

第 49 章 SPI (DSPI)

49.1 简介

注意：关于该模块实例的芯片特有的实现细节，请参见芯片配置章节。

串行设备接口（serial peripheral interface，SPI）模块提供一个在 MCU 和外部设备之间进行通信的同步串行总线。

49.1.1 框图

SPI (DSPI) 的框图如下所示：

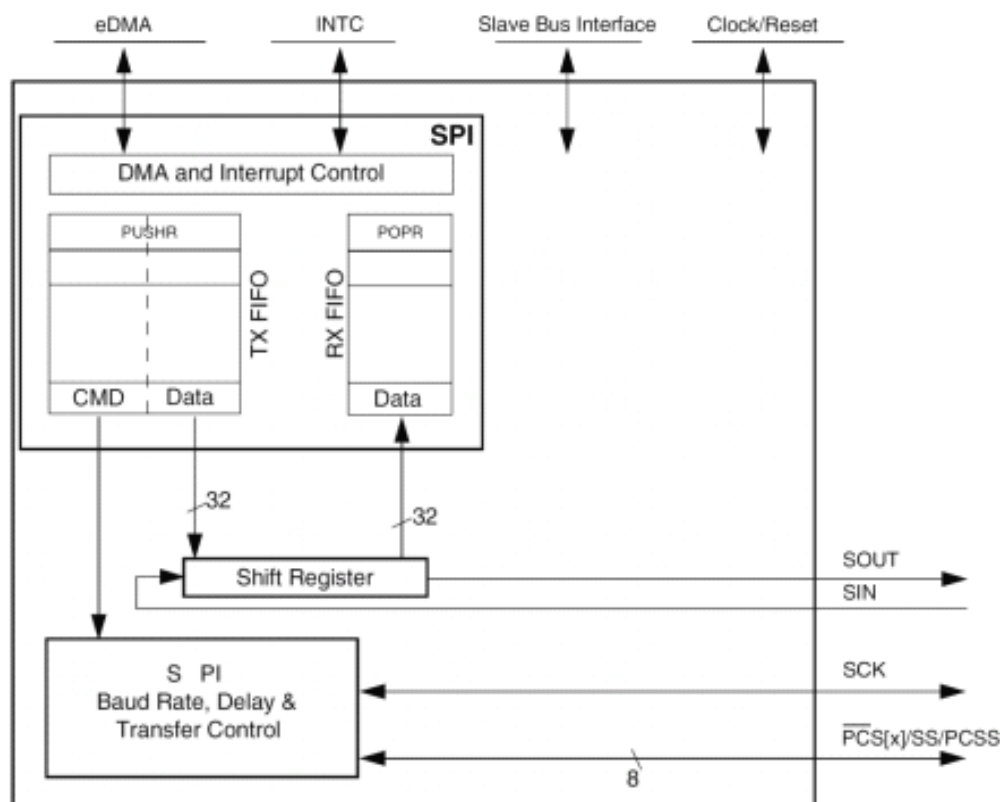


图 49-1 DSPI 框图

49.1.2 特性

DSPI 支持下列 SPI 特性：

- 全双工，四线同步传输
- 主机与从机模式
 - 连续从机选择，使数据流工作在从机模式下
- 使用 4 级 TX FIFO 缓冲进行传输操作
- 使用 4 级 RX FIFO 缓冲进行接收操作
- TX 与 RX 的 FIFO 可以被分别地禁止，低延迟更新到 SPI 队列

-
- 可对每一帧的传输属性进行编程：
 - 2 个传输属性寄存器
 - 可以对串行时钟的极性和相位进行编程
 - 多种可编程的延迟
 - 串行帧长度可被编程为 4 到 16 位，通过软件控制可以扩展
 - 可以连续保持片选
 - 可以用复用器将 6 个外设片选扩展到 64 个
 - 通过复用器稳定地支持多达 32 个设备片选
 - DMA 支持附加到 TX FIFO 的入口并且从 RX FIFO 中移除入口
 - TX FIFO 未滿 (TFFF)
 - RX FIFO 未空 (RFDF)
 - 6 个中断条件：
 - 到达队列结尾 (EQF)
 - TX FIFO 未滿 (TFFF)
 - 当前帧传输完成 (TCF)
 - 在发送 FIFO 为空时试图发送 (TFUF) 数据
 - RX FIFO 未空 (RFDF)
 - 在接收 FIFO 满时接收帧 (RFOF)
 - 全局中断请求线
 - 在与低速外设进行通信时使用优化的 SPI 传输格式
 - 低功耗结构特性
 - 支持停止模式
 - 支持休眠模式

49.1.3 DSPI 配置

DSPI 模块始终工作在 SPI 配置下。

SPI 配置允许 DSPI 发送和接收串行数据。此配置允许 DSPI 工作像基本 SPI 模块一样，使用内部 FIFO，支持外部队列操作。发送数据和接收数据在不同的 FIFO 队列。主机 CPU 或一个 DMA 控制器从接收 FIFO 队列读取接收的数据，并且写发送数据到发送 FIFO 队列。

对于队列操作，SPI 队列可以驻留在系统 RAM，并扩展到 DSPI。队列与 DSPI FIFO 之间的数据传输由 DMA 控制器或主机 CPU 完成。下图中显示了一个系统的样例，其中包含了 DMA、DSPI 和系统 RAM 中的扩展队列。

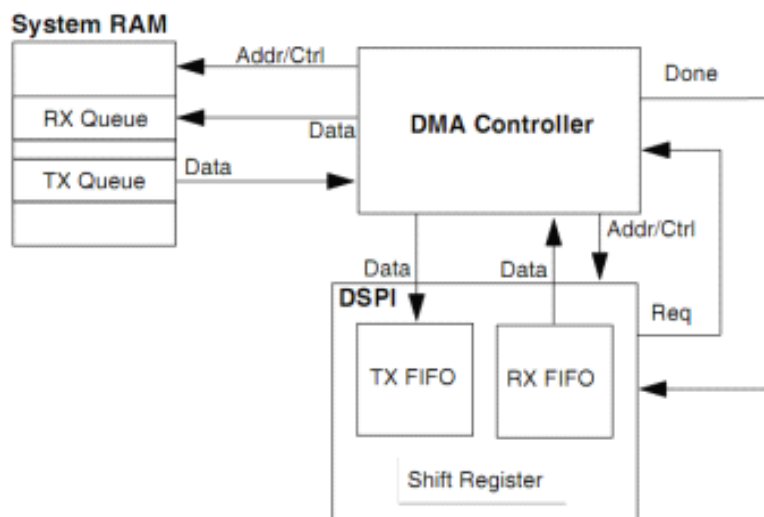


图 49-2 DSPI 与 DMA 和 RAM

49.1.4 工作模式

DSPI 支持下列操作模式，可以被分为两类：

- 模块专用模式：
 - 主机模式
 - 从机模式
 - 模块禁止模式
- MCU 专用模式
 - 外部停止模式
 - 调试模式

当主机写一个 SDPI 寄存器时，DSPI 进入模块专用模式。MCU 专用模式由信号控制，并可扩展到 SDPI。MCU 专用模式就是一个 MCU 可以并行进入 DSPI 模块专用模式的模式。

49.1.4.1 主机模式

主机模式允许 DSPI 初始化并控制串行通信。在此模式下，SCK 信号和 PCS[x] 信号由 DSPI 控制并被配置为输出。

49.1.4.2 从机模式

从机模式允许 DSPI 与 SPI 总线主机进行通信。在此模式下，DSPI 响应外部受控的串行发送器。SCK 信号和 PCS[0]/SS 信号被配置为输入并由一个 SPI 总线主机驱动。

49.1.4.3 模块禁止模式

存储映射逻辑的时钟。

49.1.4.4 外部停止模式

外部停止模式可以用于 MCU 电源管理。DSPI 支持外设总线停止模式机制。当产生进入外部的停止模式的请求时，DSPI 模块识别请求并完成传输过程。当 DSPI 达到帧边界，即意味着 DSPI 模块的系统时钟可以被关闭。

49.1.4.5 调试模式

调试模式用于系统开发与调试。MCR[FRZ]位控制 DSPI 在调试模式下的行为。当 MCU 在调试模式时，若此位被置 1，则 DSPI 停止所有的串行传输。若此位被清 0，则 MCU 调试模式对 DSPI 无效。

49.2 DSPI 信号说明

这部分提供了 DSPI 的信号说明。

下表列举了一些信号，它们可以根据设备定义进行相应地连接。

表 49-1 DSPI 信号描述

信号	描述	IO
PCS0//SS	主机模式：外设芯片选择 0 输出 从机模式：从机选择输入	I/O
PCS1-PCS3	主机模式：外设芯片选择 1-3 从机模式：没使用	O
PCS4	主机模式：外设芯片选择 4 从机模式：没使用	O
PCS4//PCSS	主机模式：外设芯片选择 5，外设芯片选中脉冲	O
SIN	串行数据输入	I
SOUT	串行数据输出	O
SCK	主机模式：串行时钟（输出） 从机模式：串行时钟（输入）	I/O

49.2.1 PCS0//SS——外设片选/从机选择

在主机模式，PCS0 信号是一个外设片选输出，选择当前与哪个从机设备进行传输。

在从机模式，激活的低/SS 信号是从机片选输入信号，允许一个 SPI 主机选择 DSPI 作为目标设备进行传输。

49.2.2 PCS1-PCS3——外设片选 1-3

在主机模式，PCS1-PCS3 是外设片选输出信号。

在从机模式，不使用这些信号。

49.2.3 PCS4——外设片选 4

在从机模式，不使用此信号。

49.2.4 PCS5//PCSS——外设片选 5/外设芯片选择频闪

PCS5 是一个外设片选输出信号。当 DSPI 在主机模式下，并且 MCR[PCSSE] 位被清 0，此信号选择当前与哪个从机进行通信。

当 DSPI 在主机模式下，并且 MCR[PCSSE] 位被置 1，则/PCSS 信号作为外部芯片选择复用器的频闪，它可以对 PCS0–PCS4 信号进行解码，避免了多路复用器输出的毛刺。

在从机模式下，不使用该信号。

49.2.5 SIN——串行输入

SIN 是串行数据输入信号。

49.2.6 SOUT——串行输出

SOUT 是串行数据输出信号。

49.2.7 SCK——串行时钟

SCK 是串行通信时钟信号。在主机模式，DSPI 产生 SCK。在从机模式，SCK 是从外部总线主机得来的一个输入。

49.3 存储映射/寄存器定义

寄存器访问保留的或未定义的存储器地址将导致一个传输错误。写 POPR 寄存器同样产生一个传输错误。

绝对地址	寄存器名	宽度（比特）	访问	复位
4002_C000	DSPI 模块配置寄存器 (SPI0_MCR)	32	R/W	0000_4001
4002_C008	DSPI 传输计数寄存器 (SPI0_TCR)	32	R/W	0000_0000
4002_C00C	DSPI 时钟与传输属性寄存器（主 机模式）(SPI0_CTAR0)	32	R/W	7800_0000
4002_C00C	DSPI 时钟与传输属性寄存器（主 机模式）(SPI0_CTAR0_SLAVE)	32	R/W	7800_0000
4002_C010	DSPI 时钟与传输属性寄存器（主 机模式）(SPI0_CTAR1)	32	R/W	7800_0000
4002_C02C	DSPI 状态寄存器 (SPI0_SR)	32	R/W	0000_0000
4002_C030	DSPI DMA/中断请求选择与使能 寄存器 (SPI0_RSER)	32	R/W	0000_0000
4002_C034	DSPI PUSH 发送队列寄存器（主 机模式）(SPI0_PUSHR)	32	R/W	0000_0000
4002_C034	DSPI PUSH 发送队列寄存器（主 机模式）(SPI0_PUSHR_SLAVE)	32	R/W	0000_0000
4002_C038	DSPI POP 接收队列寄存器 (SPI0_POPR)	32	R	0000_0000
4002_C03C	DSPI 传输队列寄存器 (SPI0_TXFR0)	32	R	0000_0000

4002_C044	DSPI 传输队列寄存器 (SPI0_TXFR2)	32	R	0000_0000
4002_C048	DSPI 传输队列寄存器 (SPI0_TXFR3)	32	R	0000_0000
4002_C07C	DSPI 接收队列寄存器 (SPI0_RXFR0)	32	R	0000_0000
4002_C080	DSPI 接收队列寄存器 (SPI0_RXFR1)	32	R	0000_0000
4002_C084	DSPI 接收队列寄存器 (SPI0_RXFR1)	32	R	0000_0000
4002_C088	DSPI 接收队列寄存器 (SPI0_RXFR3)	32	R	0000_0000
4002_D000	DSPI 模块配置寄存器 (SPI1_MCR)	32	R/W	0000_4001
4002_D008	DSPI 传输计数器 (SPI1_TCR)	32	R/W	0000_0000
4002_D00C	DSPI 时钟与传输属性寄存器 (主 机模式) (SPI1_CTAR0)	32	R/W	7800_0000
4002_D00C	DSPI 时钟与传输属性寄存器 (从 机模式) (SPI1_CTAR0_SLAVE)	32	R/W	7800_0000
4002_D010	DSPI 时钟与传输属性寄存器 (主 机模式) (SPI1_CTAR1)	32	R/W	7800_0000
4002_D02C	DSPI 状态寄存器 (SPI1_SR)	32	R/W	0000_0000
4002_D030	DSPI DMA/中断请求选择与使能 寄存器 (SPI1_RSER)	32	R/W	0000_0000
4002_D034	DSPI PUSH 发送队列寄存器 (主 机模式) (SPI1_PUSHR)	32	R/W	0000_0000
4002_D034	DSPI PUSH 发送队列寄存器 (从 机模式) (SPI1_PUSHR_SLAVE)	32	R/W	0000_0000
4002_D038	DSPI POP 接收队列寄存器 (SPI1_POPR)	32	R	0000_0000
4002_D03C	DSPI 传输队列寄存器 (SPI1_TXFR0)	32	R	0000_0000
4002_D040	DSPI 传输队列寄存器 (SPI1_TXFR1)	32	R	0000_0000
4002_D044	DSPI 传输队列寄存器 (SPI1_TXFR2)	32	R	0000_0000
4002_D048	DSPI 传输队列寄存器 (SPI1_TXFR3)	32	R	0000_0000
4002_D07C	DSPI 接收队列寄存器 (SPI1_RXFR0)	32	R	0000_0000
4002_D080	DSPI 接收队列寄存器 (SPI1_RXFR1)	32	R	0000_0000
4002_D084	DSPI 接收队列寄存器 (SPI1_RXFR1)	32	R	0000_0000
4002_D088	DSPI 接收队列寄存器 (SPI1_RXFR3)	32	R	0000_0000
400A_C000	DSPI 模块配置寄存器 (SPI2_MCR)	32	R/W	0000_4001
400A_C008	DSPI 传输计数器 (SPI2_TCR)	32	R/W	0000_0000
400A_C00C	DSPI 时钟与传输属性寄存器 (从 机模式) (SPI2_CTAR0)	32	R/W	7800_0000
400A_C00C	DSPI 时钟与传输属性寄存器 (从 机模式) (SPI2_CTAR0_SLAVE)	32	R/W	7800_0000
400A_C010	DSPI 时钟与传输属性寄存器 (主 机模式) (SPI2_CTAR1)	32	R/W	7800_0000

400A_C030	DSPI DMA/中断请求选择与使能寄存器 (SPI2_RSER)	32	R/W	0000_0000
400A_C034	DSPI PUSH 发送队列寄存器 (主机模式) (SPI2_PUSHR)	32	R/W	0000_0000
400A_C034	DSPI PUSH 发送队列寄存器 (从机模式) (SPI2_PUSHR_SLAVE)	32	R/W	0000_0000
400A_C038	DSPI POP 接收队列寄存器 (SPI2_POPR)	32	R	0000_0000
400A_C03C	DSPI 传输队列寄存器 (SPI2_TXFR0)	32	R	0000_0000
400A_C040	DSPI 传输队列寄存器 (SPI2_TXFR1)	32	R	0000_0000
400A_C044	DSPI 传输队列寄存器 (SPI2_TXFR2)	32	R	0000_0000
400A_C048	DSPI 传输队列寄存器 (SPI2_TXFR3)	32	R	0000_0000
400A_C07C	DSPI 接收队列寄存器 (SPI2_RXFR0)	32	R	0000_0000
400A_C080	DSPI 接收队列寄存器 (SPI2_RXFR1)	32	R	0000_0000
400A_C084	DSPI 接收队列寄存器 (SPI2_RXFR1)	32	R	0000_0000
400A_C088	DSPI 接收队列寄存器 (SPI2_RXFR3)	32	R	0000_0000

49.3.1 DSPI 模块配置寄存器 (SPIx_MCR)

包含用于配置与 DSPI 操作相关的各种属性的位。HALT 位与 MDIS 位可以在任何时候被改变，但是他们只在以下帧的开始起作用。当 DSPI 处于运行状态时，MCR 寄存器中只有 HALT 位及 MDIS 位可以改变。

地址：SPI0_MCR-4002_C000h (基址) +0h (偏移地址) =4002_C000h

SPI1_MCR-4002_D000h (基址) +0h (偏移地址) =4002_D000h

SPI2_MCR-400A_C000h (基址) +0h (偏移地址) =400A_C000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
读	MSTR	CONT_SCKE	DCONF		FRZ	MTFE	PCSSE		0		PCISIS[5:0]					
写																
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
读	DOZE	MDIS	SIA_TXF	DIS_RXF	0	0	SMPL_PT								0	HALT
写					无	无									无	

复位	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
----	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

SPIx_MCR 字段描述

字段	说明
31 (MSTR)	主从机模式选择位。为主机模式或者从机模式配置 DSPI 模块。0 DSPI 为从机模式；1 DSPI 为主机模式；
30 (CONT_SCKE)	连续 SCK 使能。使能串行时钟 (SCK) 使其继续运行。0 禁止连续 SCK 时钟；1 开启连续 SCK 时钟；
29-28 (DCONF)	DSPI 配置。选择 DSPI 下列配置。00 SPI；01 预留；10 预留；11 预留；
27 (FRZ)	睡眠。当设备进入到调试模式时，开启 DSPI 使其在传输完下一个帧时停止。0 在调试模式下不中止串行传输；0 在调试模式下中止串行传输。
26 (MTFE)	使能优化的时序格式。使能优化的传输模式。0 禁止 SPI 优化的传输格式；0 使能 SPI 优化的传输格式。
25 (PCSSE)	使能外设芯片选择频闪。使能 PCS[5]/PCSS 信号用于当作 PCS 频闪输出信号使用。0 PCS[5]/PCSS 用于外设芯片选择[5]信号；1 PCS[5]/PCSS 用于低有效 PCS 频闪信号。
24 (ROOE)	开启接收队列 FIFO 溢出覆盖。在 Rx FIFO 溢出的情况下，配置 DSPI 来忽略输入的数据或者是覆盖已存在的数据。如果 RX FIFO 队列已经满，并且有新的数据达到，产生溢出的来自收发器的数据将会被忽略或者被移入到移位寄存器。0 输入数据被忽略；1 输入数据被移入到移位寄存器
23-22 (预留)	只读位，读出总为 0。
21-16 (PCSSIS[5:0])	外设芯片选择信号 x 非激活状态。决定 PCSx 信号的非激活状态。0 PCSx 信号的非激活状态为低；1 PCSx 信号的非激活状态为高
15 (DOZE)	睡眠模式。对外部控制睡眠模式的低功耗机制提供支持。0 睡眠模式对 DSPI 没有影响；1 睡眠模式禁止 DSPI。
14 (MDIS)	禁止模块。在 DSPI 中，对无内存映射逻辑允许禁止时钟来置 DSPI 为软件控制的低功耗状态。MDIS 的复位值有参数决定，默认值为 0。0 使能 DSPI 时钟；1 允许外部逻辑禁止 DSPI 时钟。
13 (DIS_TXF)	禁止传输队列 FIFO。当发送队列 FIFO 被禁止之后，那么 DSPI 将会工作在简单的双缓冲 SPI 模式。0 开启 Tx FIFO；1 禁止 Tx FIFO；
12 (DIS_RXT)	禁止接收队列 FIFO。当接收队列 FIFO 被禁止之后，那么 DSPI 将会工作在简单的双缓冲 SPI 模式。0 开启 Rx FIFO；1 禁止 Rx FIFO。
11 (CLR_TXF)	清除发送队列 FIFO。清空发送队列 FIFO，向该位写 1 将清除发送队列 FIFO 计数器。该位读出总为 0。0 不清除发送队列 FIFO 计数器；1 清除发送队列 FIFO 计数器；
10 (CLR_PT)	清除接收队列 FIFO。清空接收队列 FIFO，向该位写 1 将清除接收队列 FIFO 计数器。该位读出总为 0。0 不清除接收队列 FIFO 计数器；1 清除接收队列 FIFO 计数器；
9-8 (SMPL_PT)	采样点。在可调节传输模式中控制 DSPI 何时采样 SIN 信号。00 在 SCK 边界与 SIN0 系统时钟采样；01 在 SCK 边界与 SIN1 系统时钟采样；10 在 SCK 边界与 SIN2 系统时钟采样；11 预留。
7-1 (预留)	只读位，读出总为 0。
0 (HALT)	中止传输控制位。开始或停止 DSPI 传送 0 开始传输；1 停止传输。

49.3.2 DSPI 传输计数寄存器 (SPIx_TCR)

用包含一个计数器的 TCR 寄存器来记录 SPI 所做传输的次数。传输计数器旨在帮助管理队列。当 DPI 处于运行状态时，不可写 TCR 寄存器。

地址：SPI0_TCR - 4002_C000h base + 8h offset = 4002_C008h

SPI1_TCR - 4002_D000h base + 8h offset = 4002_D008h

写																
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SPIx_CTARn 字段描述

字段	说明																																								
31 (DBR)	双重波特率。双有效串行通信时钟 (SCK) 波特率。该位只在主机模式下有用。它有效地半分波特率分频比, 支持更快的频率, 以及对串行通信时钟 (SCK) 的奇分频。当 DBR 置位时, 串行通信时钟 (SCK) 的占空比决定于波特率预分频器及时钟相位 (如下表)。关于如何计算波特率可以参见 BR 字段的描述。																																								
	表 49-32 DSPI SCK 占空比																																								
	<table><tr><th>DBR</th><th>CPHA</th><th>PBR</th><th>占空比</th></tr><tr><td>0</td><td>任意</td><td>任意</td><td>50/50</td></tr><tr><td>1</td><td>0</td><td>00</td><td>50/50</td></tr><tr><td>1</td><td>0</td><td>01</td><td>33/66</td></tr><tr><td>1</td><td>0</td><td>10</td><td>40/60</td></tr><tr><td>1</td><td>0</td><td>11</td><td>43/57</td></tr><tr><td>1</td><td>1</td><td>00</td><td>50/50</td></tr><tr><td>1</td><td>1</td><td>01</td><td>66/33</td></tr><tr><td>1</td><td>1</td><td>10</td><td>60/40</td></tr><tr><td>1</td><td>1</td><td>11</td><td>57/43</td></tr></table>	DBR	CPHA	PBR	占空比	0	任意	任意	50/50	1	0	00	50/50	1	0	01	33/66	1	0	10	40/60	1	0	11	43/57	1	1	00	50/50	1	1	01	66/33	1	1	10	60/40	1	1	11	57/43
	DBR	CPHA	PBR	占空比																																					
	0	任意	任意	50/50																																					
	1	0	00	50/50																																					
	1	0	01	33/66																																					
	1	0	10	40/60																																					
	1	0	11	43/57																																					
	1	1	00	50/50																																					
	1	1	01	66/33																																					
	1	1	10	60/40																																					
1	1	11	57/43																																						
0 波特率按照平常的 50/50 占空比计算; 1 波特率加倍, 波特率预分频器决定占空比。																																									
30-27 (FMSZ)	帧大小。每个帧被传输的比特数等于 FMSE 字段的值加 1, FMSZ 字段最小的有效值为 3。																																								
26 (CPOL)	时钟极性。选择串行通信时钟 (SCK) 的非激活状态。该位同时用于主机与从机。对于串行设备之间的成功通信, 设备必须有唯一的时钟极性。当选择了连续选择格式时, 在没有停止 DSPI 模块而在时钟极性之间进行切换时, 会导致在传输出现错误, 因为外设会把时钟极性的切换当成一个有效。0 SCK 无效状态为低; 1 SCK 无效状态为高。																																								
25 (CPHA)	时钟相位。用来选择在那个 SCK 时钟边缘来改变数据以及在那个时钟边沿来捕获数据。该位同时用于主机模式及从机模式。对于串行设备之间的成功传输, 设备必须具有唯一的时钟相位。在继续 SCK 模式下, 如果 CPHA 的值为 1, 那么该位会被忽略并且会按照 CPHA 为 1 来传输数据。0 数据在 SCK 上升沿上线, 在下降沿取数; 1 数据在 SCK 下降沿上线, 在上升沿取数。																																								
24 (LSBFE)	低字节先传输。指示在传输 SPI 数据帧时有限传输 LSB 还是 MSB。0 MSB 位先传; 1LSB 位先传。																																								
23-22 (PCSSCK)	PCS 到 SCK 的延迟预分频器。选择在 PCS 置位与 SCK 第一个边沿之间的延迟的预分频器。如何计算 PCS 到 SCK 之间的延迟参见 CSSCK 的描述。00 PCS 到 SCK 的预分频系数为 1; 01 PCS 到 SCK 的预分频系数为 3; 10 PCS 到 SCK 的预分频系数为 5; 11PCS 到 SCK 的预分频系数为 7。																																								
21-20 (PASC)	SCK 之后的延迟预分频系数。用来选择 SCK 最后一个边沿到 PCS 被清 0 之间的延迟时间的预分频系数。如何计算 SCK 之后的延迟参见 ASC 字段的详细描述。00 在传输之后的延迟预分频器系数为 1; 01 在传输之后的延迟预分频器系数为 3; 10 在传输之后的延迟预分频器系数为 5; 11 在传输之后的延迟预分频器系数为 7。																																								
19-18 (PDT)	传输时候的延迟预分频器系数。选择在帧的最后一位 PCS 信号清零到下一个帧的开始 PCS 置位之间的延迟的预分频系数值。PDT 只用于主机模式。关于如何计算传输之后的延迟参见 DT 字段的详细描述。00 传输之后的延迟预分频器的值为 1; 00 传输之后的延迟预分频器的值为 3; 00 传输之后的延迟预分频器的值为 5; 00 传输之后的延迟预分频器的值为 7。																																								

17-16 (PBR)	波特率预分频器。选择波特率的预分频器系数。在字段只在主机模式下有用。波特率为 SCK 的频率。在波特率选择开始时之前用系统时钟除以该值。关于波特率的计算参见 BR 字段的详细描述。00 波特率预分频值为 2；00 波特率预分频值为 3；00 波特率预分频值为 5；00 波特率预分频值为 7。																																		
15-12 (CSSCK)	<p>PCS 到 SCK 的延迟分频。选择 PCS 到 SCK 之间延迟的预分频参数。该字段只用于主机模式。PCS 到 SCK 的延迟为 PCS 断言到 SCK 的第一个边沿的延迟。该延迟为系统时钟周期的倍数，可以通过以下的公式来计算。$t_{CSC} = (1/f_{SYSN}) \times PCSSCK \times CSSCK$。</p> <table border="1"> <caption>表 49-33 延迟参数编码</caption> <thead> <tr> <th>字段值</th><th>延迟分频参数</th></tr> </thead> <tbody> <tr><td>0000</td><td>2</td></tr> <tr><td>0001</td><td>4</td></tr> <tr><td>0010</td><td>8</td></tr> <tr><td>0011</td><td>16</td></tr> <tr><td>0100</td><td>32</td></tr> <tr><td>0101</td><td>64</td></tr> <tr><td>0110</td><td>128</td></tr> <tr><td>0111</td><td>256</td></tr> <tr><td>1000</td><td>512</td></tr> <tr><td>1001</td><td>1024</td></tr> <tr><td>1010</td><td>2048</td></tr> <tr><td>1011</td><td>4096</td></tr> <tr><td>1100</td><td>8192</td></tr> <tr><td>1101</td><td>16384</td></tr> <tr><td>1110</td><td>32768</td></tr> <tr><td>1111</td><td>65536</td></tr> </tbody> </table>	字段值	延迟分频参数	0000	2	0001	4	0010	8	0011	16	0100	32	0101	64	0110	128	0111	256	1000	512	1001	1024	1010	2048	1011	4096	1100	8192	1101	16384	1110	32768	1111	65536
字段值	延迟分频参数																																		
0000	2																																		
0001	4																																		
0010	8																																		
0011	16																																		
0100	32																																		
0101	64																																		
0110	128																																		
0111	256																																		
1000	512																																		
1001	1024																																		
1010	2048																																		
1011	4096																																		
1100	8192																																		
1101	16384																																		
1110	32768																																		
1111	65536																																		
11-8 (ASC)	SCK 之后延迟参数。选择 SCK 之后的分频参数。该模式只用于主机模式。SCK 之后的延迟是指在 SCK 最后一个边沿到 PCS 无效之间的延迟。该延迟是系统时钟周期的倍数，并且可以通过下面的工作来计算。 $t_{ASC} = (1/f_{SYS}) \times PASC \times ASC$ 。可以参见延迟参数表在 CTARN[CSSCK] 字段位字段的描述。																																		
7-4 (DT)	<p>传输之后的延迟分频参数。用来选择传输之后的延迟分频参数。该位只用于主机模式，该延迟是在帧的最后 PCS 信号无效到下一个帧开始时 PCS 信号断言之间的时间。在连续串行通信时候总操作中，DT 的值固定为 SCK 的时钟周期。传输之后的延迟是系统时钟周期的倍数，可以通过以下的公式来计算。$t_{DT} = (f_{SYS}/PBR) [(1+DBR)/BR]$。</p> <p>在 CTARN[CSSCK] 位字段描述见延迟分频编码表。</p>																																		

3-0 BR	波特率分频系数。选择波特率分频器的值。该位只有在主模式下有效。预分频的系统时钟除以波特率分频系数来产生 SCK 频率。波特率根据以下方程计算：SCK baud rate = (fSYS/PBR) x [(1+DBR)/BR] 下表列出了波特率分频系数															
	表 49-34 DSPI 波特率分频参数															
	CTARn[BR]								波特率分频参数值							
	0000								2							
	0001								4							
	0010								6							
	0011								8							
	0100								16							
	0101								32							
	0110								64							
	0111								128							
	1000								256							
	1001								512							
	1010								1024							
	1011								2048							
	1100								4096							
	1101								8192							
	1110								16384							
	1111								32768							

49.3.4 DSPI 时钟与传输属性寄存器（从机模式下） (SPIx_CTARn_SLAVE)

当 DSPI 被配置为 SPI 总线从机时，CTAR0 被使用。

地址：SPI0_CTAR0_SLAVE - 4002_C000h base + Ch offset = 4002_C00Ch

SPI1_CTAR0_SLAVE - 4002_D000h base + Ch offset = 4002_D00Ch

SPI2_CTAR0_SLAVE - 400A_C000h base + Ch offset = 400A_C00Ch

Bit	31	30	29	28	27	26	25	24	23	22-0						
读	FMSZ					CPOL	CPHA	无效		0						
写										无效						
复位	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0

SPIx_CTARn_SLAVE 字段描述

字段	说明
31-27 (FMSZ)	帧大小。每一个被传输的帧号等于 FMSZ 字段的值加 1，最小的有效值为 3
26 (CPOL)	时钟极性。选择串行通信时钟 (SCK) 的非激活状态。0 SCK 无效状态为低；1 SCK 无效状态为高。
25 (CPHA)	时钟相位。用来选择在那个 SCK 时钟边缘来改变数据以及在那个时钟边沿来捕获数据。该位同时用于主机模式及从机模式。对于串行设备之间的成功传输，设备必须具有唯一的时钟相位。在继续 SCK 模式下，如果 CPHA 的值为 1，那么该位会被忽略并且会按照 CPHA 为 1 来传输数据。0 数据在 SCK 上升沿捕捉，在下降沿改变；1 数据在 SCK 下降沿捕捉，在上升沿改变。
24-0	只读位。该位读出总为 0。

49.3.5 DSPI 状态寄存器 (SPIx_SR)

SR 包含状态与标志位。每一个比特用来表示 DSPI 的状态，以指示可以产生中断或者 DMA 请求的事件的发生。软件可以通过向 SR 中的标志位写 1 来清除标志，写 0 无影响。该寄存器在禁止模式下不能被写入主要是为了低功耗机制。

地址：SPI0_SR - 4002_C000h base + 2Ch offset = 4002_C02Ch

SPI1_SR - 4002_D000h base + 2Ch offset = 4002_D02Ch

SPI2_SR - 400A_C000h base + 2Ch offset = 400A_C02Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
读	TCF	TXRX	0	EOQF	TFUF	0	TFFF	0					RFOF	0	RFDF	0
写	W1C	W1C	无效	W1C	W1C	无效	W1C	无效					W1C	无效	W1C	无效
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
读	TXCTR				TXNXTPTR				RXCTR				POPNXTPTR			
写	无效															
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SPIx_SR 字段描述

字段	说明
31 (TCF)	传输完成标志位。指示一个帧的所有比特都已经被移出。TCF 会一直置位直到其被写 1 清 0。0 传输没有完成；1 传输完成。
30 (TXRXS)	发送与接收状态标志位。反映 DSPI 的运行状态。0 传输与接收操作被禁止 (DSPI 处于停止状态)；1 传输与接收操作被开启 (DSPI 处于运行状态)
29 (预留)	该位只读，并且读出总为 0。
28 (EOQF)	队列尾标志。该位用来表示在队列中的最后一个数据已经被传输完成 (当 DSPI 处于主机模式时)。当发送队列 FIFO 数据使 EOQ 置位 (在命令半字) 及在传输到达最后时，该位 EOQF 被置位。EOQF 位会一直置位直到通过向该位写 1 清除。当 EOQF 为置位时，TXRXS 状态为会自动被清 0。0 在可执行命令字段 EOQ 没有被置位；在执行 SPI 命令中 EOQ 被置位。
28 (TFUF)	传输队列 FIFO 下溢标志位。指示发送队列下溢。只有当 DSPI 时钟处于从机模式并且 SPI 配置时可以检测到下溢条件。当处于从机模式的 DSPI 的发送队列为空并且一个外部 SPI 主机初始化一个传输时 TFUF 位置位。TFUF 会一直置位直到通过向该位写 1 来清除。0 没有发送队列 FIFO 下溢；1 发生了发送队列的下溢。
26 (预留)	只读位，读出总为 0。
25 (TFFF)	传输队列 FIFO 满标志位。为 DSPI 提供了一个方法来请求更多的实体来增加到发送队列 FIFO。当发送队列 FIFO 不满时，TFFF 位置位。通过向该位写 1 或者得到 DMA 控制器的对发送队列 FIFO 满请求的确认使该位被清 0。0 发送队列 FIFO 满；1 发送队列 FIFO 不满。
24-20 (预留)	只读位。该位读出总为 0。
	接收队列溢出标志。用来表示接收队列发生溢出。当接收队列 FIFO 以及移位寄存

	志。0 没有发生接收队列 FIFO 溢出；1 发生接收队列 FIFO 溢出。
18（预留）	只读位。该位读出总为 0。
17（RFDF）	接收队列 FIFO 损耗标志。为 DSPI 提供了一个方法用来请求移除接收队列 FIFO 的实体。当接收队列不空时该位被置位。可以向该位写 1 或者受到 DMA 控制器当接收队列为空时的确认时，该位被清 0。0 接收队列为空；1 接收队列不为空。
16（预留）	只读位。该位读出总为 0。
15-12（TXCTR）	发送队列计数器。用来指示发送队列中有效的实体的个数。TXCTR 每当 PUSH_R 位被写时该位自动增加。每当 SPI 命令被执行以及 SPI 数据被传输到移位寄存器时，TXCTR 自动减 1。
11-8（TXNXTPTR）	传输下一个指针。指示下次要传输的发送队列实体。每当 SPI 数据被从发送队列移入到移位寄存器时该字段的值自动更新。
7-4（RXCTR）	接收队列计数器。表示接收队列中实体的个数。每当 POP_R 为被读取时该字段的值自动减 1，每当数据从移位寄存器移入到接收队列时，RXCTR 自动增加 1。
3-0（POP_NXTPTR）	弹出下一个指针。包含一个用于返回的接收队列实体的指针，当 POP_R 被读取时。POP_NXTPTR 每当 POP_R 被读取时会自动更新。

49.3.6 DSPI DMA/中断请求选择及使能寄存器（SPIx_RSER）

RSER 控制 DMA 及中断请求。当 DSPI 处于运行状态时不要写 RSER 寄存器。

地址：SPI0_RSER - 4002_C000h base + 30h offset = 4002_C030h

SPI1_RSER - 4002_D000h base + 30h offset = 4002_D030h

SPI2_RSER - 400A_C000h base + 30h offset = 400A_C030h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
读	TCF_RN	0		EOQF_RN	TFUF_RE	0	TFFF_RE	TFFF_DIRS	0				RFOF_RE	0	RFDF_RE	RFDF_DIRS
写		无效				无效			无效							
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15-0															
读	0															
写	无效															
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SPIx_RSER 字段描述

字段	说明
31（TCF_RE）	使能传输完成请求。使能 SR 寄存器 TCF 标志产生一个中断请求。0 禁止 TCF 中断请求；1 开启 TCF 中断请求。
30-29（预留）	只读位，该位读出总为 0。
28（EOQF_RE）	开启 DSPI 完成请求。使能 SR 寄存器的 EOQF 标志产生一个中断。0 禁止 EOQF 中断请求；1 开启 WOQF 中断请求。
27（TFUF_RE）	开启传输队列下溢请求。使能 SR 寄存器的 TFUF 标志产生一个中断。0 禁止 TFUF 中断请求；1 开启 TFUF 中断请求。

25 (TFFF_RE)	使能传输队列满请求。使能 SR 寄存器的 TFFF 标志产生一个中断，TFFF_DIRS 用来选择是产生中断请求还是 DMA 请求。0 禁止 TFFF 中断或者 DMA 中断；0 开启 TFFF 中断或者 DMA 中断；
24 (TFFF_DIRS)	传输队列满 DMA 或者中断请求选择位。选择是产生 DMA 请求还是中断请求。当 SR 寄存器的 TFFF 位置位，并且 RSER 寄存器的 TFFF_RE 置位，那么该位用来选择是产生 DMA 请求还是中断请求。0 TFFF 标志产生中断请求，1 TFFF 标志产生 DMA 请求
23-20 (预留)	只读位，读出总为 0
19 (RFOF_RE)	接收队列溢出请求使能位。使能 SR 寄存器的 RFOF 标志产生一个中断。0 禁止 RFOF 中断请求；1 开启 RFOF 中断请求。
18 (预留)	只读位，读出总为 0
17 (RFDF_RE)	接收队列损耗请求使能位。使能 SR 寄存器的 RFDF 标志位产生一个中断。RFDF_DISR 位用来选择是产生中断请求还是 DMA 请求。0 禁止 RFDF 中断或者 DMA 请求；1 开启 RFDF 中断或者 DMA 请求。
16 (RFDF_DIRS)	接收队列损耗 DMA 请求或者中断请求选择位。选择是产生中断请求还是 DMA 请求。当 SR 寄存器的 RFDF 位置位及 RSER 寄存器的 RFDF_RE 位置位时，RFDF_DIRS 用来选择是产生中断请求还是 DMA 请求。0 中断请求；1 DMA 请求。
15-0 (预留)	只读位，读出总为 0

49.3.7 主机模式 DSPI PUSH 发送队列寄存器 (SPIx_PUSHR)

PUSH 提供了一种向发送队列写的方式。写入到这个寄存器的数据被传输到发送队列。8 比特或者 16 比特的写放回 PUSHR 将会传输所有的 32 个寄存器比特到发送队列。该寄存器的结构在主机和从机模式下是不同的。在主机模式下，该寄存器提供了 16 比特的命令及 16 比特的数据到队列。在从机模式下，所有的 32 个比特都都可以被用作数据，最大支持 32 比特的 SPI 帧操作。

地址：SPI0_PUSHR - 4002_C000h base + 34h offset = 4002_C034h

SPI1_PUSHR - 4002_D000h base + 34h offset = 4002_D034h

SPI2_PUSHR - 400A_C000h base + 34h offset = 400A_C034h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
读	CONT	CTAS				EOQ	CTCNT	0				PCS[5:0]					
写								无效									
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

Bit	15-0															
读	TXDATA															
写																
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

31 (CONT)	连续外设芯片选择使能位。选择一个连续选择格式。该比特只用于 SPI 主机模式，该位使能选择 PCS 信号在传输期间一直置位。0 在传输期间 PCSn 返回到他们的非激活状态；1 在传输期间 PCSn 信号保持置位。
30-28 (CTAS)	时钟与传输属性选择位。在主机模式下该字段用来选择那些 CTAR 寄存器用于指示相关的 SPI 帧的传输属性。在 SPI 从机模式下，使用 CTAR0。参看芯片配置章节来决定设备具有多少个 CTAR 寄存器。不应该对该寄存器编程一个当前不存在的寄存器。000 CTAR0；001 CTAR1；010 预留；011 预留；100 预留；101 预留；110 预留；111 预留；
27 (EOQ)	队列尾标志。主机软件使用该位向 DSPI 发送信号，指示当前的 SPI 传输是队列的最后一个。在传输结束。SR 寄存器的 EOQF 位置位。0 要传输的 SPI 数据不是最后的；1 要传输的 SPI 数据是最后的。
26 (CTCNT)	清除传送计数器。清除 TCR 寄存器的 SPI_TCNT 字段。在 DSPI 开始传输当前数据帧之前，SPI_TCNT 字段被清除。
25-22 (预留)	只读位，该位读出总为 0。
21-16 (PCS[5:0])	选择那个 PCS 信号被置位以用来进行传输。参考芯片配置章节来了解在 MCU 中所使用的 PCS 信号个数。0 清 0PCS[x]信号；0 置位 PCS[x]信号；
15-0 (TXDATA)	发送数据保持要被传输的 SPI 数据，通过相应的 SPI 命令。

49.3.8 从机模式 DSPI PUSH 发送队列寄存器 (SPIx_PUSHR_SLAVE)

PUSHR 为一种向发送队列写入数据提供了方法。被写入到该寄存器的数据将会被传输到发送队列。8 比特或者 16 比特的写访问 PUSHR 传输所有的 32 寄存器比特到发送队列。该寄存器的结构在主机模式与从机模式下是不同的。在主机模式下，该寄存器提供了 16 比特的命令及 16 比特的数据到队列。在从机模式下，所有的 32 个比特都都可以被用作数据，最大支持 32 比特的 SPI 帧操作。

地址：SPI0_PUSHR_SLAVE - 4002_C000h base + 34h offset = 4002_C034h

SPI1_PUSHR_SLAVE - 4002_D000h base + 34h offset = 4002_D034h

SPI2_PUSHR_SLAVE - 400A_C000h base + 34h offset = 400A_C034h

Bit	31-0															
读	TXDATA															
写																
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SPIx_PUSHR_SLAVE 字段描述

字段	说明
31-0 (TXDATA)	发送数据。保持要被传输的 SPI 数据，通过相应的 SPI 命令。

49.3.9 DSPI POP 接收队列寄存器 (SPIx_POPR)

POPR 用于读取接收队列。8 比特或者 16 比特读访问 POPR 将会在接收队列产生和 32 位读访问相同的影响。写这个寄存器会产生一个传送错误。

地址：SPI0_POPR - 4002_C000h base + 38h offset = 4002_C038h

SPI2_POPR - 400A_C000h base + 38h offset = 400A_C038h																
Bit	31-0															
读	RXDATA															
写																
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SPIx_POPR 字段描述

字段	说明
31-0 (RXDATA)	接收数据，包含来自接收队列的 SPI 数据。该队列实体将会是推出的下一个数据指针指向的位置。

49.3.10 DSPI 传输队列寄存器 (SPIx_TXFRn)

TXFRn 提供了一个用于调试目的的对发送队列的可视性。每一个寄存器都是发送队列的一个整体。该寄存器只读并且不能被更改。读 TXFRx 寄存器不会改变发送队列的状态。

地址：SPI0_TXFR0 - 4002_C000h base + 3Ch offset = 4002_C03Ch																
Bit	31-16								15-0							
读	TXCMD_TXDATA								TXDATA							
写																
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SPIx_TXFRn 字段描述

字段	说明
31-16 (TXCMD_TXDATA)	传输命令或者传输数据。在主机模式下，TXCMD 字段包含有设置 SPI 数据传输数据的命令。在从机模式下 TXDATA 包含有要移位出去的 16 个 MSB 位
15-0 (TXDATA)	传输数据。包含有要被移出的 SPI 数据。

49.3.11 DSPI 接收队列寄存器 (SPIx_RXFRn)

RXFRn 提供对队列调试的可视性。每一个寄存器都是一个接收队列的实体。RXFR 寄存器只读。读 RXFRx 寄存器不会改变接收队列的状态。

地址：SPI0_RXFR0 - 4002_C000h base + 7Ch offset = 4002_C07Ch																
Bit	31-0															
读	RXDATA															
写	无效															
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SPIx_RXFRn 字段描述

49.4 功能描述

串行外设接口 (DSPI) 支持在 MCU 和外设之间的全双工、同步串行通信的方式。所有的通信都是工作在 SPI 协议。DSPI 具有以下配置：

1. SPI 可以配置为处于基本 SPI 或者队列 SPI 模式；

DSPI 模块配置寄存器 (MCR) 的 DCONF 字段用来决定 DSPI 的配置。参见 DSPI 配置值。

CTARn 寄存器具有时钟和传输属性。SPI 配置允许选择在帧中使用哪一个 CTAR 通过设置 SPI 命令的字段。参看 DSPI 时钟与传输属性寄存器来获得详细的信息。

典型的主从之间的连接如下图所示。但执行一个数据传输操作时，数据会被串行地移入到一个预设的位置。因为模块是连接的，因此数据就可以在主机与从机之间进行交换。在主机移位寄存器的数据现在被移到了从机的移位寄存器，反之亦然。在传输的结束，SR 寄存器的 TCF 位被置位，以表示一个传输的完成。

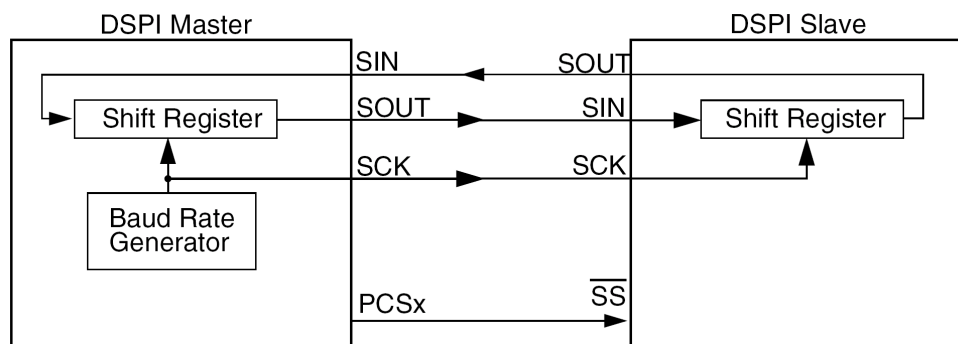


图 49-91 SPI 串行协议框图

通常情况下，会有多余一个的从机设备连接到 DSPI 主机。DSPI 主机的 6 个 PCS 信号可以被用于选择到底是和哪个从机通信。

三个 DSPI 配置共享传输协议及时序，这部分在传输格式中单独介绍。传输速率及延迟设定在“DSPI 波特率及时钟延迟产生”章节有详细描述。

49.4.1 DSPI 传输的开始与停止

DSPI 具有两个操作状态：STOPPED 及 RUNNING。这些状态是独立于 DSPI 的配置的。DSPI 的默认状态是 STOPPED。在 STOPPED 状态下，在主机模式下没有串行传输被启动并且在从机模式下没有传输的回复。STOPPED 状态对于写 DSPI 的各种配置寄存器也是一种安全的状态，不会导致不可预知的后果。在 RUNNING 状态下发送串行传输。

状态，那么该位被置位。

当所有的下列的条件都为真时，DSPI 开始（DSPI 转换到 RUNNING 状态）：

SR[EOQF]位清 0；MCU 不处于调试模式或者 MCR[FRZ]位清零；MCR[HALT] 位被清零。

当下列任一条件发生时，在当前帧之后 DSPI 停止（从 RUNNING 到 STOPPED 的转移）

SR[EOQF]位被置位；MCU 在调试模式且 MCR[FRZ]位置位；MCR[HALT]位置位

串行传输从 RUNNING 转变到 STOPPED 状态发生在下一个帧的边界如果传输正在进行，或者如果没有传输进行那么立即计入到 STOPPED 状态。

49.4.2 串行外设接口配置

SPI 配置串行地传输数据通过使用一个移位寄存器及可编程的传输属性的选择。当 MCR 寄存器的 DCONF 字段为 0b00 时，DSPI 处于 SPI 模式。SPI 帧可以达到 32 比特长。主机 CPU 或者 DMA 控制器传输来自于 DSPI 的 RAM 队列的数据到传输队列缓冲区（TX FIFO）。接收到的数据存储在接收队列缓冲区（RX FIFO）的实体内。主机 CPU 或者 DMA 控制器传输来自于接收队列到外部内存的数据到 DSPI。队列缓冲区的操作描述详见“传输队列缓冲区机制”以及“接收队列缓冲区机制”。中断及 DMA 请求条件在“中断/DMA 请求”中描述。

SPI 配置支持两种不同的特定模式——主机模式和从机模式。队列操作在两个模式下是相似的。主要的不同在于在主机模式下 DSPI 初始化并控制传输，通过接收队列实体的命令字段。在从机模式下，DSPI 只需要对由总线时钟开始的传输进行回复。SPI 命令字段空间被用于传输数据的 16 个最重要的比特。

49.4.2.1 主机模式

在 SPI 主机模式下，DSPI 通过控制串行时钟（SCK）以及外设芯片选择信号（PCS）来初始化串行传输。在执行的发送队列实体内的 SPI 命令字段决定了 CTAR 寄存器是被用于设置传输属性及来置位哪个 PCS 信号。命令字段同样包含不同的比特用来帮助队列管理及传输协议。可以看 PUSHR 寄存器来了解 SPI 命令字段的详细信息。发送队列实体的数据字段被装载到移位寄存器并且通过 SOUT 引脚移出去。在 SPI 主机模式下，每一个要传输的帧都具有一个命令，以允许传输属性通过不同帧来控制帧。

49.4.2.2 从机模式

化一个传输。当与一个 SPI 主机通信成功时特定的传输属性如时钟极性、时钟相位及帧大小必须置位。SPI 从机模式下，传输属性被设置在 CTAR0 寄存器内。数据以高位先移出。在该模式下，不支持 LSB 先移出。

49.4.2.3 队列禁止操作

队列禁止机制允许 SPI 不使用发送队列或者接收队列进行传输。当队列被禁止时，DSPI 以一种双缓冲区的机制进行操作。接收与发送队列是被分别禁止的。设置 MCR[DIS_TXF]位来禁止发送队列，设置 MCR[DIS_RXF]位来禁止接收队列。

队列禁止机制对用户及主机软件是透明的；待传输的数据及命令被写入到 PUSHHR，从 POPR 读取接收到的数据。

当发送队列被禁止时，SR 寄存器的 TFFF、TFUF 及 TXCTR 位操作就向只有一个队列一样，但是 TXFR 寄存器的内容及 TXNXTPTP 的内容是没有定义的。相似地，当接收队列被禁止时，SR 寄存器的 RFDF、RFOF 及 RXCTR 位就像是有一个实体的队列一样，当时 RXFR 及 POPNXTPTR 寄存器的值没有定义。

49.4.2.4 传输队列先进先出缓冲区机制

发送队列的功能就是为要传输的 SPI 数据以及 SPI 命令提供一个缓冲区。发送队列保持 4 个字，每一个都包含有命令字段及数据字段。在发送队列的实体的个数为设备指定的。通过写 PUSH 寄存器将 SPI 命令及数据添加到发送队列。发送队列的实体通过移出或者清空发送队列可以从发送队列中移走。

DSPI 状态寄存器的发送队列计数器字段用来表明在发送队列中可用的实体的个数。每当 DSPI PUSHHR 被写或者 SPI 数据从发送队列被传入到移位寄存器中时，该字段被更新。

TXNXTPTR 字段用来指示发送队列在下一次传输中将会被发送的实体。TXNXTPTR 字段包含来自于 TXFR0 的偏移地址以 32 位的地址形式。例如，TXNXTPTR 等于 2 意味着 TXFR2 包含有下次要传输的 SPI 数据及命令。每当有 SPI 数据被从发送队列传输到移位寄存器时 TXNXTPTR 字段自动增加 1。该字段的最大值等于 TXFR 寄存器实现的最大的个数，当达到最大时将会翻转。

49.4.2.4.1 填充发送队列

主机软件或者其他智能模块都可以通过写 PUSHHR 寄存器添加（移出）队列到发送队列。当发送队列没有满时，发送队列满标志位（TFFF）会被置位。当发送队列满并且 DMA 控制器表明写 PUSHHR 操作完成时 TFFF 位清 0。向该位写 1 同样会

DMA 请求” 章节来了解详细信息。

DSPI 会忽略试图向已满的发送队列写数据，发送队列的状态不会被改变并且不会导致任何错误条件。

49.4.2.4.2 清除发送队列

通过把 SPI 数据从一位寄存器中移出可以移除 TX FIFO 中的实体。数据从 TX FIFO 中移动一位寄存器，只要在 TX FIFO 中的有效实体都被移出。每次实体从 TX FIFO 移动一位寄存器，TX FIFO 计数器自动加一。在传输的最后，SR 的 TCF 位被置位来指示传输的完成。在 MCR 的 CLR_TXF 位写 1 使 TX FIFO 清零。

如果一个外部主机和从机启动一个传输（当从机的 DSPI TX FIFO 为空），从机 SR 的发送下溢标志置位。细节见发送 FIFO 下溢中断请求。

49.4.2.5 接收先进先出(RX FIFO)缓冲机制

RX FIFO 作为 *sin* 引脚数据接收的缓冲区。RX FIFO 保持 4 个接收的 SPI 数据帧。RX FIFO 中实体的数量是设备设定的。当移位寄存器中接收到的数据被传输到 RX FIFO，在一个传输的完成时，SPI 数据被加到 RX FIFO。通过读取 DSPI POP RX FIFO 寄存器(POPR)，从 RX FIFO 中移除数据。通过读取 POPR 或清除 RX，RX FIFO 实体可以只从 RX FIFO 中移除，FIFO。DSPI 状态寄存器 (SR) 中的 RX FIFO 计数器字段(RXCTR)指示 RX FIFO 中有效数据的数量。每次读取 POPR 或从移位寄存器中复制数据到 RX FIFO，RXCTR 更新。SR 中的 POPNXTPTR 字段指向读取 POPR 时返回的 RX FIFO 中的实体。POPNXTPTR 包含 32 位寄存器中 RXFR0 中的偏移地址。例如，POPNXTPTR 等于两个平均值，RXFR2 包含接收到的数据，该数据将被返回(当读取 POPR 时)。每次读取 POPR，POPNXTPTR 字段将自动加 1。该字段的最大值等于定义的最大 RXFR 寄存器的数量且达到最大值后翻转。

49.4.2.5.1 填充 RX FIFO

用从移位寄存器中接收到的数据填充 RX FIFO。当 RX FIFO 没有满时，从移位寄存器得到的 SPI 帧被传送到 RX FIFO。每次 SPI 帧被传送到 RX FIFO 时，RX FIFO 计数器自动加一。

如果 RX FIFO 和移位寄存器已满，且启动一个传送，SR 的 RFOF 位被置位用于指示一个溢出情况。基于 MCR 中 ROOE 位的状态，来自发送器且产生溢出的数据被忽略或者移到移位寄存器。如果 ROOE 位置位，接下来的数据被移到

49.4.2.5.2 清除 RX FIFO

主机 CPU 或者一个 DMA 请求可以通过读取 DSPI POP RX FIFO 寄存器 (POPR)来从 RX FIFO 中移除实体。读取 POPR，RX FIFO 计数器自动减一。忽略尝试从一个空的 RX FIFO 中抛出数据，RX FIFO 计数器保持不变。从空 RX FIFO 中读取的数据未定义。

当 RX FIFO 不为空时，SR 的 RX FIFO 清除标记(RFDF)被置位。当 RX_FIFO 为空时，RFDF 位被清零，DMA 控制器指示，从 POPR 读取数据完成或通过写 1 到它。

49.4.3 DSPI 波特率和时钟延迟产生

系统时钟的频率除以一个预定标器然后加上双倍波特率除以定标器，这样得到了串行传输的 SCK 频率和延迟的值。图 49-92 概念地显示串行通信时钟频率信号时如何产生的。

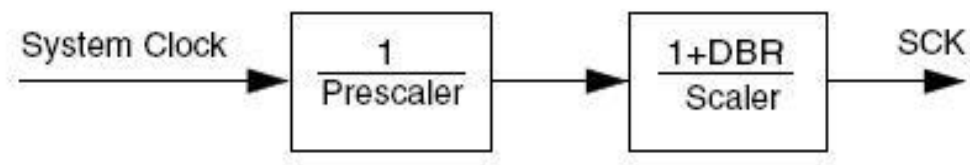


图 49-92 通信时钟预定标器和定标器

49.4.3.1. 波特率的产生

波特率是串行通信时钟(SCK)的频率。系统时钟由预定标器(PBR)和定标器(BR)划分来产生有可能使定标器减半划分的 SCK。在 CTAR 寄存器中, DBR,PBR 和 BR 领字段用 BR 领字段描述中的公式来选择 SCK 频率。表 49-106 显示了如何计算波特率的例子。

表 49-106 波特率计算例子

f _{sys}	PBR	Prescaler	BR	Scaler	DBR	Baud
100 MHz	0b00	2	0b0000	2	0	25Mb/s
20 MHz	0b00	2	0b0000	2	1	10Mb/s

注意：上表提到的时钟频率是作为例子的，参考时钟章节了解在设备中如何使用频率驱动该模块。

49.4.3.2. PCS 到 SCK 的延迟(tcsc)

度。图 49-94 是 PCS 到 SCK 延迟的图例。CTARx 寄存器中的 PCSSCK 和 CSSCK 字段用 CSSCK 领字段描述的公式来选择 PCS 到 SCK 的延迟。表 49-107 显示了如何计算 PCS 到 SCK 的延迟的一个例子。

表 49-107 PCS 到 SCK 延迟计算举例

f _{sys}	PCSSCK	Prescaler	CSSCK	Scaler	PCS to SCK Delay
100MHz	0b01	3	0b0100	32	0.96μs

注意：上表提到的时钟频率是作为例子的，参考时钟章节了解在设备中如何使用频率驱动该模块。

49.4.3.3. SCK 延迟后(t_{ASC})

SCK 延迟后是指 SCK 的最后边缘和 PCS 的反信号之间的时间的长度。图 49-94 和图 49-95 是 SCK 延迟后的图例。在 CTARx 寄存器中 PASC 和 ASC 领字段用 ASC 领字段描述中的公式来选择 SCK 延迟后。表 49-108 显示了如何计算 SCK 延迟后的一个例子。

表 49-108 SCK 延迟后的计算举例

f _{sys}	PASC	Prescaler	ASC	Scaler	After SCK Delay
100MHz	0b01	3	0b0100	32	0.96μs

注意：上表提到的时钟频率是作为例子的，参考时钟章节了解在设备中如何使用频率驱动该模块。

49.4.3.4. 转移后延迟(t_{DT})

转移后延迟是指 PCS 信号相反的一帧和 PCS 信号断言的下一帧之间的最小时间。图 49-94 是转移后延迟的一个图例。在 CTARx 寄存器中，PDT 和 DT 领字段使用 DT 领字段描述中的公式来选择转移后延迟。表 49-109 显示了如何计算转移后延迟的例子。

表 49-109 转移后延迟计算举例

f _{sys}	PDT	Prescaler	DT	Scaler	Delay after Transfer
100MHz	0 b01	3	0b1110	32768	0.98 ms

注意：上表提到的时钟频率是作为例子的，参考时钟章节了解在设备中如何使用频率驱动该模块。

当在不连续的时钟模式下，t_{DT} 延迟根据 CTAR[DT]位字段描述中指定的等式被设置。当在连续的时钟模式下，延迟在 1 个 SCK 周期固定不动。

49.4.3.5. 外设片选使能选通($\overline{\text{PCSS}}$)

$\overline{\text{PCSS}}$ 信号提供了一种延迟来允许 PCS 信号在转移出现后再定下来，这样可以避免毛刺干扰。当 DSPI 在主机模式中，并且 PCSSE 位在 MCR 中设置， $\overline{\text{PCSS}}$ 为外部解复用器提供了一个信号来将 PCS[0]-PCS[4]解码成多达 128 位的无干扰 PCS 信号。图 49-93 显示了 $\overline{\text{PCSS}}$ 信号相对于 PCS 信号的时序。



图 49-93 外设片选选通时间

CTAR 寄存器中的 PCSSCK 领字段根据下列公式选择 PCS 信号的判断和 $\overline{\text{PCSS}}$ 的判断之间的延迟：

$$t_{\text{PCSSCK}} = 1/f_{\text{sys}} * \text{PCSSCK}$$

在转移的结束时,CTAR 寄存器中的 PASC 领字段根据下列公式选择 $\overline{\text{PCSS}}$ 的相反信号和 PCS 的相反信号之间的延迟：

$$t_{\text{PASC}} = 1/f_{\text{sys}} * \text{PASC}$$

表 49-110 显示了如何计算 t_{PCSSCK} 延迟的例子。

表 49-110 外设片选选通判断计算举例

f_{sys}	PCSSCK	Prescaler	Delay before Transfer
100MHz	0b11	7	70.0ns

表 49-111 显示了如何计算 t_{pasc} 延迟的例子。

表 49-111 外设片选选通相反信号计算举例

f_{sys}	PASC	Prescaler	Delay after Transfer
100MHz	0b11	7	70.0ns

当连续串行通信 SCK 模式在使能情况下，不支持 $\overline{\text{PCSS}}$ 信号。

注意：上表提到的时钟频率是作为例子的，参考时钟章节了解在设备中如何使用频率驱动该模块。

49.4.4 传输格式

SPI 串行通信被串行通信时钟(SCK)信号和 PCS 信号控制。主设备提供的 SCK 信号在交换输入和交换输出信号上进行同步传输和数据采集。PCS 信号作

在主模式下，时钟和传输属性寄存器中的 CPOL 和 CPHA 位选择串行时钟的极性和相位，SCK。

- CPOL-选择 SCK 空闲状态的极性
- CPHA-假如交换输出上的数据在第一个 SCK 边沿上或之前有效，选择即使总线从设备没有控制 SCK 信号，在从模式中这些值必须与主设备设置完全相同，这样确保正确的传输。在 SPI 从模式中，只使用 CTAR0。

DSPI 支持四种不同的传输格式：

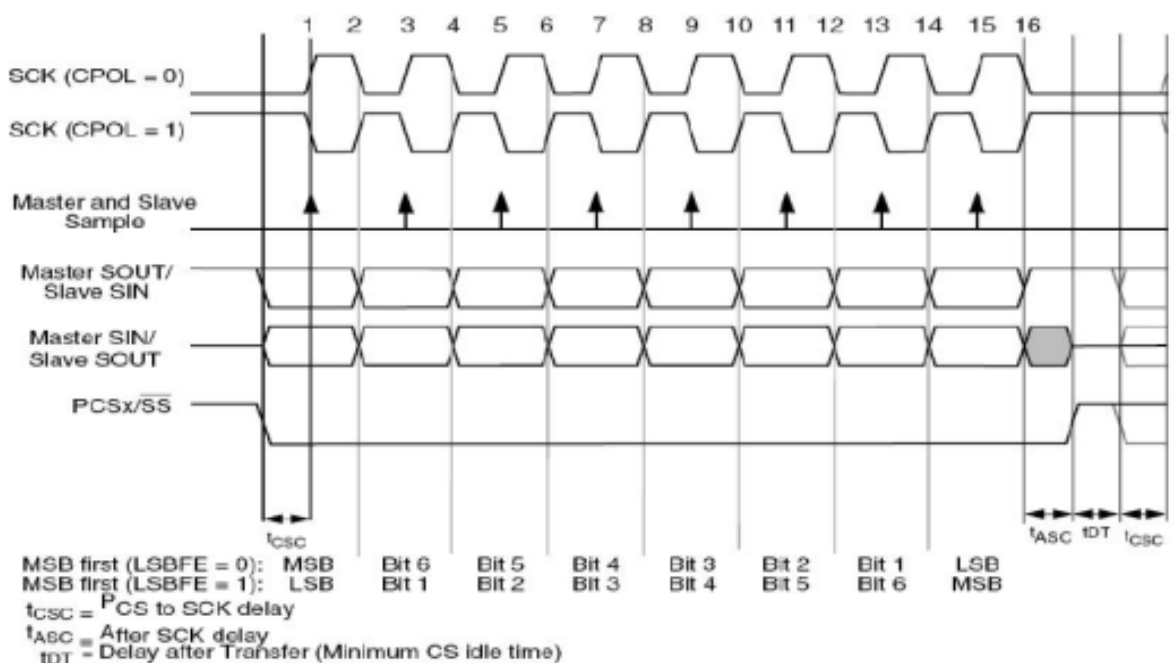
- 典型的 SPI 使用 CPHA=0
- 典型的 SPI 使用 CPHA=1
- 改进的传输格式使用 CPHA=0
- 改进的传输格式使用 CPHA=1

支持一种改进的传输格式来允许高速通信使用需要更长准备时间的外围设备。DSPI 可以在周期的后半段采集输入数据给外围设备更多的准备时间。MCR 中的 MTFE 位在典型的 SPI 格式和改进的传输格式之间选择使用。

在 SPI 配置中，DSPI 提供了 PCS 信号保持帧之间有效的判断。详细参照连续选择格式。

49.4.4.1 典型的 SPI 传输格式(CPHA=0)

在图 49-94 显示的传输格式是用来与第一个数据位在第一个时钟边沿可用的外围 SPI 从设备通信。在这种格式中，主机和从机对奇数 SCK 边沿上的交换输入信号取样，并且交换偶数 SCK 边沿上的交换输出信号上的数据。



主机通过将它的第一位数据位放在 SOUT 引脚上并且判断相应的外围信号选择信号在从

设备上启动传输。从机以将它的第一位数据位放在 SOUT 引脚上作为回应。 t_{CSC} 延迟结束后，主机输出 SCK 的第一个边沿。主从设备利用这个边沿来采样他们的串行数据输入信号的第一个输入数据位。在 SCK 的第二个边沿，主从设备将他们的第二个数据位放在他们的串行数据输出信号上。对于其余的帧，主机和从机采样奇数时钟边沿上的 SIN 引脚，改变偶数时钟边沿上的 SOUT 引脚上的数据。最后的时钟边沿出现后， t_{ASC} 延迟在主机对 PCS 信号取反之前插入。在新的帧传输能被主机启动之前 t_{DT} 延迟会被插入。

49.4.4.2 典型的 SPI 传输格式(CPHA=1)

在图 49-95 显示的传输格式是用来与要求第一个数据位之前的第一个 SCK 边沿在从机 SOUT 引脚上变成可用的外围 SPI 从设备通信。在这种格式中，主机和从机交换奇数 SCK 边沿上的交换输出信号上的数据，并且对偶数 SCK 边沿上的交换输入信号上的数据举例。

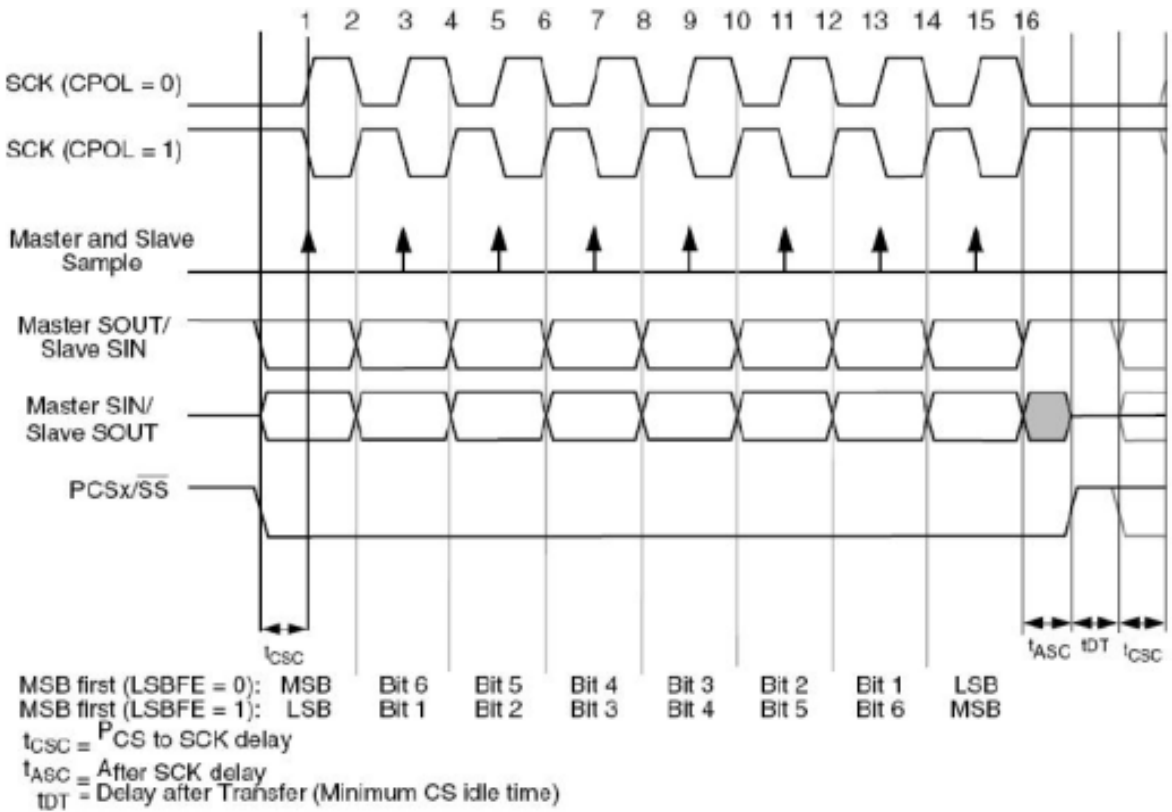


图 49-95 DSPI 传输时间图(MTFE=0,CPHA=1,FMSZ=8)

主机通过使能 PCS 信号在从机上来启动传输。 t_{CSC} 延迟结束后，主机产生第一个 SCK 边沿，同时将有效的数据放在 SOUT 引脚上。从机通过将它的第一位

在 SCK 的第二个边沿，主机和从机采样他们的 SIN 信号。对于其余的帧，主机和从机改变奇数时钟沿上的 SOUT 引脚上的数据，采样偶数时钟沿上的 SIN 引脚。最后的时钟边沿出现后， t_{ASC} 延迟在主机对 PCS 信号取反之前插入。在新的帧传输能被主机启动之前 t_{DT} 延迟会被插入。

49.4.4.3 连续选择格式

在每个传输之间一些外围设备必须被取消。另外一些外围设备在一些连续的串行传输中必须保持选定状态。连续选择格式提供了处理以下情况的灵活性。连续选择格式通过设置 SPI 命令中的 CONT 位启用 SPI 配置。配置中的 PCS 信号的行为是完全相同的，所以只有 SPI 配置会被描述。

当位 CONT = 0，DSPI 驱动判断信号选择信号到帧之间的空闲状态。芯片选择信号的空闲状态被 MCR 中的 PCSISn 位选择。图 49-96 是 2 个四位传输在 CPHA=1 和 CONT=0 的条件下。

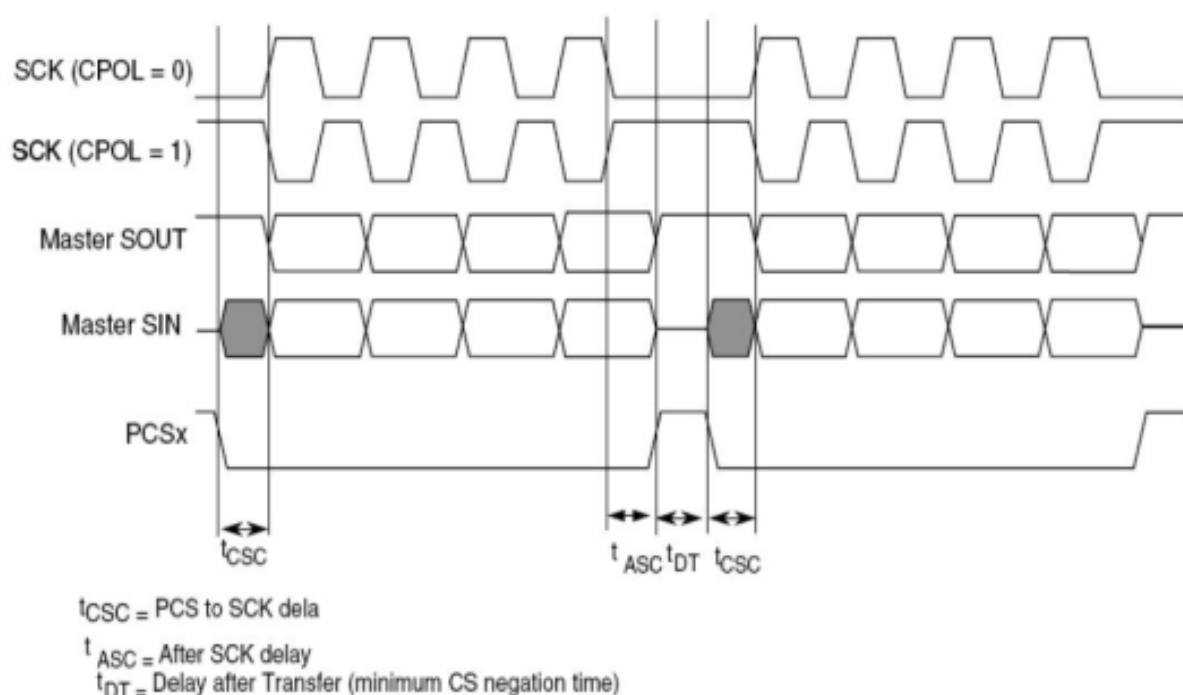


图 49-96 不连续格式举例(CPHA=1,CONT=0)

当位 CONT=1，PCS 信号保持两个传输的时间有效。传输间的延迟(t_{DT})在传输间不被插入。图 49-97 显示了两个四位传输，CPHA=1 和 CONT=1 的时序图。

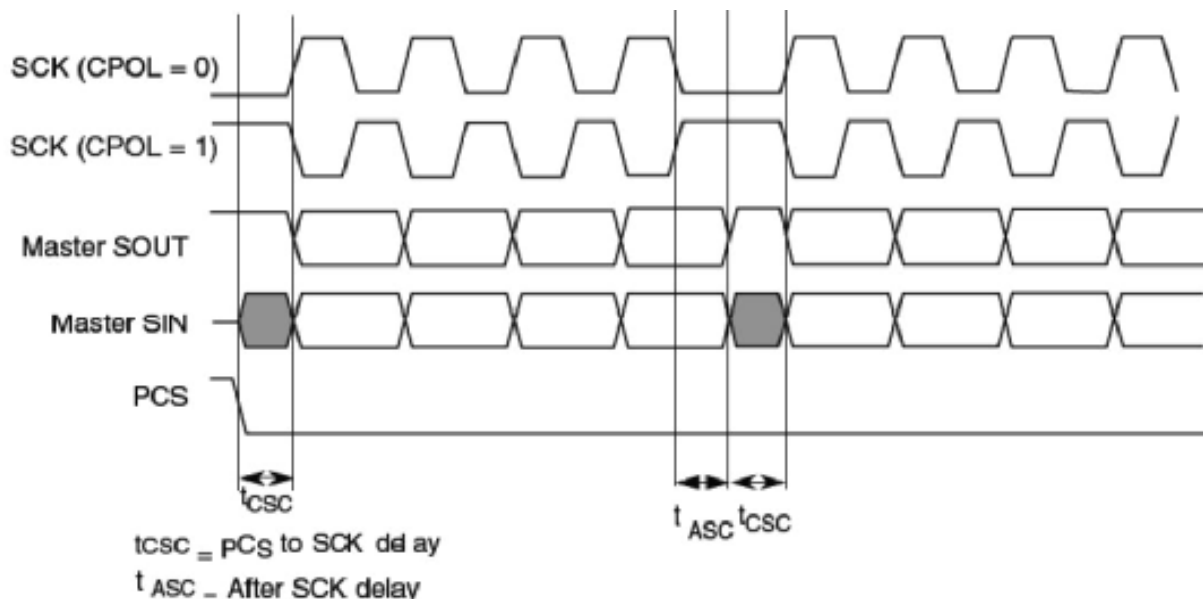


图 49-97 连续格式举例(CPHA=1,CONT=1)

当用连续选择格式使用 DSPI 时需遵循下列规则：

- 所有的传输命令必须有相同的 PCSn 位编程
- 被传输命令选择的 CTARs 必须用相同的传输属性编程。只有 FMSZ 字段可以在这些 CTARs 中不同地编程。
- 当在该模式下发送多个帧，用户软件必须确保最后帧 PUSHR[CONT]位拉高（主模式），且用户软件必须在 TX_FIFO 提供足够被发送的帧（从模式），且在最后帧的传送的最后主机拉高 PCSn
- 在判断 MCR[HALT]位之前必须拉高 PUSHR[CONT] / DSICR0[DCONT] 位（主模式）。这会确保 PCSn 信号被拉高。在连续传送期间判断 MCR[HALT] 位会引起 PCSn 信号保持判断，因此从设备不能从 RUNNING 状态到 STOPPED 状态传输。

注意：在 TXFIFO 变空之前，包括主设备和从设备，在一个 PCS 使能下，必须用将要串联到一起的实体的数量来填充 TXFIFO。当操作在从模式下，确保当 TXFIFO 中的最后实体被传输完成时（相应的 TCF 标志位判断，TXFIFO 为空），任意以后的串行通信从机被取消。否则，出现下溢错误。

49.4.5 连续串行通信时钟

DSPI 提供一个为需要连续时钟的从设备产生连续 SCK 信号的选择。

连续 SCK 通过设置 MCR 中 CONT_SCKE 位而启动。使能该位产生连续串行通信时钟，不管 MCR[HALT]位状态。在所有配置中连续 SCK 是有效的。

连续 SCK 只支持 CPHA=1。如果 CONT_SCKE 位设置为 1，那么 CPHA 清

零的请求就会被忽略。连续 SCK 支持改进的传输格式。

连续 SCK 模式的时钟和传输属性根据以下规则被设定：

- 当 DSPI 在 SPI 配置中时，CTAR0 是最初使用的。在每个 SPI 帧传输的开始，CTAS 为帧指定的 CTAR 会被使用。
- 在所有的配置中，当前选择的 CTAR 保持使用，直到一个不同的 CTAR 指定的帧开始，或者连续 SCK 模式被终止。

当使用连续 SCK 时建议保持波特率相同。当使用连续 SCK 时帧之间的转换时钟极性可能会导致传输中的错误。如果 DSPI 被放入外部停止模式或者模块禁止模式，连续 SCK 操作是不能保证的。

启用的连续 SCK 禁用 PCS 到 SCK 的延迟，传输后的延迟(t_{DT})固定为一个 SCK 周期。图 49-98 是使用连续选择禁用的连续 SCK 格式的图。

注意：当使用连续 SCK 模式，SPI 传输时 CTAR0 要被使用且 TXFIFO 位必须被清零（在初始化传输之前）使用 MCR[CLR_TXF]。

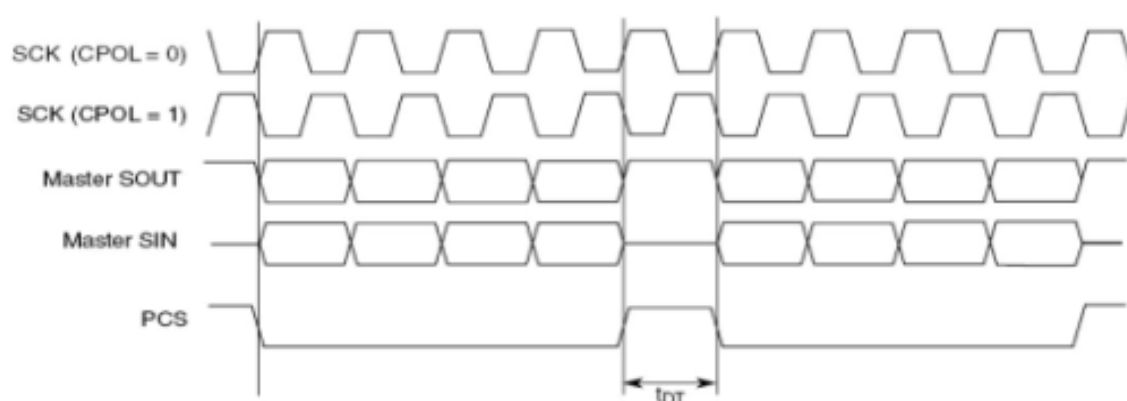


图 49-98 连续 SCK 时间图(CONT=0)

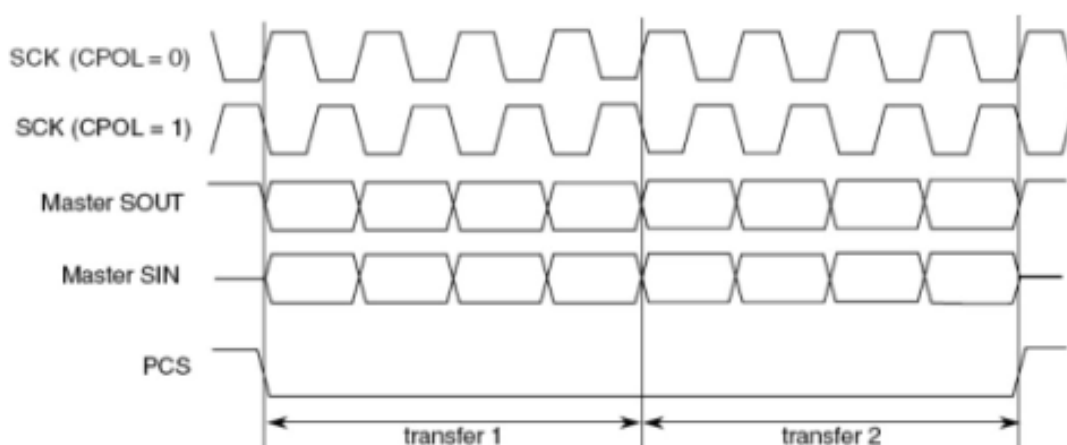


图 49-99 连续 SCK 时间图(CONT=1)

如果发送 FIFO 入口中 CONT 位被设置，PCS 在传输之间保持有效。在某些情况下，SCK 可以继续使用有效的 PCS，但是不能使用移出 SOUT(交换输出拉高)的数据。这可能会导致从机接收错误的的数据。这些情况包括：

- 连续 SCK 且设置的 CONT 位，但没有数据在传输 FIFO 中。
- 连续 SCK 且设置的 CONT 位，进入停止状态。
- 连续 SCK 且设置的 CONT 位，进入停止模式或者模块禁止模式。

图 49-99 显示了连续 SCK 格式使用连续选择启用的时间图。

49.4.6 从机模式运行约束

从机模式逻辑移位寄存器是可缓冲的。当 DSPI 是永久地选择和数据以固定比率转换时，允许数据流操作

发送数据在第二 SCK 钟每一帧的边缘转移数据到移位寄存器,如果 \overline{SS} 信号被声明，任何时候传输数据准备好了，则 \overline{SS} 信号被取消。

接收到的数据转移到接收缓冲在每帧的最后 SCK 边缘，由编程的 CTAR0/1 寄存器设定的帧大小定义。那么这些数据从缓冲区被转移到 RXFIFO 或 DDR 的寄存器。如果 \overline{SS} 在最后 SCK 边缘取消，移位寄存器中数据将丢失。

这个缓冲方案允许操作从机时钟频率比系统频率更高。时钟关系根据下面的公式定义。FrameSize 的值为 CTAR0/1[FMSZ] 加 1。

$$f_{SCK} < f_{SYS} * \text{FrameSize} / 3$$

49.4.7 中断/DMA 请求

DSPI 在几种情况下只能产生中断请求，在两个条件可以产生中断或 DMA 请求。下表列出这些条件。

表 49-112 中断/DMA 请求条件

条件	标识	中断	DMA
队列末尾(EQQ)	EOQF	Yes	
发送 FIFO Fill	TFFF	Yes	Yes
发送完成	TCF	Yes	
发送 FIFO 下溢	TFUF	Yes	
接收 FIFO 结束	RFDF	Yes	Yes
接收 FIFO 下溢	RFOF	Yes	

每种情况有一个标志位在DSPI地址寄存器(SR)和请求使能位在DSPI DMA/中断 请求选择和使能寄存器(RSER)。发送先进先出满足标志(TFFF)和接收先进先出漏极标志(RFDF) 根据RSER中的TFFF_DIRS和RFDF_DIRS位产生中断请求或DMA 请求

DSPI模块的也提供全局中断请求,即任何一个单独的中断请求被声明DSPI将被声明。

49.4.7.1 队列尾中断请求

队列尾指示达到发送队列尾。当在正在执行的 SPI 命令的 EOQ 位被置位且 RSER 中的 EOQF_RE 位被置位，产生队列尾请求。

注意：当 SPI 帧的最后一位和被置位的 EOQ 位被发送时，产生中断请求。

49.4.7.2 发送 FIFO 满中断或 DMA 请求

发送FIFO满请求表明TX FIFO不是满的。发送FIFO满请求发生在当TX FIFO中实体数目比可能实体数量的最小值还要少，TFFF_RE的RSER被设置。TFFF_DIRS的RSER位选择是DMA请求还是中断请求发生，无论DMA请求或中断请求是否发生。

注意：当DMA用于填充TXFIFO时，TFFF标志自动清零。没有使用DMA时，要想清零TFFF，按照以下步骤，每个完成的弹出使用CPU来填充TXFIFO：

1. 等待直到 TFFF = 1
2. 使用 CPU 写数据到 PUSH
3. 写 1 到它的位置清零 TFFF。如果 FIFO 不满，该位不清零。

49.4.7.3 发送完全中断请求

发送完全请求表明串行帧传输的结束。当 RSE 的 TCF_RER 位置位，在每个帧传送的结束产生传送完成请求。

49.4.7.4 发送 FIFO 下溢中断请求

发送FIFO下溢中断请求表明一个下溢情况在发送先进先出发生。发送FIFO下溢中断请求DSPI检测出，执行从机模式和SPI配置。 DSPI的TX FIFO是空的时候TFUF位被置位，一个发送从外部主SPI被初始化.如果TFUF位被设置当TFUF_RE的RSER位被设置，一个中断请求将产生。

49.4.7.5 接收 FIFO 清除中断或 DMA 请求

接收FIFO清除要求表明RX FIFO不是空的。接收FIFO清除请求产生（在RX FIFO中实体的数目不为零时），RFDF_RE 的RSER位被设置。RFDF_DIRS 的RSER位选择产生DMA请求或中断请求。

49.4.7.6 接收 FIFO 溢出中断请求

接收FIFO溢出要求表明一个溢出状态在RX FIFO。当RX FIFO和移位寄存器是满的且一个发送被初始化时，一个接收FIFO溢出请求产生。为中断请求产生，RFOF_RE的RSER位的必须置位。根据MCR的ROOE位的状态,从产生溢出的发送器得到的数据，要么是忽略或者转移到移位寄存器。如果ROOE位设置,后面的数据转移到移位寄存器。如果ROOE位被清除,以后的数据将被忽略。

49.4.8 低功耗特征

DSPI支持以下的低功耗策略:

- 外部停止模式
- 模块禁用模式-非内存映射的逻辑的时钟闸控

49.4.8.1 停止模式（外部停止模式）

DSPI 支持停止模式协议。当一个请求被用来进入外部停止模式，DSPI 块识别该请求。如果一个串行传输正在进行，DSPI 等待直到它达到帧边界（在它准备好关闭时钟前）。当时钟关闭时，DSPI 内存映射逻辑不可访问。中断和 DMA 请求信号的状态不可改变，在外部停止模式的时候。

49.4.8.2. 模块禁用模式

模块禁用模式是模块特定的模式，DSPI才能进入节约能源。

主机可以初始化模块禁用模式通过设置MDIS的MCR位。模块禁用模式也可以由硬件初始化。一个电源管理块可初始化模块禁用模式通过声明DOZE模块信号当DOZE的MCR位被设置。

当MDIS位被设置或DOZE模式信号被声明当DOZE被设置,DSPI在下一个帧边界取消时钟使能信号。如果实施,时钟使能信号可以对非内存映射的逻辑停止时钟。当使能时钟被取消,DSPI是处于睡眠状态,但内存映射寄存器仍可访问的。某些读或写操作有不同的效果当DSPI在模块禁用模式时。读RX FIFO弹出寄存器不能改变RX FIFO的状态。同样的,写TX FIFO压入寄存器不能改变TX FIFO的状态,清除任何FIFOs模块对禁用模式没有影响。MCR的DIS_TXF和DIS_RXF改变不会影响模块中禁用模式。模块中禁用模式,所有地址位和寄存器标志在DSPI读取的返回正确的值,但写入却没有影响。在启用模块禁用模式时写入TCR没有任何效果。中断和DMA请求信号不能被清除在模块禁用模式。

49.5 初始化/应用信息

这一节将描述如何初始化DSPI模块。

49.5.1 如何管理 DSPI 队列

这些队列并不属于DSPI,但DSPI包括支持队列管理的特点。队列主要是支持SPI配置。

1. 当DSPI执行最后控制命令, EOQ位在控制命令被设置表明在DSPI中这是最后一个实体。
2. 在发送的最后,相应的控制命令与EOQ设置被采样, SR中EOQ标志(EOQF)被设置。
3. 设置EOQF标志不能串行发送和接收数据,置DSPI在STOPPED状态。TXRXS位清零表明在STOPPED状态。
4. DMA 可以继续填充 TX FIFO 直到满或第 5 步发生。
5. 通过禁用DMA使能请求DMA通道禁用DSPI DMA 发送, 使DMA通道被分配给TX FIFO 和 RX FIFO。通过清除DMA控制器中相应的DMA 使能要求位完成这个设置。
6. 通过读SR的RXCNT或检查SR中RFDF在每一个POPR读操作后, 确定所有RX FIFO接收的数据发送到存储接收队列。
7. 位新队列修改DMA描述TX 和RX描述符
8. 通过写1到MCR的CLR_TXF位清除TX FIFO。通过写1到MCR的CLR_RXF清除RX FIFO。
9. 通过设置CTCNT位在新队列第一个实体的控制命令字或由CPU直接在TCR SPI_TCNT写入, 清除发送计数。
10. 通过使能DMA请求使能DMA通道, 通过设置相应的DMA 设置使能请求位使DMA通道分配给DSPI TX FIFO和 RX FIFO。
11. 通过清除EOQF位使能串行发送和串行接收可以。

49.5.2 主机和从机切换模式

当在DSPI改变模式,按照下面的步骤,以保证正常工作。

1. 通过设置MCR[HALT]停止DSPI。
2. 清除发送和接收FIFOs通过设置MCR的CLR_TXF和CLR_RXF位为1。
3. 在MCR[MSTR]设定合理的模式, 通过清除MCR[HALT] 驱动DSPI。

49.5.3 波特率设置

定标器结合产生。

假设计算值100兆赫系统频率和双波特率DBR位被清零。

注意：提到的时钟频率在这个章节是一个样例。参考时钟章节得到用于在该设备中驱动该模块的频率。

表 49-113 波特率值(bps)

		波特率定标器值			
		2	3	5	7
波特率定标器值	2	25.0M	16.7M	10.0M	7.14M
	4	12.5M	8.33M	5.00M	3.57M
	6	8.33M	5.56M	3.33M	2.38M
	8	6.25M	4.17M	2.50M	1.79M
	16	3.12M	2.08M	1.25M	893k
	32	1.56M	1.04M	625k	446k
	64	781k	521k	312k	223k
	128	391k	260k	156k	112k
	256	195k	130k	78.1k	55.8k
	512	97.7k	65.1k	39.1k	27.9k
	1024	48.8k	32.6k	19.5k	14.0k
	2048	24.4k	16.3k	9.77k	6.98k
	4096	12.2k	8.14k	4.88k	3.49k
	8192	6.10k	4.07k	2.44k	1.74k
	16384	3.05k	2.04k	1.22k	872
	32768	1.53k	1.02k	610	436

49.5.4 延迟设置

表 49-114 显示延迟以后发送的值(t_{DT})和 CS 到 SCK,延迟(TCSC), 该延迟可以基于 CTAR 寄存器中的前置分频器值和分频系数产生。假设计算值 100 MHz 系统频率。

注意：提到的时钟频率在这个章节是一个样例。参考时钟章节得到用于在该设备中驱动该模块的频率。

表 49-114 延迟值

		延迟定标器值			
		1	3	5	7
延迟定标器值	2	20.0 ns	60.0 ns	100.0 ns	140.0 ns
	4	40.0 ns	120.0 ns	200.0 ns	280.0 ns
	8	80.0 ns	240.0 ns	400.0 ns	560.0 ns
	16	160.0 ns	480.0 ns	800.0 ns	1.1 μ s
	32	320.0 ns	960.0 ns	1.6 μ s	2.2 μ s
	64	640.0 ns	1.9 μ s	3.2 μ s	4.5 μ s
	128	1.3 μ s	3.8 μ s	6.4 μ s	9.0 μ s
	256	2.6 μ s	7.7 μ s	12.8 μ s	17.9 μ s
	512	5.1 μ s	15.4 μ s	25.6 μ s	35.8 μ s
	1024	10.2 μ s	30.7 μ s	51.2 μ s	71.7 μ s
	2048	20.5 μ s	61.4 μ s	102.4 μ s	143.4 μ s
	4096	41.0 μ s	122.9 μ s	204.8 μ s	286.7 μ s

	32768	327.7 μs	983.0 μs	1.6 ms	2.3 ms
	65536	655.4 μs	2.0 ms	3.3 ms	4.6 ms

49.5.5 计算 FIFO 指针地址

通过 FIFO 寄存器 TX 和 RX FIFO 的内容完全可见是可以的。对每一个 FIFO，通过存储器映射指针和一个存储器映射计数器有效的实体可被识别。在每一个 FIFO 指针指向第一个进入的实体。TX FIFO 的第一个指针是发送到一个指针(TXNXTPTR)。RX FIFO 第一个指针是顶部下一个指针。图 49-100 说明了 FIFO 计数器中第一个和最后一个 FIFO 实体的概念和 FIFO 计数器一起。TX FIFO 被选为例子，但也可以作为说明 RX FIFO 的概念。

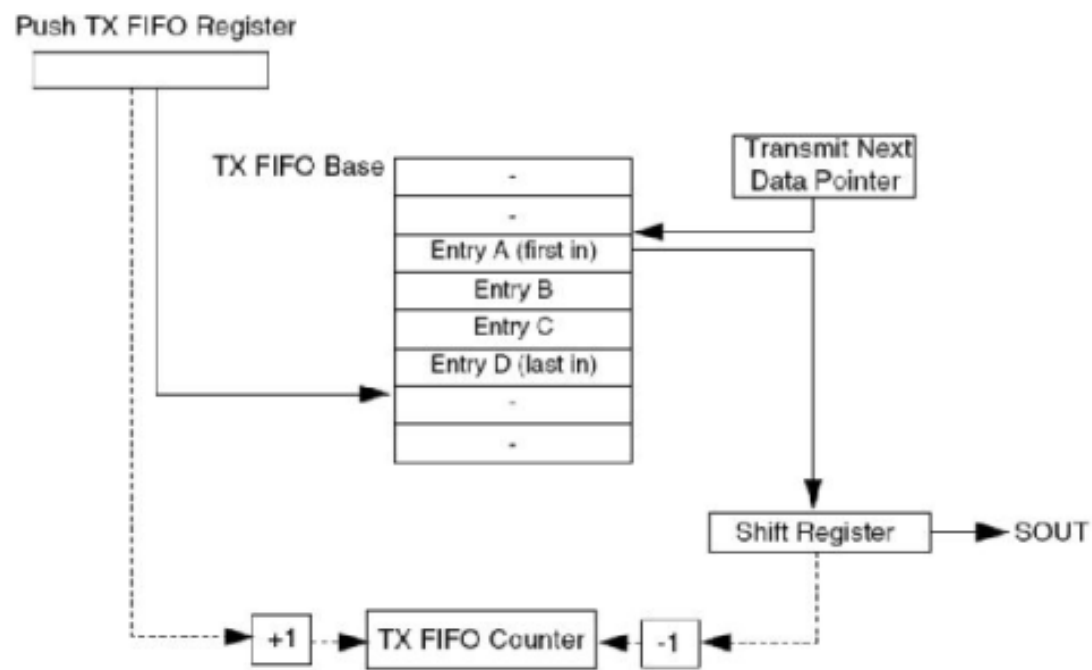


图 49-100 发送先进先出指针和计数器

49.5.5.1 TX FIFO 中第一个和最后一个实体地址计算

TX FIFO 中第一个实体的内存地址通过下列公式计算:

第一个实体地址=TXFIFOBase+(4*TXNXTPTR)

TX FIFO 中最后一个实体的内存地址通过下列公式计算:

最后一个实体地址=

$$TXFIFOBase+4*(TXCTR+TXNXTPTR-1)mod(TXFIFOdepth)$$

TX FIFO Base –TX FIFO 的基地址

TXCTR –TX FIFO 计数器

TXNXTPTR – 发送下一个指针

49.5.5.2 RX FIFO 中第一个和最后一个实体地址计算

RX FIFO 中第一个实体的内存地址通过下列公式计算:

第一个实体地址=TXFIFOBase+(4*POPNXTPTR)

RX FIFO 中最后一个实体的内存地址通过下列公式计算:

最后一个实体地址=

$$\text{RXFIFOBase} + 4 * (\text{RXCTR} + \text{POPNXTPTR} - 1) \bmod (\text{RXFIFOdepth})$$

RX FIFO Base –RX FIFO 基地址

RXCTR –RX FIFO 计数器

POPNXTPTR – 推出下一个指针

RX FIFO Depth –RX FIFO 深度，具体实施