

第 17 章 10 位 A/D 转换器

目录

本章包括下列主题：

17.1	简介	17-2
17.2	控制寄存器	17-4
17.3	ADC 工作原理、术语和转换序列	17-24
17.4	ADC 模块配置	17-26
17.5	其他 ADC 功能	17-38
17.6	初始化	17-59
17.7	中断	17-61
17.8	I/O 引脚控制	17-62
17.9	休眠和空闲模式下的操作	17-63
17.10	各种复位的影响	17-65
17.11	设计技巧	17-66
17.12	相关应用笔记	17-71
17.13	版本历史	17-72

17.1 简介

PIC32MX 10 位模数（Analog-to-Digital，A/D）转换器（或 ADC）具有以下特性：

- 逐次逼近寄存器（Successive Approximation Register，SAR）转换
- 最多 16 个模拟输入引脚
- 外部参考电压输入引脚
- 一个单极性差分采样 / 保持放大器（Sample-and-Hold Amplifier，SHA）
- 自动通道扫描模式
- 可选的转换触发源
- 16 字的转换结果缓冲区
- 可选的缓冲区填充模式
- 8 个转换结果格式选项
- 可在 CPU SLEEP（休眠）和 IDLE（空闲）模式下工作

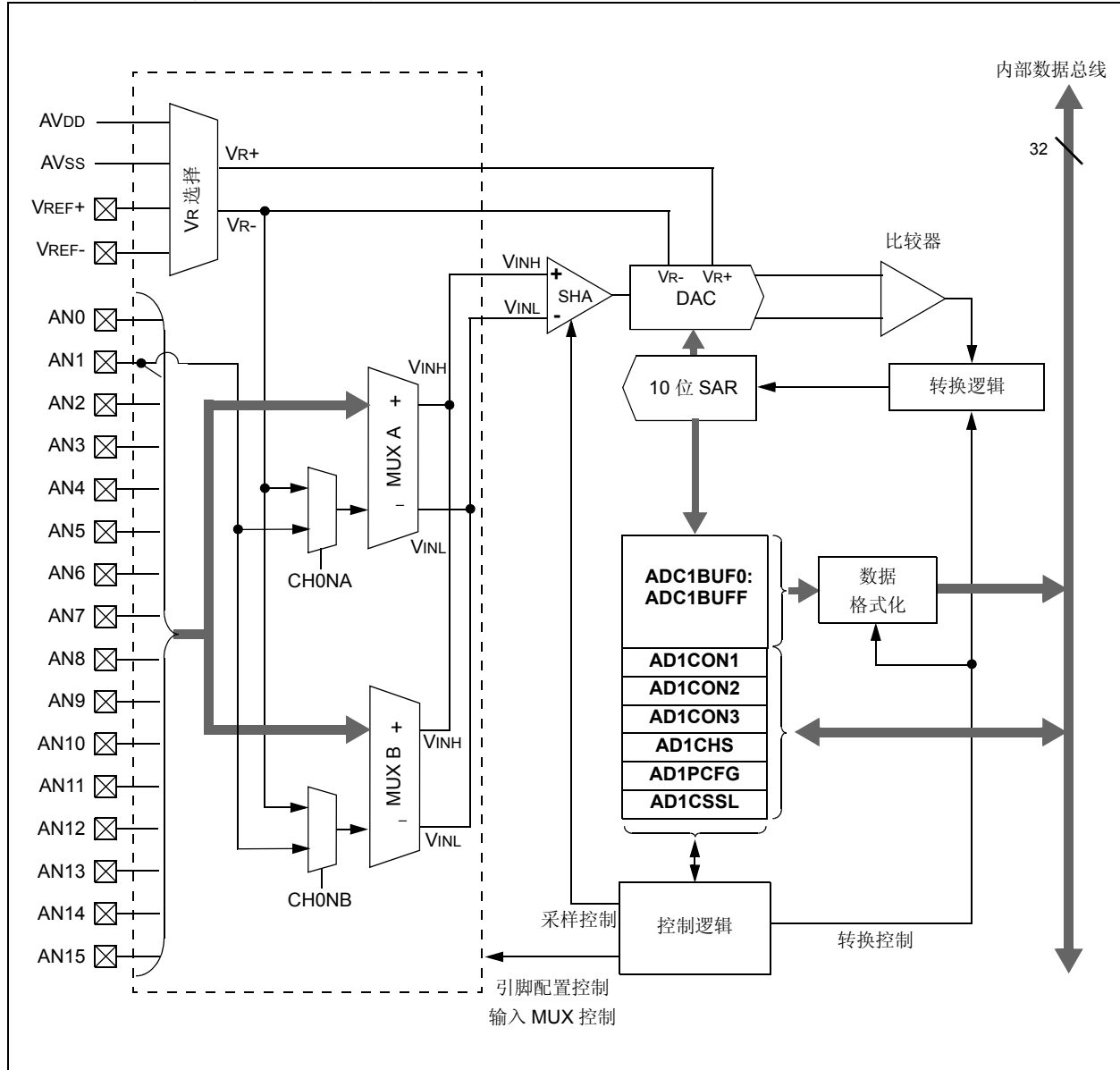
图 17-1 给出了 10 位 ADC 的框图。10 位 ADC 最多有 16 个模拟输入引脚，指定为 AN0-AN15。此外，还有两个模拟输入引脚用于连接外部参考电压。这些参考电压输入可能会与其他模拟输入引脚共用，并且可能也用作其他模拟模块的参考输入。实际的模拟输入引脚数和外部参考电压输入配置取决于具体的 PIC32MX 器件。更多详细信息，请参见器件数据手册。

模拟输入通过两个多路开关（MUX）与一个 SHA 连接。在转换期间，模拟输入 MUX 可以在两组模拟输入之间切换。使用参考输入引脚时，用作参考输入的引脚除外，所有通道上都可以进行单极性差分转换（见图 17-1）。

模拟输入扫描模式可以按顺序转换用户指定的一些通道。控制寄存器用于指定在扫描序列中要包含的模拟输入通道。

10 位 ADC 与 16 字结果缓冲区连接。在从结果缓冲区读取每个 10 位转换结果时，转换结果被转换为 8 种 32 位输出格式之一。

图 17-1: 10 位高速 A/D 转换器框图



17.2 控制寄存器

ADC 模块包含以下特殊功能寄存器（Special Function Register, SFR）：

AD1CON1、AD1CON2 和 AD1CON3 寄存器用于控制 ADC 模块的操作。

- AD1CON1: ADC 控制寄存器 1
AD1CON1CLR、AD1CON1SET 和 AD1CON1INV: AD1CON1 的原子级位操作只写寄存器。
- AD1CON2: ADC 控制寄存器 2
AD1CON2CLR、AD1CON2SET 和 AD1CON2INV: AD1CON2 的原子级位操作只写寄存器。
- AD1CON3: ADC 控制寄存器 3
AD1CON3CLR、AD1CON3SET 和 AD1CON3INV: AD1CON3 的原子级位操作只写寄存器。

AD1CHS 寄存器用于选择要连接到 SHA 的输入引脚。

- AD1CHS: ADC 输入通道选择寄存器
AD1CHSCLR、AD1CHSSET 和 AD1CHSINV: AD1CHS 的原子级位操作只写寄存器。

AD1PCFG 寄存器用于将模拟输入引脚配置为模拟输入或数字 I/O。

- AD1PCFG: ADC 端口配置寄存器
AD1PCFGCLR、AD1PCFGSET 和 AD1PCFGINV: AD1PCFG 的原子级位操作只写寄存器。

AD1CSSL 寄存器用于选择要顺序扫描的输入。

- AD1CSSL: ADC 输入扫描选择寄存器
AD1CSSLCLR、AD1CSSLSET 和 AD1CSSLINV: AD1CSSL 的原子级位操作只写寄存器。

此外，ADC 模块还具有以下用于中断控制的相关位：

- “IFS1: 中断标志状态寄存器 1”中的中断请求标志状态位（AD1IF）
- “IEC1: 中断允许控制寄存器 1”中的中断允许控制位（AD1IE）
- “IPC6: 中断优先级控制寄存器 6”中的中断优先级控制位（AD1IP<2:0>）和（AD1IS<1:0>）

17.2.1 与 10 位 ADC 相关的特殊功能寄存器

下表汇总了所有与 ADC 相关的寄存器，包括它们的地址和格式。该汇总表之后列出了相应的寄存器，并且每个寄存器均附有详细的说明。所有未实现的寄存器和 / 或寄存器中的位都读为 0。

表 17-1: ADC SFR 汇总

名称		Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
AD1CON1	31:24	—	—	—	—	—	—	—	—
	23:16	—	—	—	—	—	—	—	—
	15:8	ON	FRZ	SIDL	—	—	FORM2	FORM1	FORM0
	7:0	SSRC2	SSRC1	SSRC0	CLRASAM	—	ASAM	SAMP	DONE
AD1CON1CLR	31:0	写入时会将 AD1CON1 中的选定位置清零，读取时获得的值未定义							
AD1CON1SET	31:0	写入时会将 AD1CON1 中的选定位置 1，读取时获得的值未定义							
AD1CON1INV	31:0	写入时会将 AD1CON1 中的选定位置取反，读取时获得的值未定义							
AD1CON2	31:24	—	—	—	—	—	—	—	—
	23:16	—	—	—	—	—	—	—	—
	15:8	VCFG2	VCFG1	VCFG0	OFFCAL	—	CSCNA	—	—
	7:0	BUFS	—	SMPI3	SMPI2	SMPI1	SMPI0	BUFM	ALTS
AD1CON2CLR	31:0	写入时会将 AD1CON2 中的选定位置清零，读取时获得的值未定义							
AD1CON2SET	31:0	写入时会将 AD1CON2 中的选定位置 1，读取时获得的值未定义							
AD1CON2INV	31:0	写入时会将 AD1CON2 中的选定位置取反，读取时获得的值未定义							
AD1CON3	31:24	—	—	—	—	—	—	—	—
	23:16	—	—	—	—	—	—	—	—
	15:8	ADRC	—	—	SAMC4	SAMC3	SAMC2	SAMC1	SAMC0
	7:0	ADCS7	ADCS6	ADCS5	ADCS4	ADCS3	ADCS2	ADCS1	ADCS0
AD1CON3CLR	31:0	写入时会将 AD1CON3 中的选定位置清零，读取时获得的值未定义							
AD1CON3SET	31:0	写入时会将 AD1CON3 中的选定位置 1，读取时获得的值未定义							
AD1CON3INV	31:0	写入时会将 AD1CON3 中的选定位置取反，读取时获得的值未定义							
AD1CHS	31:24	CH0NB	—	—	—	CH0SB3	CH0SB2	CH0SB1	CH0SB0
	23:16	CH0NA	—	—	—	CH0SA3	CH0SA2	CH0SA1	CH0SA0
	15:8	—	—	—	—	—	—	—	—
	7:0	—	—	—	—	—	—	—	—
AD1CHSCLR	31:0	写入时会将 AD1CHS 中的选定位置清零，读取时获得的值未定义							
AD1CHSSET	31:0	写入时会将 AD1CHS 中的选定位置 1，读取时获得的值未定义							
AD1CHS1INV	31:0	写入时会将 AD1CHS 中的选定位置取反，读取时获得的值未定义							
AD1PCFG	31:24	—	—	—	—	—	—	—	—
	23:16	—	—	—	—	—	—	—	—
	15:8	PCFG15	PCFG14	PCFG13	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8
	7:0	PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0
AD1PCFGCLR	31:0	写入时会将 AD1PCFG 中的选定位置清零，读取时获得的值未定义							
AD1PCFGSET	31:0	写入时会将 AD1PCFG 中的选定位置 1，读取时获得的值未定义							
AD1PCFGINV	31:0	写入时会将 AD1PCFG 中的选定位置取反，读取时获得的值未定义							
AD1CSSL	31:24	—	—	—	—	—	—	—	—
	23:16	—	—	—	—	—	—	—	—
	15:8	CSSL15	CSSL14	CSSL13	CSSL12	CSSL11	CSSL10	CSSL9	CSSL8
	7:0	CSSL7	CSSL6	CSSL5	CSSL4	CSSL3	CSSL2	CSSL1	CSSL0
AD1CSSLCLR	31:0	写入时会将 AD1CSSL 中的选定位置清零，读取时获得的值未定义							

表 17-1: ADC SFR 汇总 (续)

名称		Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
AD1CSSLSET	31:0	写入时会将 AD1CSSL 中的选定位置 1，读取时获得的值未定义							
AD1CSSLINV	31:0	写入时会将 AD1CSSL 中的选定位置取反，读取时获得的值未定义							
ADC1BUF0	31:0	ADC 结果字 0（ADC1BUF0<31:0>）							
ADC1BUF1	31:0	ADC 结果字 1（ADC1BUF1<31:0>）							
ADC1BUF2	31:0	ADC 结果字 2（ADC1BUF2<31:0>）							
ADC1BUF3	31:0	ADC 结果字 3（ADC1BUF3<31:0>）							
ADC1BUF4	31:0	ADC 结果字 4（ADC1BUF4<31:0>）							
ADC1BUF5	31:0	ADC 结果字 5（ADC1BUF5<31:0>）							
ADC1BUF6	31:0	ADC 结果字 6（ADC1BUF6<31:0>）							
ADC1BUF7	31:0	ADC 结果字 7（ADC1BUF7<31:0>）							
ADC1BUF8	31:0	ADC 结果字 8（ADC1BUF8<31:0>）							
ADC1BUF9	31:0	ADC 结果字 9（ADC1BUF9<31:0>）							
ADC1BUFA	31:0	ADC 结果字 A（ADC1BUFA<31:0>）							
ADC1BUFB	31:0	ADC 结果字 B（ADC1BUFB<31:0>）							
ADC1BUFC	31:0	ADC 结果字 C（ADC1BUFC<31:0>）							
ADC1BUFD	31:0	ADC 结果字 D（ADC1BUFD<31:0>）							
ADC1BUFE	31:0	ADC 结果字 E（ADC1BUFE<31:0>）							
ADC1BUFF	31:0	ADC 结果字 F（ADC1BUFF<31:0>）							
IFS1	31:24	—	—	—	—	—	—	USBIF	FCEIF
	23:16	—	—	—	—	DMA3IF	DMA2IF	DMA1IF	DMA0IF
	15:8	RTCCIF	FSCMIF	I2C2MIF	I2C2SIF	I2C2BIF	U2TXIF	U2RXIF	U2EIF
	7:0	SPI2RXIF	SPI2TXIF	SPI2EIF	CMP2IF	CMP1IF	PMPIF	AD1IF	CNIF
IFS1CLR	31:0	写入时会将 IFS1 中的选定位置清零，读取时获得的值未定义							
IFS1SET	31:0	写入时会将 IFS1 中的选定位置 1，读取时获得的值未定义							
IFS1INV	31:0	写入时会将 IFS1 中的选定位置取反，读取时获得的值未定义							
IEC1	31:24	—	—	—	—	—	—	USBIE	FCEIE
	23:16	—	—	—	—	DMA3IE	DMA2IE	DMA1IE	DMA0IE
	15:8	RTCCIE	FSCMIE	I2C2MIE	I2C2SIE	I2C2BIE	U2TXIE	U2RXIE	U2EIE
	7:0	SPI2RXIE	SPI2TXIE	SPI2EIE	CMP2IE	CMP1IE	PMPIE	AD1IE	CNIE
IPC6	31:24	—	—	—	AD1IP<2:0>			AD1IS<1:0>	
	23:16	—	—	—	CNIP<2:0>			CNIS<1:0>	
	15:8	—	—	—	I2C1IP<2:0>			I2C1IS<1:0>	
	7:0	—	—	—	U1IP<2:0>			U1IS<1:0>	
IPC6CLR	31:0	写入时会将 IPC6 中的选定位置清零，读取时获得的值未定义							
IPC6SET	31:0	写入时会将 IPC6 中的选定位置 1，读取时获得的值未定义							
IPC6INV	31:0	写入时会将 IPC6 中的选定位置取反，读取时获得的值未定义							

寄存器 17-1: AD1CON1: ADC 控制寄存器 1

r-x	r-x	r-x	r-x	r-x	r-x	r-x	r-x
—	—	—	—	—	—	—	—
bit 31			bit 24				

r-x	r-x	r-x	r-x	r-x	r-x	r-x	r-x
—	—	—	—	—	—	—	—
bit 23			bit 16				

R/W-0	R/W-0	R/W-0	r-x	r-x	R/W-0	R/W-0	R/W-0
ON	FRZ	SIDL	—	—	FORM<2:0>		
bit 15			bit 8				

R/W-0	R/W-0	R/W-0	R/W-0	r-x	R/W-0	R/W-0	R/C-0
SSRC<2:0>			CLRASAM	—	ASAM	SAMP	DONE
bit 7			bit 0				

图注:

R = 可读位

W = 可写位

P = 可编程位

r = 保留位

U = 未实现位

-n = POR 时的值: (0, 1, x = 未知)

bit 31-16 保留: 写入 0; 忽略读操作

bit 15 ON: ADC 工作模式位

1 = A/D 转换器模块正在工作

0 = A/D 转换器关闭

注: 使用 1:1 PBCLK 分频比时, 在清零模块 ON 位的指令之后, 用户的软件不应立即在 SYSCLK 周期中读 / 写外设的 SFR。

bit 14 FRZ: 调试异常模式冻结位

1 = 在 CPU 进入调试异常模式时冻结工作

0 = 在 CPU 进入调试异常模式时继续工作

注: FRZ 仅在调试异常模式下可写。它在正常模式下读为 0。

bit 13 SIDL: IDLE (空闲) 模式停止位

1 = 当器件进入 IDLE (空闲) 模式时, 模块停止工作

0 = 在 IDLE (空闲) 模式下模块继续工作

bit 12-11 保留: 写入 0; 忽略读操作

bit 10-8 FORM<2:0>: 数据输出格式位

011 = 有符号小数, 16 位 (DOUT = 0000 0000 0000 0000 sddd dddd dd00 0000)

010 = 小数, 16 位 (DOUT = 0000 0000 0000 0000 dddd dddd dd00 0000)

001 = 有符号整数, 16 位 (DOUT = 0000 0000 0000 0000 ssss sssd dddd dddd)

000 = 整数, 16 位 (DOUT = 0000 0000 0000 0000 00dd dddd dddd)

111 = 有符号小数, 32 位 (DOUT = sddd dddd dd00 0000 0000 0000 0000 0000)

110 = 小数, 32 位 (DOUT = dddd dddd dd00 0000 0000 0000 0000 0000)

101 = 有符号整数, 32 位 (DOUT = ssss ssss ssss ssss ssss sssd dddd dddd)

100 = 整数, 32 位 (DOUT = 0000 0000 0000 0000 0000 00dd dddd dddd)

寄存器 17-1: AD1CON1: ADC 控制寄存器 1 (续)

bit 7-5

SSRC<2:0>: 转换触发源选择位

111 = 由内部计数器结束采样并启动转换 (自动转换)
110 = 保留
101 = 保留
100 = 保留
011 = 保留
010 = 由 Timer3 周期匹配结束采样并启动转换
001 = 由 INTO 引脚的有效跳变结束采样并启动转换
000 = 由清零 SAMP 位结束采样并启动转换

bit 4

CLRASAM: 停止转换序列位 (在产生第一个 A/D 转换器中断时)

1 = 在产生第一个 ADC 中断时停止转换。当产生 ADC 中断时, 硬件会清零 ASAM 位。
0 = 正常工作, 缓冲区内容会被下一个转换序列覆盖

bit 3

保留: 写入 0; 忽略读操作

bit 2

ASAM: ADC 采样自动启动位

1 = 最后一次转换结束后立即开始采样; SAMP 位自动置 1
0 = SAMP 位置 1 时开始采样

bit 1

SAMP: ADC 采样使能位

1 = ADC SHA 正在采样
0 = ADC SHA 保持
当 ASAM = 0 时, 向该位写入 1 将启动采样。
当 SSRC = 000 时, 向该位写入 0 将结束采样并启动转换。

bit 0

DONE: A/D 转换状态位

1 = A/D 转换已完成
0 = A/D 转换尚未完成或尚未启动
清零该位不会影响进行中的任何操作。

注: 在自动模式下, DONE 位不是持久性的。在下次采样开始时, 硬件会将它清零。

寄存器 17-2: AD1CON1CLR: ADC 控制 1 清零寄存器

写入时会将 AD1CON1 中的选定位清零，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 将 **AD1CON1** 中的选定位清零

在一个或多个位中写入 1 会将 **AD1CON1** 寄存器中的相应位清零，但不会影响未实现位或只读位。写入 0 不会影响该寄存器。

示例: $\text{AD1CON1CLR} = 0x00008002$ 时，会将 **AD1CON1** 寄存器中的 bit 15 和 bit 1 清零。

寄存器 17-3: AD1CON1SET: ADC 控制 1 置 1 寄存器

写入时会将 AD1CON1 中的选定位置 1，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 将 **AD1CON1** 中的选定位置 1

在一个或多个位中写入 1 会将 **AD1CON1** 寄存器中的相应位置 1，但不会影响未实现位或只读位。写入 0 不会影响该寄存器。

示例: $\text{AD1CON1SET} = 0x00008002$ 时，会将 **AD1CON1** 寄存器中的 bit 15 和 bit 1 置 1。

寄存器 17-4: AD1CON1INV: ADC 控制 1 取反寄存器

写入时会将 AD1CON1 中的选定位取反，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 将 **AD1CON1** 中的选定位取反

在一个或多个位中写入 1 会将 **AD1CON1** 寄存器中的相应位取反，但不会影响未实现位或只读位。写入 0 不会影响该寄存器。

示例: $\text{AD1CON1INV} = 0x00008002$ 时，会将 **AD1CON1** 寄存器中的 bit 15 和 bit 1 取反。

PIC32MX 系列参考手册

寄存器 17-5: AD1CON2: ADC 控制寄存器 2

r-X	r-X	r-X	r-X	r-X	r-X	r-X	r-X
—	—	—	—	—	—	—	—
bit 31				bit 24			
r-X	r-X	r-X	r-X	r-X	r-X	r-X	r-X
—	—	—	—	—	—	—	—
bit 23				bit 16			
R/W-0	R/W-0	R/W-0	R/W-0	r-X	R/W-0	r-X	r-X
VCFG<2:0>			OFFCAL	—	CSCNA	—	—
bit 15				bit 8			
R-0	r-X	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BUFS	—	SMPI<3:0>				BUFM	ALTS
bit 7				bit 0			

图注:

R = 可读位

W = 可写位

P = 可编程位

r = 保留位

U = 未实现位

-n = POR 时的值: (0, 1, x = 未知)

bit 31-16

保留: 写入 0; 忽略读操作

bit 15-13

VCFG<2:0>: 参考电压配置位

ADC Vr+		ADC Vr-
000	AVDD	AVSS
001	外部 VREF+ 引脚	AVSS
010	AVDD	外部 VREF- 引脚
011	外部 VREF+ 引脚	外部 VREF- 引脚
1xx	AVDD	AVSS

bit 12

OFFCAL: 输入失调校准模式选择位

1 = 使能失调校准模式

SHA 的 VINH 和 VINL 连接到 Vr-

0 = 禁止失调校准模式

SHA 的输入由 AD1CHS 或 AD1CSSL 控制

bit 11

保留: 写入 0; 忽略读操作

bit 10

CSCNA: MUX A 输入多路开关设置的 CH0+ SHA 输入的扫描输入选择位

1 = 扫描输入

0 = 不扫描输入

bit 9-8

保留: 写入 0; 忽略读操作

bit 7

BUFS: 缓冲区填充状态位

仅当 BUFM = 1 时有效 (ADRES 分隔为 2 个 8 字缓冲区)。

1 = ADC 当前正在填充缓冲区 0x8-0xF, 用户应访问 0x0-0x7 中的数据

0 = ADC 当前正在填充缓冲区 0x0-0x7, 用户应访问 0x8-0xF 中的数据

bit 6

保留: 写入 0; 忽略读操作

寄存器 17-5:	AD1CON2: ADC 控制寄存器 2 (续)
bit 5-2	SMPI<3:0>: 每次中断的采样 / 转换序列数选择位 1111 = 每完成 16 个采样 / 转换序列时产生中断 1110 = 每完成 15 个采样 / 转换序列时产生中断 0001 = 每完成 2 个采样 / 转换序列时产生中断 0000 = 每完成 1 个采样 / 转换序列时产生中断
bit 1	BUFM: ADC 结果缓冲区模式选择位 1 = 缓冲区配置为两个 8 字缓冲区 ADC1BUF(7...0) 和 ADC1BUF(15...8) 0 = 缓冲区配置为一个 16 字缓冲区 ADC1BUF(15...0)
bit 0	ALTS: 交替输入采样模式选择位 1 = 对于第一次采样, 使用 MUX A 输入多路开关设置, 然后对于所有后续采样, 在 MUX B 和 MUX A 输入多路开关设置之间交替 0 = 总是使用 MUX A 输入多路开关设置

寄存器 17-6: AD1CON2CLR: ADC 控制 2 清零寄存器

写入时会将 AD1CON2 中的选定位清零，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 将 AD1CON2 中的选定位清零

在一个或多个位中写入 1 会将 AD1CON2 寄存器中的相应位清零，但不会影响未实现位或只读位。写入 0 不会影响该寄存器。

示例：AD1CON2CLR = 0x00008001 时，会将 AD1CON2 寄存器中的 bit 15 和 bit 0 清零。

寄存器 17-7: AD1CON2SET: ADC 控制 2 置 1 寄存器

写入时会将 AD1CON2 中的选定位置 1，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 将 AD1CON2 中的选定位置 1

在一个或多个位中写入 1 会将 AD1CON2 寄存器中的相应位置 1，但不会影响未实现位或只读位。写入 0 不会影响该寄存器。

示例：AD1CON2SET = 0x00008001 时，会将 AD1CON2 寄存器中的 bit 15 和 bit 0 置 1。

寄存器 17-8: AD1CON2INV: ADC 控制 2 取反寄存器

写入时会将 AD1CON2 中的选定位取反，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 将 AD1CON2 中的选定位取反

在一个或多个位中写入 1 会将 AD1CON2 寄存器中的相应位取反，但不会影响未实现位或只读位。写入 0 不会影响该寄存器。

示例：AD1CON2INV = 0x00008001 时，会将 AD1CON2 寄存器中的 bit 15 和 bit 0 取反。

寄存器 17-9: AD1CON3: ADC 控制寄存器 3

r-X	r-X	r-X	r-X	r-X	r-X	r-X	r-X
—	—	—	—	—	—	—	—
bit 31				bit 24			

r-X	r-X	r-X	r-X	r-X	r-X	r-X	r-X
—	—	—	—	—	—	—	—
bit 23				bit 16			

R/W-0	r-x	r-x	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADRC	—	—	SAMC<4:0>				
bit 15				bit 8			

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W	R/W-0
ADCS<7:0>							
bit 7				bit 0			

图注:

R = 可读位

W = 可写位

P = 可编程位

r = 保留位

U = 未实现位

-n = POR 时的值: (0, 1, x = 未知)

bit 31-16 保留: 写入 0; 忽略读操作

bit 15 **ADRC:** ADC 转换时钟源位

1 = ADC 内部 RC 时钟

0 = 时钟由外设总线时钟 (PBclock) 产生

bit 14-13 保留: 写入 0; 忽略读操作

bit 12-8 **SAMC<4:0>:** 自动采样时间位

11111 = 31 TAD

.....

00001 = 1 TAD

00000 = 0 TAD (不允许)

bit 7-0 **ADCS<7:0>:** ADC 转换时钟选择位11111111 = $TPB \cdot 2 \cdot (ADCS<7:0> + 1) = 512 \cdot TPB = TAD$

.....

00000001 = $TPB \cdot 2 \cdot (ADCS<7:0> + 1) = 4 \cdot TPB = TAD$ 00000000 = $TPB \cdot 2 \cdot (ADCS<7:0> + 1) = 2 \cdot TPB = TAD$

寄存器 17-10: **AD1CON3CLR: ADC 控制 3 清零寄存器**

写入时会将 AD1CON3 中的选定位清零，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 **将 AD1CON3 中的选定位清零**
在一个或多个位中写入 1 会将 AD1CON3 寄存器中的相应位清零，但不会影响未实现位或只读位。写入 0 不会影响该寄存器。
示例：AD1CON3CLR = 0x00008001 时，会将 AD1CON3 寄存器中的 bit 15 和 bit 0 清零。

寄存器 17-11: **AD1CON3SET: ADC 控制 3 置 1 寄存器**

写入时会将 AD1CON3 中的选定位置 1，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 **将 AD1CON3 中的选定位置 1**
在一个或多个位中写入 1 会将 AD1CON3 寄存器中的相应位置 1，但不会影响未实现位或只读位。写入 0 不会影响该寄存器。
示例：AD1CON3SET = 0x00008001 时，会将 AD1CON3 寄存器中的 bit 15 和 bit 0 置 1。

寄存器 17-12: **AD1CON3INV: ADC 控制 3 取反寄存器**

写入时会将 AD1CON3 中的选定位取反，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 **将 AD1CON3 中的选定位取反**
在一个或多个位中写入 1 会将 AD1CON3 寄存器中的相应位取反，但不会影响未实现位或只读位。写入 0 不会影响该寄存器。
示例：AD1CON3INV = 0x00008001 时，会将 AD1CON3 寄存器中的 bit 15 和 bit 0 取反。

寄存器 17-13: AD1CHS: ADC 输入通道选择寄存器

R/W-0	r-x	r-x	r-x	R/W-0	R/W-0	R/W-0	R/W-0
CH0NB	—	—	—	CH0SB<3:0>			
bit 31				bit 24			

R/W-0	r-x	r-x	r-x	R/W-0	R/W-0	R/W-0	R/W-0
CH0NA	—	—	—	CH0SA<3:0>			
bit 23				bit 16			

r-x	r-x	r-x	r-x	r-x	r-x	r-x	r-x
—	—	—	—	—	—	—	—
bit 15				bit 8			

r-x	r-x	r-x	r-x	r-x	r-x	r-x	r-x
—	—	—	—	—	—	—	—
bit 7				bit 0			

图注:

R = 可读位

W = 可写位

P = 可编程位

r = 保留位

U = 未实现位

-n = POR 时的值: (0, 1, x = 未知)

bit 31

CH0NB: MUX B 的反相输入选择位

1 = 通道 0 的反相输入为 AN1

0 = 通道 0 的反相输入为 VR-

bit 30-28

保留: 写入 0; 忽略读操作

bit 27-24

CH0SB<3:0>: MUX B 的同相输入选择位

1111 = 通道 0 的同相输入为 AN15

1110 = 通道 0 的同相输入为 AN14

1101 = 通道 0 的同相输入为 AN13

...

...

...

0001 = 通道 0 的同相输入为 AN1

0000 = 通道 0 的同相输入为 AN0

bit 23

CH0NA: MUX A 多路开关设置的反相输入选择位⁽²⁾

1 = 通道 0 的反相输入为 AN1

0 = 通道 0 的反相输入为 VR-

bit 22-20

保留: 写入 0; 忽略读操作

bit 19-16

CH0SA<3:0>: MUX A 多路开关设置的同相输入选择位

1111 = 通道 0 的同相输入为 AN15

1110 = 通道 0 的同相输入为 AN14

1101 = 通道 0 的同相输入为 AN13

...

...

...

0001 = 通道 0 的同相输入为 AN1

0000 = 通道 0 的同相输入为 AN0

bit 15-0

保留: 写入 0; 忽略读操作

寄存器 17-14: AD1CHSCLR: ADC 输入通道选择清零寄存器

写入时会将 AD1CHS 中的选定位置 1，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 **将 AD1CHS 中的选定位置 1**
在一个或多个位中写入 1 会将 AD1CHS 寄存器中的相应位置 1，但不会影响未实现位或只读位。写入 0 不会影响该寄存器。
示例：AD1CHSCLR = 0x80010000 时，会将 AD1CHS 寄存器中的 bit 15 和 bit 0 清零。

寄存器 17-15: AD1CHSSET: ADC 输入通道选择置 1 寄存器

写入时会将 AD1CHS 中的选定位置 1，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 **将 AD1CHS 中的选定位置 1**
在一个或多个位中写入 1 会将 AD1CHS 寄存器中的相应位置 1，但不会影响未实现位或只读位。写入 0 不会影响该寄存器。
示例：AD1CHSSET = 0x80010000 时，会将 AD1CHS 寄存器中的 bit 15 和 bit 0 清零。

寄存器 17-16: AD1CHSINV: ADC 输入通道选择取反寄存器

写入时会将 AD1CHS 中的选定位置 1，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 **将 AD1CHS 中的选定位置取反**
在一个或多个位中写入 1 会将 AD1CHS 寄存器中的相应位置取反，但不会影响未实现位或只读位。写入 0 不会影响该寄存器。
示例：AD1CHSINV = 0x80010000 时，会将 AD1CHS 寄存器中的 bit 15 和 bit 0 清零。

寄存器 17-17: AD1PCFG: ADC 端口配置寄存器

r-X	r-X	r-X	r-X	r-X	r-X	r-X	r-X
—	—	—	—	—	—	—	—
bit 31				bit 24			

r-X	r-X	r-X	r-X	r-X	r-X	r-X	r-X
—	—	—	—	—	—	—	—
bit 23				bit 16			

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PCFG15	PCFG14	PCFG13	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8
bit 15				bit 8			

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0
bit 7				bit 0			

图注:

R = 可读位

W = 可写位

P = 可编程位

r = 保留位

U = 未实现位

-n = POR 时的值: (0, 1, x = 未知)

bit 31-16 保留: 写入 0; 忽略读操作

bit 15-0 PCFG<15:0>: 模拟输入引脚配置控制位

- 1 = 模拟输入引脚处于数字模式, 使能端口读输入, 该模拟输入的 ADC 输入多路开关输入连接到 AVss
 0 = 模拟输入引脚处于模拟模式, 数字端口读操作将不考虑引脚电压和 ADC 采样引脚电压而返回 1

注: AD1PCFG 寄存器功能会因选定器件上可用 ADC 输入的数量而不同。关于该寄存器的更多详细信息, 请参见具体器件数据手册。

寄存器 17-18: **AD1PCFGCLR: ADC 端口配置清零寄存器**

写入时会将 AD1PCFG 中的选定位清零，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 **将 AD1PCFG 中的选定位清零**
在一个或多个位中写入 1 会将 AD1PCFG 寄存器中的相应位清零，但不会影响未实现位或只读位。写入 0 不会影响该寄存器。
示例：AD1PCFGCLR = 0x00008001 时，会将 AD1PCFG 寄存器中的 bit 15 和 bit 0 清零。

寄存器 17-19: **AD1PCFGSET: ADC 端口配置置 1 寄存器**

写入时会将 AD1PCFG 中的选定位置 1，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 **将 AD1PCFG 中的选定位置 1**
在一个或多个位中写入 1 会将 AD1PCFG 寄存器中的相应位置 1，但不会影响未实现位或只读位。写入 0 不会影响该寄存器。
示例：AD1PCFGSET = 0x00008001 时，会将 AD1PCFG 寄存器中的 bit 15 和 bit 0 置 1。

寄存器 17-20: **AD1PCFGINV: ADC 端口配置取反寄存器**

写入时会将 AD1PCFG 中的选定位取反，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 **将 AD1PCFG 中的选定位取反**
在一个或多个位中写入 1 会将 AD1PCFG 寄存器中的相应位取反，但不会影响未实现位或只读位。写入 0 不会影响该寄存器。
示例：AD1PCFGINV = 0x00008001 时，会将 AD1PCFG 寄存器中的 bit 15 和 bit 0 取反。

寄存器 17-21: AD1CSSL: ADC 输入扫描选择寄存器

r-X	r-X	r-X	r-X	r-X	r-X	r-X	r-X
—	—	—	—	—	—	—	—
bit 31				bit 24			
r-X	r-X	r-X	r-X	r-X	r-X	r-X	r-X
—	—	—	—	—	—	—	—
bit 23				bit 16			
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CSSL15	CSSL14	CSSL13	CSSL12	CSSL11	CSSL10	CSSL9	CSSL8
bit 15				bit 8			
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CSSL7	CSSL6	CSSL5	CSSL4	CSSL3	CSSL2	CSSL1	CSSL0
bit 7				bit 0			

图注:
R = 可读位 W = 可写位 P = 可编程位 r = 保留位
U = 未实现位 -n = POR 时的值: (0, 1, x = 未知)

bit 31-16 保留: 写入 0; 忽略读操作
bit 15-0 **CSSL<15:0>**: ADC 输入引脚扫描选择位
 1 = 选择对 ANx 进行输入扫描
 0 = 输入扫描时跳过 ANx

注: AD1CSSL 寄存器功能会因选定器件上可用 ADC 输入的数量而不同。关于该寄存器的更多详细信息, 请参见具体器件数据手册。

寄存器 17-22: AD1CSSLCLR: ADC 输入扫描选择清零寄存器

写入时会将 AD1CSSL 中的选定位清零，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 将 AD1CSSL 中的选定位清零

在一个或多个位中写入 1 会将 AD1CSSL 寄存器中的相应位清零，但不会影响未实现位或只读位。写入 0 不会影响该寄存器。

示例：AD1CSSLCLR = 0x00008001 时，会将 AD1CSSL 寄存器中的 bit 15 和 bit 0 清零。

寄存器 17-23: AD1CSSLSET: ADC 输入扫描选择置 1 寄存器

写入时会将 AD1CSSL 中的选定位置 1，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 将 AD1CSSL 中的选定位置 1

在一个或多个位中写入 1 会将 AD1CSSL 寄存器中的相应位置 1，但不会影响未实现位或只读位。写入 0 不会影响该寄存器。

示例：AD1CSSLSET = 0x00008001 时，会将 AD1CSSL 寄存器中的 bit 15 和 bit 0 置 1。

寄存器 17-24: AD1CSSLINV: ADC 输入扫描选择取反寄存器

写入时会将 AD1CSSL 中的选定位取反，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 将 AD1CSSL 中的选定位取反

在一个或多个位中写入 1 会将 AD1CSSL 寄存器中的相应位取反，但不会影响未实现位或只读位。写入 0 不会影响该寄存器。

示例：AD1CSSLINV = 0x00008001 时，会将 AD1CSSL 寄存器中的 bit 15 和 bit 0 取反。

寄存器 17-25: IFS1: 中断标志状态寄存器 1⁽¹⁾

r-x	r-x	r-x	r-x	r-x	r-x	R/W-0	R/W-0
—	—	—	—	—	—	USBIF	FCEIF
bit 31						bit 24	
r-x	r-x	r-x	r-x	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	DMA3IF	DMA2IF	DMA1IF	DMA0IF
bit 23						bit 16	
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
RTCCIF	FSCMIF	I2C2MIF	I2C2SIF	I2C2BIF	U2TXIF	U2RXIF	U2EIF
bit 15						bit 8	
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SPI2RXIF	SPI2TXIF	SPI2EIF	CMP2IF	CMP1IF	PMPIF	AD1IF	CNIF
bit 7						bit 0	

图注:

R = 可读位	W = 可写位	P = 可编程位	r = 保留位
U = 未实现位	-n = POR 时的值: (0, 1, x = 未知)		

bit 1 **AD1IF:** 模数转换器 1 中断请求标志位
1 = 产生了中断请求
0 = 未产生中断请求

注 1: 该中断寄存器中的阴影位名称用于控制其他 PIC32MX 外设，与 ADC 无关。

寄存器 17-26: IEC1: 中断允许控制寄存器 1⁽¹⁾

r-x	r-x	r-x	r-x	r-x	r-x	R/W-0	R/W-0
—	—	—	—	—	—	USBIE	FCEIE
bit 31						bit 24	
r-x	r-x	r-x	r-x	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	DMA3IE	DMA2IE	DMA1IE	DMA0IE
bit 23						bit 16	
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
RTCCIE	FSCMIE	I2C2MIE	I2C2SIE	I2C2BIE	U2TXIE	U2RXIE	U2EIE
bit 15						bit 8	
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SPI2RXIE	SPI2TXIE	SPI2EIE	CMP2IE	CMP1IE	PMPIE	AD1IE	I2C1MIE
bit 7						bit 0	

图注:			
R = 可读位	W = 可写位	P = 可编程位	r = 保留位
U = 未实现位	-n = POR 时的值: (0, 1, x = 未知)		

bit 1 **AD1IE:** 模数转换器 1 中断允许位
1 = 允许中断
0 = 禁止中断

注 1: 该中断寄存器中的阴影位名称用于控制其他 PIC32MX 外设，与 ADC 无关。

寄存器 17-27: IPC6: 中断优先级控制寄存器 6⁽¹⁾

r-x	r-x	r-x	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	AD1IP<2:0>			AD1IS<1:0>	
bit 31			bit 24				

r-x	r-x	r-x	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	CNIP<2:0>			CNIS<1:0>	
bit 23			bit 16				

r-x	r-x	r-x	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	I2C1IP<2:0>			I2C1IS<1:0>	
bit 15			bit 8				

r-x	r-x	r-x	R/W-0	R/W-0	R/W-0	R/W	R/W-0
—	—	—	U1IP<2:0>			U1IS<1:0>	
bit 7			bit 0				

图注:

R = 可读位

W = 可写位

P = 可编程位

r = 保留位

U = 未实现位

-n = POR 时的值: (0, 1, x = 未知)

bit 28 - 26 **AD1IP<2:0>**: 模数转换器 1 中断优先级位

111 = 中断优先级为 7

110 = 中断优先级为 6

101 = 中断优先级为 5

100 = 中断优先级为 4

011 = 中断优先级为 3

010 = 中断优先级为 2

001 = 中断优先级为 1

000 = 禁止中断

bit 25-24 **AD1IS<1:0>**: 模数转换器 1 中断子优先级位

11 = 中断子优先级为 3

10 = 中断子优先级为 2

01 = 中断子优先级为 1

00 = 中断子优先级为 0

注 1: 该中断寄存器中的阴影位名称用于控制其他 PIC32MX 外设, 与 ADC 无关。

17.3 ADC 工作原理、术语和转换序列

本节将介绍 A/D 转换器的工作原理、配置转换器所需的步骤，介绍模块的特殊功能，并提供 ADC 配置示例及显示转换器预期输出的时序图与图表。

17.3.1 工作原理概述

模拟采样包含两个步骤：采集和转换（见图 17-2）。在采集期间，模拟输入引脚将与采样 / 保持放大器（SHA）连接。对引脚进行足够时间的采样之后，采样电压将等于输入电压，引脚会与 SHA 断开，以便为转换过程提供稳定的输入电压。然后，转换过程会将模拟采样电压转换为二进制表示形式。

图 17-1 给出了 ADC 的概要图。10 位 A/D 转换器具有单个 SHA。SHA 通过模拟输入 MUX（MUX A 和 MUX B）与模拟输入引脚连接。模拟输入 MUX 通过 AD1CHS 寄存器进行控制。AD1CHS 寄存器中有两组 MUX 控制位。通过这两组控制位，可以独立地控制两个不同的模拟输入。在两次转换之间，A/D 转换器可选择在 MUX A 和 MUX B 配置之间切换。A/D 转换器还可以选择使用单个 MUX 逐个扫描一系列模拟输入。

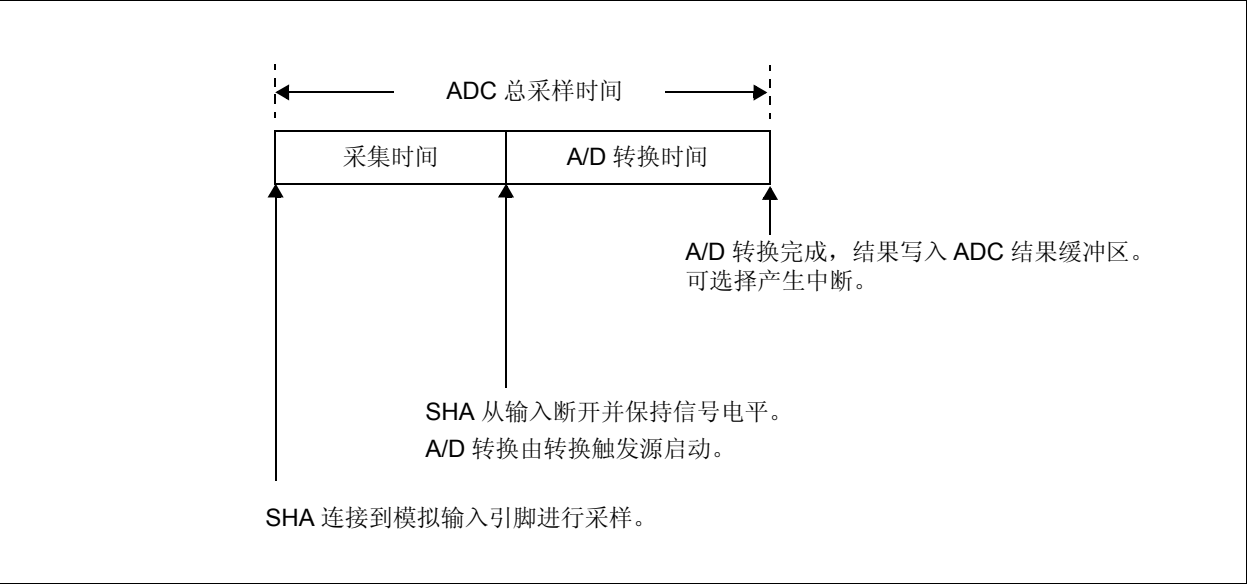
采集时间可以手动或自动控制。在用户软件中，采集时间可以通过将 SAMP 位（AD1CON1<1>）置 1 来手动启动，并通过清零 SAMP 来手动结束。采集时间也可以通过 A/D 转换器硬件自动启动，并通过转换触发源自动结束。采集时间通过 SAMC 位（AD1CON3<12:8>）进行设置。SHA 对于采集周期存在一个最小时间的要求。关于采集时间规范，请参见器件数据手册。

转换时间是 A/D 转换器转换 SHA 保持电压所需的时间。除了两个时钟周期外，A/D 转换器还需要一个 ADC 时钟周期（TAD）来转换结果中的每个位。因此，执行完整的转换总共需要 12 个 TAD 周期。当转换时间结束时，结果写入 16 个 ADC 结果寄存器（ADC1BUF0...ADC1BUFF）之一。

采集时间和 A/D 转换时间之和就是总采样时间（见图 17-2）。A/D 转换器具有多个输入时钟选项，用于产生 TAD 时钟。用户必须选择不会违反最小 TAD 规范的输入时钟选项。

采样过程可以定期执行一次，也可以基于模块配置定义的触发源。

图 17-2: ADC 采样 / 转换序列



采样开始时间可通过软件将 SAMP 控制位置 1。采样开始时间还可以由硬件自动控制。当 A/D 转换器工作在自动采样模式时，在采样/转换序列中转换结束后，SHA 会重新连接到模拟输入引脚。自动采样功能由 ASAM 控制位（AD1CON1<2>）控制。

转换触发源结束采样时间并开始 A/D 转换或采样/转换序列。转换触发源通过控制位 SSRC<2:0>（AD1CON1<7:5>）选择。转换触发源可从多种硬件源中选择，或通过在软件中将 SAMP 控制位清零来手动控制。转换触发源之一是自动转换。自动转换之间的时间通过计数器和 ADC 时钟设置。自动采样模式和自动转换触发可以一起使用，提供无需软件干预的无限自动转换功能。

在每个或多个采样序列结束时都可产生中断，这由 SMPI<3:0>（AD1CON2<5:2>）的值决定。中断之间的采样序列数在 1 至 16 之间变化。用户应当注意 A/D 转换缓冲区将保存单个转换序列的结果。即使前一个序列中的采样数小于 16，下一个序列也会从顶部开始填充缓冲区。两次中断之间的转换结果总数为 SMPI 的值。两次中断之间的转换总数不能超出物理缓冲区长度。

17.4 ADC 模块配置

ADC 模块的操作通过相应寄存器中的位设置指定。以下信息概要介绍了这些操作和设置。后面几节中提供了每个配置步骤的选项和详细信息。

1. 要配置 ADC 模块，请执行以下步骤：

- A-1. 在 AD1PCFG<15:0> 中配置模拟端口引脚，如第 17.4.1 节“配置模拟端口引脚”中所述。
- B-1. 在 AD1CHS<32:0> 中选择 ADC MUX 的模拟输入，如第 17.4.2 节“选择 ADC MUX 的模拟输入”中所述。
- C-1. 使用 FORM<2:0> (AD1CON1<10:8>) 选择 ADC 结果格式，如第 17.4.3 节“选择 ADC 结果格式”中所述。
- C-2. 使用 SSRC<2:0> (AD1CON1<7:5>) 选择采样时钟源，如第 17.4.4 节“选择采样时钟源”中所述。
- D-1. 使用 VCFG<2:0> (AD1CON2<15:13>) 选择参考电压源，如第 17.4.7 节“选择参考电压源”中所述。
- D-2. 使用 CSCNA (AD1CON2<10>) 选择扫描模式，如第 17.4.8 节“选择扫描模式”中所述。
- D-3. 如果要使用中断，则设置每个中断的转换数 SMP<3:0> (AD1CON2<5:2>)，如第 17.4.9 节“设置每次中断的转换数”中所述。
- D-4. 使用 BUFM (AD1CON2<1>) 设置缓冲区填充模式，如第 17.4.10 节“缓冲区填充模式”中所述。
- D-5. 在 ALTS AD1CON2<0> 中选择要与 ADC 连接的 MUX，如第 17.4.11 节“选择与 ADC 连接的 MUX (交替采样模式)”中所述。
- E-1. 使用 ADRC (AD1CON3<15>) 选择 ADC 时钟源，如第 17.4.12 节“选择 ADC 转换时钟源和预分频比”中所述。
- E-2. 如果要使用自动转换，则使用 SAMC<4:0> (AD1CON3<12:8>) 选择采样时间，如第 17.4.13 节“采集时间的注意事项”中所述。
- E-3. 使用 ADCS<7:0> (AD1CON3<7:0>) 选择 ADC 时钟预分频比，如第 17.4.12 节“选择 ADC 转换时钟源和预分频比”中所述。
- F. 使用 AD1CON1<15> 开启 ADC 模块，如第 17.4.14 节“开启 ADC”中所述。

注： 上面的步骤 A 至 E 可以按任意顺序执行，但在任何情况下，步骤 F 都必须为最后一个步骤。

2. 配置 ADC 中断 (如果需要)。

- A-1. 清零 AD1IF 位 (IFS1<1>)，如第 17.7 节“中断”中所述。
- A-2. 如果要使用中断，则选择 ADC 中断优先级 AD1IP<2:0> (IPC<28:26>) 和子优先级 AD1IS<1:0> (IPC<24:24>)，如第 17.7 节“中断”中所述。

3. 通过启动采样来启动转换序列，如第 17.4.15 节“启动采样”中所述。

17.4.1 配置模拟端口引脚

AD1PCFG 寄存器和 TRISB 寄存器用于控制 ADC 端口引脚的操作。

AD1PCFG 用于指定要用作模拟输入的器件引脚的配置。当相应的 PCFGn 位 (AD1PCFG<n>) = 0 时, 引脚被配置为模拟输入。当该位为 1 时, 引脚被设置为数字控制引脚。当引脚被配置为模拟输入时, 相关的端口 I/O 数字输入缓冲器被禁止, 因此不消耗电流。AD1PCFG 寄存器在复位时清零, 在默认情况下, 这使得 ADC 输入引脚复位时被配置为模拟输入。

TRIS 寄存器用于控制端口引脚的数字功能。对于希望作为模拟输入的端口引脚, 其对应的 TRIS 位必须置 1, 从而将该引脚指定为输入。如果与 ADC 输入关联的 I/O 引脚被配置为输出, 则 TRIS 位被清零, 且端口的数字输出电平 (VOH 或 VOL) 将被转换。在器件复位后, 所有 TRIS 位均置 1。

注: 如果某个 PORT 寄存器与 ADC 共用引脚, 则读取 PORT 锁存器时, 配置为模拟输入的任何引脚都读为 0。
任何定义为数字输入 (包括 AN15:AN0 引脚)、但未配置为模拟输入的引脚上的模拟电平都可能导致输入缓冲器消耗的电流超出器件规范。

17.4.2 选择 ADC MUX 的模拟输入

AD1CHS 寄存器用于选择要与 MUX A 和 MUX B 连接的模拟输入引脚。每个 MUX 都具有两种输入, 称为同相输入和反相输入。MUX A 的同相输入通过 CH0SA<4:0> 进行控制, 反相输入通过 CH0NA 进行控制。MUX B 的同相输入通过 CH0SB<4:0> 进行控制, 反相输入通过 CH0NB 进行控制。

同相输入可以从任意可用模拟输入引脚中进行选择。反相输入可以选择 ADC 负参考电压或 AN1。使用 AN1 作为反相输入时, ADC 可以在单极性差分模式下使用。关于用作负参考电压时的 AN1 输入电压限制, 请参见器件数据手册。

注: 使用扫描模式时, CH0SA<4:0> 可能会被改写。更多信息, 请参见第 17.4.8 节“选择扫描模式”。

17.4.3 选择 ADC 结果格式

ADC 结果寄存器中的数据可以使用 8 种格式读取。格式通过 FORM<2:0> (AD1CON1<10:8>) 进行控制。用户可以选择整数、有符号整数、小数或有符号小数作为 16 位或 32 位结果。图 17-3 和图 17-4 显示了各种格式下的结果。表 17-2 和表 17-3 给出了 32 位和 16 位结果选择使用 4 种格式时的结果示例。

注: 32 位和 16 位模式之间没有数字差异。在 32 位模式下, 会对全部 32 位应用符号扩展; 而在 16 位模式下, 只会对结果的低 16 位应用符号扩展。

图 17-3: ADC 输出数据格式，32 位模式

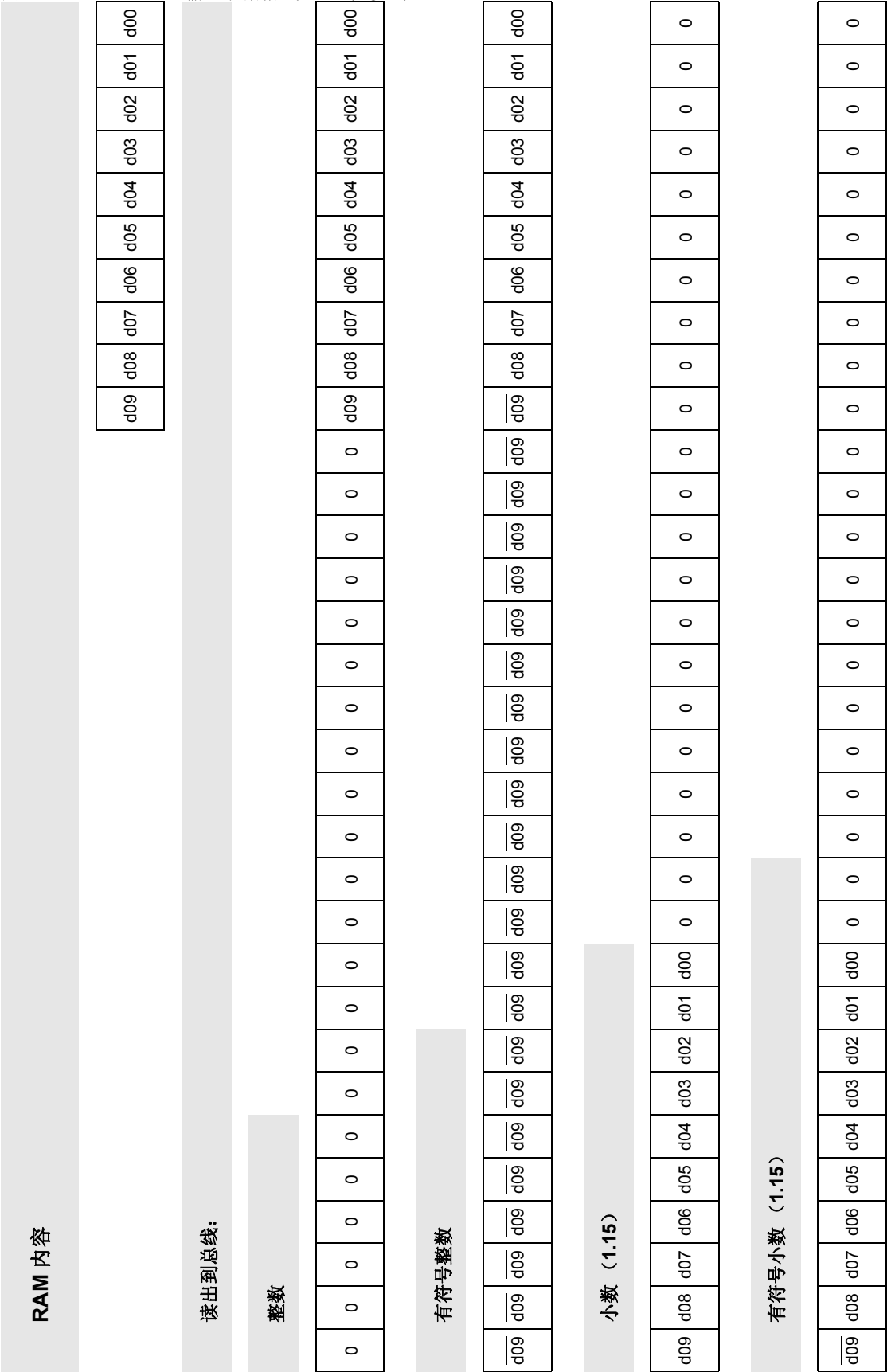


图 17-4: ADC 输出数据格式，16 位模式

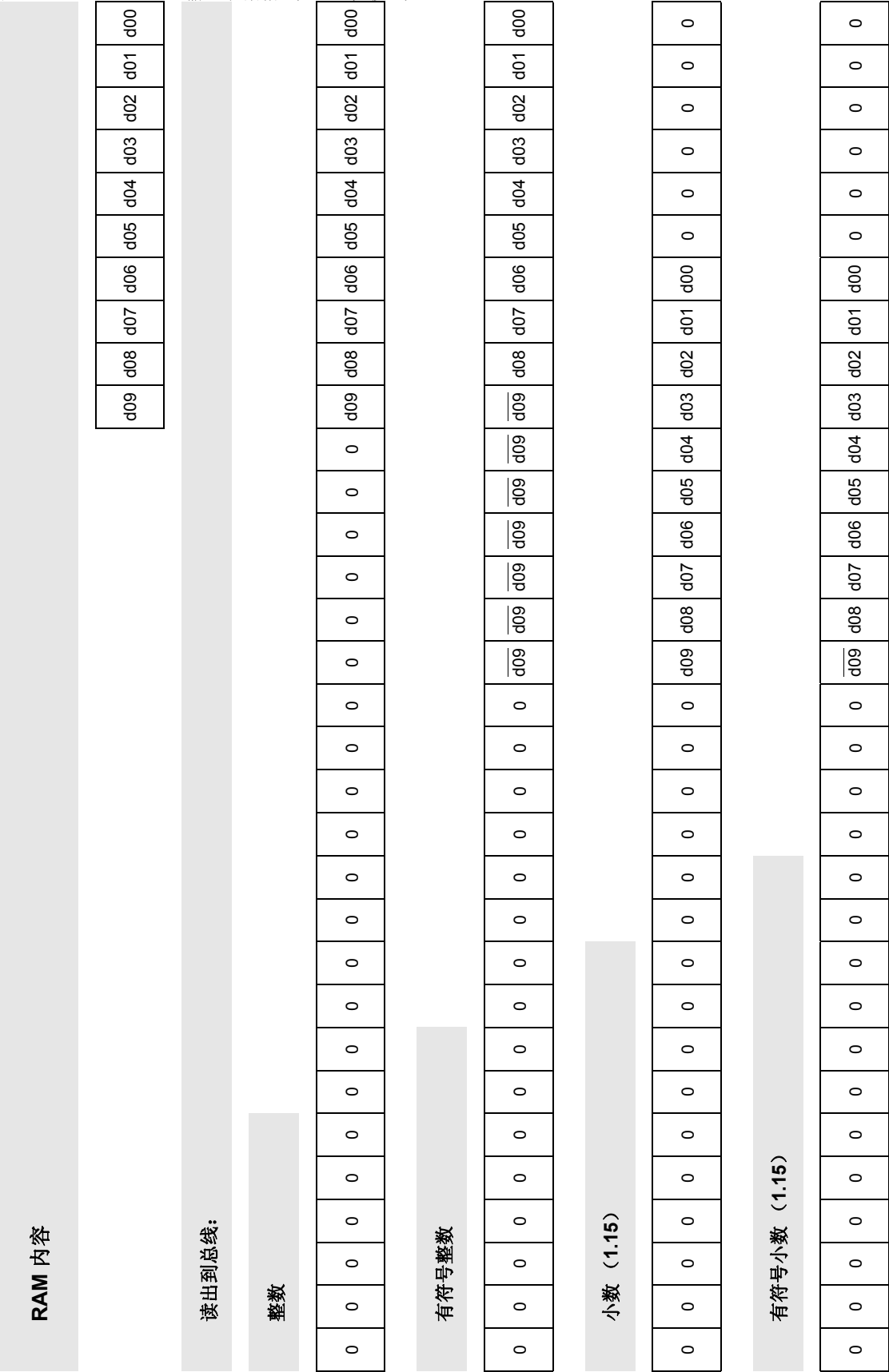


表 17-2: FORM<2> (AD1CON1 <10>) = 1 (32 位结果) 时, 选定结果代码的等价数值

Vin/ VR	10 位 输出码	32 位整数格式	32 位有符号 整数格式	32 位小数格式	32 位有符号 小数格式
1023/1024	11 1111 1111	0000 0000 0000 0000 0000 0011 1111 1111 = 1023	0000 0000 0000 0000 0000 0001 1111 1111 = 511	1111 1111 1100 0000 0000 0000 0000 0000 = 0.999	0111 1111 1100 0000 0000 0000 0000 0000 = 0.499
1022/1024	11 1111 1110	0000 0000 0000 0000 0000 0011 1111 1110 = 1022	0000 0000 0000 0000 0000 0001 1111 1110 = 510	1111 1111 1000 0000 0000 0000 0000 0000 = 0.998	0111 1111 1000 0000 0000 0000 0000 0000 = 0.498
...					
513/1024	10 0000 0001	0000 0000 0000 0000 0000 0010 0000 0001 = 513	0000 0000 0000 0000 0000 0000 0000 0001 = 1	1000 0000 0100 0000 0000 0000 0000 0000 = 0.501	0 000 0000 0100 0000 0000 0000 0000 0000 = 0.001
512/1024	10 0000 0000	0000 0000 0000 0000 0000 0010 0000 0000 = 512	0000 0000 0000 0000 0000 0000 0000 0000 = 0	1000 0000 0000 0000 0000 0000 0000 0000 = 0.500	0000 0000 0000 0000 0000 0000 0000 0000 = 0.000
511/1024	01 1111 1111	0000 0000 0000 0000 0000 0001 1111 1111 = 511	1111 1111 1111 1111 1111 1111 1111 1111 = -1	0111 1111 1100 0000 0000 0000 0000 0000 = .499	1111 1111 1100 0000 0000 0000 0000 0000 = -0.001
...					
1/1024	00 0000 0001	0000 0000 0000 0000 0000 0000 0000 0001 = 1	1111 1111 1111 1111 1111 1110 0000 0001 = -511	0000 0000 0100 0000 0000 0000 0000 0000 = 0.001	1000 0000 0100 0000 0000 0000 0000 0000 = -0.499
0/1024	00 0000 0000	0000 0000 0000 0000 0000 0000 0000 0000 = 0	1111 1111 1111 1111 1111 1110 0000 0000 = -512	0000 0000 0000 0000 0000 0000 0000 0000 = 0.000	1000 0000 0000 0000 0000 0000 0000 0000 = -0.500

表 17-3: FORM<2> (AD1CON1 <10>) = 0 (16 位结果) 时, 选定结果代码的等价数值

Vin/ VR	10 位 输出码	16 位整数格式	16 位有符号 整数格式	16 位小数格式	16 位有符号 小数格式
1023/1024	11 1111 1111	0000 0011 1111 1111 = 1023	0000 0001 1111 1111 = 511	1111 1111 1100 0000 = 0.999	0111 1111 1100 0000 = 0.499
1022/1024	11 1111 1110	0000 0011 1111 1110 = 1022	0000 0001 1111 1110 = 510	1111 1111 1000 0000 = 0.998	0111 1111 1000 0000 = 0.498
...					
513/1024	10 0000 0001	0000 0010 0000 0001 = 513	0000 0000 0000 0001 = 1	1000 0000 0100 0000 = 0.501	0 000 0000 0100 0000 = 0.001
512/1024	10 0000 0000	0000 0010 0000 0000 = 512	0000 0000 0000 0000 = 0	1000 0000 0000 0000 = 0.500	0000 0000 0000 0000 = 0.000
511/1024	01 1111 1111	0000 0001 1111 1111 = 511	1111 1111 1111 1111 = -1	0111 1111 1100 0000 = .499	1111 1111 1100 0000 = -0.001
...					
1/1024	00 0000 0001	0000 0000 0000 0001 = 1	1111 1110 0000 0001 = -511	0000 0000 0100 0000 = 0.001	1000 0000 0100 0000 = -0.499
0/1024	00 0000 0000	0000 0000 0000 0000 = 0	1111 1110 0000 0000 = -512	0000 0000 0000 0000 = 0.000	1000 0000 0000 0000 = -0.500

17.4.4 选择采样时钟源

通常需要将采样结束和转换启动与某个其他时间事件同步。ADC 模块可以使用 4 个触发源之一作为转换触发信号。转换触发源的选择通过 SSRC<2:0> (AD1CON1<7:5>) 位控制。

17.4.4.1 手动转换

要将 ADC 配置为在 SAMP 清零时结束采样并启动转换，则需要设置 SSRC 为 000。

17.4.4.2 定时器比较触发

通过设置 SSRC<2:0> = 010 将 ADC 配置为该触发模式。当 32 位定时器 TMR3/TMR2 或 16 位 Timer3 发生周期匹配时，Timer3 会产生一个特殊的 A/D 转换器触发事件信号。TMR5/TMR4 定时器对或 Timer3 之外的其他 16 位定时器不存在该功能。更多详细信息，请参见第 14 章“定时器”。

17.4.4.2.1 外部 INT0 引脚触发

要将 ADC 配置为在 INT0 引脚上发生有效电平跳变时启动转换，SSRC<2:0> 需要设置为 001。INT0 引脚可以设定为上升沿输入或下降沿输入以触发转换过程。

17.4.4.2.2 自动转换

ADC 可以配置为按照自动采样时间位 SAMC<4:0> 选择的速率自动执行转换。通过设置 SSRC<2:0> = 111 将 ADC 配置为该触发模式。在该模式下，ADC 会对选定通道执行连续转换。

17.4.5 将 ADC 操作与内部或外部事件同步

外部事件触发脉冲结束采样和启动转换的模式 (SSRC2:SSRC0 = 001、010 或 011) 可以与自动采样 (ASAM = 1) 结合使用，使 ADC 将采样转换事件与触发脉冲源同步。例如，在图 17-13 中 (其中，SSRC = 010 且 ASAM = 1)，ADC 的结束采样和启动转换总是与定时器比较触发事件同步。ADC 有与定时器比较事件速率对应的采样转换速率。代码示例请参见例 17-5。

17.4.6 选择自动或手动采样

采样可以手动启动，也可以在前一个转换完成时自动启动。

17.4.6.1 手动

清零 ASAM (AD1CON1<2>) 位会禁止自动采样模式。采集会在软件将 SAMP (AD1CON1<1>) 位置 1 时开始。只有 SAMP 位再次置 1 之后，才会继续采集。示例请参见图 17-8。

17.4.6.2 自动

将 ASAM (AD1CON1<2>) 位置 1 会使能自动采样模式。在该模式下，前一个采样转换之后，会自动开始新的采样。示例请参见图 17-9。

17.4.7 选择参考电压源

用户可以选择 ADC 模块的参考电压。参考电压可以是内部或外部电压。

VCFG<2:0> 控制位 (AD1CON2<15:13>) 用于选择 A/D 转换的参考电压。参考电压高电压 (VR+) 和参考电压低电压 (VR-) 可作为内部 AVDD 和 AVSS 电压轨, 也可作为 VREF+ 和 VREF- 输入引脚电压。外部 ADC 参考电压可用于降低转换器中的噪声。

在低引脚数器件上, 外部参考电压引脚可与 AN0 和 AN1 输入共用。当这些引脚与 VREF+ 和 VREF- 输入引脚共用时, A/D 转换器仍然可以在这些引脚上执行转换。

加到外部参考电压引脚上的电压必须符合特定的规范。关于电气规范, 请参见器件数据手册的电气规范章节。

注: 对于高电压转换, 必须选择外部参考电压 VREF+ 和 VREF-。更多信息, 请参见数据手册。

外部 VREF+ 和 VREF- 可以与其他模拟外设共用。更多信息, 请参见器件数据手册。

17.4.8 选择扫描模式

ADC 模块能够逐个扫描一组选定输入。CSCNA 位 (AD1CON2<10>) 用于允许在选定数量的模拟输入之间扫描 MUX A 输入。

17.4.8.1 扫描模式使能

扫描模式通过将 CSCNA (AD1CON2<10>) 置 1 来使能。使能扫描模式时, MUX A 的同相输入由 AD1CSSL 寄存器内容控制。AD1CSSL 寄存器中的每个位对应于一个模拟输入。bit 0 对应 AN0, bit 1 对应 AN1, 依此类推。如果 AD1CSSL 寄存器中的某个位为 1, 则在扫描序列中将扫描对应的输入。输入扫描总是在每次发生中断后从第一个选定通道开始, 从编号较低的输入扫描到编号较高的输入。当使能扫描模式时, CH0SA<3:0> 位被忽略。

注: 如果所选的扫描输入个数大于每次中断的采样数, 则编号较高的输入将不会被采样。AD1CSSL 位仅指定通道同相输入的输入源。CH0NA 位用于选择扫描时通道反相输入的输入源。

17.4.8.2 扫描模式禁止

当 CSCNA = 0 时, 扫描模式会被禁止, MUX A 的同相输入由 CH0SA<3:0> 控制。

17.4.8.3 组合使用扫描和交替模式

通过将扫描和交替模式组合在一起使用, 可以对一组输入进行扫描, 并且每隔一个采样转换单个输入。

该模式的使能方式是设置 CSCNA 位 = 1, 并设置 ALTS (AD1CON2<0>) 位 = 1。

CSCNA 位用于使能对 MUX A 进行扫描, CH0SB<3:0> (AD1CHS<27:24>) 和 CH0NB (AD1CHS<31>) 用于配置 MUX B 的输入。扫描仅应用于 MUX A 输入选择。MUX B 输入选择 (由 CH0SB<3:0> 指定) 将仍选择单个输入。

以下序列是 3 个扫描通道 (MUX A) 和单个固定通道 (MUX B) 的示例:

1. 对扫描列表中的第 1 个输入进行采样。
2. 对由 CH0SB<3:0> 和 CH0NB 选择的输入进行采样。
3. 对扫描列表中的第 2 个输入进行采样。
4. 对由 CH0SB<3:0> 和 CH0NB 选择的输入进行采样。
5. 对扫描列表中的第 3 个输入进行采样。
6. 对由 CH0SB<3:0> 和 CH0NB 选择的输入进行采样。

重复该过程。

17.4.9 设置每次中断的转换数

SMPI<3:0> 位 (AD1CON2<5:2>) 用于选择在产生 CPU 中断之前发生的 A/D 转换数。它同时也定义将在使用 ADC1BUF0 (对于双缓冲区模式为 ADC1BUF0 或 ADC1BUF8) 指定的结果缓冲区中写入的单元数量。它的取值范围为 1 至 16 个采样 (对于双缓冲区模式为 1 至 8 个采样)。在产生中断之后, 采样序列会重新开始; 第一个采样的结果会被写入第一个缓冲单元。

例如, 如果 SMPI<3:0> = 0000, 则转换结果将总是被写入 ADC1BUF0。在该示例中, 将不使用任何其他缓冲单元。

例如, 如果 SMPI<3:0> = 1110, 则将会转换 15 个采样, 并且结果存储到缓冲单元 ADC1BUF0 至 ADC1BUFE。在写入 ADC1BUFE 之后, 将会产生中断。下一个采样会被写入 ADC1BUF0。在该示例中, 将不使用 ADC1BUFF。

结果寄存器中的数据会被下一个采样序列覆盖。产生中断之后, 必须在第一个采样完成之前读取结果缓冲区中的数据。缓冲区填充模式可用于提高中断产生和数据覆盖之间的时间。请参见“缓冲区填充模式”章节。

当 BUFM 位 (AD1CON2<1>) 为 1 时, 用户不能将采样数和 SMPI 位的组合设定为导致在两次中断之间产生超出 16 次转换的设置; 当 BUFM 位 (AD1CON2<1>) 为 0 时, 不能设定为产生超出 8 次转换的设置。尝试产生采样数超出 16 个的转换列表时, 采样序列会被截短为 16 个采样。

17.4.10 缓冲区填充模式

缓冲区填充模式允许输出缓冲区用作单个 16 字缓冲区或两个 8 字缓冲区。

当 BUFM 为 0 时, 对于所有转换序列都使用完整的 16 字缓冲区。转换结果将从 ADC1BUF0 开始, 按顺序写入缓冲区, 直到达到 SMPI<3:0> (AD1CON2<5:2>) 所定义的采样数为止。下一个转换结果将被写入 ADC1BUF0, 并且该过程一直重复。如果允许 ADC 中断, 则在缓冲区中的采样数等于 SMPI<3:0> 时, 会产生中断。

当 BUFM 位 (AD1CON2<1>) 为 1 时, 16 字结果缓冲区 (ADRES) 被分隔为两个 8 字缓冲区。转换结果将从 ADC1BUF0 开始按顺序写入第一个缓冲区, BUFS (AD1CON2<7>) 将会清零, 直到达到 SMPI<3:0> (AD1CON2<5:2>) 所定义的采样数为止。然后, ADC 中断标志将会置 1。

在 ADC 中断标志置 1 之后, 接下来的结果将从 ADC1BUF8 开始按顺序写入第二个缓冲区。下一个转换结果将从 ADC1BUF8 开始写入第二个缓冲区, BUFS (AD1CON2<7>) 将会置 1, 直到达到 SMPI<3:0> (AD1CON2<5:2>) 所定义的采样数为止。然后, ADC 中断标志将会置 1。

然后, BUFS = 0, 该过程重新开始, 结果将被写入第一个缓冲区。

具体使用哪一种缓冲区填充模式取决于在发生 A/D 中断之后有多少时间可用于移动缓冲区内容, 以及由应用程序决定的中断响应延时。如果处理器可在对一个通道进行采样和转换的时间内卸空一个已满的缓冲区, 则 BUFM 位可以为 0, 且每次中断最多可进行 16 次转换。在第一个缓冲单元被覆盖之前, 处理器具有一个采集 + 转换的时间周期。

如果处理器无法在采样 + 转换时间内卸空缓冲区, 则应使用双缓冲区模式 BUFM 位 = 1 来防止覆盖结果数据。例如, 如果 SMPI<3:0> = 0111, 则会向第一个缓冲区中写入 8 个转换数据, 之后发生中断。接下来的 8 个转换数据会被写入第二个缓冲区。因此, 处理器将可以使用中断之间的全部时间, 以从缓冲区读出 8 个转换结果。

17.4.11 选择与 ADC 连接的 MUX（交替采样模式）

ADC 具有两个与 SHA 连接的 MUX。这两个 MUX 用于选择对哪个模拟输入进行采样。每个 MUX 都具有同相输入和反相输入（见图 17-5 和图 17-6）。

注： 模拟输入的数量会随器件而不同。请对照相应器件数据手册验证模拟输入的可用性。

17.4.11.1 单输入选择

用户可以在最多 16 个模拟输入（由器件上的模拟通道数量决定）中选择一个作为 SHA 的同相输入。CH0SA<3:0> 位（AD1CHS<19:16>）用于选择同相模拟输入。

用户可以选择 VR- 或 AN1 作为反相输入。CH0NA 位（AD1CHS<23>）用于选择通道 0 反相输入的模拟输入。使用 AN1 作为反相输入时，将可以进行单极性差分测量。

对于该工作模式，ALTS 位（AD1CON2<0>）必须清零。

17.4.11.2 交替输入选择

ALTS 位会使模块在两个输入 MUX 之间交替。

由 CH0SA<3:0> 和 CH0NA 指定的输入称为 MUX A 输入。由 CH0SB<3:0> 和 CH0NB 指定的输入称为 MUX B 输入。

当 ALTS 为 1 时，模块将交替采样，即在前一次采样 MUX A 输入，而在下一次采样 MUX B 输入。当 ALTS 为 0 时，只会选择对 CH0SA<3:0> 和 CH0NA 指定的输入进行采样。

例如，如果在第一次采样 / 转换序列时 ALTS 为 1，则选择对 CH0SA<3:0> 和 CH0NA 指定的输入进行采样。在下一次采样时，选择对 CH0SB<3:0> 和 CH0NB 指定的输入进行采样。然后，该模式一直重复。

17.4.12 选择 ADC 转换时钟源和预分频比

ADC 模块可以使用内部 RC 振荡器或 PBCLK 作为转换时钟源。

使用内部 RC 振荡器作为时钟源（ADRC（AD1CON3<15>）= 1）时，TAD 为振荡器周期（不使用预分频比）。使用内部振荡器时，ADC 可以在 SLEEP（休眠）和 IDLE（空闲）模式下继续工作。

注： ADRC 旨在用于 Sleep（休眠）模式下的 ADC 操作，它不是校准时钟。要求精确 ADC 采集时序的应用应对 ADC 使用稳定的校准时钟源。

使用 PBCLK 作为转换时钟源（ADRC = 0）时，TAD 是应用预分频比 ADCS<7:0>（AD1CON3<7:0>）之后的 PBCLK 周期。

A/D 转换器存在一个最高的转换速率。模拟模块时钟 TAD 用于控制转换时序。A/D 转换需要 12 个时钟周期（12 TAD）。

ADC 转换时钟的周期通过一个 8 位计数器用软件选择。TAD 存在 256 种可能选项，它们通过 ADCS<7:0> 位（AD1CON3<7:0>）指定。

如公式 17-1 所示，TAD 值受 ADCS 控制位和器件指令周期时钟周期 Tcy 的影响。

公式 17-1: ADC 转换时钟周期

$$TAD = 2 \cdot (TPB(ADCS + 1))$$
$$ADCS = (TAD / (2 \cdot TPB)) - 1$$

为了正确进行 A/D 转换，所选择的 ADC 转换时钟（TAD）的最小 TAD 时间必须确保为 83.33 ns（见第 17.11.1 节）。

公式 17-2: 可用采样时间，顺序采样

$$TSMP = \text{触发脉冲时间间隔 (TSEQ) - 转换时间 (TCONV)}$$
$$TSMP = TSEQ - TCONV$$

注： TSEQ 是触发脉冲间隔时间。

17.4.13 采集时间的注意事项

不同的采集 / 转换序列为采样 / 保持通道提供不同时间来采集模拟信号。用户必须确保采集时间满足第 17.11.3 节 “ADC 采样要求” 中规定的采样要求。

当 SSRC<2:0> (AD1CON1<7:5>) = 111 时, 转换触发处于 A/D 时钟控制下。SAMC<4:0> 位 (AD1CON3<12:8>) 用于选择启动采集和启动转换之间的 T_{AD} 时钟周期数。该触发选项提供了多通道上最快的转换速率。在启动采集之后, 模块会对 SAMC 位指定的 T_{AD} 时钟周期计数。

17.4.14 开启 ADC

当 ON 位 (AD1CON1<15>) 为 1 时, 模块处于工作模式, 处于完全供电状态, 可使用所有的功能。

当 ON 为 0 时, 模块被禁止。电路的数字和模拟部分被关闭, 以最大限度地节省电流消耗。

为了从关闭模式返回工作模式, 用户必须等待模拟级稳定下来。关于稳定时间, 请参见器件数据手册的 “电气特性” 章节。

注:	在 A/D 转换器正在运行时, 建议不要对除 ON (AD1CON1<15>)、SAMP (AD1CON1<1>) 和 DONE (AD1CON1<0>) 之外的 ADC 控制位进行写操作。
-----------	---

17.4.15 启动采样

17.4.15.1 手动模式

在手动采样模式下, 采集通过向 SAMP (AD1CON1<1>) 位写入 1 来启动。软件必须通过这种方式来手动启动和结束采集周期: 先将 SAMP 置 1, 然后在经过所需采集周期之后将 SAMP 清零。

17.4.15.2 自动采样模式

在自动采样模式下, 采样过程通过向 ASAM (AD1CON1<2>) 位写入 1 来启动。在自动采样模式下, 采集周期由 ADCS<7:0> (AD1CON3<7:0>) 定义。采集会在转换完成之后自动开始。自动采样模式可以使用除手动之外的任何触发源。

17.5 其他 ADC 功能

以下章节将介绍前面章节中未介绍的一些位。

17.5.1 中止采样

处于手动采样模式时，清零 SAMP (AD1CON1<1>) 位会终止采样，但如果 SSRC (AD1CON1<7:5>) = 000，则也可能会启动转换。

在自动采样模式下，清零 ASAM (AD1CON1<2>) 位不会终止正在进行的采集 / 转换序列；但是，在当前采样转换完成之后，将不会自动继续采样。

17.5.2 中止转换

在转换期间清零 ON (AD1CON1<15>) 位将中止当前的转换。不会用部分完成的 A/D 转换样本来更新 ADC 结果寄存器。即，对应的结果缓冲单元将仍然保持上一次转换完成后的值（即上一次写入该缓冲区的值）。

17.5.3 缓冲区填充状态

当使用 BUFM 控制位将转换结果缓冲区拆分时，BUFS 状态位 (AD1CON2<7>) 指示 A/D 转换器当前正在填充缓冲区的哪一半。如果 BUFS = 0，则说明 A/D 转换器正在填充 ADC1BUF0-ADC1BUF7，用户软件应从 ADC1BUF8-ADC1BUFF 读取转换值。如果 BUFS = 1，则说明情况相反，用户软件应从 ADC1BUF0-ADC1BUF7 读取转换值。

17.5.4 失调校准

ADC 模块提供了一种测量内部失调误差的方法。测量该失调误差之后，可以用软件从 A/D 转换结果中减去该误差。使用以下步骤来执行失调测量：

1. 将 A/D 转换器配置为与将在应用程序中使用的 A/D 转换器相同的形式。
2. 将 OFFCAL 位 (AD1CON2<12>) 置 1。这会改写输入选择，将采样/保持输入与 AVss 连接。
3. 如果要使用自动采样，则将 CLRASAM 位 (AD1CON1<4>) 置 1，以便在达到 SMPI 指定的采样数时停止转换。
4. 使能 A/D 转换器并执行转换。写入 ADC 结果缓冲区中的结果就是内部失调误差。
5. 清零 OFFCAL (AD2CON<12>) 位，使 A/D 转换器恢复正常工作。

注： 使用该方法只能测量同相 ADC 失调。

17.5.5 在发生中断之后终止转换序列

CLRASAM 位提供了一种在完成第一个序列之后终止自动采样的方法。将 CLRASAM 置 1 并启动自动采样序列会导致 A/D 转换器完成一个自动采样序列（采样数由 SMPI<3:0> (AD1CON2<5:2>) 定义）。硬件会清零 ASAM (AD1CON1<2>) 并置 1 中断标志。这将会停止采样过程，使得可以检查结果缓冲区，而结果不会被下一个自动转换序列覆盖。要禁止该模式，必须用软件将 CLRASAM 清零。

注： 禁止中断或屏蔽 ADC 中断对 CLRASAM 位的操作没有任何影响。

17.5.6 DONE 位操作

DONE (AD1CON1<0>) 位会在转换序列完成时置 1。

在手动模式下，DONE 位是持久性的。它会一直保持置 1，直到软件将它清零为止。可以通过查询 DONE 位来确定转换何时完成。

在所有自动采样模式 (ASAM = 1) 下，DONE 位不是持久性的。它会在转换序列结束时置 1，并在下一次采集开始被硬件清零。当 ADC 工作于自动模式时，建议不要查询 DONE 位。在转换序列完成之后，AD1IF (IFS1<1>) 标志会被锁存，因而可以进行查询。

图 17-5: 简化的 10 位高速 A/D 转换器框图 (交替采样模式)

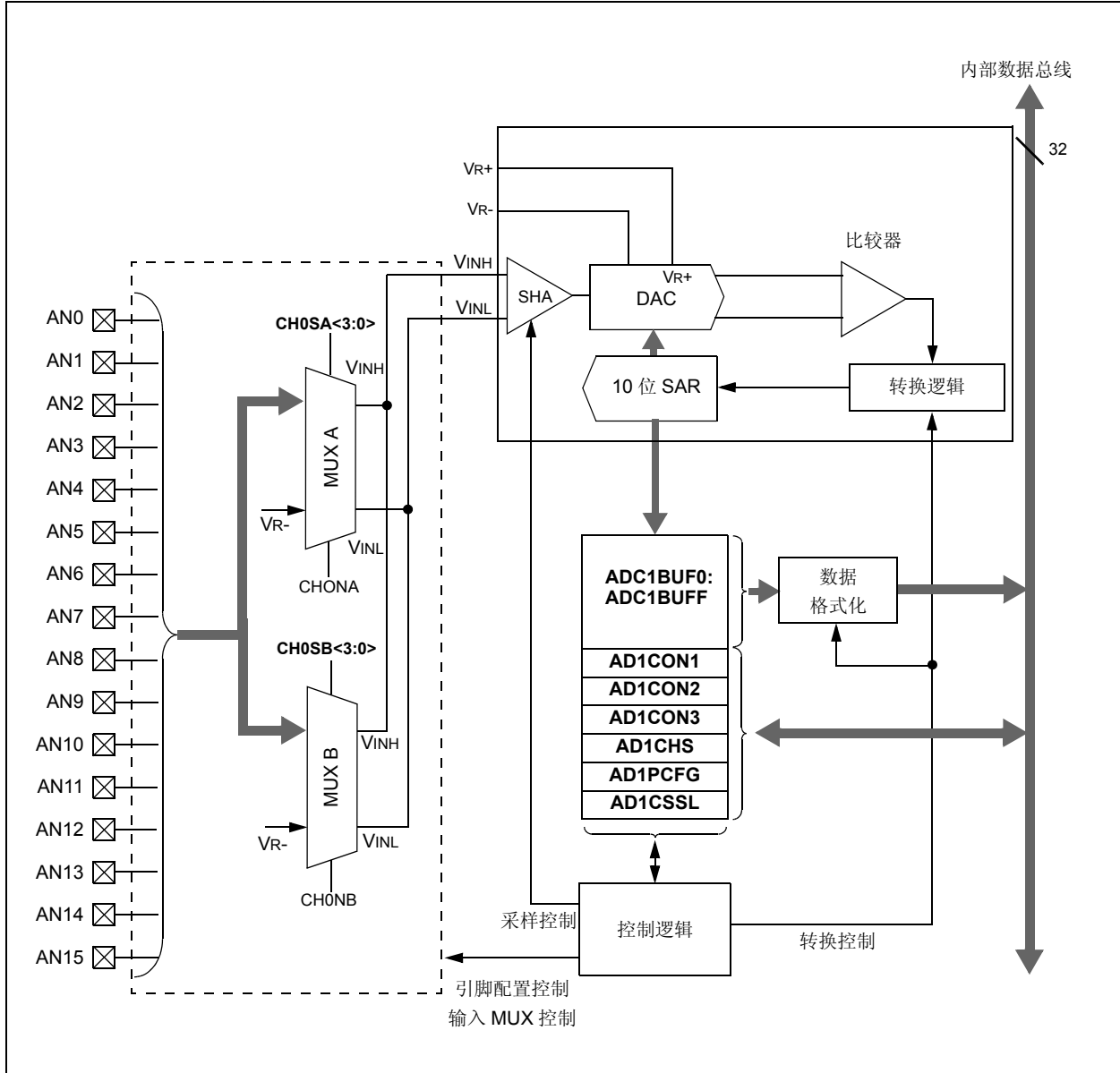


图 17-6: 简化的 10 位高速 A/D 转换器框图（扫描模式）

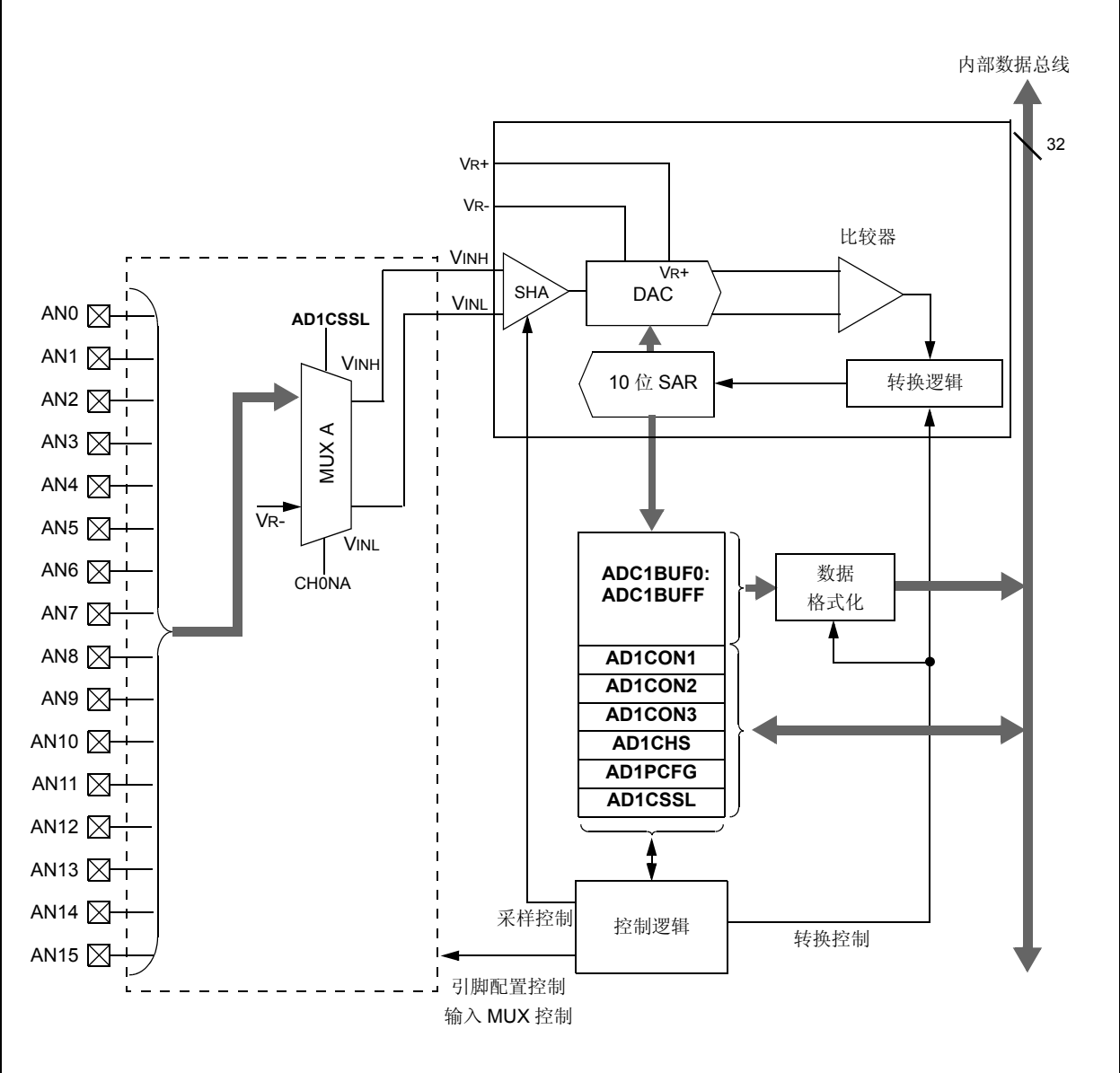
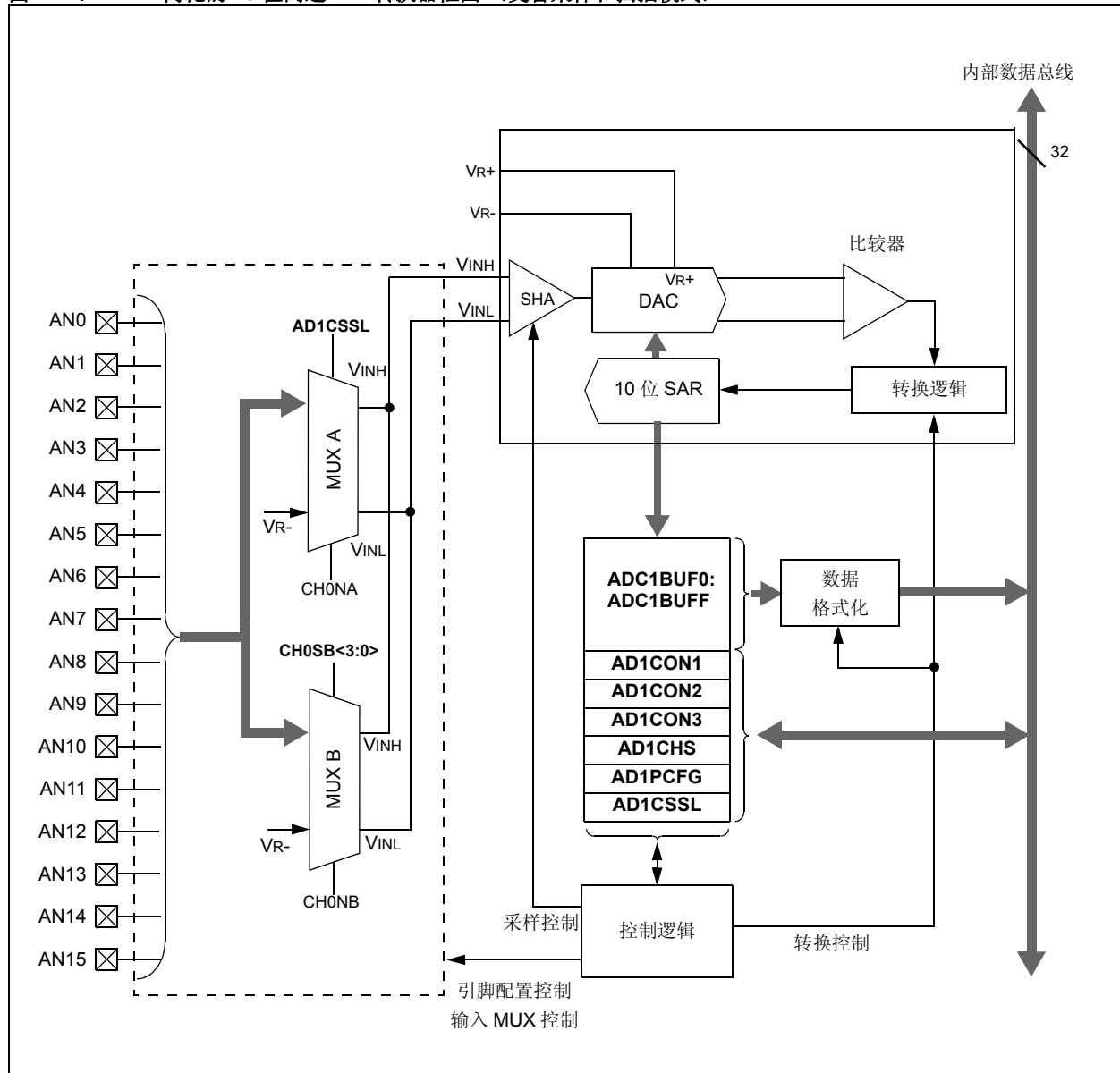


图 17-7: 简化的 10 位高速 A/D 转换器框图 (交替采样和扫描模式)



17.5.7 转换序列示例

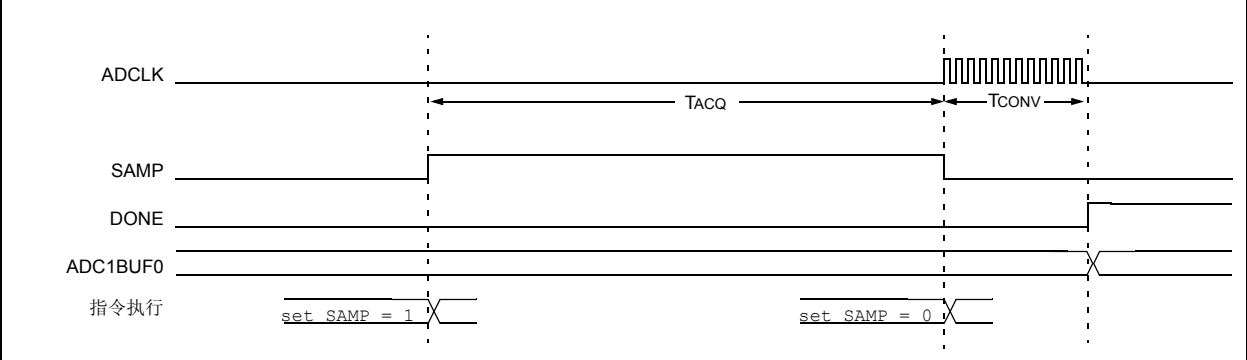
下面的配置示例说明了在不同采样和缓冲配置下的 ADC 操作。在每个示例中，将 ASAM 位置 1 会启动自动采样。转换触发结束采样并启动转换。

17.5.8 手动转换控制

当 SSRC<2:0> = 000 时，转换触发处于软件控制下。清零 SAMP 位（AD1CON1<1>）会启动转换序列。

图 17-8 的示例说明了将 SAMP 位置 1 以启动采样，将 SAMP 位清零以终止采样并启动转换的过程。用户软件必须对置 1 和清零 SAMP 位定时，以确保有足够的输入信号采集时间。代码示例请参见例 17-1。

图 17-8: 转换 1 个模拟输入，手动采样启动，手动转换启动



例 17-1: 转换 1 个通道，手动采样启动，手动转换启动的代码

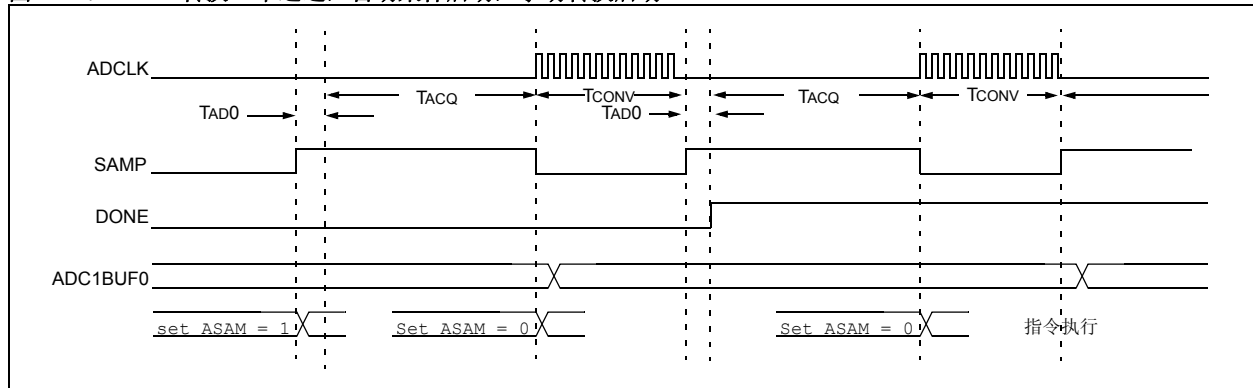
```
AD1PCFG = 0xFFFB;           // PORTB = Digital; RB2 = analog
AD1CON1 = 0x0000;           // SAMP bit = 0 ends sampling ...
                             // and starts converting
AD1CHS = 0x00020000;        // Connect RB2/AN2 as CH0 input ..
                             // in this example RB2/AN2 is the input
AD1CSSL = 0;
AD1CON3 = 0x0002;           // Manual Sample, Tad = internal 6 TPB
AD1CON2 = 0;

AD1CON1SET = 0x8000;        // turn ADC ON
while (1)                   // repeat continuously
{
    AD1CON1SET = 0x0002;    // start sampling ...
    DelayNmSec(100);        // for 100 mS
    AD1CON1CLR = 0x0002;    // start Converting
    while (!(AD1CON1 & 0x0001)); // conversion done?
    ADCValue = ADC1BUF0;    // yes then get ADC value
}                            // repeat
```

17.5.9 自动采集

图 17-9 的示例说明了将 ASAM (AD1CON1<2>) 位置 1 以启动自动采集, 将 SAMP (AD1CON1<1>) 位清零以终止采样并启动转换的过程。转换完成之后, 模块将自动恢复为采集状态。在采集间隔开始时, SAMP 位会自动置 1。用户软件必须定时对 SAMP 位清零, 以确保有足够的输入信号采集时间, 因为每两次 SAMP 位清零之间的时间中包括转换时间和采集时间。代码示例请参见例 17-2。

图 17-9: 转换 1 个通道, 自动采样启动, 手动转换启动



例 17-2: 转换 1 个通道, 自动采样启动, 手动转换启动的代码

```

AD1PCFG = 0xFF7F;           // all PORTB = Digital but RB7 = analog
AD1CON1 = 0x0004;           // ASAM bit = 1 implies acquisition ..
                              // starts immediately after last
                              // conversion is done
AD1CHS = 0x00070000;         // Connect RB7/AN7 as CH0 input ..
                              // in this example RB7/AN7 is the input
AD1CSSL = 0;
AD1CON3 = 0x0002;           // Sample time manual, Tad = internal 6 Tps
AD1CON2 = 0;

AD1CON1SET = 0x8000;         // turn ADC ON
while (1)                   // repeat continuously
{
    DelayNmSec(100);         // sample for 100 mS
    AD1CON1SET = 0x0002;     // start Converting
    while (!(AD1CON1 & 0x0001)); // conversion done?
    ADCValue = ADC1BUF0;     // yes then get ADC value
}

```

17.5.10 对转换触发计时

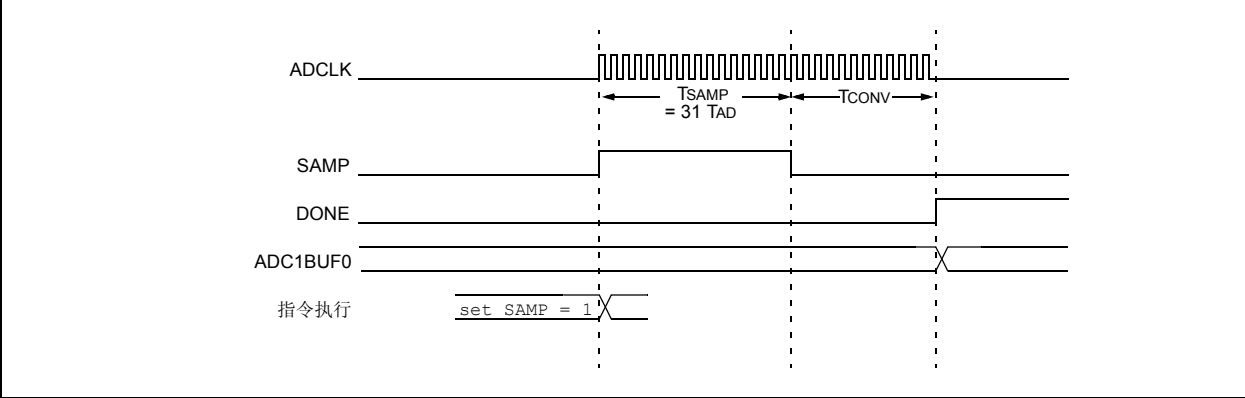
当 SSRC<2:0> = 111 时，转换触发处于 ADC 时钟控制下。SAMC 位（AD1CON3<4:0>）用于选择启动采集和启动转换之间的 TAD 时钟周期数。该触发选项提供了多通道上最快的转换速率。在启动采集之后，模块会对 SAMC 位指定的 TAD 时钟周期计数。

公式 17-3: 计时转换触发时间

$$T_{SMP} = SAMC<4:0> * T_{AD}$$

SAMC 必须总是设定为至少一个时钟周期。代码示例请参见例 17-3。

图 17-10: 转换 1 个通道，手动采样启动，基于 TAD 的转换启动



例 17-3: 转换 1 个通道，手动采样启动，基于 TAD 的转换启动的代码

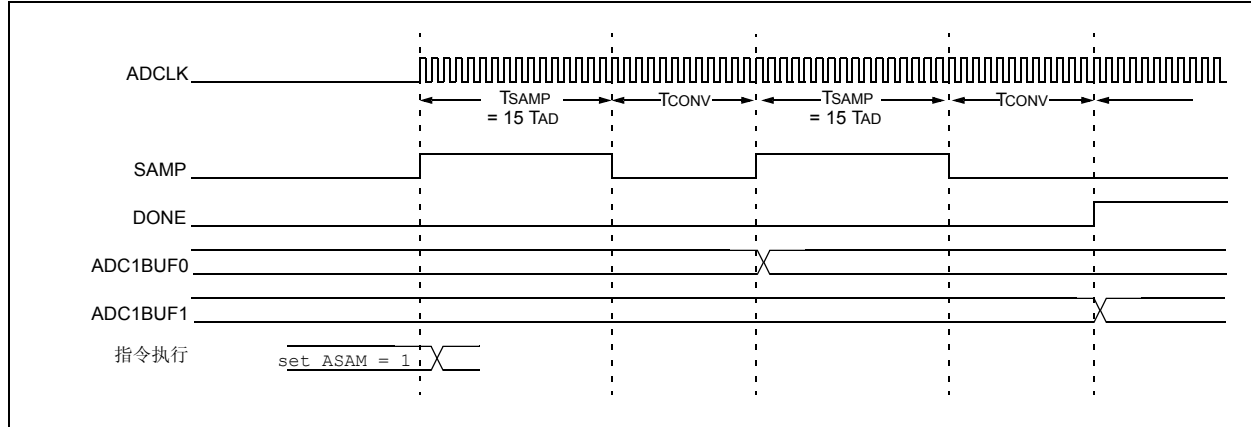
```
AD1PCFG = 0xEFFF;           // all PORTB = Digital; RB12 = analog
AD1CON1 = 0x00E0;           // SSRC bit = 111 implies internal
                             // counter ends sampling and starts
                             // converting.
AD1CHS = 0x000C0000;        // Connect RB12/AN12 as CH0 input ..
                             // in this example RB12/AN12 is the input
AD1CSSL = 0;
AD1CON3 = 0x1F02;           // Sample time = 31Tad
AD1CON2 = 0;

AD1CON1SET = 0x8000;        // turn ADC ON
while (1)                   // repeat continuously
{
    AD1CON1CLR = 0x0002;    // start sampling then ...
                             // after 31Tad go to conversion
    while (!(AD1CON1 & 0x0001)); // conversion done?
    ADCValue = ADC1BUF0;    // yes then get ADC value
}                            // repeat
```

17.5.11 自由运行采样转换序列

如图 17-11 所示，使用自动转换触发模式（SSRC = 111），配合自动采样启动模式（ASAM = 1），可使 ADC 模块确定采集 / 转换序列，而无需用户干预或其他器件资源。此“计时”模式允许模块在初始化之后进行连续数据收集。代码示例请参见例 17-4。

图 17-11: 转换 1 个通道，两次，自动采样启动，基于 TAD 的转换启动



例 17-4: 转换 1 个通道，自动采样启动，基于 TAD 的转换启动的代码

```

AD1PCFG = 0xFFFB;           // all PORTB = Digital; RB2 = analog
AD1CON1 = 0x00E0;           // SSRC bit = 111 implies internal
                              // counter ends sampling and starts
                              // converting.
AD1CHS = 0x00020000;         // Connect RB2/AN2 as CH0 input ..
                              // in this example RB2/AN2 is the input

AD1CSSL = 0;
AD1CON3 = 0x0F00;           // Sample time = 15Tad
AD1CON2 = 0x0004;           // Interrupt after every 2 samples

AD1CON1SET = 0x8000;         // turn ADC ON
while (1)                   // repeat continuously
{
    ADCValue = 0;           // clear value
    ADC16Ptr = &ADC1BUF0;   // initialize ADC1BUF0 pointer
    IFS1CLR = 0x0002;       // clear ADC interrupt flag
    AD1CON1SET = 0x0004;     // auto start sampling
                              // for 31Tad then go to conversion
    while (!IFS1 & 0x0002);   // conversion done?
    AD1CON1CLR = 0x0004;     // yes then stop sample/convert
    for (count = 0; count < 2; count++) // average the 2 ADC values
    {
        ADCValue = ADCValue + *(ADC16Ptr++);
        ADCValue = ADCValue >> 1;
    }
    // repeat
}

```

17.5.12 使用计时转换触发和自动采样时关于采集时间的注意事项

不同的采集 / 转换序列为采样 / 保持通道提供不同的可用采集时间来采集模拟信号。用户必须确保采集时间超出第 17.11.3 节 “ADC 采样要求” 中规定的采集要求。

假设模块设置为自动采样并使用计时转换触发，则采集时间间隔由 SAMC (AD1CON3<12:8>) 位决定。

公式 17-4: 可用采样时间

$$T_{SMP} = SAMC<4:0> * T_{AD}$$

图 17-12: 转换 1 个通道，手动采样启动，基于转换触发的转换启动

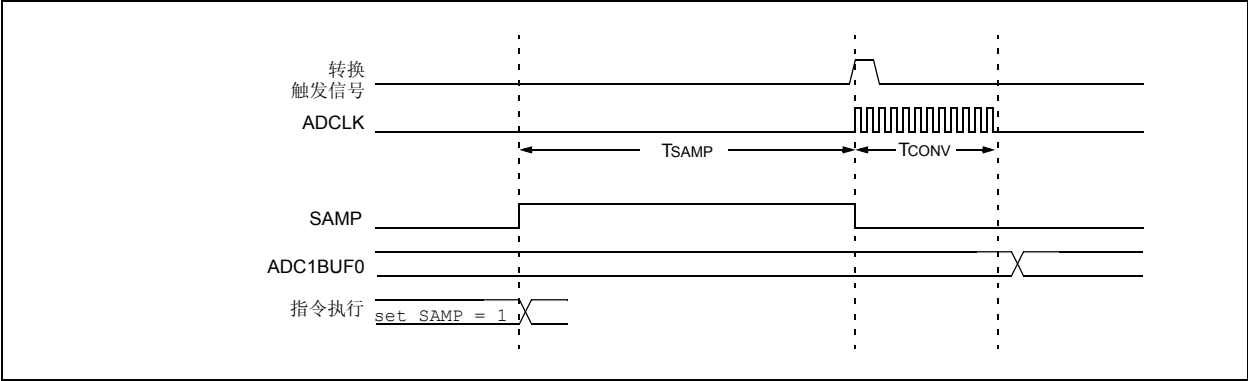
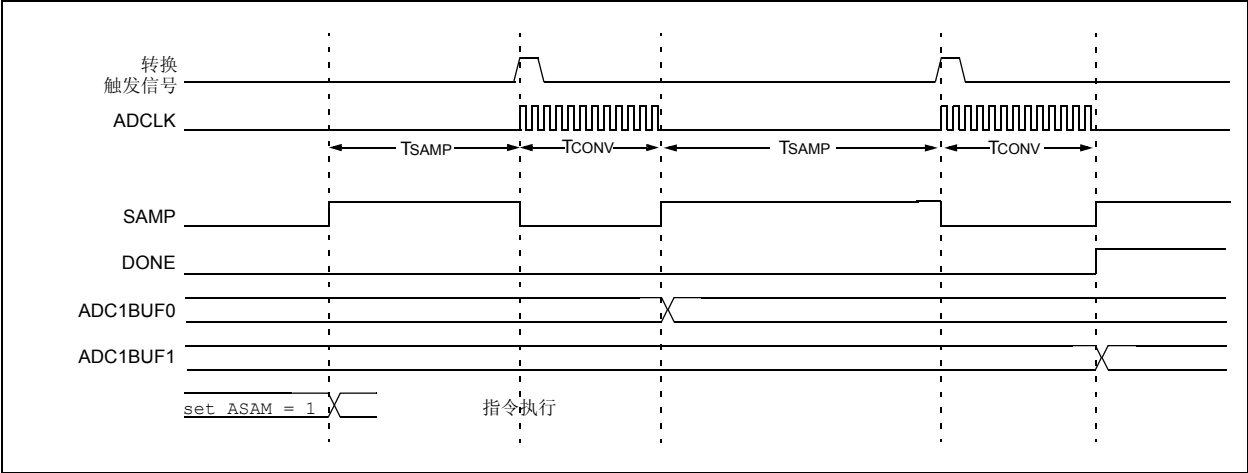


图 17-13: 转换 1 个通道，自动采样启动，基于转换触发的转换启动



例 17-5: 转换 1 个通道, 自动采样启动, 基于转换触发的转换启动的代码

```

AD1PCFG = 0xFFFFB;           // all PORTB = Digital; RB2 analog
AD1CON1 = 0x0040;             // SSRC bit = 010 implies GP TMR3
                                // compare ends sampling and starts
                                // converting.
AD1CHS = 0x00020000;          // Connect RB2/AN2 as CH0 input ..
                                // in this example RB2/AN2 is the input

AD1CSSL = 0;
AD1CON3 = 0x0000;             // Sample time is TMR3, Tad = internal TPB*2
AD1CON2 = 0x0004;             // Interrupt after 2 conversions

                                // set TMR3 to time out every 125 mSecs

TMR3 = 0x0000;
PR3 = 0x3FFF;
T3CON = 0x8010;

AD1CON1SET = 0x8000;           // turn ADC ON
AD1CON1SET = 0x0004;           // start auto sampling every 125 mSecs
while (1)                      // repeat continuously
{
    while (!IFS1 & 0x0002){};   // conversion done?
    ADCValue = ADC1BUF0;        // yes then get first ADC value
    IFS1CLR = 0x0002;           // clear ADIF
}                                // repeat

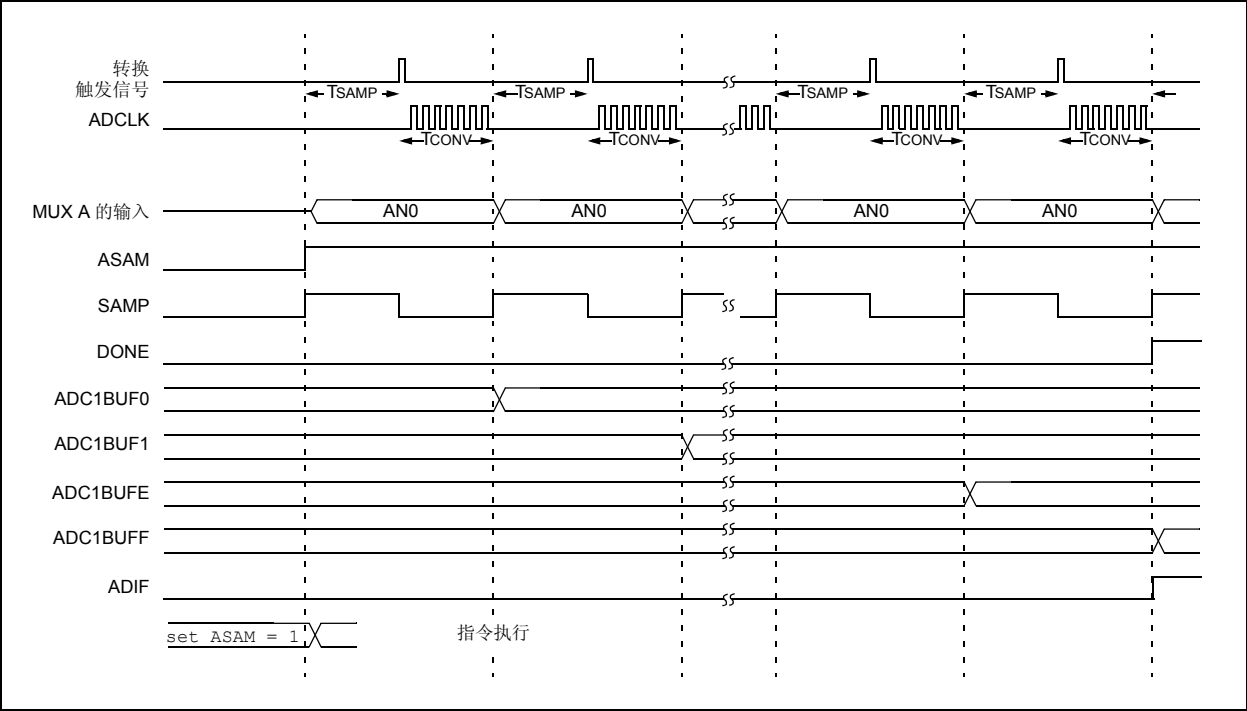
```

17.5.13 多次采样单个通道

图 17-14 和表 17-4 给出了 A/D 转换器的基本配置。在这种情况下，将对一个 ADC 输入 AN0 进行采集和转换。结果存储在 ADC1BUF 缓冲区中。该过程会重复 15 次，直到缓冲区满为止，然后模块产生中断。然后重复整个过程。

ALTS (AD1CON2<0>) 清零时，只有 MUX A 输入有效。CH0SA (AD1CHS<19:16>) 位和 CH0NA (AD1CHS<23>) 位指定 (AN0-VREF-) 为采样 / 保持通道的输入。不使用所有其他输入选择位。

图 17-14: 转换 1 个通道 15 次，每次中断采样 15 次



工作序列	
采样 MUX A 输入: AN0	
	转换, 写缓冲区 0x0
采样 MUX A 输入: AN0	
	转换, 写缓冲区 0x1
采样 MUX A 输入: AN0	
	转换, 写缓冲区 0x2
采样 MUX A 输入: AN0	
	转换, 写缓冲区 0x3
采样 MUX A 输入: AN0	
	转换, 写缓冲区 0x4
采样 MUX A 输入: AN0	
	转换, 写缓冲区 0x5
采样 MUX A 输入: AN0	
	转换, 写缓冲区 0x6
采样 MUX A 输入: AN0	
	转换, 写缓冲区 0x7
采样 MUX A 输入: AN0	
	转换, 写缓冲区 0x8
采样 MUX A 输入: AN0	
	转换, 写缓冲区 0x9
采样 MUX A 输入: AN0	
	转换, 写缓冲区 0xA
采样 MUX A 输入: AN0	
	转换, 写缓冲区 0xB
采样 MUX A 输入: AN0	
	转换, 写缓冲区 0xC
采样 MUX A 输入: AN0	
	转换, 写缓冲区 0xD
采样 MUX A 输入: AN0	
	转换, 写缓冲区 0xE
	中断
	重复

缓冲区分割表	
	AN0 采样 16
	AN0 采样 17
	AN0 采样 18
	AN0 采样 19
	AN0 采样 20
	AN0 采样 21
	AN0 采样 22
	AN0 采样 23
	AN0 采样 24
	AN0 采样 25
	AN0 采样 26
	AN0 采样 27
	AN0 采样 28
	AN0 采样 29
	AN0 采样 30

17.5.14 示例：扫描模拟输入时的 A/D 转换

图 17-15 和表 17-5 给出了一个典型的设置，对所有可用的模拟输入通道进行采样和转换。将 CSCNA (AD1CON2<10>) 位置 1 指定对 ADC 输入进行扫描。其他情况与前面的示例类似（见第 17.5.13 节“多次采样单个通道”）。

首先，对 AN0 输入进行采集和转换。结果存储在 ADC1BUF 缓冲区中。然后对 AN1 输入进行采集和转换。该扫描输入的过程重复 16 次，直到缓冲区满为止，然后模块产生中断。然后重复整个过程。

图 17-15: 扫描 16 个输入，每次中断采样 16 次

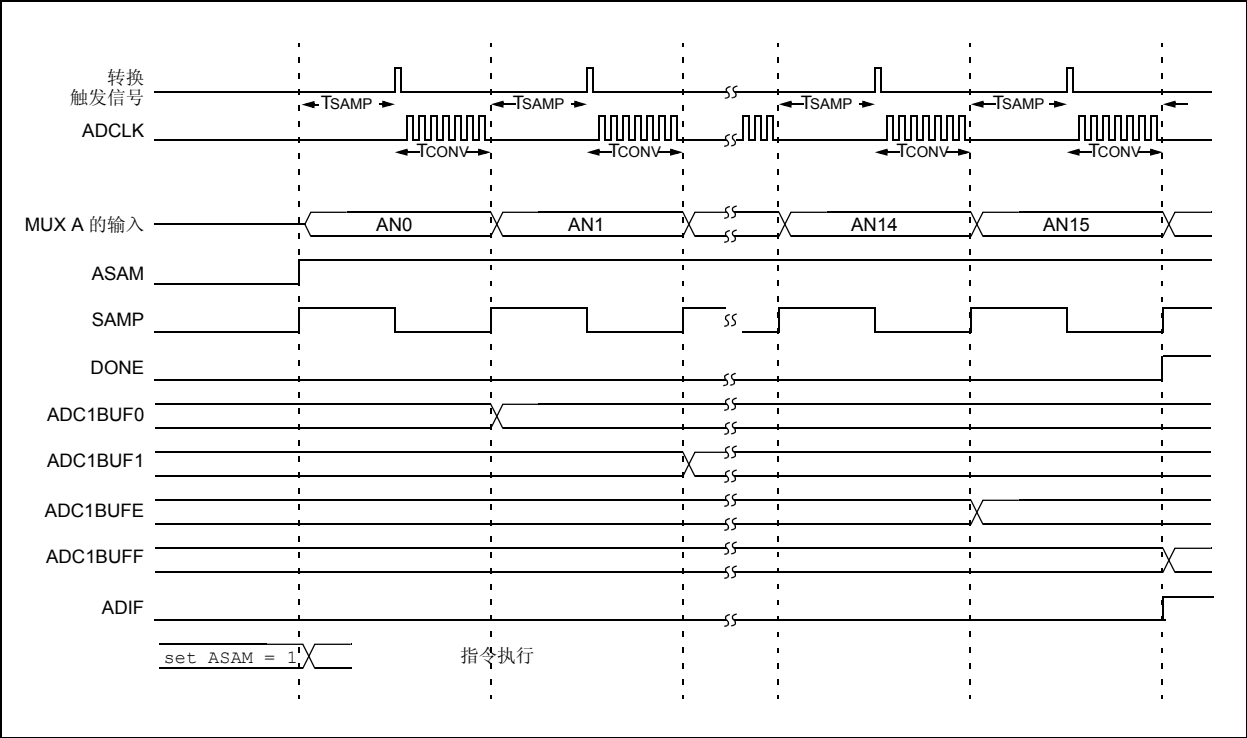


表 17-5: 每次中断扫描 16 个输入

控制位 序列选择		工作序列	
SMPI<2:0> = 1111	第 16 次采样后产生中断	采样 MUX A 输入: AN0	转换, 写缓冲区 0x0
—	—	采样 MUX A 输入: AN1	转换, 写缓冲区 0x1
—	—	采样 MUX A 输入: AN2	转换, 写缓冲区 0x2
BUFM = 0	单个 16 字结果缓冲区	采样 MUX A 输入: AN3	转换, 写缓冲区 0x3
ALTS = 0	总是使用 MUX A 输入选择	采样 MUX A 输入: AN4	转换, 写缓冲区 0x4
MUX A 输入选择		采样 MUX A 输入: AN5	转换, 写缓冲区 0x5
CH0SA<3:0> = n/a	被 CSCNA 覆盖	采样 MUX A 输入: AN6	转换, 写缓冲区 0x6
CH0NA = 0	选择 Vr- 作为 MUX A 反相输入	采样 MUX A 输入: AN7	转换, 写缓冲区 0x7
CSCNA = 1	扫描输入	采样 MUX A 输入: AN8	转换, 写缓冲区 0x8
CSSL<15:0> = 1111 1111 1111 1111	扫描输入选择	采样 MUX A 输入: AN9	转换, 写缓冲区 0x9
—	—	采样 MUX A 输入: AN10	转换, 写缓冲区 0xA
—	—	采样 MUX A 输入: AN11	转换, 写缓冲区 0xB
MUX B 输入选择		采样 MUX A 输入: AN12	转换, 写缓冲区 0xC
SB<3:0> = n/a	MUX B 同相输入未使用	采样 MUX A 输入: AN13	转换, 写缓冲区 0xD
CH0NB = n/a	MUX B 反相输入未使用	采样 MUX A 输入: AN14	转换, 写缓冲区 0xE
—	—	采样 MUX A 输入: AN15	转换, 写缓冲区 0xF
—	—	中断	
		重复	

缓冲区 地址	缓冲区 (第 1 次中断)	缓冲区 (第 2 次中断)
ADC1BUF0	AN0 采样 1	AN0 采样 17
ADC1BUF1	AN1 采样 2	AN1 采样 18
ADC1BUF2	AN2 采样 3	AN2 采样 19
ADC1BUF3	AN3 采样 4	AN3 采样 20
ADC1BUF4	AN4 采样 5	AN4 采样 21
ADC1BUF5	AN5 采样 6	AN5 采样 22
ADC1BUF6	AN6 采样 7	AN6 采样 23
ADC1BUF7	AN7 采样 8	AN7 采样 24
ADC1BUF8	AN8 采样 9	AN8 采样 25
ADC1BUF9	AN9 采样 10	AN9 采样 26
ADC1BUFA	AN10 采样 11	AN10 采样 27
ADC1BUFB	AN11 采样 12	AN11 采样 28
ADC1BUFC	AN12 采样 13	AN12 采样 29
ADC1BUFD	AN13 采样 14	AN13 采样 30
ADC1BUFE	AN14 采样 15	AN14 采样 31
ADC1BUFF	AN15 采样 16	AN15 采样 32

17.5.14.1 示例：使用双 8 字缓冲区

图 17-16 和表 17-6 说明了使用双 8 字缓冲区和交替填充缓冲区。将 BUFM (AD1CON2<1>) 位置 1 可使能双 8 字缓冲区。BUFM 设置不会影响其他工作参数。首先，转换序列会开始填充 ADC1BUF0 (缓冲单元 0 x 0) 处的缓冲区。第一次中断发生之后，缓冲区从 ADC1BUF8 (缓冲单元 0 x 8) 开始填充。BUFS (AD1CON2<7>) 状态位会在每次中断之后交替置 1 和清零，以指示正在填充的是哪个缓冲区。在该示例中，对 3 个模拟输入进行采样，并且每隔 3 次采样产生一次中断。

图 17-16: 使用双 8 字缓冲区转换 3 个输入，每次中断采样 3 次

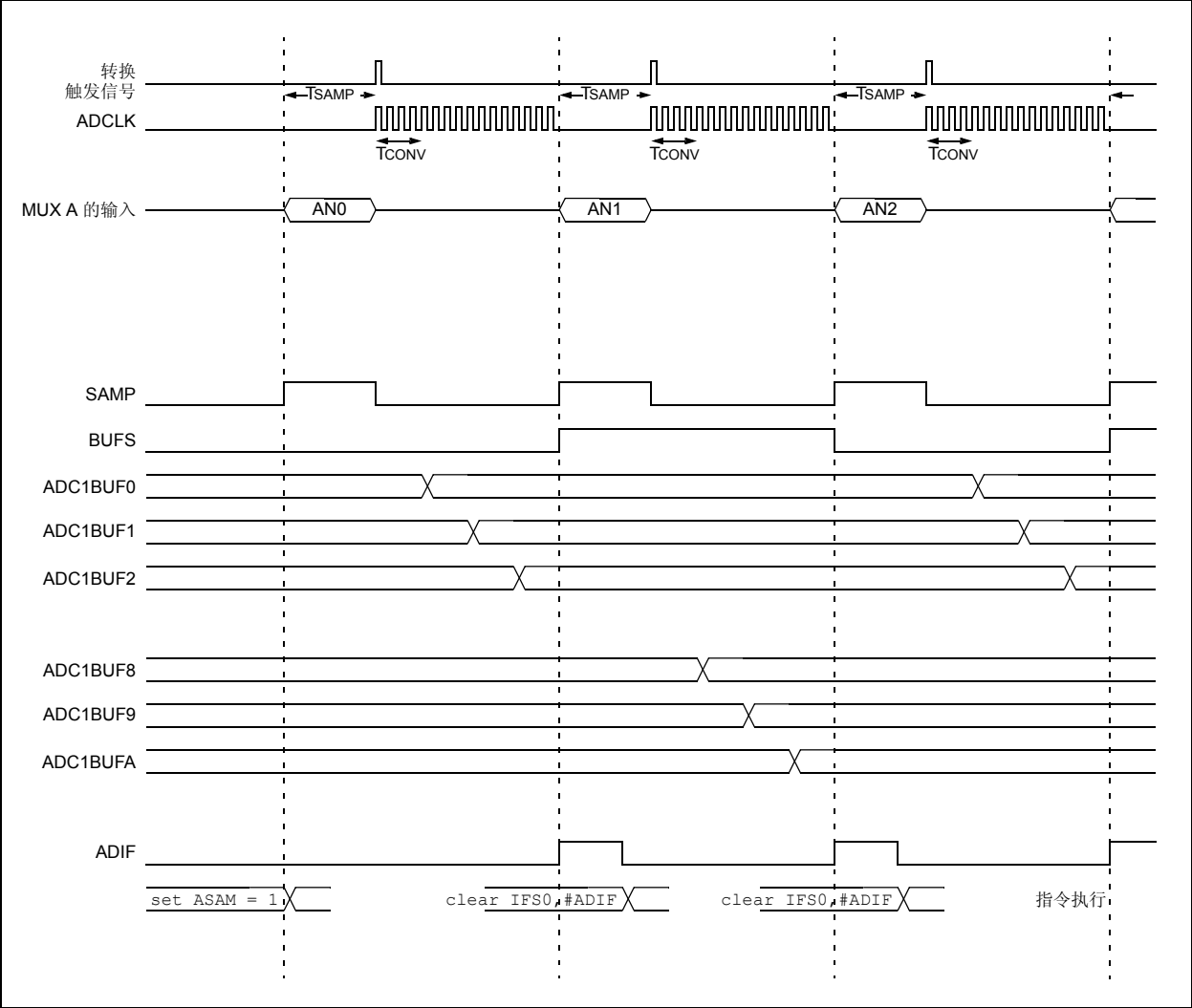


表 17-6: 使用双 8 字缓冲区转换 3 个输入，每次中断采样 3 次

控制位 序列选择		工作序列	
SMPI<2:0> = 0010	每 3 次采样后产生中断	采样 MUX A 输入: AN0	转换 AN0, 写缓冲区 0x0
—	—	采样 MUX A 输入: AN1	转换 AN1, 写缓冲区 0x1
—	—	采样 MUX A 输入: AN2	转换 AN2, 写缓冲区 0x2
BUFM = 1	双 8 字结果缓冲区	—	—
ALTS = 0	总是使用 MUX A	—	中断: 更改缓冲区
MUX A 输入选择		—	—
CH0SA<3:0> = n/a	MUX A 同相输入选择未使用	采样 MUX A 输入: AN0	转换 AN0, 写缓冲区 0x8
CH0NA = 0	选择 VR- 作为 MUX A 反相输入	采样 MUX A 输入: AN1	转换 AN1, 写缓冲区 0x9
CSCNA = 1	使能输入扫描	采样 MUX A 输入: AN2	转换 AN2, 写缓冲区 0xA
CSSL<15:0> = 0x0007	扫描输入选择扫描列表由 AN0、AN1 和 AN2 组成	—	—
AD1PCFG = 0X0007	为 AN0、AN1 和 AN2 选择模拟输入模式	—	中断: 更改缓冲区
—	—	—	重复
MUX B 输入选择			
CH0SB<3:0> = n/a	MUX B 同相输入未使用		
CH0NB = n/a	MUX B 反相输入未使用		
—	—		
—	—		

缓冲区 地址	缓冲区 (第 1 次中断)	缓冲区 (第 2 次中断)
ADC1BUF0	AN0 采样 1	
ADC1BUF1	AN1 采样 1	
ADC1BUF2	AN2 采样 1	
ADC1BUF3		
ADC1BUF4		
ADC1BUF5		
ADC1BUF6		
ADC1BUF7		
ADC1BUF8		
ADC1BUF9		
ADC1BUFA		AN0 采样 2
ADC1BUFB		AN1 采样 2
ADC1BUFC		AN2 采样 2
ADC1BUFD		
ADC1BUFE		
ADC1BUFF		

• • •

17.5.14.2 示例：使用交替 MUX A 和 MUX B 输入选择

图 17-17 和表 17-7 说明了对分配给 MUX A 和 MUX B 的输入进行交替采样。将 ALTS (AD1CON2<0>) 位置 1 可使能交替输入选择。第一次采样使用由 CH0SA (AD1CHS<19:16>) 和 CH0NA (AD1CHS<23>) 位指定的 MUX A 输入。下一次采样使用由 CH0SB (AD1CHS<27:24>) 和 CH0NB (AD1CHS<31>) 位指定的 MUX B 输入。

在以下示例中，MUX B 输入规范之一是使用 2 个模拟输入作为采样 / 保持的差分源。

该示例同样使用双 8 字缓冲区进行说明。每 4 次采样后产生中断，使得在每次中断时在缓冲区中填充 4 个字。

图 17-17： 通过每次中断交替进行 4 次采样来转换两个模拟输入

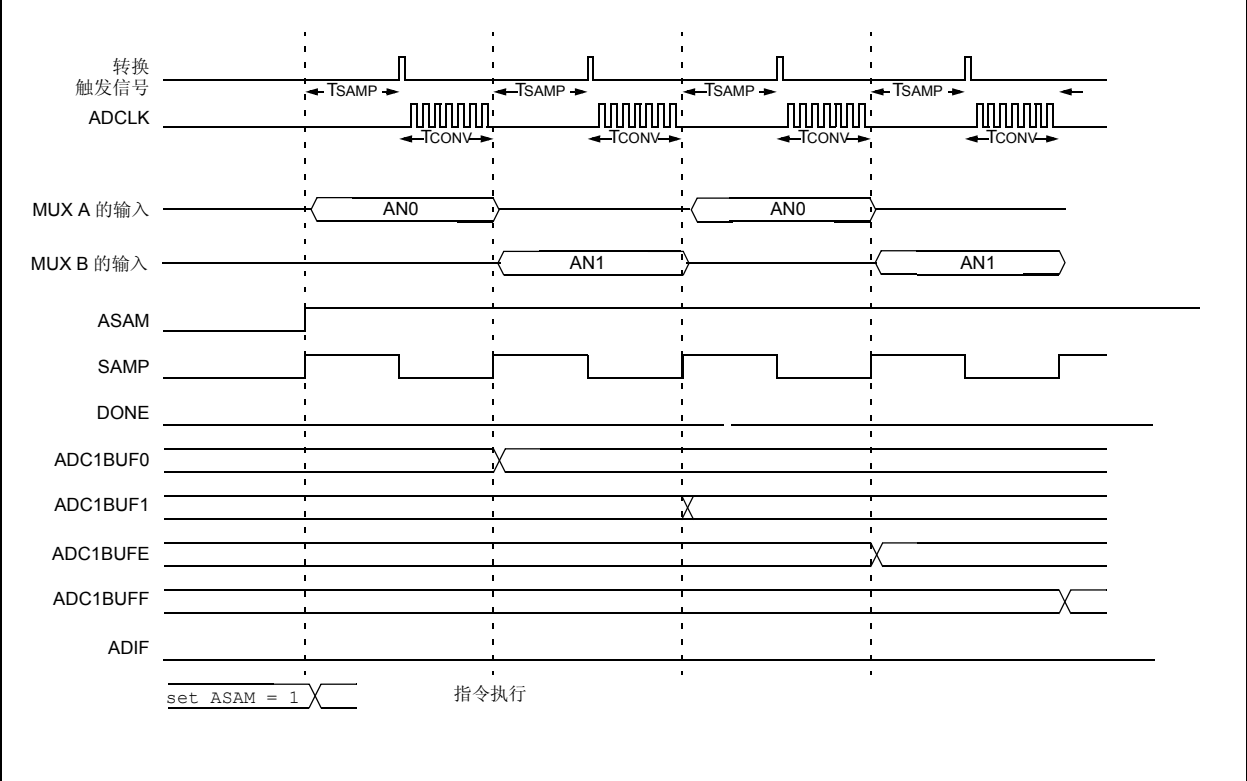


表 17-7: 使用交替输入选择转换两组输入

控制位 序列选择	工作序列
SMPI<2:0> = 0011 第 4 次采样后产生中断	采样 MUX A 输入: AN0 转换, 写缓冲区 0x0
—	采样 MUX B 输入: AN1 转换, 写缓冲区 0x1
—	采样 MUX A 输入: AN0 转换, 写缓冲区 0x2
BUFM = 1 双 8 字结果缓冲区	采样 MUX B 输入: AN1 转换, 写缓冲区 0x3
ALTS = 1 交替 MUX A/B 输入选择	中断; 更改缓冲区
MUX A 输入选择	采样 MUX A 输入: AN0 转换, 写缓冲区 0x8
CH0SA<3:0> = 0000 选择 AN0 作为 MUX A 同相输入	采样 MUX B 输入: AN1 转换, 写缓冲区 0x9
CH0NA = 0 选择 VR- 作为 MUX A 反相输入	采样 MUX A 输入: AN0 转换, 写缓冲区 0xA
CSCNA = 0 无输入扫描	采样 MUX B 输入: AN1 转换, 写缓冲区 0xB
CSSL<15:0> = n/a 扫描输入选择未使用	中断; 更改缓冲区
—	重复
—	
MUX B 输入选择	
CH0SB<3:0> = 0001 选择 AN1 作为 MUX B 同相输入	
CH0NB = 0 选择 VR- 作为 MUX B 反相输入	
—	
—	

缓冲区 地址	缓冲区 (第 1 次中断)	缓冲区 (第 2 次中断)
ADC1BUF0	AN0 采样 1	
ADC1BUF1	AN1 采样 1	
ADC1BUF2	AN0 采样 2	
ADC1BUF3	AN1 采样 2	
ADC1BUF4		
ADC1BUF5		
ADC1BUF6		
ADC1BUF7		
ADC1BUF8		
ADC1BUF9		
ADC1BUFA		AN0 采样 3
ADC1BUFB		AN1 采样 3
ADC1BUFC		AN0 采样 4
ADC1BUFD		AN1 采样 4
ADC1BUFE		
ADC1BUFF		

• • •

17.5.14.3 示例：使用交替采样模式和扫描列表转换 3 个模拟输入

图 17-18、图 17-19 和表 17-8 给出了逐个扫描输入并在 MUX A 和 MUX B 之间交替时的采样图示。选择交替采样模式时，要采样的第一个输入将是 MUX A 选择的输入，第二个采样将是 MUX B 选择的输入。然后，此过程一直重复。组合使用扫描与交替输入模式时，MUX A 的同相输入通过 AD1CSSL 寄存器（不是 CH0SA）的内容进行选择。对于选择 MUX A 的每次采样，都会采样扫描列表中的下一项。MUX B 的同相输入通过 CH0SB（AD1CHS<27:24>）进行选择。

当 ASAM（AD1CON1<2>）清零时，转换结束之后将不会继续采样，但在 SAMP（AD1CON1<1>）位置 1 时则会继续采样。

图 17-18： 使用交替采样模式和扫描列表转换 3 个模拟输入

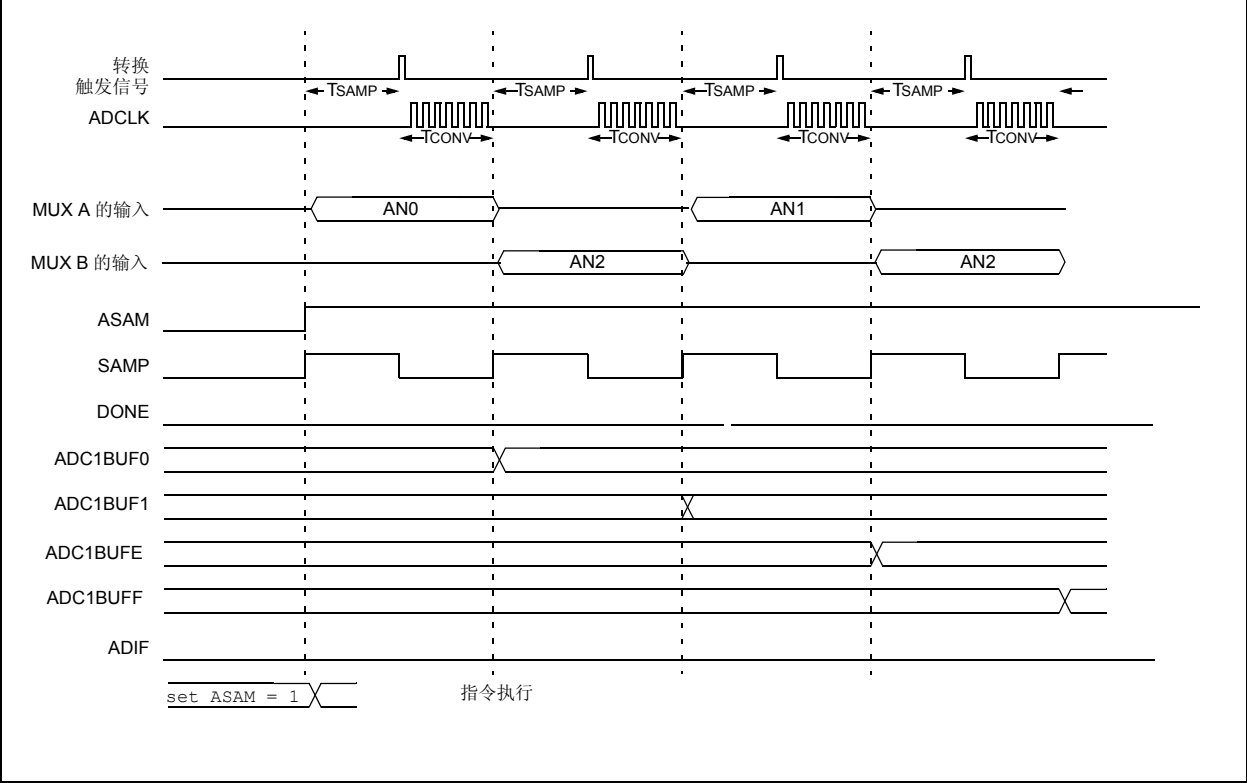


图 17-19: 10 位高速 A/D 转换器框图 (交替采样和扫描)

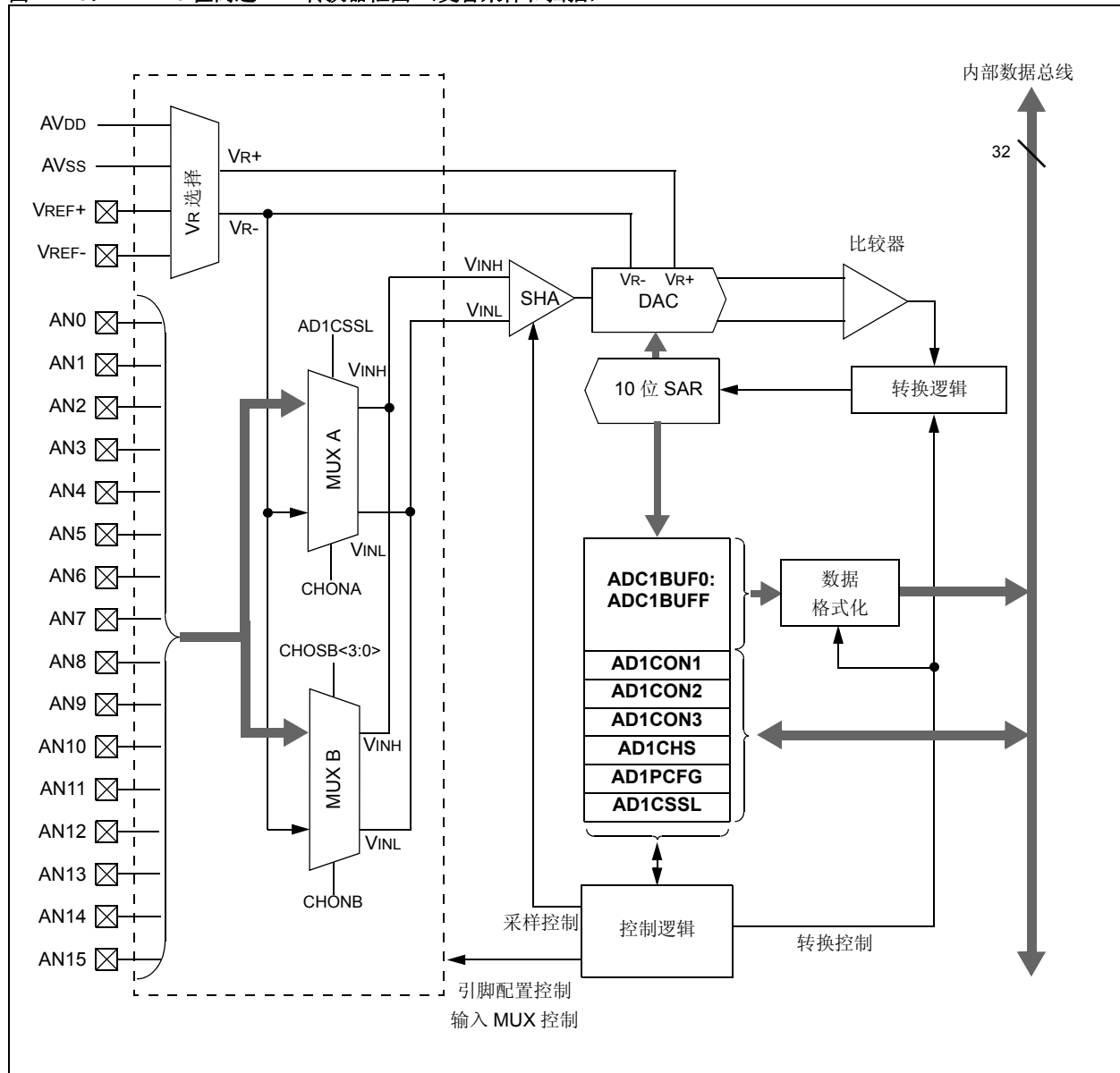


表 17-8：使用顺序采样对 8 个输入进行采样

控制位 序列选择		工作序列	
SMPI<2:0> = 0011	第 4 次采样后产生中断	采样：AN0	转换，写缓冲区 0x0
—	—	采样：AN2	转换，写缓冲区 0x1
—	—	采样：AN1	转换，写缓冲区 0x2
BUFM = 0	单个 16 字结果缓冲区	采样：AN2	转换，写缓冲区 0x3
ALTS = 1	交替 MUX A/B 输入选择		
MUX A 输入选择			
CH0SA<3:0> = n/a	未使用		中断
CH0NA = 0	选择 VR- 作为 CH0- 输入		重复
CSCNA = 1	使能输入扫描		
CSSL<15:0> = n/a	扫描输入选择扫描列表由 AN0 和 AN1 组成		
—	—		
—	—		
MUX B 输入选择			
CH0SB<3:0> = 0010	选择 AN7 作为 CH0+ 输入		
CH0NB = 0	选择 VR- 作为 CH0- 输入		
—	—		
—	—		

缓冲区 地址	缓冲区 (第 1 次中断)	缓冲区 (第 2 次中断)
ADC1BUF0	AN0 采样 1	AN0 采样 5
ADC1BUF1	AN2 采样 2	AN2 采样 6
ADC1BUF2	AN1 采样 3	AN1 采样 7
ADC1BUF3	AN2 采样 4	AN2 采样 8
ADC1BUF4		
ADC1BUF5		
ADC1BUF6		
ADC1BUF7		
ADC1BUF8		
ADC1BUF9		
ADC1BUFA		
ADC1BUFB		
ADC1BUFC		
ADC1BUFD		
ADC1BUFE		
ADC1BUFF		

• • •

17.6 初始化

例 17-6 给出了 ADC 模块初始化的简单代码示例。

在这种特定配置中，所有 16 个模拟输入引脚（AN0-AN15）均设置为模拟输入。禁止在 IDLE（空闲）模式下工作，输出数据使用无符号小数格式，AVDD 和 AVSS 用作 VR+ 和 VR-。通过软件手动执行采集启动和转换启动（转换触发）。CH0 SHA 用于进行转换。输入扫描被禁止，并且在每个采集 / 转换序列（1 个转换结果）之后会产生中断。ADC 转换时钟为 TPB/2。

由于采集通过在每次转换完成之后将 SAMP 位（AD1CON1<1>）置 1 来手动启动，所以自动采样时间位 SAMC<4:0>（AD1CON3<12:8>）会被忽略。并且，由于转换启动（即采集结束）也是手动触发的，所以每次需要转换新采样时都需要清零 SAMP 位。

例 17-6: ADC 初始化代码示例

```
AD1PCFG = 0x0000;          /* Configure ADC port
                             all input pins are analog */

AD1CON1 = 0x2208;          /* Configure sample clock source and Conversion Trigger mode.
                             Unsigned Fractional format, Manual conversion trigger,
                             Manual start of sampling, Simultaneous sampling,
                             No operation in IDLE mode. */

AD1CON2 = 0x0000;          /* Configure ADC voltage reference
                             and buffer fill modes.
                             VREF from AVDD and AVSS,
                             Inputs are not scanned,
                             Interrupt every sample */

AD1CON3 = 0x0000;          /* Configure ADC conversion clock */

AD1CHS = 0x0000;           /* Configure input channels,
                             CH0+ input is AN0.
                             CH0- input is VREFL (AVss)

AD1CSSL = 0x0000;          /* No inputs are scanned.
                             Note:Contents of AD1CSSL are ignored when CSCNA = 0 */

IFS1CLR = 2;               /*Clear ADC conversion interrupt*/

// Configure ADC interrupt priority bits (AD1IP<2:0>) here, if
// required.(default priority level is 4)

IEC1SET = 2;               /* Enable ADC conversion interrupt*/

AD1CON1SET = 0x8000;        /* Turn on the ADC module */
AD1CON1SET = 0x0002;        /* Start sampling the input */
DelayNmSec(100);           /* Ensure the correct sampling time has elapsed before
                             starting a conversion.*/

AD1CON1CLR = 0x0002;        /* End Sampling and start Conversion*/
:                            /* The DONE bit is set by hardware when the convert sequence
                             is finished. */
:                            /* The ADIF bit will be set. */
```

例 17-7: 在 400 ksps 时转换 1 个通道, 自动采样启动, 2 T_{AD} 采样时间的代码示例

```
AD1PCFG = 0xFFFF;           // all PORTB = Digital; RB2 = analog
AD1CON1 = 0x00E0;           // SSRC bit = 111 implies internal
                             // counter ends sampling and starts
                             // converting.
AD1CHS  = 0x00020000;        // Connect RB2/AN2 as CH0 input
                             // in this example RB2/AN2 is the input

AD1CSSL = 0;
AD1CON3 = 0x0203;           // Sample time = 2TAD

AD1CON2 = 0x6004;           // Select external VREF+ and VREF- pins
                             // Interrupt after every 2 samples
AD1CON1bits.ADON = 1;       // turn ADC ON
while (1)                   // repeat continuously
{
    ADCValue = 0;           // clear value
    ADC16Ptr = &ADC1BUF0;   // initialize ADC1BUF0 pointer
    IF1bits.AD1IF = 0;      // clear ADC interrupt flag
    AD1CON1bits.ASAM = 1;    // auto start sampling
                             // for 31TAD then go to conversion
    while (!IFS0bits.ADIF);  // conversion done?
    AD1CON1bits.ASAM = 0;    // yes then stop sample/convert
    for (count = 0; count < 2; count++)
    {
        ADCValue = ADCValue + *ADC16Ptr++;
        ADCValue = ADCValue >> 1;
    }
    // average the two
    // repeat
}
```

17.7 中断

ADC 具有专用的中断位 AD1IF 和相应的中断允许 / 屏蔽位 AD1IE。这些位用于决定中断源和使能 / 禁止各个中断源。每个通道的优先级还可以独立于其他通道进行设置。

当满足由每次中断采样数位 SMPI<3:0> (AD1CON2<5:2>) 设置的条件时, AD1IF 会置 1。AD1IF 位是否置 1 与相应 AD1IE 位的状态无关。如果需要, 可以用软件查询 AD1IF 位。

AD1IE 位用于控制中断产生。如果 AD1xIE 位置 1, 则每当发生 SMPI<3:0> 定义的事件时, CPU 均会中断, 并且相应的 AD1IF 位会置 1 (受下面所介绍的优先级和子优先级制约)。

处理特定中断的程序负责在服务程序完成之前清零相应的中断标志位。

ADC 中断的优先级可以通过 AD1IP<2:0> (IPC6<28:26>) 位独立设置。该优先级定义了中断源将分配到的优先级组。优先级组值的范围为 7 (最高优先级) 到 0 (不产生中断)。较高优先级组中的中断会抢占正在处理、但优先级较低的中断。

子优先级位用于设置中断源在优先级组中的优先级。子优先级位 AD1xIS<1:0> (IPC6<25:24>) 值的范围为 3 (最高优先级) 到 0 (最低优先级)。处于相同优先级组, 但具有更高子优先级值的中断会抢占子优先级较低、但正在进行的中断。

优先级组和子优先级位让多个中断源可以共用相同的优先级和子优先级。如果在该配置下同时发生若干个中断, 则中断源在优先级 / 子优先级组对中的自然顺序将决定所产生的中断。自然优先级基于中断源的向量编号。向量编号越小, 中断的自然优先级就越高。在当前中断的中断标志清零之后, 所有不按照自然顺序执行的中断会基于优先级、子优先级和自然顺序产生相应的中断。

产生允许的中断之后, CPU 将跳转到为该中断分配的向量处 (见表 17-9)。该中断的向量编号与自然顺序编号相同。由于一些中断共用单个向量, IRQ 编号并不总是与向量编号相同。然后, CPU 将在向量地址处开始执行代码。该向量地址处的用户代码应执行所需的操作 (如重新装入占空比和清零中断标志), 然后退出。关于向量地址表的详细信息和中断的更多信息, 请参见第 8 章 “中断” (DS61108)。

表 17-9: 各种偏移量的 ADC 中断向量 (EBASE = 0x8000:0000)

中断	向量 / 自然顺序	IRQ 编号	向量地址 IntCtl.VS = 0x01	向量地址 IntCtl.VS = 0x02	向量地址 IntCtl.VS = 0x04	向量地址 IntCtl.VS = 0x08	向量地址 IntCtl.VS = 0x10
ADC	27	32	8000 0560	8000 08C0	8000 0F80	8000 1D00	8000 3800

例 17-8: ADC 中断配置代码示例

```

IPS6SET = 0x0014;           // Set Priority to 5
IPS6SET = 0x0003;           // Set Sub Priority to 3
                               //
IFS1CLR = 0x0002;           // Ensure the interrupt flag is clear
IEC1SET = 0x0002;           // Enable ADC interrupts

```

17.8 I/O 引脚控制

用作模拟输入的引脚也可以用作数据 I/O。将引脚配置为模拟输入需要 3 个步骤。共用所需引脚的任何数字外设都必须禁止。引脚必须配置为数字输入，方法是通过将相应的 TRIS 位设置为 1 来禁止输出驱动器。然后，必须通过将AD1PCFG 寄存器中的相应位置1来将引脚置为模拟模式。

表 17-10: 与 ADC 模块相关的引脚

引脚名称	模块控制	控制位域	引脚类型	缓冲器类型	TRIS	说明
AN0	ON	AD1PCFG<0>	A	—	输入	模拟输入
AN1	ON	AD1PCFG<1>	A	—	输入	模拟输入
AN2	ON	AD1PCFG<2>	A	—	输入	模拟输入
AN3	ON	AD1PCFG<3>	A	—	输入	模拟输入
AN4	ON	AD1PCFG<4>	A	—	输入	模拟输入
AN5	ON	AD1PCFG<5>	A	—	输入	模拟输入
AN6	ON	AD1PCFG<6>	A	—	输入	模拟输入
AN7	ON	AD1PCFG<7>	A	—	输入	模拟输入
AN8	ON	AD1PCFG<8>	A	—	输入	模拟输入
AN9	ON	AD1PCFG<9>	A	—	输入	模拟输入
AN10	ON	AD1PCFG<10>	A	—	输入	模拟输入
AN11	ON	AD1PCFG<11>	A	—	输入	模拟输入
AN12	ON	AD1PCFG<12>	A	—	输入	模拟输入
AN13	ON	AD1PCFG<13>	A	—	输入	模拟输入
AN14	ON	AD1PCFG<14>	A	—	输入	模拟输入
AN15	ON	AD1PCFG<15>	A	—	输入	模拟输入
VREF+	ON	AD1CON2<15:13>	P	—	—	正参考电压
VREF-	ON	AD1CON2<15:13>	P	—	—	负参考电压

图注：

ST = 带 CMOS 电平的施密特触发器输入

I = 输入

O = 输出

A = 模拟

P = 电源

17.9 休眠和空闲模式下的操作

注： 在本手册中，对于特定模块中使用的功耗模式和器件使用的功耗模式进行了区分：例如，比较器的 **Sleep**（休眠）模式和 CPU 的 **SLEEP**（休眠）模式。为了指示所期望功耗模式的类型，模块功耗模式使用大写字母加小写字母（**Sleep**, **Idle**, **Debug**）（休眠、空闲和调试）来表示，器件功耗模式使用全大写字母（**SLEEP**, **IDLE**, **DEBUG**）（休眠、空闲和调试）来表示。

SLEEP（休眠）和 **IDLE**（空闲）模式有助于将转换噪声降至最小，因为 CPU、总线和其他外设的数字活动被减到最少。

17.9.1 不使用 RC ADC 时钟时的 CPU 休眠模式

当器件进入 **SLEEP**（休眠）模式时，模块的所有时钟源被关闭并保持为逻辑 0。

如果在一次转换过程中进入 **SLEEP**（休眠）模式，转换会中止，除非 **ADC** 将其内部 **RC** 时钟发生器作为时钟源。在从 **SLEEP**（休眠）模式退出时，转换器不会继续进行已部分完成的转换。

器件进入或退出 **SLEEP**（休眠）模式不会影响 **ADC** 寄存器的内容。

17.9.2 使用 RC ADC 时钟时的 CPU 休眠模式

如果将内部 **ADC RC** 振荡器设置为 **ADC** 时钟源（**ADRC** = 1），**ADC** 模块就可以在 **SLEEP**（休眠）模式下工作。这样做可以降低转换中的数字开关噪声。转换完成时，**DONE** 位将置 1，结果装入 **ADC** 结果缓冲区 **ADC1BUF**。

如果允许 **ADC** 中断（**AD1IE** = 1），**ADC** 中断发生时器件将从 **SLEEP**（休眠）模式唤醒。如果 **ADC** 中断的优先级大于当前 **CPU** 优先级，程序执行将在 **ADC** 中断服务程序执行后恢复。否则，程序将从将器件置为 **SLEEP**（休眠）模式的 **WAIT** 指令之后的指令处继续执行。

如果禁止 **ADC** 中断，即使 **ON** 位保持置 1，还是会禁止 **ADC** 模块。

为了将数字噪声对 **ADC** 模块操作的影响降至最低，用户应选择转换触发源以确保 **A/D** 转换可在 **SLEEP**（休眠）模式下进行。自动转换触发选项可用于 **SLEEP**（休眠）模式下的采样和转换（**SSRC**<2:0> = 111）。要使用自动转换选项，应在 **WAIT** 指令之前的指令中将 **ADC ON** 位置 1。

注： 为了使 **ADC** 模块可以在 **SLEEP**（休眠）模式下工作，必须将 **RC** 设置为 **ADC** 时钟源（**ADRC** = 1）。

17.9.3 CPU 空闲模式下的 ADC 操作

对于 **A/D** 转换器，**ADC SIDL** 位（**AD1CON1**<13>）用于选择模块在 **IDLE**（空闲）模式下是停止还是继续工作。如果 **ADC SIDL** = 0，则当器件进入 **IDLE**（空闲）模式时，模块将继续正常工作。如果允许 **ADC** 中断（**AD1IE** = 1），**ADC** 中断发生时器件将从 **IDLE**（空闲）模式唤醒。如果 **ADC** 中断的优先级大于当前 **CPU** 优先级，程序执行将在 **ADC** 中断服务程序执行后恢复。否则，程序将从将器件置为 **IDLE**（空闲）模式的 **WAIT** 指令之后的指令处继续执行。

如果 **ADC SIDL** = 1，则在 **IDLE**（空闲）模式下模块将停止。如果器件在一次转换过程中进入 **IDLE**（空闲）模式，转换将中止。在从 **IDLE**（空闲）模式退出时，转换器不会继续进行已部分完成的转换。

17.9.4 冻结对 ADC 操作的影响

如果在 ADC 执行转换时进入 **Freeze**（冻结）模式，则转换结果将会丢失。

处于 **Freeze**（冻结）模式时，可以读取 ADC 寄存器。在处于 **Freeze**（冻结）模式时，对 ADC 寄存器的任意写操作都不会起效，直到退出 **Freeze**（冻结）模式之后才会起效。

注：	只有 CPU 在调试异常模式下执行时，FRZ 位才可读写。在所有其他模式下，FRZ 位读为 0。如果 FRZ 位在 DEBUG （调试）模式期间发生改变，则只有退出当前调试异常模式并重新进入该模式之后，新值才会生效。在调试异常模式期间，在进入 DEBUG （调试）模式时 FRZ 位会读取外设状态。
-----------	---

17.10 各种复位的影响

17.10.1 $\overline{\text{MCLR}}$ 复位

在发生 $\overline{\text{MCLR}}$ 事件之后，所有 ADC 控制寄存器（AD1CON1、AD1CON2、AD1CON3、AD1CHS、AD1PCFG 和 AD1CSSL）都会复位为值 0x00000000。这会禁止 ADC，并将模拟输入引脚设置为模拟输入模式。正在进行的转换将会终止，结果不会被写入结果缓冲区。

在 $\overline{\text{MCLR}}$ 复位期间，会对 ADC1BUF 寄存器中的值进行初始化。ADC1BUF0...ADC1BUFF 将包含 0x00000000。

17.10.2 上电复位

在发生 POR 事件之后，所有 ADC 控制寄存器（AD1CON1、AD1CON2、AD1CON3、AD1CHS、AD1PCFG 和 AD1CSSL）都会复位为值 0x00000000。这会禁止 ADC，并将模拟输入引脚设置为模拟输入模式。

在上电复位期间，会对 ADC1BUF 寄存器中的值进行初始化。ADC1BUF0...ADC1BUFF 将包含 0x00000000。

17.10.3 看门狗定时器复位

在发生看门狗定时器（WDT）复位之后，所有 ADC 控制寄存器（AD1CON1、AD1CON2、AD1CON3、AD1CHS、AD1PCFG 和 AD1CSSL）都会复位为值 0x00000000。这会禁止 ADC，并将模拟输入引脚设置为模拟输入模式。正在进行的转换将会终止，结果不会被写入结果缓冲区。

WDT 复位之后，会对 ADC1BUF 寄存器中的值进行初始化。

ADC1BUF0...ADC1BUFF 将包含 0x00000000。

17.11 设计技巧

问 1: *如何优化 A/D 转换器的系统性能?*

答 1: 以下技巧有助于优化性能:

1. 请确保满足了所有时序规范要求。如果关闭模块后再打开, 必须等待一个最小延时后再开始采样。如果改变了输出通道, 同样需要等待一个最小延时。还有 T_{AD} , 它是为每一位的转换所选择的时间。该时间值在 $AD1CON3$ 中选择, 且应在电气特性指定的范围内。如果 T_{AD} 太短, 转换终止时有可能还未对结果进行完全转换; 而如果 T_{AD} 太长, 采样电容上的电压会在转换结束前衰减。在器件数据手册的“电气规范”章节中提供了这些时序规范。
2. 模拟信号源阻抗经常很高 (大于 $10\text{ k}\Omega$), 因此从模拟信号源为采样电容充电的电流可能会影响精度。如果输入信号的变化不是太快, 可以尝试在模拟输入端连接一个 $0.1\text{ }\mu\text{F}$ 的电容。该电容可充电到所采样的模拟电压, 并为 4.4 pF 的内部保持电容提供充电所需的瞬态电流。
3. 在启动 A/D 转换前使器件进入 SLEEP (休眠) 模式。在 SLEEP (休眠) 模式下的转换需要选择 RC 时钟源。这种技术可以提高精度, 因为来自 CPU 和其他外设的数字噪声被降至最小。

问 2: *是否可以推荐关于 ADC 的优秀参考资料?*

答 2: 以下手册可以帮助更好地理解 A/D 转换:

Analog Devices, Inc., and Scheingold, D. H., ed. *Analog-Digital Conversion Handbook*. 3rd ed., Englewood Cliffs, NJ: Prentice Hall, 1986. ISBN 0-13-032848-0.

问 3: *当通道/采样和采样/中断组合超出缓冲区大小时, 缓冲区会发生什么情况?*

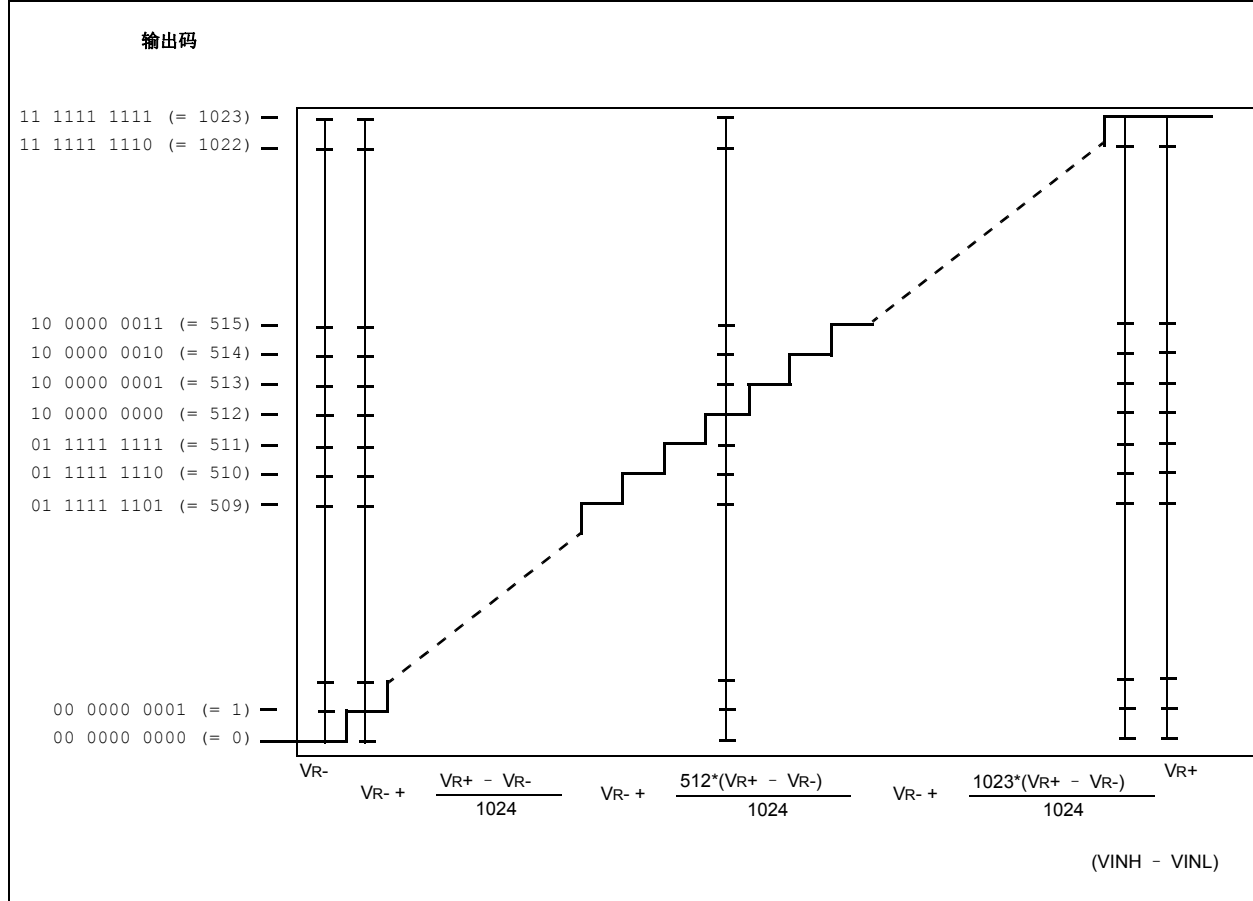
答 3: 不建议这种配置。缓冲区将包含转换序列中前 16 次采样 (或者 8 次, 如果使用双缓冲区模式) 的结果。转换序列中的其他项目会被忽略。

17.11.1 传递函数

图 17-20 给出了 A/D 转换器的理想传递函数。输入电压差值 ($V_{INH} - V_{INL}$) 与参考电压差值 ($V_{R+} - V_{R-}$) 进行比较。

- 当输入电压为 ($V_{R+} - V_{R-}/2048$) 或 0.5 LSb 时, 发生第一个代码转换。
- 00 0000 0001 代码中点位于 ($V_{R+} - V_{R-}/1024$) 或 1.0 LSb。
- 10 0000 0000 代码中点位于 ($512 \times (V_{R+} - V_{R-})/1024$)。
- 小于 ($1 \times (V_{R+} - V_{R-})/2048$) 的输入电压被转换为 00 0000 0000。
- 大于 ($2045 \times (V_{R+} - V_{R-})/2048$) 的输入电压被转换为 11 1111 1111。

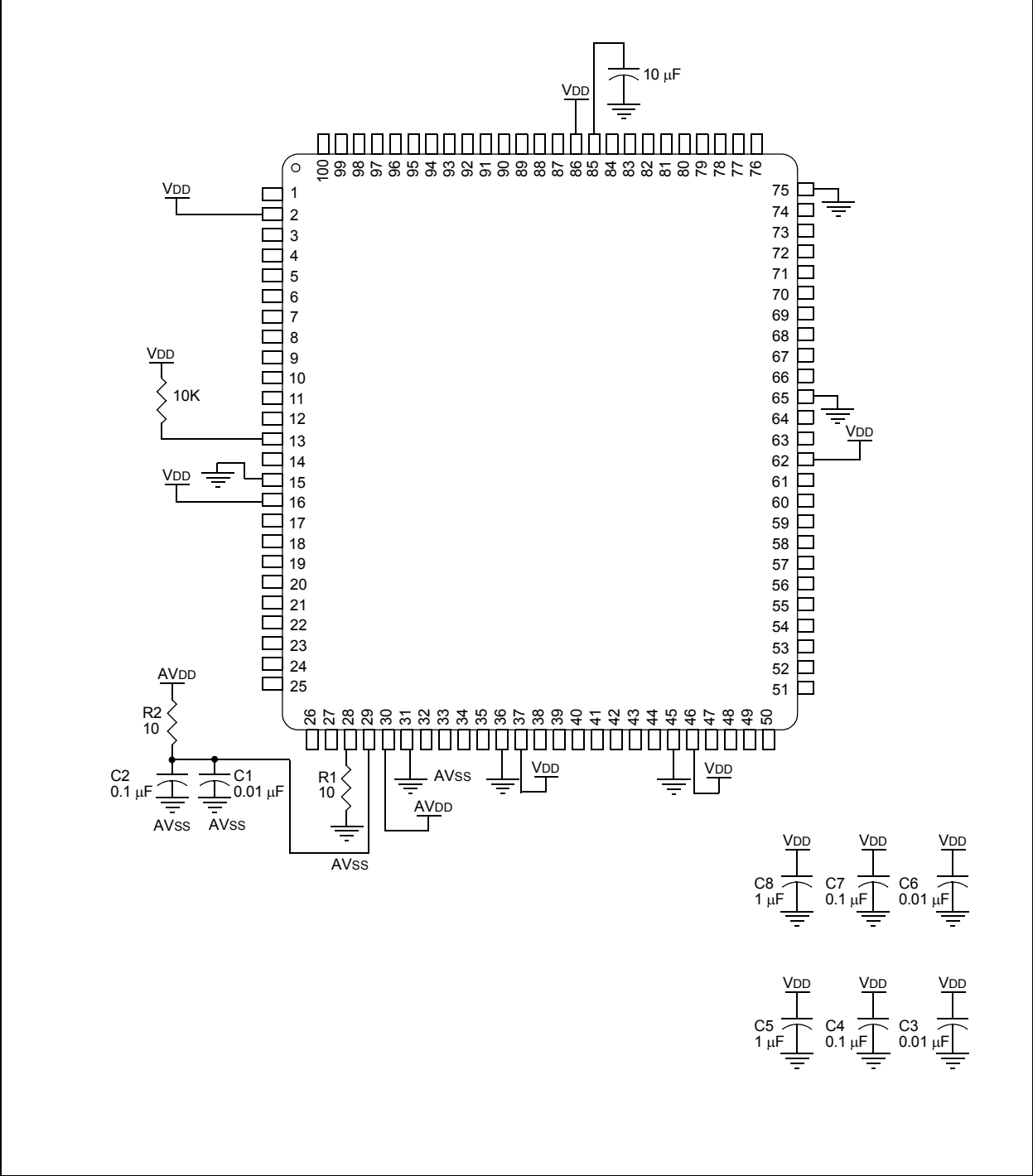
图 17-20: ADC 传递函数



17.11.2 ADC 精度 / 误差

请参见第 17.12 节 “相关应用笔记” 获取讨论 ADC 精度的文档列表。
下图给出了转换速率高于 400 ksps 时的推荐电路。以 PIC32MX 为例。

图 17-21: A/D 转换器参考电压原理图



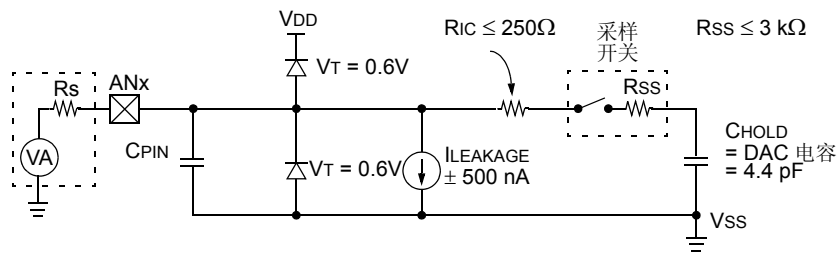
17.11.3 ADC 采样要求

图 17-22 给出了 10 位 A/D 转换器的模拟输入模型。A/D 转换的总采集时间是内部放大器稳定时间和保持电容充电时间的函数。

为了使 A/D 转换器达到规定的精度，必须让充电保持电容（CHOLD）充分充电至模拟输入引脚的电压。模拟输出信号源阻抗（ R_s ）、片内走线等效电阻（ R_{ic} ）和内部采样开关阻抗（ R_{ss} ）共同直接影响着 CHOLD 充电所需的时间。因此模拟信号源的总阻抗必须足够小，以便在选择的采样时间内对保持电容充分充电。选择（改变）了模拟输入通道后，采集工作必须在启动转换前完成。在每次采样操作之前，内部保持电容将处于放电状态。

两次转换之间应留出至少 1 个 TAD 时间段作为采集时间。更多详细信息，请参见器件电气规范。

图 17-22: 10 位 A/D 转换器模拟输入模型



注：CPIN 值取决于器件封装，未经测试。如果 $R_s \leq 5 \text{ k}\Omega$ ，CPIN 的影响可忽略。

图注：

CPIN = 输入电容

R_{ss} = 采样开关电阻

R_s = 信号源阻抗

ILEAKAGE = 各连接点在引脚上产生的泄漏电流

V_T = 门限电压

R_{ic} = 片内走线等效电阻

CHOLD = 采样 / 保持电容

17.11.4 连接注意事项

因为模拟输入采用静电放电（Electrostatic Discharge, ESD）保护，它们通过二极管连接到 V_{DD} 和 V_{SS} 。这就要求模拟输入电压必须介于 V_{DD} 和 V_{SS} 之间。如果输入电压超出此范围 0.3V 以上（任一方向上），就会有一个二极管正向偏置，如果超出输入电流规范可能会损坏器件。

有时可以通过外接一个 RC 滤波器来对输入信号进行抗混叠滤波。应选择合适的 R 元件以确保达到采集时间要求。任何通过高阻抗连接到模拟输入引脚上的外部元件（如电容和齐纳二极管等）在引脚上的泄漏电流都应极小。

17.12 相关应用笔记

本节列出了与手册本章内容相关的应用笔记。这些应用笔记可能并不是专为 PIC32MX 器件系列而编写的，但其概念是相近的，通过适当修改并受到一定限制即可使用。当前与 10 位 A/D 转换器模块相关的应用笔记有：

标题	应用笔记编号
Using the Analog-to-Digital (A/D) Converter	AN546
Four Channel Digital Voltmeter with Display and Keyboard	AN557
理解 A/D 转换器的性能参数	AN693
dsPIC30F 在无传感器 BLDC 控制中的应用	AN901
使用 dsPIC30F 实现交流感应电机的矢量控制	AN908
使用 dsPIC30F2010 控制带传感器的 BLDC 电机	AN957
使用 dsPIC30F MCU 控制交流感应电机	AN984

注： 如需获取更多 PIC32MX 系列器件的应用笔记和代码示例，请访问 Microchip 网站 (www.microchip.com)。

17.13 版本历史

版本 A（2007 年 10 月）

这是本文档的初始版本。

版本 B（2007 年 10 月）

更新了文档（删除了“机密”状态）。

版本 C（2008 年 4 月）

将状态修改为“初稿”；将 U-0 修改为 r-x。

版本 D（2008 年 6 月）

修改了寄存器 17-1 的注释；修改了寄存器 17-13、17-17、17-21、17-25 和 17-26；修改了公式 17-1；增加了第 17.5.6 节；修改了表 17-4、17-5、17-6、17-7 和 17-8；删除了第 17.11.5 节（500 KSPS 配置指南）；将保留位从“保持为”更改为“写入”；为 ON 位（AD1CON1 寄存器）增加了注释。

请注意以下有关 Microchip 器件代码保护功能的要点：

- Microchip 的产品均达到 Microchip 数据手册中所述的技术指标。
- Microchip 确信：在正常使用的情况下，Microchip 系列产品是当今市场上同类产品中最安全的产品之一。
- 目前，仍存在着恶意、甚至是非法破坏代码保护功能的行为。就我们所知，所有这些行为都不是以 Microchip 数据手册中规定的操作规范来使用 Microchip 产品的。这样做的人极可能侵犯了知识产权。
- Microchip 愿与那些注重代码完整性的客户合作。
- Microchip 或任何其他半导体厂商均无法保证其代码的安全性。代码保护并不意味着我们保证产品是“牢不可破”的。

代码保护功能处于持续发展中。Microchip 承诺将不断改进产品的代码保护功能。任何试图破坏 Microchip 代码保护功能的行为均可视为违反了《数字器件千年版权法案（Digital Millennium Copyright Act）》。如果这种行为导致他人在未经授权的情况下，能访问您的软件或其他受版权保护的成果，您有权依据该法案提起诉讼，从而制止这种行为。

提供本文档的中文版本仅为了便于理解。请勿忽视文档中包含的英文部分，因为其中提供了有关 Microchip 产品性能和使用情况的有用信息。Microchip Technology Inc. 及其分公司和相关公司、各级主管与员工及事务代理机构对译文中可能存在的任何差错不承担任何责任。建议参考 Microchip Technology Inc. 的英文原版文档。

本出版物中所述的器件应用信息及其他类似内容仅为您提供便利，它们可能由更新之信息所替代。确保应用符合技术规范，是您自身应负的责任。Microchip 对这些信息不作任何明示或暗示、书面或口头、法定或其他形式的声明或担保，包括但不限于针对其使用情况、质量、性能、适销性或特定用途的适用性的声明或担保。Microchip 对因这些信息及使用这些信息而引起的后果不承担任何责任。如果将 Microchip 器件用于生命维持和 / 或生命安全应用，一切风险由买方自负。买方同意在由此引发任何一切伤害、索赔、诉讼或费用时，会维护和保障 Microchip 免于承担法律责任，并加以赔偿。在 Microchip 知识产权保护下，不得暗中或以其他方式转让任何许可证。

商标

Microchip 的名称和徽标组合、Microchip 徽标、dsPIC、KEELOQ、KEELOQ 徽标、MPLAB、PIC、PICmicro、PICSTART、PIC³² 徽标、rfPIC 和 UNI/O 均为 Microchip Technology Inc. 在美国和其他国家或地区的注册商标。

FilterLab、Hampshire、HI-TECH C、Linear Active Thermistor、MXDEV、MXLAB、SEEVAL 和 The Embedded Control Solutions Company 均为 Microchip Technology Inc. 在美国的注册商标。

Analog-for-the-Digital Age、Application Maestro、CodeGuard、dsPICDEM、dsPICDEM.net、dsPICworks、dsSPEAK、ECAN、ECONOMONITOR、FanSense、HI-TIDE、In-Circuit Serial Programming、ICSP、Mindi、MiWi、MPASM、MPLAB Certified 徽标、MPLIB、MPLINK、mTouch、Omniscient Code Generation、PICC、PICC-18、PICDEM、PICDEM.net、PICKit、PICKtail、REAL ICE、rFLAB、Select Mode、Total Endurance、TSHARC、UniWinDriver、WiperLock 和 ZENA 均为 Microchip Technology Inc. 在美国和其他国家或地区的商标。

SQTP 是 Microchip Technology Inc. 在美国的服务标记。

在此提及的所有其他商标均为各持有公司所有。

© 2010, Microchip Technology Inc. 版权所有。

ISBN: 978-1-60932-532-9

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
== ISO/TS 16949:2002 ==

Microchip 位于美国亚利桑那州 Chandler 和 Tempe 与位于俄勒冈州 Gresham 的全球总部、设计和晶圆生产厂及位于美国加利福尼亚州和印度的设计中心均通过了 ISO/TS-16949:2002 认证。公司在 PIC[®] MCU 与 dsPIC[®] DSC、KEELOQ[®] 跳码器件、串行 EEPROM、单片机外设、非易失性存储器和模拟产品方面的质量体系流程均符合 ISO/TS-16949:2002。此外，Microchip 在开发系统的设计和生产方面的质量体系也已通过了 ISO 9001:2000 认证。

全球销售及服务中心

美洲

公司总部 **Corporate Office**
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 1-480-792-7200
Fax: 1-480-792-7277

技术支持:
<http://support.microchip.com>
网址: www.microchip.com

亚特兰大 Atlanta

Duluth, GA
Tel: 1-678-957-9614
Fax: 1-678-957-1455

波士顿 Boston

Westborough, MA
Tel: 1-774-760-0087
Fax: 1-774-760-0088

芝加哥 Chicago

Itasca, IL
Tel: 1-630-285-0071
Fax: 1-630-285-0075

克里夫兰 Cleveland

Independence, OH
Tel: 1-216-447-0464
Fax: 1-216-447-0643

达拉斯 Dallas

Addison, TX
Tel: 1-972-818-7423
Fax: 1-972-818-2924

底特律 Detroit

Farmington Hills, MI
Tel: 1-248-538-2250
Fax: 1-248-538-2260

科科莫 Kokomo

Kokomo, IN
Tel: 1-765-864-8360
Fax: 1-765-864-8387

洛杉矶 Los Angeles

Mission Viejo, CA
Tel: 1-949-462-9523
Fax: 1-949-462-9608

圣克拉拉 Santa Clara

Santa Clara, CA
Tel: 1-408-961-6444
Fax: 1-408-961-6445

加拿大多伦多 Toronto

Mississauga, Ontario, Canada
Tel: 1-905-673-0699
Fax: 1-905-673-6509

亚太地区

亚太总部 Asia Pacific Office

Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

中国 - 北京

Tel: 86-10-8528-2100
Fax: 86-10-8528-2104

中国 - 成都

Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

中国 - 重庆

Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

中国 - 香港特别行政区

Tel: 852-2401-1200
Fax: 852-2401-3431

中国 - 南京

Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

中国 - 青岛

Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

中国 - 上海

Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

中国 - 沈阳

Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

中国 - 深圳

Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

中国 - 武汉

Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

中国 - 西安

Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

中国 - 厦门

Tel: 86-592-238-8138
Fax: 86-592-238-8130

中国 - 珠海

Tel: 86-756-321-0040
Fax: 86-756-321-0049

台湾地区 - 高雄

Tel: 886-7-213-7830
Fax: 886-7-330-9305

台湾地区 - 台北

Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

亚太地区

台湾地区 - 新竹

Tel: 886-3-6578-300
Fax: 886-3-6578-370

澳大利亚 Australia - Sydney

Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

印度 India - Bangalore

Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

印度 India - New Delhi

Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

印度 India - Pune

Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

日本 Japan - Yokohama

Tel: 81-45-471-6166
Fax: 81-45-471-6122

韩国 Korea - Daegu

Tel: 82-53-744-4301
Fax: 82-53-744-4302

韩国 Korea - Seoul

Tel: 82-2-554-7200
Fax: 82-2-558-5932 或
82-2-558-5934

马来西亚 Malaysia - Kuala Lumpur

Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

马来西亚 Malaysia - Penang

Tel: 60-4-227-8870
Fax: 60-4-227-4068

菲律宾 Philippines - Manila

Tel: 63-2-634-9065
Fax: 63-2-634-9069

新加坡 Singapore

Tel: 65-6334-8870
Fax: 65-6334-8850

泰国 Thailand - Bangkok

Tel: 66-2-694-1351
Fax: 66-2-694-1350

欧洲

奥地利 Austria - Wels

Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

丹麦 Denmark - Copenhagen

Tel: 45-4450-2828
Fax: 45-4485-2829

法国 France - Paris

Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

德国 Germany - Munich

Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

意大利 Italy - Milan

Tel: 39-0331-742611
Fax: 39-0331-466781

荷兰 Netherlands - Druenen

Tel: 31-416-690399
Fax: 31-416-690340

西班牙 Spain - Madrid

Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

英国 UK - Wokingham

Tel: 44-118-921-5869
Fax: 44-118-921-5820

07/15/10