

## 32. 同步串行控制器 (SSC)

### 32.1 描述

Atmel 同步串行控制器(SSC)提供与外部设备的同步通信。支持多种用于语音和电信应用中常用的串行同步通信协议，如 I2S，短帧同步，长帧同步等。

SSC 包含一个独立的接收器和发送器和一个共用时钟分频器。接收器和发送器每个接口由三个信号：针对数据的 TD/RD 信号，针对时钟的 TK/RK 信号和针对帧同步的 TF/RF 信号。可编程设定为自动启动或在帧同步信号上检测到不同事件时启动。

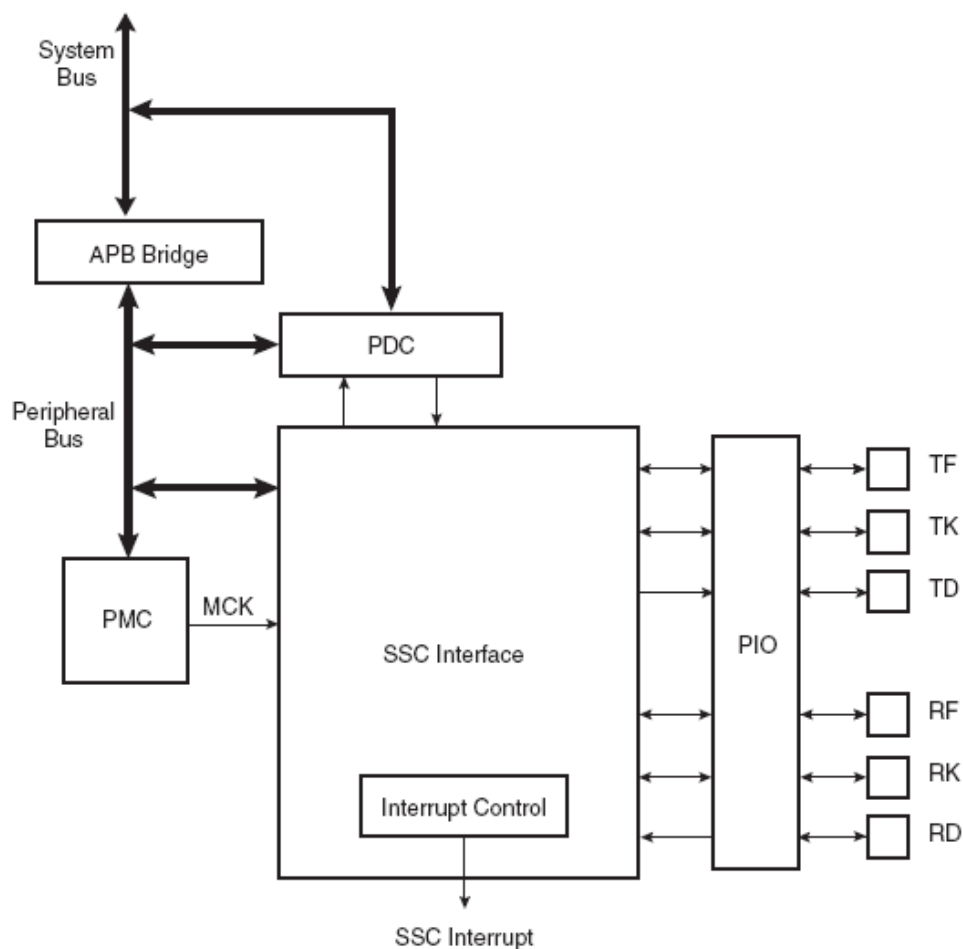
SSC 的可编程高电平和两个 32 位专用 PDC 通道，可在无处理器干涉时进行连续的高比特率的数据传输。

基于与两个 PDC 通道连接的特性，SSC 允许在低处理器开销下连接以下设备：

- 主/从模式下 CODEC
- 专用串行接口的 DAC，特别是 I2S
- 磁卡读取器

### 32.2 框图

图 32-1 框图



### 32.3 应用框图

图 32-2 应用框图

OS 或 RTOS 驱动	电源管理	中断管理	测试管理
SSC			
串行 AUDIO	CODEC	时间槽管理	帧管理
			线接口

### 32.4 引脚名称列表

表 32-1 I/O 口线描述

引脚名称	引脚描述	类型
RF	接收器帧同步	输入/输出
RK	接收器时钟	输入/输出
RD	接收器数据	输入
TF	发送器帧同步	输入/输出
TK	发送器时钟	输入/输出
TD	发送器数据	输出

### 32.5 附属产品

#### 32.5.1 I/O 口线

用于连接兼容的外部设备的引脚可与 PIO 口线复用。

使用 SSC 接收器之前，必须将 PIO 控制器配置为专用的 SSC 接收器 I/O 口线的 SCC 外设模式。

使用 SSC 发送器之前，必须将 PIO 控制器配置为专用的 SSC 发送器 I/O 口线的 SCC 外设模式。

#### 32.5.2 电源管理

SSC 时钟不是连续的。可以通过电源管理控制器 (PMC) 协调同步 SSC 接口，因此编程者必须首先配置 PMC 来使能 SSC 时钟。

#### 32.5.3 中断

SSC 接口连接于高级中断控制器 (AIC) 的一个中断口线。处理中断需要在配置 SSC 前编程 AIC。

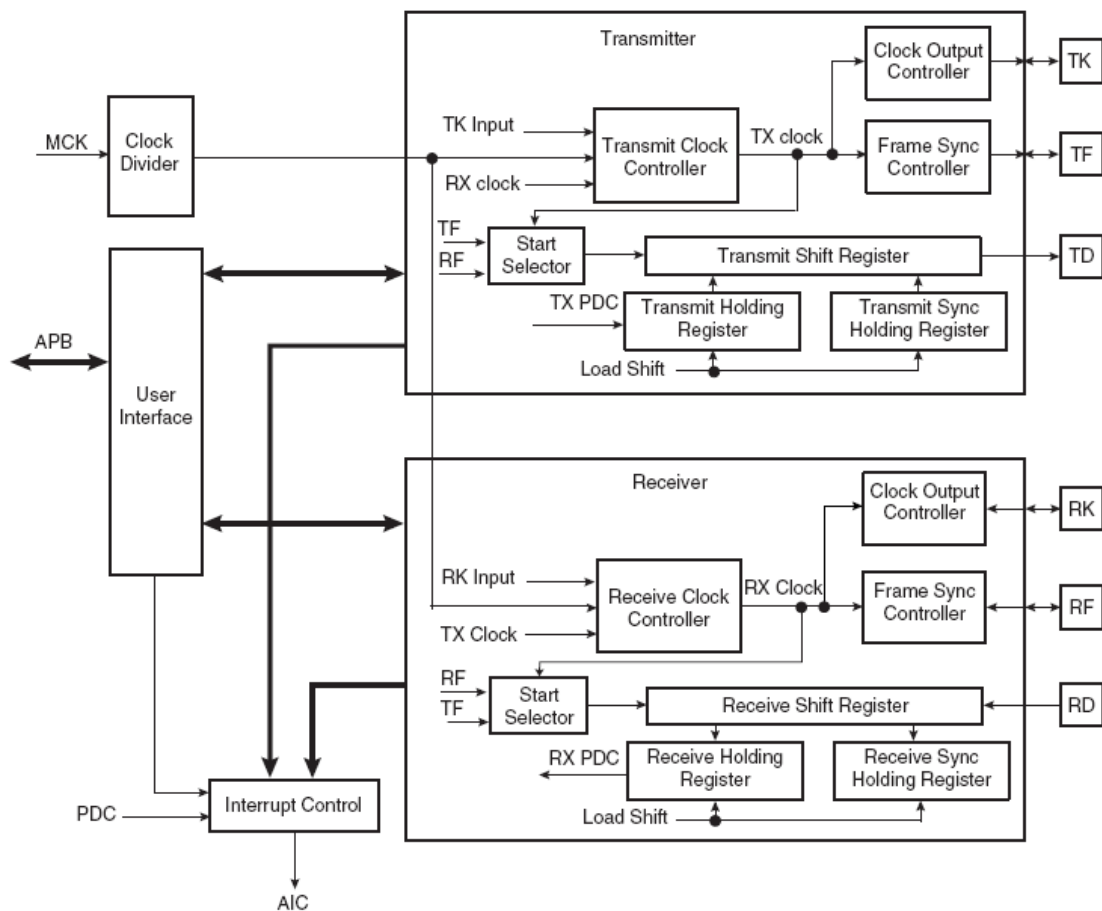
可通过配置 SSC 中断屏蔽寄存器使能和禁用所有的 SSC 中断。每个挂起的或未屏蔽的 SSC 中断源将出现在 SSC 中断线上。SSC 中断服务子程序可通过读取 SSC 中断状态寄存器来得到中断源。

### 32.6 功能描述

这一章包括以下说明：SSC 功能框图，时钟管理器，数据格式，启动，发送器，接收器和帧同步。

接收器和发送器独立运行。然而，可以通过编程接收器来使用发送时钟或当发送器启动时开始数据传输，使得接收器和发送器同步工作。另外，可通过编程发送器使用接收时钟或当接收启动时开始数据传输实现同步。可编程发送器和接收器使其运行于 TK 或 RK 提供的时钟信号。这就允许 SSC 支持很多的从模式数据传输。TK 和 RK 引脚上允许的最大时钟速率为主控时钟的 2 分频。

图 32-2 SSC 功能框图



### 32.6.1 时钟管理

发送器时钟可由以下产生：

- TK I/O 上接收的外部时钟
- 接收器时钟
- 内部时钟分频器

接收器时钟可以由以下产生：

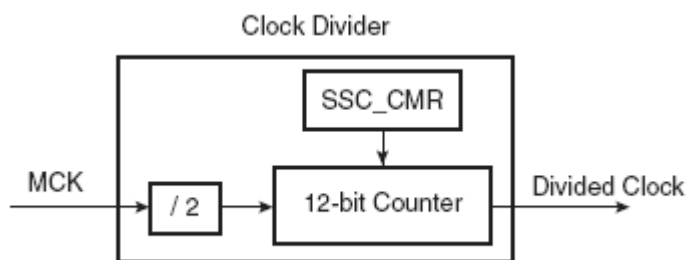
- RK I/O 上接收的外部时钟
- 发送器时钟
- 内部时钟分频器

此外，发送器时钟可以在产生 TK I/O 焊盘上产生外部时钟，接收器时钟可在 RK I/O 上产生外部时钟

这就使得 SSC 支持很多的主从模式数据传输。

#### 32.6.1.1 时钟分频器

图 32-4 分频时钟框图



时钟模式寄存器 SSC\_CMCR 中的 12 位域 DIV 计数器和比较器（所以起最大值为 4095）决定主控时钟分频器，允许主控时钟多达 8190 分频。分频时钟提供给接收器和发送器。当将此域编程为 0，则不使用时钟分频器并保持无效。

当 DIV 设置为大于等于 1 的值，分频时钟为 2 倍 DIV 分频的主控时钟。分频时钟的每级电平持续时间为 DIV 倍的主控时钟。这就确保了 50% 占空比的周期分频时钟，不管 DIV 值是奇数或偶数。

图 32-5 分频时钟的产生

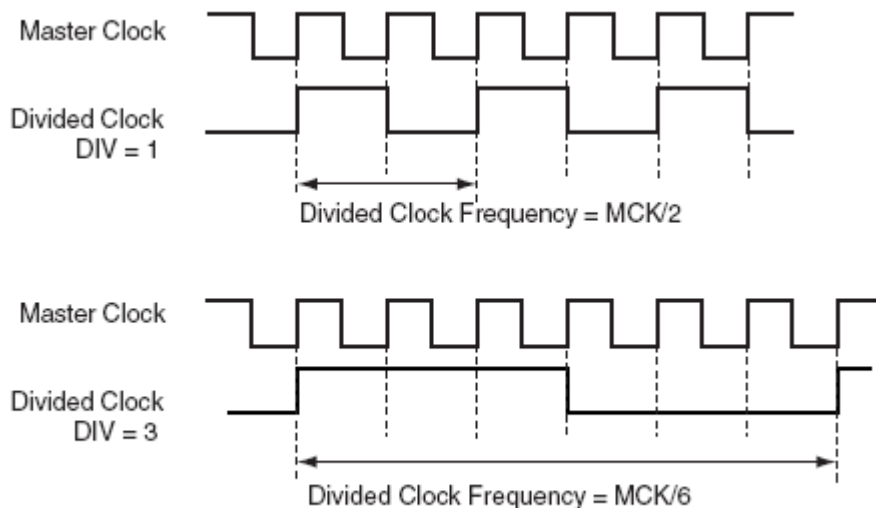


表 32-2

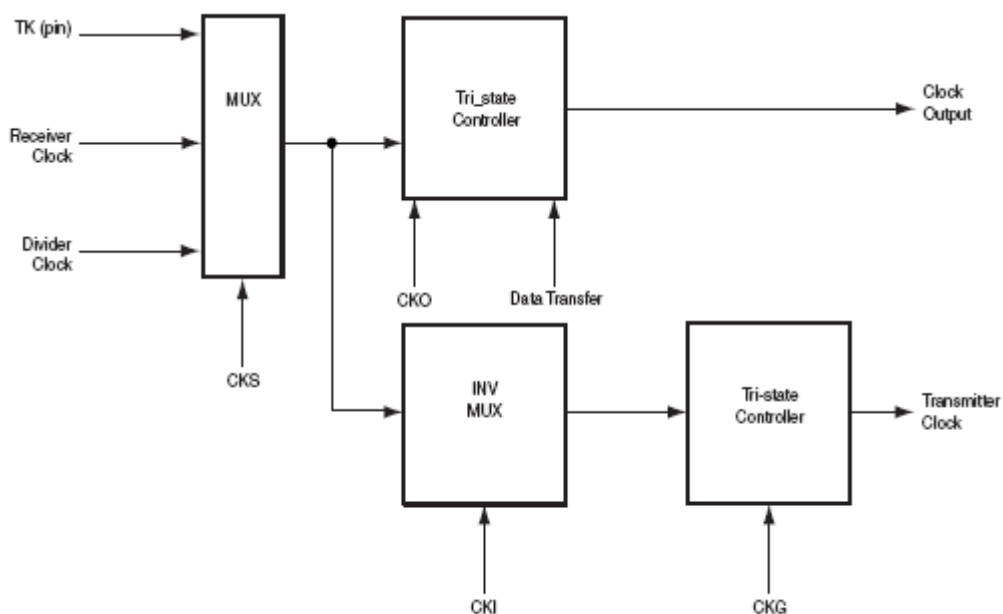
最大值	最小值
$\text{MCK} / 2$	$\text{MCK} / 8190$

### 32.6.1.2 发送器时钟管理

发送器时钟来自接收器时钟或分频器时钟或 TK I/O 上扫描的外部时钟。通过 SSC\_TCMR（发送时钟模式寄存器）中的 CKS 域选择发送器时钟。可通过 SSC\_TCMR 中的 CKI 位独立的反转发送时钟。

发送器还可连续的驱动 TK I/O 或受限于实际数据传输。通过 SSC\_TCMR 寄存器配置时钟输出。发送时钟反转位（CKI）对时钟输出无影响。编程 TCMR 寄存器选择 TK 引脚（CKS 域）并且同时连续发送时钟（CKO 域）可能导致不可预期的结果。

图 32-6 发送器时钟管理

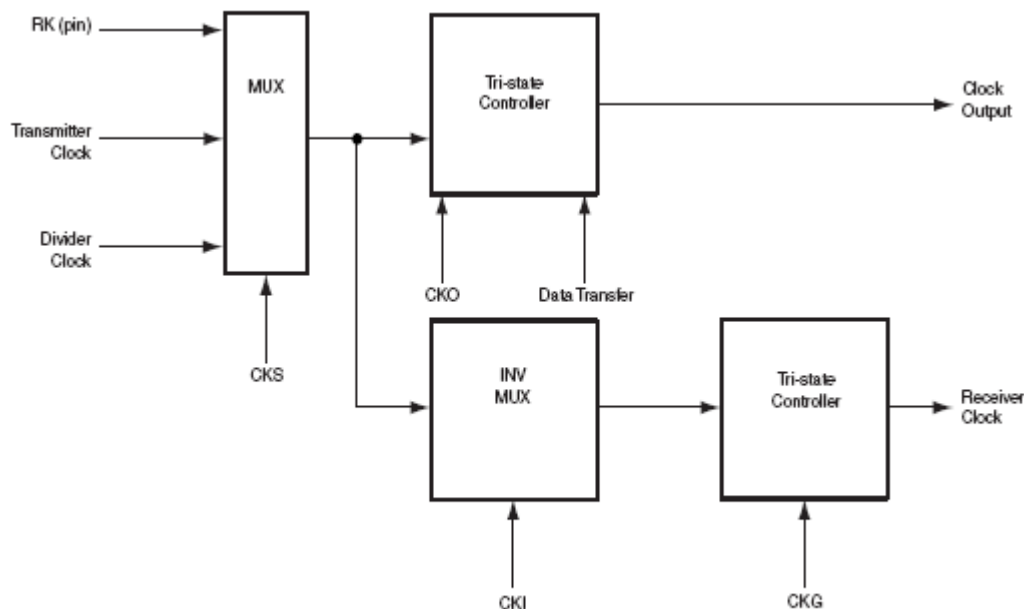


### 32.6.1.3 接收器时钟管理

接收器时钟来自发送器时钟或分频器时钟或 RK I/O 上的外部时钟。通过 SSC\_RCMR (接收时钟模式寄存器) 中的 CKS 域选择接收时钟。SSC\_RCMR 中的 CKI 位可独立的反转接收时钟。

接收器还可连续的驱动 RK I/O 或受限于实际的数据传输。通过 SSC\_RCMR 寄存器配置时钟输出。接收时钟反转位 (CKI) 不影响时钟输出。编程 RCMR 寄存器来选择 RK 引脚 (CKS 域)，同时连续接收时钟 (CKO 域) 可导致不可预期的结果。

图 32-7 接收器时钟管理



### 32.6.1.4 串行时钟率事宜

可编程发送器和接收器使其工作于 TK 或 RK 引脚上提供的时钟信号。这就允许 SSC 支持很多从模式的数据传输。此时，RK 引脚上允许的最高时钟速

率为：

-若接收器帧同步为输入则为 2 分频的主控时钟

-若接收器正同步为输出则为 3 分频的主控时钟

另外，TK 引脚上允许的最大时钟速率为：

-若发送帧同步为输入则为 6 分频的主控时钟

-若发送帧同步为输出则为 2 分频的主控时钟

### 32.6.2 发送器操作

通过一个启动事件触发一个发送帧，并可在数据发送前加入同步数据。

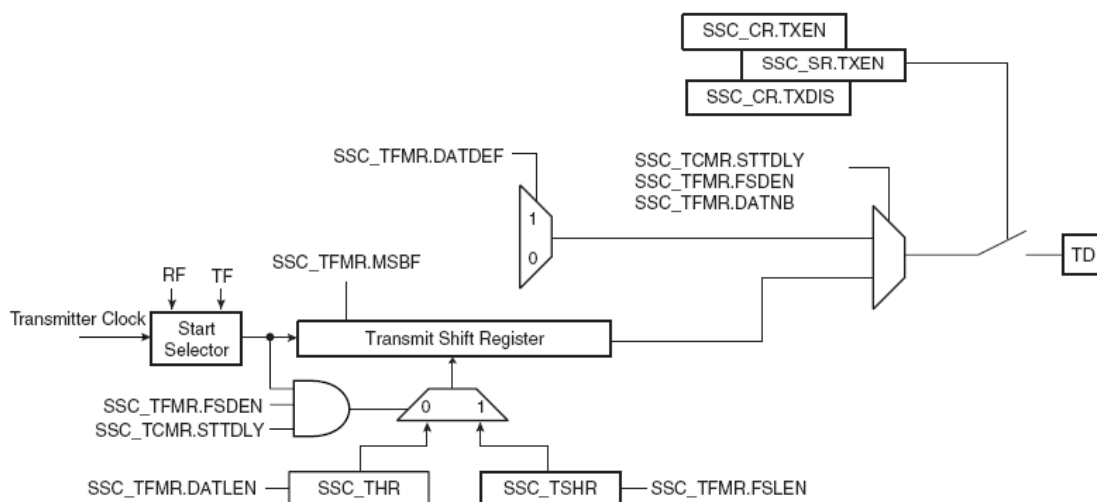
通过设置发送时钟模式寄存器(SSC\_TCMR)配置启动事件。见 452 页“启动”

通过设置发送帧同步模式寄存器(SSC\_TFMR)配置帧同步。见 454 页“帧同步”

发送数据时，发送器使用由发送器时钟信号协调同步的移位寄存器和在 SSC\_TCMR 中选择的启动模式。应用程序将数据写入 SSC\_THR 寄存器接着根据选择的数据格式将数据传输到移位寄存器。

当 SSC\_THR 和发送移位寄存器都空的时候，置位 SSC\_SR 中的状态标志位 TXEMPTY。当把发送保持寄存器数据传输到发送移位寄存器时，置位 SSC\_SR 中的状态标志位 TXRDY 并且可将附加数据装载到保持寄存器。

图 32-8 发送器框图



### 32.6.3 接收器操作

可通过一个启动事件触发一个接收帧并可在数据发送前加入同步数据。

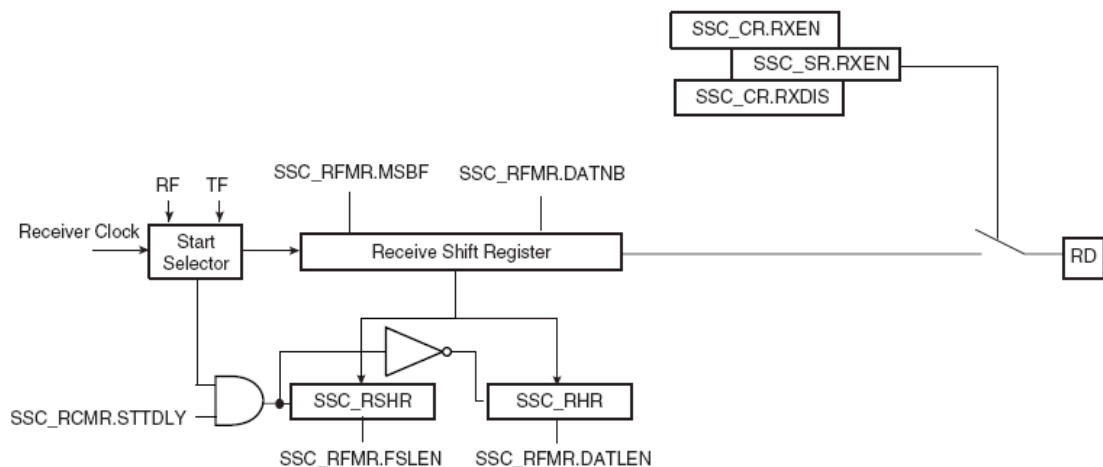
通过设置接收时钟模式寄存器(SSC\_RCMR)配置启动事件。见 452 页“启动”

通过设置接收帧模式寄存器(SSC\_RFMR)配置帧同步。见 454 页“帧同步”

接收器使用由接收器时钟信号协调同步的移位寄存器和 SSC\_RCMR 中选择的启动模式。从移位寄存器传输数据取决于选择的数据格式。

当接收器移位寄存器满时，SSC 将此数据传输到保持寄存器，置位 SSC\_SR 中的状态标志位 RXRDY 并且可在接收器保持寄存器中读取此数据。若在读取 RHR 寄存器前发生其他的传输，则置位 SSC\_SR 中的状态标志位 OVERUN 并且将接收器移位寄存器数据传输到 RHR 寄存器。

图 32-9 接收器框图



### 32.6.4 启动

当一事件发生,可分别在 SSC\_TCMR 的发送启动选择(START)域和 SSC\_RCMR 的接收启动选择域中编程发送器和接收器来启动其操作。

以下情况中,启动事件可独立编程:

- 连续。此时,只要向 SSC\_THR 写入一个字就启动发送,只要接收器使能就启动接收。
- 与发送器/接收器同步
- TF/RF 上检测到一个下降/上升沿
- TF/RF 上检测到一个低电平/高电平
- TF/RF 上检测到一个电平变化或一个边沿

可在发送/接收时钟寄存器(RCMR/TCMR)的任意边以同样方式编程启动。因此,可以在 TF(发送)上或 RF(接收)上启动

此外,当在比特流中比较检测到数据则接收器可以启动发送/接收帧模式寄存器(TFMR/RFMR)的FSOS域可完成TF/RF上输入/输出的检测。

图 32-10 发送启动模式

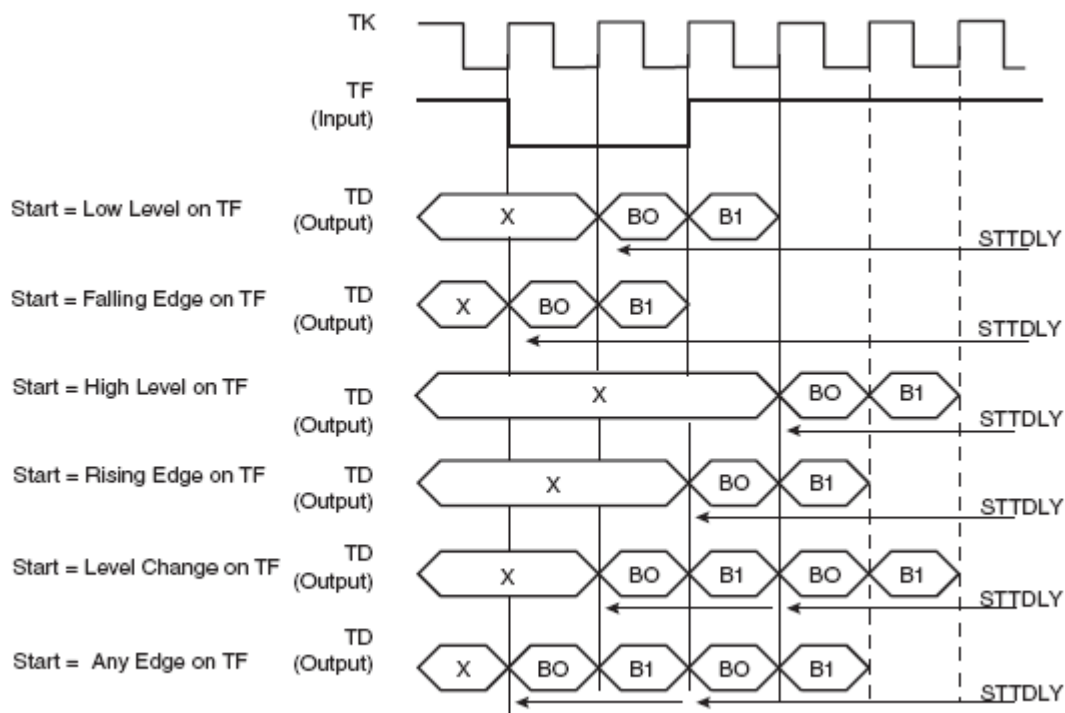
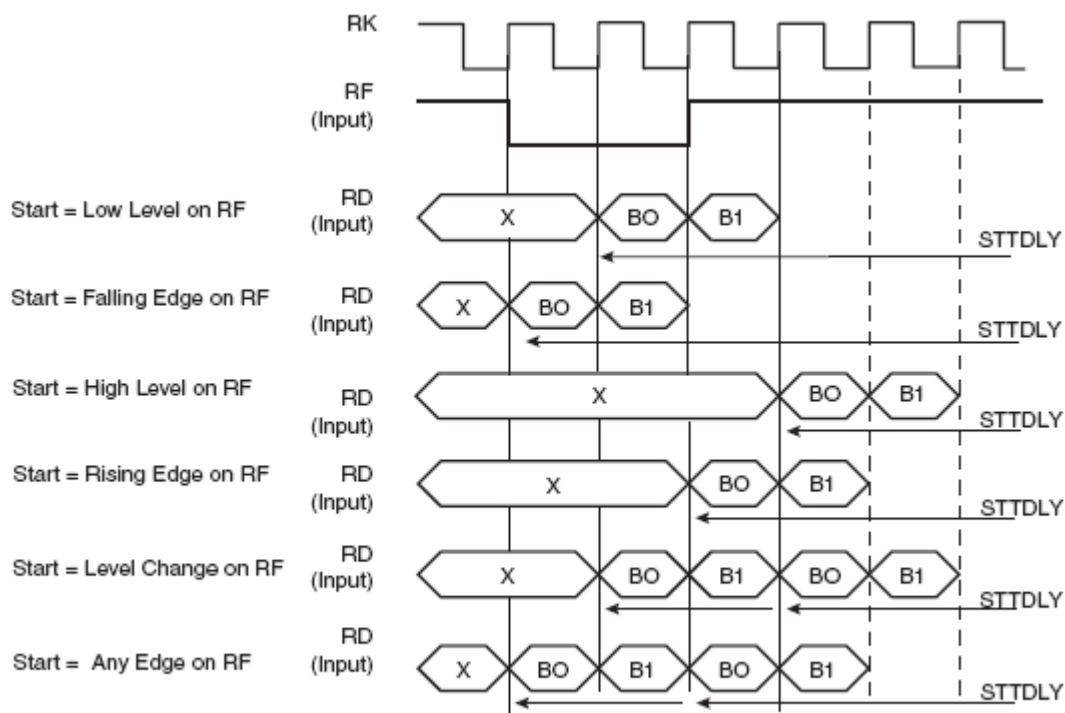


图 32-11 接收脉冲/边沿启动模式



### 32.6.5 帧同步

发送器和接收器帧同步引脚，TF 和 RF，可配置其用于产生各种不同的帧同步信号。接收帧模式寄存器 (SSC\_RFMR) 和发送帧模式寄存器 (SSC\_TFMR) 中的帧同步输出选择 (FSOS) 域用来选择所需波形。

- 支持数据传输期间可编程的低电平/高电平
- 支持数据传输前可编程的高电平或电平切换

若选择一个脉冲波形，SSC\_RFMR 和 SSC\_TFMR 中的帧同步长度 (FSLEN) 域编程脉冲的长度，从 1 比特时间到 16 比特时间

可通过 SSC\_RCMR 和 SSC\_TCMR 中的周期分频选择 (PERIOD) 域编程接收和发送帧同步脉冲输出的周期。

### 32.6.5.1 帧同步数据

帧同步数据在帧同步信号内发送或接收一个特定的标记。

帧同步信号内，接收器可采样 RD 口线并将数据存入接收同步保持寄存器并且发送器可将发送同步保持寄存器数据传输到移位寄存器。通过 SSC\_RFMR/SSC\_TFMR 中的 FSLEN 域来编程帧同步信号内采样/移出的数据长度并且起最大值为 16。

关于接收帧同步数据操作，若帧同步长度等于或小于启动时间和实际数据接收间的延迟，则通过接收移位寄存器在接收同步保持寄存器中执行数据采样操作。

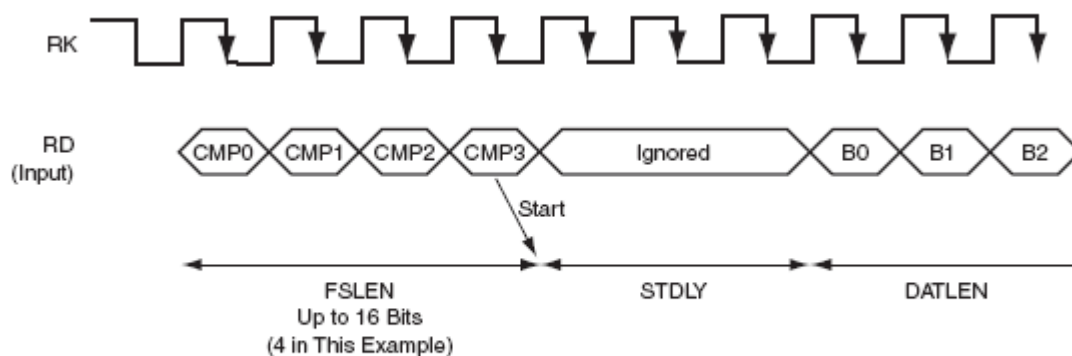
只要 SSC\_TFMR 中的帧同步数据使能位置位则发送器就发送帧同步操作。若帧同步长度等于或低于启动事件和实际数据发送间的延迟，则正常发送优先并且将发送同步保持寄存器中的数据传输到发送寄存器，然后移出。

### 32.6.5.2 帧同步边沿检测

由 SSC\_RFMR/SSC\_TFMR 中的 FSEDGE 域执行帧同步边沿检测。将 SSC 状态寄存器 (SSC\_SR) 中相应的标志位 RXSYN/TXSYN 设置为帧同步边沿检测 (RF/TF 信号)

### 32.6.6 接收比较模式

图 32-12 接收比较模式



#### 32.6.6.1 比较功能

FSLEN 定义比较模式 (Compare0, Compare1) 的长度和相比较的位数，但其最大值为 16 位。通常比较上次接收到的位和比较模式，Compare 0 可以是接收器的一个启动事件。此时接收器以上次接收的位的每个新采样值和 Compare 0 寄存器中 (SSC\_RCOR) Compare 0 模式做比较。当检测到此启动事件，则用户可通过写一个新的 Compare 0，或通过连续接收编程接收器来启动一个新的数据传输直到发生 Compare 1。SSC\_RCMR 中的 (STOP) 位完成此选择。

#### 32.6.7 数据格式

发送器和接收器的数据成帧格式都可通过发送器帧模式寄存器 (SSC\_TFMR)

和发送器帧模式寄存器 (SSC\_RFMR) 编程。

无论何种情况，用户可独立的选择：

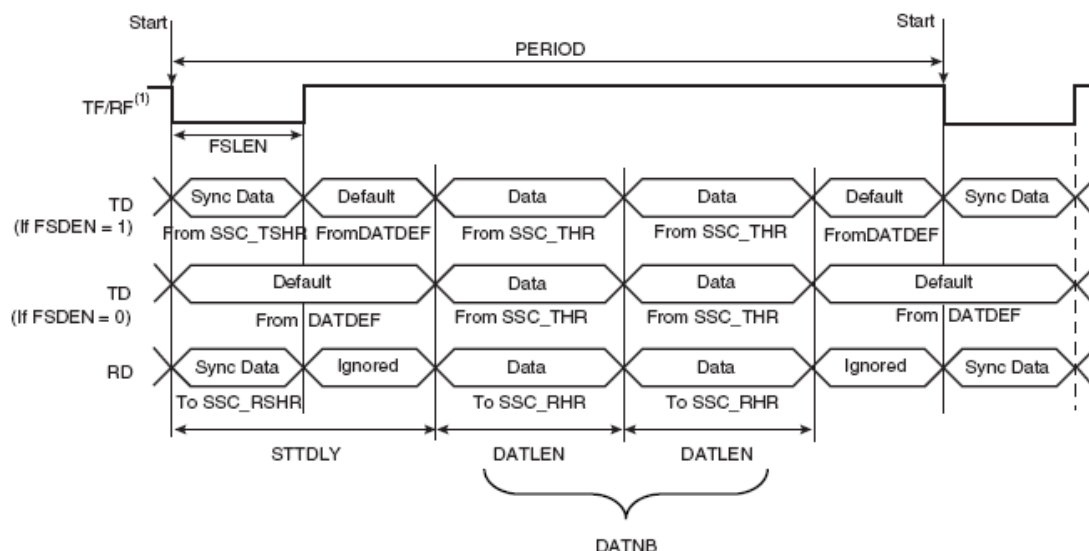
- 启动数据传输的事件 (START)
- 启动事件和首个数据位之间的比特周期数延迟 (STTDLY)
- 数据长度 (DATLEN)
- 每次启动事件传输的数据数 (DATNB)
- 每次启动事件同步传输长度 (FSLEN)
- 比特意义：高位或低位在先 (MSBF)

另外，当未数据传输时，可以使用发送器来传输同步并且选择 TD 引脚启动电平。可分别通过 SSC\_TFMR 中的帧同步数据使能 (FSDEN) 和数据缺省值 (DATDEF) 位完成。

表 32-3 数据帧寄存器

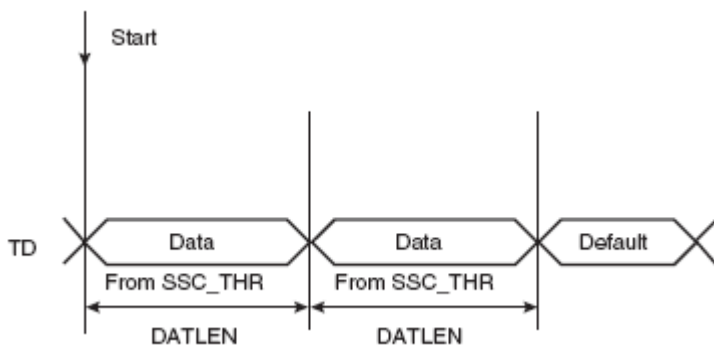
发送器	接收器	域	长度	备注
SSC_TFMR	SSC_RFMR	DATLEN	达 32	字长
SSC_TFMR	SSC_RFMR	DATNB	达 16	发送帧字数
SSC_TFMR	SSC_RFMR	MSBF		最高位在先
SSC_TFMR	SSC_RFMR	FSLEN	达 16	同步数据寄存器大小
SSC_TFMR		DATDEF	0 或 1	数据缺省值结束
SSC_TFMR		FSDEN		使能发送 SSC_TSHR
SSC_TCMR	SSC_RCMR	PERIOD	达 512	帧长度
SSC_TCMR	SSC_RCMR	STTDLY	达 255	发送启动延迟大小

图 32-13 边沿/脉冲启动模式下发送和接收帧格式



注意：1.TF/RF 下降沿输入举例

图 32-14 连续模式下发送帧格式

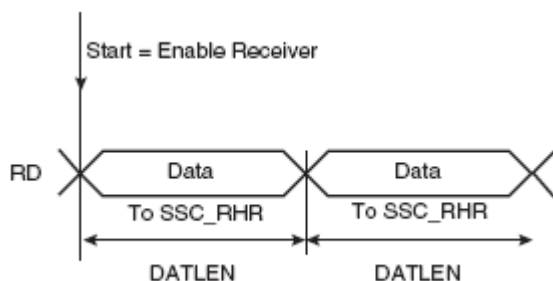


启动: 1.TXEMPTY 置为 1

2.写入 SSC\_THR

注意: 1.STTDLY 置为 0.此例中 SSC\_THR 被装载了两次。FSDEN 值不影响发送。连续模式下同步数据不能为输出。

图 32-15 连续模式下接收帧格式



注意: 1.STTDLY 置为 0

### 32.6.8 循环模式

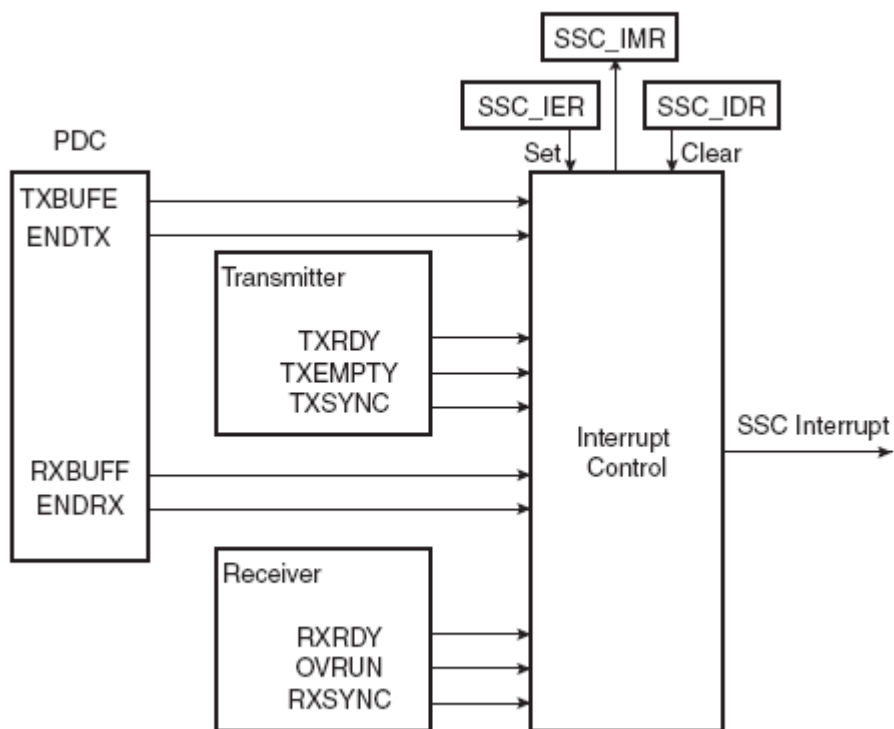
可编程接收器来从发送器接收发送。通过设置 SSC\_RFMR 中的循环模式 (LOOP)位完成。此时 RD 连接 TD, RF 连接 TF, RK 连接 TK。

### 32.6.9 中断

SSC\_SR 中大多数位有中断管理寄存器中相应的位

SSC 可被编程来当检测到一个事件时产生中断。通过写 SSC\_IER (中断使能寄存器) 和 SSC\_IDR (中断禁用寄存器) 来控制中断。通过分别置位和清零 SSC\_IMR (中断屏蔽寄存器) 中相应位来使能和和禁用这些寄存器, 通过激活连接于 AIC 的 SSC 中断口线来使得 SSC\_IMR 控制中断的产生。

图 32-16 中断框图



### 32.7 应用举例

SSC 可支持若干使用音频或高速串行链接的串行通信模式。一些标准应用如下图所示。所有通过 SSC 支持的串行链接应用未列出。

图 32-17 音频应用框图

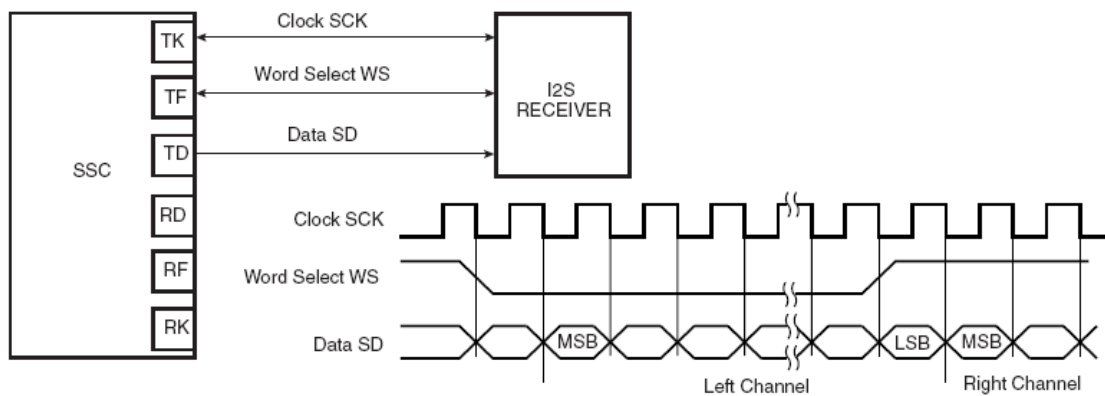


图 32-18 Codec 应用框图

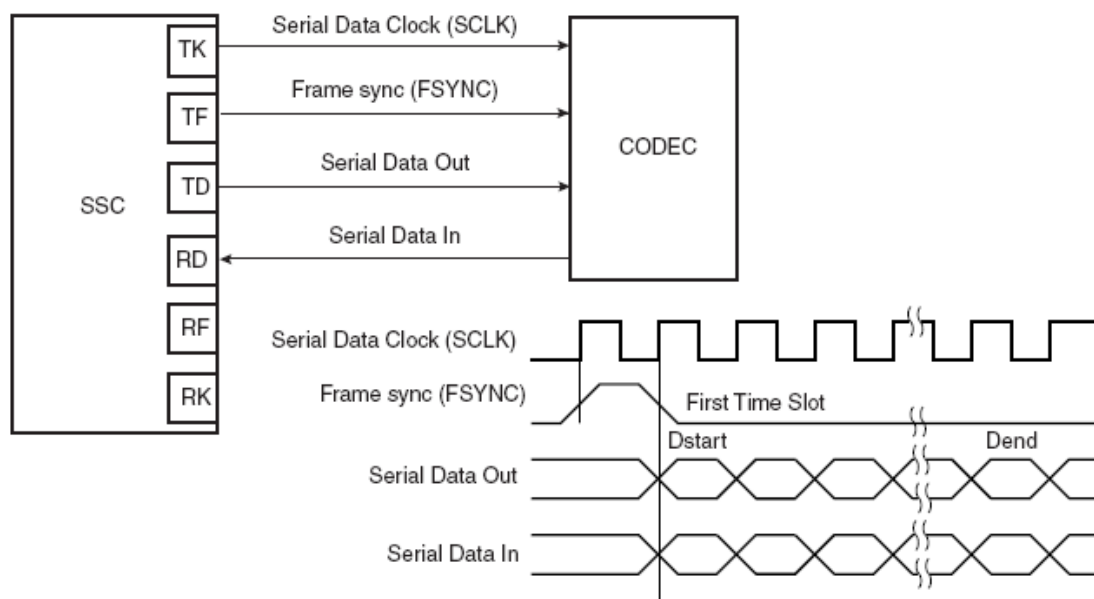
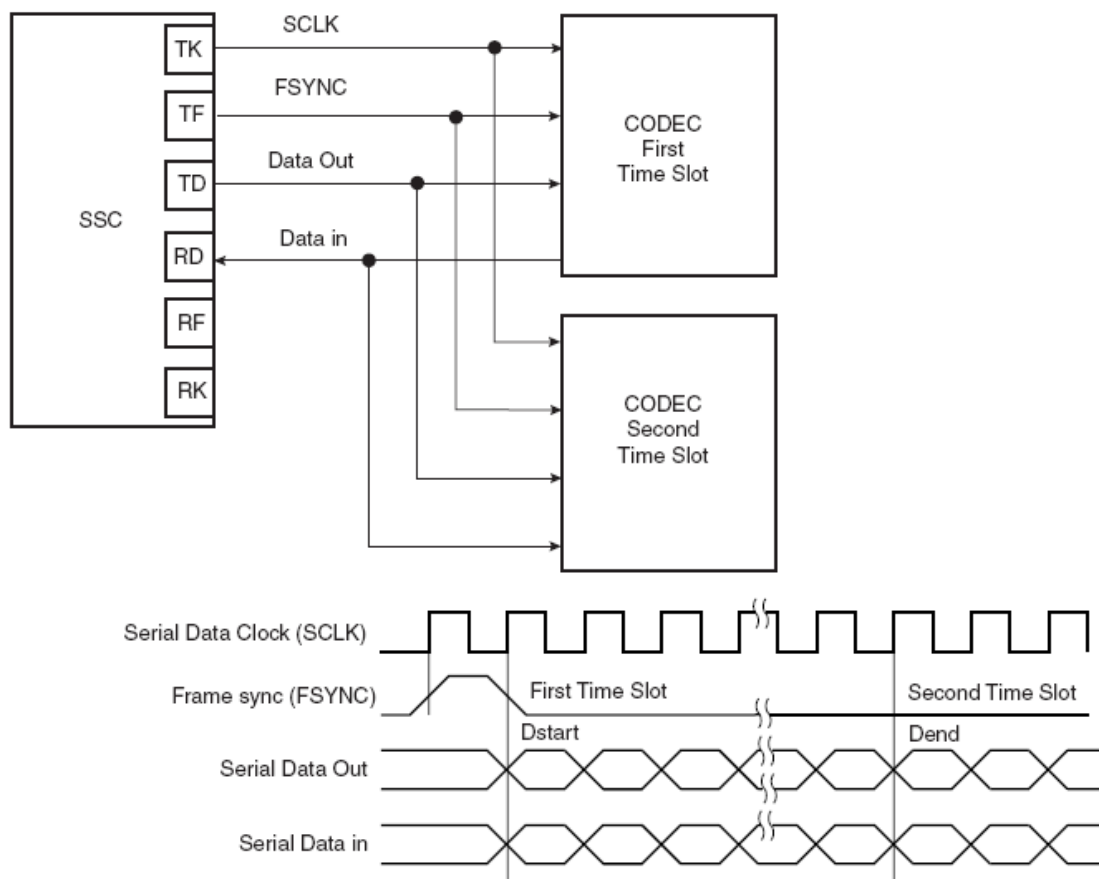


图 32-19 时间槽应用框图



### 32.8 同步串行控制器 (SSC) 用户接口

表 32-4 寄存器映射

偏移量	寄存器	寄存器名称	访问类型	复位值
0x0	控制寄存器	SSC_CR	只写	-
0x4	时钟模式寄存器	SSC_CMR	读/写	0x0
0x8	保留	-	-	-
0xC	保留	-	-	-
0x10	接收时钟模式寄存器	SSC_RCMR	读/写	0x0
0x14	接收帧模式寄存器	SSC_RFMR	读/写	0x0
0x18	发送时钟模式寄存器	SSC_TCMR	读/写	0x0
0x1C	发送帧模式寄存器	SSC_TFMR	读/写	0x0
0x20	接收保持寄存器	SSC_RHR	只读	0x0
0x24	发送保持寄存器	SSC_THR	只写	-
0x28	保留	-	-	-
0x2C	保留	-	-	-
0x30	接收同步保持寄存器	SSC_RSHR	只读	0x0
0x34	发送同步保持寄存器	SSC_TSHR	读/写	0x0
0x38	接收 Compare0 寄存器	SSC_RC0R	读/写	0x0
0x3C	接收 Compare1 寄存器	SSC_RC1R	读/写	0x0
0x40	状态寄存器	SSC_SR	只读	0x000000CC
0x44	中断使能寄存器	SSC_IER	只写	-
0x48	中断禁用寄存器	SSC_IDR	只写	-
0x4C	中断屏蔽寄存器	SSC_IMR	只读	0x0
0x50-0xFC	保留	-	-	-
0x100-0x124	为外设数据控制器 (PDC) 保留	-	-	-

#### 32.8.1 SSC 控制寄存器

名称: SSC\_CR

访问类型: 只写

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
SWRST	-	-	-	-	-	TXDIS	TXEN
7	6	5	4	3	2	1	0
-	-	-	-	-	-	RXDIS	RXEN

● RXEN: 接收使能

0: 无效

1: 若 RXDIS 未置位则使能接收

● RXDIS: 接收禁用

0: 无效

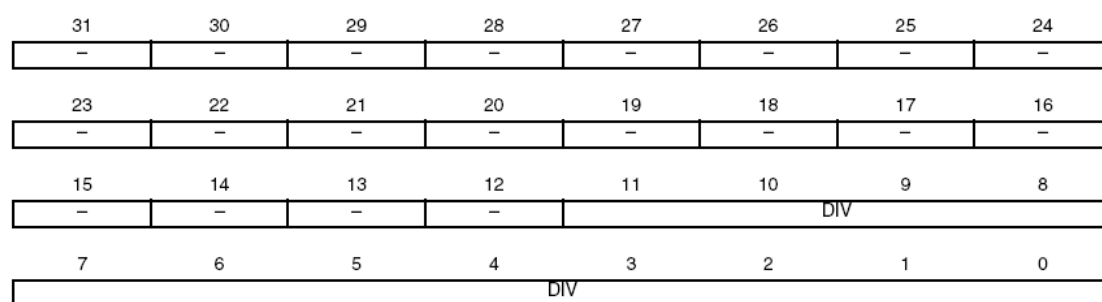
1: 禁用接收。若当前正在传输一个字符，在当前接收字符的结尾禁用。

- TXEN: 发送使能
- 0: 无效
- 1: 若 TXDIS 未置位则使能发送
- TXDIS: 发送禁用
- 0: 无效
- 1: 禁用发送。若当前正在发送一个字符，在发送当前字符的结尾禁用
- SWRST: 软件复位
- 0: 无效
- 1: 执行一个软件复位。比 SSC\_CR 中任何其它位优先级高

### 32.8.2 SSC 时钟模式寄存器

名称: SSC\_CMR

访问类型: 读/写



DIV: 时钟分频器

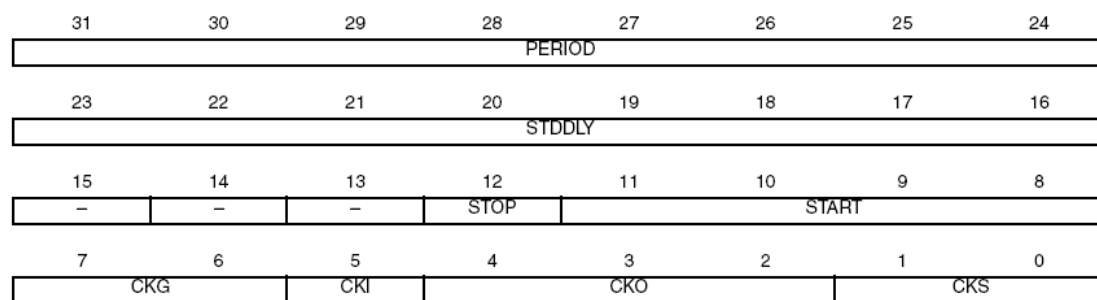
0: 时钟分频起无效

任何其它值: 分频时钟等于 2 倍 DIV 分频的主控时钟。最大比特率为 MCK/2。最小比特率为 MCK/2 x 4095 = MCK/8190

### 32.8.3 SSC 接收时钟模式寄存器

名称: SSC\_RCMR

访问类型: 读/写



- CKS: 接收时钟选择

CKS	选择的接收时钟
0x0	分频的时钟
0x1	TK 时钟信号
0x2	RX 引脚
0x3	保留

● CKO: 接收时钟输出模式选择

CKO	接收时钟输出模式	RK 引脚
0x0	无	只输入
0x1	连续接收时钟	输出
0x2	仅在数据传输期间的接收时钟	输出
0x3-0x7	保留	输出

● CKI: 接收时钟反转

0: 在接收时钟下降沿采样数据输入（数据和帧同步信号）。帧同步信号输出在接收时钟上升沿时被移出。

1: 在接收时钟上升沿采样数据输出（数据和帧同步信号）。帧同步信号输出在接收时钟下降沿时被移出。

CKI 仅影响接收时钟而不影响输出时钟信号

● CKG: 接收时钟选通选择

CKG	接收时钟选通
0x0	无，连续时钟
0x1	只有在 RF 低电平时接收时钟使能
0x2	只有在 RF 高电平时接收时钟使能
0x3	保留

● START: 接收启动选择

START	接收启动
0x0	连续，只要接收器使能，在前一次数据传输结束后立即启动
0x1	发送启动
0x2	检测 RF 信号上的低电平
0x3	检测 RF 信号上的高电平
0x4	检测 RF 信号上的下降沿
0x5	检测 RF 信号上的上升沿
0x6	检测 RF 信号上的任何电平变化
0x7	检测 RF 信号上的任何边沿
0x8	Compare 0
0x9-0xF	保留

● STOP: 接收停止选择

0: 数据传输完成后当启动 Compare0 时，接收器停止数据传输并且等待一个新的 compare0

1: 启动接收 Compare0 后，接收器运行在连续模式直到检测到 Compare1。

● STTDLY: 接收启动延迟

若 STTDLY 不为 0，将在启动事件和实际接收启动之间插入一个 STTDLY 时钟周期的延迟。当编程接收器使能和发送器同步启动时，延迟仍然适用。

注意：仔细设置 STTDLY 非常重要。若必须设置 STTDLY，则应该关于 TAG（接收同步数据）接收。

● PERIOD: 接收周期分频器选择

此域选择分频器来应用于选择的接收以产生新的帧同步信号。若为 0，则不产生 PERIOD 信号。若不为 0，则每  $2 \times (\text{PERIOD} + 1)$  个接收周期产生一个 PERIOD 信号

### 32.8.4 SSC 接收帧模式寄存器

名称: SSC\_RFMR

访问类型: 读/写

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	FSDGE
23	22	21	20	19	18	17	16
-	FSOS				FSLEN		
15	14	13	12	11	10	9	8
-	-	-	-	DATNB			
7	6	5	4	3	2	1	0
MSBF	-	LOOP	DATLEN				

● **DATLEN:** 数据长度

0: 禁用值 (不支持 1 位数据长度)

任何其它值: 比特流包括 DATLEN+1 数据位。此外, 它定义了接收器 PDC2 执行的传输大小。若 DATLEN 低于或等于 7, 则数据以字节传输。若 DATLEN 在 8 和 15 之间 (包含 15), 以半字传输, 其它任何值时, 以 32 位字传输。

● **LOOP:** 循环模式

0: 正常运行模式

1: TD 驱动 RD, TF 驱动 RF, TK 驱动 RK

● **MSBF:** 最高位在先

0: 首先采样比特流中数据寄存器的最低位

1: 首先采样比特流中数据寄存器的最高位

● **DATNB:** 每帧数据数

此域定义了每次传输启动后待发送数据字数, 其值等于 (DATNB + 1)

● **FSLEN:** 接收帧同步长度

此域定义了采样和存储在接收同步数据寄存器中的位数。当通过接收时钟模式寄存器中的 START 域选择此模式时, 他还决定了要和 Compare 0 或 Compare 1 寄存器比较的采样数据的长度。

此域和 FSLEN\_EXT 一起使用来决定接收帧同步信号的脉冲长度。

脉冲长度等于 FSLEN + 1 个接收时钟周期

● **FSOS:** 接收帧同步输出选择

FSOS	选择的接收帧同步信号	RF 引脚
0x0	无	仅输入
0x1	负脉冲	输出
0x2	正脉冲	输出
0x3	数据传输期间驱动为低	输出
0x4	数据传输期间驱动为高	输出
0x5	每次数据传输启动时切换	输出
0x6-0x7	保留	未定义

● **FSDGE:** 帧同步边沿检测

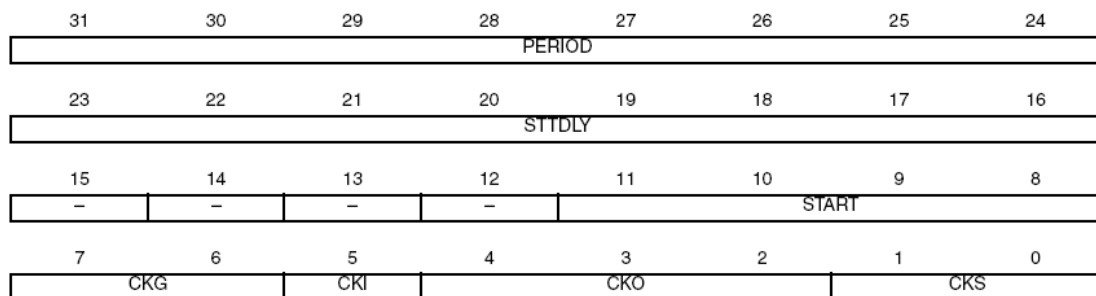
决定帧同步哪个边沿产生 SSC 状态寄存器中的中断 RXSYN

FSDGE	帧同步边沿检测
0x0	正边沿检测
0x1	负边沿检测

### 32.8.5 SSC 发送时钟模式寄存器

名称: SSC\_TCMR

访问类型: 读/写



● **CKS: 发送时钟选择**

CKS	选择的发送时钟
0x0	分频时钟
0x1	RK 时钟信号
0x2	TK 引脚
0x3	保留

● **CKO: 发送时钟输出模式选择**

CKO	发送时钟输出模式	TK 引脚
0x0	无	仅输入
0x1	连续发送时钟	输出
0x2	仅在数据传输期间的发送时钟	输出
0x3-0x7	保留	

● **CKI: 发送时钟反转**

0: 在发送时钟下降沿移出数据输出（数据和帧同步信号），在发送时钟上升沿采样帧同步信号输入

1: 在发送时钟上升沿移出数据输出（数据和帧同步信号），在发送时钟下降沿采样帧同步信号输入

CKI 仅影响发送时钟而不影响输出时钟信号。

● **CKG: 发送时钟选通选择**

CKG	发送时钟选通
0x0	无，连续时钟
0x1	只有 TF 低电平时发送时钟才使能
0x2	只有 TF 高电平时发送时钟才使能
0x3	保留

● **START: 发送启动选择**

START	发送启动
0x0	连续，只要向 SSC_THR 寄存器写入一个字（若发送使能），则在前一个数据传输结束后立即启动
0x1	接收启动
0x2	检测 TF 低电平信号
0x3	检测 TF 高电平信号
0x4	检测 TF 下降沿信号

0x5	检测 TF 上升沿信号
0x6	检测 TF 任何电平变化信号
0x7	检测 TF 任何边沿信号
0x8-0xF	保留

● **STTDLY**: 发送启动延迟

若 STTDLY 不为 0，则在启动事件和实际数据发送启动之间插入一个 STTDLY 时钟周期延迟。当编程发送器和接收器同步启动时，延迟仍然适用。

注意：必须仔细设置 STTDLY。若 STTDLY 对于 TAG（发送同步数据）发射太短，则数据代替 TAG 结束发送

● **PERIOD**: 发送周期分频器选择

此域选择应用于所选发送时钟的分频器以产生一个新的帧同步信号。若为 0.不产生周期信号。若不为 0，在每 2 x (PERIOD+1) 个发送周期时产生一个周期信号。

32.8.6 SSC 发送帧模式寄存器

名称: SSC\_TFMR

访问类型: 读/写

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	FSEDGE
23	22	21	20	19	18	17	16
FSDEN	FSOS			FSLEN			
15	14	13	12	11	10	9	8
-	-	-	-	DATNB			
7	6	5	4	3	2	1	0
MSBF	-	DATDEF	DATLEN				

● **DATLEN**: 数据长度

0: 禁用值（不支持 1 位数据长度）

任何其它值: 比特流含 DATLEN + 1 个数据位。此外，它定义了发送器的 PDC2 执行的传输大小。若 DATLEN 低于或等于 7，则以字节传输数据，若 DATLEN 在 8 到 15 之间（包含 15），则以半字传输，对任何其它值，以 32 位字传输。

● **DATDEF**: 数据缺省值

此位定义发送输出时 TD 引脚上的驱动电平。注意若 PIO 控制器将此引脚定义为多驱动，则只有在 SCC TD 输出为 1 时该引脚使能

● **MSBF**: 最高位在先

0: 首先移出数据寄存器的最低位

1: 首先移出数据寄存器的最高位

● **DATNB**: 每帧数据数

此域定义每次传输启动后待传输的数据字数，其值等于 (DATNB + 1)

● **FSLEN**: 发送帧同步长度

若 FSDEN 为 1，此域定义发送帧同步信号的长度和来自发送同步数据寄存器移出的比特数。

此域和 FSLEN\_EXT 被用来决定发送帧同步信号的脉冲长度。

脉冲长度等于 FSLEN + 1 个发送时钟周期

● FSOS: 发送帧同步输出选择

FSOS	所选发送帧同步信号	TF 引脚
0x0	无	仅输入
0x1	负脉冲	输出
0x2	正脉冲	输出
0x3	数据传输期间驱动为低电平	输出
0x4	数据传输期间驱动为高电平	输出
0x5	每次数据传输启动时切换	输出
0x6-0x7	保留	未定义

● FSDEN: 帧同步数据使能

0: 发送帧同步信号内用缺省值驱动 TD 口线

1: 发送帧同步信号的发送期间移出 SSC\_TSHR 值

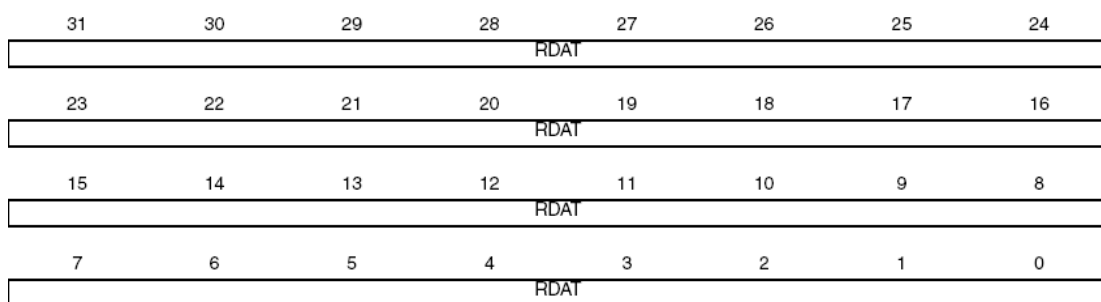
● FSEDGE: 帧同步边沿检测

定义哪个边沿将产生中断 TXSYN (状态寄存器)

### 32.8.7 SSC 接收保持寄存器

名称: SSC\_RHR

访问类型: 只读



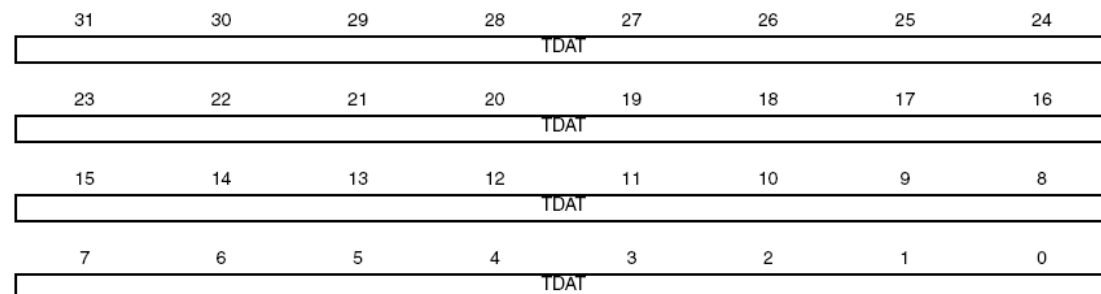
● RDAT: 接收数据

不管 SSC\_RFMR 中 DATLEN 定义的数据位数，数据右对齐

### 32.8.8 SSC 发送保持寄存器

名称: SSC\_THR

访问类型: 只写



● TDAT: 发送数据

不管 SSC\_TFMR 中 DATLEN 定义的数据位数，数据右对齐

### 32.8.9 SSC 接收同步保持寄存器

名称: SSC\_RSHR

方外类型: 只读

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
RSDAT							
7	6	5	4	3	2	1	0
RSDAT							

- RSDAT: 接收同步数据

### 32.8.10 SSC 发送同步保持寄存器

名称: SSC\_TSHR

访问类型: 读/写

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
TSDAT							
7	6	5	4	3	2	1	0
TSDAT							

- TSDAT: 发送同步数据

### 32.8.11 SSC 接收 Compare0 寄存器

名称: SSC\_RC0R

访问类型: 读/写

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
CP0							
7	6	5	4	3	2	1	0
CP0							

- CP0: 接收比较数据 0

### 32.8.12 SSC 接收 Compare1 寄存器

名称: SSC\_RC1R

访问类型: 读/写

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
CP1							
7	6	5	4	3	2	1	0
CP1							

- CP1: 接收比较数据 1

### 32.8.13 SSC 状态寄存器

名称: SSC\_SR

访问类型: 只读

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	RXEN	TXEN
15	14	13	12	11	10	9	8
-	-	-	-	HXSYN	TXSYN	CP1	CP0
7	6	5	4	3	2	1	0
RXBUFF	ENDRX	OVRUN	RXRDY	TXBUFE	ENDTX	TXEMPTY	TXRDY

- TXRDY: 发送就绪

0: 数据被装入 SSC\_THR 并且等待被装入发送移位寄存器 (TSR)

1: SSC\_THR 空

- TSEMPY: 发送空

0: 数据保持在 SSC\_THR 或正从 TSR 发送

1: 上次写入 SSC\_THR 的数据已被装载到 TSR 并且上次被装载到 TSR 的数据已被发送

- ENDTX: 发送结束

0: 自上次写 SSC\_TCR 或 SSC\_TNCR 寄存器 SSC\_TCR 未达到 0

1: 自上次写 SSC\_TCR 或 SSC\_TNCR 寄存器 SSC\_TCR 已达到 0

- TXBUFE: 发送缓冲区空

0: SSC\_TCR 或 SSC\_TNCR 值不为 0

1: SSC\_TCR 和 SSC\_TNCR 的值都为 0

- RXRDY: 接收就绪

0: SSC\_RHR 为空

1: 数据已接收并装载到 SSC\_RHR

- OVRUN: 接收超限

0: 自上次读取状态寄存器, 当前一次数据未读取时不向 SSC\_RHR 装载数据

1: 自上次读取状态寄存器, 当前一次数据未读取时向 SSC\_RHR 装载数据

- ENDRX: 接收结束

0: 数据写入接收计数器寄存器或接收下一次计数器寄存器

1: 当接收计数器寄存器达到 0 时 PDC 传输结束

● RXBUFF: 接收缓冲满

0: SSC\_RCR 或 SSC\_RNCR 值不为 0

1: SSC\_RCR 和 SSC\_RNCR 值都为 0

● CP0: Compare0

0: 自上次读取状态寄存器未产生 Compare0

1: 自上次读取状态寄存器未产生 Compare0

● CP1: Compare1

0: 自上次读取状态寄存器未产生 Compare1

1: 自上次读取状态寄存器未产生 Compare1

● TXSYN: 发送同步

0: 自上次读取状态寄存器未产生 Tx 同步

1: 自上次读取状态寄存器产生 Tx 同步

● RXSYN: 接收同步

0: 自上次读取状态寄存器未产生 Rx 同步

1: 自上次读取状态寄存器产生 Rx 同步

● TXEN: 发送使能

0: 发送禁用

1: 发送使能

● RXEN: 接收使能

0: 接收禁用

1: 接收使能

### 32.8.14 SSC 中断使能寄存器

名称: SSC\_IER

访问类型: 只写

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	RXSYN	TXSYN	CP1	CP0
7	6	5	4	3	2	1	0
RXBUFF	ENDRX	OVRUN	RXRDY	TXBUFE	ENDTX	TXEMPTY	TXRDY

● TXRDY: 发送就绪中断使能

0: 无效

1: 时钟发送就绪中断

● TXEMPTY: 发送空中断使能

0: 无效

1: 时钟发送空中断

● ENDTX: 发送结束中断使能

0: 无效

1: 时钟发送结束中断

● TXBUFE: 发送缓冲空中断使能

- 0: 无效
- 1: 时钟发送缓冲空中断
- RXRDY: 接收就绪中断使能
- 0: 无效
- 1: 时钟接收就绪中断
- OVRUN: 接收超限中断使能
- 0: 无效
- 1: 使能接收超限中断
- ENDRX: 接收结束中断使能
- 0: 无效
- 1: 时钟接收结束中断
- RXBUFF: 接收缓冲区满中断使能
- 0: 无效
- 1: 使能接收缓冲器满中断
- CP0: Compare0 中断使能
- 0: 无效
- 1: 使能 Compare0 中断
- CP1: Compare1 中断使能
- 0: 无效
- 1: 使能 Compare1 中断
- TXSYN: Tx 同步中断使能
- 0: 无效
- 1: 使能 Tx 同步中断
- RXSYN: RX 同步中断使能
- 0: 无效
- 1: 使能 RX 同步中断

### 32.8.15 SSC 中断禁用寄存器

名称: SSC\_IDR

访问类型: 只写

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	RXSYN	TXSYN	CP1	CP0
7	6	5	4	3	2	1	0
RXBUFF	ENDRX	OVRUN	RXRDY	TXBUFE	ENDTX	TXEMPTY	TXRDY

- TXRDY: 发送就绪中断禁用
- 0: 无效
- 1: 时钟发送就绪中断
- TXEMPTY: 发送空中断禁用
- 0: 无效
- 1: 时钟发送空中断

- **ENDTX**: 发送结束中断禁用
- 0: 无效
- 1: 时钟发送结束中断
- **TXBUFE**: 发送缓冲空中断禁用
- 0: 无效
- 1: 时钟发送缓冲空中断
- **RXRDY**: 接收就绪中断禁用
- 0: 无效
- 1: 时钟接收就绪中断
- **OVRUN**: 接收超限中断禁用
- 0: 无效
- 1: 禁用接收超限中断
- **ENDRX**: 接收结束中断禁用
- 0: 无效
- 1: 时钟接收结束中断
- **RXBUFF**: 接收缓冲区满中断禁用
- 0: 无效
- 1: 禁用接收缓冲器满中断
- **CP0**: Compare0 中断禁用
- 0: 无效
- 1: 禁用 Compare0 中断
- **CP1**: Compare1 中断禁用
- 0: 无效
- 1: 禁用 Compare1 中断
- **TXSYN**: Tx 同步中断禁用
- 0: 无效
- 1: 禁用 Tx 同步中断
- **RXSYN**: RX 同步中断禁用
- 0: 无效
- 1: 禁用 RX 同步中断

### 32.8.16 SSC 中断屏蔽寄存器

名称: SSC\_IMR

访问类型: 只读

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	RXSYN	TXSYN	CP1	CP0
7	6	5	4	3	2	1	0
RXBUF	ENDRX	OVRUN	RXRDY	TXBUFE	ENDTX	TXEMPTY	TXRDY

- **TXRDY**: 发送就绪中断屏蔽
- 0: 无效

- 1: 时钟发送就绪中断
  - TXEMPTY: 发送空中断屏蔽
- 0: 无效
- 1: 时钟发送空中断
  - ENDTX: 发送结束中断屏蔽
- 0: 无效
- 1: 时钟发送结束中断
  - TXBUFE: 发送缓冲空中断屏蔽
- 0: 无效
- 1: 时钟发送缓冲空中断
  - RXRDY: 接收就绪中断屏蔽
- 0: 无效
- 1: 时钟接收就绪中断
  - OVRUN: 接收超限中断屏蔽
- 0: 无效
- 1: 屏蔽接收超限中断
  - ENDRX: 接收结束中断屏蔽
- 0: 无效
- 1: 时钟接收结束中断
  - RXBUFF: 接收缓冲区满中断屏蔽
- 0: 无效
- 1: 屏蔽接收缓冲器满中断
  - CP0: Compare0 中断屏蔽
- 0: 无效
- 1: 屏蔽 Compare0 中断
  - CP1: Compare1 中断屏蔽
- 0: 无效
- 1: 屏蔽 Compare1 中断
  - TXSYN: Tx 同步中断屏蔽
- 0: 无效
- 1: 屏蔽 Tx 同步中断
  - RXSYN: RX 同步中断屏蔽
- 0: 无效
- 1: 屏蔽 RX 同步中断



Powered by Team Mcuzone

QQ:8204136

Website: [www.mcuzone.com](http://www.mcuzone.com)

2008

