

AN14485

i.MX RT1180上的ADC的使用

第1.0版—2025年2月7日

应用笔记

文档信息

信息	内容
关键词	AN14485、i.MX RT1180、ADC、ADCK、DMA、SAR
摘要	本文档详细介绍了如何配置、实现和使用恩智浦i.MX RT1180微控制器（MCU）上的模数转换器。



1 介绍

本文档说明如何配置、实现和使用恩智浦i.MX RT1180微控制器（MCU）上的模数转换器。

i.MX RT1180 MCU内置了两个模数转换器（ADC）模块：ADC1和ADC2。每个ADC模块包含两个逐次逼近寄存器（SAR）转换模块，分别称为A侧和B侧。也就是说，每个ADC模块拥有一个双SAR转换器，能够同时对两个单端信号进行采样和转换（Sample/Convert，简称S/C）。

每个ADC模块有8个触发输入，连接到XBAR1的输出。每个触发输入上的信号都能启动一次转换序列。如果高优先级的触发输入端接收到信号，它可以打断由低优先级触发输入触发的转换序列。每个ADC模块还配有两个FIFO（先进先出存储器）用来存储转换结果，同时附带一些状态标志。每次转换的结果最多可以占用一个FIFO条目的16位。每个FIFO条目的宽度为32位，最多可存储16个条目。

ADC1有16个外部通道（来自SoC引脚），而ADC2有14个外部通道。

2 转换序列的实现

在i.MX RT1180 MCU中，每个ADC模块包含三个主要组成部分（或概念）：

- **触发输入（触发器或TRIG）端口：**

这些输入端口连接到i.MX RT1180的XBAR1模块的XBAR_OUT140 – XBAR_OUT147信号。在这种情况下，内部信号（例如来自PWM或定时器模块）或外部信号（例如来自芯片引脚）都可以方便地用于触发模数（AD）转换。每个ADC模块拥有8个TRIG端口：TRIG0 – TRIG7。

- **AD通道：**

在i.MX RT1180中，可以为ADC1分配16个引脚作为通道（CH0A – CH7A, CH0B – CH7B）。同样，可以为ADC2分配14个引脚作为通道（CH0A – CH6A, CH0B – CH6B）。

注：

- 通道名称带“A”后缀的属于A侧转换器。
- 通道名称带“B”后缀的属于B侧转换器。

- **命令缓冲区（CMD）：**

命令缓冲区是一个64位的缓冲区，用于存储AD转换相关的多种配置信息。每个ADC模块有15个命令缓冲区：CMD1 – CMD15。每个CMD分为两部分：CMDH（命令缓冲区的高32位）和CMDL（命令缓冲区的低32位）。

以下两个寄存器位字段存储与AD转换相关的重要信息：

- CMDLm[ADCH]：当前CMDL寄存器的ADCH位字段指明了待转换的AD通道。
- CMDHm[NEXT]：当前CMDH寄存器的NEXT位字段指明了当前CMD转换完成后，下一次转换序列要使用的CMD序号。

每个TRIG输入可以关联一个CMD。当TRIG输入端口检测到上升沿信号时，ADC模块开始转换与该TRIG输入端口关联的CMD中包含的AD通道。

当该通道的转换完成后，ADC模块会根据当前CMDH寄存器中NEXT位字段值的指引，跳转到下一个CMD，并开始转换该CMD中包含的AD通道。

这个转换过程会持续进行，直到当前CMDH寄存器中的NEXT位字段值为0。当满足此条件时，AD转换停止。这个完整的转换过程被称为某个TRIG输入端口的转换序列。

[图1](#)显示了一个转换序列。

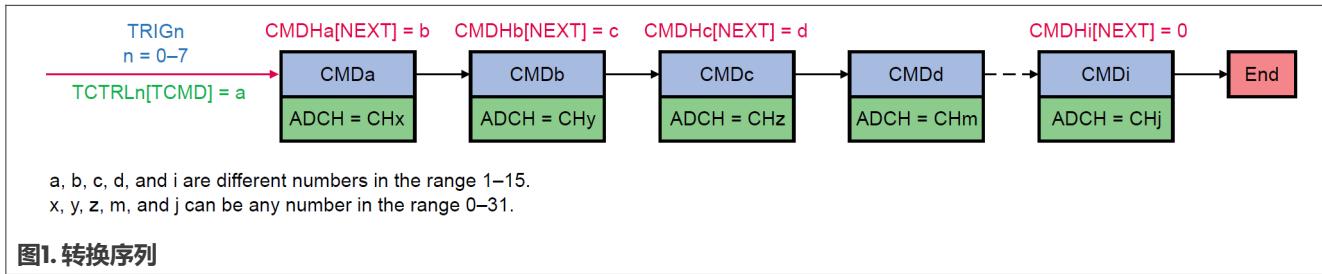


图1. 转换序列

图1中：

- TCTRLn (n = 0–7) 是用于配置TRIGn输入的寄存器。TCTRLn[TCMD]位字段确定与TRIGn关联的第一个CMD。例如, TCTRL5[TCMD] = 3表示当TRIG5端口检测到上升沿信号时, AD转换从CMD3开始。
- CMDHm和CMDLm (m = 1–15) 是CMD缓冲区寄存器, 其中:
- CMDLm[ADCH]表示CMDm中包含的AD通道号。例如, CMDL3[ADCH] = 5表示CMD3包含通道CH5A、CH5B或CH5A/CH5B对。详情参见[2.2.3节](#)。
- CMDHm[NEXT]表示CMDm中的通道AD转换完成后要使用的下一个CMD。例如, CMDH3[NEXT] = 1表示在CMD3中标识的通道转换完成后立即使用CMD1进行AD转换。

2.1 AD转换序列类型

基于CMDHm[NEXT]的值, 转换序列可以分为以下几种类型:

2.1.1 顺序转换

在顺序转换中:

- 没有CMD指向自身或先前使用过的CMD。例如, CMDH3[NEXT] = 3表示CMD3指向自身。
- 序列中最后一个CMD的CMDH寄存器的NEXT位字段的值为0。

顺序转换在所有CMD执行完成后自动结束。这种执行流程在转换序列中应用广泛。图2显示了一个顺序转换的示例。

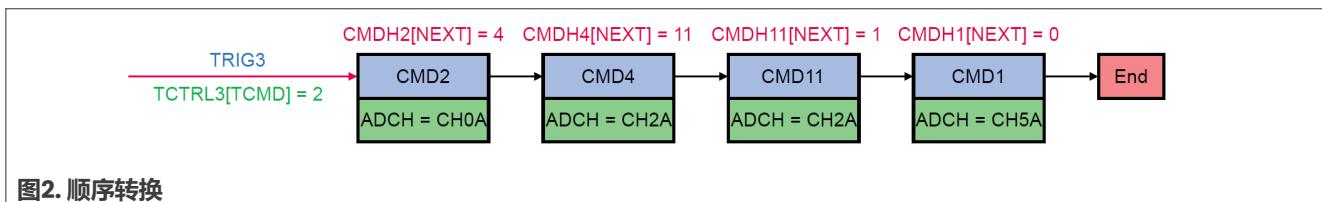


图2. 顺序转换

在图2中:

- 如果TRIG3检测到有效信号, 使用的第一个CMD将是CMD2, 因为TCTRL3[TCMD] = 2。由于CMD2中配置了CH0A, 因此该序列中第一个被转换的通道是CH0A。
- 通过每个CMD中CMDH寄存器的NEXT位字段的配置, 控制序列最终到达CMD1, 其中CMDH1[NEXT] = 0。因此, 在CMD1中的CH5A完成转换后, 整个转换序列结束。

每次TRIG3上出现有效信号时, 上述转换序列都会被执行一次。

注: 转换序列中CMD的执行顺序由CMD缓冲区CMDH寄存器的NEXT位字段决定。

2.1.2 无终止循环转换

如果顺序转换配置中最后一个CMD的CMDH寄存器的NEXT位字段指向转换序列中的第一个CMD，则该序列在触发后会无限运行下去。这种类型的转换称为“无终止循环转换”。

[图3](#)显示了无终止循环转换的示例。

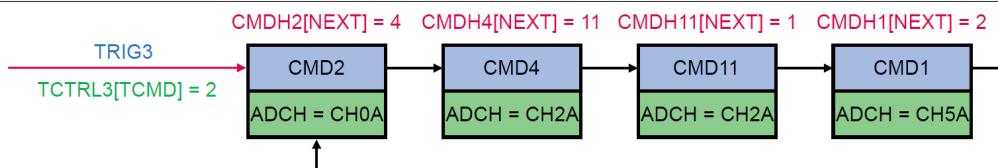


图3. 无终止循环转换

在[图3](#)中，最后一个CMD是CMD1。对于CMD1，其CMDH寄存器(CMDH1)的NEXT位字段值为2，即NEXT位字段指向序列中的第一个CMD：CMD2。这个序列会无限运行下去，直到被更高优先级TRIG上的有效信号中断或终止。

2.1.3 连续转换

当某个CMD指向自身（由当前CMDH寄存器的NEXT位字段指示）时，该CMD中指定的通道会反复进行转换，直到该CMD被中断或终止。这种类型的转换称为“连续转换”。

[图4](#)显示了连续转换的示例。

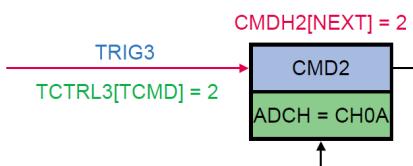


图4. 连续转换

2.2 通过CMD来配置一个序列中的每次转换

在转换序列中，每次转换可以通过CMD来单独配置。例如，可以为某个CMD中指定的通道启用“硬件平均”功能，而其他CMD中的通道则禁用此功能。

“硬件平均”和循环功能可以在某个CMD中的通道同时启用。此外，某些通道可能需要其他功能，例如比较功能。所有这些设置均可在CMDLm和CMDHm (m = 1-15) 寄存器中进行配置。

2.2.1 硬件平均

“硬件平均”功能可以在转换序列中的某个CMD上启用或禁用。[图5](#)演示了“硬件平均”功能。

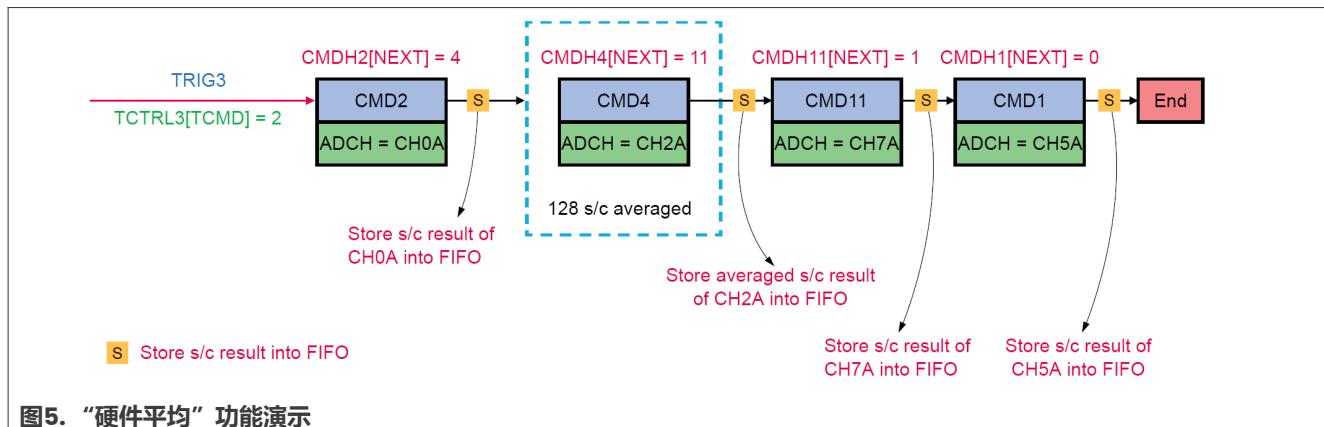


图5. “硬件平均”功能演示

如图5所示，当CMD2中的通道CH0A完成转换后，由于CMDH2寄存器的NEXT位字段指向CMD4，接下来将开始转换CMD4中的CH2A。

由于CMD4启用了“硬件平均”功能 (CMDH4[AVGS] = 7)，CH2A被转换 $2^7 = 128$ 次，并将转换结果的平均值存储在FIFO中。在计算平均值的情况下，一个通道最多可以转换1024次。

2.2.2 循环功能

在转换序列中某个CMD被触发后，可设置CMDH寄存器的LOOP位字段 (CMDHm[LOOP]) 来启用对该CMD指定通道的多次采样和转换。这一功能被称为“循环功能”。

当CMDHm[LOOP] = N时，CMDm中的通道被采样和转换N+1次，并且每次转换的结果都存储在结果FIFO中。N的最大值为15，即最多允许16次循环。

循环功能与硬件平均功能的一个主要区别在于：在循环功能中，每次转换的结果都会存储在结果FIFO中，而硬件平均功能仅将平均后的结果存储在FIFO中。硬件平均可以视为一种原子操作，通常不可中断，且仅生成一个结果。

图6演示了循环功能。

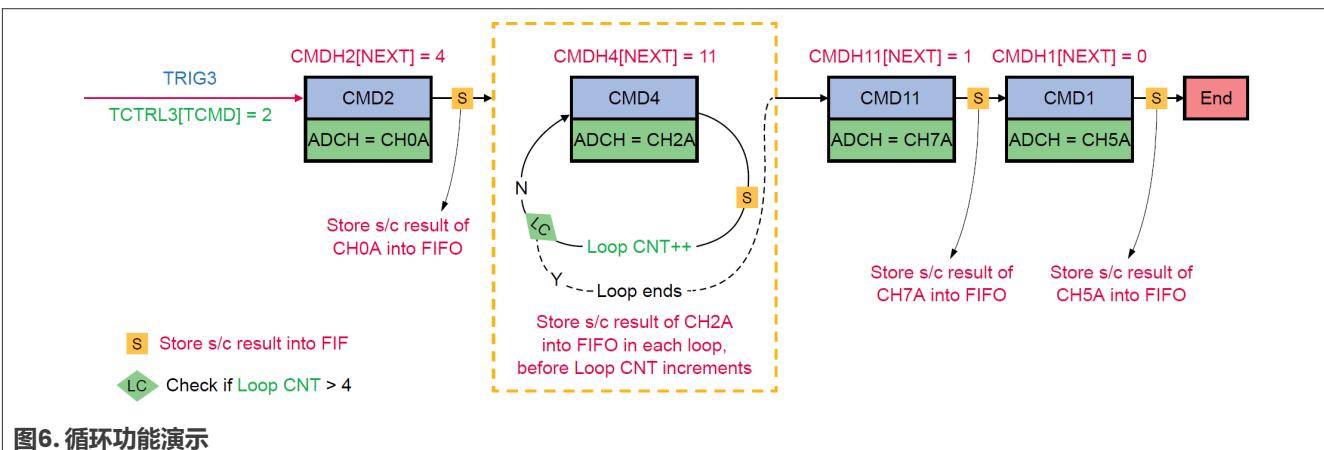


图6. 循环功能演示

在图6中，ADC模块使用硬件计数器Loop CNT来记录循环的次数。CMD4启用了循环功能，因此，CMD4中指定的通道被多次采样和转换。每次转换完成后，结果存储在结果FIFO中，同时Loop CNT加1。

当Loop CNT > CMDH4[LOOP]时，循环结束，序列跳转至CMD11执行。由于CMDH4[LOOP] = 4，CMD4中的通道CH2A被采样和转换5次，每次转换的结果都存储在FIFO中。当CMDHm[LOOP] = 0时，CMDHm中的通道仅转换一次（循环功能被禁用）。

2.2.3 转换类型

ADC通道以成对的方式命名，如CH0A/CH0B和CH1A/CH1B。对于每个CMD，可能出现四种类型的转换：

- **CMDLm[CTYPE] = 0:** 单端模式，A侧通道转换
- **CMDLm[CTYPE] = 1:** 单端模式，B侧通道转换。

在这一转换类型中：

- 当CMDLm[ALTBEN] = 0时，B侧通道由CMDLm[ADCH]选择。
- 当CMDLm[ALTBEN] = 1时，B侧通道由CMDLm[ALTB_ADCH]选择。

- **CMDLm[CTYPE] = 2:** 差分模式。在这一转换类型中，转换的是A侧通道和B侧通道之间的电压之差（通道A到通道B的电压）。在这种情况下，A侧和B侧必须具有匹配的通道号，例如，CH0A/CH0B和CH1A/CH1B。

- **CMDLm[CTYPE] = 3:** 双单端模式。在这一转换类型中，A侧和B侧通道同时进行转换。在这种情况下，A侧和B侧的通道编号不需要满足匹配的要求，例如可以是CH1A和CH4B，或者CH2A和CH2B。

在这种转换类型下：

- 如果CMDLm[ALTBEN] = 0，A侧和B侧的通道都由CMDLm[ADCH]选择。这意味着A侧和B侧通道是成对的，例如CH0A/CH0B和CH1A/CH1B。

- 当CMDLm[ALTBEN] = 1时，A侧通道由CMDLm[ADCH]选择，而B侧通道由CMDLm[ALTB_ADCH]选择。

在这种情况下，A侧和B侧可以具有不匹配的通道号，例如，CH1A/CH4B，CH2A/CH2B和CH5A/CH3B。

图7显示了转换序列中不同CMD使用的不同转换类型。

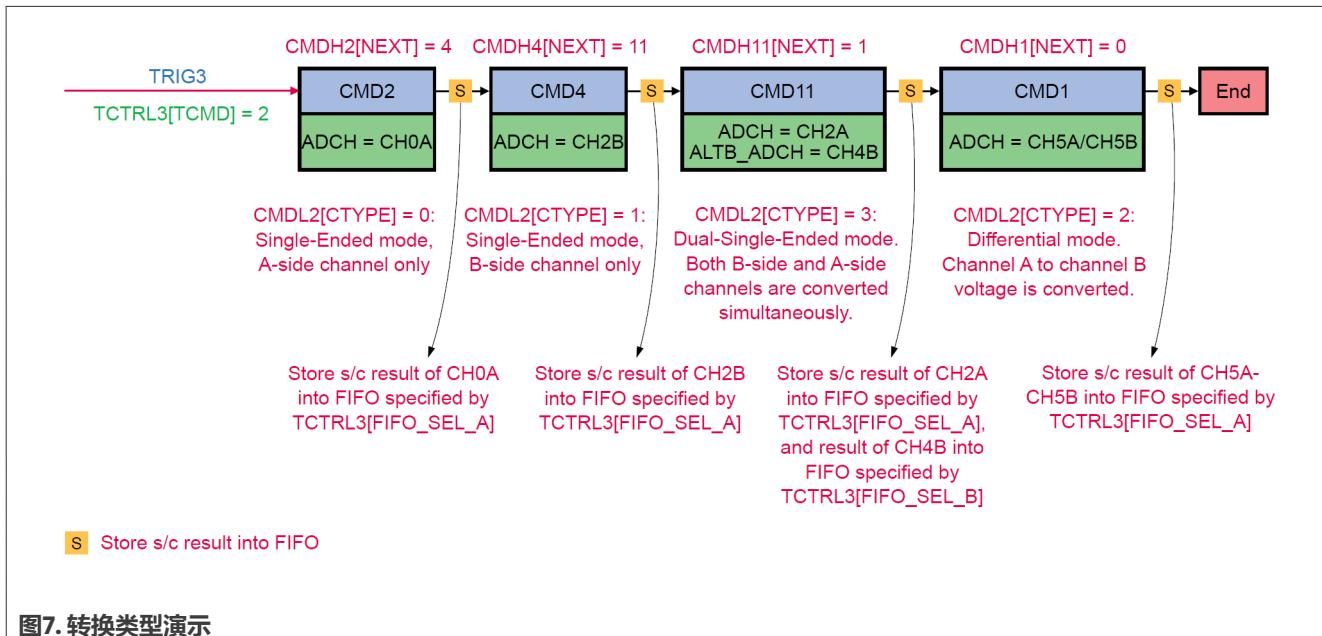


图7. 转换类型演示

在图7中：

- 第一个CMD是CMD2，转换类型是单端模式，使用A侧通道进行转换（CTYPE = 0），通道是CH0A。

- 第二个CMD是CMD4，转换类型是单端模式，使用B侧通道进行转换 (CTYPE = 1)，通道是CH2B。
- 第三个CMD是CMD11，转换类型是双单端模式 (CTYPE = 3)。要同时采样和转换的通道是CH2A和CH4B。因为A侧和B侧的通道编号不匹配，所以：
 - CMDL11[ALTBEN] = 1
 - B侧通道由CMDL11[ALTB_ADCH]选择。
- 最后一个CMD是CMD1，转换类型是差分模式 (CTYPE = 2)。在这种情况下，转换的是CH5A和CH5B之间的电压差。

2.2.4 结果 FIFO

每个 ADC 模块配备两个 32 位宽的 FIFO，用于存储转换结果：FIFO0 和 FIFO1。每个 FIFO 可存储 16 个 (32 位) 条目。

当TRIGN信号 ($n = 0-7$) 触发一个转换序列后，会通过TCTRLn[FIFO_SEL_A]和TCTRLn[FIFO_SEL_B]选择一个 FIFO 来存储该转换序列的结果：

- 对于采用双单端模式的 CMD (CMDLm[CTYPE] = 3)，A 侧通道的转换结果存储在 TCTRLn[FIFO_SEL_A] 指定的 FIFO 中，B 侧通道的转换结果存储在 TCTRLn[FIFO_SEL_B] 指定的 FIFO 中。
如果 FIFO_SEL_A = FIFO_SEL_B，则 A 侧和 B 侧通道的结果会存储在同一个 FIFO 中，且 A 侧通道的结果会先于 B 侧通道的结果存储。
- 对于非双单端模式的 CMD (CMDLm[CTYPE] != 3)，转换结果始终存储在 TCTRLn[FIFO_SEL_A] 指定的 FIFO 中。

图7显示了转换序列中每个 CMD 使用的 FIFO。由于只有两个 FIFO 用于存储所有的转换结果，因此需要知道与结果相关的 TRIG 转换和 CMD。因此，每个 FIFO 条目除了包含转换结果外，还包含了与 TRIGn、CMDm 和循环计数 (Loop CNT) 相关的信息。这些信息有助于确定与结果相关的转换序列、CMD 和循环时间。

图8展示了结果 FIFO 的结构。



图8. 具有一个32位条目的结果FIFO

FIFO 条目的低 16 位用于存储转换结果。高 (上部) 16 位用于存储相关的 TRIG 编号、CMD 编号和循环计数值。如第2.2.2节所述，在循环中的每次转换完成且循环计数递增之前，CMD 编号、循环计数、TRIG 编号和结果将被存储在指定的 FIFO 中。

2.2.5 如果CMDHm[LWI] = 0，通道编号在循环过程中保持不变

如果 CMDHm[LWI] = 0，则 CMDHm 指定的通道编号在循环过程中保持不变。例如，如果图6所示的转换序列中 CMDH4[LWI] = 0，则结果 FIFO 的内容类似于图9 (假设该序列未被更高优先级的 TRIGn 信号中断) 所示。

↑
FIFO increases

...
CMD1	Loop CNT = 0	TRIG3	CH5A result
CMD11	Loop CNT = 0	TRIG3	CH7A result
CMD4	Loop CNT = 4	TRIG3	CH2A result
CMD4	Loop CNT = 3	TRIG3	CH2A result
CMD4	Loop CNT = 2	TRIG3	CH2A result
CMD4	Loop CNT = 1	TRIG3	CH2A result
CMD4	Loop CNT = 0	TRIG3	CH2A result
CMD2	Loop CNT = 0	TRIG3	CH0A result
...

图9. 启用循环且CMD4[LWI] = 0的结果FIFO

2.2.6 如果CMDHm[LWI] = 1, 通道编号在循环过程中递增

当 CMDHm[LWI] = 1 时, CMDHm 指定的通道编号会在每次循环迭代结束后递增1 (类似于循环计数 Loop CNT 的递增方式)。例如, 如果图6所示的转换序列中 CMDH4[LWI] = 1, 则结果 FIFO 的内容类似于图10所示 (假设该序列未被更高优先级的 TRIGn 信号中断)。

↑
FIFO increases

...
CMD1	Loop CNT = 0	TRIG3	CH5A result
CMD11	Loop CNT = 0	TRIG3	CH7A result
CMD4	Loop CNT = 4	TRIG3	CH6A result
CMD4	Loop CNT = 3	TRIG3	CH5A result
CMD4	Loop CNT = 2	TRIG3	CH4A result
CMD4	Loop CNT = 1	TRIG3	CH3A result
CMD4	Loop CNT = 0	TRIG3	CH2A result
CMD2	Loop CNT = 0	TRIG3	CH0A result
...

图10. 启用循环且CMD4[LWI] = 1的结果FIFO

2.2.7 在同一 CMD 上启用循环和硬件平均功能

当 CMDHm[AVGS] = M (其中 M != 0) 且 CMDHm[LOOP] = N (其中 N != 0) 时, CMDm 在循环中执行 N+1 次。在每次循环迭代中, CMDm 指定的通道被采样并转换 2^M 次, 结果的平均值存储在 FIFO 中。

在图11 中, 当序列到达 CMD4 时, 由于 CMDH4[AVGS] = 7 且 CMDH4[LOOP] = 4, CMD4 的采样和转换因循环而执行了 5 次。

在每次循环迭代中, CMD4 指定的通道 CH2A 被采样并转换 $2^7 = 128$ 次, 结果的平均值存储在 FIFO 中。在整个循环中, 通道 CH2A 被采样并转换了 $5 \times 128 = 640$ 次。

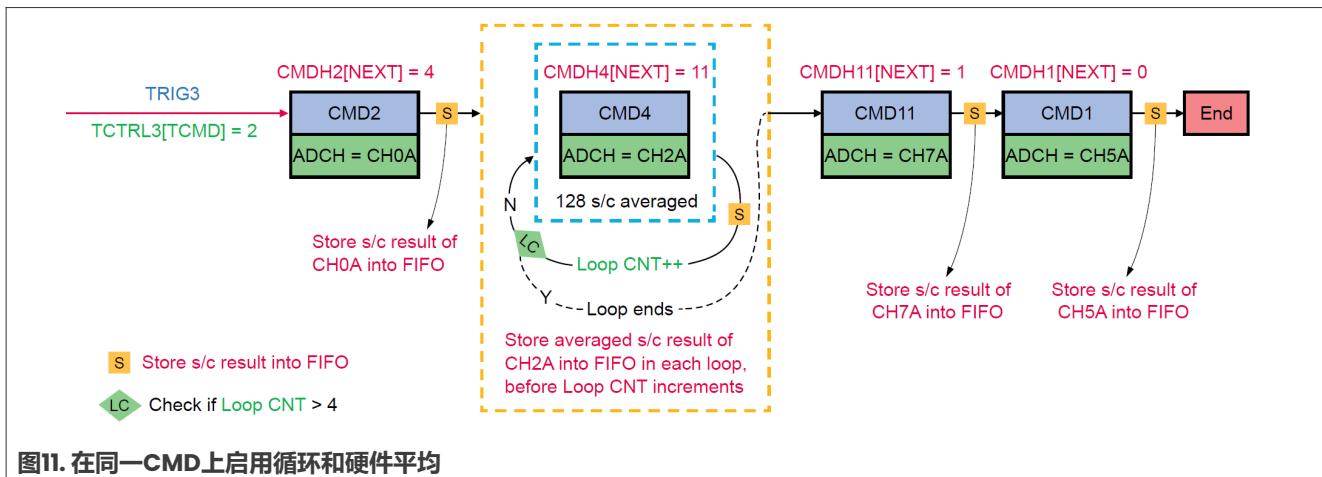


图11. 在同一CMD上启用循环和硬件平均

图12显示了图11中存储在FIFO中的转换结果（假设该序列未被更高优先级的TRIGn信号中断）。CMD4占用五个FIFO条目，每个条目存储128个结果平均后的转换结果。

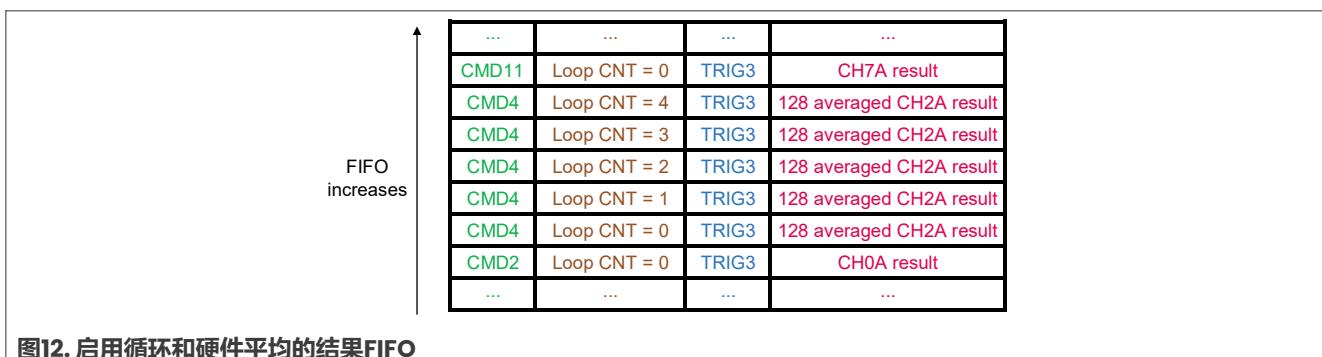


图12. 启用循环和硬件平均的结果FIFO

2.2.8 在序列执行过程中等待触发信号

有时，在转换序列被触发后，需要在某个通道处暂停执行。稍后可以在特定时间点从暂停的通道处恢复执行。这一目标可以通过为相应的CMD启用“等待触发”功能来实现。

要启用此功能，需要配置CMDHm寄存器的WAIT_TRIGGER位字段 (CMDHm[WAIT_TRIGGER] = 1)。图13演示了“等待触发”功能。

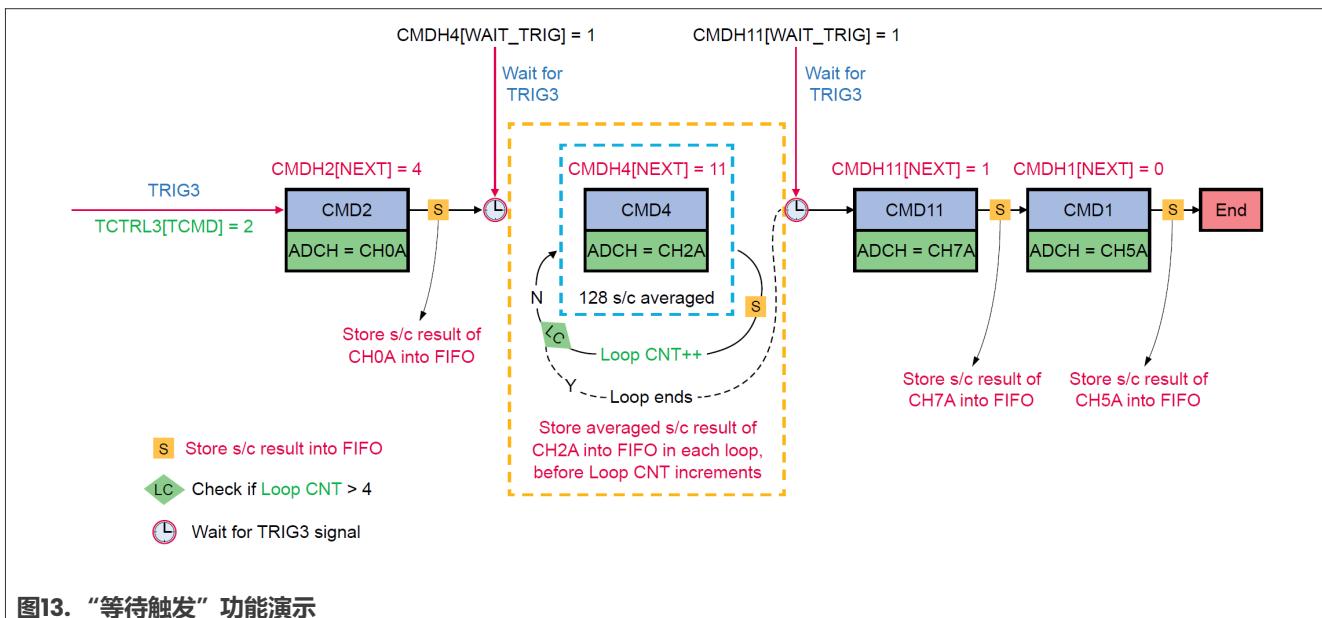


图13. “等待触发”功能演示

在图13中，TRIG3上发生的有效信号触发了一个转换序列，该序列包含CMD2、CMD4、CMD11和CMD1。由于CMDH4[WAIT_TRIGGER] = 1且CMDH11[WAIT_TRIGGER] = 1：

- 在CMD2执行完成后，转换序列会暂停，并等待TRIG3上的有效触发信号，之后从CMD4恢复执行。
- CMD4执行完毕（包括硬件平均和循环操作）后，序列会再次暂停，等待TRIG3上的有效触发信号，之后从CMD11恢复执行。
- CMD11执行完成后，序列会继续执行CMD1。随着CMD1的执行结束，转换序列完成执行。

2.2.9 比较功能

对于CMD1-CMD4，有一种称为比较的功能可用。要启用这个功能，需要将CMDHa寄存器（其中“a”代表1到4）的CMPEN位字段设置为2或3。

当为某个CMD启用了比较功能，并且该通道的转换（包括硬件平均）完成后，硬件会将转换结果与CVa[CVH]和CVa[CVL]位字段中的值进行比较。根据比较结果，再决定是否将转换结果存储到结果FIFO中。

根据CMDHa[CMPEN]的值不同，比较功能分为两种模式：

- “若真存储”比较选项 (CMDHa[CMPEN] = 2)：**在这种模式中，如果比较结果为“真”，转换结果将存储在结果FIFO中：
 - 如果比较结果为“假”，硬件将会丢弃转换结果（不存储到结果FIFO中），然后循环计数器递增。
 - 如果比较结果为“真”，硬件会将转换结果存储到结果FIFO中，然后循环计数器递增。**注：**无论比较结果如何，每次转换完成后，循环计数器都会递增。
- “重复直至为真”比较选项 (CMDHa[CMPEN] = 3)：**在这种模式中，比较操作重复进行，直到比较结果为“真”：
 - 如果比较结果为“假”，硬件将会丢弃转换结果（不存储到结果FIFO中）。循环计数器的值保持不变，然后当前CMD中的通道将再次进行转换。

– 如果比较结果为“真”，硬件将转换结果存储到结果FIFO中，然后循环计数器递增。

在图14中，CMD4启用了“若真存储”比较选项 (CMDH4[CMPPEN] = 2) 的功能，同时还结合了“硬件平均”和循环功能。

在每次循环迭代中，会计算128次转换/采样结果的平均值，并进行一次比较操作。无论比较的结果如何，循环计数器都会加1。只有当比较结果为“真”时，这128次转换结果的平均值才会存储到结果FIFO中。

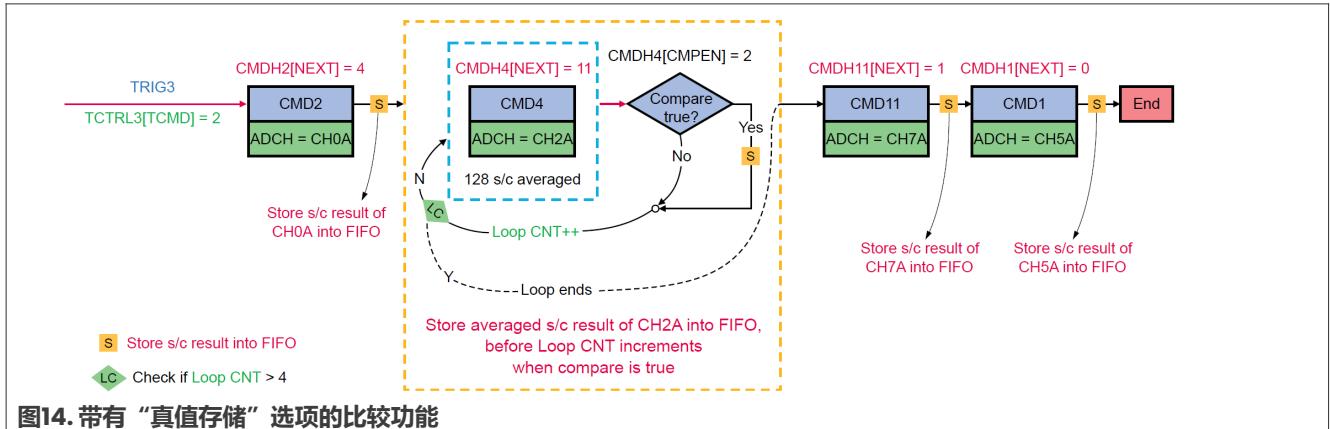


图14. 带有“真值存储”选项的比较功能

在图15中，CMD4启用了“重复直至为真”比较选项 (CMDH4[CMPPEN] = 3) 功能，同时还结合了“硬件平均”和循环功能。

如果128个转换结果的平均值比较结果为“假”，则循环计数器保持不变，通道再次开始进行128次采样/转换，并计算平均值。只有当比较结果为“真”时，平均的结果才会存储到结果FIFO中，同时循环计数器递增。

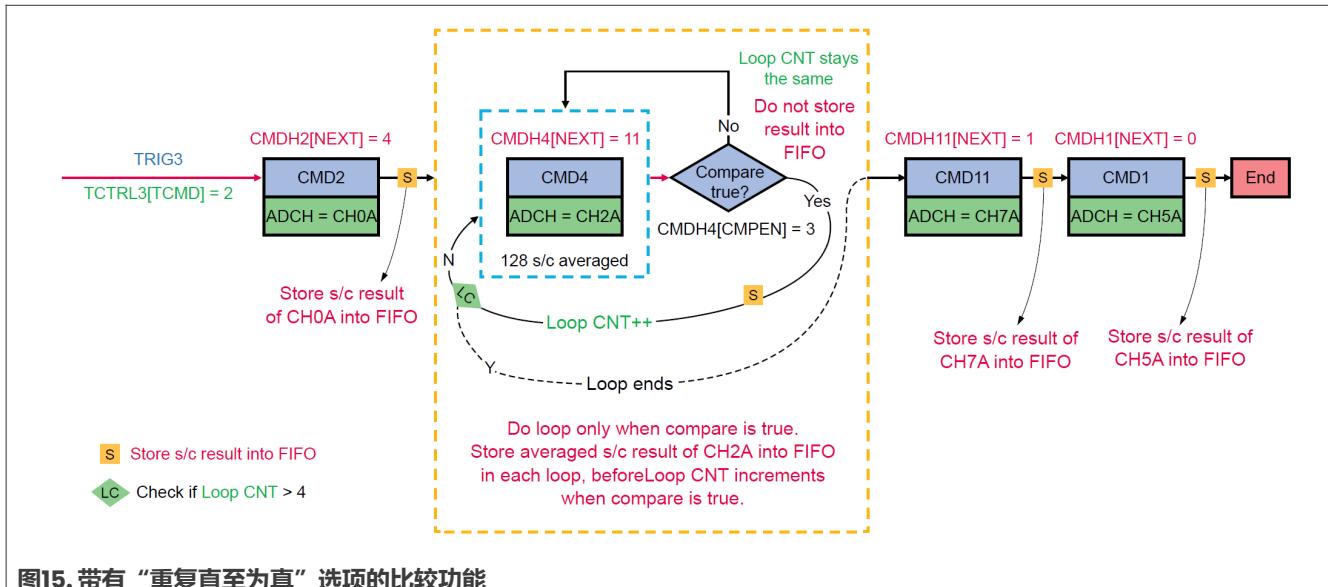


图15. 带有“重复直至为真”选项的比较功能

在双单端模式 (CMDLm[CTYPE] = 3) 下，比较仅针对带有后缀“A”的通道的转换结果。

由比较结果来决定是否将当前这对转换结果存储在结果FIFO中：

- 如果带“A”后缀的通道的比较结果为“真”，则A侧和B侧通道的转换结果都会被存储到结果FIFO中。循环行为由CMDHa[CMPEN]决定。
- 如果带“A”后缀的通道比较结果为“假”，则两个转换结果（A侧通道或B侧通道）都不会存储在结果FIFO中。循环行为由CMDHa[CMPEN]决定。

2.2.9.1 比较结果的解释

根据CVa[CVL]和CVa[CVH]中的值，可能出现以下四种情况：

- 最小值 < CVL < CVH < 最大值**：当转换结果小于CVL或大于CVH时，比较结果为“真”。
- 最小值 < CVL < CVH = 最大值**：当转换结果小于CVL时，比较结果为“真”。
- 最小值 = CVL < CVH < 最大值**：当转换结果大于CVH时，比较结果为“真”。
- CVL > CVH**：当转换结果小于CVL且大于CVH时，比较结果为“真”。

2.2.10 转换结果格式

对于每个CMD，可以通过设置CMDLm[MODE]位字段的值，将转换结果配置为以下格式之一：

- 12位/13位格式 (CMDLm[MODE] = 0) :**
 - 对于单端模式的转换类型，转换结果使用12位的格式。
 - 对于差分模式的转换类型，转换结果使用13位的格式，并以二进制补码形式输出。
- 16位格式 (CMDLm[MODE] = 1) :**
 - 对于单端模式的转换类型，转换结果使用16位的格式
 - 对于差分模式的转换类型，转换结果使用16位的格式，并以二进制补码形式输出。

注：在差分模式下，转换结果会以二进制补码格式存储到结果FIFO中。

[图16](#)显示了当CMDLm[MODE] = 0时，FIFO中的转换结果示例。

Bit 31	Bit 15	Bit 0	
CMD	Loop CNT	TRIG	S/C result
CMDLm[MODE] = 0, CMDLm[CTYPE] = 2	S D D D D D D D D D D D D 0 0 0		
CMDLm[MODE] = 0, CMDLm[CTYPE] != 2, CFG2[JLEFT] = 0	0 D D D D D D D D D D D D 0 0 0		
CMDLm[MODE] = 0, CMDLm[CTYPE] != 2, CFG2[JLEFT] = 1	D D D D D D D D D D D D 0 0 0		

图16. 具有12位/13位转换结果的结果FIFO

在[图16](#)中，‘D’表示数据位，‘S’表示符号位。

[图17](#)显示了当CMDLm[MODE] = 1时，FIFO中的转换结果示例。

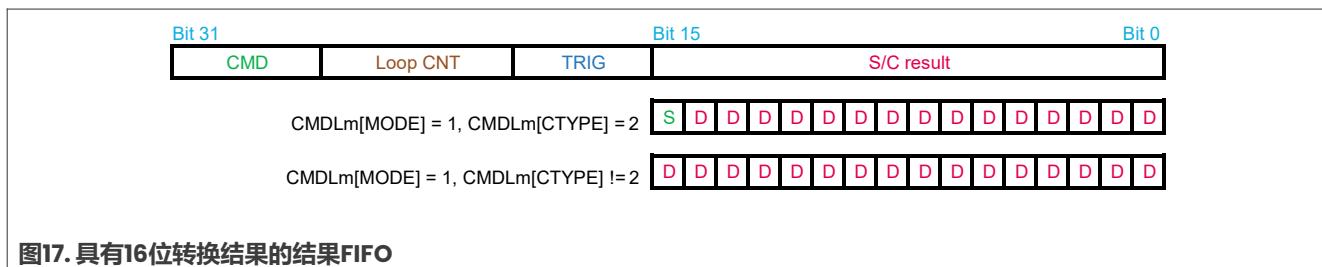


图17. 具有16位转换结果的结果FIFO

在图17中，‘D’ 表示数据位，‘S’ 表示符号位。

2.2.11 通道缩放

对于ADC模块，参考电压通常为1.8V。如果某个通道的输入信号电压高于1.8V，就需要通过缩放来减小信号的幅度。为实现这一目标，可以使用“通道缩放”功能。这一功能适用于所有CMD。

“通道缩放”功能可以通过设置CMDLm寄存器的CSCALE位字段 (CMDLm[CSCALE]) 为某个CMD的通道进行配置，如下所示：

- **CMDLm[CSCALE] = 1**: 满量程，通道信号不做任何缩减
- **CMDLm[CSCALE] = 0**: 通道信号按%的比例缩放，通道信号幅值减半
 - 当CMDLm[ALTBEN] = 0时，CMDLm[CSCALE]控制该CMD的A侧和B侧通道。
 - 当CMDLm[ALTBEN] = 1时，CMDLm[CSCALE]控制该CMD的A侧通道，而CMDLm[ALTB_CSCALE]控制该CMD的B侧通道。

注：在i.MX RT1180 MCU 复位后，CMDLm[ALTB_CSCALE] 默认值为1，CMDLm[CSCALE] 默认值也为1。这意味着所选通道的信号不会被缩减。

2.3 通过PAUSE寄存器在序列中的两次转换间添加延迟

当PAUSE[PAUSEEN] = 1时，在一个转换序列中，两次连续的AD转换之间会产生延迟。延迟时间为4×PAUSE[PAUSEDLY]个ADC的时钟周期。

图18展示了在为CMD4配置了“若真存储”比较选项 (CMDH4[CMPPEN] = 2) 的情况下，设置PAUSE寄存器引起的延迟在序列中的位置。

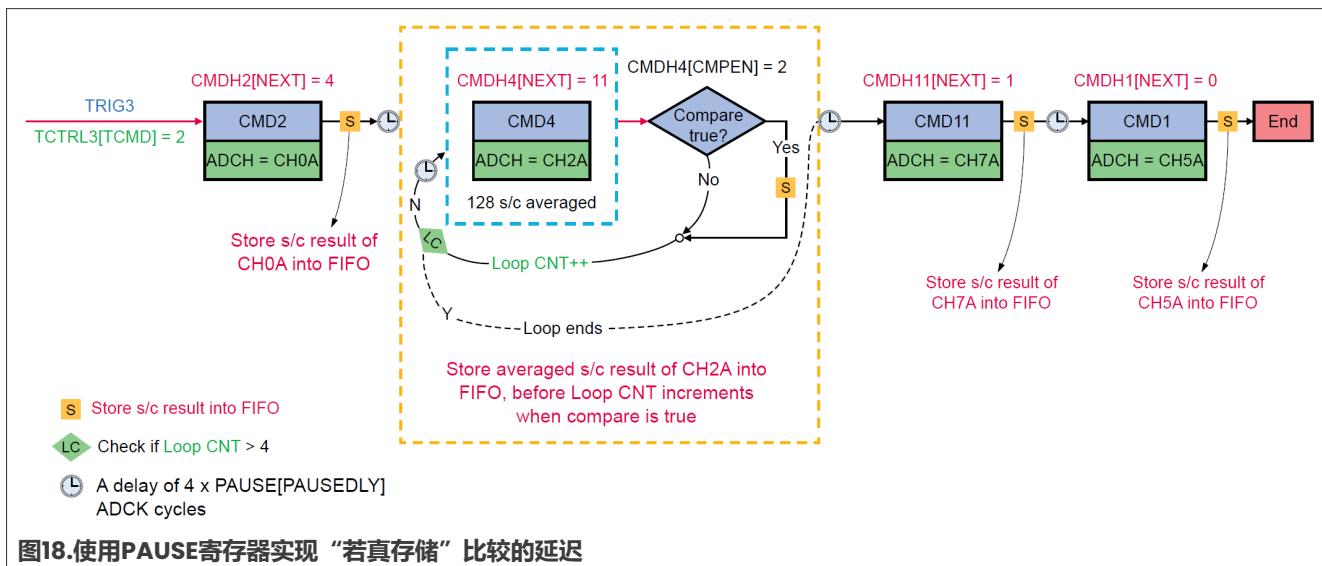


图18. 使用PAUSE寄存器实现“若真存储”比较的延迟

图19展示了在为CMD4配置了“重复直到真”比较选项 (CMDH4[CMPPEN] = 3) 的情况下, 设置PAUSE寄存器引起的延迟在序列中的位置。

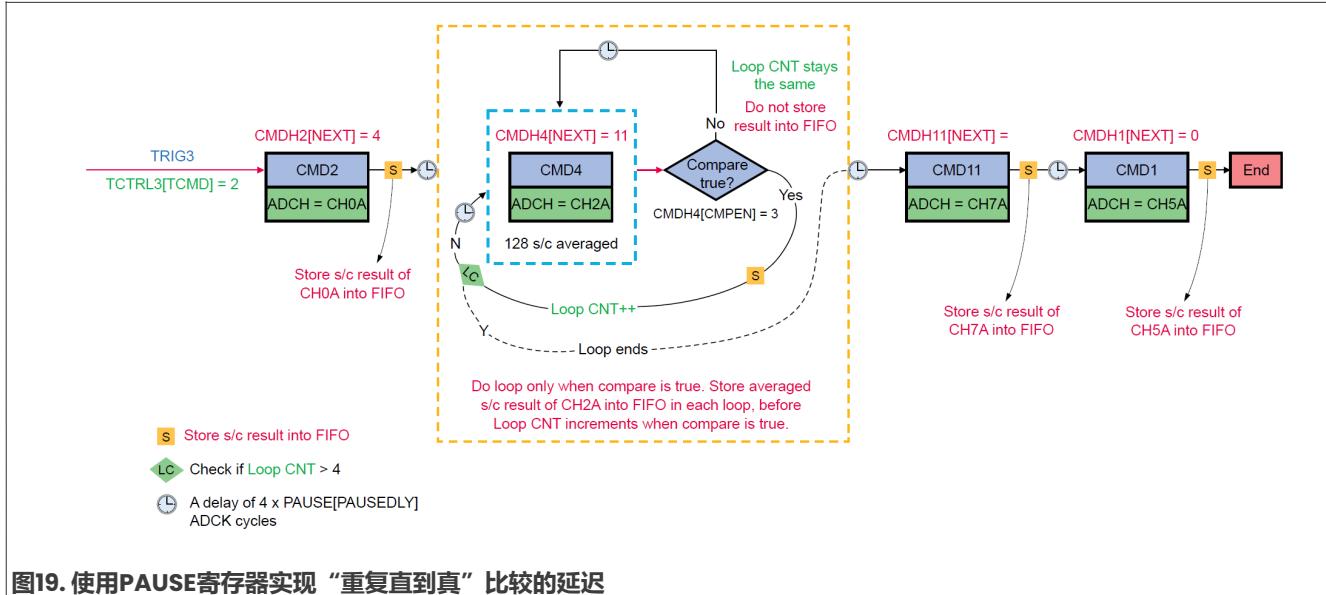


图19. 使用PAUSE寄存器实现“重复直到真”比较的延迟

2.3.1 PAUSE 寄存器与“等待触发”功能不兼容

“等待触发”功能和PAUSE寄存器均可用来在转换序列中增加延迟。但如果在某个CMD的转换序列中同时使用这两者(即CMDHm[WAIT_TRIG] = 1且PAUSE[PAUSEEN] = 1), 则只有“等待触发”功能会生效, 而PAUSE寄存器不会产生任何影响。

图20显示了对图18中的转换序列进行修改后的情况, 为CMD4和CMD1启用了“等待触发”功能:

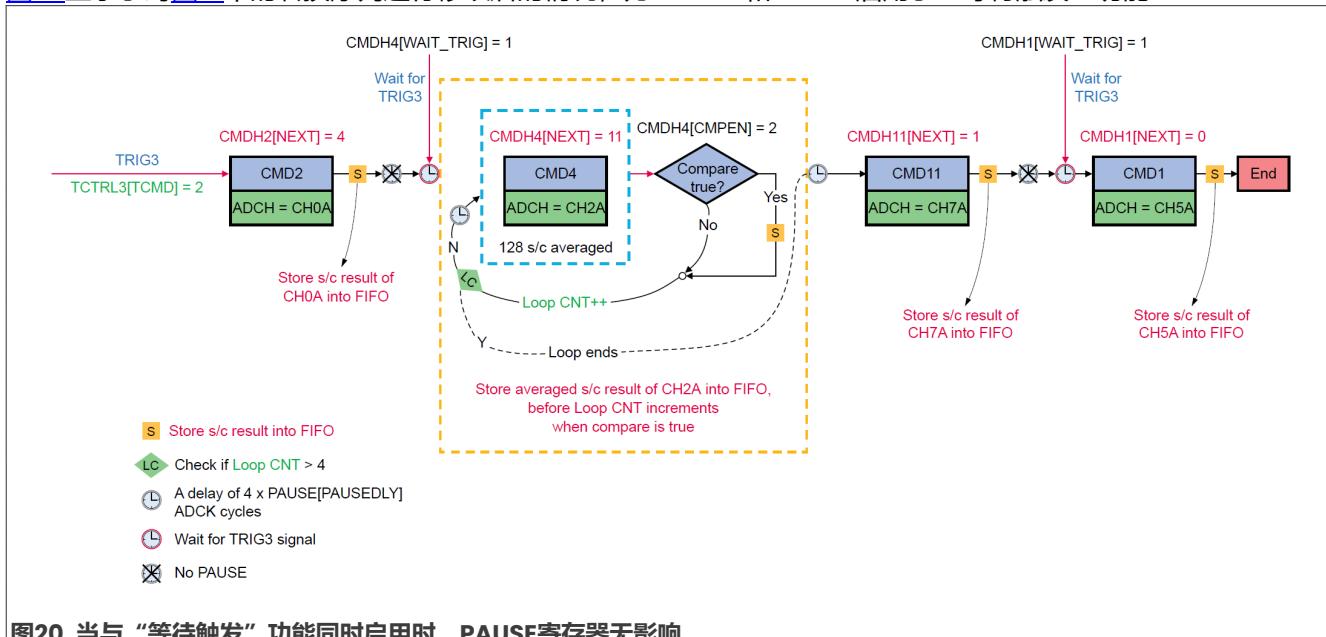


图20. 当与“等待触发”功能同时启用时, PAUSE寄存器无影响

CMD2完成后, 序列会暂停, 等待TRIG3上的另一个有效信号以继续执行CMD4。此时, PAUSE寄存器配置的延迟被忽略。

CMD4 完成后，在执行 CMD11 之前，会出现由 PAUSE 寄存器引起的时间延迟。CMD11 执行完毕后，序列暂停，等待 TRIG3 上的另一个有效信号以继续执行 CMD1。此时，PAUSE 寄存器配置的延迟同样被忽略。

2.4 通过触发控制寄存器TCTRLn配置整个序列

触发控制寄存器 (TCTRLn) 可用于为转换序列设置以下配置信息：

- 序列中第一次 AD 转换所使用的 CMD。
- 触发信号与第一次 AD 转换之间允许的最大延迟。
- 序列的优先级
- 用于存储转换结果的FIFO
- 是否允许硬件信号触发该序列。

TCTRLn寄存器的不同位的字段具有以下用途：

- **TCTRLn[TCMD]**：确定由TRIGn触发的转换序列中的第一个CMD。在图2中，TCTRL3[TCMD] = 2，表示由TRIG3 触发的转换序列中，CMD2 中指定的通道是第一个发生转换的通道。
当TCTRLn[TCMD] = 0时，TRIGn不会触发任何转换序列。
 - **TCTRLn[TDLY]**：表示TRIGn上出现有效触发信号后，到开始执行转换序列中第一个CMD之间的延迟时间。
延迟时间为 2^{TDLY} 个 ADC 时钟 (ADCK) 周期：
 - 对于ADC1模块，ADCK指的是adc1_clk_root，由CLOCK_ROOT44相关寄存器控制。
 - 对于ADC2模块，ADCK指的是adc2_clk_root，由CLOCK_ROOT45相关寄存器控制。
 当TCTRLn[TDLY] = 0时，不会发生此类延迟。
- ADC1和ADC2的根时钟可以在Config Tool中配置，如图21所示。

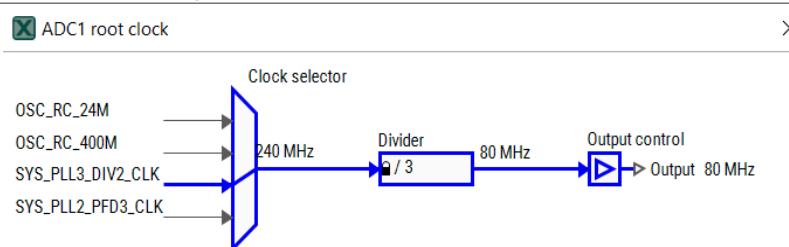


图21. 配置工具中的ADC1根时钟配置

- **TCTRLn[TPRI]**：表示TRIGn信号触发的转换序列的优先级。共有 8 个优先级可选：从 1 级（最高，TCTRLn[TPRI] = 0）到 8 级（最低，TCTRLn[TPRI] = 7）。
如果两个或多个触发信号具有相同的优先级，则编号小的触发事件优先。例如，当TCTRL0[TPRI] = TCTRL1[TPRI]时，TRIG0优先于TRIG1。
- **TCTRLn[FIFO_SEL_A]和TCTRLn[FIFO_SEL_B]**：这些位字段用来帮助确定与TRIGn信号相关的序列使用哪个FIFO。更多详情，请参见第2.2.4节。
- **TCTRLn[HTEN]**：控制是否允许通过硬件信号触发转换序列：
 - 如果TCTRLn[HTEN] = 1，则由TRIGn上的上升沿信号触发相关的转换序列。
 - 如果TCTRLn[HTEN] = 0，则TRIGn上的信号不会触发任何转换序列。但是，无论TCTRLn[HTEN]位字段值是多少，都可以通过配置SWTRIG[SWTn]位字段来手动触发与TRIGn相关的序列。

2.4.1 转换序列优先级

每个 ADC 模块有 8 个 TRIG 输入，可以配置 1 到 8 级优先级。当多个关联了不同 TRIGn 的转换序列同时被触发时，与最高优先级 TRIGn 关联的转换序列将优先于其他序列执行。

2.4.1.1 如果 $\text{CFG}[\text{HPT_EXDI}] = 1$ ，那么触发异常功能将被禁用

ADC 模块支持一种名为“触发异常”的功能，允许高优先级的触发信号中断正在进行的低优先级转换序列。这一功能类似于 CPU 内核的嵌套中断。

“触发异常”功能可以使用 $\text{CFG}[\text{HPT_EXDI}]$ 位字段进行配置：

- 如果 $\text{CFG}[\text{HPT_EXDI}] = 0$ （默认设置），与高优先级 TRIGn 信号关联的转换序列可以中断低优先级的转换序列。
- 如果 $\text{CFG}[\text{HPT_EXDI}] = 1$ ，关联高优先级 TRIGn 信号的转换序列无法中断任何低优先级的转换序列。

2.4.2 低优先级序列的中断点选项

当触发异常功能启用时（ $\text{CFG}[\text{HPT_EXDI}] = 0$ ），一个高优先级的有效 TRIGn 信号可以中断正在进行的低优先级转换序列，使关联高优先级的转换序列开始执行。低优先级转换序列可能还包含硬件平均或循环操作。

低优先级序列被中断的点（即中断点）由 $\text{CFG}[\text{TPRICTRL}]$ 位字段的值决定：

- 如果 $\text{CFG}[\text{TPRICTRL}] = 0$ ：**如果出现高优先级 TRIGn 信号，正在进行的低优先级序列会立即被中止（不等待当前转换完成）。与高优先级 TRIGn 信号相关的序列将在高优先级 TRIGn 信号出现后的 10–11 个 ADC 时钟周期后开始执行。
关于 $\text{CFG}[\text{TPRICTRL}] = 0$ 的使用详情，请参见[第2.4.3节](#)。
- 如果 $\text{CFG}[\text{TPRICTRL}] = 1$ ：**如果出现高优先级的 TRIGn 信号，正在进行的低优先级序列会在当前执行的 CMD（包括硬件平均，但不包括循环）完成后被中止。随后，与高优先级 TRIGn 信号相关的序列开始执行。
关于 $\text{CFG}[\text{TPRICTRL}] = 1$ 的使用详情，请参见[第2.4.4节](#)。
- 如果 $\text{CFG}[\text{TPRICTRL}] = 2$ ：**如果出现高优先级 TRIGn 信号，正在进行的低优先级序列会在当前执行的 CMD（包括硬件平均和循环）完成后被中止。随后，与高优先级 TRIGn 信号相关的序列开始执行。
关于 $\text{CFG}[\text{TPRICTRL}] = 2$ 的使用详情，请参见[第2.4.5节](#)。

2.4.2.1 被中断的低优先级序列的恢复选项

被高优先级 TRIGn 信号中断的低优先级序列是否恢复执行，取决于 $\text{CFG}[\text{TRES}]$ 位字段的值：

- 如果 $\text{CFG}[\text{TRES}] = 0$ ：**被中断的低优先级序列不会恢复执行，也不会重新启动。如果触发异常中断被启用（ $\text{IE}[\text{TEXC_IE}] = 1$ ）：
 - STAT[TEXC_INT]标志可用于确定是否有某个触发序列被中断。
 - TSTAT[TEXC_NUM]标志可用于确定哪个触发序列被中断了。
- 如果 $\text{CFG}[\text{TRES}] = 1$ ：**被中断的低优先级序列会在高优先级序列执行完成后恢复执行。

恢复点由CFG[TCMDRES]位字段的值确定：

- **如果CFG[TCMDRES] = 0：**低优先级序列从第一个CMD开始恢复执行。也就是说，低优先级序列会重新启动。
- **如果CFG[TCMDRES] = 1：**低优先级序列从被中断时正在执行的CMD开始恢复执行。如果被中断的CMD包含循环操作，该循环重新开始。

2.4.3 异常触发，断点选项CFG[TPRICTRL] = 0且恢复选项CFG[TRES] = 0

在图22中，TRIG6的优先级高于TRIG5。当TRIG6上出现一个有效信号时，由于CFG[TPRICTRL] = 0，正在进行的TRIG5转换序列会立即中止。随后，TRIG6的转换序列开始执行。

当TRIG6的转换序列执行完成后，由于CFG[TRES] = 0，TRIG5的转换序列不会恢复执行。

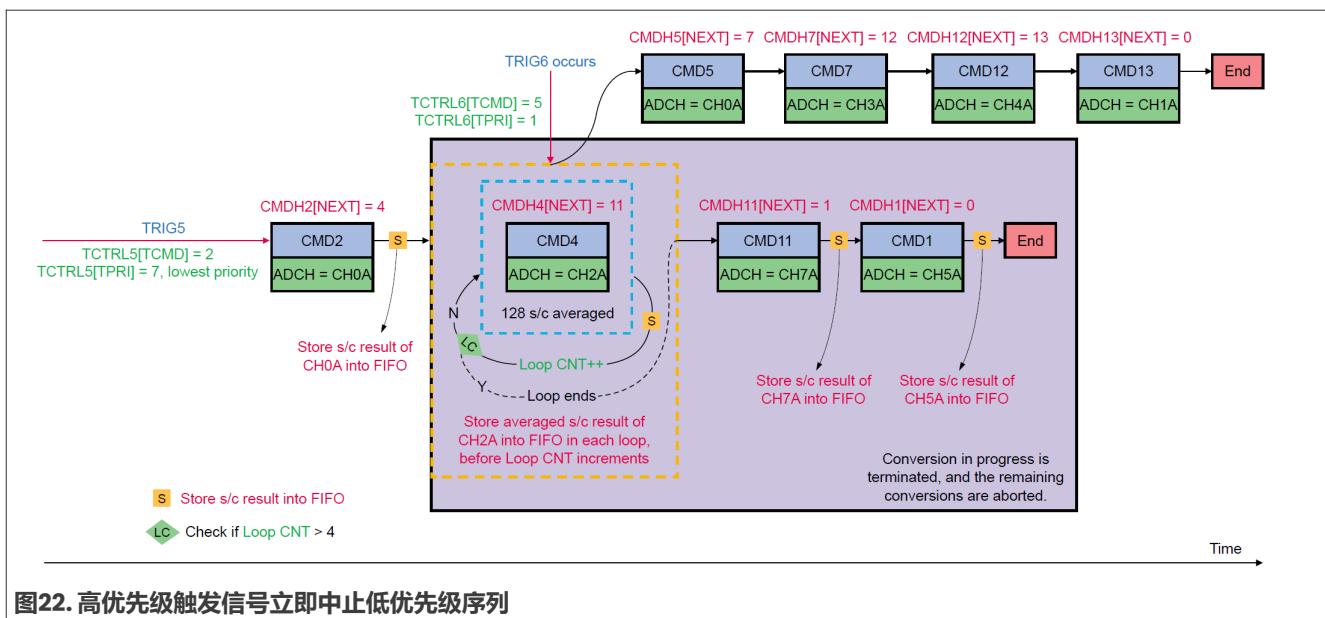


图22. 高优先级触发信号立即中止低优先级序列

2.4.4 异常触发，断点选项CFG[TPRICTRL] = 1、恢复选项CFG[TRES] = 1且CFG[TCMDRES] = 1

在图23中，TRIG6的优先级高于TRIG5。当TRIG5的转换序列执行到CMD4时，TRIG6上出现一个有效信号。由于CFG[TPRICTRL] = 1，TRIG6的转换序列会等待TRIG5的转换序列中CMD4的硬件平均操作完成后才开始执行，但不会等待CMD4中的后续循环操作完成。

TRIG6的转换序列执行完毕后（如没有其他更高优先级的序列等待执行），TRIG5的转换序列会自动从被中断时正在执行的CMD4处继续执行，这是因为CFG[TRES] = 1且CFG[TCMDRES] = 1。在这种情况下，CMD4的循环操作会重新开始。

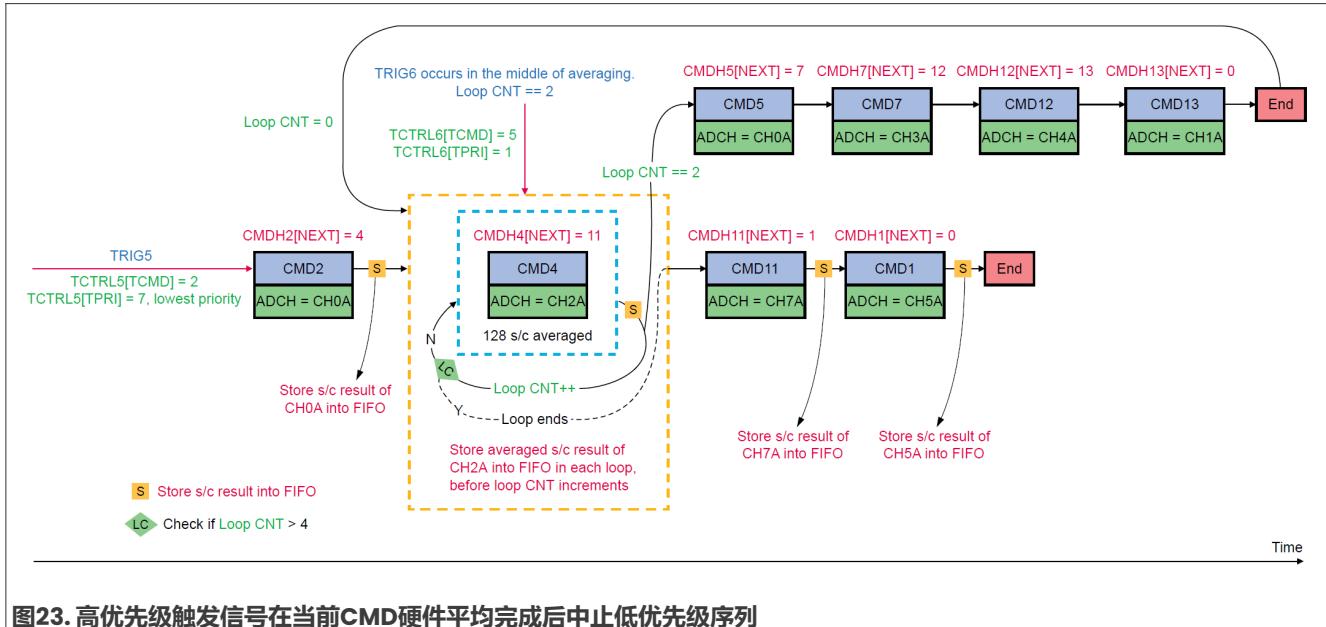


图23. 高优先级触发信号在当前CMD硬件平均完成后中止低优先级序列

图24展示了 TRIG5 的执行序列被中断及恢复时, 结果 FIFO 中的条目情况。



图24. 当CFG[TCMDRES] = 1时, 结果FIFO显示转换序列的中断与恢复

2.4.5 异常触发, 断点选项CFG[TPRICTRL] = 2、恢复选项CFG[TRES] = 1且CFG[TCMDRES] = 1

在图25中, TRIG6的优先级高于TRIG5。当 TRIG5 的序列执行到 CMD4 时, TRIG6 上出现一个有效信号。由于 CFG[TPRICTRL] = 2, TRIG6 的序列会等到 TRIG5的 序列中 CMD4 (包括硬件平均和循环操作) 完成后才开始执行。

TRIG6 的序列执行完毕后 (如没有其他更高优先级的序列等待执行), TRIG5 的序列会自动从 CMD11 开始继续执行, 这是因为 CFG[TRES] = 1且 CFG[TCMDRES] = 1。在这种情况下, TRIG5 的序列从 CMD11处 恢复执行, 因为 CMD4 已经执行完毕。

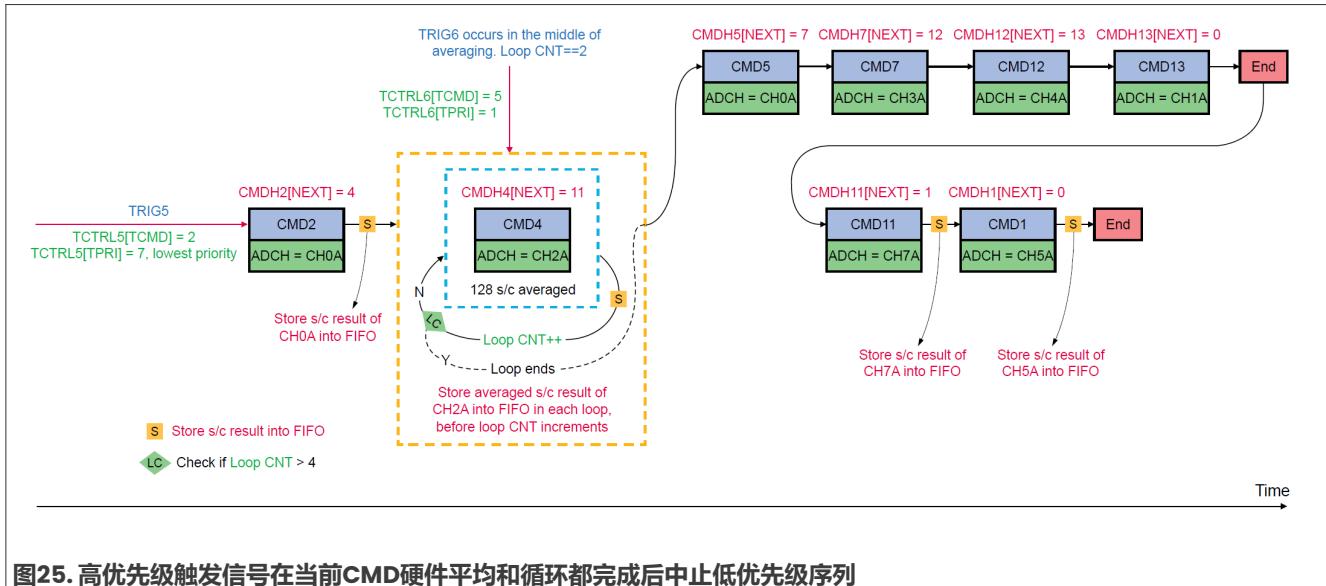


图25. 高优先级触发信号在当前CMD硬件平均和循环都完成后中止低优先级序列

2.4.6 触发延迟和上电延迟

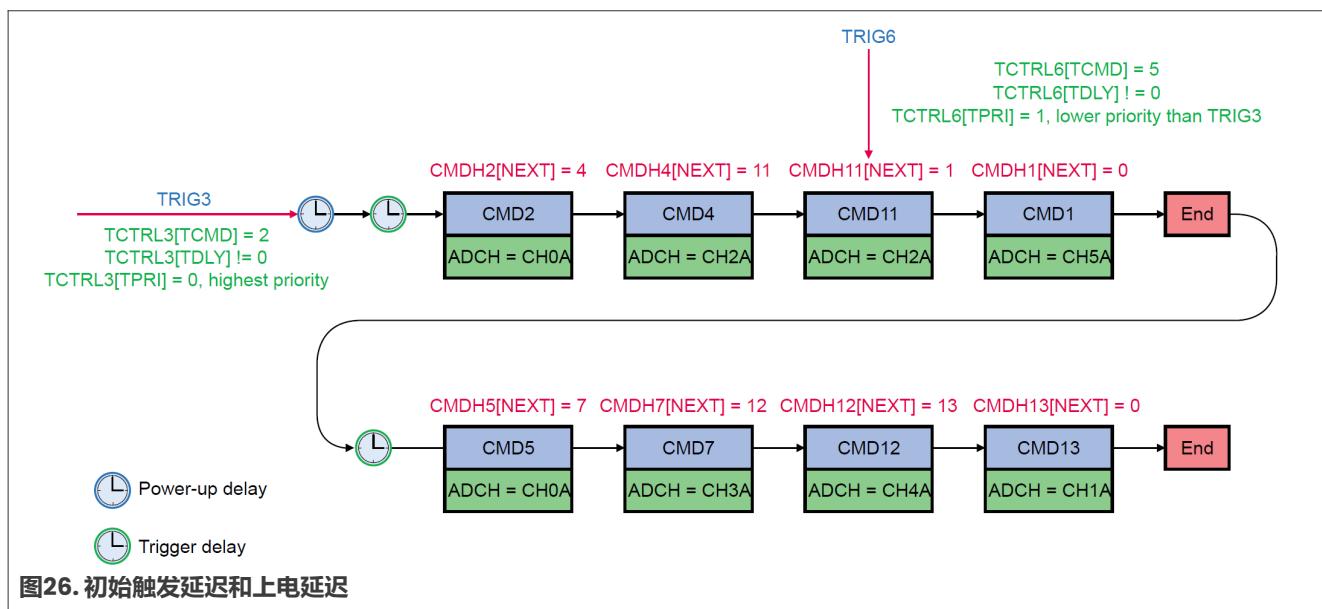
如第2.4节所述, TCTRLn[TDLY] 位字段可用于配置TRIGn上出现有效触发信号后, 到转换序列中第一个 CMD 开始执行之间的延迟时间。

除了转换序列的初始延迟 (即触发延迟) 外, ADC 模块在开始执行转换序列前可能还需要一个上电延迟。上电延迟的时间为 $4 \times \text{CFG}[\text{PUDLY}]$ 个 ADC 时钟周期。ADC 的上电延迟可以使用 CFG[PWREN] 位字段进行配置:

- 如果CFG[PWREN] = 0, 则 ADC 模块仅在执行转换时启用。这意味着每次 ADC 模块开始执行一个序列时, 都需要经历上电延迟。
- 如果CFG[PWREN] = 1, 则 ADC 模块在通过设置 CFG[PWREN] 位字段值启用后, 将始终保持启用状态。

在初始启用时, ADC 模块需要从 CFG[PWREN] 位字段设为 1 的时刻起经历一个上电延迟。此后, ADC 模块将始终处于启用状态。此时, 如相关 TRIGn 上出现有效信号 (假设 TCTRLn[TDLY] = 0), ADC 模块会立即开始执行转换序列。

图26演示了触发延迟和上电延迟。



在图26中，当 CFG[PWREN] 位字段刚被设为 1 时，TRIG3 上立即出现了一个有效信号。因此，在 TRIG3 关联的序列开始执行前，会先后经历上电延迟（由于 ADC 模块上电）和触发延迟（因为 TCTRL3[TDLY] != 0）。

在 TRIG3 的序列执行期间，TRIG6 上出现了一个优先级较低的有效信号。由于 TRIG6 信号的优先级低于 TRIG3 信号的优先级，它需要等待 TRIG3 序列执行完成。此外，因为 TCTRL6[TDLY] != 0，TRIG6 的序列开始时也会经历一个触发延迟。

3 ADC参考电压

在i.MX RT1180 MCU中，ADC的参考电压根据CFG[REFSEL]位字段的值进行选择：

- 如果CFG[REFSEL] = 0，ADC 使用其 VREFH 引脚的电压作为参考电压。VREF 模块的 VREF_OUT 引脚与 ADC 模块的 VREFH 引脚在 SoC 封装中绑定在同一引脚 (ADC_VREFH) 上。
注：当使用 VREFH 作为参考电压时，需确保 ADC 时钟频率低于 20 MHz，否则无法保证 i.MX RT1180 数据手册中描述的 ADC 性能。
- 如果CFG[REFSEL] = 1或2，ADC 使用VDDA_ADC_IP8引脚的电压作为参考电压。

要通过VREFH引脚提供ADC参考电压，可以使用以下选项之一：

选项1：ADC参考电压来自内部源

在此选项中，ADC 参考电压由内部源（VREF 模块）提供，如图27所示。要使用此选项，必须配置并启用VREF 模块。

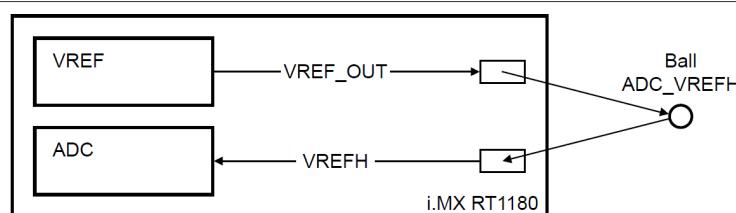


图27. 使用内部VREF作为ADC参考

选项2：ADC参考电压来自外部源

在此选项中，ADC 参考电压由外部源（如 LDO 稳压器）提供，如图28 所示。要使用此选项，需禁用VREF模块。

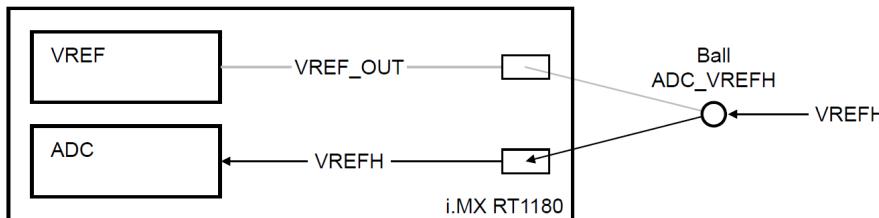


图28.使用外部VREFH作为ADC参考电压并禁用内部VREF

4 CMD通道的采样与转换时间

不同CMD通道的采样和AD转换所需时间可能有所不同。采样时间取决于CMDHm[STS]位字段的值，而转换时间则取决于转换结果使用的格式（12位/13位格式或16位格式）。

表1展示了在不同采样和转换设置下，一个CMD通道的采样加转换消耗的总时间。

表1. 采样+转换时间

CMDHm[STS] 值	转换结果格式(CMDLm[MODE]值)	采样+转换总时间(以 ADC 时钟周期为单位)
CMDHm[STS] = 0	12位/13位格式(CMDLm[MODE] = 0)	3.5 + 15.5
	16位格式(CMDLm[MODE] = 1)	3.5 + 20.5
CMDHm[STS] > 0	12位/13位格式(CMDLm[MODE] = 0)	3.5 + 2 ^{STS} + 15.5
	16位格式(CMDLm[MODE] = 1)	3.5 + 2 ^{STS} + 20.5

5 ADC中断

在i.MX RT1180 MCU中，ADC模块在以下情况下可以触发中断：

- 当结果FIFO_n (n = 0或1) 包含数据，且有效数据字数超过 FCTRL[FWMARK] 位字段设定的水印级别时：**此中断通过将 IE[FWMIE_n] 位字段设为1来启用。当有效数据字数超过水印（即 FCTRL[FWMARK] 位字段的值）时，1位标志 (STAT[RDY_n]位字段) 被设置为1；否则，该标志会被清除。STAT[RDY_n] 位字段无法通过软件设置或清除。
注：如果结果 FIFO 中的数据字数恰好达到水印级别，STAT[RDY_n] 标志不会被设置。
- 当结果FIFO_n (n = 0或1) 溢出时：**此中断通过将IE[FOFIE_n]位字段设为1来启用。当FIFO溢出时，1位标志 (STAT[FOFn]位字段) 被设置为1，且FIFO不再接收新数据，但现有数据保持不变。STAT[FOFn]位字段可通过向其写入1来手动清除。
- 当与TRIG_n (n = 0–7) 相关的转换序列执行完成时：**要启用/禁用与TRIG0至TRIG相关的序列完成中断，可使用8位大小的IE[TCOMP_IE]位字段。该位字段的每一位启用/禁用相应的序列完成中断。当一个转换序列完成执行时，1位标志 (STAT[TCOMP_INT]位字段) 被设置为1。要确定是哪个TRIG序列完成了执行，可以通过读取8位大小的TSTAT[TCOMP_FLAG]位字段实现。要清除STAT[TCOMP_INT]和TSTAT[TCOMP_FLAG]位字段中已使用的位，只需在对应位写入1即可。

- 当高优先级的触发信号中断正在执行的转换序列，且CFG[TRES]位字段设为0（即被中断的低优先级序列不会恢复执行或重启）时：此中断通过将IE[TEXC_IE]位字段设为1来启用。中断触发时，1位标志（STAT[TEXC_INT]位字段）被设为1。STAT[TEXC_INT]位字段可通过写入1手动清除。要确定哪个序列被中断，通过读取8位大小的TSTAT[TEXC_NUM]位字段来实现。

对于每个ADC模块，这四种中断在内核处共享一个中断向量。

6 由ADC触发的DMA中断

当ADC模块的结果FIFO（FIFO0或FIFO1）中的有效数据的字数超过水印级别时，结果FIFO可以触发DMA中断，如表2所述。

表2. DMA中断

ADC模块	结果FIFO	触发的DMA中断
ADC1	FIFO0	通过源57触发eDMA4
	FIFO1	通过源220触发eDMA4
ADC2	FIFO0	通过源158触发eDMA4
	FIFO1	通过源221触发eDMA4

7 ADC偏移和增益误差校准

为了在初始化时达到规定的精度，每个ADC模块必须对偏移和增益进行校准。偏移误差有两个值，分别存储在OFSTRIM16和OFSTRIM12寄存器中，它们的位宽分别为16位和12位。A侧和B侧的增益误差则存储在GCR0[GCALR]和GCR1[GCALR]位字段中。

有关校准过程的更多详细信息，请参阅《i.MX RT1180 参考手册》中“模数转换器（ADC）”章节的“初始化”部分。

8 ADC温度传感器

每个ADC模块都内置了一个温度传感器，连接在CH26A和CH26B通道上。要获取正确的温度数据，请使用以下配置：

- 对于用于温度感应的CMD：
 - 通道号必须为26 (CMDLm[ADCH] = 26)。
 - CMD必须处于差分模式 (CMDLm[CTYPE] = 2)。
- 对于CMD，转换结果可以是12位/13位格式或16位格式。
- 使用最大硬件平均时间 (CMDHm[AVGS] = 10)。
- 使用最长采样时间 (CMDHm[STS] = 7)。
- 循环两次 (CMDHm[LOOP] = 1)。
- 在循环期间，CMD的通道号保持不变，例如设置CMDHm[LWI] = 0。
- 禁用比较功能 (CMDHm[CMPPEN] = 0)。

结果FIFO有两个与此CMD对应的结果。这些结果可用于计算环境温度，如下所示：

$$Temp = A * \left[\frac{\alpha * (V_{be8} - V_{be1})}{V_{be8} + (\alpha * (V_{be8} - V_{be1}))} \right] - B$$

其中：

- V_{be1} 是FIFO 中存储的第一个结果。
- V_{be8} 是FIFO 中存储的第二个结果。
- A是压摆率因子，值为789.2。
- B是偏移因子，值为319.2。
- α 是带隙系数，值为11.2。

计算得出的温度单位为摄氏度 (°C)。为了实现最高的计算精度，需要将CMDLm[CScale]位字段设为1。

9 注意事项

- 当选择ADC_VREFH作为ADC参考电压时，ADC时钟频率必须低于20MHz，才能达到数据手册中提到的性能。
- 当选择VDDA_ADC_1P8作为ADC参考电压时，最大ADC时钟频率为88MHz。
- CTRL[RST]位字段可用于重置ADC的内部逻辑和寄存器。但是，它无法用于重置CTRL寄存器本身。
- 如果CTRL[ADCEN] = 1，则 CFG 寄存器的位字段值无法更改。在写入CFG寄存器之前，请清除 CTRL[ADCEN]位字段。
- STAT[TCOMP_INT] 标志仅在 IE[TCOMP_IE] = 1 时，且某个序列执行完成后，才会被置位。
TSTAT[TCOMP_FLAG] 的各位仅在 IE[TCOMP_IE] = 1 时，且对应序列执行完成后，才会被置位。
- STAT[TEXC_INT] 标志仅在 IE[TEXC_IE] = 1 且 CFG[TRES] = 0，且发生高优先级信号触发中断时，才会被置位。
- 使用ADC前必须进行校准；否则无法保证其性能。
- 当对应CMD通道的信号电压高于1.8 V时，必须清除CScale位。

10 参考资料

以下是一些可供参考的文档，可提供有关i.MX RT1180 MCU的更多信息：

- [《i.MX RT1180 参考手册》 \(IMXRT1180RM\)](#)
- [《i.MX RT1180跨界处理器工业产品数据手册》 \(IMXRT1180IEC\)](#)
- [《i.MX RT1180跨界处理器扩展工业产品数据手册》 \(IMXRT1180XEC\)](#)

11 缩略语

[表3](#)列出了本文档中使用的缩略语。

表3. 缩略语

缩略语	描述
AD	模数转换
ADC	模数转换器
ADCK	ADC时钟
CMD	命令缓冲区

表3. 缩略语(续)

缩略语	描述
LDO	低压差
SAR	逐次逼近寄存器
S/C	采样+转换
SoC	系统级芯片
XBAR	外设间跨界开关

12 修订历史

[表4](#)总结了本文档的修订情况。

表4. 修订历史

文档ID	发布日期	描述
AN14485 v.1.0	2025年2月7日	首次公开发布

Legal information

Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at <https://www.nxp.com.cn/profile/terms>, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Suitability for use in non-automotive qualified products — Unless this document expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

HTML publications — An HTML version, if available, of this document is provided as a courtesy. Definitive information is contained in the applicable document in PDF format. If there is a discrepancy between the HTML document and the PDF document, the PDF document has priority.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately. Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

NXP B.V. — NXP B.V. is not an operating company and it does not distribute or sell products.

Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

目录

1	介绍	2	6	由ADC触发的DMA中断	22
2	转换序列的实现	2	7	ADC偏移和增益误差校准	22
2.1	AD转换序列类型	3	8	ADC温度传感器	22
2.1.1	顺序转换	3	9	注意事项	23
2.1.2	无终止循环转换	4	10	参考资料	23
2.1.3	连续转换	4	11	缩略语	23
2.2	通过CMD来配置一个序列中的每次转换	4	12	修订历史	24
2.2.1	硬件平均	4		法律声明	25
2.2.2	循环功能	5			
2.2.3	转换类型	6			
2.2.4	结果 FIFO	7			
2.2.5	如果CMDHm[LWI] = 0, 通道编号在循环过程中保持不变	7			
2.2.6	如果CMDHm[LWI] = 1, 通道编号在循环过程中递增	8			
2.2.7	在同一 CMD 上启用循环和硬件平均功能	8			
2.2.8	在序列执行过程中等待触发信号	9			
2.2.9	比较功能	10			
2.2.9.1	比较结果的解释	12			
2.2.10	转换结果格式	12			
2.2.11	通道缩放	13			
2.3	通过PAUSE寄存器在序列中的两次转换间添加延迟	13			
2.3.1	PAUSE 寄存器与“等待触发”功能不兼容	14			
2.4	通过触发控制寄存器TCTRLn配置整个序列	15			
2.4.1	转换序列优先级	16			
2.4.1.1	如果CFG[HPT_EXDI] = 1, 那么触发异常功能将被禁用	16			
2.4.2	低优先级序列的中断点选项	16			
2.4.2.1	被中断的低优先级序列的恢复选项	16			
2.4.3	异常触发, 断点选项CFG[TPRICTRL] = 0且恢复选项CFG[TRES] = 0	17			
2.4.4	异常触发, 断点选项CFG[TPRICTRL] = 1、恢复选项CFG[TRES] = 1且CFG[TCMDRES] = 1	17			
2.4.5	异常触发, 断点选项CFG[TPRICTRL] = 2、恢复选项CFG[TRES] = 1且CFG[TCMDRES] = 1	18			
2.4.6	触发延迟和上电延迟	19			
3	ADC参考电压	20			
4	CMD通道的采样与转换时间	21			
5	ADC中断	21			

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.