

## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5

### 5.1 ОБЗОР

Синхронные последовательные порты (SPORT0 и SPORT1), поддерживают множество протоколов обмена данными через последовательный порт и могут обеспечить прямое соединение процессоров в системе, состоящей из нескольких процессоров. Ниже перечислены процессоры семейства ADSP-2100, имеющие последовательные порты:

<i>Процессор</i>	<i>Количество последовательных портов</i>
ADSP-2101	2
ADSP-2105	1
ADSP-2115	2
ADSP-2111	2
ADSP-2171	2
ADSP-2181	2
ADSP-21msp58/59	2

Последовательные порты SPORT0 и SPORT1 немного отличаются друг от друга, о чем будет более подробно рассказано далее в данной главе. В процессоре ADSP-2105 имеется только один последовательный порт SPORT1.

### 5.2 ОБЩЕЕ ОПИСАНИЕ ПОСЛЕДОВАТЕЛЬНОГО ПОРТА

Аппаратно каждый последовательный порт имеет интерфейс с пятью выводами.

Таблица 5.1

**Описание выводов последовательного порта**

<i>Название выводов</i>	<i>Функция</i>
SCLK	Тактовый сигнал
RFS	Прием кадровых синхроимпульсов
TFS	Передача кадровых синхроимпульсов
DR	Прием данных
DT	Передача данных

Последовательный порт принимает последовательно передаваемые данные на выводе DR и последовательно передает данные через вывод DT. Работая в дуплексном режиме он может одновременно принимать и передавать данные. Эти биты данных синхронны с тактовыми синхроимпульсами генератора SCLK,

## 5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ

которые являются выводом, когда процессор генерирует эти синхроимпульсы, или вводом, когда эти синхроимпульсы генерируются внешне. Сигналы кадровой синхронизации RFS и TFS используются для указания на начало слова последовательно передаваемых данных или потока последовательно передаваемых слов.

На рис. 5.1 показана упрощенная блок-схема последовательного порта. Данные, предназначенные для передачи, записываются через шину ДПД из внутреннего регистра процессора в регистр передачи TX. Эти данные могут сжиматься в схеме компандирования, а затем автоматически передаются в передающий регистр сдвига. Из регистра сдвига биты передаются на вывод последовательного порта DT, начиная с самых старших бит, синхронно с тактовыми синхроимпульсами. Принимающая часть последовательного порта принимает данные с вывода DR, также синхронно с тактовыми синхроимпульсами. После приема одного слова данные расширяются в схеме компандирования, а затем передаются в регистр приема данных RX последовательного порта.

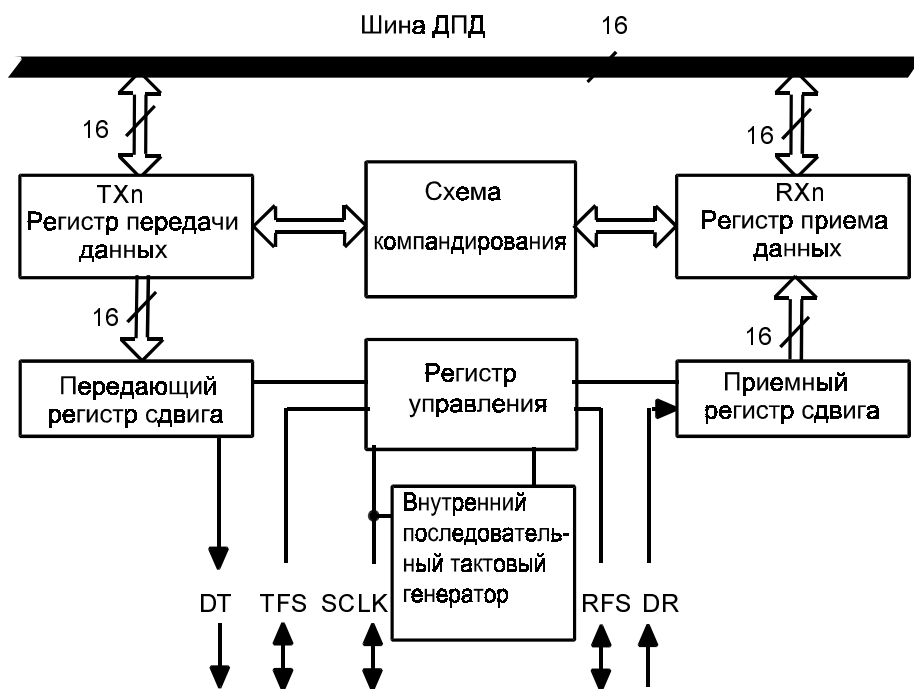


Рис. 5.1 Блок-схема последовательного порта

## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5

Далее рассмотрим свойства последовательного порта. Многие из них обладают перестраиваемой конфигурацией, за счет чего достигается гибкость в работе последовательного порта.

- Двухнаправленность: в каждом последовательном порту принимающая и передающая части независимы друг от друга.
- Двойная буферизация: в каждой части последовательного порта (и принимающей, и передающей) имеется регистр данных для передачи слов данных в процессор и из него, а также регистр сдвига данных. Двойная буферизация сокращает время для обслуживания последовательного порта.
- Синхронизация: каждый последовательный порт может использовать внешние тактовые синхроимпульсы или генерировать свои собственные в широком диапазоне частот от 0 Гц. См. раздел 5.5.
- Длина слова: каждый последовательный порт поддерживает длины слов последовательно передаваемых данных от трех до шестнадцати бит. См. раздел 5.6.
- Кадровая синхронизация: каждая часть последовательного порта (принимающая и передающая) может функционировать с или без использования сигналов кадровой синхронизации для каждого слова данных; с внутренне или внешне генерируемыми сигналами кадровой синхронизации; с активными по высокому или низкому уровню сигналами кадровой синхронизации; с импульсами любой из двух длительностей и стробированием сигналов кадровой синхронизации. См. раздел 5.7.
- Аппаратное компандирование: каждый последовательный порт может выполнять компандирование с А- и  $\mu$ -характеристикой согласно рекомендациям Международного консультативного комитета по телеграфии и телефонии (МККТТ) G.711. См. раздел 5.10.
- Автобуферизация с затратами не более, чем в один цикл: используя генераторы адреса данных, каждый последовательный порт может автоматически принимать и/или передавать целый циклический буфер данных с затратами в один цикл на слово данных. Передачи между последовательным портом и циклическим буфером осуществляются в этом режиме автоматически и не требуют дополнительного программирования. См. 5.11.
- Прерывания: каждая часть последовательного порта (принимающая и передающая) генерирует прерывания по завершению передачи слова данных или после передачи целого буфера при использовании автобуферизации. См. раздел 5.13.
- Многоканальность: SPORT0 может селективно принимать и передавать данные из потока последовательно передаваемых бит с использованием мультиплексирования 24 или 32 каналов с временным разделением информации. Это существенно важно для интерфейсов T1 или может использоваться для сетевой коммуникации

## 5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ

нескольких процессоров. См. раздел 5.12. **Примечание:** ADSP-2105 имеет только один последовательный порт (SPORT1) и не поддерживает режим работы с несколькими каналами.

- Альтернативная конфигурация: SPORT1 может быть сконфигурирован не как последовательный порт, а как два входа прерываний  $\overline{\text{IRQ0}}$  и  $\overline{\text{IRQ1}}$  и сигналы флагов "Flag In" и "Flag Out". Внутренне генерируемые тактовые синхроимпульсы могут, тем не менее, использоваться в данной конфигурации. См. раздел 5.4.

### 5.2.1 Прерывания

Каждый последовательный порт имеет прерывания приема и передачи. Приоритеты этих прерываний показаны в табл. 5.2..

Таблица 5.2

#### Приоритеты прерываний последовательного порта

<i>Высший</i>	Передача SPORT0 (в процессорах с двумя последовательными портами)
	Прием SPORT0 (в процессорах с двумя последовательными портами)
	Передача SPORT1
<i>Низший</i>	Прием SPORT1

Более подробная информация по обработке прерываний дается в разделе "Прерывания" в главе 3 "Управление программой".

### 5.2.2 Работа последовательного порта

После записи данных в регистр TX последовательного порта этот порт готов для передачи; побитовая передача данных инициализируется сигналом TFS. После начала передачи каждое значение, записанное в регистр TX, подается на внутренний передающий регистр сдвига, откуда передается последовательно по битам, начиная с самого старшего бита. Каждый бит сдвигается по переднему фронту тактового синхроимпульса.

После окончания передачи первого бита слова (самого старшего бита) последовательный порт генерирует прерывание передачи. Несмотря на то, что передача первого слова еще продолжается, становится возможной запись нового слова данных в регистр TX.

В принимающей части последовательного порта биты накапливаются в том порядке, как они поступают во внутренний регистр приема. По окончании приема одного целого слова оно записывается в регистр RX, и последовательный порт генерирует прерывание приема.

## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5

При разрешенном режиме автобуферизации прерывания генерируются несколько отличным образом (см. раздел "Автобуферизация" данной главы).

### 5.3 ПРОГРАММИРОВАНИЕ ПОСЛЕДОВАТЕЛЬНОГО ПОРТА

С точки зрения программиста, последовательный порт можно представить состоящим из двух секций. Секция конфигурации представляет собой блок управляющих регистров (отраженных в карте памяти данных), которые должны быть инициализированы программой перед тем, как использовать последовательные порты. Секцию данных образует регистровый файл, который используется для передачи и приема значений через последовательный порт.

#### 5.3.1 Конфигурация последовательного порта

Конфигурация последовательного порта формируется при помощи установки соответствующих битов в регистрах конфигурации. Эти регистры отображены в карте памяти данных. Регистры конфигурации SPORT0 занимают ячейки с 0x3FF3 по 0x3FFA; регистры конфигурации SPORT1 занимают ячейки с 0x3FEF по 0x3FF2. Содержимое этих регистров представлено в табл. 5.3 и в обзоре регистров в Приложении Е. Последствия различных установок подробно разъясняются в следующих далее разделах.

Таблица 5.3

**Регистры конфигурации последовательного порта**

<i>Адрес</i>	<i>Содержимое</i>
0x3FFA	Разрешение многоканального приема слова через SPORT0* (31-16)
0x3FF9	Разрешение многоканального приема слова через SPORT0* (15-0)
0x3FF8	Разрешение многоканальной передачи слова через SPORT0* (31-16)
0x3FF7	Разрешение многоканальной передачи слова через SPORT0* (15-0)
0x3FF6	Управляющий регистр SPORT0* Управление многоканальным режимом работы Источник тактовых синхроимпульсов Управление кадровой синхронизацией Режим компандирования Длина последовательно передаваемого слова

---

\* Регистры конфигурации SPORT0 определены только в процессорах с двумя портами.

## 5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ

табл. 5.3 (продолжение)

1	2
0x3FF5	Коэффициент деления тактовых синхроимпульсов SPORT0* (определяет частоту)
0x3FF4	Коэффициент деления кадровых синхроимпульсов приема SPORT0* (определяет частоту)
0x3FF3	Регистр управления автобуферизацией SPORT0*
0x3FF2	Регистр управления SPORT1 Значение флага Источник тактовых синхроимпульсов Управление кадровой синхронизацией Режим компандирования Длина последовательно передаваемого слова
0x3FF1	Коэффициент деления тактовых синхроимпульсов SPORT1 (определяет частоту)
0x3FF0	Коэффициент деления кадровых синхроимпульсов приема SPORT1 (определяет частоту)
0x3FEF	Регистр управления автобуферизацией SPORT1 (отсутствует в процессоре ADSP-21msp58/59)

Имеется два способа инициализации или изменения значений в регистрах конфигурации последовательного порта: запись содержимого регистра по непосредственному адресу (тип команды 3) или запись непосредственно получаемых данных по косвенному адресу (тип команды 2). При любом из вышеназванных способов важно задать конфигурацию последовательного порта перед его разблокированием.

При первом способе программирования регистров конфигурации не требуется устанавливать регистры генератора адреса данных, зато для выполнения записи требуются две команды. Например:

$AX0 = 0x6B27$  ; {содержимое регистра AX0 записывается}  
 $DM(0x3FF2) = AX0$  ; {по адресу 0x3FF2}

$AX0 = 0$  ; {содержимое регистра AX0 записывается}  
 $DM(0x3FF3) = AX0$  ; {по адресу 0x3FF3}

При втором способе индексные регистры (I) генератора адреса данных должны содержать адреса записываемых регистров конфигурации в карте памяти. Регистр модификации (M), содержимое которого обновляет регистр I после

## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5

записи, также должен содержать требуемое значение. А регистр длины L с тем же номером, что и индексный регистр I, должен содержать значение ноль, чтобы циклические буферы не были активны. Например:

```
I 0 = 0 x 3 F F 2 ;
M 0 = 1 ;
L 0 = 0 ;
D M ( I 0 , M 0 ) = 0 x 6 B 2 7 ; {постоянная 0x6B27 записывается по}
                                   {адресу, указанному в регистре I0; затем}
                                   {указатель модифицируется на величину,}
                                   {ну, содержащуюся в M0}
D M ( I 0 , M 0 ) = 0 ;           {адрес 0x3FF3 устанавливается равным 0}
```

При использовании второго метода для задания конфигурации регистров требуется только один цикл, когда регистры I, M и L инициализированы. Однако при использовании этого метода возможно гораздо большее число ошибок, так как осуществляется косвенная запись всех регистров. Перед записью следует убедиться, что регистр I содержит требуемое значение.

### 5.3.2 Прием и передача данных

В каждом последовательном порте имеются регистры приема и передачи. Эти регистры не отображены в карте памяти, но распознаются мнемоникой ассемблера. Регистры передачи называются TX0 и TX1 в портах SPORT0 и SPORT1 соответственно. Регистры приема соответственно называются RX0 для SPORT0 и RX1 для SPORT1. Доступ к этим регистрам обеспечивается в любой момент во время выполнения программы используя доступ к памяти данных с непосредственным адресом, загрузку регистров непосредственно получаемыми данными или передачу между регистрами (команды типа 3, 7 и 17). Например, следующая команда подготовит SPORT1 для последовательной передачи данных, считая, что SPORT1 уже сконфигурирован и разблокирован:

```
T X 1 = A X 0 ;           {содержимое регистра AX0 передается в порт }
                           {SPORT1}
```

Следующая команда обеспечит доступ к последовательно переданному значению, принятому через SPORT0:

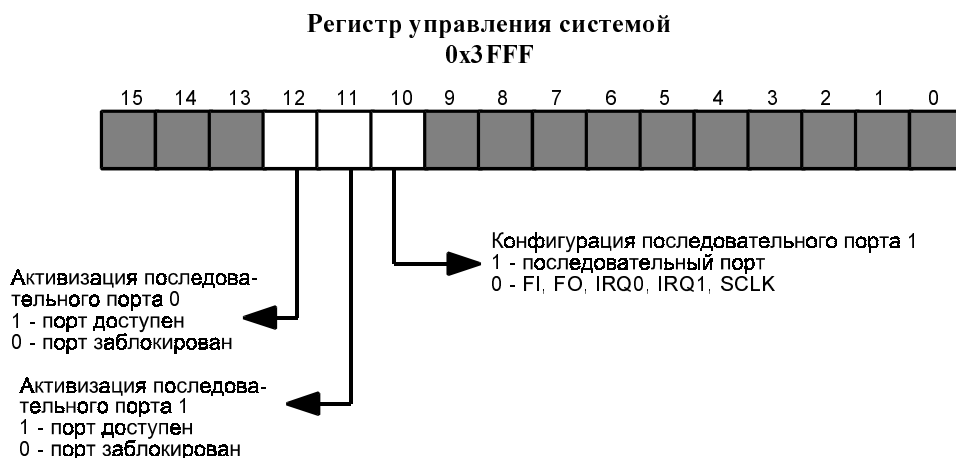
```
A Y 0 = R X 0 ;           {содержимое регистра приема SPORT0 передает-}
                           {ся в регистр AY0}
```

## 5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ

Так как последовательные порты управляются прерываниями, то эти команды, как правило, будут выполняться в ответ на прерывание последовательного порта.

### 5.4 АКТИВИЗАЦИЯ ПОСЛЕДОВАТЕЛЬНОГО ПОРТА

Последовательные порты активизируются битами в регистре управления системой. Этот регистр отображен в карте памяти данных по адресу 0x3FFF. Когда бит 12 равен 1, работа SPORT0 разрешена, а когда бит 11 равен 1, разрешена работа SPORT1. Оба этих бита сбрасываются при перезапуске системы, блокируя при этом оба последовательных порта.



**Рис. 5.2** Активизация последовательного порта установкой бит в регистре управления системой

Бит 10 регистра управления системой определяет конфигурацию SPORT1, задавая его либо как последовательный порт, либо как прерывания и флаги, согласно табл. 5.4. Если бит 10 установлен равным 1, SPORT1 функционирует как последовательный порт; если он установлен равным 0, рабочими становятся альтернативные функции (а бит 11 игнорируется). При перезапуске, бит 10 равен 1, таким образом SPORT1 функционирует как последовательный порт.

## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5

Таблица 5.4

Альтернативная конфигурация SPORT1		
Название вывода	Альтернативное название	Альтернативная функция
RFS1	$\overline{RQ0}$	Внешнее прерывание 0
TFS1	$\overline{RQ1}$	Внешнее прерывание 1
DR1	FI	Вход флага
DT1	FO	Выход флага
SCLK1	То же самое	Та же самая

### 5.5 ТАКТОВЫЕ СИНХРОИМПУЛЬСЫ ПОСЛЕДОВАТЕЛЬНЫХ ПОРТОВ

Каждый последовательный порт работает со своим собственным последовательным тактовым сигналом. Тактовые синхроимпульсы могут быть внутренними или приниматься от внешнего источника.

Бит ISCLK - бит 14 регистра управления SPORT0 или SPORT1 определяет источник тактовых синхроимпульсов для данного последовательного порта. Когда этот бит равен 1, процессор сам генерирует сигнал тактовой синхронизации; когда этот бит равен 0, процессор ожидает приема синхронизирующего сигнала от внешнего генератора тактовых синхроимпульсов. При перезапуске бит ISCLK сбрасывается, таким образом, для обоих портов устанавливается режим работы с внешними тактовыми синхроимпульсами. Со следующего цикла после установки бита ISCLK начинается генерация тактового сигнала, независимо от того разблокирован ли соответствующий последовательный порт.



**Рис. 5.3 Бит ISCLK регистра управления последовательным портом**

Внешние тактовые синхроимпульсы могут иметь частоты равные частоте циклов процессора, т.е. до 13,824 МГц; внутренние тактовые синхроимпульсы могут иметь частоту равную половине тактовой частоты процессора. Частота генерируемых внутренних тактовых синхроимпульсов функционально связана с тактовой частотой процессора (как показано на выводе CLKOUT) и определяется

## 5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ

содержимым 16-битового регистра SCLKDIV тактового генератора (0x3FF5 для SPORT0 и 0x3FF1 для SPORT1).

$$\text{Частота SCLK} = \frac{\text{Частота SCLKOUT}}{2 \times (\text{SCLKDIV} + 1)}$$

В табл. 5.5 показаны некоторые наиболее распространенные значения частоты тактовых синхроимпульсов (SCLK) и соответствующие им значения регистра SCLKDIV.

Таблица 5.5

### Наиболее распространенные частоты внутренне генерируемых тактовых синхроимпульсов

<i>SCLKDIV</i>	<i>Частота тактовых синхроимпульсов (SCLK)</i>
20479	300 Гц
5119	1200 Гц
639	9600 Гц
95	64 кГц
3	1,536 МГц
2	2,048 МГц
0	6,144 МГц

(Тактовая частота процессора (SCLKOUT) принимается равной 12,288 МГц)

Если значение в регистре SCLKDIV меняется, когда внутренний тактовый генератор разблокирован, то частота тактовых синхроимпульсов изменяется по переднему фронту следующего синхроимпульса.

Обратите внимание, что, когда SPORT1 используется в его альтернативной конфигурации (как FO, FI и два прерывания), тактовый генератор этого порта (вывод SCLK), тем не менее, функционирует. В этом случае тактовый генератор не восприимчив к внешним тактовым синхроимпульсам, но может генерировать внутренний синхронизирующий сигнал, как было описано выше.

### 5.6 ДЛИНА СЛОВА

Каждый последовательный порт независимо обрабатывает слова длиной от 3 до 16 бит. Слова данных, имеющие длину менее 16 бит, выравниваются по правому разряду в регистрах данных последовательного порта. В регистре управления каждого последовательного порта имеется группа разрядов (SLEN), которые определяют длину последовательно переданного слова по формуле:

$$\text{Длина последовательно переданного слова} = \text{SLEN} + 1$$

Например, если вы используете последовательно переданное слово длиной 8 бит, установите SLEN равным 7 (0111 в двоичной системе счисления). Группа

122

## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5

разрядов SLEN образуется битами 3-0 в регистре управления последовательным портом (0x3FF6 для SPORT0 и 0x3FF2 для SPORT1). См. рис. 5.4.

Не устанавливайте значение SLEN равным 0 или 1; эти значения не допустимы для этой группы разрядов.



**Рис. 5.4** Группа разрядов SLEN в регистре управления последовательным портом

### 5.7 ОПЦИИ ПОКАДРОВОЙ ПЕРЕДАЧИ СЛОВ

Сигналы кадровой синхронизации указывают на начало побитовой передачи каждого слова данных. В последовательных портах предусмотрено множество способов кадровой синхронизации. Кадровые синхронизации приема и передачи осуществляются независимо друг от друга. Дискретизация всех сигналов кадровой синхронизации осуществляется по заднему фронту сигнала тактовой синхронизации (SCLK).

#### 5.7.1 Кадровая синхронизация

Сигналы кадровой синхронизации слов данных не являются обязательными. При возникновении потребности в кадровой синхронизации приема или передачи бит (RFSR или TFSR соответственно) регистра управления последовательным портом устанавливается равным 0; сигнал кадровой синхронизации необходим только для начальной установки связи, сразу после передачи первого бита сигнал кадровой синхронизации игнорируется. Затем слова передаются постоянно, без кадровой синхронизации. Когда биты RFSR или TFSR равны

## 5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ

сигнал кадровой синхронизации требуется в начале передачи или приема каждого слова данных.

Битом RFSR является бит 13 в регистре управления последовательным портом (0x3FF6 для SPORT0 и 0x3FF2 для SPORT1), а битом TFSR - бит 11 того же регистра. Оба этих бита сбрасываются при перезапуске, и связь в обоих направлениях на обоих последовательных портах осуществляется без кадровой синхронизации.

Примеры согласования кадровой синхронизации во времени можно найти в разделе "Примеры конфигураций" данной главы.



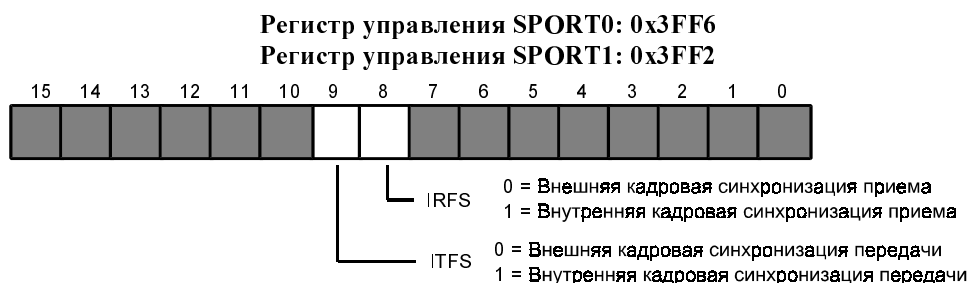
Рис. 5.5 Биты TFSR и RFSR в регистре управления последовательным портом

### 5.7.2 Источник сигнала кадровой синхронизации

Процессор может генерировать внутренние сигналы кадровой синхронизации или принимать их от внешнего источника. Источники сигналов кадровой синхронизации приема и передачи могут устанавливаться независимо друг от друга. Когда бит внутренней кадровой синхронизации приема (IRFS) или передачи (ITFS) в регистре управления последовательным портом равен 0, процессор ожидает приема соответствующего сигнала на внешних выводах кадровой синхронизации (RFS или TFS). Если бит IRFS или ITFS равен 1, процессор генерирует собственный сигнал кадровой синхронизации и подает на выводы RFS или TFS соответствующие выходные сигналы.

Битом IRFS является бит 8 в регистре управления последовательным портом (0x3FF6 для SPORT0 и 0x3FF2 для SPORT1), а битом ITFS - бит 9. Оба этих бита сбрасываются при перезапуске, и для обоих последовательных портов требуются внутренние сигналы кадровой синхронизации как для приема, так и для передачи данных.

## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5



**Рис. 5.6 Биты IRFS и ITFS в регистре управления последовательным портом**

Если сигналы кадровой синхронизации генерируются внешне, то на выходы RFS и TFS поступают сигналы из вне, а передачей и приемом данных управляет внешний источник. Последовательный порт будет ожидать сигнала кадровой синхронизации передачи перед передачей данных и сигнала кадровой синхронизации приема перед началом приема данных. Однако, если сигналы кадровой синхронизации генерируются процессором, то на выходы RFS и TFS подаются эти сгенерированные сигналы, и процессор сам управляет синхронизацией операций с данными.

Последовательный порт передает внутренне генерируемый сигнал кадровой синхронизации после загрузки данных в регистр передачи (TX0 или TX1) для обеспечения непрерывной передачи данных после того, как был передан последний бит передаваемого слова (точное время зависит от используемого режима кадровой синхронизации; см. следующий раздел "Нормальный и альтернативный режим кадровой синхронизации"). Наличие сигнала кадровой синхронизации передачи определяется доступностью данных в регистре передачи.

При внутренней генерации сигналов кадровой синхронизации приема процессор сам управляет согласованием принимаемых данных во времени. Внешний источник данных может подавать на последовательный порт процессора данные, синхронизированные сигналом кадровой синхронизации приема (согласование во времени зависит от используемого режима кадровой синхронизации; см. следующий раздел "Нормальный и альтернативный режим кадровой синхронизации"). Процессор генерирует сигнал кадровой синхронизации приема (RFS) с периодичностью, кратной циклам тактового генератора, на основе значения, содержащегося в 16-разрядном регистре делителя кадровой синхронизации приема RFSDIV (0x3FF4 для SPORT0 и 0x3FF0 для SPORT1).

Число циклов тактового генератора между подтверждениями RFS = RFSDIV + 1

## 5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ

Например, чтобы между подтверждениями сигнала кадровой синхронизации приема (RFS) проходило 256 циклов тактового генератора (SCLK), регистр RFSDIV должен содержать значение 255 (0xFF).

Не рекомендуется использовать значения RFSDIV+1 меньше, чем длина слова.

Заметим, что даже при внешних тактовых синхроимпульсах сигналы кадровой синхронизации могут генерироваться внутренне. Это дает возможность делить внешние тактовые синхроимпульсы в любых целях.

Можно также использовать один и тот же сигнал кадровой синхронизации как для приема, так и для передачи данных. Например, вывод RFS, на который выводится внутренне генерируемый сигнал синхронизации приема, может быть подсоединен к выводу TFS, предназначенному для ввода внешнего сигнала кадровой синхронизации передачи того же последовательного порта, в результате чего обеспечивается одновременное выполнение операций приема и передачи. Это внутреннее соединение особенно полезно в интерфейсах кодеков.

### 5.7.3 Нормальный и альтернативный режим кадровой синхронизации

При нормальном режиме кадровой синхронизации сигнал кадровой синхронизации проверяется по заднему фронту тактового синхроимпульса. При установке сигнала кадровой синхронизации, принимаемые данные фиксируются по *заднему фронту следующего* тактового синхроимпульса, а передаваемые данные выводятся по *переднему фронту* тактового синхроимпульса. Сигнал кадровой синхронизации не проверяется далее до окончания передачи или приема целого слова. Если прием или передача слова осуществляется непрерывно, т.е. за последним битом одного слова без перерыва следует первый бит следующего слова, то сигнал кадровой синхронизации должен подаваться в том же цикле тактового генератора, что и последний бит каждого слова.

При альтернативном режиме кадровой синхронизации сигнал кадровой синхронизации должен устанавливаться в том же цикле тактового генератора, что и первый бит слова. Биты принимаемых данных фиксируются по заднему фронту тактового синхроимпульса, а передаваемые данные выводятся по переднему фронту тактового синхроимпульса. Однако, сигнал кадровой синхронизации проверяется в данном случае только для первого бита. Внутренне генерируемые сигналы кадровой синхронизации остаются подтвержденными только на время приема или передачи целого слова. Внешне генерируемые сигналы кадровой синхронизации проверяются только во время передачи или приема первого бита.

## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5

Режимы кадровой синхронизации для приема и передачи данных независимы друг от друга. Когда бит, задающий ширину сигнала кадровой синхронизации приема (RFSW) или передачи (TFSW) в регистре управления последовательным портом, равен 0, активизирован нормальный режим синхронизации. Если бит RFSW или TFSW равен 1, используется альтернативный режим синхронизации. Битом RFSW является бит 12 в регистре управления последовательным портом (0x3FF6 для SPORT0 и 0x3FF2 для SPORT1), а битом TFSW бит 10 того же регистра. Оба этих бита сбрасываются при перезапуске, что разрешает нормальный режим кадровой синхронизации в обоих направлениях.



**Рис. 5.7 Биты TFSW и RFSW регистра управления последовательным портом**

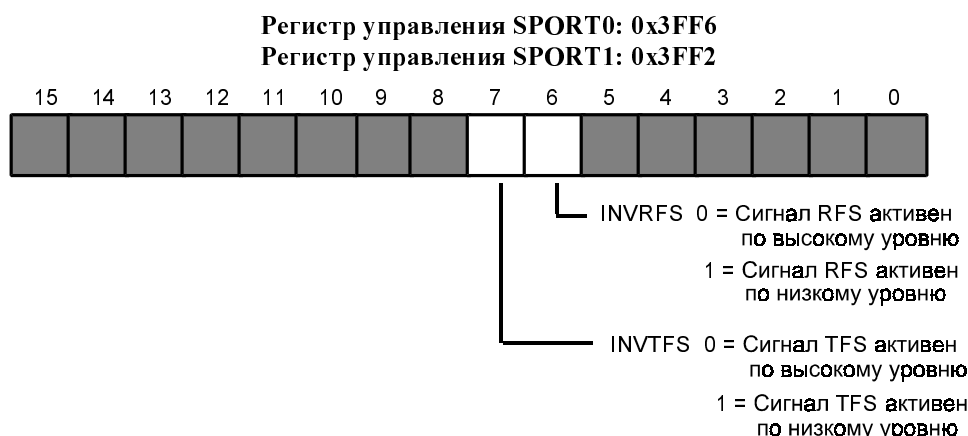
См. примеры нормальной и альтернативной кадровой синхронизации в разделе "Примеры конфигураций" в данной главе.

### 5.7.4 Активный высокий или активный низкий уровень

Сигналы кадровой синхронизации для приема или передачи данных могут быть активными либо по высокому либо по низкому уровню, причем конфигурация этих сигналов задается независимо. Если бит инвертирования RFS (INVRFS) или бит инвертирования TFS (INVTFS) регистра управления последовательным портом равен 0, то соответствующий сигнал кадровой синхронизации активен по высокому уровню. Если же бит INVRFS или INVTFS равен 1 - сигнал кадровой синхронизации активен по низкому уровню. Эти средства управления применимы ко всем сигналам кадровой синхронизации независимо от их источника; они либо управляют полярностью внутренних сигналов, либо определяют способ интерпретации внешних сигналов кадровой синхронизации.

## 5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ

Битом INVRFS является 6-й бит регистра управления последовательным портом (0x3FF6 для SPORT0 и 0x3FF2 для SPORT1), а битом INVTFS - 7-й бит этого регистра. Оба этих бита сбрасываются при перезапуске, и сигналы кадровой синхронизации являются активными по высокому уровню.



**Рис. 5.8 Биты INVRFS и INVTFS регистра управления последовательным портом**

### 5.8 ПРИМЕРЫ КОНФИГУРАЦИЙ

Приводимый ниже фрагмент программы иллюстрирует способ задания конфигурации последовательных портов. В данном примере устанавливается конфигурация обоих последовательных портов SPORT0 и SPORT1. Конфигурация SPORT0 задана для работы с внутренними тактовыми синхроимпульсами, внутренними сигналами кадровой синхронизации и словами данных длиной в 8 бит, компандируемыми по закону с  $\mu$ -характеристикой. Такая установка является типичной для связи с кодеком. Конфигурация SPORT1 задана для работы с внешними тактовыми синхроимпульсами, внешними сигналами кадровой синхронизации, некомпандируемыми словами данных длиной 16 бит и с использованием автобуферизации. Такая установка может использоваться для передачи данных между процессорами в системе из нескольких процессоров.

Инициализируются только необходимые регистры, отображенные в карте памяти. Обратите внимание, что конфигурация последовательных портов задается до их активизации, а любые посторонние фиксированные прерывания сбрасываются перед активизацией прерываний.

## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5

```

{ - - - - П Р О Г Р А М М А   И Н И Ц И А Л И З А Ц И И
  П О С Л Е Д О В А Т Е Л Ь Н Ы Х   П О Р Т О В   - - - }
{Устройства SPORT1}
A X 0 = 0 x 0 0 1 7 ;
D M ( 0 x 3 F E F ) = A X 0 ;      {активизация автобуферизации SPORT1}
                                   {буфер TX использует I0 и M0}    {буфер
                                   RX использует I1 и M1}

A X 0 = 0 x 2 8 0 F ;
D M ( 0 x 3 F F 2 ) = A X 0 ;      {внешние тактовые синхроимпульсы }
                                   {RFS, TFS; требуются RFS и TFS}
                                   {нормальная кадровая синхронизация,}
                                   {нет компандирования; слова по 16 бит}

{Устройства SPORT0}
{Предполагаем SCLKIN равной 12,288 МГц. Частота внутренних тактовых}
{синхроимпульсов (SCLK) равна 2,048 МГц, а кадровая частота - 8 кГц}
A X 0 = 2 5 5 ;
D M ( 0 x 3 F F 4 ) = A X 0 ;      {RFSDIV=256, между сигналами кадровой}
                                   {синхронизации 256 циклов тактового генератора;}
                                   {кадровая синхронизация частотой 8 кГц}

A X 0 = 2 ;
D M ( 0 x 3 F F 5 ) = A X 0 ;      {SCLK=2.048 МГц}
A X 0 = 0 x 6 B 2 7 ;
D M ( 0 x 3 F F 6 ) = A X 0 ;      {внутренние SCLK, RFS, TFS, нормальная }
                                   {кадровая синхронизация, компандирование с }
                                   {μ-характеристикой, 8-битовые слова}

{А К Т И В И З А Ц И Я   П О С Л Е Д О В А Т Е Л Ь Н О Г О   П О Р Т А }
I F C = 0 x 1 E ;                  {сброс любых посторонних прерываний}
I C N T L = 0 ;                    {вложенность прерываний не разрешена}

A X 0 = 0 x 1 C 1 F ;              {активизированы оба последовательных порта, }
D M ( 0 x 3 F F F ) = A X 0 ;      {биты BWAIT и PWAIT устанавливаются по}
                                   {умолчанию}

I M A S K = 0 x 1 E ;              {прерывания последовательного порта}
                                   {разрешены}
{ - - - - О К О Н Ч А Н И Е   И Н И Ц И А Л И З А Ц И И
  П О С Л Е Д О В А Т Е Л Ь Н Ы Х   П О Р Т О В   - - - }

```

## 5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ

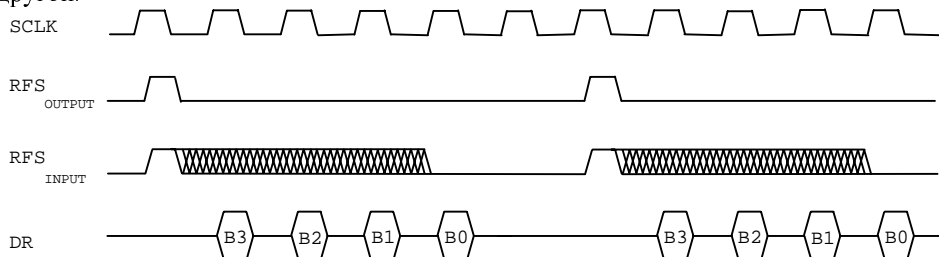
### 5.9 ПРИМЕРЫ ВРЕМЕННЫХ ХАРАКТЕРИСТИК

В данном разделе содержатся примеры некоторых комбинаций различных методов кадровой синхронизации. Приводимые ниже временные диаграммы показывают отношения между сигналами, но они не промасштабированы для отражения реальных временных характеристик процессора. Реальные временные характеристики и значения можно найти в технических характеристиках процессоров.

В приводимых примерах длина слова данных предполагается равной четырем битам, т.е.  $SLEN = 3$ . Сигналы кадровой синхронизации активны по высокому уровню, т.е.  $INVRFS = 0$  и  $INVTFS = 0$ .

Для каждого примера показано содержимое регистра управления последовательным портом (0x3FF6 для SPORT0 и 0x3FF2 для SPORT1). В этих двоичных величинах 1 означает высокий уровень, 0 - низкий уровень, а X может означать любое из двух. Биты, которые задают иллюстрируемые режимы, подчеркнуты.

На рис. 5.9 - 5.14 показаны примеры кадровой синхронизации для приема данных. На рис. 5.9 и 5.10 показан нормальный режим кадровой синхронизации для приема данных с перерывом (между словами задается любое число циклов тактового генератора) и непрерывного приема данных (между словами нет циклов тактового генератора). На рис. 5.11 и 5.12 показан альтернативный режим кадровой синхронизации прерываемого и непрерывного приема данных. На этих четырех рисунках показаны временные требования к внешнему сигналу кадровой синхронизации и временные характеристики внутреннего сигнала кадровой синхронизации. Обратите внимание, что выходные временные характеристики согласуются с входными; таким образом, в процессорах с двумя последовательными портами один последовательный порт может подавать сигнал RFS на другой.



Состояния регистра управления последовательным портом:

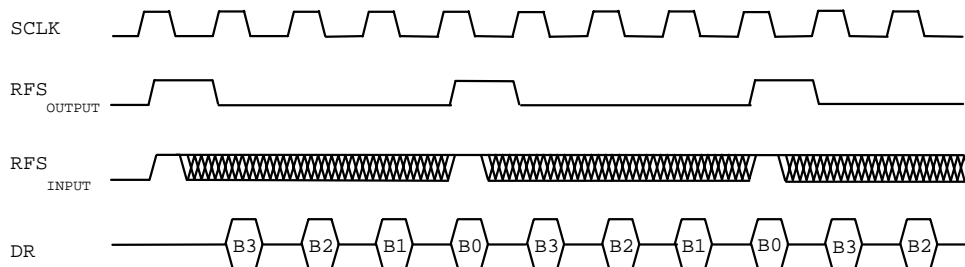
при внутреннем сигнале кадровой синхронизации 0X10XXX1X0XX0011;

при внешнем сигнале кадровой синхронизации 0X10XXX0X0XX0011.

Показана возможность как внутренней, так и внешней кадровой синхронизации

**Рис. 5.9 Прием данных через последовательный порт, нормальный режим кадровой синхронизации**

## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5



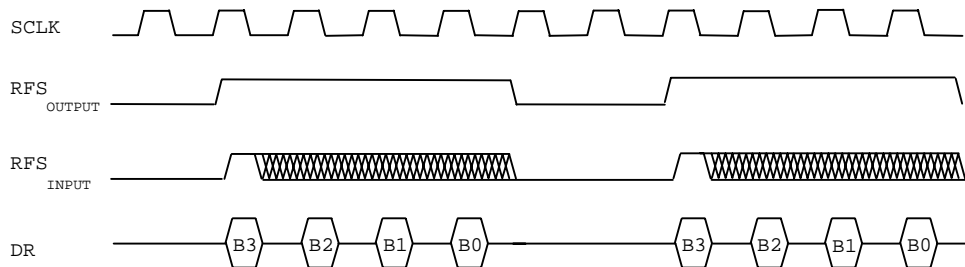
Состояния регистра управления последовательным портом:

при внутреннем сигнале кадровой синхронизации  $0X10XXX1X0XX0011$ ;

при внешнем сигнале кадровой синхронизации  $0X10XXX0X0XX0011$ .

Показана возможность как внутренней, так и внешней кадровой синхронизации

**Рис. 5.10 Непрерывный прием данных через последовательный порт, нормальный режим кадровой синхронизации**



Состояния регистра управления последовательным портом:

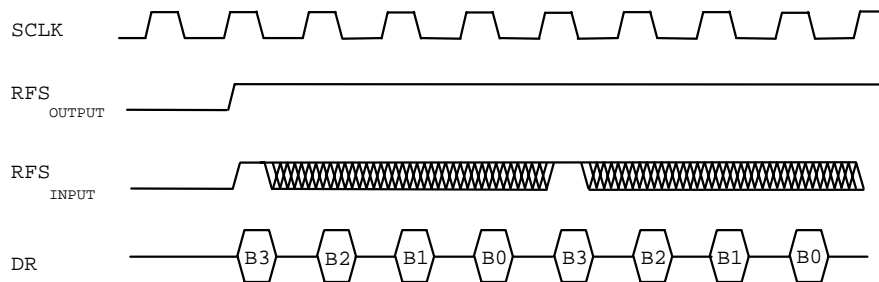
при внутреннем сигнале кадровой синхронизации  $0X11XXX1X0XX0011$ ;

при внешнем сигнале кадровой синхронизации  $0X11XXX0X0XX0011$ .

Показана возможность как внутренней, так и внешней кадровой синхронизации

**Рис. 5.11 Прием данных через последовательный порт, альтернативный режим кадровой синхронизации**

## 5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ



Состояния регистра управления последовательным портом:

при внутреннем сигнале кадровой синхронизации

0X11XXX1X0XX0011;

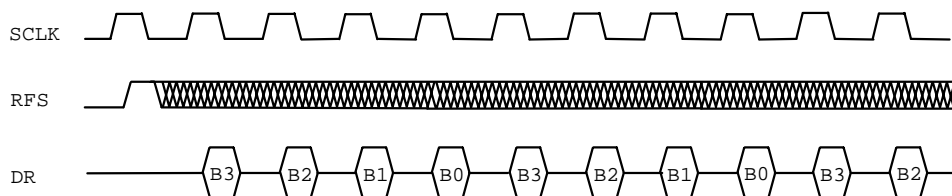
при внешнем сигнале кадровой синхронизации

0X11XXX0X0XX0011.

**Показана возможность как внутренней, так и внешней кадровой синхронизации**

**Рис. 5.12 Непрерывный прием данных через последовательный порт, альтернативный режим кадровой синхронизации**

На рис. 5.13 и 5.14 показана операция приема данных при нормальном и альтернативном режиме, соответственно, в режиме без кадровой синхронизации. Единичный сигнал кадровой синхронизации подается только в начале приема первого слова, либо на один тактовый синхроимпульс раньше приема первого бита (нормальный режим), либо одновременно с приемом первого бита (альтернативный режим). Этот режим подходит для приема пакетов, состоящих из множества слов (непрерывный прием).



Состояния регистра управления последовательным портом:

при внутреннем сигнале кадровой синхронизации

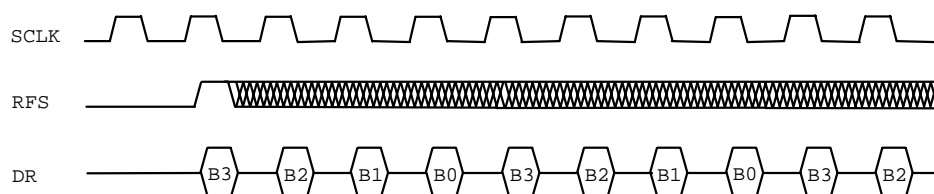
0X00XXX1X0XX0011;

при внешнем сигнале кадровой синхронизации

0X11XXX0X0XX0011.

**Рис. 5.13 Прием данных через последовательный порт без кадровой синхронизации, нормальный режим.**

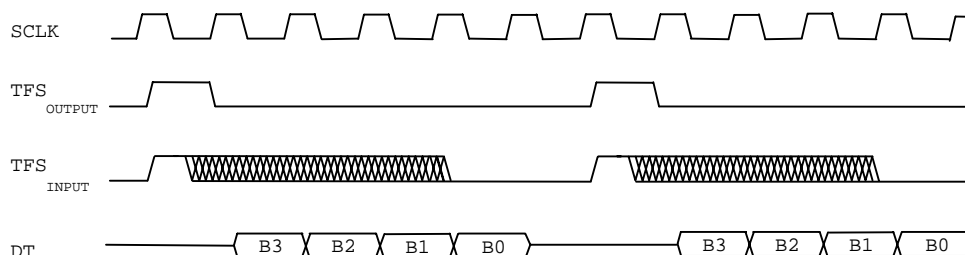
## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5



Состояния регистра управления последовательным портом:  
 при внутреннем сигнале кадровой синхронизации 0X**01**XXX**1**X0XX0011;  
 при внешнем сигнале кадровой синхронизации 0X**01**XXX**0**X0XX0011.

**Рис. 5.14 Прием данных через последовательный порт без кадровой синхронизации, альтернативный режим**

На рис. 5.15 - 5.20 показана кадровая синхронизация для передачи данных, рисунки очень похожи на рис. 5.9 - 5.14. На рис. 5.15 и 5.16 показан нормальный режим кадровой синхронизации для прерываемой и непрерывной передачи данных. На рис. 5.17 и 5.18 показана прерываемая и непрерывная передача данных в альтернативном режиме. Аналогично временным характеристикам приема, выходной сигнал TFS также согласуется с временными требованиями ко входному сигналу TFS.

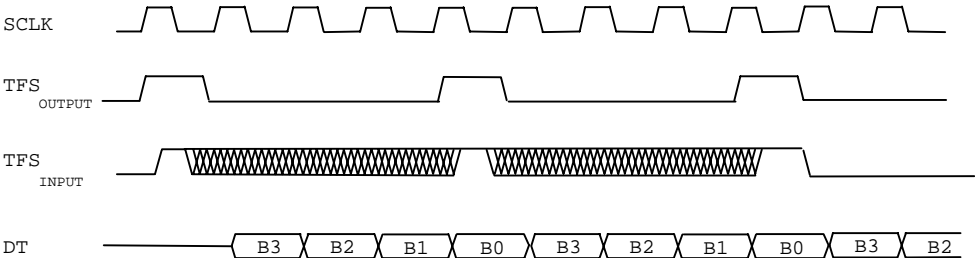


Состояния регистра управления последовательным портом:  
 при внутреннем сигнале кадровой синхронизации 0XXX**101**X0XX0011;  
 при внешнем сигнале кадровой синхронизации 0XXX**100**X0XX0011.

**Показана возможность как внутренней, так и внешней кадровой синхронизации**

**Рис. 5.15 Передача данных через последовательный порт, нормальный режим кадровой синхронизации**

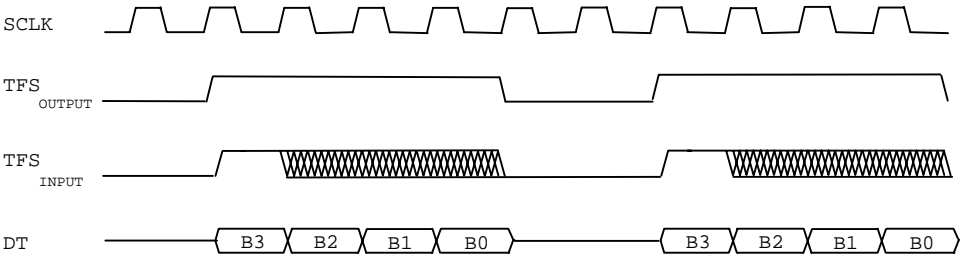
# 5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ



Состояния регистра управления последовательным портом:  
при внутреннем сигнале кадровой синхронизации 0XXX**101**X0XXX0011;  
при внешнем сигнале кадровой синхронизации 0XXX**100**X0XXX0011.

Показана возможность как внутренней, так и внешней кадровой синхронизации

**Рис. 5.16 Непрерывная передача данных через последовательный порт, нормальный режим кадровой синхронизации**



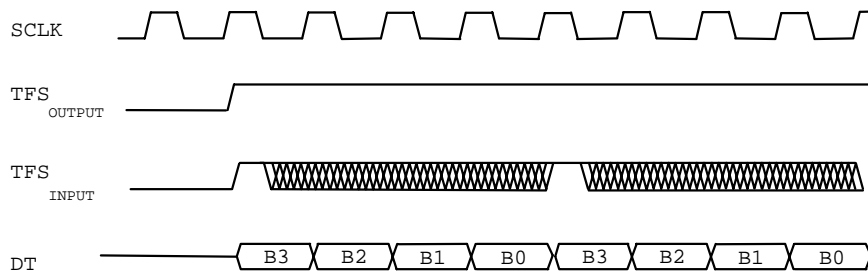
Состояния регистра управления последовательным портом:  
при внутреннем сигнале кадровой синхронизации 0XXX**111**X0XXX0011;  
при внешнем сигнале кадровой синхронизации 0XXX**110**X0XXX0011.

Показана возможность как внутренней, так и внешней кадровой синхронизации

**Примечание:** Между входным сигналом TFS и DT происходит асинхронная задержка. См. описание в соответствующей карте данных.

**Рис. 5.17 Передача данных через последовательный порт, альтернативный режим кадровой синхронизации**

## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5



Состояния регистра управления последовательным портом:

при внутреннем сигнале кадровой синхронизации 0XXX**111**X0XXX0011;

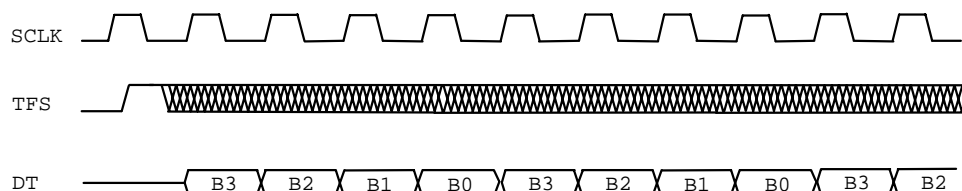
при внешнем сигнале кадровой синхронизации 0XXX**110**X0XXX0011.

**Показана возможность как внутренней, так и внешней кадровой синхронизации**

**Примечание:** Между входным сигналом TFS и DT происходит асинхронная задержка. См. описание в соответствующей карте данных.

**Рис. 5.18 Непрерывная передача данных через последовательный порт, альтернативный режим кадровой синхронизации**

На рис. 5.19 и 5.20 показана операция передачи данных при нормальном и альтернативном режиме соответственно, в режиме без кадровой синхронизации. Единственный сигнал кадровой синхронизации подается в начале первого слова, либо на один тактовый синхроимпульс раньше первого бита (нормальный режим), либо одновременно с передачей первого бита (альтернативный режим).



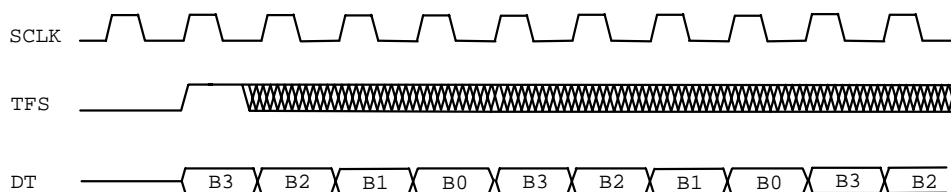
Состояния регистра управления последовательным портом:

при внутреннем сигнале кадровой синхронизации 0XXX**001**X0XXX0011;

при внешнем сигнале кадровой синхронизации 0XXX**000**X0XXX0011.

**Рис. 5.19 Передача данных через последовательный порт без кадровой синхронизации, нормальный режим**

## 5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ



Состояния регистра управления последовательным портом:

при внутреннем сигнале кадровой синхронизации 0XXX**011**X0XXX0011;

при внешнем сигнале кадровой синхронизации 0XXX**010**X0XXX0011.

**Рис. 5.20** Передача данных через последовательный порт без кадровой синхронизации, альтернативный режим

### 5.10 КОМПАНДИРОВАНИЕ И ФОРМАТ ДАННЫХ

Компандирование (от английского "companding" - COMpressing (сжатие) + exPANDing (расширение)) - это процесс логарифмического кодирования и декодирования данных с целью уменьшения числа бит передаваемых данных. Оба последовательных порта имеют аппаратное обеспечение компандирования; в каждом цикле процессора может производиться по одной операции расширения и сжатия. В случае соперничества приоритетом обладает SPORT0.

Процессоры семейства ADSP-2100 поддерживают оба широко используемых алгоритма компандирования: с  $\mu$ - и с А-характеристикой. Процессор компандирует данные согласно рекомендации G.711 МККТТ. Тип компандирования может выбираться независимо для каждого последовательного порта.

Если компандирование не разрешено, то для принимаемых слов данных длиной менее 16 бит доступными являются два формата: в одном неиспользуемые старшие биты заполняются нулями, в другом старшие биты дополняются знаковыми битами.

Тип компандирования, так же как и формат данных при неразрешенном компандировании, управляется группой разрядов DTYPE (биты 5-4) в регистре управления последовательным портом (0x3FF6 для SPORT0 и 0x3FF2 для SPORT1), как показано на рис. 5.21.

При разрешенном компандировании, действительным значением регистра RX0 или RX1 является выровненное по правому разряду, дополненное по знаку, расширенное значение восьми принятых самых младших бит. Подобным образом, запись в регистры TX0 или TX1 вызывает сжатие 16-разрядного значения до восьми его самых младших бит (дополненных по знаку до ширины передаваемого слова) перед записью этого значения во внутренний регистр передачи. Если

## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5

модуль этого 16-разрядного значения превышает 13-разрядный (для компандирования по закону с А-характеристикой) или 14-разрядный (для компандирования по закону с  $\mu$ -характеристикой) максимум, то это значение автоматически сжимается до максимально возможного положительного или отрицательного значения.



**Рис. 5.21** Группа разрядов DTYPE в регистре управления последовательным портом

### 5.10.1 Примеры операции компандирования

При использовании схемы компандирования, сопряжение с кодеком не требует больших усилий с точки зрения его программирования. Пример сопряжения кодека можно найти в последнем разделе данной главы. Ниже приводится типичная последовательность операций при передаче компандированных данных:

- Данные записываются в регистр TXn
- Значение в регистре TXn сжимается
- Сжатое значение снова записывается в регистр TXn
- После прихода сигнала кадровой синхронизации (при его необходимости) значение, содержащееся в регистре TXn, записывается во внутренний регистр передачи и пересылается по битам, начиная с самого старшего бита.

Как только последовательный порт начинает передачу второго бита текущего слова, в регистр TXn может быть записано новое слово данных, несмотря на то, что передача первого слова еще не завершена. После окончания передачи самого старшего бита последовательный порт генерирует прерывание передачи для указания на то, что регистр TXn готов для принятия следующего слова данных. При внешнем сигнале кадровой синхронизации следующее слово должно записываться в регистр TXn таким образом, чтобы до прихода следующего сигнала

## 5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ

кадровой синхронизации оставалось достаточно времени для сжатия этого слова.

Ниже приводится типичная последовательность операций при приеме компандированных данных.

- Биты накапливаются по мере их получения во внутреннем регистре приема
- После приема целого слова данных это слово записывается в регистр RXn
- Значение, содержащееся в регистре RXn расширяется
- Расширенное значение записывается обратно в регистр RXn

Затем генерируется прерывание приема для рассматриваемого порта.

### 5.10.2 Конфликт аппаратных средств компандирования

Так как оба последовательных порта совместно используют одно и то же устройство компандирования, в течение одного машинного цикла может иметь место только одна операция сжатия и расширения. При возникновении конфликта, например, когда в одном и том же цикле требуются два расширения, приоритетом обладает SPORT0, в то время как SPORT1 находится в состоянии принудительного ожидания в течение этого цикла.

Данный конфликт аппаратных средств имеет очень незначительные последствия. Система команд не поддерживает загрузку обоих регистров TX0 и TX1 в течение одного и того же цикла; следовательно, во многих случаях эти операции, естественно, не могут совпасть во времени для возникновения конфликтной ситуации. Лишний непроизводительный цикл предшествует прерыванию приема и не может увеличить время, необходимое для обработки этого прерывания, хотя и увеличивает время ожидания до получения прерывания.

### 5.10.3 Компандирование внутренних данных

Так как содержимое регистров RX и TX компандируется "на месте", имеется возможность внутреннего использования устройства компандирования вообще без приема или передачи данных и активизации последовательного порта. Такая операция может использоваться для отладки системы или преобразования данных и требует всего одного непроизводительного цикла.

Для сжатия данных нужно разрешить операцию компандирования, а затем:

1. Записать данные в регистр TXn (сжатие вычисляется).
2. Выждать один цикл (в регистр TXn записывается сжатое значение).
3. Считать содержимое регистра TXn (в итоге возвращается 8-разрядное сжатое значение).

## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5

Программа (в кодах) может иметь следующий вид:

$T X 0 = A X 0 ;$     {в регистр передачи записываются данные}  
 $N O P ;$                 {любая команда}  
 $A X 1 = T X 0 ;$     {сжатые данные передаются в регистр AX1}

Та же самая процедура используется и для расширения данных, но вместо регистра  $T X n$  используется регистр  $R X n$ .

$R X 0 = A X 0 ;$     {сжатые данные записываются в регистр приема}  
 $N O P ;$                 {любая команда}  
 $A X 1 = R X 0 ;$     {расширенное значение передается в регистр AX1}

### 5.11 АВТОБУФЕРИЗАЦИЯ

В нормальном режиме работы последовательный порт генерирует прерывание сразу же по приему или началу передачи слова данных. Автобуферизация обеспечивает механизм последовательного приема или передачи целого блока данных до генерирования прерывания. Обслуживающая программа может скорее оперировать целым блоком данных, нежели одним единственным словом, что значительно сокращает непроизводительные издержки. Автобуферизация является доступной в обоих последовательных портах, за исключением процессора ADSP-21msp58/59, в котором такая возможность предусмотрена только в SPORT0.

Автобуферизация использует способность генераторов адреса данных осуществлять адресацию циклических буферов. При разрешенной автобуферизации каждое слово данных последовательно передается (или в случае разрешения многоканальной операции, каждое активное слово передается) в или из памяти данных за один непроизводительный цикл. (Автобуферизация данных в память программы не поддерживается). Непроизводительный цикл не зависит от исполняемой в данный момент команды и задерживает ее выполнение на один цикл (или более, если требуются состояния ожидания). Для таких индивидуальных передач слов данных не генерируется никаких прерываний.

Передача данных при автобуферизации не может дублироваться какой-либо командой. Однако, эквивалентной командой на языке ассемблера могла бы быть:

$D M ( I , M ) = R X 0$

**ИЛИ**

$T X 0 = D M ( I , M )$

*Команды только эквивалентны автобуферизации*

## 5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ

Используемые при пересылке данных регистры I и M указываются соответствующими группами разрядов в регистре управления автобуферизацией последовательного порта.

Перед началом непроизводительного цикла процессор ожидает окончания выполнения текущей команды. Когда необходимость в пересылке данных в связи с автобуферизацией возникает во время выполнения команды, требующей нескольких циклов (например, для состояний ожидания), автобуферизация может быть задержана. Если такая пересылка данных требуется, когда процессор находится в состоянии ожидания по команде IDLE, то сначала выполняется пересылка, а затем процессор возвращается в состояние IDLE.

Если пересылка данных приводит к циклическому возврату указателя циклического буфера, генерируется прерывание последовательного порта. Прерывание приема подается после приема целого буфера данных. Прерывание передачи подается после того, как последнее слово было загружено в регистр TXn, но до начала передачи.

Исключая выполнение команды, требующей нескольких циклов, автоматическая пересылка индивидуальных слов данных обладает высшим приоритетом над любыми операциями, кроме перезапуска, включая все прерывания. Таким образом, пересылка данных с использованием автобуферизации может увеличить время ожидания отклика на прерывание, если прерывание случайно совпадает с такой пересылкой. С автобуферизацией может производиться до четырех пересылок данных; в случае, когда две или более из них необходимы в течение одного и того же цикла, они имеют следующие приоритеты, которые совпадают с приоритетами прерываний последовательного порта.

<i>Высший</i>	Передача SPORT0
	Прием SPORT0
	Передача SPORT1
<i>Низший</i>	Прием SPORT1

В худшем случае, когда требуется осуществить все четыре пересылки с использованием автобуферизации примерно в одно и то же время, обработка прерывания задержится на время, которое необходимо для осуществления всех этих пересылок и определяется состояниями ожидания и запросом шины.

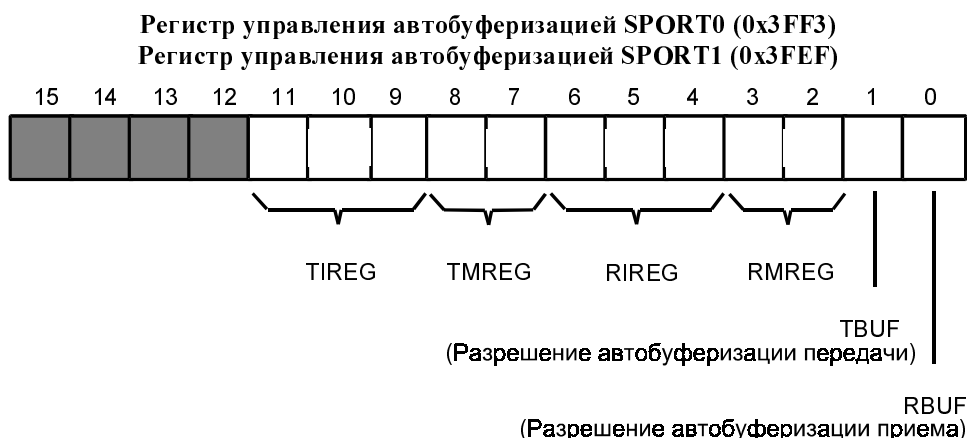
### 5.11.1 Регистр управления автобуферизацией

В режиме автобуферизации прерывание генерируется в случае, когда при модификации заданного регистра I (в генераторе адреса данных) на величину, содержащуюся в заданном регистре M (в генераторе адреса данных), происходит

## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5

переполнение по модулю (циклический возврат указателя). Это означает, что обнаружен конец буфера.

Режим автобуферизации разрешается отдельно для приема и передачи данных с помощью бит в регистре управления буферизацией последовательного порта (0x3FF3 для SPORT0 и 0x3FEF для SPORT1), что показано на рис. 5.22.



**Рис. 5.22 Регистр управления автобуферизацией последовательного порта**

Используемые при автобуферизации регистры I и M указываются соответствующими группами бит в регистре управления автобуферизацией. TIREG и TMREG представляют собой двоичные значения, указывающие, соответственно, на число регистров I и M, связанных с буфером передачи. Правила совместного использования регистров I и M те же, что и для других операций генератора адреса данных: регистры I и M должны принадлежать одному генератору адреса данных, иметь номера 0-3 для DAG1 или 4-7 для DAG2. Следовательно, на регистр I указывают три бита, в то время как только два бита необходимо для указания регистра M, так как третий (самый старший) бит, указывающий номер регистра M, совпадает с соответствующим битом номера регистра I.

Аналогичным образом, группы разрядов RIREG и RMREG содержат номера регистров I и M соответственно, связанных с буфером приема.

Биты TBUF и RBUF разрешают автобуферизацию приема и передачи, соответственно. Эти биты становятся равными нулю при перезапуске и повторной начальной загрузке. Следовательно, текущая автобуферизация не может продолжаться во время операции повторной загрузки, и после повторной начальной загрузки следует заново разрешить автобуферизацию.

## 5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ

### 5.11.2 Пример автобуферизации

Ниже приводится фрагмент программы, который иллюстрирует установку SPORT1 для операции автобуферизации. Предполагается, что процессор работает с тактовой частотой 12,288 МГц. Последовательный порт автоматически передает значения из циклического буфера с именем *tx\_buffer*. Последовательный порт также принимает посланные на него значения и автоматически передает данные в буфер с именем *rx\_buffer*. Прерывание передачи генерируется, как только все значения были переданы из буфера *tx\_buffer* в регистр TX1, но до того, как последнее значение загружается в регистр передачи сдвига. Прерывание приема генерируется по окончании полного заполнения буфера *rx\_buffer*.

```
.MODULE/RAM    code_to_init_AB_SPORT1;
{- программа инициализации автобуферизации -}

.VAR/DM/CIRC    tx_buffer[10];
.VAR/DM/CIRC    rx_buffer[10];
.ENTRY          sport1_inits;
{установка регистров I, M, L}
sport1_inits:   I0 = ^tx_buffer;    {I0 содержит адрес tx_buffer}
                M0 = 1;             {заполняется каждая ячейка}
                L0 = %tx_buffer;    {установка L0 на длину}
                                {буфера tx_buffer}
                I1 = ^rx_buffer;    {I1 указывает на rx_buffer}
                L1 = %rx_buffer;    {установка L1 на длину}
                                {буфера rx_buffer}

{установка SPORT1 для автобуферизации}

                AX0 = 0x0013;        {TX использует I0, M0;}
                                {RX использует I1, M0}
                DM(0x3FEF) = AX0;    {автобуферизация
                                разрешена}

{установка частоты дискретизации SPORT1 8 кГц и частоты тактовых
синхроимпульсов 2,048 МГц}
                AX0 = 255;           {установка RFSDIV на 255}
                DM(0x3FF0) = AX0;    {для 8кГц}
                AX0 = 2;             {установка SCLKDIV на 2}
                DM(0x3FF5) = AX0;    {для частоты тактовых синх-
                                } {роимпульсов 2,048 МГц}
```

## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5

{установка нормального режима кадровой синхронизации SPORT1, внутренне}  
{генерируемые тактовые синхроимпульсы, внутренняя кадровая синхронизация}

$A X 0 = 0 \times 6 B 2 7 ;$	{нормальный режим кад- {ровой синхронизации, дли- {на слова 8 бит, компан- {дирование с
$D M ( 0 \times 3 F F 2 ) = A X 0 ;$	{μ-характерис-} {тикой} {внутренние тактовые син- {хроимпульсы и кадровая } {синхронизация}

{установка прерываний}

IFC = 6 ;	{очистка посторонних пре-}
	{рываний SPORT}
ICNTL = 0 ;	{вложенность прерываний }
	{не разрешена}
IMASK = 6 ;	{разрешение прерываний }
	{SPORT1}

{активизация SPORT1}

$A \times 0 = 0 \times 0 \text{ C } 1 \text{ F};$	{активизация SPORT1}
$D M (0 \times 3 \text{ F F F}) = A \times 0;$	{PWAIT, BWAIT остаются}
	{по умолчанию}

{помещение первого пересылаемого значения в регистр TX1}

```

A X 0 = D M ( I 0 , M 0 ) ;
T X 1 = A X 0 ;
R T S ;

```

```

.ENDMOD;

```

## 5.12 МНОГОКАНАЛЬНЫЕ ОПЕРАЦИИ

SPORT0 поддерживает многоканальные операции. В режиме многоканальных операций последовательно передаваемые данные мультиплексируются с разделением во времени. Каждое последующее слово принадлежит следующему каналу, например, блок данных длиной в 24 слова содержит по одному слову для каждого из 24 каналов. SPORT0 поддерживает 32 или 24 канала и может автоматически выбирать слова для определенных каналов, игнорируя при этом другие.

## 5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ

В одноканальном режиме работы, кадровая синхронизация приема и передачи указывает на начало единичного слова или непрерывного потока слов, причем операции приема и передачи независимы друг от друга. В многоканальном режиме работы сигнал кадровой синхронизации приема (RFS0) указывает на начало 24- или 32-словного блока последовательно передаваемых данных, а операции приема и передачи осуществляются параллельно. TFS0 может иметь альтернативную функцию, о которой будет рассказано ниже. **Примечание:** В процессоре ADSP-2105 имеется только один последовательный порт (SPORT1), этот процессор не поддерживает многоканальные операции.

### 5.12.1 Установка многоканального режима работы

Многоканальные операции разрешаются установкой бита 15 в регистре управления SPORT0 (0x3FF6). Когда этот бит равен 1, многоканальный режим работы разрешен, а некоторые управляющие биты регистра управления SPORT0 должны быть заданы заново. Биты, зависящие от разрешения многоканального режима, показаны на рис. 5.23. При перезапуске системы, бит 15 сбрасывается, блокируя многоканальные операции и разрешая нормальный режим работы.

**Регистр управления SPORT0 (0x3FF6) при многоканальном режиме работы**



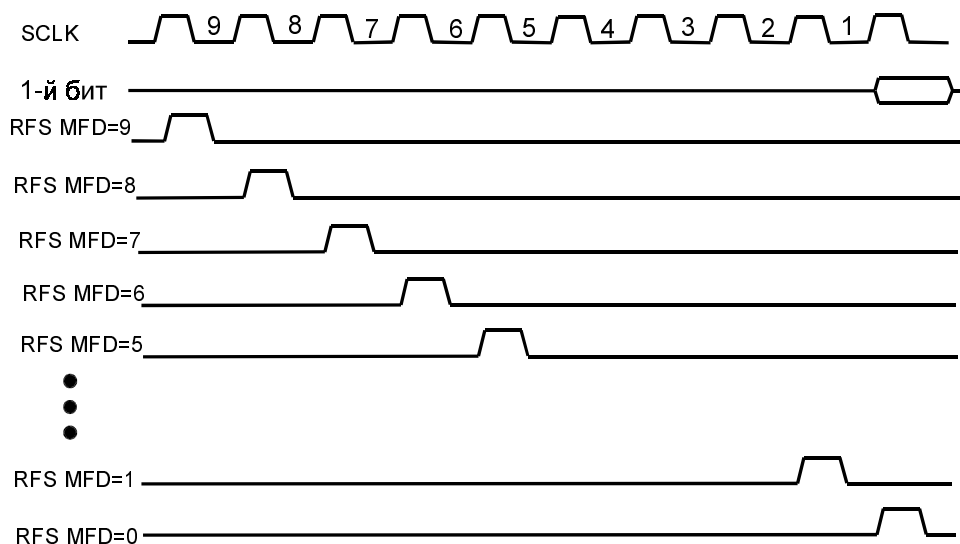
**Рис. 5.23 Регистр управления SPORT0 при разрешенном многоканальном режиме работы**

Состояние бита длины блока данных (MCL) (бит 9 регистра управления SPORT0) определяет, используется ли 24 или 32 канала для пересылки блока данных длиной в 24 или 32 слова соответственно. Когда этот бит равен 0, выбирается 24-словный блок, когда он равен 1, 32-словный. При многоканальном режиме работы длина слова по-прежнему устанавливается группой разрядов SLEN в регистре управления последовательного порта и может принимать значения от 3 до 16 бит.

Задержка кадровой синхронизации при многоканальном режиме работы устанавливается группой из 4 разрядов (MFD), которая определяет (в двоичном

## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5

коде) число циклов тактового генератора между сигналом кадровой синхронизации и первым битом данных. Это позволяет процессору работать с различными типами сопрягаемых устройств T1. Различные виды задержек приведены на рис. 5.24.



**Рис. 5.24** Примеры задержки кадровой синхронизации при многоканальном режиме работы последовательного порта

Отображенные в карте памяти регистр разрешения приема и регистр разрешения передачи имеют разрядность 32 бита и состоят из двух 16-битовых регистров каждый, как показано на рис. 5.25. Каждый бит соответствует каналу; установка бита разрешает использование этого канала, таким образом, процессор сможет выбрать слово из 24- или 32-словного блока данных. Например, установка 0-го бита выбирает слово 0, 12-й бит выбирает слово 12, и т.д.

### 5.12.2 Работа в многоканальном режиме

Принимаемые слова для недоступных каналов игнорируются, т.е. для таких слов не генерируются прерывания, не используется автобуферизация, и данные не записываются в регистр RX0. Подобным образом, для неразрешенных передаваемых слов не генерируются прерывания, и нет автобуферизации. В течении всего временного интервала, необходимого для передачи слова по

## 5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ

недоступному каналу, вывод DT находится в третьем состоянии.



**Рис. 5.25 Регистры разрешения многоканального приема/передачи слов данных**

### SPORT0.

Большинство рабочих параметров SPORT0 не меняются в многоканальном режиме работы. Это относится, в частности, к установленному значению длины слова (SLEN), внутренней или внешней кадровой синхронизации (IRFS), инвертированию сигнала кадровой синхронизации (INVRFS), компандированию (DTYPE) и автобуферизации. **Примечание:** Важным обстоятельством является то, что при многоканальном режиме работы сигнал кадровой синхронизации приема RFS происходит не более одного раза за кадр.

При многоканальном режиме работы сигнал TFS0, вместо обеспечения кадровой синхронизации, выполняет функцию сигнала, подтверждающего достоверность передаваемых данных (TDV). Сигнал TDV подтверждается, пока передающее устройство находится в активном состоянии. Сигнал TDV может быть активным по высокому или по низкому уровню, его полярность управляется битом INVTFS, переименованным в данном контексте в INVTDV. При INVTDV равно 1, сигнал TDV активен по низкому уровню, в обратном случае, он активен по высокому уровню. При необходимости сигнал TDV может использоваться для активизации дополнительной буферной логики.

На рис. 5.26 показано начало многоканальной передачи данных. Как и в предыдущих примерах, длина слова равна 4 битам (SLEN=3), сигналы кадровой синхронизации активны по высокому уровню. Задержка кадровой синхронизации

## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5

при многоканальном режиме работы составляет один цикл тактового генератора. В иллюстративных целях для приема выбраны слова 0 и 2, а для передачи - слова 1 и 2.

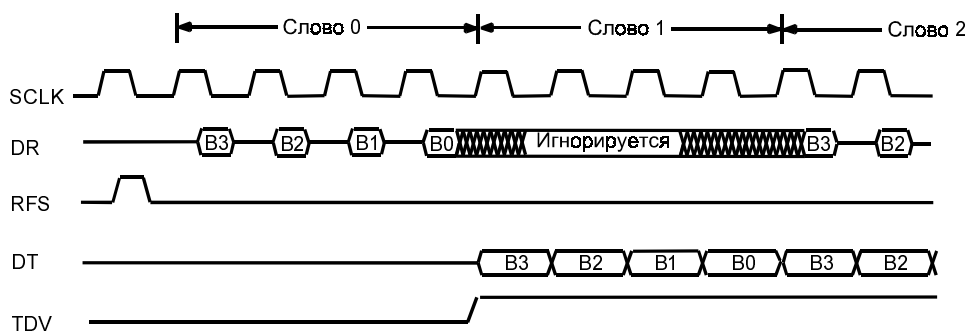


Рис. 5.26 Начало многоканальной операции

На рис. 5.27 показан полный блок данных длиной в 24 слова в многоканальном режиме, причем полные слова представлены в форме сигналов, а не в виде отдельных бит. Прием активен для всех слов, передача - только для слов 0-3, 8-11 и 16-19.

**Примечание:** ADSP-2105 имеет только один последовательный порт (SPORT1) и не поддерживает многоканальных операций.

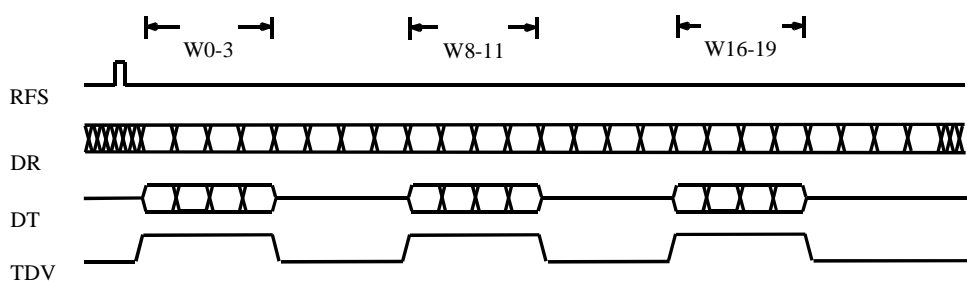


Рис. 5.27 Полный пример работы в многоканальном режиме

### 5.13 СИНХРОНИЗАЦИЯ РАБОТЫ ПОСЛЕДОВАТЕЛЬНОГО ПОРТА

Последовательные порты полностью поддерживают дуплексный режим работы и обычно управляются с помощью прерываний. То есть, каждый раз по окончании операции последовательным портом процессор генерирует внутреннее

## 5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ

прерывание. Для большинства операций реальное согласование прерываний последовательного порта во времени не играет существенной роли. Однако в некоторых сложных системах цифровой обработки сигналов важно знать временные отношения между прерыванием и операцией последовательного порта.

### 5.13.1 Задержка компандирования

Использование схемы компандирования вводит временную задержку двумя способами. Во-первых, сжатие или расширение значения данных занимает один цикл процессора. Во-вторых, когда оба последовательных порта запрашивают об операции сжатия или расширения данных в одном цикле, приоритетом обладает SPORT0; в этом случае SPORT1 вынужден ожидать в течение одного цикла процессора. Более подробно о компандировании см. в предыдущем разделе данной главы.

### 5.13.2 Задержка тактовой синхронизации

Некоторые временные характеристики последовательного порта зависят от тактового генератора процессора. Другие - от последовательного тактового генератора (SCLK0 или SCLK1). Эти тактовые генераторы асинхронны. Независимо от того, генерируются ли последовательные тактовые синхроимпульсы внутренне или внешне, существует задержка, связанная с синхронизацией последовательных тактовых синхроимпульсов с тактовыми импульсами процессора. Эта задержка различна для прерываний приема и передачи, как объясняется в нижеследующих разделах.

#### 5.13.2.1 Задержка, связанная с началом работы последовательного порта

Последовательный порт становится доступным только через два цикла последовательного тактового генератора (SCLK) после записи соответствующего значения в регистр управления системой. С наступлением следующего (третьего) цикла последовательного тактового генератора порт ожидает прихода сигнала кадровой синхронизации.

### 5.13.3 Временные характеристики сигналов кадровой синхронизации

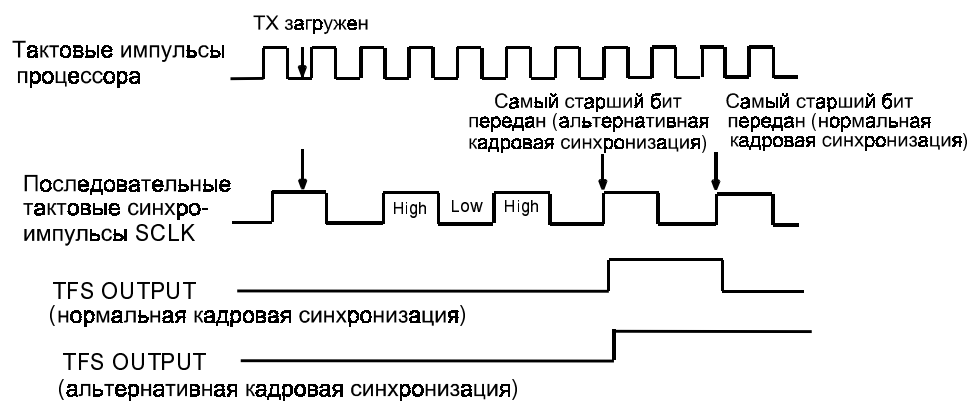
Все, что требуется от программиста для передачи данных через последовательный порт, - это осуществить пересылку данных в соответствующий регистр TX при помощи команды типа:

$TX0 = A X 0$  ;

## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5

Как только данные записываются в регистр TX, после синхронизирующей задержки генерируется сигнал кадровой синхронизации. Эта задержка в свою очередь влияет на временные характеристики прерывания передачи последовательного порта. Длительность задержки определяется пятью факторами: частотой последовательных тактовых синхроимпульсов, тем, разрешено или нет компандирование, имеет ли место конфликтная ситуация аппаратных средств компандирования, закончена ли передача текущего слова и логического уровня тактового генератора (SCLK) после загрузки значения данных в регистр передачи.

TX загружен, SCLK по высокому уровню



TX загружен, SCLK по низкому уровню

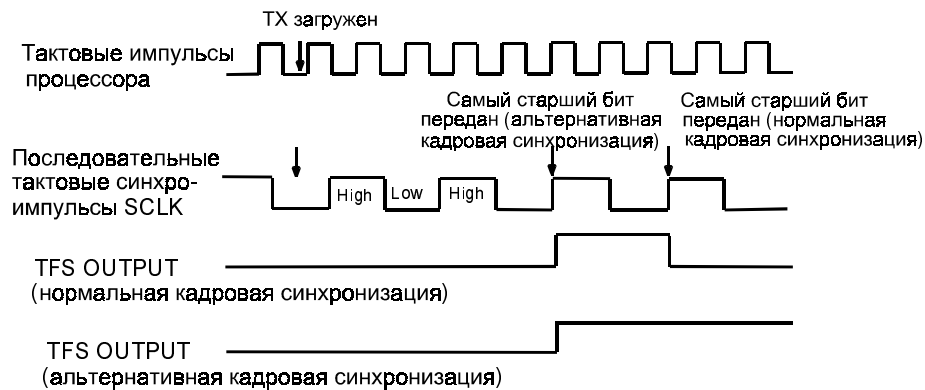


Рис. 5.28 Синхронизация тактовых синхроимпульсов

## 5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ

При внешне генерируемом сигнале кадровой синхронизации передача данных начинается по получению этого сигнала.

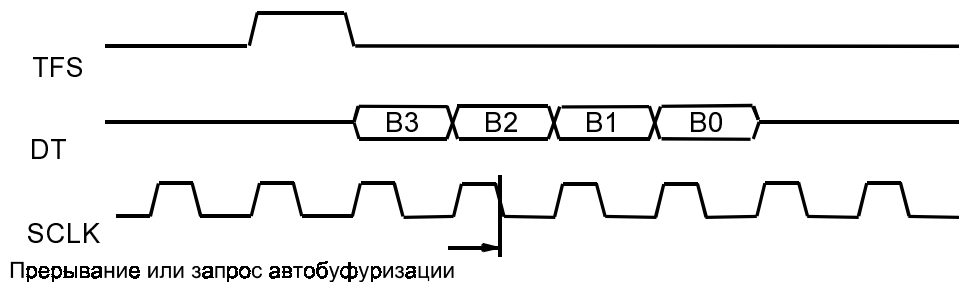
После загрузки регистра TX для обеспечения синхронизации требуется три фазы последовательного тактового синхроимпульса, HIGH, LOW, HIGH (высокий, низкий, высокий), - в приведенном порядке (см. рис. 5.28 на предыдущей стр.). После обеспечения синхронизации и генерирования сигнала кадровой синхронизации самый старший бит передаваемого слова смещается по тому же переднему фронту, что и сигнал кадровой синхронизации при использовании альтернативной кадровой синхронизации, и по переднему фронту следующего тактового синхроимпульса при нормальной кадровой синхронизации. Таким образом, в самом худшем случае задержка синхронизации составляет два цикла SCLK.

Если предыдущая пересылка данных не была завершена, возникает дополнительная задержка; до полного завершения предыдущей передачи невозможно загрузить регистр TX в передающий сдвигающий регистр.

### 5.13.4 Временные характеристики прерывания передачи

По окончании передачи самого старшего бита последующие биты передаются по переднему фронту тактовых синхроимпульсов (SCLK). Прерывание передачи (или запрос автобуферизации) генерируются внутренне по заднему фронту тактового синхроимпульса во время передачи второго бита (см. рис. 5.29). Такое согласование во времени дает программе время для загрузки следующих данных в регистр TX при непрерывной передаче данных.

Прерывание передачи, как и все другие прерывания, должно быть синхронизировано с тактовым генератором процессора. Обслуживание этого прерывания осуществляется с теми же временными задержками, что и обслуживание других прерываний.



**Рис. 5.29 Временные характеристики прерывания последовательного порта или запроса автобуферизации, передача слов длиной 4 бита, компандирование не разрешено**

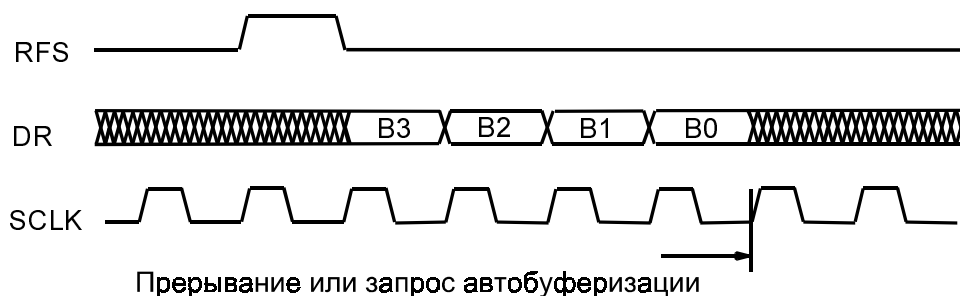
Прерывание передачи означает, главным образом, что все готово для записи данных в регистр TX.

## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5

### 5.13.5 Временные характеристики прерывания приема

Принимающая часть последовательного порта фиксирует данные на выводе DR по заднему фронту тактового синхроимпульса (SCLK).

Временные характеристики прерывания приема отличаются от прерывания передачи. Прерывание приема или запрос автобуферизации происходит только после того, как было получено все слово данных. Запрос на прерывание происходит по переднему фронту SCLK после приема слова (рис. 5.30) и указывает, что с регистра RX можно считать новые данные.

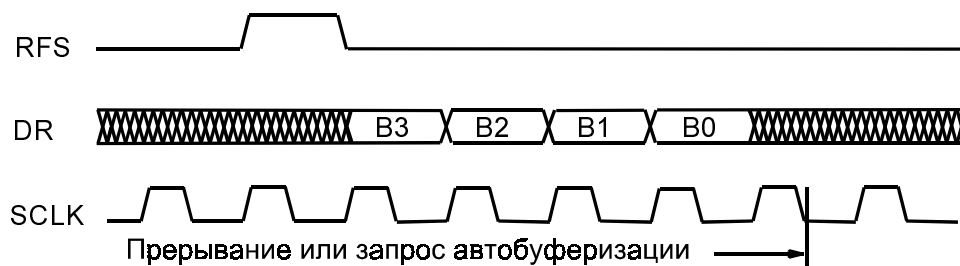


**Рис. 5.30** Временные характеристики прерывания последовательного порта или запроса автобуферизации, прием слов длиной 4 бита, компандирование не разрешено

Компандирование вызывает задержку таким же образом, как и при передаче. Однако, эта задержка прозрачна (незаметна), так как прерывание приема генерируется после того, как было осуществлено расширение данных.

Самый младший бит принимается по заднему фронту SCLK. На синхронизацию с тактовым генератором процессора уходит один цикл процессора. Через один цикл процессора, если компандирование разрешено, последовательный порт пытается расширить данные, а другой последовательный порт не использует в это время схему компандирования. Задержки, связанные с компандированием, возникают, как было описано выше, до генерирования прерывания приема. Обслуживание прерывания приема осуществляется аналогично обслуживанию других прерываний.

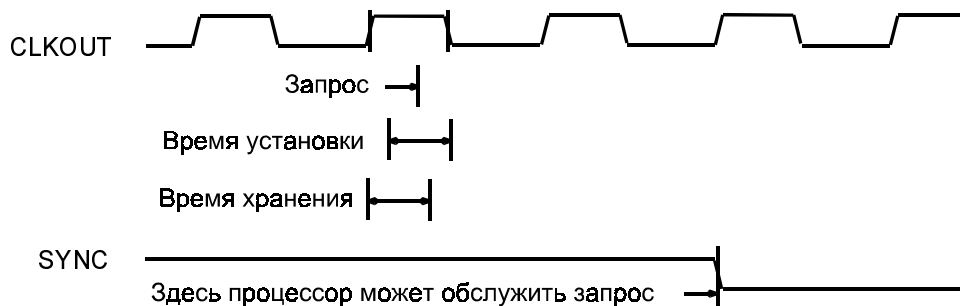
## 5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ



**Рис. 5.31** Временные характеристики прерывания последовательного порта или запроса автобуферизации, прием слов длиной 4 бита, компандирование разрешено

### 5.13.6 Синхронизация автобуферизации и прерываний

Последовательные порты являются системой, асинхронной процессору, даже в том случае, когда процессор сам обеспечивает последовательные тактовые синхроимпульсы. В процессоре имеется внутренняя схема, которая синхронизирует запросы с тактами процессора. На рис. 5.32 показана синхронизирующая задержка для последовательных портов при допущении, что время установки и хранения согласовываются в текущем цикле процессора. Время установки и хранения для запроса последовательного порта те же самые, что показаны в карте данных для сигнала **IRQ 2**. Если время установки и хранения не согласовываются, добавляется дополнительный цикл задержки.



**Рис. 5.32** Синхронизация запросов на прерывание или автобуферизацию с тактовым генератором процессора

Как показано на схеме выше, перед подачей запроса последовательного порта на процессор имеется задержка длительностью в два цикла процессора. Те же самые задержки возникают для всех внешних прерываний. Процессор может

## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5

обслужить прерывание или запрос автобуферизации только в конце или начале командного цикла, поэтому до достижения завершения команды могут добавляться дополнительные циклы задержки.

### **5.13.7 Задержки, связанные с ожиданием завершения выполнения команды**

В некоторых ситуациях выполнение команды может занять более одного цикла процессора. Любая из нижеследующих ситуаций может привести к задержке возможности процессора обслужить отложенное прерывание или запрос автобуферизации:

- Состояния ожидания внешней памяти
- Запрос шины, когда требуется внешний доступ ( в режиме "GO")
- Запрос шины при неразрешенном режиме "GO"
- Выполнение единичной команды требует нескольких внешних доступов
- Отложенное прерывание или запрос автобуферизации с более высоким приоритетом
- Прерывание маскируется

В конце командного цикла процессор обслужит многочисленные отложенные прерывания или запросы автобуферизации в следующем порядке, согласно их приоритету:

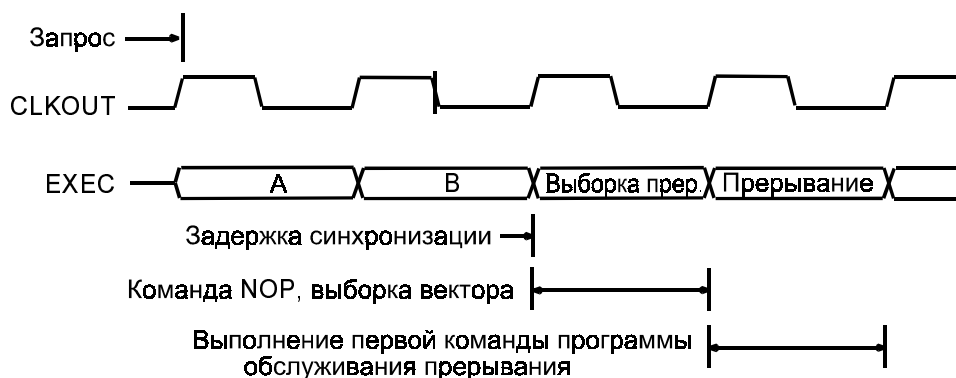
- Автобуферизация передачи SPORT0 (кроме ADSP-2105) - высший приоритет
- Автобуферизация приема SPORT0 (кроме ADSP-2105)
- Автобуферизация передачи SPORT1
- Автобуферизация приема SPORT1
- Немаскируемые отложенные прерывания в порядке их приоритета

### **5.13.8 Пример обслуживания прерывания и запроса автобуферизации**

На следующей схеме (рис. 5.33) показано осуществление прерывания последовательного порта на основе запроса, удовлетворяющего требованиям к времени установки и хранения.

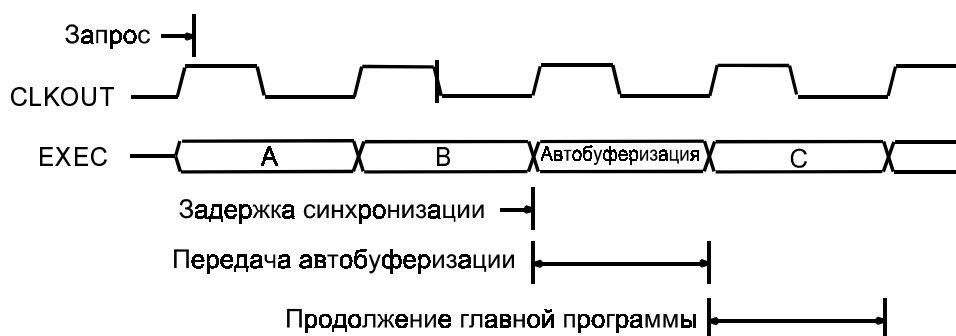
Дополнительный цикл задержки требуется на выборку первой команды программы обслуживания прерываний. Прерывание может обслуживаться только в конце командного цикла. В примере выше было сделано допущение, что все команды выполнялись за один цикл процессора.

## 5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ



**Рис. 5.33 Пример обслуживания прерывания**

Схема на рис. 5.34 иллюстрирует последствия запроса автобуферизации, который отвечает требованиям к времени установки и хранения.



**Рис. 5.34 Пример обслуживания автобуферизации**

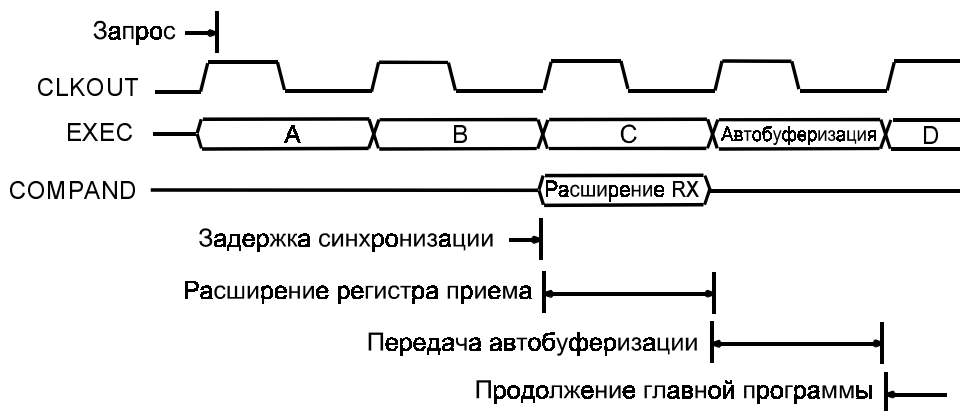
При автобуферизации определенное количество циклов требуется для выполнения передачи данных, однако подбор команд не вызывает дополнительных потерь и задержек. В приведенной выше схеме было сделано допущение, что все команды и пересылки данных происходят в одном цикле процессора.

### 5.13.9 Задержка при компандировании принимаемых данных

Кроме циклов, используемых для синхронизации, могут возникать дополнительные задержки, связанные с компандированием принимаемых данных. Синхронизированный запрос используется процессором для принятия решения, должно ли расширенное значение быть записано в регистр приема. Это возможно только в конце командного цикла, причем в определенный момент времени содержимое только одного регистра приема может подвергнуться расширению. В процес-

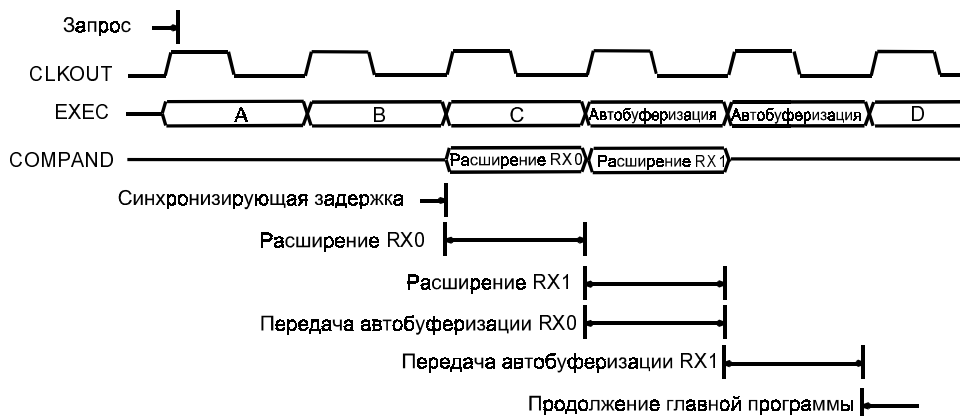
## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5

сорах с двумя последовательными портами (т.е. всех процессорах семейства, кроме ADSP-2105) может возникнуть задержка из-за недоступности схемы компандирования одновременно для обоих портов. SPORT0 обладает высшим приоритетом. При разрешенном компандировании запрос автобуферизации или прерывания не генерируется до тех пор, пока содержимое регистра приема не подвергается расширению. На схемах ниже показаны примеры автобуферизации с компандированием и связанные с этим задержки.



**Рис. 5.35 Пример компандирования принимаемых данных**

Следующая схема иллюстрирует задержку, возникающую, когда имеется два отложенных запроса автобуферизации приема при разрешенном компандировании.



**Рис. 5.36 Пример компандирования принимаемых данных при работе двух последовательных портов**

## 5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ

### 5.13.10 Прерывания при разрешенной автобуферизации

При разрешенной автобуферизации в случаях, когда модификация адреса во время операции автобуферизации приводит к возвращению по модулю, происходит прерывание последовательного порта. В этом случае также должна учитываться синхронизирующая задержка. Пример такого прерывания приводится на схеме ниже.



Рис. 5.37 Пример прерывания при разрешенной автобуферизации

### 5.13.11 Осложнения, возникающие в особых ситуациях

В большинстве случаев задержки, связанные с компандированием, автобуферизацией последовательного порта, и задержки прерываний незаметны для программы пользователя. При попытке использовать один и тот же регистр I для более, чем одного канала автобуферизации важно убедиться в том, что задержки не нарушают правильный порядок выполнения операций. Например, если последовательный порт осуществляет непрерывную пересылку данных и при этом приемное и передающее устройства работают с одним и тем же сигналом кадровой синхронизации, то порядок операций автобуферизации передачи и приема или прерываний может нарушиться под влиянием показанных на схеме ниже задержек.

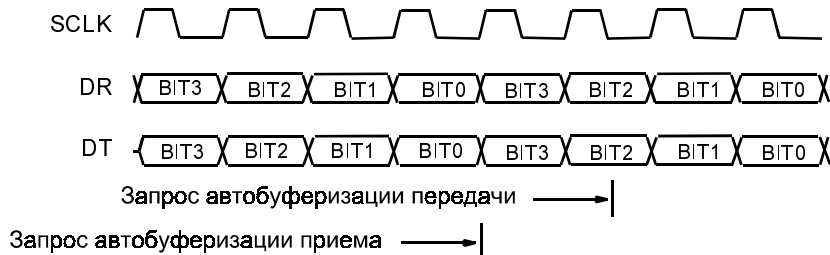


Рис. 5.38 Использование одного индексного регистра для автобуферизации приема и передачи

## ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ 5

Если процессор может свободно обработать запросы автобуферизации в том порядке, в каком они были сгенерированы, автобуферизация приема предшествует автобуферизации передачи. Порядок этих операций может измениться, когда процессор не способен обработать запросы из-за любой из вышеупомянутых задержек. В этом случае между запросами проходит  $1\frac{1}{2}$  цикла тактовых синхроимпульсов. Если процессор отрабатывает запрос шины, состояние ожидания или другие задержки, превышающие  $1\frac{1}{2}$  цикла тактовых синхроимпульсов, обе операции автобуферизации могут быть отложены. Так как автобуферизация передачи имеет более высокий приоритет, запрос на нее будет выдан первым. Приоритет запросов автобуферизации затрудняет или, в некоторых случаях, делает невозможным использование только одного регистра I. Пока не возникает задержек, превышающих разницу временных интервалов запросов, использование одного регистра I для автобуферизации последовательного порта вполне возможно.

## **5 ПОСЛЕДОВАТЕЛЬНЫЕ ПОРТЫ**