

Application Note

隔离式调制器与 MCU 之间的数字接口的时钟边沿延迟补偿



Gizem Yucece, Alex Smith, Martin Staebler

摘要

具有高速数字接口的隔离式 Δ - Σ 调制器 (例如 [AMC1306M25](#)) 通常用于伺服驱动器和机器人应用中基于分流器且具有低延时和高抗噪性能的精确相电流检测。特别是在较高的时钟频率下, 为了实现可靠运行, 适当的布线、端接以及符合相应 MCU 的建立时间和保持时间很重要。满足 MCU 时序要求的一种常用方法和折衷方法是降低调制器时钟频率, 这也会降低数据输出速率。本应用手册介绍了更合适的时钟边沿补偿方法, 可在高达调制器最大时钟速率的情况下满足建立时间和保持时间要求。这使得系统能够以最大数据速率运行。本应用手册概述了时钟边沿补偿选项, 并显示了使用连接到 C2000™ 和 Sitara™ MCU 的 TI 隔离式调制器 AMC130x 进行的示例测量。此外, 还提供了一个计算工具来验证数字接口时序。

内容

1 引言.....	3
2 数字接口时序规格的设计挑战.....	4
3 具有时钟边沿延迟补偿的设计方法.....	7
3.1 具有软件可配置相位延迟的时钟信号补偿.....	7
3.2 具有硬件可配置相位延迟的时钟信号补偿.....	8
3.3 通过时钟返回进行时钟信号补偿.....	8
3.4 通过 MCU 的时钟反相来实现时钟信号补偿.....	8
4 测试和验证.....	10
4.1 测试设备和软件.....	10
4.2 具有软件可配置相位延迟的时钟信号补偿测试.....	10
4.3 通过 MCU 上的时钟反相进行时钟信号补偿的测试.....	12
4.4 通过计算工具进行数字接口时序验证.....	15
5 结语.....	16
6 参考资料.....	18
7 Revision History.....	18

插图清单

图 1-1. 简化的三相逆变器方框图, 带有从 MCU 到隔离式调制器的数字接口.....	3
图 2-1. AMC1306M25 与 TMS320F28379D 之间数字接口的简化图.....	4
图 2-2. AMC1306x 数字接口时序.....	4
图 2-3. TMS320F28379D SDFM 时序图 - 模式 0.....	5
图 2-4. AMC1305L25 与 TMS320F28379D 的数字接口.....	5
图 2-5. AMC1303M2520 用于连接 TMS320F28379D 的 3.3V CMOS 数字接口.....	6
图 3-1. 具有软件可配置时钟相位延迟的 AMC1306M25 至 AM243x MCU 接口.....	7
图 3-2. 在 SD0_CLK (GPIO1_1) 使用 30ns 相移时钟信号输入时的 AM243x SDFM 时序.....	7
图 3-3. AMC1306M25 与 MCU 连接的数字接口, 通过硬件可配置相位延迟进行补偿.....	8
图 3-4. AMC1306M25 与 MCU 连接的数字接口, 通过时钟返回进行补偿.....	8
图 3-5. TMS320F28379D SDFM/GPIO 方框图.....	9
图 3-6. 在 GPIO123 使用反相时钟时的 TMS320F28379D SDFM 时序.....	9
图 4-1. 使用 AMC1306EVM 和 C2000 TMS320F28379D LaunchPad 在软件中配置相位延迟的时钟信号补偿测试设置.....	11
图 4-2. 使用 AMC1306EVM 和 Sitara AM243x LaunchPad 在软件中配置相位延迟的时钟信号补偿测试设置.....	11
图 4-3. 使用具有软件可配置相位延迟的时钟信号补偿的 AMC1306EVM 和 TMS320F28379D 数字接口时序测量结果.....	12
图 4-4. 使用具有软件可配置相位延迟的时钟信号补偿的 AMC1306EVM 和 AM243x 数字接口时序测量结果.....	12
图 4-5. 数字接口时序验证的测试设置.....	13

图 4-6. CCS 中的时钟和数据输入测试信号 (同相 GPIO123) 和 Sinc3 OSR 64 滤波器输出.....	14
图 4-7. CCS 中的时钟和数据输入测试信号 (同相 GPIO123) 和 Sinc3 OSR 64 滤波器输出.....	14
图 4-8. 基于数据表中的典型规格、在 20MHz 的时钟频率下且使用软件可配置的相位延迟进行时钟边沿补偿的 C2000 与 AMC1305L25 的数字接口时序图.....	16

表格清单

表 4-1. 测试设备列表.....	10
表 4-2. 在 20MHz 时钟频率下使用 AMC1305L25 的 C2000 MCU TMS320F28379D 数字接口时序结果.....	15
表 4-3. 在 17MHz 时钟下使用 AMC1305L25 时的 TMS320F28379D 数字接口时序.....	15
表 4-4. AMC1305L25 在 20MHz 时钟频率下且具有软件可配置相位延迟时的 TMS320F28379D 数字接口时序.....	16
表 5-1. 时钟边沿补偿方法的比较.....	17
表 5-2. 为具有内部或外部时钟的调制器建议的时钟边沿补偿方法.....	17

商标

C2000™, Sitara™, OSR™, and Code Composer Studio™ are trademarks of Texas Instruments.

所有商标均为其各自所有者的财产。

1 引言

因为准确、低延时的隔离式相电流检测对三相逆变器的性能会产生重大影响，所以隔离式 Δ - Σ 调制器通常用于伺服驱动器和机器人应用中基于分流器的相电流检测。 Δ - Σ 调制器通过 LVDS 或 CMOS 接口与 MCU 连接来提供数字位流，从而实现出色的抗噪性、高精度和低延时相电流测量。有关隔离调制器的额外信息，请参阅[比较隔离式放大器和隔离式调制器](#)应用手册。

通常，分流器和隔离式 Δ - Σ 调制器放置在功率级印刷电路板 (PCB) 上，而 MCU 放置在单独的控制电路板 PCB 上，如图 1 所示。为了实现数字信号完整性，对 PCB 和接口连接器使用适当的布线方案至关重要。在[电机驱动器中使用隔离式 \$\Delta\$ - \$\Sigma\$ 调制器提高信号完整性 \(ti.com\)](#)应用报告中讨论了时钟和数据线路布线和端接的最佳实践。

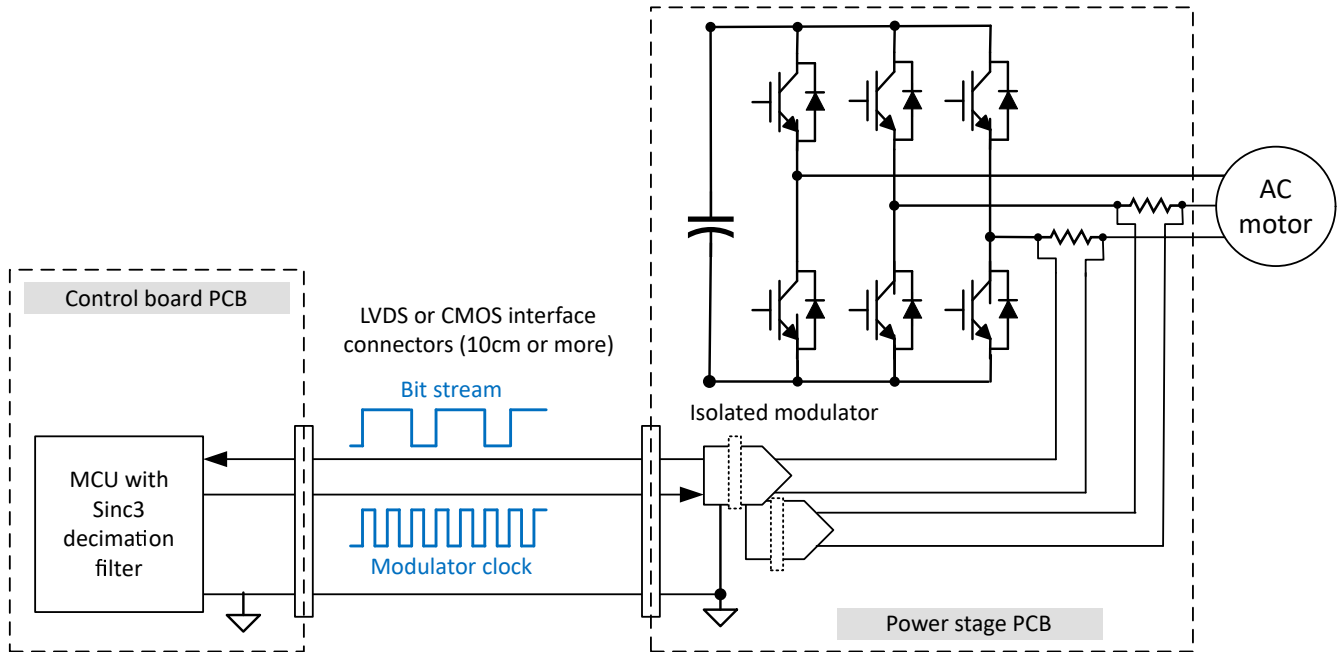


图 1-1. 简化的三相逆变器方框图，带有从 MCU 到隔离式调制器的数字接口

为了满足调制器时钟边沿和数字位流之间的时序要求，尤其是当信号布线相当长，使用了额外的缓冲器和电平转换器时，会出现进一步的设计挑战。然后，调制器时钟和位流信号的额外传播延迟甚至会使设计人员不得不将调制器时钟从最大 21MHz (AMC1306) 降低到 15MHz，以满足 MCU 的时钟边沿和位流数据之间的时序要求。因此，总体相电流测量延时与所选调制器时钟成反比增加。例如，通常使用的过采样率为 64 的 Sinc3 抽取滤波器在 20MHz 调制器时钟下的测量延时（传播延迟）为 4.8us，而当仅可使用 15MHz 调制器时钟时，延迟会增加到 6.4us。

本文档的以下各节概述了为克服这个设计难题而采用的数字时序补偿方法，并展示了使用隔离式调制器进行设计不仅可提供最高精度的测量，而且可以提供最简单的测量方法。

2 数字接口时序规格的设计挑战

隔离式 Δ - Σ 调制器为外部和内部生成的时钟信号提供接口选项，包括 CMOS 接口或 LVDS 接口。对于具有外部提供时钟源的器件（例如具有 CMOS 接口的 [AMC1306M25](#) 或具有 LVDS 接口的 [AMC1305L25](#)），时钟信号从 MCU 路由到 Δ - Σ 调制器的时钟输入；而对于具有内部提供时钟源的器件（例如 [AMC1303M2520](#)），输出位流与内部生成的时钟同步。还有具有曼彻斯特编码输出位流、支持单线数据和时钟传输的隔离式 Δ - Σ 调制器器件，例如 [AMC1306E25](#)。对于所有隔离式 Δ - Σ 调制器，调制器的数据输出提供由数字 1 和 0 组成的位流，该位流同步移出到时钟边沿。

图 2-1 显示了隔离式 Δ - Σ 调制器 [AMC1306M25](#) 与 [C2000 MCU TMS320F28379D](#) 之间具有 3.3V I/O 的 CMOS 接口的简化示例。由于 [AMC1306M25](#) 需要外部提供的时钟源，因此由 MCU [TMS320F28379D](#) 生成时钟信号并提供给 Δ - Σ 调制器时钟输入 `CLKIN`。同时，生成的时钟信号也会路由到 MCU Σ - Δ 滤波器模块 (SDFM) `SD1_C1` (`GPIO123`) 的时钟输入。根据系统设计，MCU 和隔离式 Δ - Σ 调制器之间的时钟接口中包括一个时钟缓冲器。 Δ - Σ 调制器的隔离式数据输出 `DOUT` 直接连接到 MCU Σ - Δ 滤波器模块 (SDFM) 数据输入 `SD1_D1` (`GPIO122`)。 (`GPIO122`)。

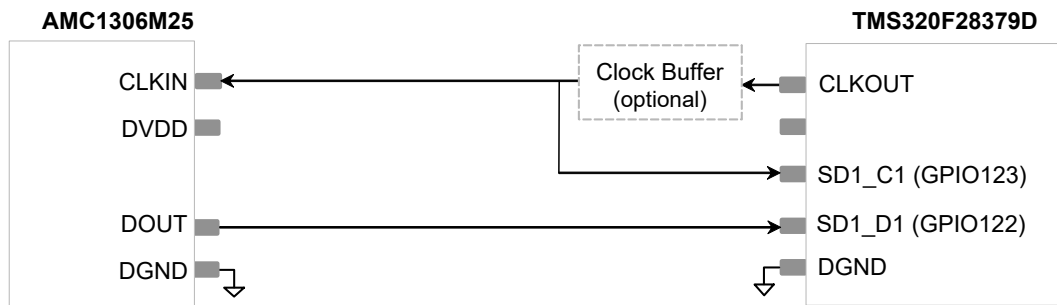


图 2-1. [AMC1306M25](#) 与 [TMS320F28379D](#) 之间数字接口的简化图

在隔离式 Δ - Σ 调制器和 MCU 之间进行有效通信需要满足相应器件数据表中描述的建立时间和保持时间要求。建立时间是指在转换时钟信号来捕获 MCU 中的数据信号之前，数据信号必须有效且保持稳定的时间量。保持时间是在时钟信号转换发生后信号必须保持有效和稳定的时间量。满足 MCU 建立时间和保持时间要求至关重要，因为任何违例都会导致采集到错误数据。隔离式 Δ - Σ 调制器的数字接口建立时间和保持时间要求与 MCU 之间不兼容会带来设计挑战。

图 2-2 概述了 [AMC1306x](#) 的建立时间和保持时间的数字接口时序，该器件支持从 5MHz 至 21MHz 的建议时钟频率 (`CLKIN`)，数据保持时间 $t_{\text{h}}(\text{MIN}) = 3.5\text{ns}$ ，数据延迟时间 $t_{\text{d}}(\text{MAX}) = 15\text{ns}$ 。

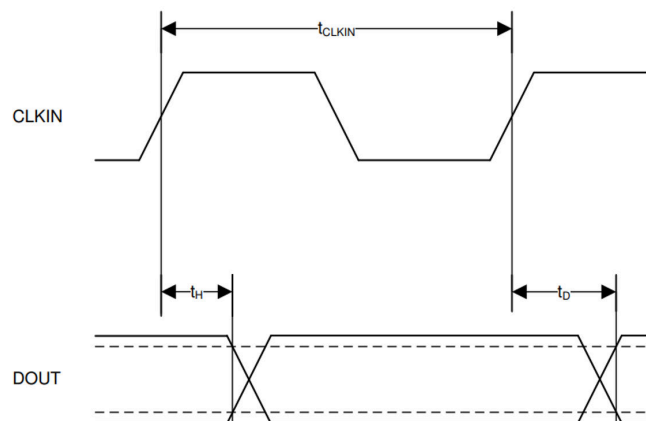


图 2-2. [AMC1306x](#) 数字接口时序

图 2-3 概述了 [TMS320F28379D](#) Σ - Δ 滤波器模块 (SDFM) 在模式 0 下的时序图。根据 SDFM 模块中 `SDx_Cy` 信号的上升时钟边沿，`SDx_Dy` 处的数据输入需要满足最短建立时间 $t_{\text{su}}(\text{SDDV-SDCH})\text{M}0$ 和最短保持时间 $t_{\text{h}}(\text{SDCH-SDD})\text{M}0$ 。

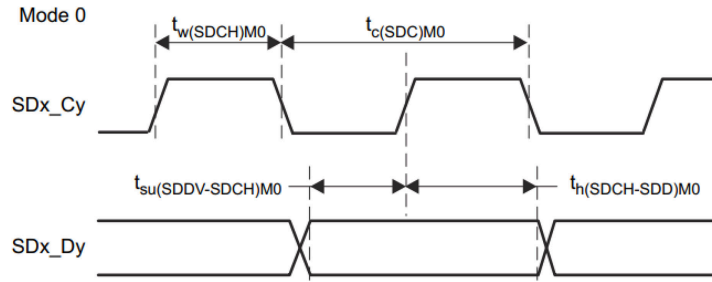


图 2-3. TMS320F28379D SDFM 时序图 - 模式 0

对于模式 0 下的 TMS320F28379D SDFM 模块，我们建议使用具有有限 GPIO (3 样本窗口) 的 SDFM 操作。此模式针对输入时钟信号 (SDx_Cy) 和数据输入 (SDx_Dy) 的随机噪声干扰提供保护，避免出现错误的比较器过流跳闸和错误的 Sinc 滤波器输出。使用 TMS320F28379D 的 200MHz 系统时钟的最短建立时间和保持时间均为 10ns： $t_{su}(SDDV-SDCH)M0 (MIN) = 10ns$ 和 $t_h(SDCH-SDD)M0 (MIN) = 10ns$ 。

这带来了设计挑战，因为 AMC1306M25 最短保持时间 $t_h(MIN)$ 为 3.5ns，但为了根据 SDx_Cy 信号的上升时钟边沿在数据输入 SDx_Dy 处维持正确的采集，SDFM 模块需要 10ns。

另一个挑战是，信号链中带有数字接口的附加元件 (如时钟缓冲器) 的传播延迟以及由 PCB 上的布线长度引入的时钟和数据信号的传播延迟都会对 SDx_Cy 和 SDx_Dy 输入之间的时序产生影响，并使数据输入的正确采集时序变得复杂。

这同样适用于具有 LVDS 接口的 $\Delta-\Sigma$ 调制器，例如 AMC1305L25。与具有 CMOS 接口类型的 AMC1306M25 $\Delta-\Sigma$ 调制器的唯一差异是，连接到具有 CMOS 接口的 MCU 时需要 LVDS 驱动器和接收器等附加元件，这会进一步增加传播延迟。图 2-4 显示了具有 LVDS 接口的隔离式 $\Delta-\Sigma$ 调制器 AMC1305L25 与具有 CMOS 接口的 MCU TMS320F28379D 之间的简化数字接口。

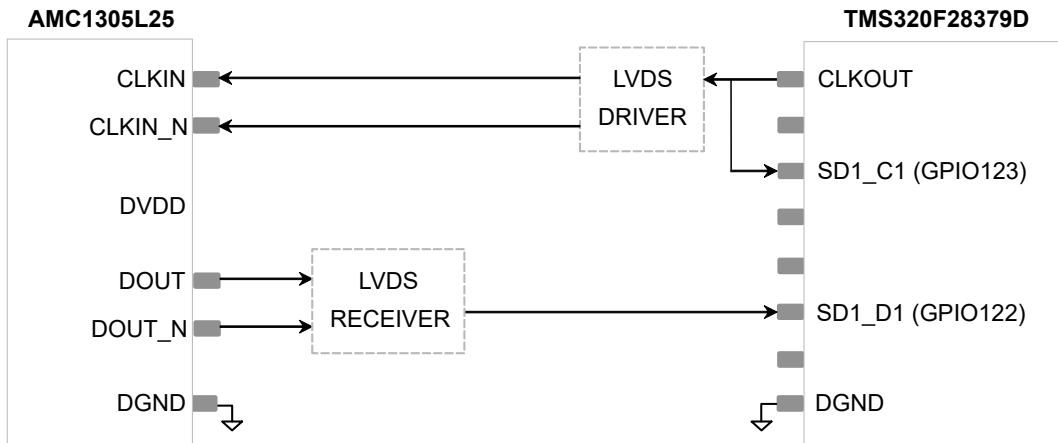


图 2-4. AMC1305L25 与 TMS320F28379D 的数字接口

图 2-5 显示了具有内部创建的时钟源且带 CMOS 接口的隔离式 $\Delta-\Sigma$ 调制器 AMC1303Mx 与带 CMOS 接口的 TMS320F28379D 之间的简化数字接口。AMC1303Mx 内部生成的时钟信号 CLKOUT 是 MCU $\Sigma-\Delta$ 滤波器模块 (SDFM) SD1_C1 (GPIO123) 的输入。 $\Delta-\Sigma$ 调制器的隔离式数据输出 DOUT 直接连接到 SDFM 的 MCU 数据输入 SD1_D1 (GPIO122)。

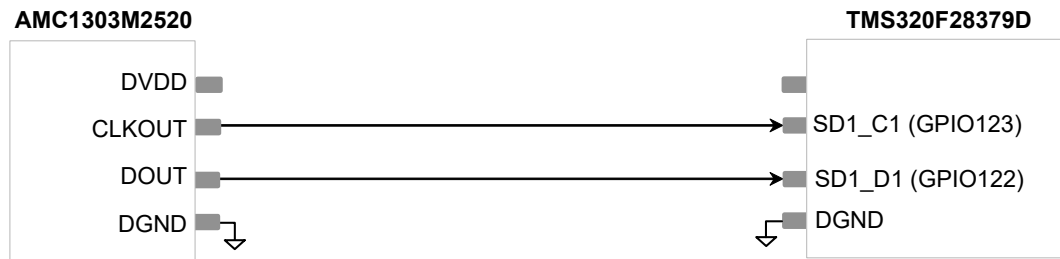


图 2-5. AMC1303M2520 用于连接 TMS320F28379D 的 3.3V CMOS 数字接口

使用具有内部时钟的隔离式调制器时，数字接口面临的挑战仅限于隔离式 Δ - Σ 调制器与 MCU 建立时间和保持时间之间的不同时序规格。如果时钟和数据信号的布线长度相同，则可以忽略 PCB 上的布线长度引入的时钟和数据信号传播延迟。通常，调制器直接连接到 MCU，无需缓冲器或电平转换器（这会增加额外的传播延迟）。

对于 10MHz 和 20MHz 时钟版本，AMC1303Mx 保持时间 $t_h(\text{MIN})$ 为 7ns，延迟时间 $t_d(\text{MAX})$ 为 15ns。面临的挑战是，AMC1303Mx 最短保持时间 $t_h(\text{MIN})$ 为 7ns，但 SDFM 模块需要 10ns 的保持时间，才能在不违反任何建立时间和保持时间的情况下在 SDx_Dy 处正确采集数据输入。

对于具有曼彻斯特编码位流输出的隔离式 Δ - Σ 调制器（例如 AMC1306E25），数据和时钟通过单线传输。因此，不需要考虑接收器件的建立时间和保持时间要求与调制器时钟信号之间的关系。

满足 MCU 建立时间和保持时间要求的一种常用方法和折衷方法是降低时钟频率。但是，降低时钟频率也会降低隔离式 Δ - Σ 调制器的数据输出速率，并增加电流测量的延时。一种更合适的方法是使用时钟边沿延迟补偿，这样可以使时钟信号的时钟边沿移到数据信号的理想采样点，从而满足建立时间和保持时间要求。使用此方法可消除时钟频率限制，从而使隔离式 Δ - Σ 调制器和系统以出色性能运行。

3 具有时钟边沿延迟补偿的设计方法

为了满足并进一步优化 MCU 建立时间和保持时间要求，从而实现可靠的数据采集，建议使用时钟边沿延迟补偿。可以通过各种方法来实施时钟边沿延迟补偿，下面概述了这些方法，并将在接下来的章节进行详细阐述：

1. 具有软件可配置相位延迟的附加时钟信号
2. 具有硬件可配置相位延迟的时钟信号
3. 时钟返回
4. MCU 的时钟反相

3.1 具有软件可配置相位延迟的时钟信号补偿

图 3-1 显示了第一种补偿方法，其中使用了具有软件可配置相位延迟的附加锁相时钟信号。对于此补偿方法，相移时钟信号 CLKOUT_delay 用作 Σ - Δ 滤波器模块 (SDFM) SD0_CLK 的时钟输入。对于其他类型的 Δ - Σ 调制器和 MCU (例如 C2000 MCU)，补偿方法适用同样的原理。

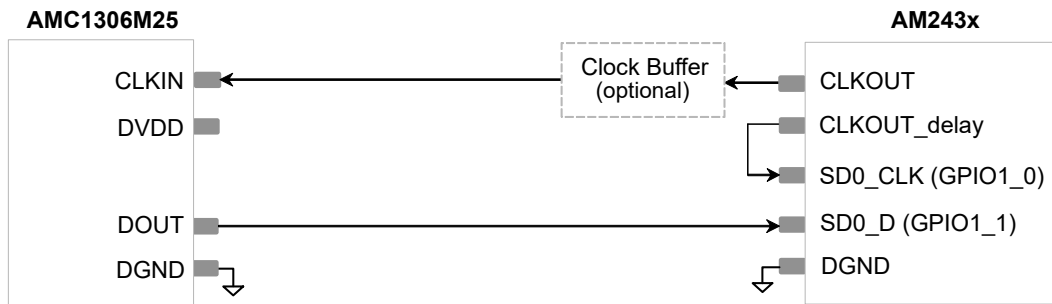


图 3-1. 具有软件可配置时钟相位延迟的 AMC1306M25 至 AM243x MCU 接口

第二个相移时钟信号的实施提供了最高的自由度和用户可配置性。这意味着，各种隔离式调制器的各种最短保持时间 $t_h(\text{MIN})$ 值可通过在软件中简单地更改相移值来进行补偿。SD0_CLK 输入端的时钟信号上升沿发生相移，使得时钟信号符合 SDFM 的数据采样点，如图 3-2 所示。在 Σ - Δ 模式下的 AM243x PRU_ICSSG PRU 时序要求中，最短建立时间为 10ns， $t_{su}(\text{SD}_D\text{-SD_CLK}) (\text{MIN}) = 10\text{ns}$ ；最短保持时间为 5ns， $t_h(\text{SD_CLK-SD}_D) (\text{MIN}) = 5\text{ns}$ 。因为 AMC1306M25 最短保持时间 $t_h(\text{MIN})$ 为 3.5ns，但可能需要 5ns，因此需要进行补偿，以便根据 SDx_CLK 信号的上升时钟沿在数据输入 SDx_D 处保持正确的采集。采用这种补偿方法后，满足 AM243x PRU_ICSSG PRU 的 Σ - Δ 模式的 10ns 最短建立时间和 5ns 保持时间要求，请参阅图 3-2。

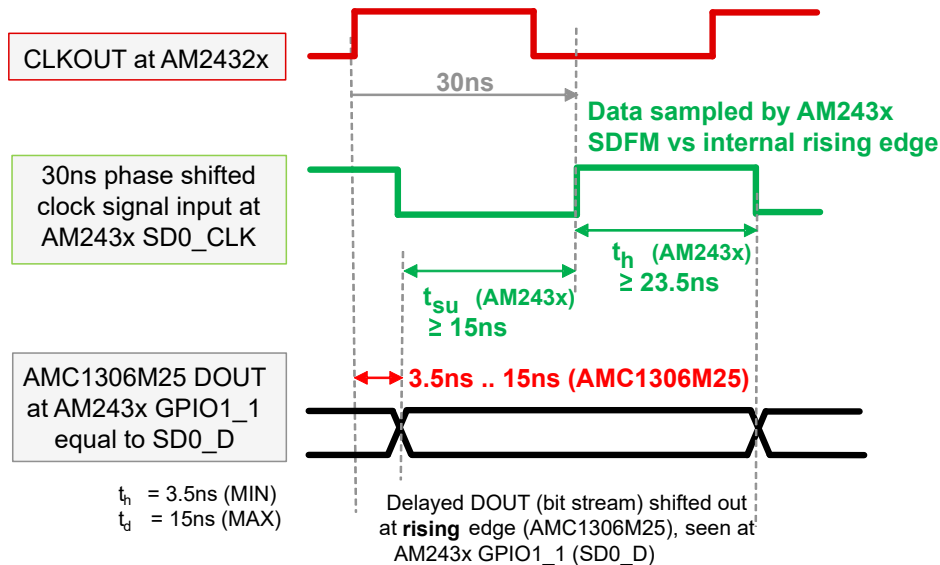


图 3-2. 在 SD0_CLK (GPIO1_1) 使用 30ns 相移时钟信号输入时的 AM243x SDFM 时序

3.2 具有硬件可配置相位延迟的时钟信号补偿

图 3-3 显示了 AMC1306M25 与 MCU 之间的数字接口具有硬件可配置相位延迟的时钟信号补偿。借助这种补偿方法，硬件中通过相位延迟实现的相移时钟信号会连接到 MCU 的 SDFM 模块的时钟输入 SDFM_CLKIN。这种类型的补偿适用于任何具有 Σ - Δ 滤波器模块的 MCU，但建议仅用于具有外部时钟源和 CMOS 接口的隔离式 Δ - Σ 调制器。

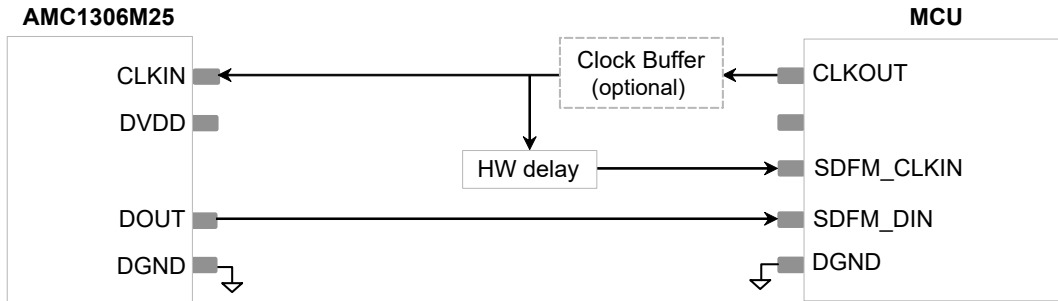


图 3-3. AMC1306M25 与 MCU 连接的数字接口，通过硬件可配置相位延迟进行补偿

为了在硬件中实现相位延迟，可以使用逻辑门或缓冲区在时钟信号中引入传播延迟。然而，在硬件中实施延迟时，延迟的值在很大程度上取决于硬件模块的传播延迟，从而限制了自由度和用户可配置性。通过具有硬件可配置相位延迟的时钟信号进行补偿的工作原理与节 3.1 中所述的原理相同。

3.3 通过时钟返回进行时钟信号补偿

图 3-4 中显示了通过时钟返回进行时钟信号补偿。采用这种补偿方法时，馈入 AMC1306M25 的时钟输入 CLKIN 的时钟信号会从 AMC1306M25 的 CLKIN 引脚路由回 MCU 的 SDFM 时钟输入 SDFM_CLKIN。

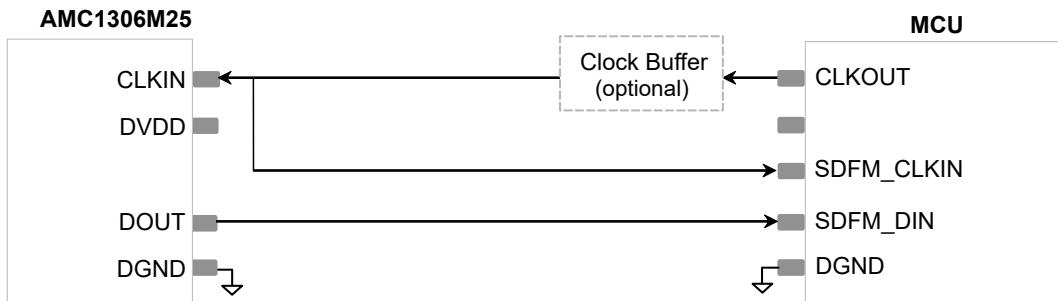


图 3-4. AMC1306M25 与 MCU 连接的数字接口，通过时钟返回进行补偿

通过使用此方法，时钟缓冲器引入的传播延迟和 PCB 布线长度引入的传播延迟对于时钟和数据信号都实现了相同的延迟。因此，时钟和数据信号中的这些延迟相互抵消。

但是，AMC1306M25 数字接口时序会保持不变，数据保持时间 $t_h(\text{MIN}) = 3.5\text{ns}$ ，数据延迟时间 $t_d(\text{MAX}) = 15\text{ns}$ 。这意味着，在构建 PCB 后需要检查时序，以便确认满足 MCU 中的 SDFM 的建立时间和保持时间要求。建议仅将此类补偿方法用于具有外部时钟源和 CMOS 接口的隔离式 Δ - Σ 调制器。

3.4 通过 MCU 的时钟反相来实现时钟信号补偿

时钟信号补偿的最后一种方法是在 MCU 实现时钟反相，适用于具有外部和内部时钟源的 Δ - Σ 调制器。在这种情况下，所选 MCU 必须能够使 GPIO 输入反相。SDFM (Σ - Δ 滤波器模块) 之前的 TMS320F28379D GPIO 输入可以配置为在任意 GPIO 使输入信号反相，如图 3-5 中所示。例如，时钟输入信号在 GPIO123 反相，因此 SD1_C1 时钟信号与 AMC1303Mx 时钟信号之间是反相的。因此，SDFM 在 GPIO123 的输入端使用外部时钟信号的下降沿来触发输入数据 SD1_D1 的采样，如图 3-6 所示。

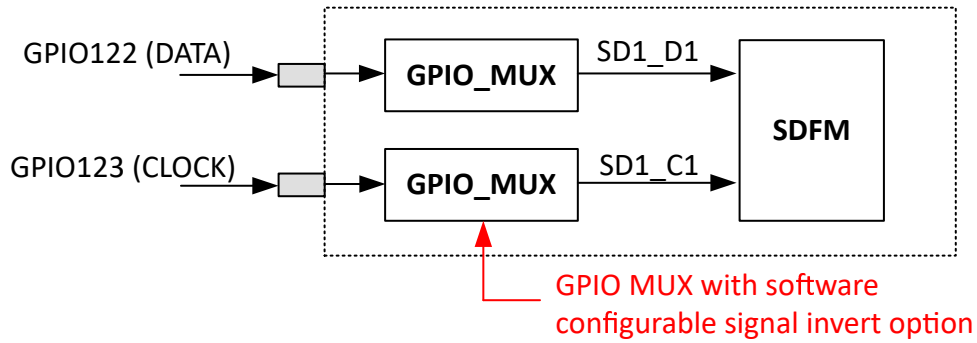


图 3-5. TMS320F28379D SDFM/GPIO 方框图

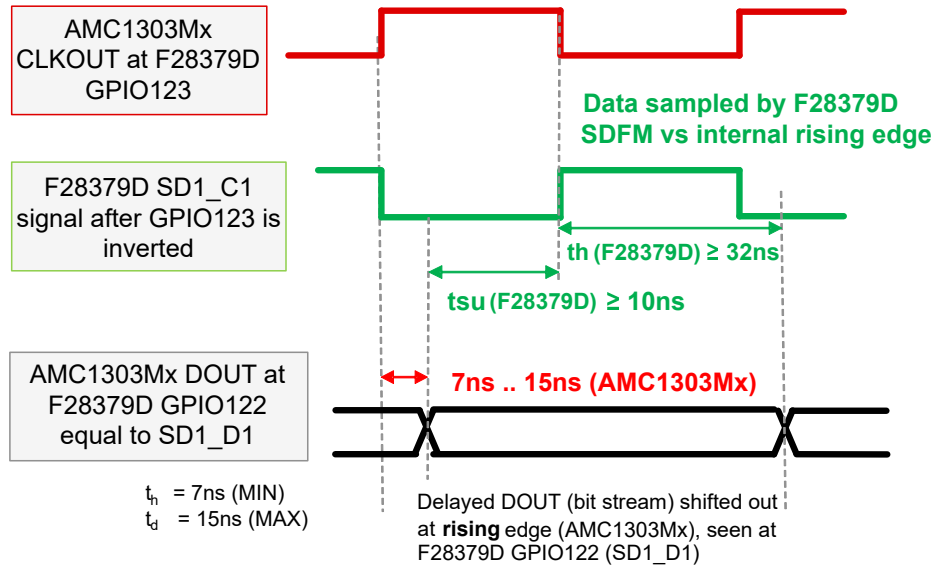


图 3-6. 在 GPIO123 使用反相时钟时的 TMS320F28379D SDFM 时序

通过使用 GPIO 使时钟输入信号反相，可以向时钟信号添加半个时钟周期的固定延迟。根据系统设置的时序规格和传播延迟，这个额外的延迟可能足以满足 SDFM 限定 GPIO (3 样本) 模式 0 满足 TMS320F28379D 至少 10ns 的建立时间和保持时间。然而，由于这种时钟信号补偿方法的额外延迟时间是固定的，无法更改，因此必须验证每个系统设计是否满足 SDFM 限定 GPIO (3 样本) 模式 0 所产生的 MCU 建立时间和保持时间。

此补偿方法也适用于 Sitara MCU，其中外部时钟信号的上升沿和下降沿都可以通过软件设置为数据采集点。

4 测试和验证

以下各节介绍了使用具有软件可配置相位延迟（如节 3.1 所述）和时钟反相（如节 3.4 所述）的附加时钟信号的时钟边沿补偿测试结果。首先，介绍测试设备和软件，然后介绍时钟信号补偿方法的测试设置、测量值和测试结果。

4.1 测试设备和软件

表 4-1 中列出了用于测量的主要测试设备。

表 4-1. 测试设备列表

说明	器件型号
AMC1306 增强型隔离式调制器评估模块	AMC1306EVM
F28379D C2000™ Delfino™ MCU LaunchPad™ 开发套件	LAUNCHXL-F28379D
AM243x Arm® MCU 通用 LaunchPad™ 开发套件	LP-AM243
高速示波器	Tektronix MSO 4104
单端探头	Tektronix P6139A

使用 [Code Composer Studio \(CCS\) 12.4.0](#) 完成软件开发和调试。CCS 是一个集成开发环境 (IDE)，支持德州仪器 (TI) 微控制器 (MCU) 和嵌入式处理器产品组合。为 TMS320F28379D 使用了基于 C2000WARE 的内部 TI 测试软件 - 用于 C2000 微控制器的 C2000Ware。对于 Sitara AM243x LaunchPad，为 Sitara™ 微控制器使用了基于 AM243x 软件开发套件 (SDK) 的内部 TI 测试软件 [MCU-Plus-SDK-AM243X 09.00.00.35](#)。有关 C2000 和 Sitara 的具体实施和软件支持，请参阅 [TI E2E 支持论坛](#)。

4.2 具有软件可配置相位延迟的时钟信号补偿测试

此测量确认使用具有软件可配置相位延迟的附加时钟信号来进行时钟信号补偿可满足建立时间和保持时间要求。使用 C2000 TMS320F28379D LaunchPad 和 Sitara AM243x LaunchPad 执行和验证此测试。

4.2.1 测试设置

图 4-1 中显示了使用附加时钟信号以及通过 AMC1306EVM 和 C2000 TMS320F28379D LaunchPad 进行软件可配置相位延迟测量的时钟信号补偿测试设置。对于此测量，使用单端探头来测量 AMC1306EVM 时钟输入 CLKIN 的时钟信号，以及 Δ - Σ 调制器的数据输出 DOUT，该输出在 SDFM 的 MCU 数据输入 SD1_D1 (GPIO122) 处测量。具有软件可编程相位延迟的时钟信号在 MCU Σ - Δ 滤波器模块 (SDFM) SD1_C1 (GPIO123) 的时钟输入端测量。AMC1306EVM 的输入引脚 AINP 和 AINN 短接在一起并接地，从而输出 50/50 的 1 和 0 信号密度。使用 EVM 上的隔离式变压器电路生成模拟电源 AVDD。隔离式调制器数字电源 DVDD (3.3V) 由 C2000 TMS320F28379D LaunchPad 提供。

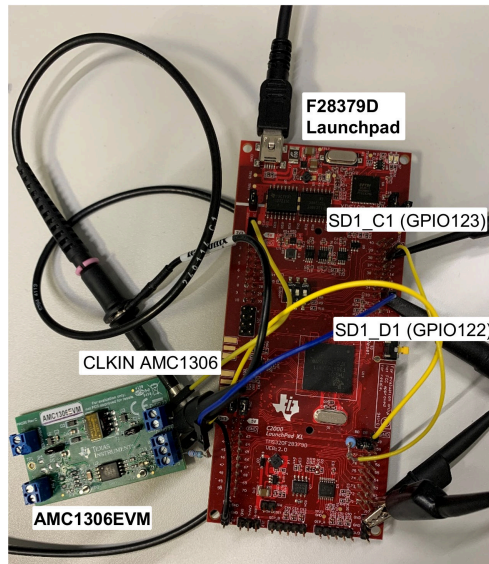


图 4-1. 使用 AMC1306EVM 和 C2000 TMS320F28379D LaunchPad 在软件中配置相位延迟的时钟信号补偿测试设置

图 4-2 显示了采用 Sitara AM243x LaunchPad 的相同测量设置以及相应的测量点。

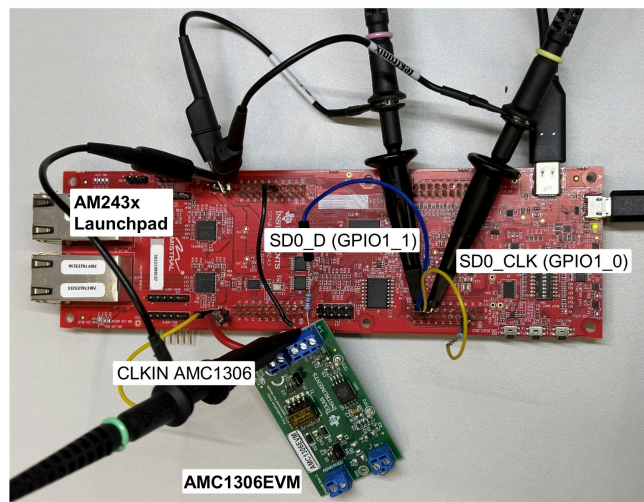


图 4-2. 使用 AMC1306EVM 和 Sitara AM243x LaunchPad 在软件中配置相位延迟的时钟信号补偿测试设置

4.2.2 测试测量结果

TMS320F28379D 正在运行一个内部 TI SDFM 软件工程，其中为 SDFM 模式配置了两个 GPIO：GPIO122 和 GPIO123。SDFM 数据滤波器配置为 Sinc3，过采样率为 64 (OSR64)。为进行测试，由 ePWM4 模块生成占空比为 50% 的 20MHz 时钟信号并将其馈入 AMC1306EVM 的 CLKIN 引脚。ePWM5 模块配置为输出具有 50% 占空比和 30ns 相移的锁相 20MHz 时钟信号。此信号馈送到 SD1_C1 (GPIO123)。请注意，AMC1306EVM DOUT 数据位流仅在上升时钟沿变化，因此每个时钟周期变化一次，如 AMC1306 数据表的第 7.11 节“开关特性”所述。

图 4-3 显示了示波器测量和接口图。馈入 AMC1306EVM CLKIN 引脚的时钟信号由通道 3 上的绿色波形表示。AMC1306EVM 输出的数据信号是通道 2 上的红色 SD1_D1 (GPIO122) 信号。馈送到 SD1_C1 (GPIO123) 的相移时钟信号是通道 1 上测量的蓝色波形。当 SDFM 模块在相移时钟信号 SD1_C1 (GPIO123) 的上升沿对数据信号进行采样时，产生的建立时间约为 18ns，产生的保持时间约为 24ns。这满足了 SDFM 限定 GPIO (3 样本) 模式 0 最短 10ns 的 TMS320F28379D 建立时间和保持时间要求。此外，此设计还提供了合适的裕度，为系统传播延迟变化 (正或负) 留出容差。

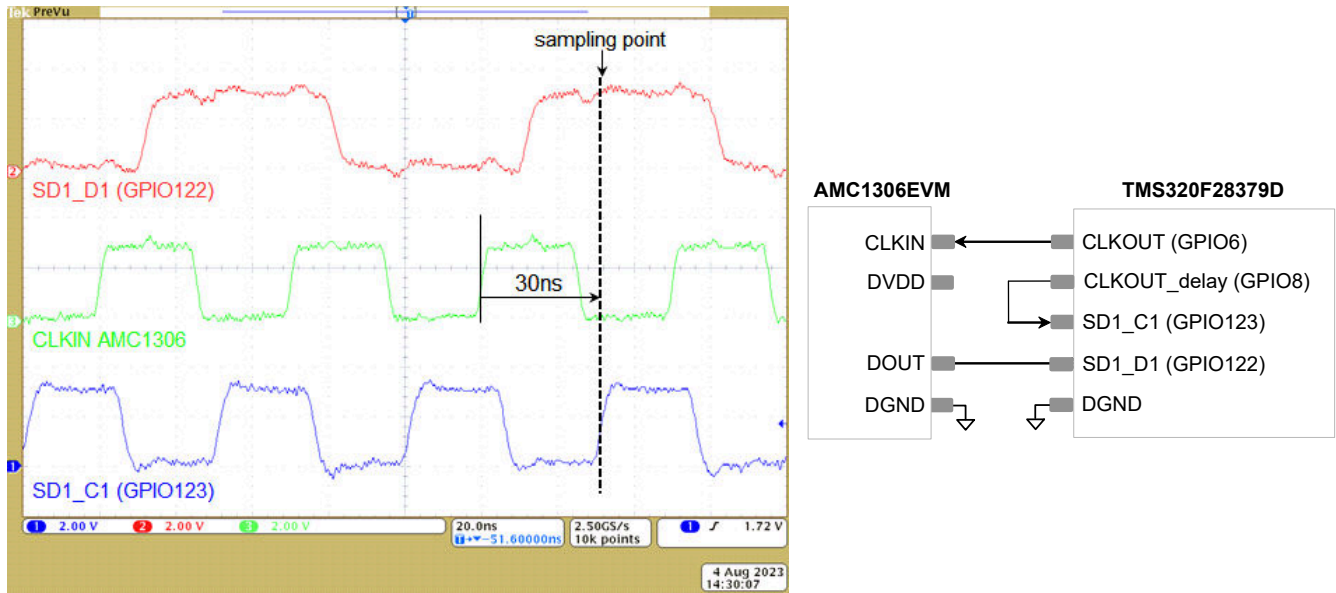


图 4-3. 使用具有软件可配置相位延迟的时钟信号补偿的 AMC1306EVM 和 TMS320F28379D 数字接口时序测量结果

图 4-4 显示了与使用 Sitara AM243x LaunchPad 执行的测试相同的测量结果。总之，为了满足 MCU 建立时间和保持时间要求，允许使用具有软件可配置相位延迟的附加时钟信号来进行时钟信号补偿。此方法具有很大的自由度，因为不仅可以配置相移值，而且由于只需要一个额外的 GPIO 引脚来实施相移时钟信号，因此该方法也适用于各种 MCU。

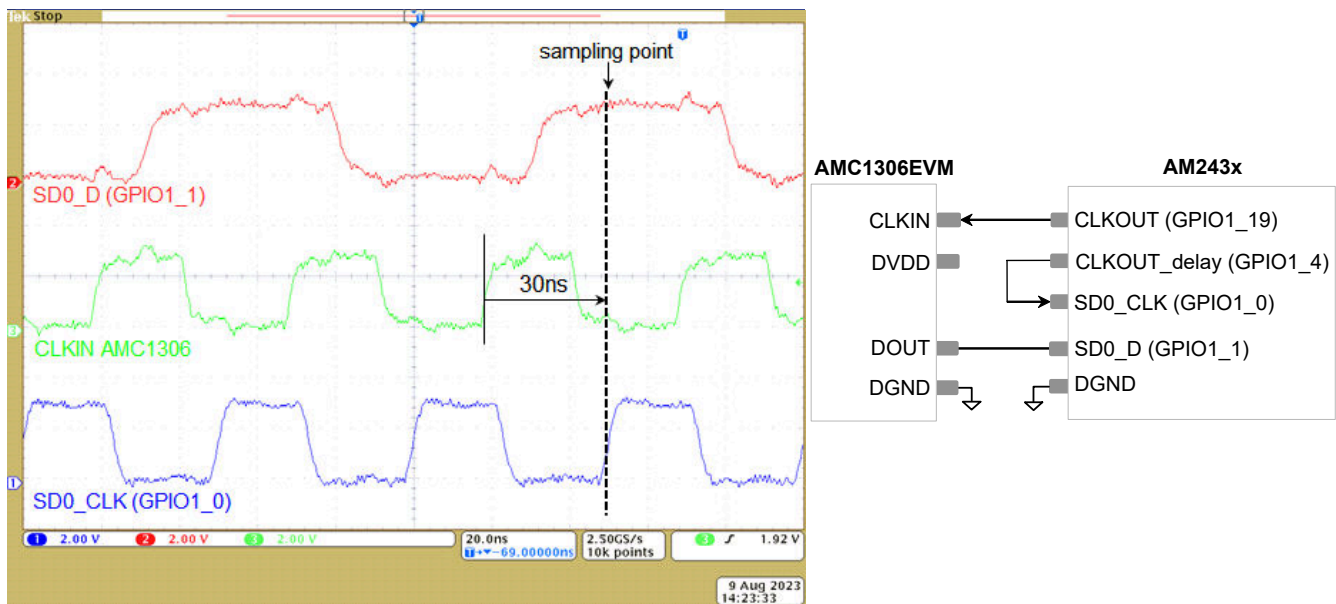


图 4-4. 使用具有软件可配置相位延迟的时钟信号补偿的 AMC1306EVM 和 AM243x 数字接口时序测量结果

4.3 通过 MCU 上的时钟反相进行时钟信号补偿的测试

此配置已通过 C2000 TMS320F28379D LaunchPad 进行测试和验证。

4.3.1 测试设置

图 4-5 中显示了使用 C2000 TMS320F28379D LaunchPad 在 MCU 使用时钟反相来进行时钟信号补偿的测试设置。对于此测量，MCU 生成两个测试信号。一个信号连接到 MCU SDFM 的时钟输入 SD1_C1 (GPIO123)，另一个信号连接到 SDFM 的数据输入 SD1_D1 (GPIO122)。

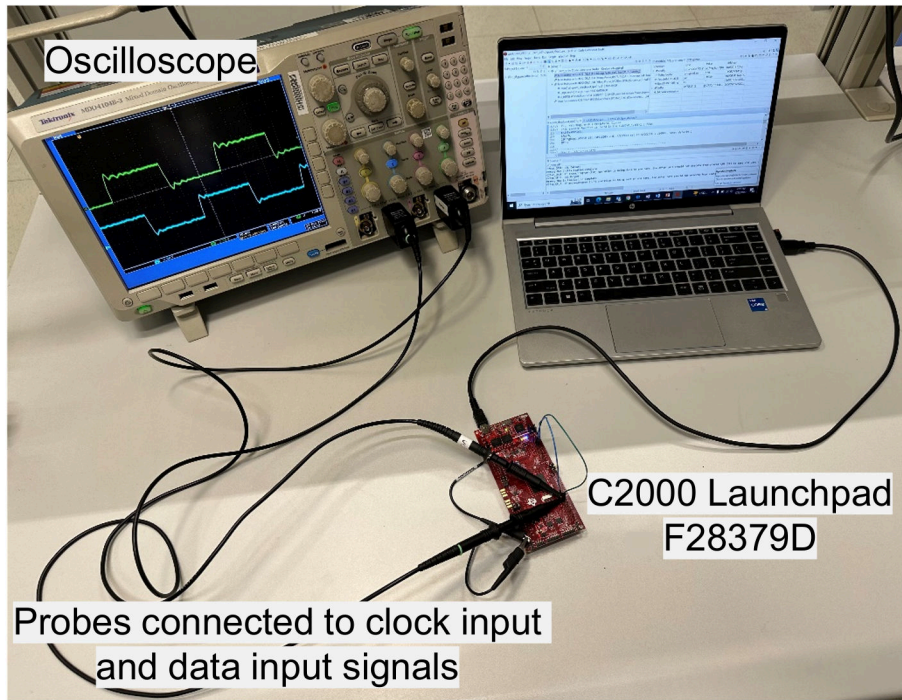


图 4-5. 数字接口时序验证的测试设置

4.3.2 测试测量结果

TMS320F28379D 正在运行一个内部 TI SDFM 软件工程，其中为 SDFM 模式配置了两个 GPIO：GPIO122 和 GPIO123。SDFM 滤波器配置为 Sinc3 和 OSR™ 64 滤波器。Sinc3 OSR64 滤波器输出一个 16 位二进制补码整数，最大满标量程为 +16384 至 -16384。

为执行测试，两个占空比为 50% 的 90 度相移 10MHz 时钟信号分别馈送到 GPIO123 (SD1_C1) 和 GPIO122 (SD1_D1)。请注意，AMC1306EVM DOUT 数据位流仅在上升时钟沿变化，因此每个时钟周期变化一次。对于此测试，SD1_D1 数据每半个时钟周期在 0 和 1 之间切换一次。这与 AMC1306EVM DOUT 数据信号（该信号在每个时钟周期发生变化）不同。

由于应用了这个特定的测试信号，GPIO122 (SD1_D1) 的输入数据在 GPIO123 的上升时钟沿始终为逻辑 ‘1’，在下降沿始终为逻辑 ‘0’。因此，Sinc3 OSR 64 滤波器的输出取决于在 SDFM 中的哪个时钟边沿对测试数据进行采样；如果 GPIO123 处没有时钟反相，则 Sinc3 OSR 64 滤波器输出为 16384（始终对 ‘1’ 采样），如果 GPIO123 处有时钟反相（始终对 ‘0’ 采样），则输出为 -16384。

4.3.2.1 测试结果 - GPIO123 时钟输入无时钟反相

图 4-6 显示了对输入到 GPIO123 的时钟信号 SD1_C1 和输入到 GPIO122 的相移数据信号 SD1_D1 的示波器测量。在此测量中，软件不会使 GPIO123 反相，如下所示。

```
// Set 3-sample qualifier for GPIO122 and GPIO123 and do not invert GPIO123
GPIO_SetupPinOptions(122, GPIO_INPUT, GPIO_QUAL3); // GPIO123 not inverted
GPIO_SetupPinMux(122,GPIO_MUX_CPU1,7); // MUX position 7 for SD1_D1
GPIO_SetupPinMux(123,GPIO_MUX_CPU1,7); // MUX position 7 for SD1_C1
```

由 TMS320F28379D SDFM 在 SD1_C1 上升沿对数据 SD1_D1 进行采样。这对应于 GPIO123 的同相时钟信号的上升沿。TMS320F28379D 采样的数据始终为逻辑 ‘1’，通过 Code Composer Studio™ (CCS) 中的 Sinc3 OSR64 滤波器 = +16384 输出进行验证，如下所示。

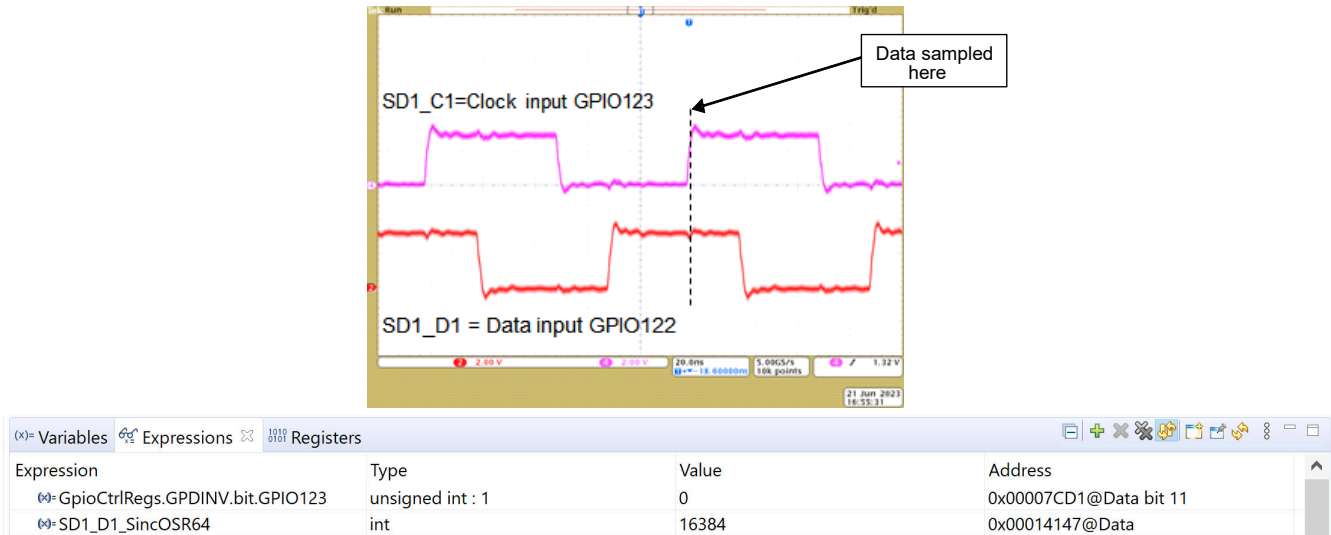


图 4-6. CCS 中的时钟和数据输入测试信号 (同相 GPIO123) 和 Sinc3 OSR 64 滤波器输出

4.3.2.2 测试结果 - GPIO123 时钟输入的时钟反相

图 4-7 显示了输入到 GPIO123 的时钟信号 SD1_C1 和输入到 GPIO122 的相移数据信号 SD1_D1。在此测试设置中，GPIO123 由软件反相，如下所示。

```
// Set 3-sample qualifier for GPIO122 and GPIO123 and do not invert GPIO123
GPIO_SetupPinOptions(123, GPIO_INPUT, GPIO_INVERT | GPIO_QUAL3);
GPIO_SetupPinMux(122,GPIO_MUX_CPU1,7); // MUX position 7 for SD1_D1
GPIO_SetupPinMux(123,GPIO_MUX_CPU1,7); // MUX position 7 for SD1_C1
```

现在，F28379D SDFM 在 SD1_C1 的下降沿对数据 SD1_D1 进行采样，这对应于 GPIO123 输入端反相时钟信号的上升沿。F28379D 采样的数据始终为逻辑 ‘0’，通过 Code Composer Studio 中的 Sinc3 OSR64 滤波器 = -16384 的输出进行验证，如下所示。

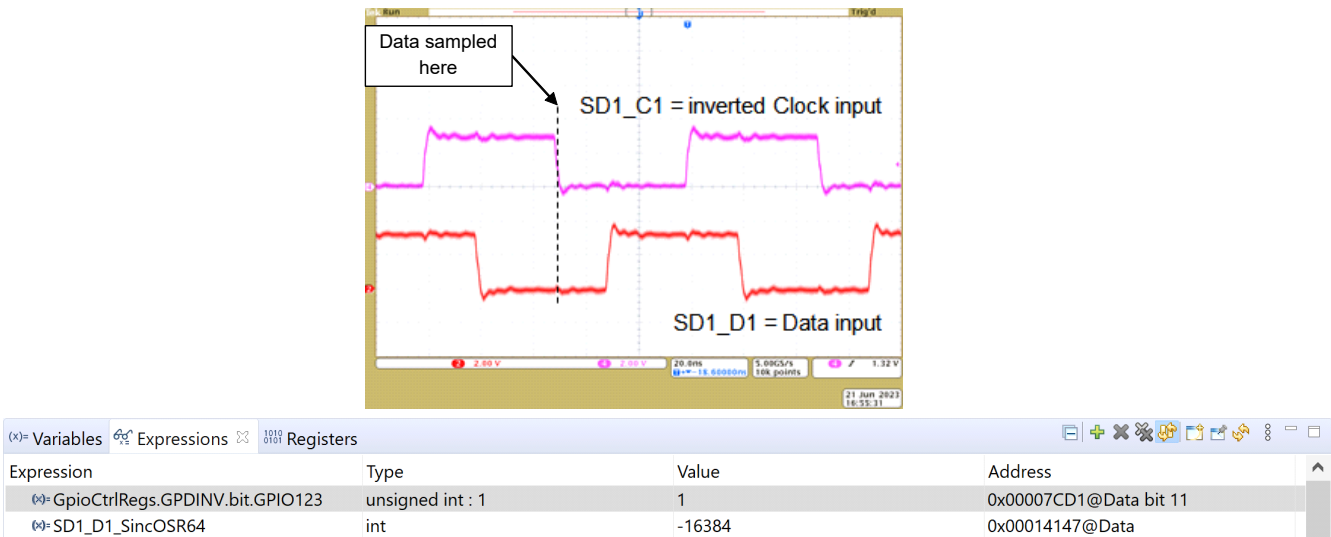


图 4-7. CCS 中的时钟和数据输入测试信号 (同相 GPIO123) 和 Sinc3 OSR 64 滤波器输出

总之，在软件中通过使 GPIO 输入的时钟输入反相来进行时钟信号补偿的方法经过验证。通过使时钟反相，时钟信号会添加半个时钟周期的固定延迟，这个延迟足以满足 SDFM 限定 GPIO (3 样本) 模式 0 最短 10ns 的 TMS320F28379D 建立时间和保持时间要求。但是，如果可满足为 SDFM 限定 GPIO (3 样本) 模式 0 产生的 MCU 建立时间和保持时间的要求，则需要单独检查每个系统设计。

4.4 通过计算工具进行数字接口时序验证

为了模拟和验证 MCU 和隔离式 Δ - Σ 调制器之间的数字接口时序，我们开发了一种计算工具。这里选择了常用的隔离式 Δ - Σ 调制器 [AMC1306M25](#) 和 [AMC1305L25](#) 进行数字接口时序分析。[AMC1305L25](#) 具有 LVDS 接口类型，在使用 CMOS 接口连接 MCU 时，需要 LVDS 驱动器和 LVDS 接收器。用户可以单独选择 MCU，因为在计算工具中只输入建立时间和保持时间要求。下面分步说明了使用计算工具来优化 [AMC1305L25](#) 与 C2000 MCU [TMS320F28379D](#) 之间的数字接口时序。

4.4.1 不使用补偿方法的数字接口

C2000 MCU [TMS320F28379D](#) 在 200MHz 系统时钟和模式 0 下以 SDFM GPIO 输入限定 (3 样本窗口) 选项运行。最短建立时间和保持时间都是 10ns：在计算工具中输入 $t_{su}(SDDV-SDCH)M0(MIN) = 10ns$ 和 $t_h(SDCH-SDD)M0(MIN) = 10ns$ 。此外，还将输入 LVDS 驱动器 [DSLVD51047](#) 和 LVDS 接收器 [DSLVD51048](#) 的传播延迟作为参考。在隔离式 Δ - Σ 调制器时钟输入处使用 20MHz 时钟信号，这是数据表中指定的最大时钟频率，当 [AMC1305L25](#) 的数据延迟时间 t_D 等于数据表中给出的最小规格 (即 $t_D(MIN) = 0ns$) 时，违反了 MCU 建立时间要求，如表 2 所示。

表 4-2. 在 20MHz 时钟频率下使用 [AMC1305L25](#) 的 C2000 MCU [TMS320F28379D](#) 数字接口时序结果

最短建立时间 @MCU	5.6ns
最长建立时间 @MCU	23.3ns
最短保持时间 @MCU	26.7ns
最长保持时间 @MCU	44.4ns

4.4.2 常用方法 - 降低时钟频率

满足 MCU 时序要求的折衷办法是降低调制器时钟频率。在本例中，17MHz 的时钟频率即可满足 MCU 的建立时间和保持时间要求。表 4-3 显示了计算出的建立时间和保持时间，包括 17MHz 时钟频率下的最小值和最大值。最短建立时间与 MCU 建立时间要求之间的裕度为 0ns。这意味着系统中的容差有可能导致程序错误的数据采集。通过进一步降低时钟频率，可以在系统中实现更大的容差裕度，但这会对系统性能产生负面影响。

表 4-3. 在 17MHz 时钟下使用 [AMC1305L25](#) 时的 C2000 MCU [TMS320F28379D](#) 数字接口时序

最短建立时间 @MCU	10.0ns
最长建立时间 @MCU	27.7ns
最短保持时间 @MCU	31.1ns
最长保持时间 @MCU	48.8ns

4.4.3 具有软件可配置相位延迟的时钟边沿补偿

采用具有软件可配置相位延迟的时钟边沿补偿的数字接口如图 4-8 显示。时序图显示时钟频率为 20MHz 的时钟信号 (表示馈入隔离式 Δ - Σ 调制器的时钟信号) 作为第一个信号。时序图中绘制的第二个信号表示在数据表中给出的典型规格下隔离式 Δ - Σ 调制器的数据输出。第三个信号表示 20MHz 时钟信号，相对于第一个信号相移 10ns，该信号馈送到 MCU SDFM 的时钟输入。

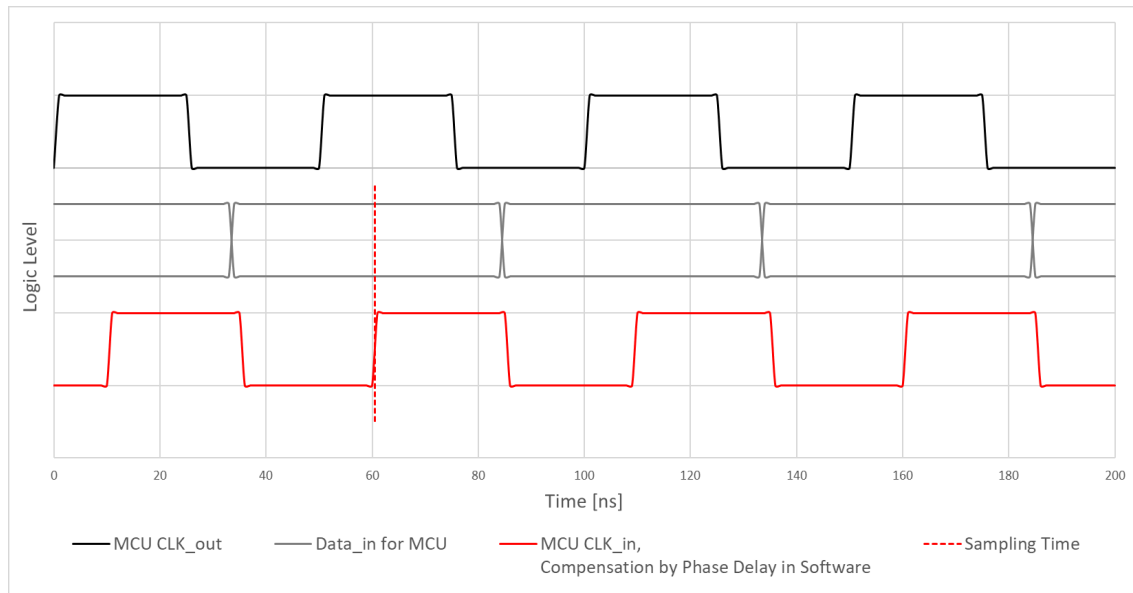


图 4-8. 基于数据表中的典型规格、在 20MHz 的时钟频率下且使用软件可配置的相位延迟进行时钟边沿补偿的 C2000 与 AMC1305L25 的数字接口时序图

表 4-4 显示了计算出的建立时间和保持时间，包括最小值和最大值。由于相位延迟可在软件中配置，因此可以选择相位延迟的值，以使数据采集时序位于数据信号的中央。这样就可为建立时间和保持时间提供最大的可能裕度，从而使系统中的容差不会影响数据采集。该计算工具提供数字时序接口的裕度，有助于了解系统的可接受容差。对于选定的 10ns 相位延迟，最短建立时间为 15.6ns，在减去 10ns 的 MCU 建立时间要求后，得到 5.6ns 的裕度。相应地计算最短保持时间的裕度，此裕度为 6.7ns。

表 4-4. AMC1305L25 在 20MHz 时钟频率下且具有软件可配置相位延迟时的 TMS320F28379D 数字接口时序

相位延迟	建议的相位延迟		所选相位延迟
	min	4.4ns	
	max	16.7ns	
最短建立时间 @MCU			15.6ns
最长建立时间 @MCU			33.3ns
最短保持时间 @MCU			16.7ns
最长保持时间 @MCU			34.4ns

5 结语

时钟边沿延迟补偿有助于通过隔离式 Δ - Σ 调制器和 MCU 数字接口满足建立时间和保持时间要求，而不必降低调制器时钟频率。这使得系统能够以完整性能运行。

可以通过多种方法实现时钟边沿延迟补偿，这些补偿方法包括：

- 具有软件可配置相位延迟的附加时钟信号
- 具有硬件可配置相位延迟的时钟信号
- 时钟返回
- MCU 的时钟反相

对于常用的隔离式 Δ - Σ 调制器型号，我们更详细地分析了具有软件可配置相位延迟的附加时钟信号和 MCU 时钟反相等补偿方法，并通过 AMC1306EVM 评估模块和 C2000 TMS320F28379D LaunchPad 以及选为 MCU 的 Sitara AM243x LaunchPad 进行了验证。测试结果对于具有 CMOS 接口和 SDFM 的 MCU 以及在处理 PRU 时没有 SDFM 的 Sitara MCU 同样适用。

表 5-1 显示了各种时钟信号补偿方法的优点和缺点。在下表中，使用缩写“软件相位延迟”和“硬件相位延迟”来表示具有软件可配置相位延迟的补偿和具有硬件可配置相位延迟的补偿。

表 5-1. 时钟边沿补偿方法的比较

方法	优势	缺点
软件相位延迟	<ul style="list-style-type: none"> • 补偿任何传播延迟 • 允许使用高时钟频率来实现可靠的通信 • 实施精确的相位延迟 • 可在运行时更改 • 无额外 BOM 成本 	<ul style="list-style-type: none"> • 需要一个额外的 MCU GPIO 和内部锁相时钟源 • 额外的 MCU 软件
硬件相位延迟	<ul style="list-style-type: none"> • 无需更改 MCU 软件 • 无需额外的 MCU GPIO 	<ul style="list-style-type: none"> • 补偿取决于实施的硬件延迟 • 硬件元件产生的相位延迟精度存在容差 • 在运行时无法进行任何更改 • 增加 BoM 成本
时钟返回	<ul style="list-style-type: none"> • 无需进行软件和硬件设置 	<ul style="list-style-type: none"> • 并非适用于所有配置 • 布局调整 • 更长的时钟信号对瞬态噪声更敏感
时钟反相	<ul style="list-style-type: none"> • 如果通过时钟周期的一半进行补偿可解决时序差异，那么实施会很简单 	<ul style="list-style-type: none"> • 并非适用于所有配置 • 固定补偿，仅为时钟周期的一半 • MCU 需要能够在 GPIO 输入处使时钟信号反相

根据 Δ - Σ 调制器的类型，因外部或内部时钟源和 CMOS 或 LVDS 接口存在差异，不同的时钟信号补偿方法可能各有优劣。表 5-2 比较了常用的各种 Δ - Σ 调制器的建议补偿方法。

表 5-2. 为具有内部或外部时钟的调制器建议的时钟边沿补偿方法

方法	AMC1306M25 外部时钟 (CMOS)	AMC1305L25 外部时钟 (LVDS)	AMC1303M2520/10 内部时钟 (CMOS)
软件相位延迟	+	+	不适用
硬件相位延迟	o	o	o
时钟返回	o	-	不适用
时钟反相	o	o	+

对于需要外部时钟的调制器，具有软件可配置相位延迟的时钟信号补偿可提供出色性能，如果固定的一半时钟周期可满足要求，则随后在 MCU 实施时钟反相方法。这两种时钟信号补偿方法都有助于满足 MCU 的建立时间和保持时间要求，尤其是在调制器时钟频率较高时。使用 Δ - Σ 调制器 AMC1306M25 和 AMC1305L25 时，可使用以下计算工具来验证 MCU 的建立时间和保持时间要求。

6 参考资料

- 德州仪器 (TI), [在电机驱动器中使用隔离式 \$\Delta\$ - \$\Sigma\$ 调制器提高信号完整性](#) 应用报告
- 德州仪器 (TI), [适用于高电压系统的高性能隔离式 ADC](#) 概述
- 德州仪器 (TI), [比较隔离式放大器和隔离式调制器](#) 白皮书
- 德州仪器 (TI), [AMC1306x 具有高 CMTI 的小型、高精度、增强型隔离式 \$\Delta\$ - \$\Sigma\$ 调制器](#) 数据表
- 德州仪器 (TI), [AMC1305x 高精度、增强型隔离式 \$\Delta\$ - \$\Sigma\$ 调制器](#) 数据表
- 德州仪器 (TI), [AMC1303x 具有内部时钟的小型、高精度、增强型隔离式 \$\Delta\$ - \$\Sigma\$ 调制器](#) 数据表
- 德州仪器 (TI), [TMS320F2837xD 双核微控制器](#) 数据表
- 德州仪器 (TI), [TMS320F2837xD 双核微控制器](#) 技术参考手册
- 德州仪器 (TI), [AM243x Sitara™ 微控制器](#) 数据表
- 德州仪器 (TI), [MCU-PLUS-SDK-AM243X 软件开发套件 \(SDK\)](#) 工具

7 Revision History

Changes from Revision * (December 2023) to Revision A (January 2024)	Page
• 更新了整个文档中的表格、图和交叉参考的编号格式.....	1
• 更新了计算工具超链接.....	15

重要声明和免责声明

TI“按原样”提供技术和可靠性数据（包括数据表）、设计资源（包括参考设计）、应用或其他设计建议、网络工具、安全信息和其他资源，不保证没有瑕疵且不做任何明示或暗示的担保，包括但不限于对适销性、某特定用途方面的适用性或不侵犯任何第三方知识产权的暗示担保。

这些资源可供使用 TI 产品进行设计的熟练开发人员使用。您将自行承担以下全部责任：(1) 针对您的应用选择合适的 TI 产品，(2) 设计、验证并测试您的应用，(3) 确保您的应用满足相应标准以及任何其他功能安全、信息安全、监管或其他要求。

这些资源如有变更，恕不另行通知。TI 授权您仅可将这些资源用于研发本资源所述的 TI 产品的应用。严禁对这些资源进行其他复制或展示。您无权使用任何其他 TI 知识产权或任何第三方知识产权。您应全额赔偿因在这些资源的使用中对 TI 及其代表造成的任何索赔、损害、成本、损失和债务，TI 对此概不负责。

TI 提供的产品受 [TI 的销售条款](#) 或 [ti.com](#) 上其他适用条款/TI 产品随附的其他适用条款的约束。TI 提供这些资源并不会扩展或以其他方式更改 TI 针对 TI 产品发布的适用的担保或担保免责声明。

TI 反对并拒绝您可能提出的任何其他或不同的条款。

邮寄地址：Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

Copyright © 2024，德州仪器 (TI) 公司