущими версиями моста.

Выполнение команды сброса шины 1-Wire влияет на скорость обмена данными. Заметим, что когда обмен данными на шине 1-Wire происходит на обычной скорости, в данной реализации для микросхемы DS2480B используется режим гибкой скорости.

Время, требуемое для выполнения рассматриваемой операции, зависит от того, имеется ли на шине сигнальный импульс присутствия. Это является основной причиной, по которой данная операция не объединяется в один пакет с другими операциями 1-Wire. Обратите внимание на необходимость дополнительной задержки (5 мс) и очистки буфера в том случае, если на шине присутствует прибор DS1994 или DS2404. Микросхема DS2480B не обрабатывает ни один из сигнальных импульсов сброса шины 1-Wire, генерируемых этими приборами, поэтому обмен данными должен быть задержан на время, необходимое для завершения их передачи.

Блок-схема функции *OWReset* приведена на Рис. 4.

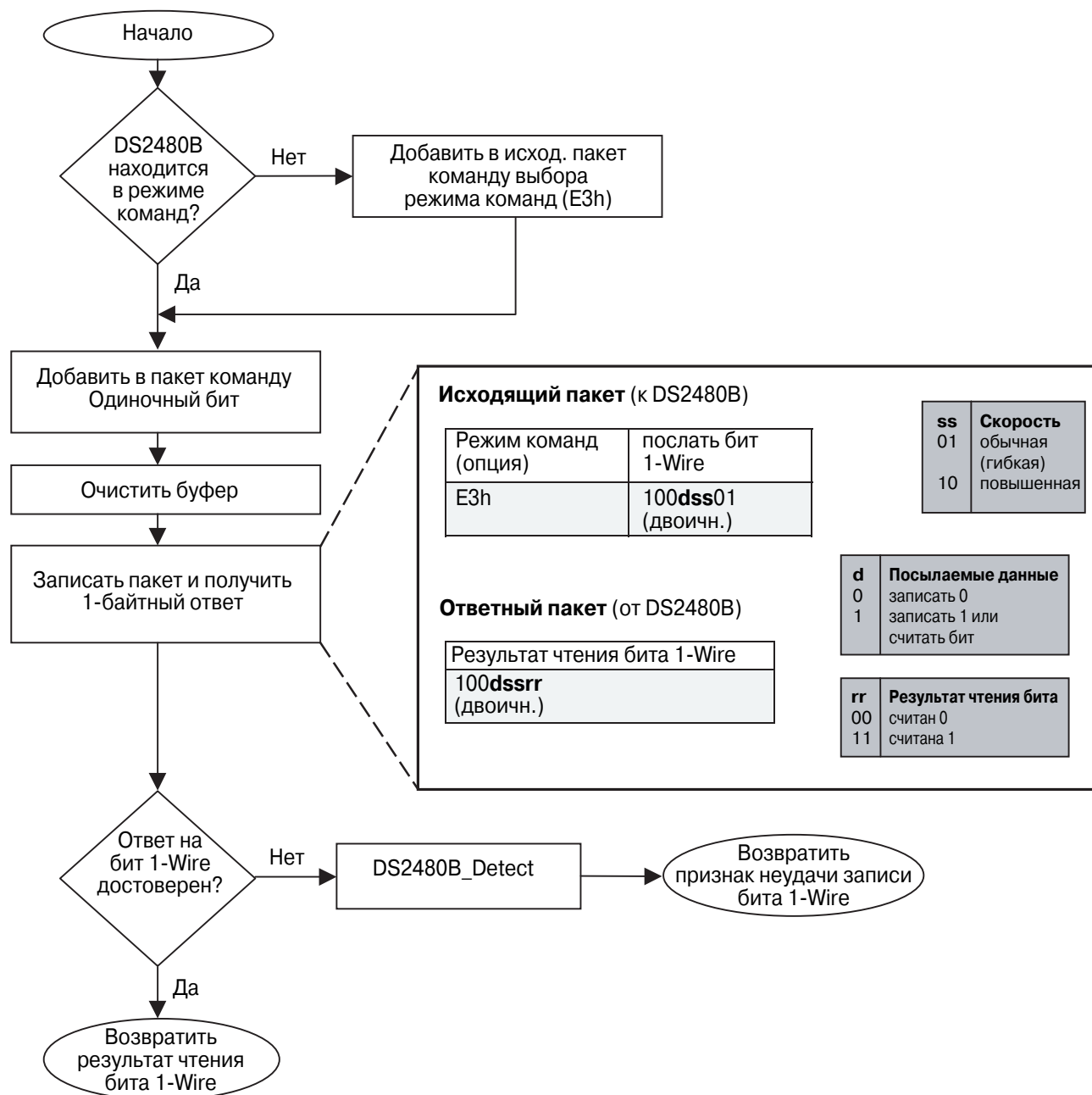
Рис. 4. Блок-схема функции OWRReset



OWriteBit / OReadBit

Операции с одиночными битами на шине 1-Wire в приложениях обычно не используются и описываются здесь для полноты. Если по протоколу требуется запись бита, его значение просто записывается (выдается) на шину 1-Wire, как показано на Рис. 5. Если требуется чтение, то осуществляется запись единицы и выборка результата считывания.

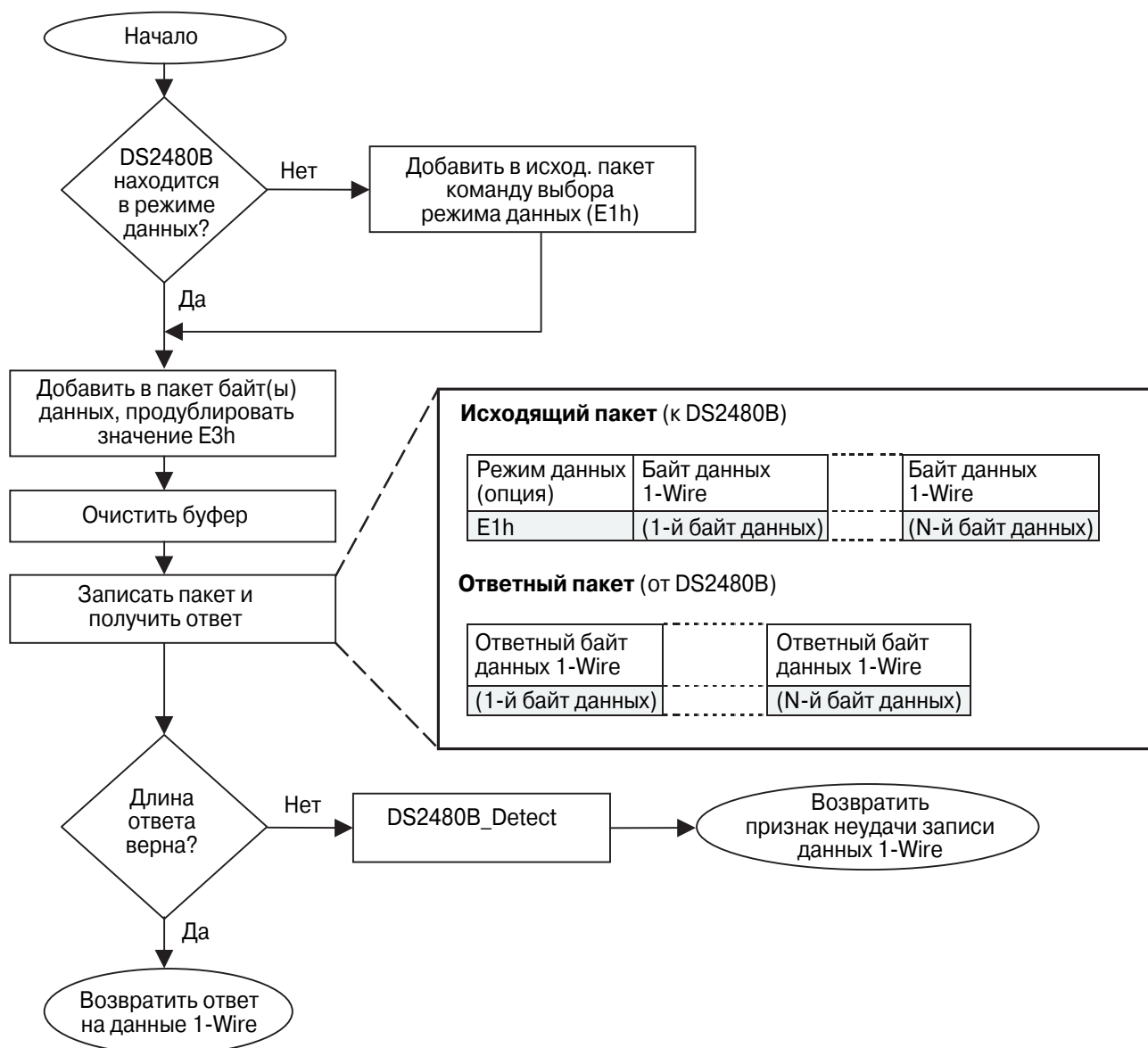
Рис. 5. Блок-схема функций OWriteBit / OReadBit



OWWriteByte / OWReadByte / OWBlock

Операции с одиночными байтами и с группами байтов очень похожи. Сначала необходимо перевести микросхему DS2480B в режим данных. Аналогично командам, работающим с одиночными битами, когда согласно протоколу требуется записать байт, просто производится запись данных. Чтение байта осуществляется путем записи FFh (все «1»), а затем выборки данных, которые и являются результатом считывания. Блочная операция представляет собой группу операций с одиночными байтами, в которой могут сочетаться операции записи и чтения. Позиции считывания должны быть предварительно заполнены значением FFh. Обратите внимание на то, что необходимо всегда дублировать байты данных, имеющие то же самое значение, что и команда переключения микросхемы в режим команд (E3h). В этом случае микросхема DS2480B воспримет их как данные и не перейдет в режим команд.

Рис. 6. Блок-схема функций OWWriteByte / OWReadByte / OWBlock



OWSearch

Алгоритм поиска представляет собой поиск по двоичному дереву, в котором переход по ветви осуществляется до тех пор, пока не будет найден идентификатор ПЗУ прибора (лист). Последующие операции поиска производятся по другим ветвям, пока не будут обнаружены все имеющиеся листья, то есть приборы.

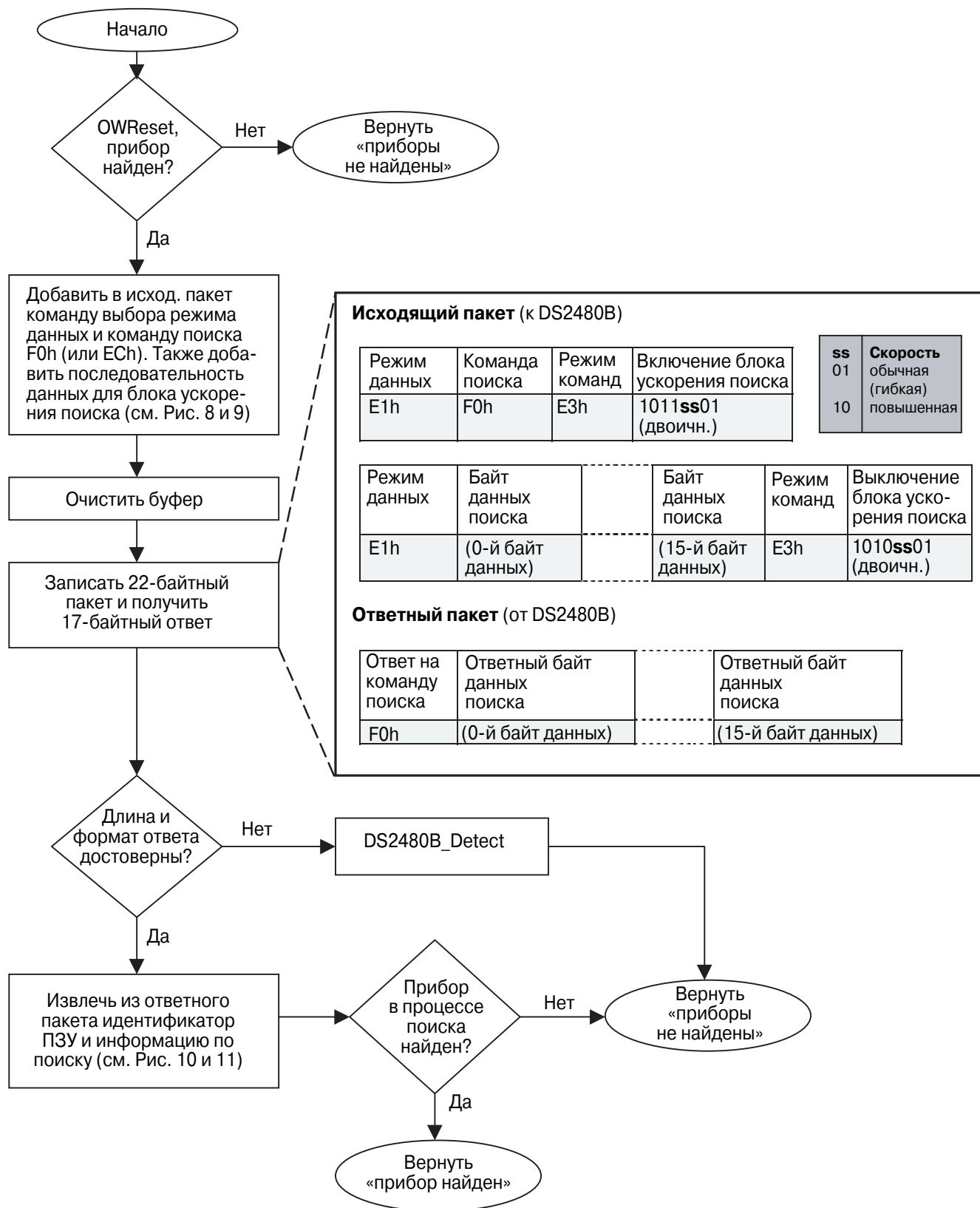
Алгоритм поиска начинается со сброса всех приборов, подключенных к шине 1-Wire, используя последовательность сброса и обнаружения импульса присутствия (см. функцию *OWReset*). Если эта операция выполнена успешно, посылается однобайтная команда поиска (обычная F0h или сигнальная ECh). Команда поиска подготавливает приборы 1-Wire к началу операции поиска.

После передачи этой команды процесс поиска начинается с того, что все участвующие приборы одновременно посылают на шину первый бит (младший) своих идентификаторов ПЗУ (называемых также регистрационными номерами). Как и при всех обменах данными по шине 1-Wire, каждый бит иницируется мастером шины, независимо от того, будут ли данные считываться из ведомого прибора или записываться в него. Поскольку все приборы отвечают одновременно, результат будет представлять собой «логическое И» посылаемых битов. После передачи приборами первого бита своих идентификаторов ПЗУ мастер иницирует следующий бит, и приборы посылают дополнение первого бита. Если оба бита равны 0, значит, в данной позиции бита имеются как нули, так и единицы. Такая ситуация называется несовпадением и является точкой ветвления поиска. Затем мастер шины 1-Wire записывает бит, значение которого определяет направление поиска. Если идентификатор ПЗУ прибора имеет это значение бита, данный прибор будет продолжать участвовать в процессе поиска, а все остальные приборы перейдут в режим ожидания. Эта процедура («считать два бита», «записать один бит») затем повторяется для оставшихся 63 битов идентификатора ПЗУ. Подробно процесс поиска на шине 1-Wire и опции выборочного поиска описаны в руководстве по применению AN187 «*1-Wire Search Algorithm*» («Алгоритм поиска в сети 1-Wire») (<http://pdfserv.maxim-ic.com/arpdf/AppNotes/app187.pdf>). Базовые операции поиска включают поиск всех приборов в сети 1-Wire. Операции выборочного поиска позволяют осуществлять поиск приборов 1-Wire только конкретной группы (семейства).

Большая часть процесса поиска на шине 1-Wire выполняется микросхемой DS2480B. Блок-схема последовательности поиска приведена на Рис. 7. Исходящие данные для поиска формируются на основе результатов последнего поиска (см. Рис. 8 и 9), выполняется поиск, а затем анализируются ответные данные, полученные в результате поиска (см. Рис. 10 и 11).

Необходимо соблюдать осторожность и не выполнять функцию *OWSearch* в ускоренном режиме, используя приводимую реализацию функции *OWSpeed*, поскольку при этом устанавливается скорость 115 200 бод. При такой скорости произойдет переполнение входного буфера микросхемы DS2480B. Функцию можно легко модифицировать, чтобы при выполнении ускоренного поиска снизить скорость передачи данных до 57 200 бод.

Рис. 7. Блок-схема функции OWSearch



Состояние поиска на шине 1-Wire между операциями поиска должно сохраняться, чтобы можно было найти последующие приборы. Термины, характеризующие состояние поиска, приведены в Табл. 5 и совпадают с терминами, используемыми в руководстве по применению AN187 «1-Wire Search Algorithm».

Таблица 5. Состояние поиска

Термин	Описание
id_bit_number	Номер бита ПЗУ (от 1 до 64) прибора, который ищется в данный момент
LastDeviceFlag	Флаг, показывающий, что в предыдущем поиске был найден последний прибор
LastDiscrepancy	Индекс (номер) бита, определяющий, с какого бита должна начаться проверка несовпадения при (следующем) поиске
LastFamilyDiscrepancy	Индекс (номер) бита, идентифицирующий LastDiscrepancy в первом 8-битном групповом коде идентификатора ПЗУ
ROM_ID	8-байтный буфер, содержащий текущий обнаруженный регистрационный номер (идентификатор) ПЗУ
search_direction	Значение бита, указывающее направление поиска. Все приборы, имеющие бит с этим значением, продолжают участвовать в процессе поиска, а остальные переходят в режим ожидания до сброса шины 1-Wire

Формирование данных для поиска на шине 1-Wire

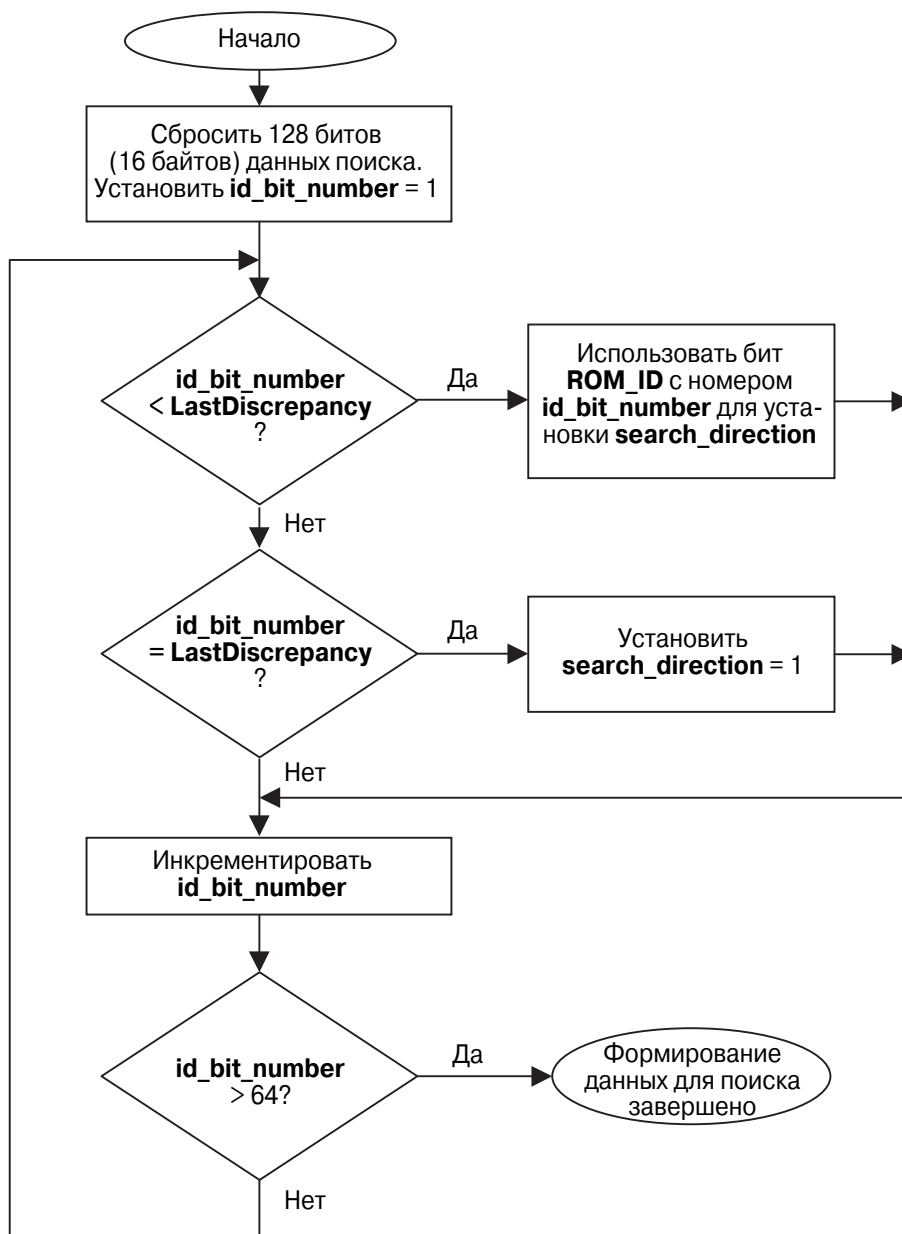
Шестнадцать байтов входных данных для поиска на шине 1-Wire можно рассматривать как 128 битов данных. Данные сгруппированы в 64 двухбитные пары. Первый бит не используется и должен быть равен 0. Второй бит указывает направление поиска в случае обнаружения несовпадения. Если несовпадения не обнаружено, то DS2480B автоматически будет продолжать процесс поиска по единственному возможному пути. При формировании исходящих данных биты направления поиска следует установить в соответствии с битами ROM_ID, полученными в результате предыдущего поиска, вплоть до позиции бита, в котором было обнаружено последнее несовпадение. В этой позиции бит направления поиска следует установить в 1, а все последующие биты — в 0. Формат данных приведен на Рис. 8, а блок-схема формирования пакета — на Рис. 9.

Рис. 8. Исходящие данные для поиска

128	127	...	L**	L - 1	...	4	3	2	1
0	0		1	0		r2	0	r1*	0

* Биты ROM_ID, соответствующие id_bit_number из предыдущего поиска
 ** Позиция бита LastDiscrepancy

Рис. 9. OWSearch: формирование исходящих данных для поиска



Анализ данных, полученных в результате поиска на шине 1-Wire

Данные, полученные от микросхемы DS2480B в результате поиска на шине 1-Wire, также представляют собой 16 байтов в виде 64 двухбитных пар. Первый бит в каждой паре является флагом, показывающим, было ли в этой позиции обнаружено несовпадение, что потребовало использования предусмотренного бита направления поиска. Второй бит в каждой паре указывает выбранное направление поиска и представляет собой бит идентификатора ПЗУ прибора, найденного в результате поиска. Формат этих данных представлен на Рис. 10. Флаги несовпадения и выбранное направление поиска анализируются для установки состояния поиска, как показано на Рис. 11.

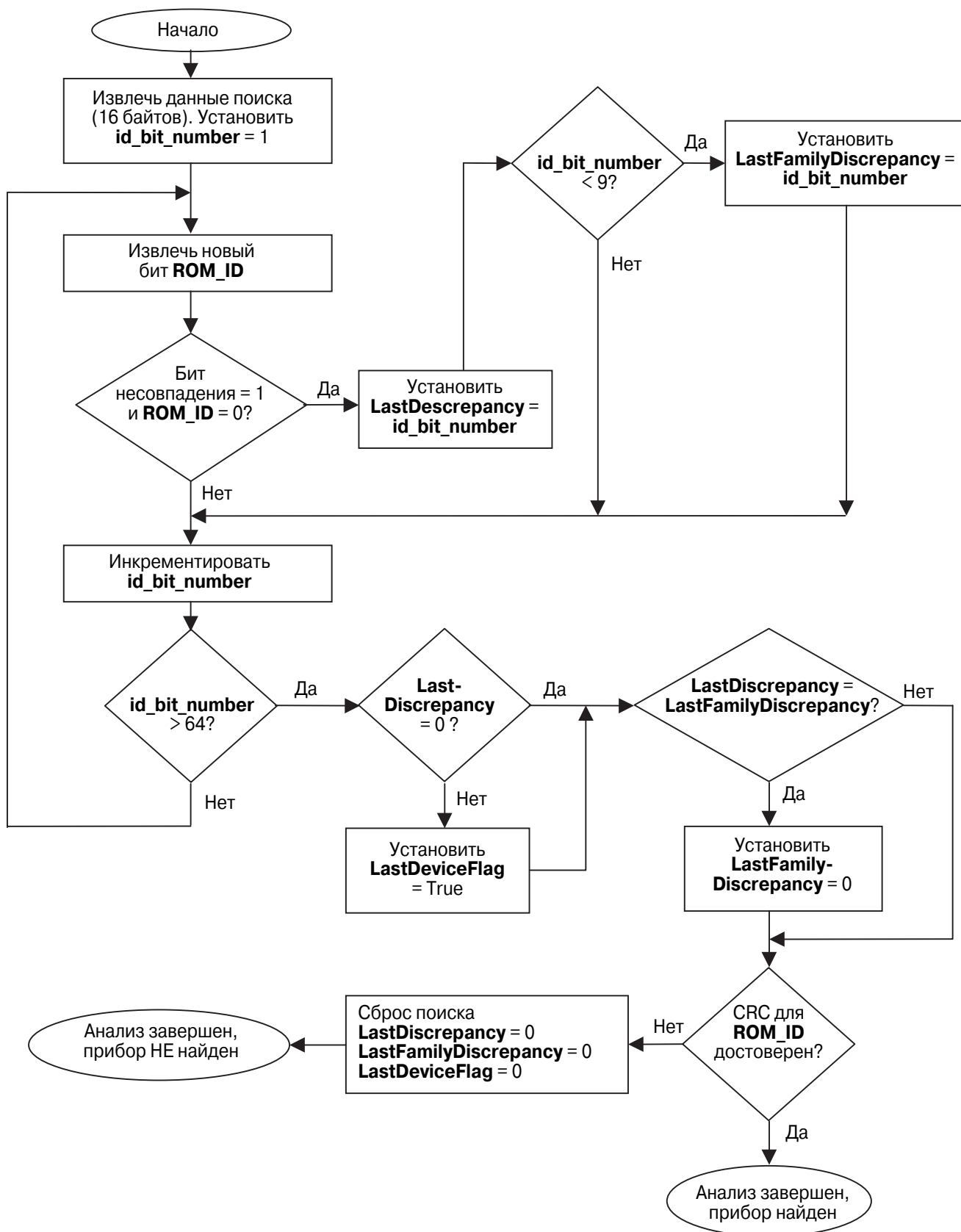
Рис. 10. Ответные данные поиска

128	127	...	4	3	2	1
r64	d64		r2	d2	r1*	d1**

* Новые биты **ROM_ID**, соответствующие позициям **id_bit_number** из данного поиска

** 1, если обнаружено несовпадение в этой позиции бита; самая старшая позиция, содержащая 0, является новым значением **LastDiscrepancy**

Рис. 11. OWSearch: анализ ответных данных поиска



OWSpeed

Для того чтобы воспользоваться преимуществами более высокой пропускной способности шины 1-Wire при работе на повышенной скорости, необходимо увеличить скорость передачи данных при последовательном обмене между хостом и мостом DS2480B. В данной реализации используются только две скорости передачи данных: 9600 бод в обычном режиме и 115 200 бод — в ускоренном (кроме операций поиска). Переключение микросхемы DS2480B на новую скорость шины 1-Wire осуществляется при выполнении операции выключения блока ускорения поиска. Благодаря такому решению корректное значение скорости будет использоваться даже в том случае, если следующим действием на шине 1-Wire будет обмен байтами в режиме данных. Блок-схема функции *OWSpeed* приведена на Рис. 12.

OWLevel

Основным назначением функции *OWLevel* является отключение схемы мощной подтяжки, подключенной ранее для подачи питания, которое было инициировано в результате вызова функции *OWReadBitPower* или *OWWriteBytePower*. Дополнительным назначением данной функции является ручное включение подачи питания с помощью мощной подтяжки без активирования схемы подтяжки, хотя такая возможность используется нечасто. Эта функция может быть вызвана как *OWLevel(normal)* или как *OWLevel(power)*. Заметим, что вызов функции *OWLevel(normal)* предполагается в начале выполнения всех операций 1-Wire, чтобы обеспечить обычный режим питания шины. Для отключения генерируемого в данный момент импульса неопределенной длительности при активированной схеме подтяжки, необходимо завершить формирование импульса, начать генерирование нового импульса без активирования подключения схемы подтяжки, а затем завершить и этот импульс. В результате вызова функции *OWLevel(power)* начинается генерирование нового импульса неопределенной длительности (схема подтяжки при этом не активизируется). Блок-схема функции *OWLevel* приведена на Рис. 13.

OWProgramPulse

Функция *OWProgramPulse* обеспечивает поступление на шину 1-Wire импульса программирования 12 В. Данная операция используется для программирования приборов памяти ППЗУ (однократно программируемых) 1-Wire. При выполнении функции проверяется наличие напряжения программирования. Блок-схема функции *OWProgramPulse* приведена на Рис. 14.

Рис. 12. Блок-схема функции OWSpeed

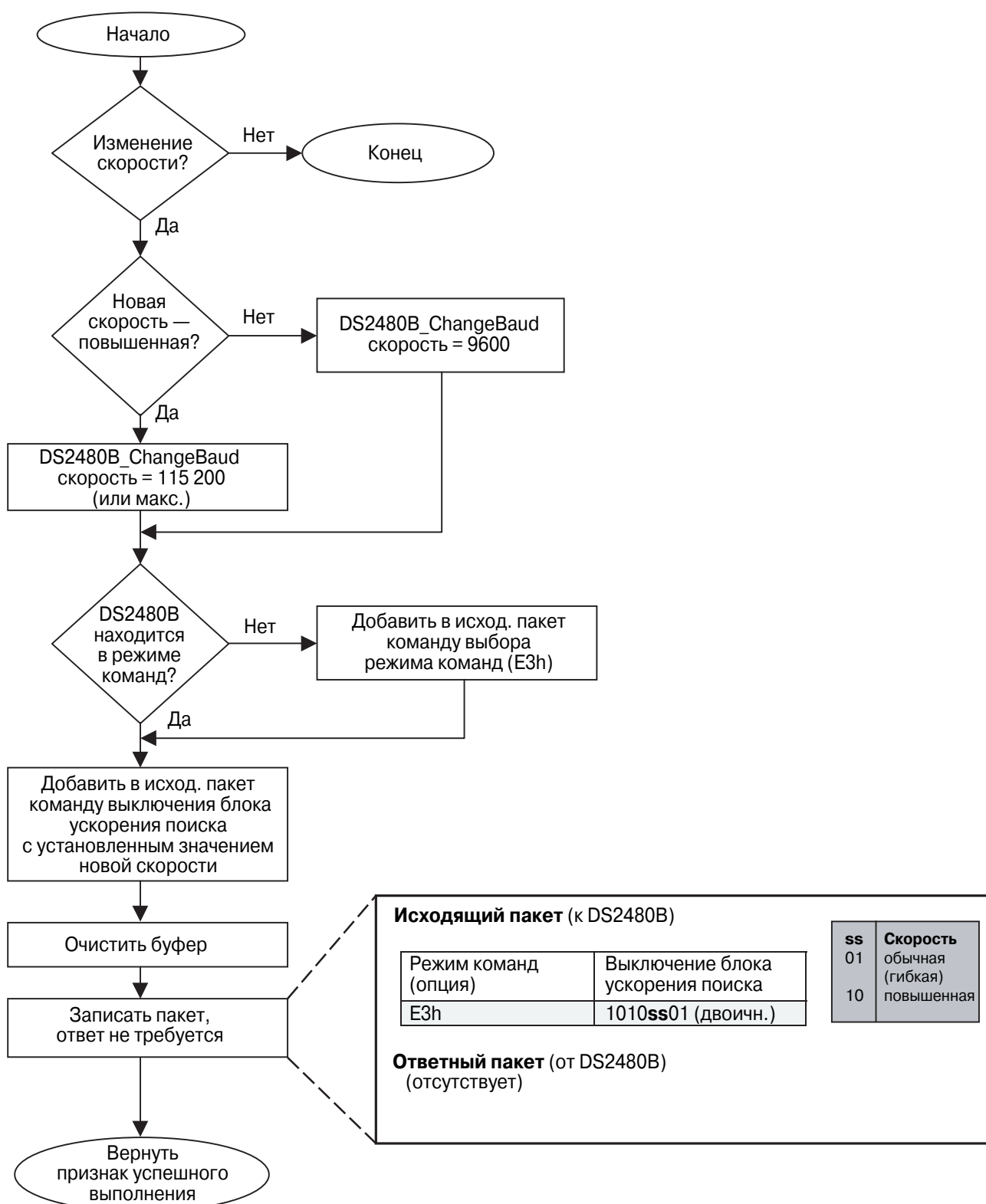


Рис. 13. Блок-схема функции OWLevel

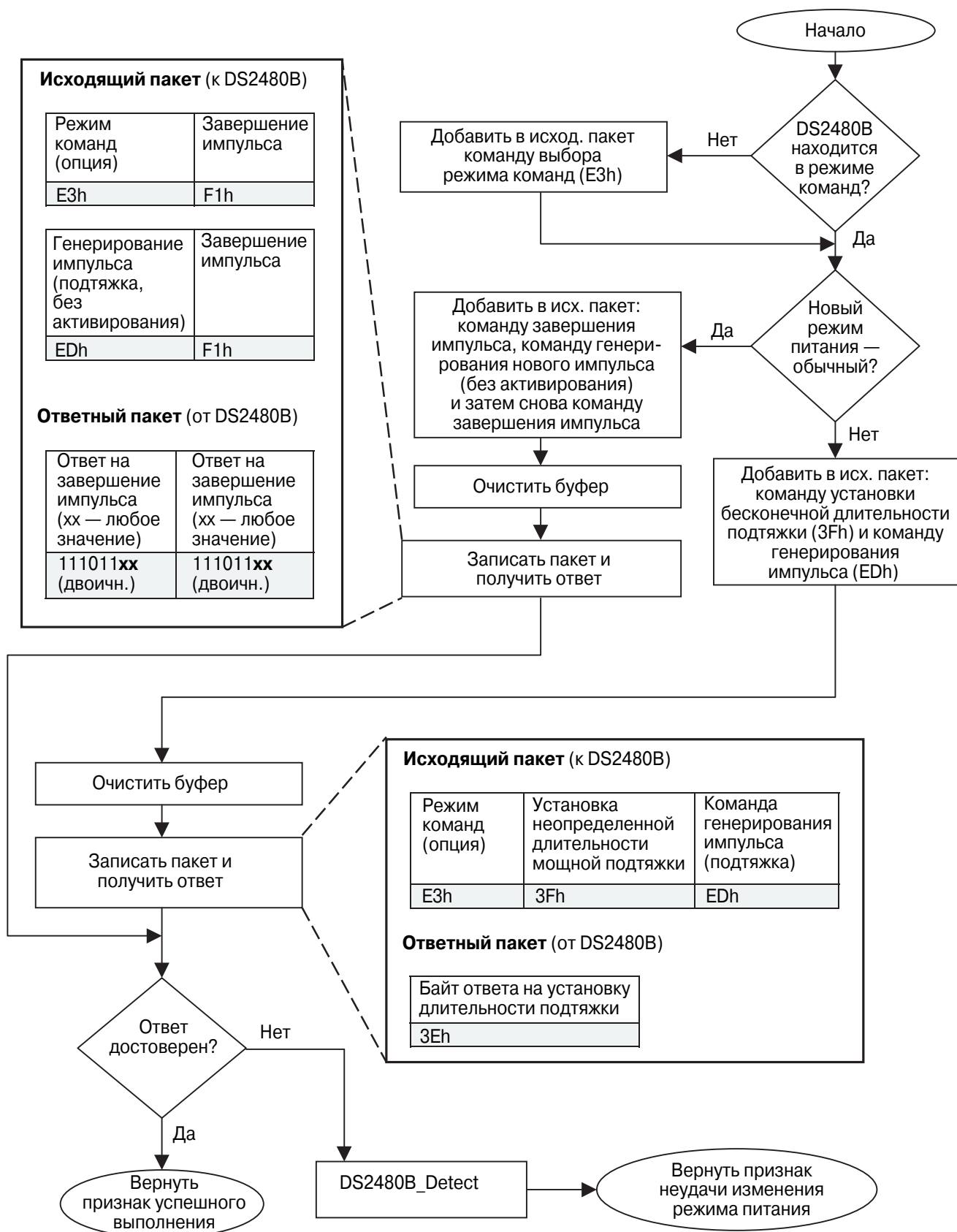
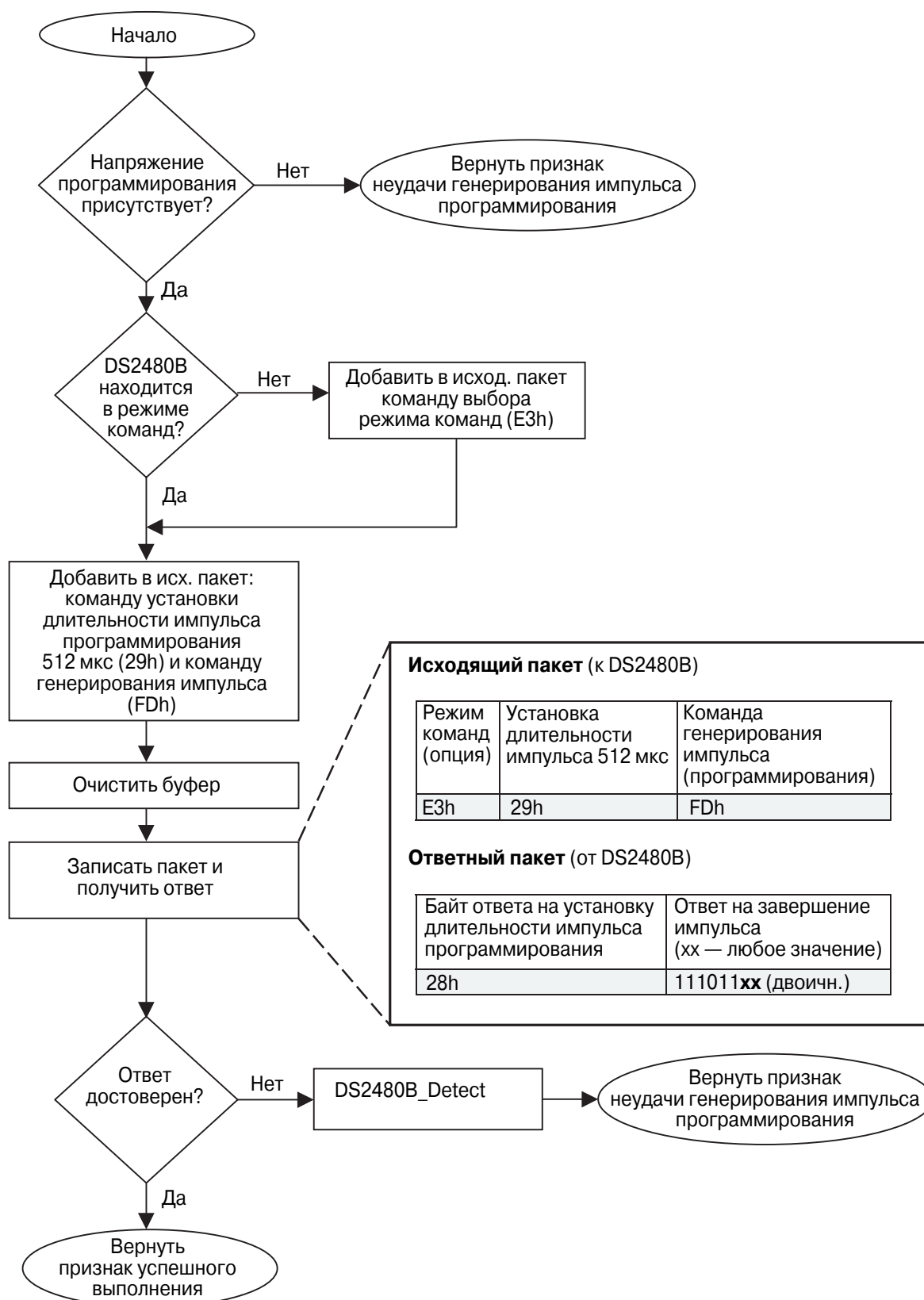


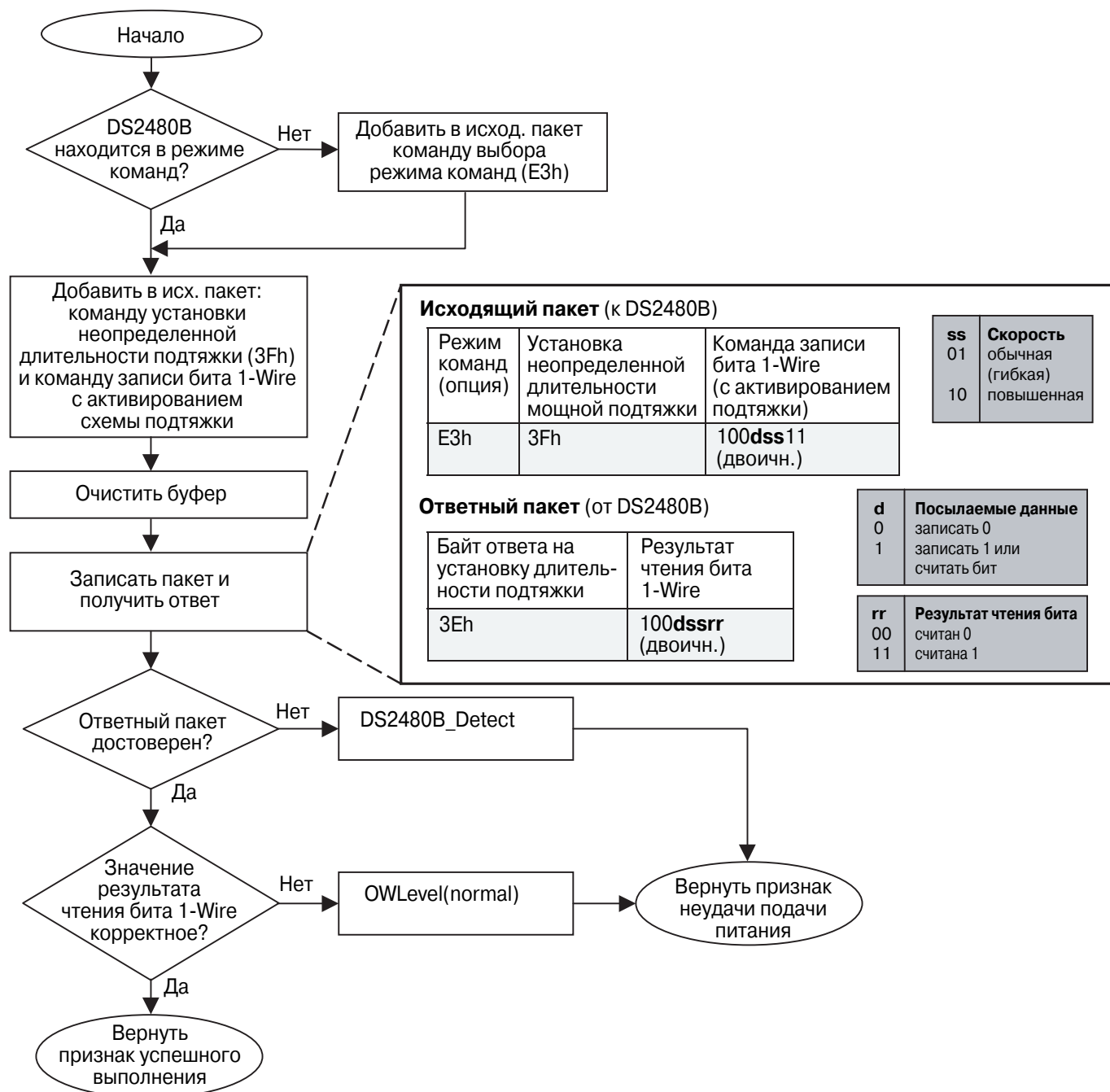
Рис. 14. Блок-схема функции OWProgramPulse



OWReadBitPower

Функция *OWReadBitPower* используется исключительно с приборами iButton®, поддерживающими Java. В этих приборах iButton реализована отключающая последовательность, которая должна обеспечивать подачу питания сразу же после бита подтверждения. Если бит подтверждения имеет неверное значение, подача питания прекращается и операция считается невыполненной. Если использование приборов iButton, поддерживающих Java, не предполагается, то реализовывать данную функцию не требуется. Для прекращения подачи питания после выполнения рассматриваемой операции, следует вызвать функцию *OWLevel(normal)*. Блок-схема функции *OWReadBitPower* приведена на Рис. 15.

Рис. 15. Блок-схема функции OWReadBitPower

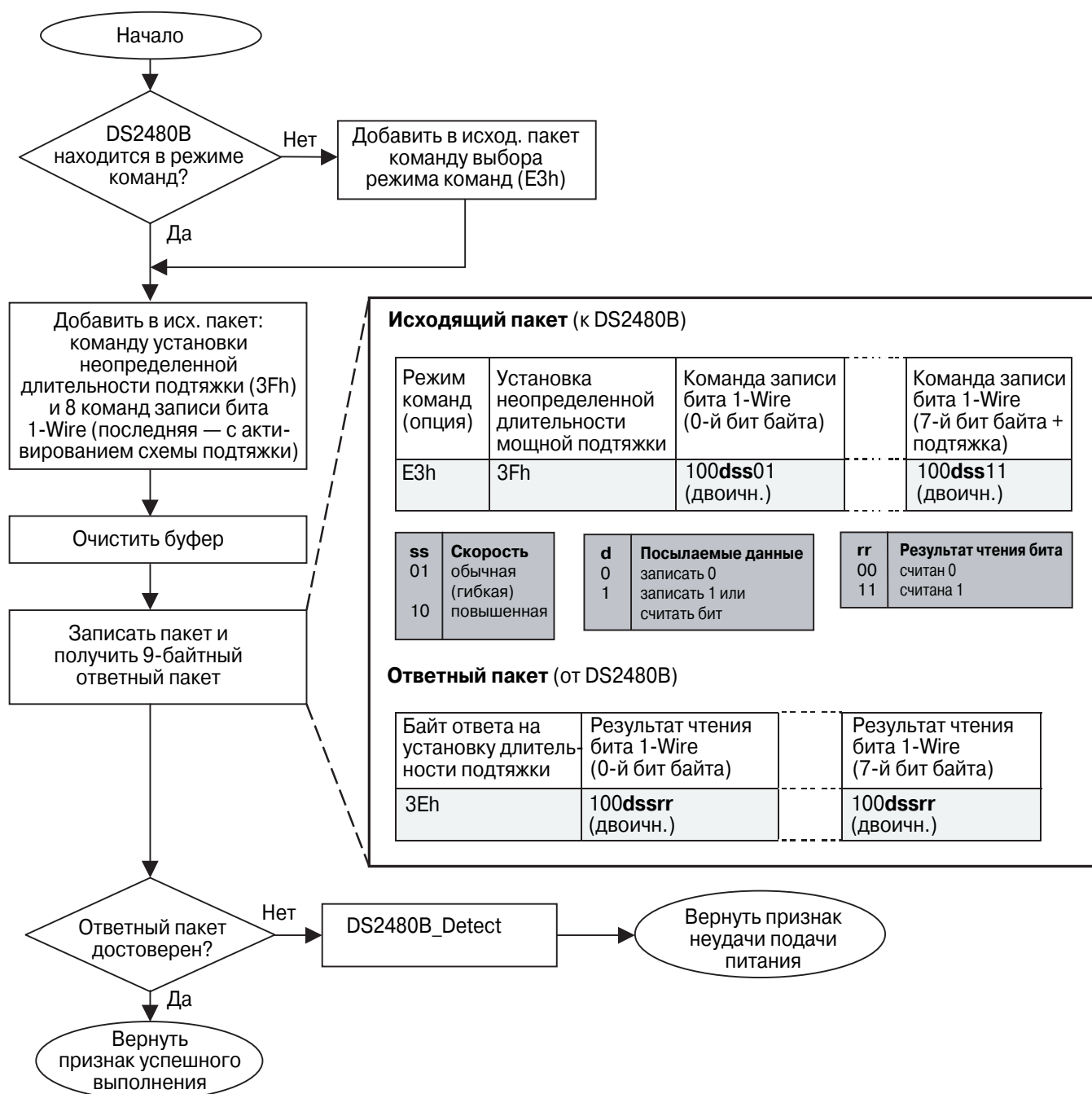


iButton является зарегистрированной торговой маркой Dallas Semiconductor.

OWWriteBytePower

Функция *OWWriteBytePower* обеспечивает подачу питания путем подключения схемы мощной подтяжки сразу же после записи байта на шину 1-Wire. Это наиболее распространенный способ подачи питания. Например, прибор iButton DS1920 (датчик температуры) имеет однобайтную команду преобразования температуры, для выполнения которого требуется дополнительная энергия после поступления соответствующей команды. Для прекращения подачи питания после выполнения этой операции, следует вызвать функцию *OWLevel(normal)*. Блок-схема функции *OWWriteBytePower* приведена на Рис. 16. Заметим, что записываемый байт преобразуется в восемь операций записи одиночных битов, при этом последний бит активирует подачу питания. Эти действия можно было бы реализовать с помощью однобайтной последовательности, но данная операция специально осуществляется аналогично функции *OWReadBitPower*, чтобы их можно было объединить.

Рис. 16. Блок-схема функции *OWWriteBytePower*



ПРИМЕРЫ

В этом разделе документа приведено несколько примеров обмена данными по шине 1-Wire с использованием базовых и расширенных операций 1-Wire.

Команда ускоренного сравнения, как показано в Примере 2, используется для того, чтобы задействовать возможность работы прибора 1-Wire на повышенной скорости. После успешного завершения этих действий и микросхема DS2480B, и прибор 1-Wire будут работать на повышенной скорости, и можно будет выполнять любые операции 1-Wire, за исключением *OWSearch*.

Пример 2. Псевдокод ускоренного сравнения

```
trans_block – временный буфер передачи, значения представлены в шестнадцатеричной системе

// установить обычную скорость шины 1-Wire
OWSpeed(normal)

// сбросить шину 1-Wire (на обычной скорости)
If OWRReset = TRUE

    // команда ускоренного сравнения
    OWWriteByte(69h)

    // изменить скорость шины 1-Wire на повышенную
    OWSpeed(overdrive)

    // послать идентификатор ПЗУ прибора 1-Wire для завершения сравнения
    // идентификатор ПЗУ содержится в байтах R0...R7
    trans_block = R0,R1,R2,R3,R4,R5,R6,R7
    OWBlock(trans_block)

    // Успех
    ...
Else
    // прибор отсутствует
    ...
EndIf
```

Прибор iButton DS1920 представляет собой датчик температуры, выполняющий преобразование температуры при получении соответствующей команды. Во время преобразования температуры мастер шины 1-Wire должен обеспечивать подачу питания с помощью схемы мощной подтяжки. В Примере 3 приведена последовательность действий при преобразовании температуры с использованием расширенных операций 1-Wire для подачи питания.

Пример 3. Псевдокод преобразования температуры прибором DS1920

```
trans_block – временный буфер передачи, значения представлены в шестнадцатеричной системе

// сбросить шину 1-Wire
If OWRReset = TRUE

    // послать команду Сравнение ПЗУ прибору, который будет считываться,
    // идентификатор ПЗУ содержится в байтах R0...R7
    trans_block = 55,R0,R1,R2,R3,R4,R5,R6,R7
    OWBlock(trans_block)

    // команда преобразования и подача питания
    OWWriteBytePower(44h)

    // время задержки, необходимое для завершения преобразования
    Delay(1000ms)

    // отключить подачу питания
    OWLevel(normal)
```

```

// проверить завершение преобразования
If OWReadByte = FFh
    // Успех
    ...
Else
    // преобразование не завершено, неудача
    ...
EndIf

Else
    // прибор отсутствует
    ...
EndIf

```

Запись в однократно программируемые (One-Time-Programmable — OTP) приборы памяти ППЗУ 1-Wire осуществляется побайтно, используя последовательность действий, приведенную в Примере 4. Для выполнения данной операции к микросхеме DS2480B должно быть подключено напряжение питания 12 В.

Пример 4. Псевдокод программирования ППЗУ DS1986

```

trans_block - временный буфер передачи, значения представлены в шестнадцатеричной системе

// сбросить шину 1-Wire
If OWReset = TRUE

    // послать команду Сравнение ПЗУ прибору, в который будет производиться
    // запись (идентификатор ПЗУ содержится в байтах R0...R7),
    // с командой записи памяти (0F), адресом 0000, данными 66 и считанным CRC16
    trans_block = 55,R0,R1,R2,R3,R4,R5,R6,R7,0F,00,00,66,FF,FF
    OWBlock(trans_block)

    // вычислить CRC16 последних 6 байтов блока для проверки корректности
    // установки адреса/данных
    If CRC16 correct
        // послать импульс программирования
        If Not OWProgramPulse
            // Напряжение программирования отсутствует
            ...
        EndIf

        // считать данные для верификации
        If OWReadByte != 66
            // Успех
            ...
        Else
            // отказ в программировании, страница заблокирована,
            // байт уже запрограммирован?
            ...
        EndIf
    Else
        // ошибка в передаваемых данных и адресе
        ...
    EndIf

Else
    // прибор отсутствует
    ...
EndIf

```


Приложение

Условные обозначения параметров конфигурации

Параметр	Условное обозначение	Код параметра
Скорость спада при подтяжке вниз	PDSRC	001
Длительность импульса программирования	PPD	010
Длительность мощной подтяжки к 5 В	SPUD	011
Длительность интервала НИЗКОГО уровня при записи 1	W1LT	100
Время сдвига момента выборки данных и время восстановления при записи 0	DSO	101
	W0RT	
Порог переключения датчика нагрузки	LOAD	110
Скорость передачи данных порта RS232	RBR	111