

引言

DFSDM（数字滤波器，用于 $\Sigma\Delta$ 调制器）是一种创新的嵌入式外设，可用于一系列STM32微控制器（参见 [表 1: 适用产品](#)），对于处理外部模拟信号的应用尤为重要。

虽然DFSDM是纯数字外设，但它可以支持各种外部模拟前端。通过将模拟前端部件（ $\Sigma\Delta$ 调制器）保持在微控制器外部，用户可以根据应用要求（模拟量程、噪声、采样速度）灵活选择模拟属性。

来自 $\Sigma\Delta$ 调制器的原始转换数字数据由DFSDM外设（数字滤波）进行处理。DFSDM配置足够灵活，可支持各种转换数据属性：输出数据宽度、输出数据速率、输出频率范围。

从应用的角度来看，带有外部模拟前端的DFSDM就像一个ADC转换器。DFSDM中还提供ADC的典型附加功能，如模拟看门狗、极值检测器和偏移校正。

参考：

[TUTORIAL] 在本文档中，[\[TUTORIAL\]](#)指的是以Microsoft® Excel®工作簿形式提供的DFSDM模拟器，可以使用带有关键字“DFSDM_tutorial”的主页搜索引擎，可从www.st.com下载。

表1. 适用产品

类型	适用范围	系列、产品线、参考
微控制器	完整系列	STM32L4系列、STM32L4+系列、STM32H7系列
	完整产品线	STM32F412产品线、STM32F413/423产品线
	STM32F76xxx	STM32F765BG、STM32F765BI、STM32F765IG、STM32F765II、STM32F765NG、STM32F765NI、STM32F765VG、STM32F765VI、STM32F765ZG、STM32F765ZI、STM32F767BG、STM32F767BI、STM32F767IG、STM32F767II、STM32F767NG、STM32F767NI、STM32F767VG、STM32F767VI、STM32F767ZG、STM32F767ZI、STM32F768AI、STM32F769AG、STM32F769AI、STM32F769BG、STM32F769BI、STM32F769IG、STM32F769II、STM32F769NG、STM32F769NI、STM32F769SL
	STM32F77xxx	STM32F777BI、STM32F777II、STM32F777NI、STM32F777VI、STM32F777ZI、STM32F778AI、STM32F779AI、STM32F779BI、STM32F779II、STM32F779NI

目录

1	使用DFSDM的A/D转换原理概述	6
1.1	使用DFSDM的A/D转换基础概念	6
1.2	$\Sigma\Delta$ 调制器	6
1.3	数字滤波器	8
2	$\Sigma\Delta$调制原理（外部模拟前端机能，模拟）	9
2.1	$\Sigma\Delta$ 调制原理	9
2.2	$\Sigma\Delta$ 调制的优点	11
2.2.1	噪声整形	11
2.2.2	A/D转换的线性	12
2.2.3	可扩展的ADC分辨率	12
2.3	$\Sigma\Delta$ 调制的缺点	13
2.3.1	偏移和增益误差	13
2.3.2	低数据率	13
2.4	$\Sigma\Delta$ 调制器的模拟	14
2.4.1	使用[TUTORIAL]进行模拟	14
3	数字滤波 - 原理和设计	15
3.1	功能描述	15
3.2	Sinc滤波器功能示例 - 提高分辨率	15
3.3	Sinc滤波器的硬件设计	17
4	DFSDM外设操作	22
4.1	框图	22
4.2	DFSDM组件	22
4.2.1	串行收发器	22
4.2.2	并行收发器	23
4.2.3	数字滤波器	23
4.2.4	积分器	23
4.2.5	输出数据单元	23
4.2.6	模拟看门狗	24
4.2.7	短路检测器	24
4.2.8	极值检测器	25

4.3	DFSDM模拟	25
4.3.1	$\Sigma\Delta$ 调制器原理	25
4.3.2	DFSDM滤波模拟（滤波器和积分器）	26
4.3.3	Sinc滤波器的频率特性	27
4.3.4	$\Sigma\Delta$ 调制的噪声整形	28
4.3.5	高阶滤波器操作	31
4.3.6	$\Sigma\Delta$ DAC模拟	33
4.3.7	高通滤波器模拟	34
4.4	DFSDM的其他功能	35
4.4.1	支持数字麦克风 (MEMS)	35
4.4.2	支持波束整形	37
4.4.3	音频时钟支持 - 独立时钟操作	39
4.5	DFSDM功耗优化	40
4.5.1	睡眠模式下的功耗优化	40
5	DFSDM外设配置教程	42
5.1	配置介绍	42
5.2	时钟配置	42
5.3	收发器	43
5.3.1	串行收发器配置	43
5.3.2	并行收发器	46
5.4	滤波器	47
5.4.1	Sinc滤波器	47
5.4.2	积分器	49
5.5	模拟看门狗	50
5.6	短路检测器	50
5.7	脉冲跳跃	51
5.8	使用 [TUTORIAL] 进行配置	52
6	结论	54
7	版本历史	55

表格索引

表1. 适用产品 1

表2. DFSDM应用示例 44

表3. DFSDM模拟看门狗参数 50

表4. 文档版本历史 55

表5. 中文文档版本历史 55



图片索引

图1.	使用DFSDM的A/D转换框图	6
图2.	PWM调制示例	7
图3.	$\Sigma\Delta$ 调制示例	8
图4.	$\Sigma\Delta$ 调制原理	9
图5.	$\Sigma\Delta$ 调制器电压时序图	10
图6.	PWM和 $\Sigma\Delta$ 调制的频谱	12
图7.	$\Sigma\Delta$ 调制器模拟	14
图8.	每个滤波长度有一个1的三阶滤波器输出示例	16
图9.	输入脉冲密度较高的三阶滤波器输出示例	17
图10.	简单滑动平均法实现的基本原理图	18
图11.	简化Sinc滤波器设计 - 步骤1	18
图12.	简化Sinc滤波器设计 - 步骤2	19
图13.	简化Sinc滤波器设计 - 步骤3	20
图14.	简化Sinc滤波器设计 - 步骤4	20
图15.	高阶Sinc滤波器的实现	21
图16.	DFSDM外设框图	22
图17.	$\Sigma\Delta$ 调制器模拟	26
图18.	滤波模拟	27
图19.	滤波器频率特性	28
图20.	$\Sigma\Delta$ 和PWM调制信号（用于频谱比较）	29
图21.	$\Sigma\Delta$ 信号和PWM信号的频谱	30
图22.	$\Sigma\Delta$ 信号和PWM信号的频谱，信号幅度降至满量程的10%	31
图23.	高阶滤波器 - 多重平均原理	32
图24.	一阶 $\Sigma\Delta$ DAC原理	33
图25.	$\Sigma\Delta$ DAC模拟	34
图26.	HP滤波器模拟	35
图27.	MEMS麦克风输出（L和R声道）	36
图28.	MEMS麦克风连接到DFSDM（支持立体声）	36
图29.	波束整形原理	37
图30.	用于波束整形的脉冲跳跃实现	38
图31.	脉冲跳跃示例 (FOSR=8)	39
图32.	Sinc滤波器频率特性形状	48
图33.	[TUTORIAL] 中的DFSDM配置	53

1 使用DFSDM的A/D转换原理概述

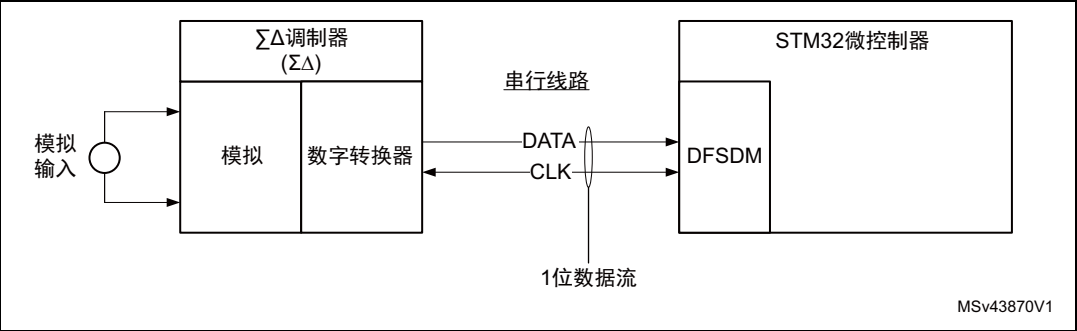
本文档支持基于Arm^{®(a)}的设备。



1.1 使用DFSDM的A/D转换基础概念

使用DFSDM的模数转换基础框图请参见 图 1。

图1. 使用DFSDM的A/D转换框图



外部 $\Sigma\Delta$ 调制器将模拟信号转换为数字1位流（DATA和CLK信号），从而对外部模拟信号进行处理。1位流是逻辑1和0的快速串行线流：DATA信号由CLK（时钟信号）采样。在足够长的持续时间内计算的这些逻辑1和0的平均值表示模拟输入值。取平均值周期的持续时间决定了模拟输入信号捕获的精度。

由STM32微控制器DFSDM外设（DFSDM = 数字滤波器，用于 $\Sigma\Delta$ 调制器）对1位流取平均值。DFSDM获取并处理1位数据流（数字滤波，取平均）。DFSDM以低于输入1位流的数据速率、更高的分辨率输出数据样本。DFSDM数字滤波器设置定义输出分辨率和数据速率。

1.2 $\Sigma\Delta$ 调制器

DFSDM外设需要一个外部模拟前端，用于执行模拟信号源的A/D转换。该外部模数转换在 $\Sigma\Delta$ 调制器中执行。

$\Sigma\Delta$ 调制器包含1位^(b)A/D转换器，后者将输入模拟数据进行数字化处理，转换为串行数字数据流。对模拟输入进行采样，并将其转换为具有交替的0和1的1位数字数据流。在给定的时间间隔期间计算的数字流的平均值，表示在相同时间间隔期间输入模拟信号的平均值。 $\Sigma\Delta$ 调制原理可以表示为特殊的PWM调制，其中，周期和占空比都将进行调制（但周期是固定的，按照典型的PWM调制方式对占空比进行调制）。参见图 2和图 3，了解PWM和 $\Sigma\Delta$ 调制之间的比较结果。

a. Arm是Arm Limited（或其子公司）在美国和/或其他地区的注册商标。

b. 通常情况下， $\Sigma\Delta$ 调制器的输出可以是多位的，但本文档仅关注1位A/D转换器（这是最常见的情况）。



接下来，STM32微控制器DFSDM外设对输出 $\Sigma\Delta$ 调制器的数字数据流进行处理。DFSDM使用需要根据应用要求而配置的参数，执行数字过滤。

注：

为了进行分析，数字流通常从二进制0和二进制1权重“转换”为+1和-1权重，用于与无任何DC分量的输入电压进行比较。零输入电压产生占空比50:50（使用一阶 $\Sigma\Delta$ 调制器）。

图2. PWM调制示例

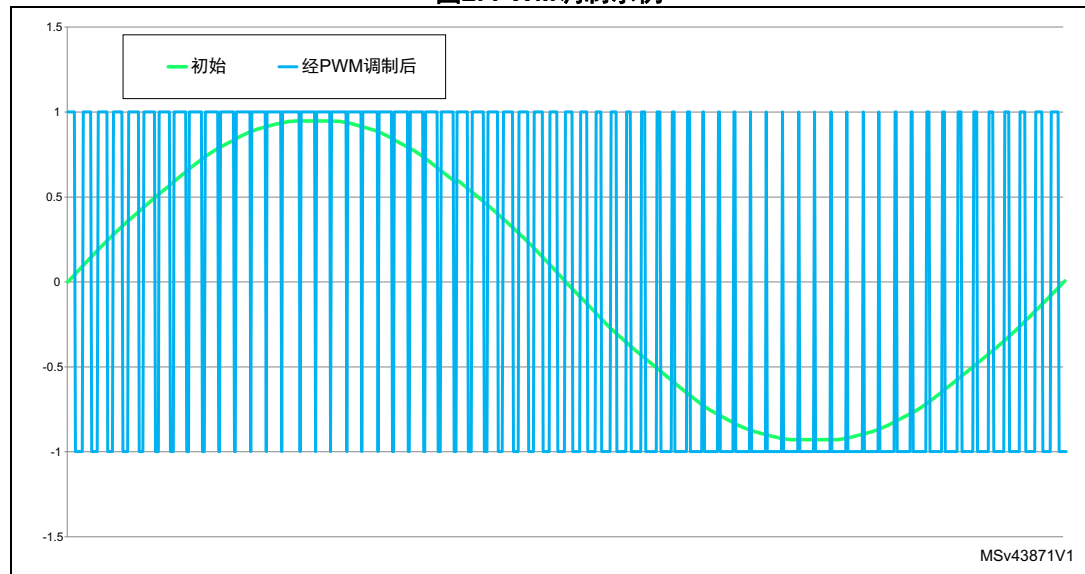
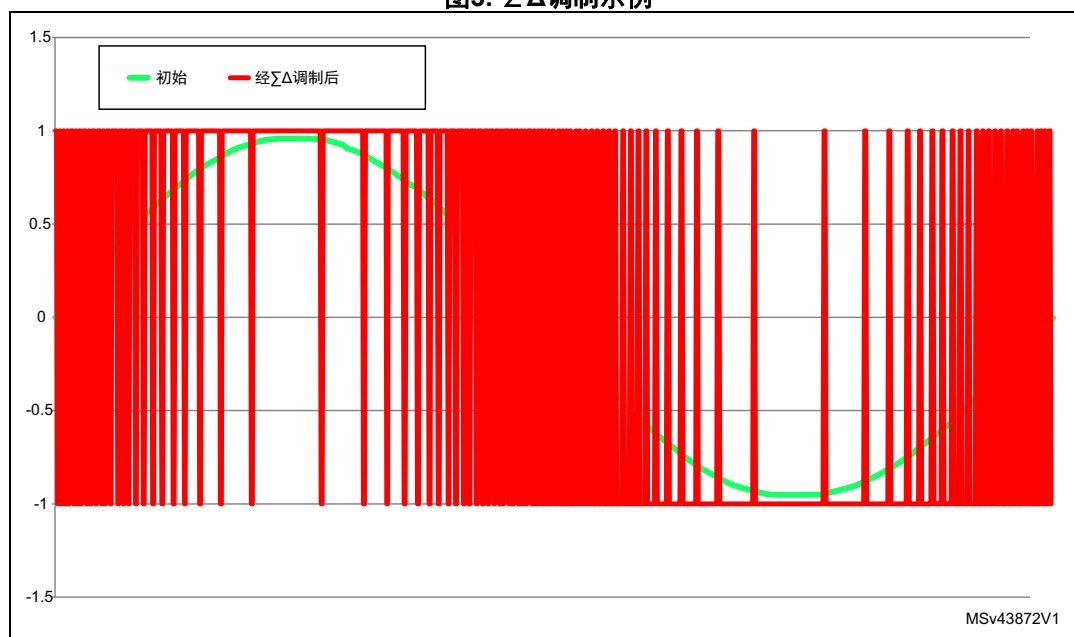


图3. $\Sigma\Delta$ 调制示例

1.3 数字滤波器

DFSDM外设（数字滤波器，用于 $\Sigma\Delta$ 调制器）处理A/D转换的数字部分。数字数据流由外部 $\Sigma\Delta$ 调制器提供。DFSDM的基本功能是实现数字滤波器。DFSDM处理流程包括对快速速率输入串行流取平均值，产生具有更高分辨率的并行、低速率数据输出。DFSDM嵌入式滤波器具有一组可配置参数，可以调整输出分辨率和数据速率，满足应用要求。

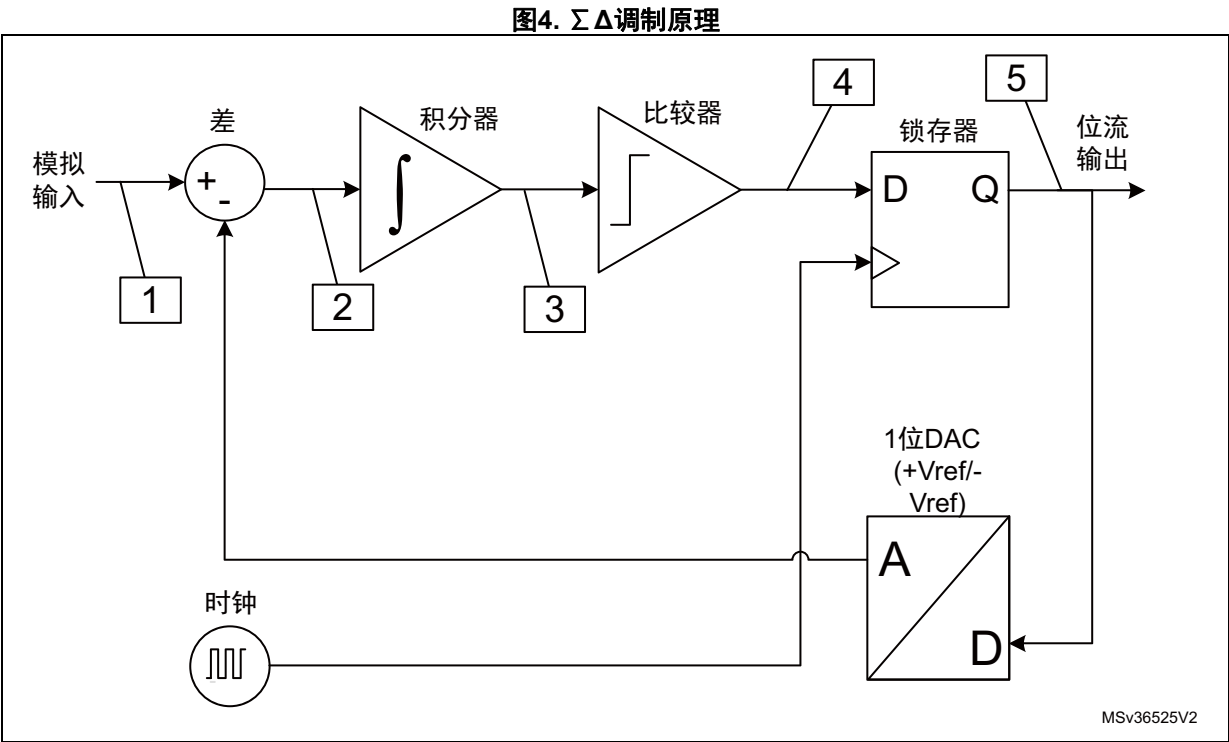
DFSDM具有其他与ADC相关的功能，包括：

- 每个通道上的独立快速看门狗，具有可编程速度和分辨率，可检测超过最小或最大允许电压电平的输入信号。
- 刹车信号生成，用于即时向其他外设（定时器）报告模拟看门狗或短路检测等事件。
- 每个通道上的短路检测器，可以非常快速地检测信号钳位：当输入电压达到其中一个模拟量程限制，并在超过给定的持续时间内保持稳定时（独立于主转换）。
- 极值检测器，用于记录最小和最大输入电压偏移。

2 ΣΔ调制原理（外部模拟前端机能，模拟）

2.1 ΣΔ调制原理

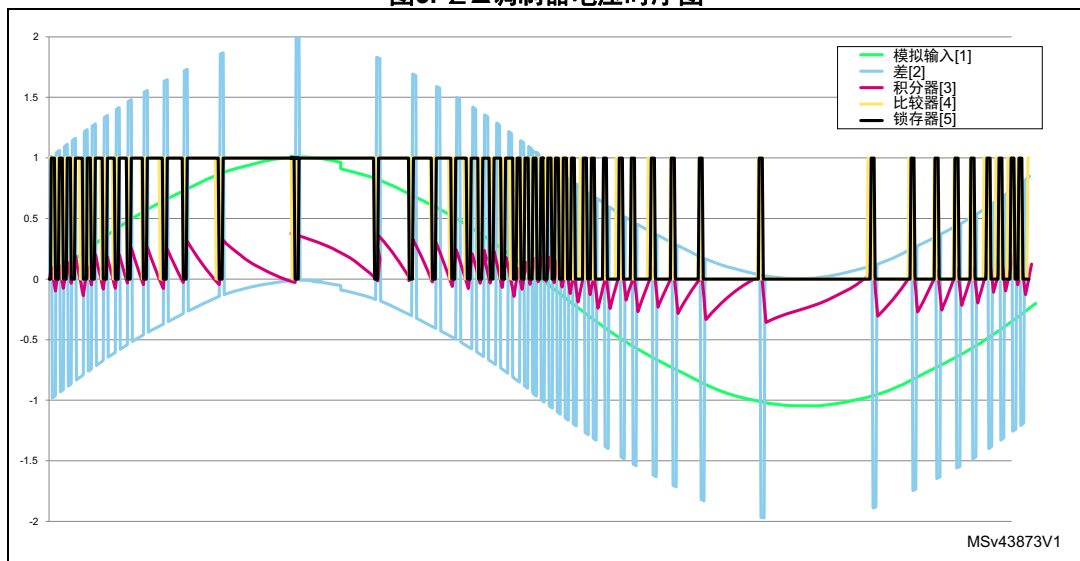
图 4 给出了ΣΔ调制器的基本功能框图。



1. 上图中的参考电压[1]至[5]用于以下段落。

图 5 是在模数转换的不同阶段的可用信号示例。

图5. ΣΔ调制器电压时序图



1. 上图中的参考电压[1]至[5]用于以下段落。

以下使用图 4 和图 5 所示的参考来描述ΣΔ调制：

模拟输入信号[1]被添加到比较器的1位DAC输出反馈（+Vref或-Vref电压），结果[2]进入积分器。积分器累计模拟输入信号[1]和1位DAC输出反馈（+Vref或-Vref电压）之间的差值。然后，使用比较器，对积分器输出[3]与零电压参考进行比较。按照时钟频率，使用D锁存器定期锁存比较器输出[4]，从而以量化的时步（时钟节拍）将比较器结果传导至调制器输出。D锁存器输出[5]即为ΣΔ调制器的数字1位输出。输出被馈送至1位D/A转换器，后者仅输出2个可能的模拟电压（通常用作+Vref和-Vref参考电压之间的开关）。1位输出数据流的数据速率由调制器时钟频率定义。

ΣΔ调制的输出是由调制器时钟定时的数字数据流（图 5 上的黑色曲线）。该输出的平均值（在数字域中计算）表示输入模拟电压。该数字平均值应被计算为：在ΣΔ输出流内，在给定数量的时钟周期期间中观察到的1的数量与0的数量之间的比率。

该数字数据流作为STM32微控制器DFSDM外设的输入，可在上述外设中对其进行滤波。需要根据应用要求设置DFSDM的可配置参数。

2.2 $\Sigma\Delta$ 调制的优点

2.2.1 噪声整形

必须对来自 $\Sigma\Delta$ 调制的输出信号进行滤波，去除高频内容（量化噪声），仅保留有用的频带。需要了解 $\Sigma\Delta$ 调制信号频谱才能正确设计此类滤波器。 $\Sigma\Delta$ 调制信号频谱不同于PWM调制频谱。

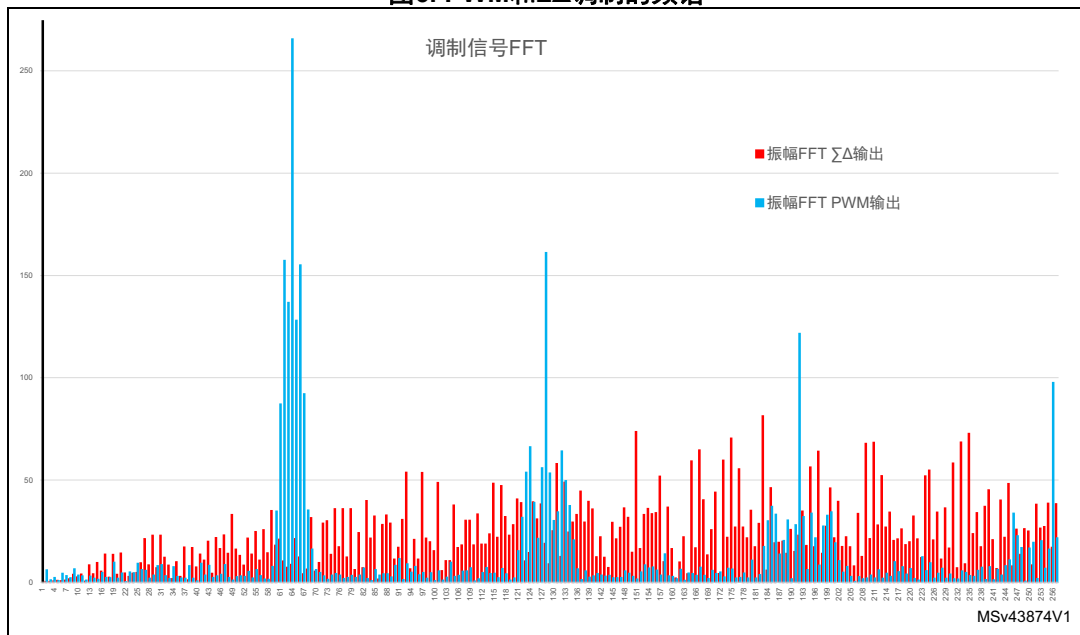
PWM调制信号的特征在于具有固定周期和可变占空比。由于具有固定的调制周期（或频率），PWM频谱显示出与调制基频及其谐波对应的典型能量峰值。模拟滤波（RC或LC滤波器）较难以去除这些谐波峰值。

$\Sigma\Delta$ 调制使用可变占空比和可变频率。如此一来， $\Sigma\Delta$ 频谱上的能量更均匀地扩散，而不会如同PWM（无固定的调制频率）一般集中出现在规则间隔的峰值上。此外，与PWM相比， $\Sigma\Delta$ 调制通常在高频时具有更高能量（因为调制频率更高）。模拟滤波（RC或LC滤波器）可以轻松去除 $\Sigma\Delta$ 调制的噪声内容。

[图 6](#) 介绍了PWM和 $\Sigma\Delta$ 调制的典型频谱（分别对应于[图 2：PWM调制示例](#)和[图 3： \$\Sigma\Delta\$ 调制示例](#)中介绍的信号）。PWM调制在较低频处产生较高的峰值，因此PWM处理需要更高阶的滤波器，以便适当地抑制与PWM调制基频相对应的第一能量峰值。在 $\Sigma\Delta$ 调制频谱中，在较低频率下，能量较低，因此滤波器的设计更简单。滤波能力必须适应 $\Sigma\Delta$ 调制器的阶数。例如，数字麦克风通常具有四阶 $\Sigma\Delta$ 调制器，其在有用频带中具有低量化噪声，但在频带外具有非常强的量化噪声。必须选择滤波器阶数，以便在不影响有用频带的情况下，抑制强频带外量化噪声。

$\Sigma\Delta$ 调制将量化噪声更均匀地传导至更高频率，这一效果被称为“噪声整形”。此特性可用于设计更简单的滤波器，形成较少噪声的基带信号。

图6. PWM和ΣΔ调制的频谱



2.2.2 A/D转换的线性

ΣΔ调制器的输出是1位串行数据流。此数据流的分辨率只有一位，通常不足用于此应用而言。提高信号分辨率的方法包括在给定的时段内，对1位流取平均值。平均的数据流具有更宽的分辨率（通常为16位），但采样率更低。取平均（滤波）操作基于数字域中的线性数学运算，因此不会由于滤波而增加非线性失真。

通常，具有并行数据输出的A/D转换器所使用的模拟元件数量多于ΣΔ调制器。例如，具有N位分辨率的SAR ADC类型在内部使用具有N个电阻器（或电容器）的R-2R电阻器网络（或C-2C电容器网络）。这些类型的ADC中使用的电阻（电容）必须具有精确的1:1或1:2电阻（电容）比。SAR ADC的线性取决于电阻器（电容器）的精度。在实际应用中，比率永远无法十全十美，不精确性是造成非线性、从而影响传输曲线的根本原因。

由于上述原因，使用ΣΔ调制的A/D转换的线性通常优于其他类型的A/D转换。ΣΔADC转换的非线性仅取决于ΣΔ调制器设计（参见 [图 4：ΣΔ调制原理](#)）。此外，非线性仅取决于输入电压对模拟元件固有特性的影响（积分器和开关中的电容器/电阻器，其中电容/电阻随输入电压变化），而不取决于不同组件标称值之间的比率。对于音频应用而言，线性更为重要，因为在这些应用中，非线性会导致信号失真（静态线性INL与动态线性THD关联）。

2.2.3 可扩展的ADC分辨率

ΣΔ转换的最终输出分辨率并非固定（如12位SAR ADC的示例所示）。ΣΔ调制器的输出分辨率是1位，通过跟随数字滤波（取平均值）至所需分辨率，即可增加分辨率。

此分辨率增加方法的缺点在于降低了输出数据速率。数据速率的降低情况是可预测的，必须针对给定的应用要求进行计算。例如，可以针对1 kHz输出数据速率，将数字滤波器配置为24位输出，或者，针对50 kHz输出数据速率，将数字滤波器配置为16位输出。

理论上，可以无限制增加分辨率，但在实际操作中，需要考虑噪声和转换路径中的组件所引起的误差（ΣΔ调制器设计、注入噪声.....）。另一个需要纳入考虑的因素是采样周期内的信号稳定性，因为信号稳定性会影响测量精度，特别是对于极低数据速率和高分辨率的应用。

2.3 ΣΔ调制的缺点

2.3.1 偏移和增益误差

来自ΣΔ调制器的1位数据流信号的平均值表示模拟输入信号的平均值。此数字输出平均值的精度（在[0..1]范围内）可能受到以下ΣΔ调制器组件的影响（参见 [图 4：ΣΔ调制原理](#)）：

- 积分器中的电阻器和电容器
- 1位DAC中的参考电压（+Vref/-Vref）
- 积分器的偏移

以下组件具有容差：电容器/电阻器值、偏移电压、+Vref和-Vref绝对电压之间的差值。

在理想情况下，1位数字输出平均具有的0和1的数量与零伏输入信号完全对应。由于组件容差（如上所述），对应于具有相同数量的0和1的输入模拟电压并非精确到零伏，存在偏移误差。偏移误差可以通过软件或硬件补偿（校准过程）。

在理想情况下，占空比为50:50的输出1位数字信号（0的数量等于1的数量）应与零输入模拟电压完全对应。由于组件容差（如上所述），对应于占空比为50:50的输入模拟电压并非精确到零伏，存在偏移误差。偏移误差可以通过软件或硬件补偿（校准过程）。

增益系数即输出数据和输入电压之间的比率，也受到ΣΔ组件容差的影响。理论增益和实际测量的转换增益之间的差值表示增益误差。它也可以通过校准（通常在软件中）进行补偿。

ΣΔ组件的部分特性还取决于温度，这反过来影响到偏移和增益误差。温度对积分电阻器和电容器的影响仅体现在增益误差上。偏移误差受温度变化的影响较小，因为这些参考值是对称的，因此影响+Vref和-Vref的误差可以进行自补偿。

2.3.2 低数据率

借助ΣΔ调制器来提高分辨率的方法是，增加1位数据流的平均值（更长的取平均时间或更高的滤波阶数）。因此，ΣΔ转换器主要用于较低数据速率的应用（通常用于音频范围和较低频率）。但在要求具有出色线性的特殊情况下，ΣΔ转换器可用于较高数据速率的应用。

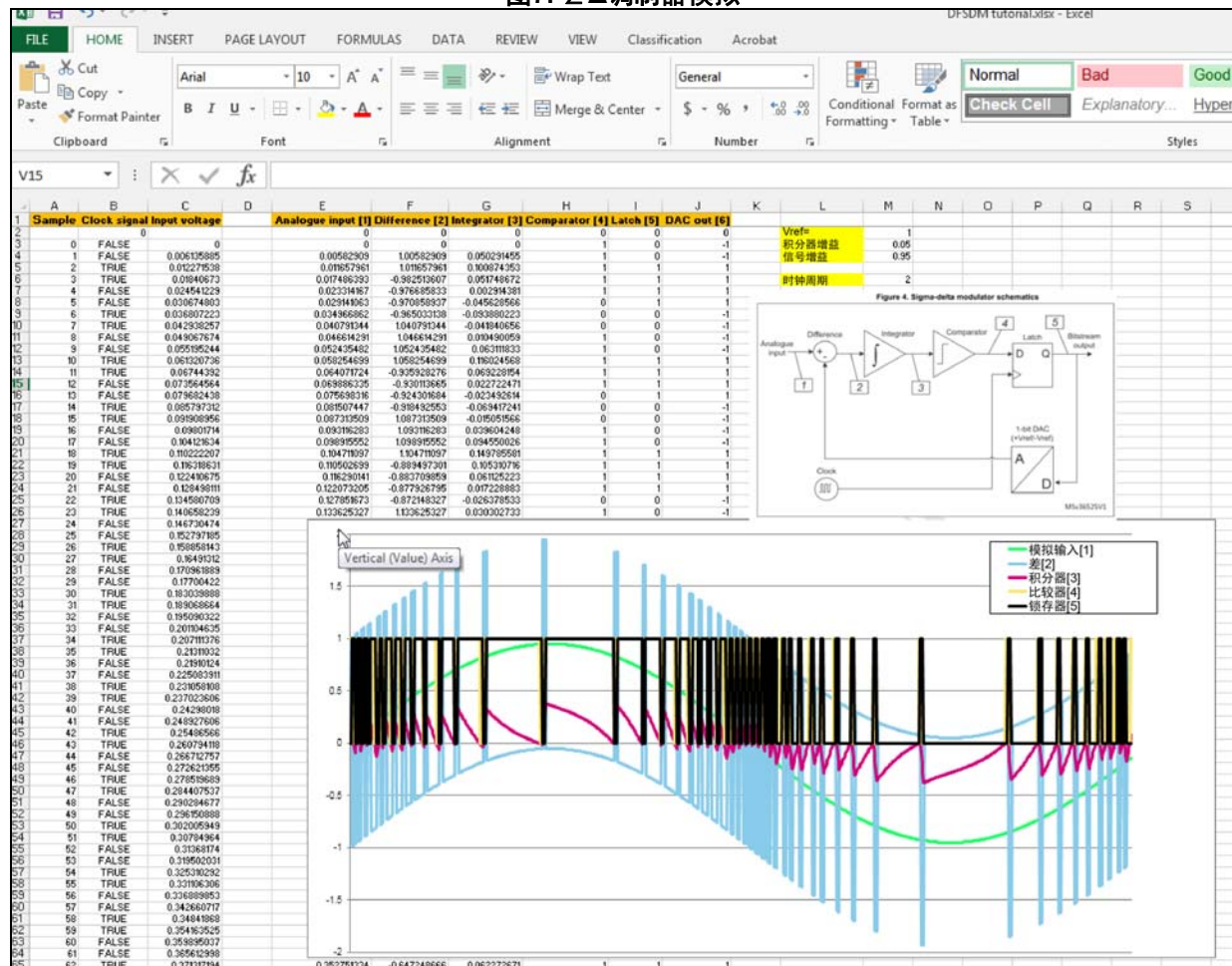
对于准静态应用（温度传感器），数据速率并不是约束条件，此时，由于ΣΔ调制器拥有可扩展的分辨率能力（低数据速率下具有高分辨率），通常会选择ΣΔ调制器。

2.4 ΣΔ调制器的模拟

2.4.1 使用[TUTORIAL]进行模拟

为了帮助理解ΣΔ调制，[TUTORIAL]展现了ΣΔ模型（如图4：ΣΔ调制原理所示）。在图4中，ΣΔ调制原理的每个参考电压信号提供时序图。用户可以更改部分参数和输入电压，查看其对ΣΔ调制器各个级的影响。在[TUTORIAL]中提供了一个模拟示例，示例基于正弦输入信号，如图7所示。

图7. ΣΔ调制器模拟



3 数字滤波 - 原理和设计

3.1 功能描述

数字滤波器对 $\Sigma\Delta$ 调制器产生的1位数据流执行滤波（取平均值）。滤波器输出是指具有更高分辨率（通常为12-24位）但数据速率降低（抽取）的数据字。数字滤波器的功能包括去除频带外频率分量（量化噪声、无用信号.....）和根据有用带宽降低数据速率（抽取）。

滤波器的设计显著影响到A/D转换的特性，是所需参数（滤波器的锐化度、滤波器调谐、最终分辨率.....）与硬件实现复杂性（导致成本）之间折衷的结果。我们的目标是在满足所需A/D特性的同时最小化滤波器设计的复杂性。

注： 与简单地对1位数据流取平均值相比，信号滤波通常需要更复杂的处理程序。

以下是设计滤波器时需要考虑的一些因素：

- 过滤器类型：

在各种类型的滤波器中，Sinc滤波器具备有趣的特性，完美结合了低价的硬件实现与可接受的性能水平。可以使用 $\text{sinc}(x)$ 函数（得名原因）对Sinc滤波器的频率响应进行建模。

Sinc滤波器是最常用的 $\Sigma\Delta$ A/D转换器实现类型。由于无需使用乘法器，因此价格便宜，且滤波器系数为整数。Sinc滤波器对1位样本执行简单的“滑动平均”计算。

- 滤波器长度（FOSR - 滤波器过采样率）：

较长的滤波器（对较多样本取平均值）会产生更高的分辨率，但会降低输出数据速率（抽取）。因此，滤波器长度是所需转换速率和最终数据分辨率之间的折衷结果。

- 滤波阶数（FORD）：

“滑动平均”计算可以多次应用于已经取得平均值的样本。滤波阶数定义“滑动平均”计算应用于相同输入样本的次数。高阶滤波器（通过添加更多的取平均值循环）产生更高的分辨率，但增加了延迟。

3.2 Sinc滤波器功能示例 - 提高分辨率

本节重点介绍多个“滑动平均”计算如何增加Sinc滤波器的分辨率。

此处详细展示了三阶滤波器（FORD = 3），其长度为FOSR = 10（在每个滤波器级的滑动平均）：

- 对输入流的3个周期进行观察： $3 \times \text{FOSR} = 30$ 个样本（参见图8和图9）。
- 除了每个取平均周期（FOSR = 10）的“1”脉冲外，输入的1位数据流几乎总是“0”（参见图8和图9的“输入”曲线）。测试两个输入流：一个输入流具有完全等距脉冲（在水平轴的位置10、20和30，参见图8）；在另一个输入流中，可通过一个周期来模拟稍高的脉冲密度，从而预测第二个脉冲（在水平轴的位置10、19和29，参见图9）。

- 对一阶滤波器级执行滑动平均（参见“一阶”曲线），在每个FOSR周期进行最终结果采样，从而得到一阶滤波器的结果（在第10、20和30个周期）。在上述两个输入流（图 8和图 9）的情况下，由于对一阶滤波器级执行简单滑动平均法（每个周期只有一个'1'），最终结果始终为'1'。
- 二阶滤波器级对来自一阶滤波器级的样本执行滑动平均（参见“二阶/10”曲线）。在每个FOSR周期进行最终结果采样，得到二阶滤波器结果（在第20、30个周期）。两个输出流（图 8和图 9）的最终结果之间存在明显差异：由于'1'的密度较高（图 9具有较高密度的输出流，它产生的值高于图 8低密度流的值）。
- 三阶滤波器级对二阶滤波器级的样本执行滑动平均（参见“三阶/100”曲线）。在每个FOSR周期进行最终结果采样，得到三阶滤波器的结果（在第30个周期）。低密度流（图 8）和高密度流（图 9）的最终结果之间存在显著差异。由于执行了滑动平均，第三个输出更平滑，更精确。
- 滤波器输出端的信号动态范围为FOSR^{FORD}，但需要FOSR^{FORD}个样本才能获得第一个结果，因为每个滤波器级必须填充前一个滤波器级的有效样本（参见图 8和图 9）。总之，高阶滤波器能够在给定的流持续时间内提供更高的分辨率，或者在较短的流持续时间内提供相同的分辨率。其缺点在于硬件设计更复杂，滤波器初始化时间更长。

图8. 每个滤波长度有一个1的三阶滤波器输出示例

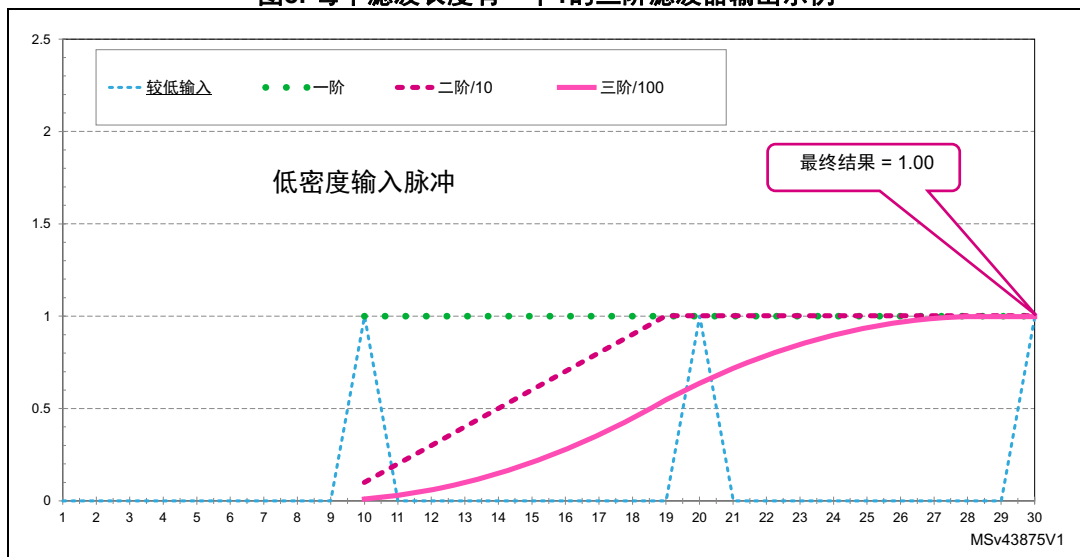
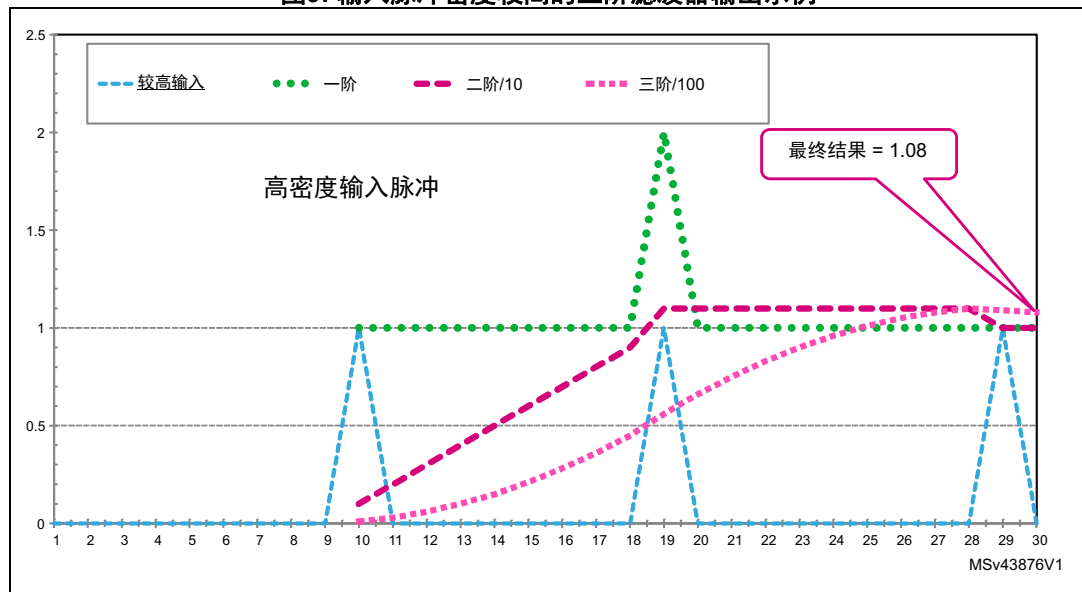


图9. 输入脉冲密度较高的三阶滤波器输出示例



3.3 Sinc滤波器的硬件设计

Sinc传递函数能够表示Sinc滤波器特征，以下提供其在数字域中的公式。可以理解为：输入1位数据的多个滑动平均值。本节的其余部分重点介绍如何实施简化，以实现高效且极其简单的硬件实现。

公式1：

$$y(n) = x(n) + x(n-1) + x(n-2) + \dots + x(n - (\text{FOSR} - 1))$$

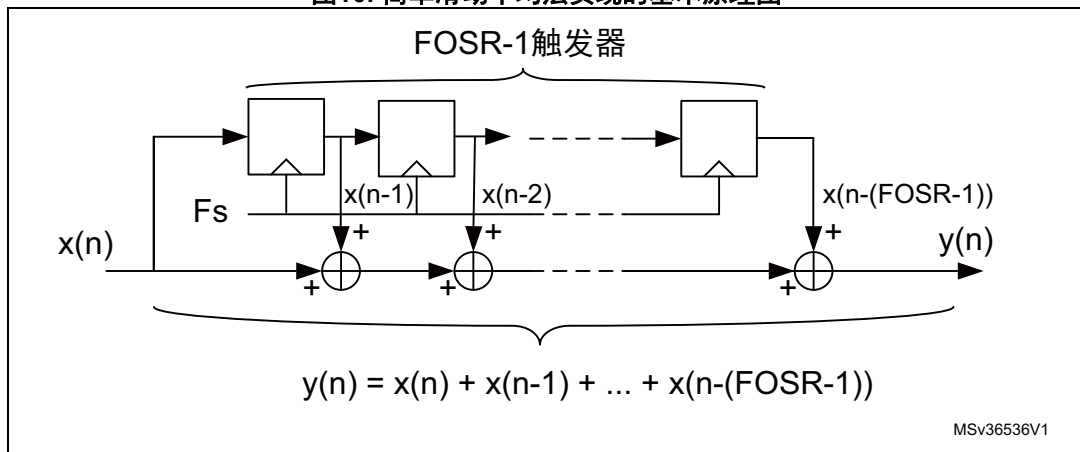
其中：

$X(n)$ 是输入的第 n 个样本

$y(n)$ 是输出的第 n 个样本

图 10将上述公式直接转换为硬件实现：

图10. 简单滑动平均法实现的基本原理图



上述硬件实现需要 (FOSR-1) 个加法器和 (FOSR-1) 个触发器。可以使用下文介绍的部分简化方法。

可以使用先前的输出结果 $y(n-1)$ 来简化 [公式1](#)：

公式2：

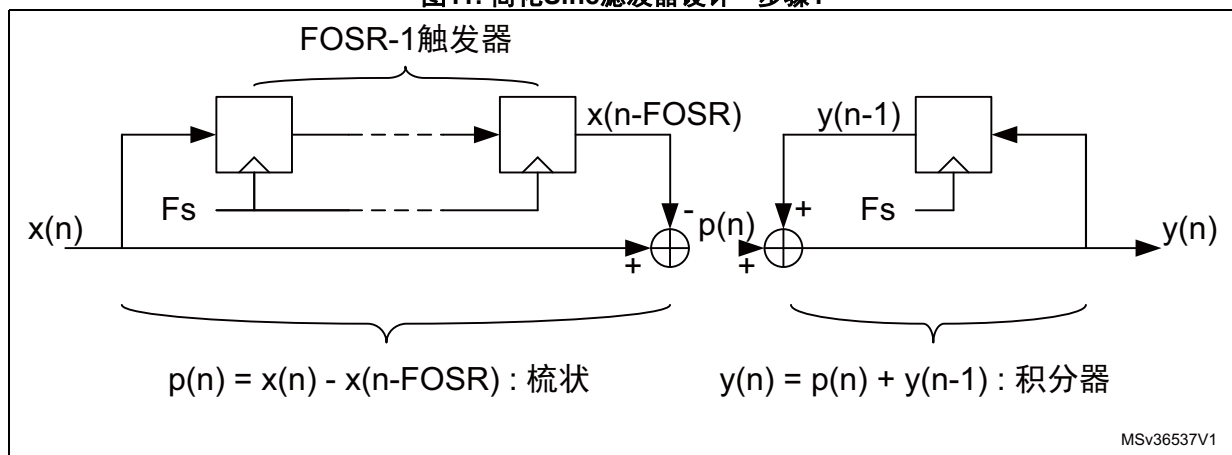
$$y(n) = x(n) + y(n-1) - x(n-FOSR)$$

公式3：

$$y(n) = x(n) - x(n-FOSR) + y(n-1)$$

[图 11](#) 是简化的原理图，只需2个加法器和FOSR + 1个触发器。现在可以将原理图分为两部分：梳状级和积分器级。

图11. 简化Sinc滤波器设计 - 步骤1



根据这些对应关系，将公式转换为Z域（频域），可以进一步简化：

- Z域中的 $X(z)$ 对应于离散时域中的 $x(n)$ ，其中， n 是采样时钟周期的整数值（采样时钟周期= $1/F_s$ ； F_s = 采样频率）
- $z^{-N} \times X(z)$ 对应于 $x(n-N)$ ，即，延迟 N 个采样周期的 $x(n)$ 。

用于Z域的一阶Sinc滤波器公式变为：

$$Y(z) = X(z) - z^{-FOSR} \times X(z) + z^{-1} \times Y(z)$$

对应的传递函数 $H(z)$ 是:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1 - z^{-FOSR}}{1 - z^{-1}}$$

在图 11 中，我们通过分解的方式介绍了 Sinc 滤波器的 2 个级：“梳状级”和“积分器级”。此类滤波器的传递函数（称为 CIC：级联积分器-梳状滤波器）由下式给出：

Z域中的梳状传递函数:

$$H_C(z) = \frac{P(z)}{X(z)} = 1 - z^{-FOSR}$$

积分器传递函数:

$$H_I(z) = \frac{Y(z)}{P(z)} = \frac{1}{1 - z^{-1}}$$

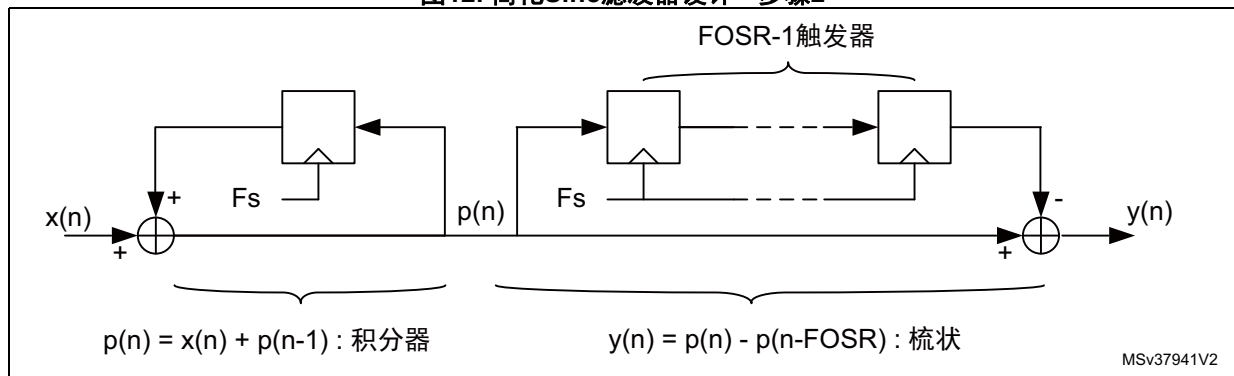
整体传递函数相当于Sinc滤波器传递函数：

$$H_C(z) \cdot H_I(z) = \frac{P(z)}{X(z)} \cdot \frac{Y(z)}{P(z)} = (1 - z^{-FOSR}) \cdot \left(\frac{1}{1 - z^{-1}} \right) = \frac{1 - z^{-FOSR}}{1 - z^{-1}} = H(z)$$

另一个简化滤波器的示例包括置换梳状和积分器运算，如 [图 12](#) 所示：

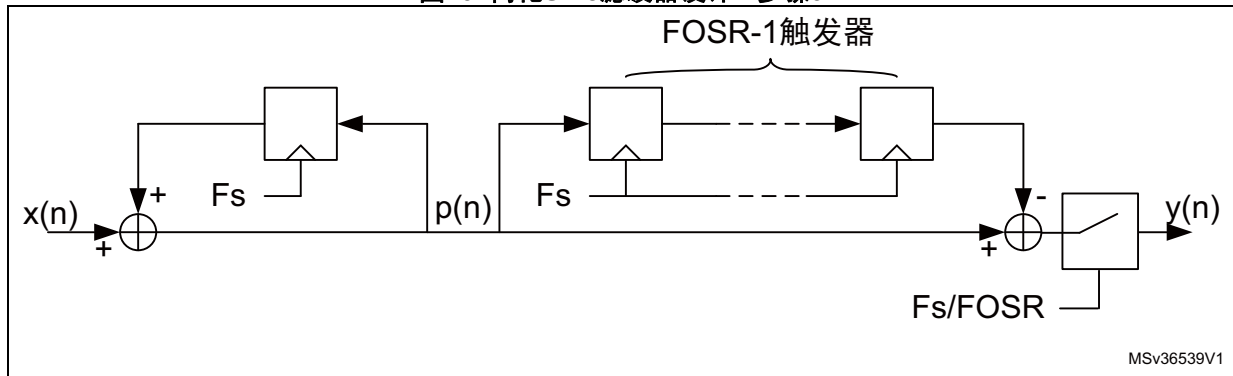
$$H(z) = H_I(z) \cdot H_C(z) = \left(\frac{1}{1 - z^{-1}} \right) \cdot (1 - z^{-FOSR})$$

图12. 简化Sinc滤波器设计 - 步骤2



由于滤波限制了输出信号带宽，因此可以按FOSR因数，对输出 $y(n)$ 进行下采样。在每个FOSR时钟周期中取一个 $y(n)$ 样本，即可实现上述下采样（参见图13）。

图13. 简化Sinc滤波器设计 - 步骤3



下采样时钟周期的定义如下：

$$T = \frac{\text{FOSR}}{F_s} = \text{FOSR} \cdot \tau \quad \text{where: } \tau = \frac{T}{\text{FOSR}}$$

下采样梳状部分的公式可以重写如下：

$$y(n \cdot \tau) = p(n \cdot \tau) - p(n \cdot \tau - \text{FOSR} \cdot \tau)$$

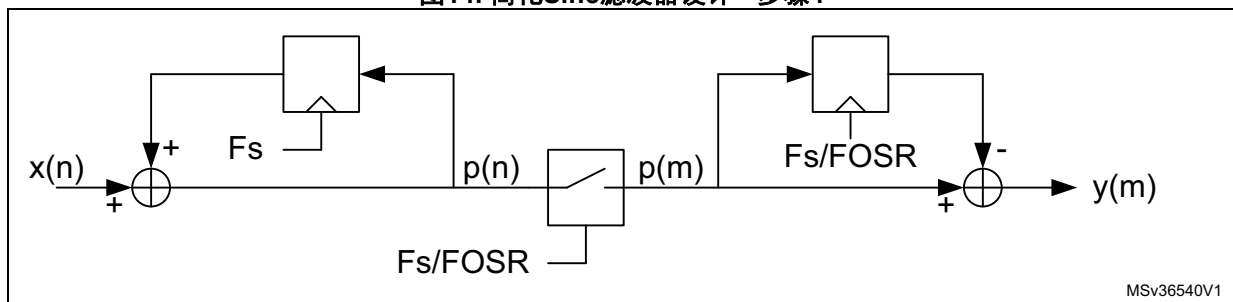
$$y\left(\frac{n \cdot T}{\text{FOSR}}\right) = p\left(\frac{n \cdot T}{\text{FOSR}}\right) - p\left(\frac{n \cdot T}{\text{FOSR}} - \frac{\text{FOSR} \cdot T}{\text{FOSR}}\right)$$

$$y(m) = p(m) - p(m-1) \quad \text{where: } m = \frac{n}{\text{FOSR}}$$

此运算相当于按FOSR因数对p(n)进行下采样（随后是一阶微分器）。以高频采样的触发器系列可以替换为具有下采样频率的单个触发器。最终原理图如[图 14](#)所示。

下采样的影响在于取消了滑动平均操作（因为输出数据速率降低）。

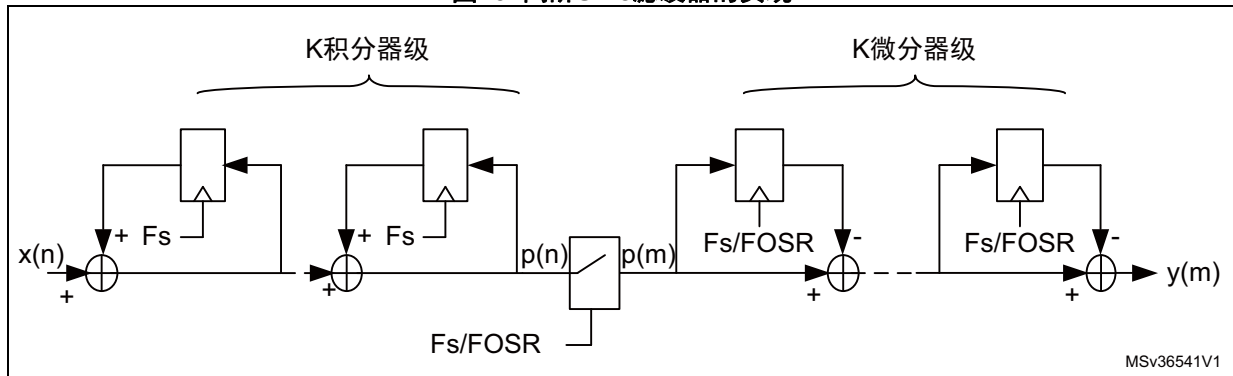
图14. 简化Sinc滤波器设计 - 步骤4



最终原理图（[图 14](#)）减少为2个触发器和2个加法器（均为FOSR位宽）。

高阶Sinc滤波器是一阶滤波器级的串联级联，其中，梳状级和积分级可以组合在一起（[图 15](#)）。

图15. 高阶Sinc滤波器的实现



高阶Sinc滤波器的传递函数（滤波器阶数= FORD）由下式提供：

- 对FORD一阶滤波器进行级联：

$$H(z) = \left[\left(\frac{1}{1-z^{-1}} \right) \cdot (1-z^{-FOSR}) \right] \cdot \left[\left(\frac{1}{1-z^{-1}} \right) \cdot (1-z^{-FOSR}) \right] \dots$$

- 重新排列阶数：

$$H(z) = \left[\left(\frac{1}{1-z^{-1}} \right) \cdot \left(\frac{1}{1-z^{-1}} \right) \dots \right] \cdot [(1-z^{-FOSR}) \cdot (1-z^{-FOSR}) \dots]$$

- 最终公式：

$$H(z) = \frac{Y(z)}{X(z)} = \left(\frac{1-z^{-FOSR}}{1-z^{-1}} \right)^{FORD}$$

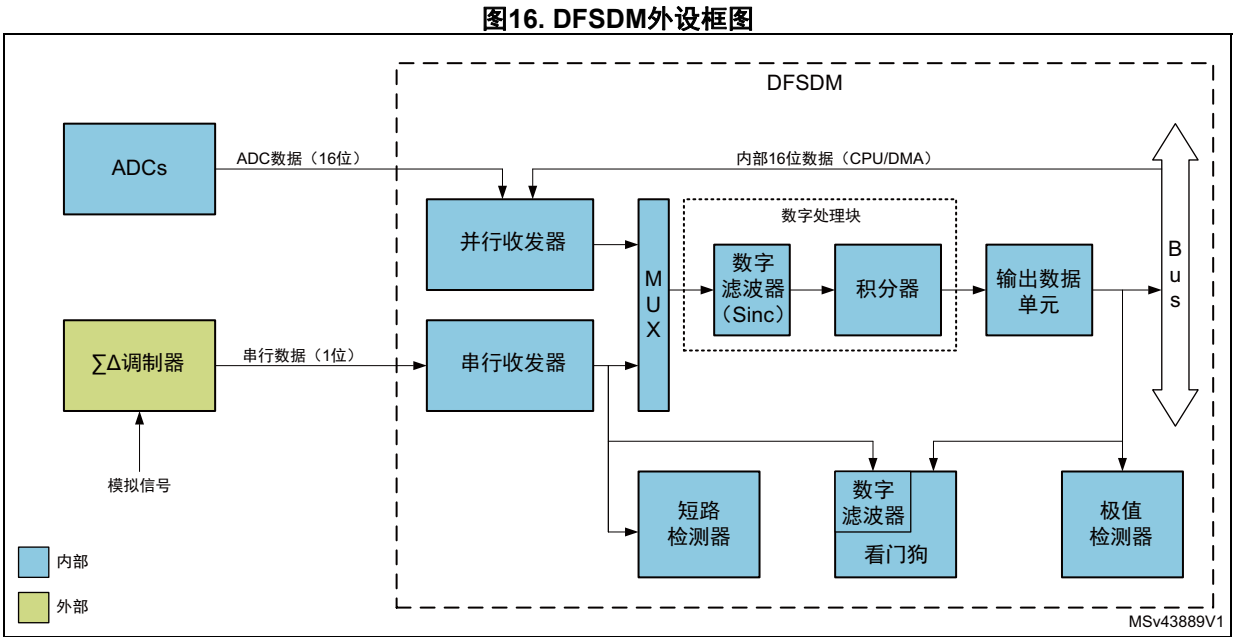
在最终积分器级的输出端执行下采样，允许在下采样速率下，以统一的延迟微分器替换每个梳状级（参见图 15）。

注： Sinc滤波器实现可以扩展到多位宽的输入信号（例如，用于并行数据输入，而非串行数据输入）。此时，应扩展触发器和加法器的位宽。

4 DFSDM外设操作

4.1 框图

图 16 是DFSDM外设以及所有内部功能块及其内部和外部连接的框图。DFSDM不能简化为仅仅一个数字滤波器，而是一个完整的数字外设，可处理整个A/D转换过程（当与外部 $\Sigma\Delta$ 调制器相关联时）。



4.2 DFSDM组件

4.2.1 串行收发器

“串行收发器”模块接收来自外部 $\Sigma\Delta$ 调制器的串行数据。它采用SPI和曼彻斯特串行协议格式，同时具有可配置的上升/下降采样时钟沿，可支持大多数 $\Sigma\Delta$ 调制器类型。它还支持数字麦克风使用的PDM信号格式请参见图 28: MEMS麦克风连接到DFSDM（支持立体声）。

一个DFSDM_DATINy引脚最多可连接两个数字麦克风（配置为立体声麦克风）。在此情况下，每个麦克风配置为对不同的采样时钟沿敏感（来自两个麦克风的信号出现在一条DFSDM_DATIN数据线上）。为了将此复合信号传输到两个不同的通道，两个通道必须从与两个麦克风物理连接的引脚（DFSDM_DATINy、DFSDM_CKINy）上获取串行输入（参见图 28: MEMS麦克风连接到DFSDM（支持立体声））。接下来，配置每个通道，使其在不同的采样沿上进行数据采样。

外部时钟信号可以连接到DFSDM_CKINy引脚（假设 $\Sigma\Delta$ 调制器提供此时钟信号），或者可以从内部时钟生成器获取时钟信号（假设 $\Sigma\Delta$ 调制器需要外部时钟信号）。内部时钟生成器通过DFSDM_CKOUT引脚驱动 $\Sigma\Delta$ 调制器。曼彻斯特格式协议不需要外部时钟信号，因为该协议是单线（在DFSDM_DATINy引脚上），时钟信号可根据曼彻斯特编码流重建。在丢失时钟（外部硬件故障）时，时钟信号存在检测器可用于触发中断。

串行收发器还提供“脉冲跳跃”。脉冲跳跃能够在给定数量的采样时钟脉冲期间暂停串行收发器的输出（丢弃1位数据样本的给定计数）。在实际应用中，“脉冲跳跃”用于波束整形应用，相对于一个通道数据延迟另一个通道数据。波束整形技术使用检测器阵列（麦克风），从首选方向感测信号（声音）。相对于先前的麦克风信号，向每个麦克风信号注入延迟。延迟定义了首选的感测角度。

4.2.2 并行收发器

并行收发器是一个内部16位并行输入寄存器，可通过APB总线由CPU、DMA访问，或者直接从内部ADC访问。其功能在于对内部信号进行后期处理（滤波）。使用示例：

- 对内部ADC捕获的数据进行滤波
- 对SPI外设捕获的数据进行滤波（通过存储器缓冲区）
- 对存储器缓冲区中保存的任何16位数据进行滤波（DMA传输到DFSDM并行收发器）。

4.2.3 数字滤波器

数字滤波器是处理输入数据的关键组件。可配置的数字滤波器用于支持最终应用需求（速度、分辨率）。可以配置下列参数：

- 滤波阶数：Sinc1 ... Sinc5, FastSinc
- 过采样率：
 - FOSR = 1 ... 1024（对于Sinc1 ... Sinc3, FastSinc）
 - FOSR = 1 ... 215（对于Sinc4）
 - FOSR = 1 ... 73（对于Sinc5）

上述FOSR范围考虑了滤波阶数和滤波器内部分辨率（32位宽），以避免溢出（在1位输入信号的情况下；对于多位并行输入信号，范围减小）。

4.2.4 积分器

可选的积分器可用作一个简单的加法器。它对滤波器输出提供的给定数量的样本进行求和。经过求和，得出单个积分器输出的滤波器输出样本数，这个数量的可配置范围为IOSR = 1 ... 256。积分器的工作方式类似于Sinc1滤波器（对给定采样数取平均值）。

4.2.5 输出数据单元

输出数据单元执行最终校正，这包括向右移位，以及对来自积分器的数据执行偏移校正。

向右移位用于：

- 将32位内部输出适配到最终的24位寄存器中
- 进一步限制最终分辨率（例如，在音频数据的情况下为16位）

偏移校正可校准外部 $\Sigma\Delta$ 调制器的偏移误差。用户使用带符号的24位校正来配置偏移寄存器，后者自动添加到输出结果中。偏移校正通常是嵌入微控制器软件中的校准程序的结果，该程序执行偏移校准计算并将校正存储到偏移寄存器中。

DFSDM外设中的所有操作都采用带符号格式（滤波、积分、偏移校正、右移位）。

4.2.6 模拟看门狗

模拟看门狗功能与ADC外设相同，其作用在于，当信号超过预定义阈值时，触发微控制器CPU（中断）或其他外设（刹车信号），从而控制ADC数据偏移保持在给定限制内。

不仅可以监控DFSDM输出数据，还可以监控来自 $\Sigma\Delta$ 调制器（串行收发器输出）的原始数据。

可以通过专用的可配置数字滤波器（FOSR = 1 ... 32，FORD = 1 ... 3），直接从串行收发器监控串行数据。如此一来，无论主数据转换速度如何，用户都能够在监控速度和监控分辨率之间取得最佳折衷结果。部分应用要求超过阈值的输入信号的反应时间要快于主转换速度。对主转换本身阈值的监控通常更为精确，但速度更慢。

4.2.7 短路检测器

短路检测器设计用于在信号超出或低于最大允许范围时，快速检测突发模拟输入信号的饱和情况。在正常情况下，模拟输入信号增益是受控的（电流或电压感应回路），永远不会达到此饱和水平。在某些极端情况下（如短路），感测到的信号测量值可能超出工作范围限制。此时，反应时间必须尽可能快（例如，关闭电源）。

短路检测器可实现此类非常快速的检测。信号饱和的检测基于对串行收发器的1位数据流的分析结果。典型的 $\Sigma\Delta$ 调制器输出1位信号，在“0”和“1”之间频繁转换（以再现模拟信号波动）。当发生输入信号饱和时，来自 $\Sigma\Delta$ 调制器的输出要么是长串逻辑'0'（低于负阈值的信号），要么是长串逻辑'1'（高于正阈值的信号）。如果此情况超过一定的持续时间，则短路检测器触发“短路”事件。可以在DFSDM短路检测器阈值设置中（SCDT）设置触发检测的持续时间阈值，范围为1到256个输入样本计数。例如，如果SCDT = 100，则一旦在输入的1位数据流中检测到至少100个连续的“0”或“1”，便会触发短路检测器事件。短路检测器可以触发中断（软件干预）或激活刹车信号（快速硬件干预）。

此类短路检测速度快于基于模拟看门狗的检测，因为后者涉及在较长串的1位输入样本中执行数字滤波。

4.2.8 极值检测器

极值检测器简单地分析最终输出数据样本，将最小值和最大值存储到极值寄存器中。通过读取这些极值寄存器，可以在一段时间内监控转换信号的最大和最小电平。这些电平在软件后期处理中十分有用，例如，可用于信号归一化或用于自动增益控制。每一次读取极值寄存器时，都会使用复位值对其进行重新初始化，并重新启动极值检测。

4.3 DFSDM模拟

在[TUTORIAL]中，针对 $\Sigma\Delta$ 调制器和DFSDM内部块的行为进行建模。可以模拟输入信号的发生率和DFSDM参数的影响，还可以观察和分析内部信号与输出信号。

提出的模拟包括：

- $\Sigma\Delta$ 调制器（工作原理演示）
- Sinc滤波器+积分器（数字滤波器功能演示）
- Sinc滤波器的频率特性（LP滤波器形状演示）
- PWM和 $\Sigma\Delta$ 信号的FFT（噪声整形演示）
- 高阶滤波器工作（演示如何通过多个信号平均值提高分辨率）
- $\Sigma\Delta$ 调制器（DAC转换器）

4.3.1 $\Sigma\Delta$ 调制器原理

该模拟请参见[TUTORIAL]的第一张工作表。

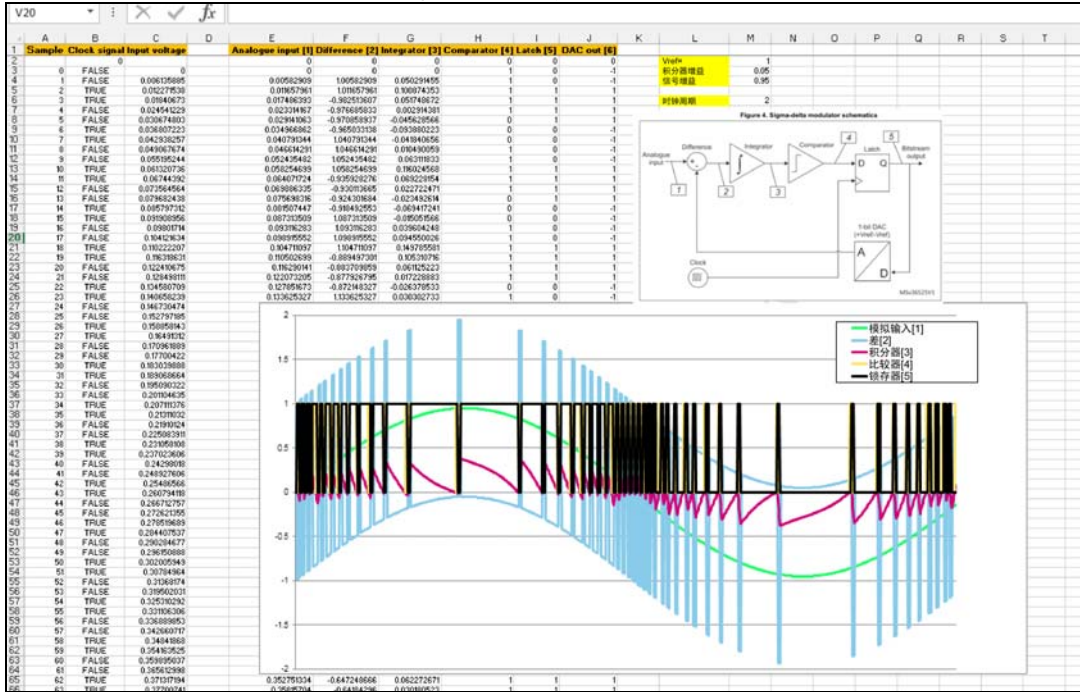
一阶 $\Sigma\Delta$ 原理的模拟基于图 4： $\Sigma\Delta$ 调制原理提供的原理图，其中，每个模块都已建模。调制器的输入信号对应于正弦波信号的一个周期。每个观察点（如图 4所述）对应于图 17所述图表的曲线。

用户可以修改输入信号形状或修改调制器的一些参数（积分器增益、 V_{ref} 电平），并显示 $\Sigma\Delta$ 调制器的数字输出的影响。

该模拟说明了 $\Sigma\Delta$ 调制器的原理，可以在处理的不同阶段实现内部信号的可视化。

此处，将 $\Sigma\Delta$ 调制器建模为一阶，便于解释基本原理。在实际应用中，除了通常使用四阶数字麦克风之外，大多数 $\Sigma\Delta$ 调制器使用二阶。

图17. $\Sigma\Delta$ 调制器模拟



4.3.2 DFSDM滤波模拟（滤波器和积分器）

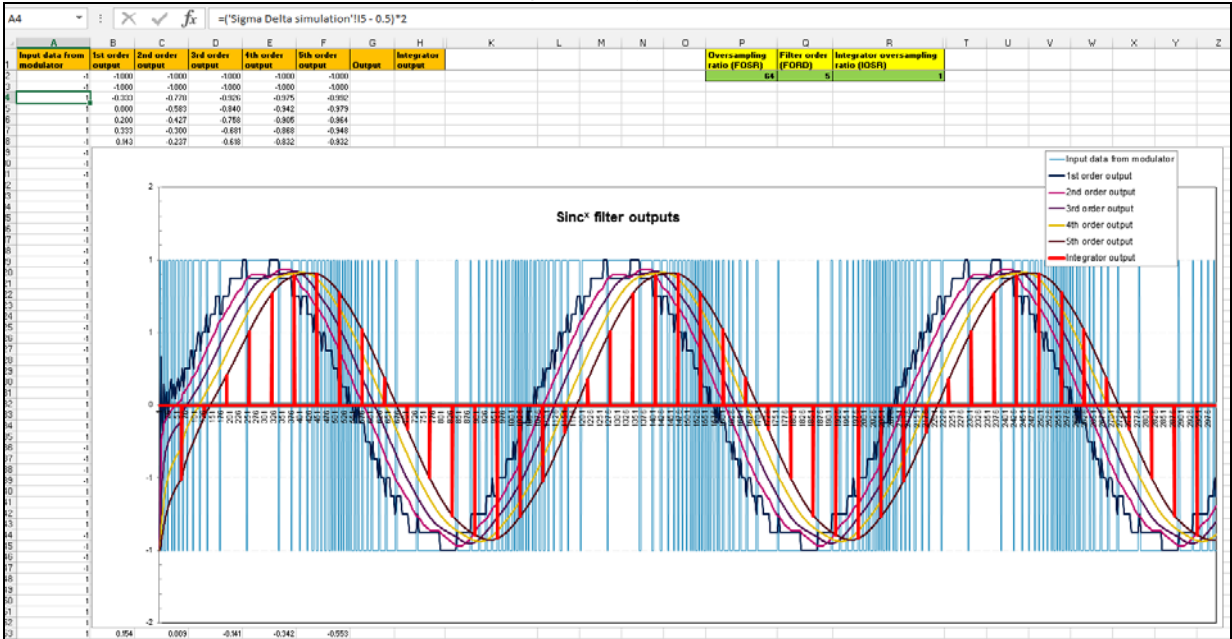
该模拟请参见[TUTORIAL]的第二张工作表。DFSDM模拟器的这一部分对应于Sinc滤波器和积分器级（详细信息，请参见第 4.2.3 节：数字滤波器和第 4.2.4 节：积分器）。

Sinc滤波器和积分器模型均在[TUTORIAL]内构建，且可以模拟。用户可以更改滤波器的FORD和FOSR参数，以及积分器的IOSR参数。模拟结果与实际DFSDM块获得的结果相同。用户可以根据其应用，在模拟中调整滤波器和积分器参数，无需等待实际的原型即可观察对输出信号形状的影响。

该模拟需要来自 $\Sigma\Delta$ 调制器的数字信号输入（1位数据流）（如同在实际应用中一样）。在这个特定的模拟中，由[TUTORIAL]中的一阶 $\Sigma\Delta$ 调制器模拟结果提供输入（参见第 4.3.1 节： $\Sigma\Delta$ 调制器原理）。如此一来，可以将输出数字信号（DFSDM滤波模拟曲线）与施加到 $\Sigma\Delta$ 调制器的输入模拟信号进行比较。接下来，应调整滤波器参数，以便确保以下方面达到应用要求：进入 $\Sigma\Delta$ 调制器的模拟输入信号与来自DFSDM的输出数字化信号（最终输出样本）之间的可接受误差。

图 18 不仅提供输出曲线（FORD=5），还提供每个滤波阶数输出端的内部数据结果。它显示了通过执行更多滑动平均循环获得的分辨率增益。

图18. 滤波模拟



4.3.3 Sinc滤波器的频率特性

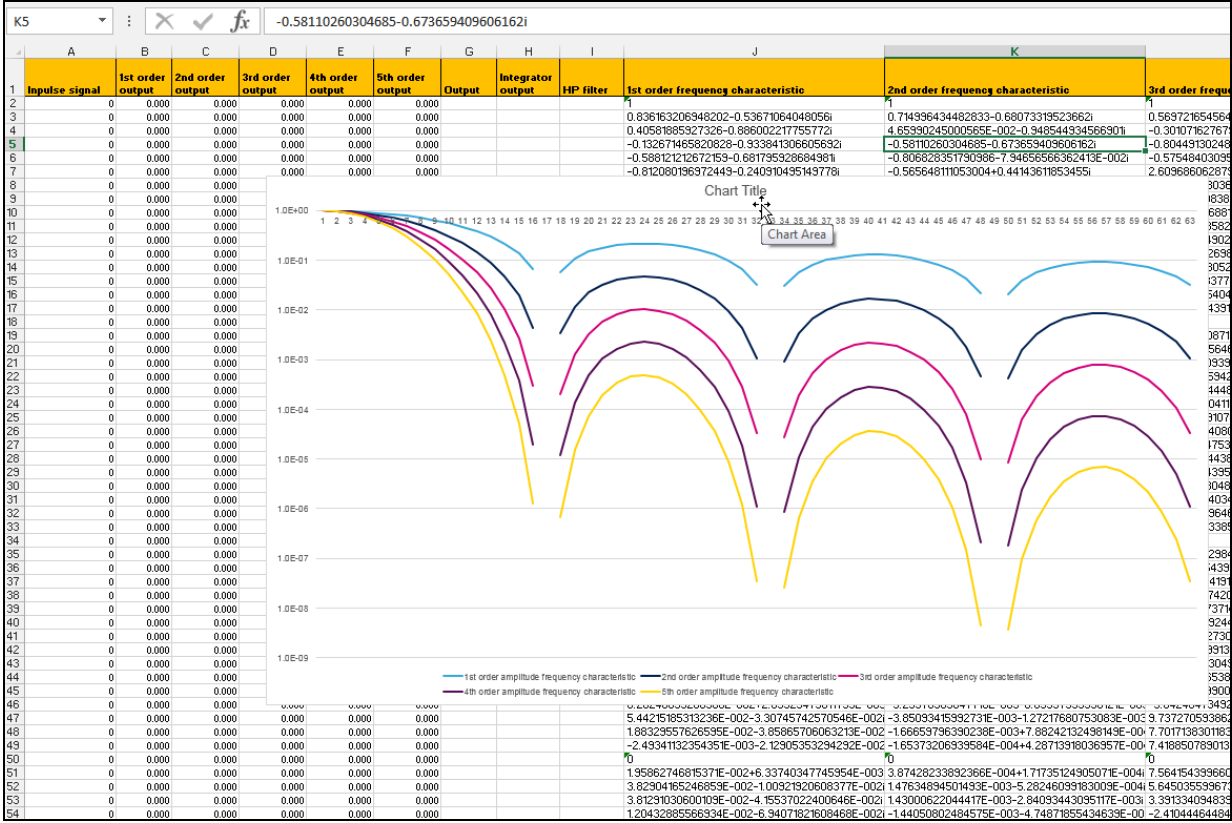
该模拟请参见[TUTORIAL]的第三张工作表。可根据滤波阶数可视化Sinc滤波器的频率特性。

可以在滤波器的输入端施加脉冲信号，并计算滤波器输出端的脉冲响应的FFT变换，从而确定滤波器的频率特性。以上便是在模拟模型中计算滤波器特性的方法。该模型计算滤波器的脉冲响应，并将结果提交给512点FFT变换。使用不同滤波阶数的结果请参见[TUTORIAL]和图 19。

梳状频率响应具有周期性衰减点（陷波 - 也称为“零”），在较高频率下存在降低趋势（低通滤波）。第一个陷波的频率取决于所选的FOSR，并由 $f_{\text{sampling}}/\text{FOSR}$ 给出。在连续采样的情况下， $f_{\text{sampling}}/\text{FOSR}$ 也是输出数据速率频率。与陷波相对应的输入信号的频率分量被滤波器完全抑制。在部分应用中，可借助该属性，按照预定的频率，从输入信号中去除外部噪声。例如，有时需要使用长导线将远距离传感器连接到微控制器，这种长导线易于受到电网中收集的噪声（50/60Hz）的影响，此时便可根据上述属性进行滤波。

另一个特性在于，在较高频率下的衰减（陷波除外，陷波的频率仅取决于FOSR）与滤波阶数（FORD）成比例。因此，滤波阶数越高，在较高频率下的抑制情况更明显。在感应准静态信号（如温度或压力传感器）的应用中，建议使用高阶滤波器（FORD）以及高过采样率（FORD）来抑制AC扰动产生的噪声。可以使用位于滤波器之后的积分器来完成对准静态信号的进一步滤波。增加积分器过采样率（IOSR）可以对信号执行额外的取平均值操作。在实际应用中，用户应首先设置（高）FORD，随后正确设置FOSR和IOSR，从而抑制来自主电源频率（50/60Hz）的噪声。由[Sinc滤波器+积分器]组成的整个模块的第一个陷波频率为： $f_{\text{sampling}} / (\text{FOSR} * \text{IOSR})$ 。

图19. 滤波器频率特性



4.3.4 $\Sigma\Delta$ 调制的噪声整形

该模拟请参见[TUTORIAL]的第四和第五张工作表。它对比PWM调制信号的频谱与 $\Sigma\Delta$ 调制信号的频谱，突出了后者在噪声整形方面的优势。在进行FFT变换之前，对相同的输入信号（正弦波的一个周期，参见图 20）执行 $\Sigma\Delta$ 调制或PWM调制。图 21比较了PWM调制信号和 $\Sigma\Delta$ 调制信号的频谱。X轴表示频率，是基本信号频率（信号基频及其谐波）的倍数，Y轴表示频谱密度。逻辑上，指数1（信号基频）的频谱幅度非常大（~470）。为了重建原始信号，必须滤除位于指数>1的位置上的所有其他峰值。



图20. $\Sigma\Delta$ 和PWM调制信号（用于频谱比较）

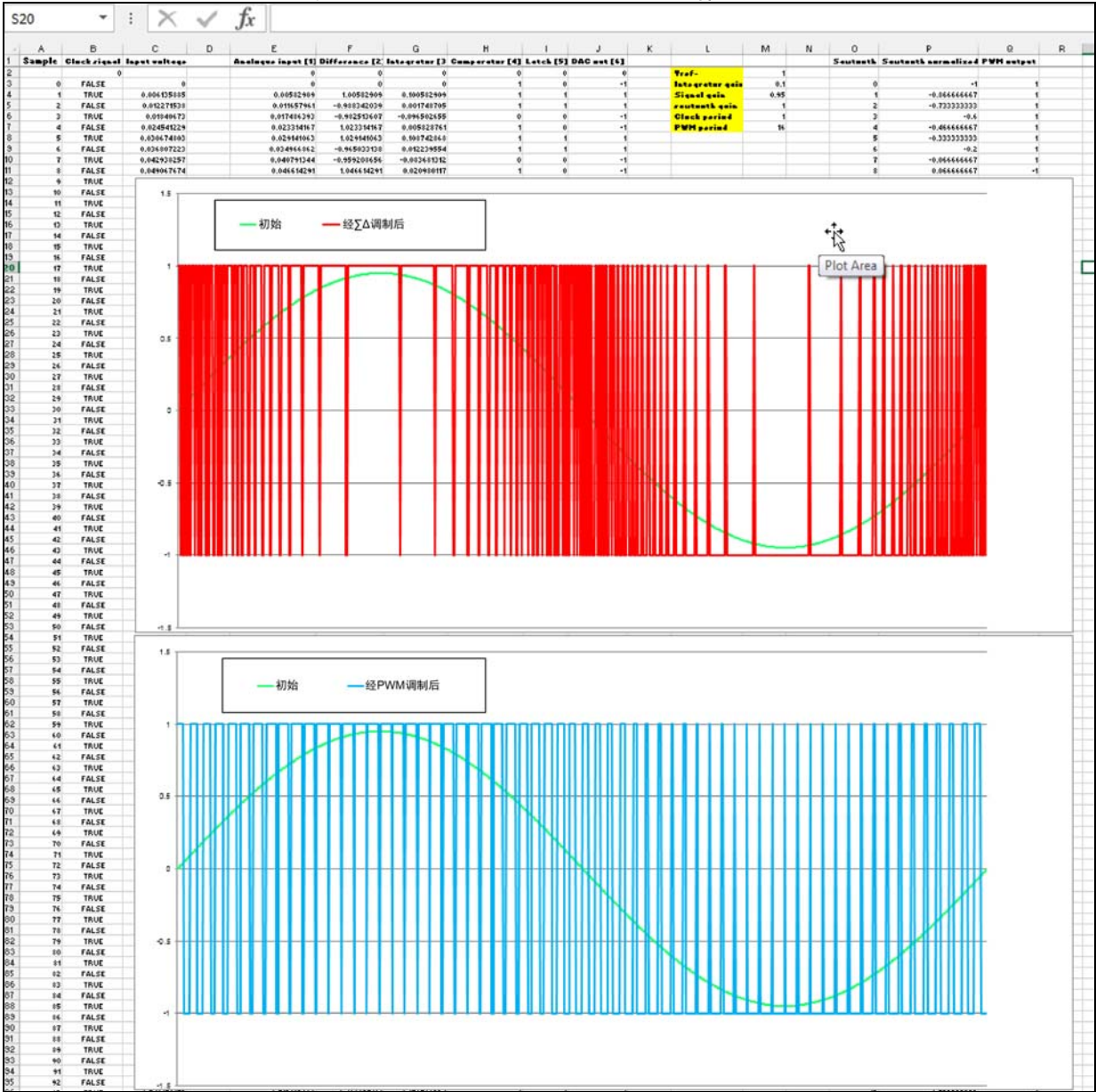


图21. $\Sigma\Delta$ 信号和PWM信号的频谱

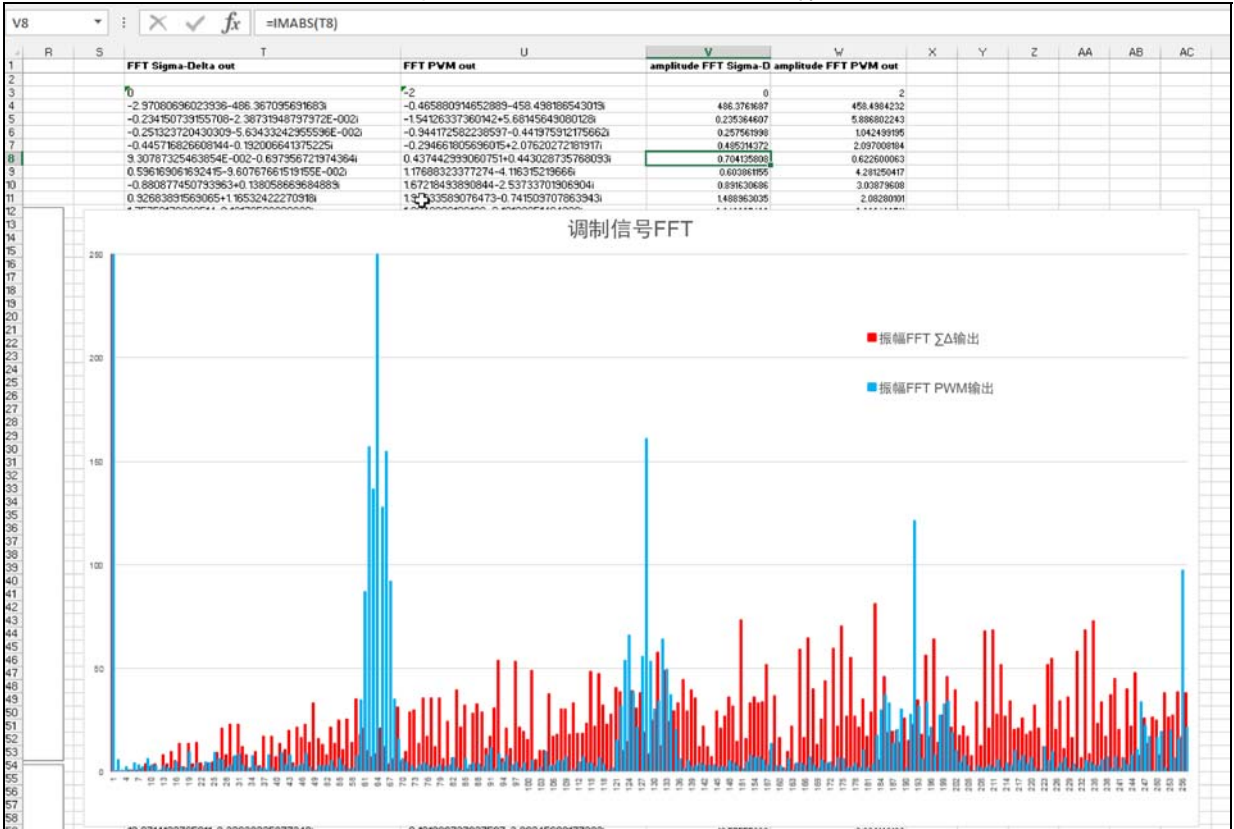
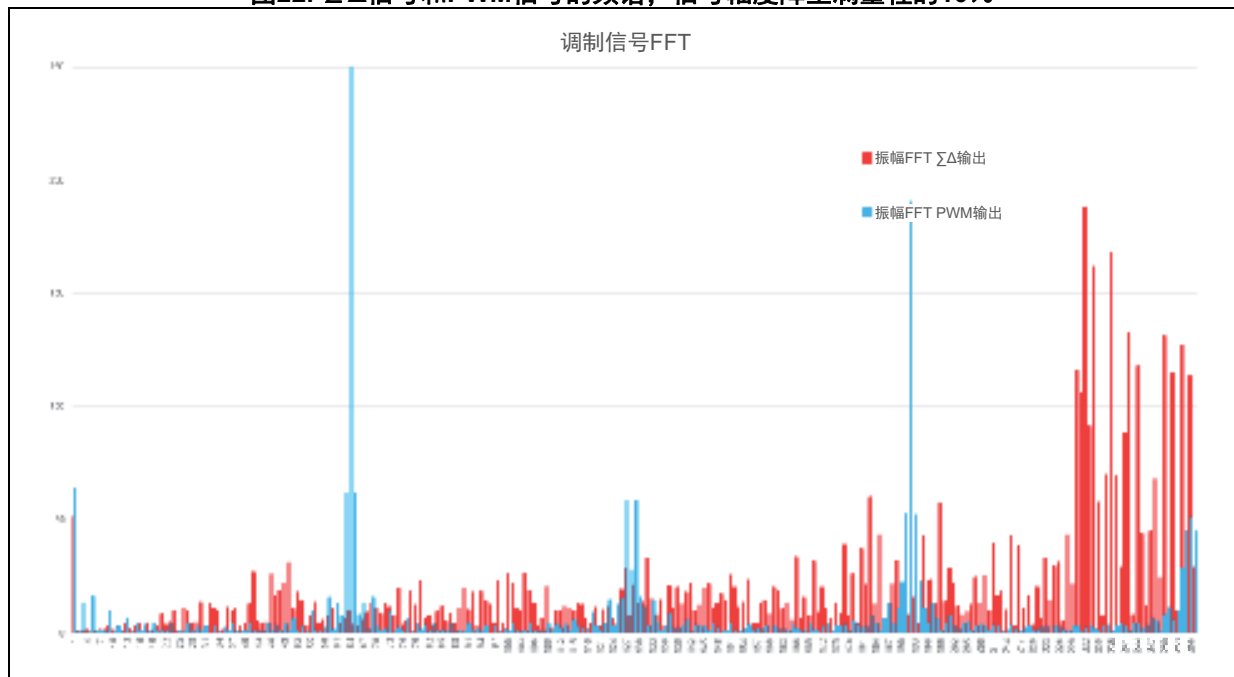


图 21 表明，PWM 信号频谱的能量集中在多个调制频率周围的峰值上（此时，调制频率 = $16 \times$ 信号基频）。这归因于 PWM 调制的特性，即，具有恒定调制频率和可变占空比。峰值的幅度随着频率的增加而减小。

相反地，一阶 $\Sigma\Delta$ 调制频谱表明，在一次谐波下的幅度较低，而后，幅度随着频率而逐渐增加。

根据先前观察的结果，与 PWM 调制信号相比，在重建原始信号时（如图 21 所示，指数 1 处的正弦波信号）， $\Sigma\Delta$ 调制信号对低通模拟滤波器的要求更低。低通滤波器（模拟滤波器）的截止频率必须略高于待观察的原始信号的有用模拟带宽，且必须足够清晰，以便充分抑制由信号调制产生的所有高频分量。由于与 $\Sigma\Delta$ 调制信号相比，PWM 信号有更多不必要的能量集中在低频，因此，低通模拟滤波器必须更锐化（高阶模拟滤波器）。

$\Sigma\Delta$ 调制信号的另一个特点是，在频谱下部的量化噪声几乎是恒定的，与输入信号分辨率无关；而对于 PWM 调制信号而言，量化噪声的降级会影响整个频谱，特别是一次谐波。将输入信号幅度从满量程的 95%（图 21）降低至满量程的 10%（图 22），即可明显地观察到，PWM 调制信号对量化噪声降级具有较高的灵敏度。

图22. $\Sigma\Delta$ 信号和PWM信号的频谱，信号幅度降至满量程的10%

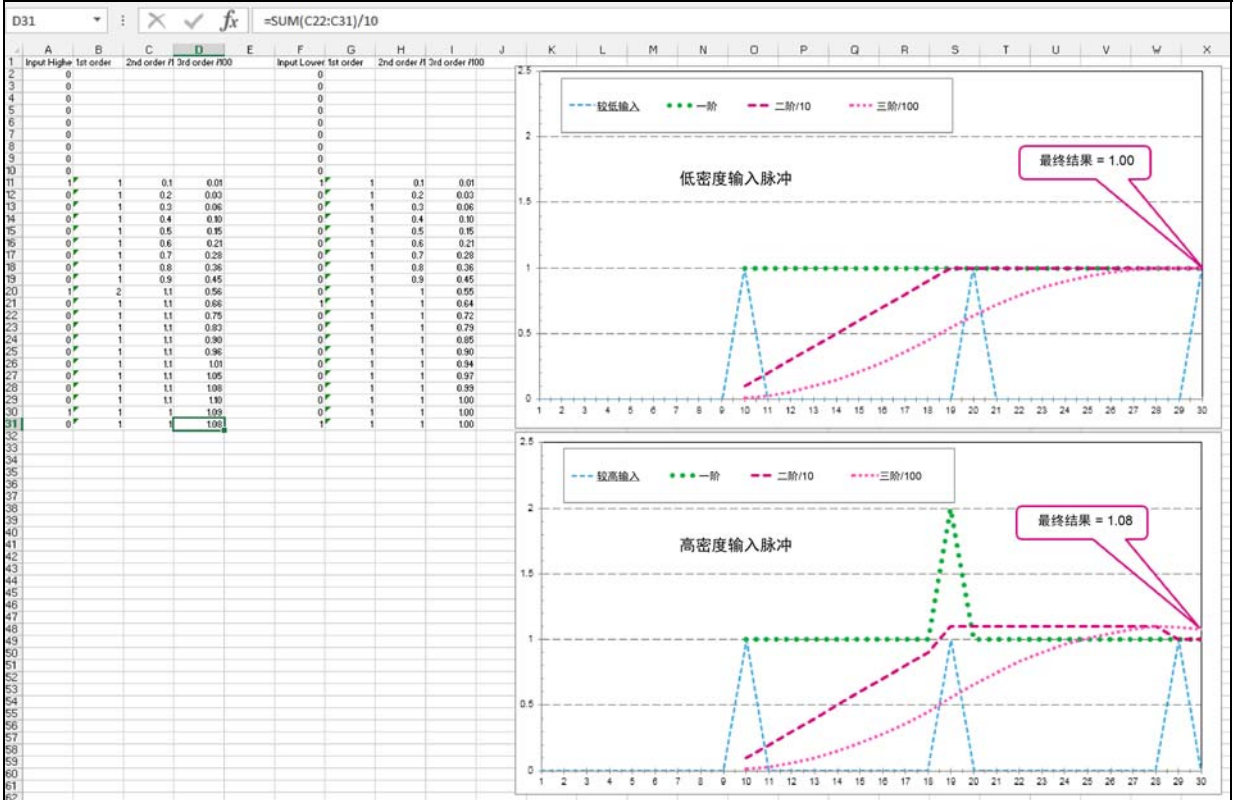
4.3.5 高阶滤波器操作

该模拟请参见[TUTORIAL]的第六张工作表。它显示了如何通过对相同数据应用多重平均技术（高阶滤波器的原理）来提高滤波器输出分辨率。

对于该模拟，取平均值的周期是10个样本（FOSR = 10）。在取平均值的周期内，输入信号由单个脉冲组成（一个样本为1，另外九个样本为0）。执行多重平均。观察到两种不同的情况（参见图 23）：

1. 在第10、20和30个样本处发生等距脉冲。这种情况对应于恒定密度信号。
2. 在第10、19和29个样本处发生非等距脉冲。这种情况对应于密度变化的信号。

图23. 高阶滤波器 - 多重平均原理



显示在上述两种输入信号情况下，以及进行1、2或3次取平均值之后的结果（类似于滤波阶数FORD=1，FORD=2，FORD=3）。每10个采样周期（取平均值周期结束）进行最终输出的采样。

在第一种情况下，由于输入脉冲是等距间隔（每10个样本），无论滤波阶数如何，输出（在第10、20、30个样本处的采样结果）始终为1.00。

在第二种情况下，输入脉冲略微靠近（在第10、19和29个样本处的脉冲），因此脉冲密度略高。这对于第一种情况所述的简单平均滤波器并无影响，因为这种滤波器始终提供相同的结果1.00（在第1、2和3个取平均值周期中保持恒定）。这是因为，它仅在一个周期（10个样本）内执行平均。这意味着一阶滤波器无法检测脉冲密度的变化（或者，必须设计具有较高FOSR、较低数据速率的滤波器）。

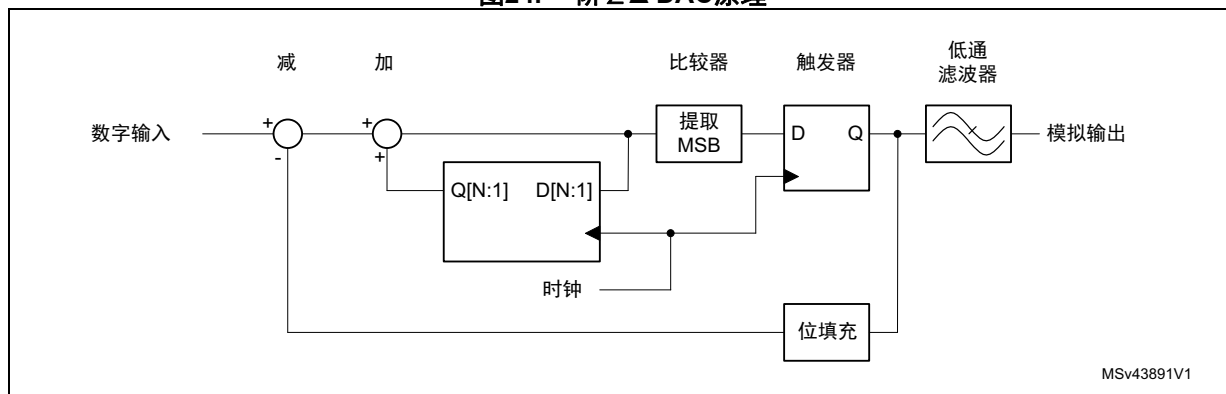
双重平均则略微不同。第一个有效结果出现在第2个取平均值周期后的输出上，因为双重平均使用2个周期的信号来构建一个最终样本。根据双重平均得出的结果如下：1.10（在第2个周期之后得到的结果）和1.00（在第3个周期之后得到的结果）。这意味着二阶滤波器能够检测到在信号的第2个取平均值周期期间发生的脉冲密度增加，以及在第3个取平均值周期期间发生的脉冲密度减少。

三重平均可提供更高的精度。第一个有效结果出现在第3个取平均值周期后的输出上，因为三重平均使用3个周期的信号来构建一个最终样本。根据三重平均得出的结果如下：1.08（在第3个周期之后得到的结果）。这意味着三阶滤波器可以提供更高的脉冲密度精度。

4.3.6 $\Sigma\Delta$ DAC模拟

该模拟请参见[TUTORIAL]的第七张工作表。 $\Sigma\Delta$ 调制不仅可用于ADC转换，还可用于DAC转换。在这种情况下， $\Sigma\Delta$ 调制器图的所有模拟模块（参见图 4： $\Sigma\Delta$ 调制原理）均替换成等效的数字模块（图 24）。

图24. 一阶 $\Sigma\Delta$ DAC原理

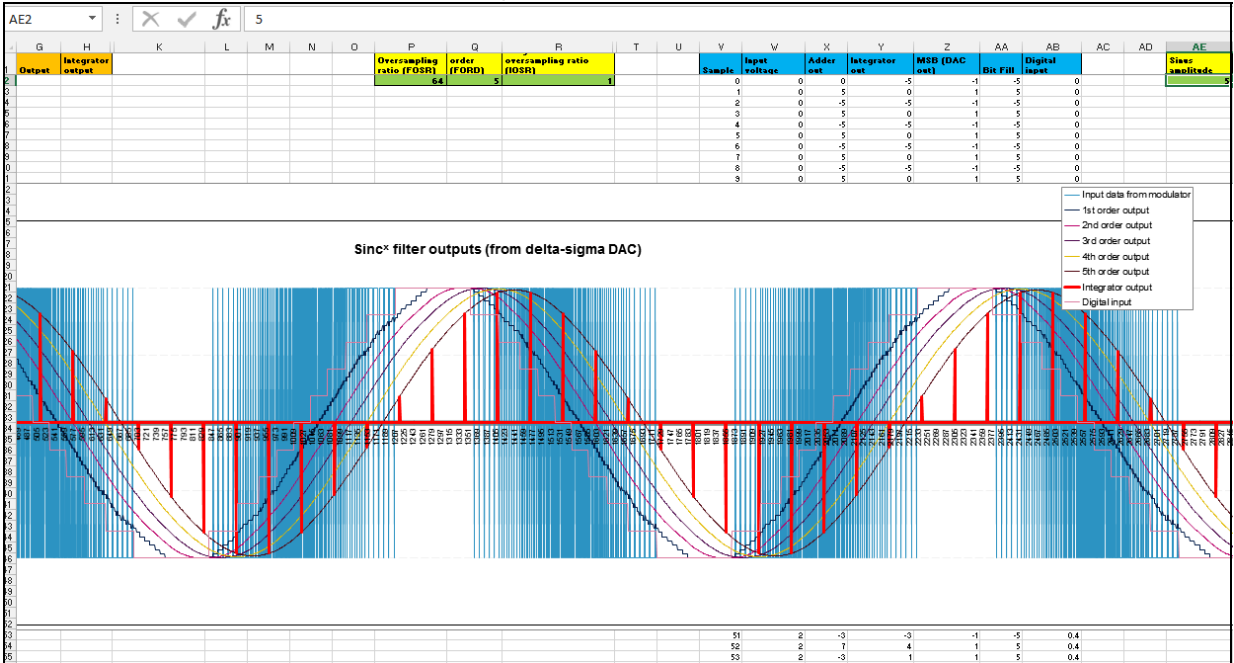


以下是每个块在数字域中的特性：

- “减法”块是数字运算。
- “积分器”块也是一个简单的累加运算（输入位的总和）。
- “比较器”块是数字比较器。它确定来自积分器的数字数据字的符号（最高有效位=符号位）。
- “位填充”块相当于图 4： $\Sigma\Delta$ 调制原理使用的1位DAC。它替换成ADC，根据比较器输出提供的符号，其输出结果为 $\pm$ *digital_reference_word*。“*digital_reference_word*”的幅度应当对应于进入比较器的数字信号的输入范围（相当于 $\Sigma\Delta$ 转换器的 $+V_{ref}$ 、 $-V_{ref}$ ）。
- “低通滤波器”块是一个模拟滤波器，它对快速1位数字流进行模拟滤波。快速1位数字流表示 $\Sigma\Delta$ 信号，在这种情况下，由并行N位宽数字输入信号（ $\Sigma\Delta$ 调制器）构建。1位数字流具有与 $\Sigma\Delta$ 调制器（噪声整形）中的1位数字流相同的属性，并且可以使用更简单的模拟滤波器（相对于PWM信号）进行滤波。

该 $\Sigma\Delta$ DAC的模拟得到的输入是一个数字化的正弦波信号，幅度为 ± 5 （参见图 25 的“数字输入”粉红色信号）。在实际应用中，快速1位数字流输出（图 25 的蓝色信号）应执行模拟低通滤波。出于模拟的需要，模拟低通滤波由数字Sincx滤波器代替。

图25. $\Sigma\Delta$ DAC模拟



尽管数字输入正弦波量化仅使用11个离散电平[-5, ..., + 5]，最终的 $\Sigma\Delta$ DAC输出（参见图 25的各种滤波阶数输出）显示了妥善重建的“模拟”正弦波。

同样地，与PWM调制相比，使用 $\Sigma\Delta$ 的优点是输出信号具有更简单的低通滤波（在DAC情况下为模拟）。这是出于与第 4.3.4节： [\$\Sigma\Delta\$ 调制的噪声整形](#)开发的信号相同的原因。

4.3.7 高通滤波器模拟

该模拟请参见[TUTORIAL](#)的第八张工作表。高通滤波器适用于需要从输入信号中去除DC分量和/或低频噪声的多种应用（音频应用，包括麦克风输出中的静压变化、AC电能测量、随温度变化的DC偏移.....）。

DFSDM不实现任何高通滤波，但它可以作为对DFSDM采样数据的后期处理，在软件中轻松实现（作为使用外部硬件高通滤波器的替代方案）。

模拟中使用的高通滤波器由以下公式建模：

$$y(n) = (\text{coeff} / 256) \times (y(n-1) + x(n) - x(n-1))$$

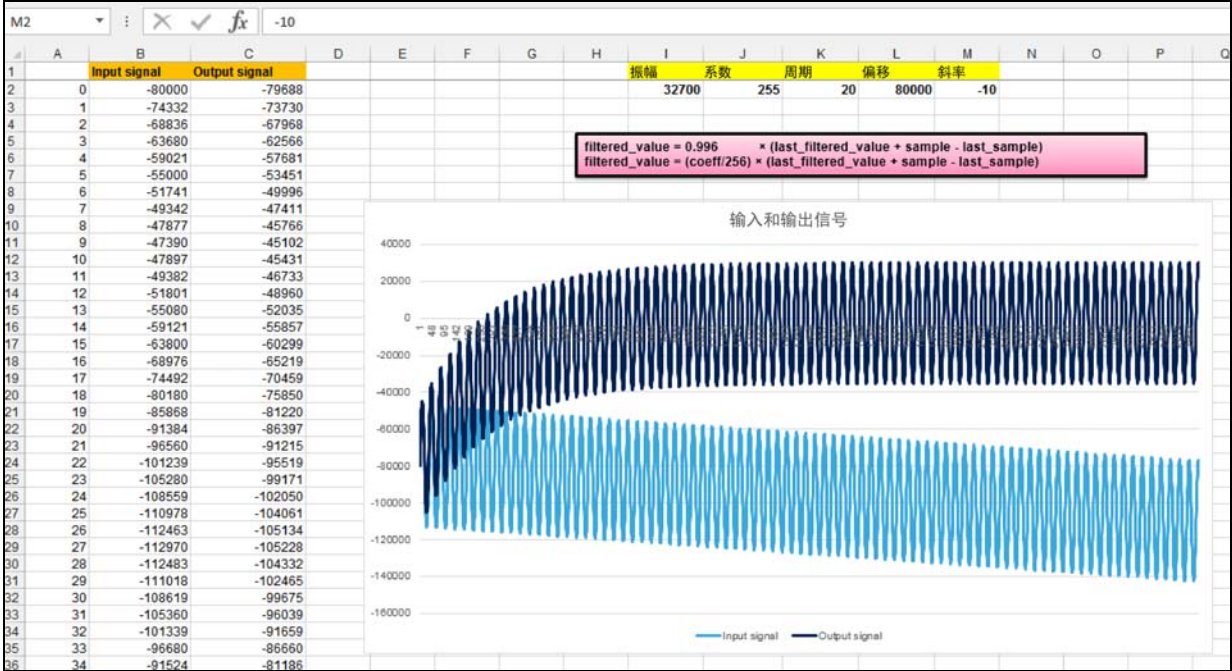
其中：

- y(n): =模拟中的“filtered_value”（滤波器输出）
- coeff: =模拟中的“coeff”（滤波器截止频率）
- y(n-1)=模拟中的“last_filtered_value”（前一个滤波器输出）
- x(n)=模拟中的“样本”（滤波器输入样本）
- x(n-1)=模拟中的“last_sample”（滤波器前一个输入样本）

该滤波器的复杂性有所降低，只需一次乘法、两次加法和一次移位运算（/256）。

在模拟中，将具有不断减少的DC偏移值的正弦波应用于高通滤波器（图 26 的浅蓝色曲线）。

图26. HP滤波器模拟



高通滤波器输出由图 26 的深蓝色线表示。在适配周期之后，高通滤波器抑制DC和低频内容。适配速度取决于“coeff”值（滤波器的截止频率）。

4.4 DFSDM的其他功能

4.4.1 支持数字麦克风 (MEMS)

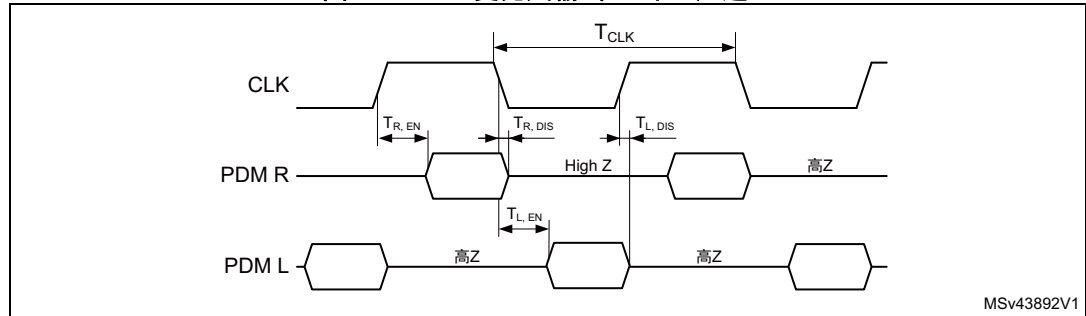
根据音频应用领域的标准化规定，PDM调制（脉冲密度调制）是数字麦克风的常见输出格式。PDM信号相当于ΣΔ调制信号，因此，DFSDM支持PDM信号。

数字麦克风是使用半导体技术制造的MEMS器件（微机电系统）。此类麦克风的有源致动器包括一张膜和一对微电极，其中一个电极是固定的，另一个电极结合在膜内。当空气压力（声音）施加到膜上时，移动电极远离其默认位置，两个电极之间产生电容变化。感应信号由内置电子处理，作为PDM调制信号输出（脉冲密度调制）。

数字麦克风需要外部时钟信号（麦克风CLK输入信号），数据作为PDM调制信号，通过DATA输出线发送。时钟速度的范围通常介于1至3.2MHz之间。DFSDM_CKOUT输出信号提供时钟信号，后者将麦克风输出数据速率定义至DFSDM。图 28 是立体声数字麦克风和DFSDM之间典型接线的原理图。

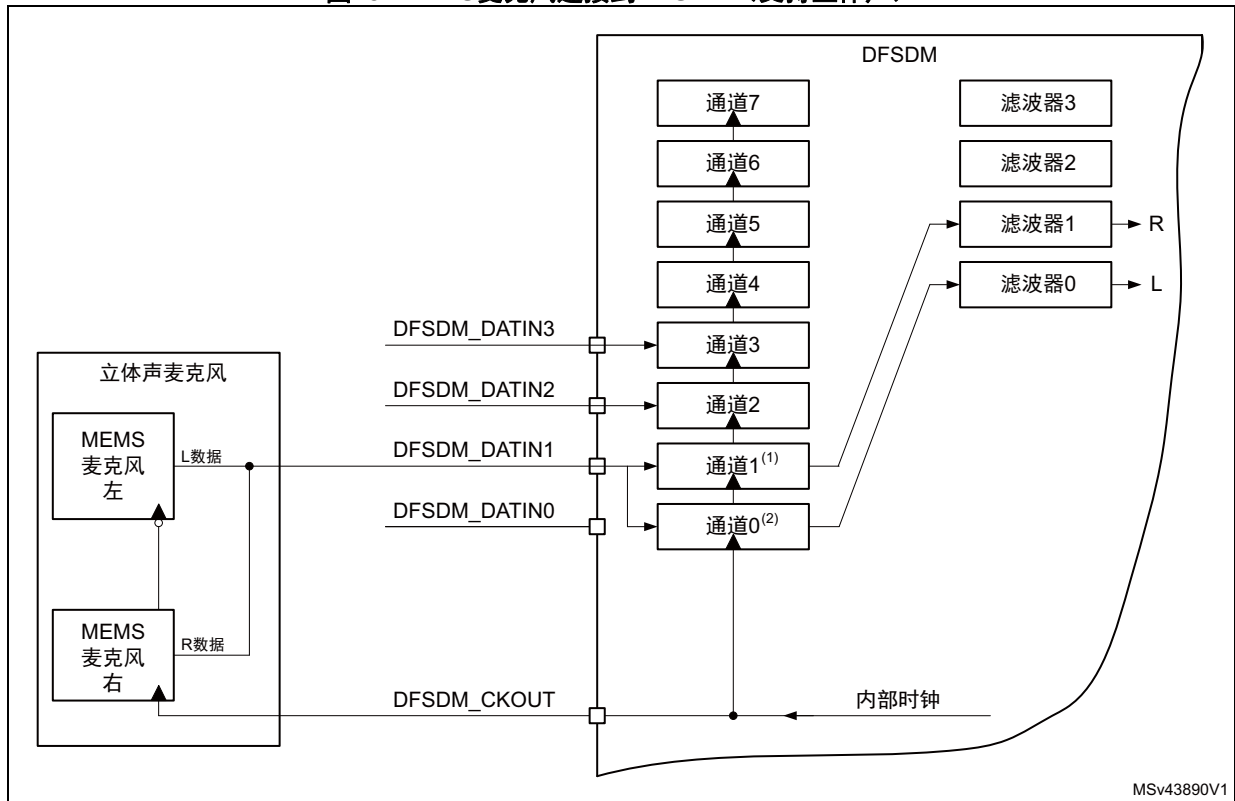
DFSDM可通过一条线路并联2个麦克风（立体声配置：左右声道）。两个麦克风共用数据和时钟信号。时钟信号从DFSDM_CLKOUT引脚分配到左右麦克风。来自两个麦克风的输出数据信号在同一条线路上复用：左侧麦克风在时钟上升沿提供数据，右侧麦克风在时钟下降沿提供数据（参见图 27）。左声道或右声道麦克风的配置通常是通过在麦克风上配置引脚来实现的（L/R选择引脚）。

图27. MEMS麦克风输出（L和R声道）



MSv43892V1

图28. MEMS麦克风连接到DFSDM（支持立体声）



MSv43890V1

1. 直接输入，下降沿采样（R数据）。
2. 从下一个声道重定向，上升沿采样（L数据）。

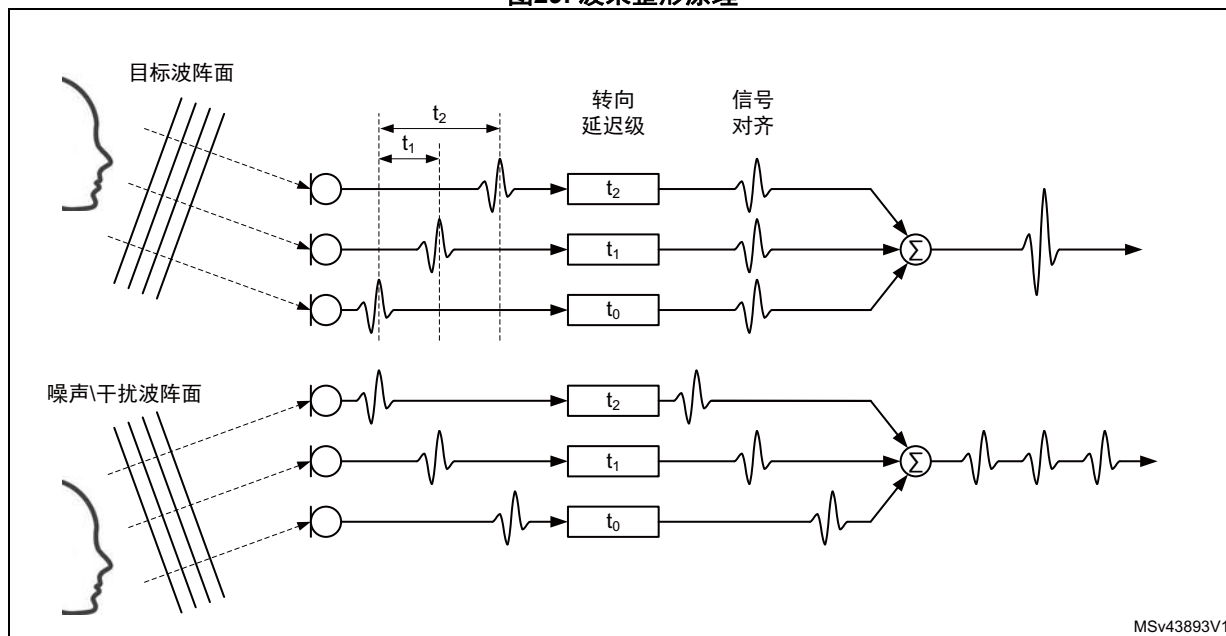
DFSDM将DATA线上的两个麦克风信号进行分离。可以重定向DFSDM声道x的输入，以便将此声道 (x+1) 作为输入。接下来，将声道x配置为在上升沿进行数据采样，将声道(x+1)配置为在下降沿进行数据采样。两个声道的时钟信号相同，且内部连接至DFSDM_CLKOUT信号。通过此配置，声道x从左侧麦克风接收数据，而声道(x+1)从右侧麦克风接收数据。两个声道向各自的数字滤波器馈送数据，最终为左右麦克风声道输出两个独立的并联数据流。参见图 28，了解使用DFSDM的立体声麦克风应用的完整图形。

4.4.2 支持波束整形

借助波束整形技术，一组固定的多向传感器阵列可以支持来自特定方向的信号（下例中的麦克风阵列）。可以通过固件更改首选方向（无需更改传感器的位置）。

在以下示例中，麦克风阵列位于一条等时间间隔的线上（参见图 29）。音频源以任意角度放置在麦克风阵列的前方，同时，其周围存在不必要的噪声和干扰。如果将来自所有麦克风的信号简单加总在一起，则仅对垂直于阵列的声音进行相干求和（因此被放大）。来自非垂直方向的声音无法相干求和。

图29. 波束整形原理



为了放大来自另一个方向（而不是垂直方向）的信号，一种解决方案是机械地旋转麦克风阵列，另一种解决方案是为阵列中的每个麦克风增加一个特定的延迟，以便在加法器的输入端，对经过各个麦克风到达的有用信号的不同传导延迟进行补偿。当调整所有麦克风的延迟线时，仅来自有用音频源方向的声音同相且被放大（相干组合），不同角度的声音的放大幅度较低（不相干组合）（参见图 29）。

以上概述了波束整形算法（简化视图）。可以实现更精细的算法，提供更出色的性能。

可使用DFSDM实现波束整形，详见下述。DFSDM能够感应来自多个麦克风的信号，并执行分别处理，从而提供并行输出样本。来自不同麦克风的输出样本存储在微控制器存储器的独立数据缓冲区中（例如，来自4个麦克风的4个存储器缓冲区）。接着，微控制器将样本相加，确定最终输出。延迟线分两步实现：

1. 粗步长：可以相对于一个缓冲区，将DFSDM输出缓冲区（用于给定的麦克风）移动给定数量的样本（一个样本对应于 $1/f_{\text{datarate}}$ 的延迟，其中， f_{datarate} 表示输出数据速率（典型音频速率：44.1kHz、22.05kHz、16kHz.....）。1个样本的移位对应于声音在空气中的传播距离 $s = v/f_{\text{datarate}}$ （ v 表示声音在空气中的速度~343m/s）。
粗步长示例：
 - $f_{\text{datarate}} = 44.1 \text{ kHz}$: $s = 343/44100 = \sim 7.8 \text{ mm}$
 - $f_{\text{datarate}} = 16 \text{ kHz}$: $s = 343/16000 = \sim 21.4 \text{ mm}$
2. 细步长：为了支持阵列中短距离排列的麦克风，并且能够微调首选声源的角度，需要更精细的步长。移动麦克风和DFSDM之间的1位数据流上的样本，而不是移动存储器中的DFSDM输出样本，即可实现细步长。按照PDM频率，对输入的1位样本进行采样。以下是细步长示例：
 - $f_{\text{PDM}} = 3 \text{ MHz}$: $s = 343\text{ms}^{-1} / 3000000\text{Hz} = \sim 0.11\text{mm}$
 - $f_{\text{PDM}} = 200 \text{ kHz}$: $s = 343\text{ms}^{-1} / 200000\text{Hz} = \sim 1.7\text{mm}$

这些值的范围也能够兼容于微型麦克风阵列，用于对接收声音的首选角度进行微调。

用于波束整形微调的DFSDM细步长电路不是基于延迟线，而是跳过输入1位流的部分样本。来自串行收发器的部分1位样本被（时钟信号门控）屏蔽，不会被发送至数字滤波器。基于跳跃时钟脉冲的原理如图 30 所示。

图30. 用于波束整形的脉冲跳跃实现

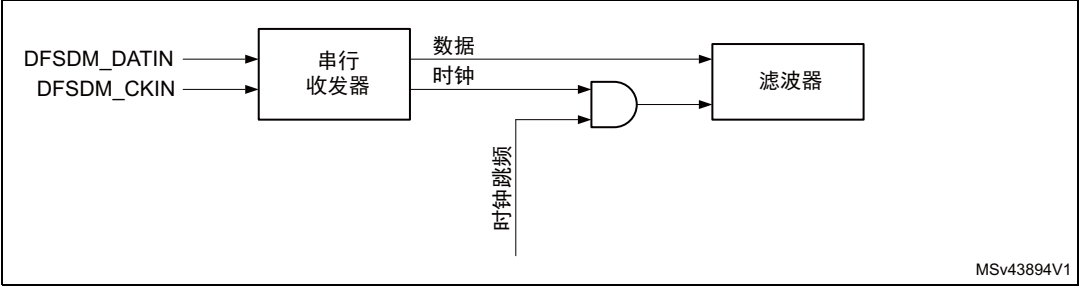


图 31 提供了一个示例，说明了脉冲跳跃机制及其与延迟线的相似性。

图31. 脉冲跳跃示例 (FOSR=8)

Input samples (no sample skipping applied):																									
sample #	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	
Left	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20	S21	S22	S23	S24	
Right	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20	S21	S22	S23	S24	
Output samples (FOSR=8):																									
sample #	1								2								3								
Left	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20	S21	S22	S23	S24	
Right	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20	S21	S22	S23	S24	
	Right sample built from same input samples number like on Left channel => <u>no delay line on Right channel</u>								Right sample built from same input samples number like on Left channel => <u>no delay line on Right channel</u>								Right sample built from same input samples number like on Left channel => <u>no delay line on Right channel</u>								
Input samples (with 3 samples skipped on Left channel):																									
sample #	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	
Left	S1	S2	S3	S4	S5	S6	S7	S8	S9	-	-	-	S13	S14	S15	S16	S17	S18	S19	S20	S21	S22	S23	S24	
Right (3 samples delay applied)	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20	S21	S22	S23	S24	
Output samples (FOSR=8):																									
sample #	1								2								3								
Left	S1	S2	S3	S4	S5	S6	S7	S8	S9	S13	S14	S15	S16	S17	S18	S19	S20	S21	S22	S23	S24	S25	S26	S27	
Right	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20	S21	S22	S23	S24	
	Right sample built from same input samples number like on Left channel => <u>no delay line on Right channel</u>								Right sample built from older input samples => like <u>delay line on Right channel</u>								Right sample built from older input samples => like <u>delay line on Right channel</u>								

图 31 的顶部表示正常操作，未应用时钟跳跃。每个输出样本均由8个输入样本构建（FOSR = 8）。如要形成波束，需要同时启动多个DFSDM滤波器。

在图 31 的底部，左声道跳过了样本S10、S11、S12。滤波器（配置：FOSR=8）再等待3个样本，以便为每个声道获得8个完整的1位样本，从而能够构建一个输出样本。从滤波器输出的角度而言，对于相同的最终输出样本索引，右声道使用的1位输入样本比左声道的更早。如果右声道具有3样本延迟线（延迟线缓冲区保持最后3个样本），也会出现上述相同的行为。

以上概述了波束整形算法（简化视图）。可以实现更精细的算法，提供更出色的性能。

4.4.3 音频时钟支持 - 独立时钟操作

DFSDM可以通过DFSDM_CKOUT引脚提供的时钟信号，为外部 $\Sigma\Delta$ 调制器提供时钟。此引脚的时钟频率决定了输入采样频率，后者也取决于输出数据速率频率。可以选择以下一种时钟源，用于驱动DFSDM_CKOUT：

- DFSDM 时钟：
 - APB 时钟
 - 系统时钟（独立于APB时钟分频器）
- PLL时钟（用于I2S的音频PLL）

将所选时钟源的频率除以2-256范围内的因数（根据DFSDM_CHyCFGR1寄存器中的CKOUTDIV字段设置的预分频比），得到DFSDM_CKOUT频率。

为了对DFSDM_CKOUT频率进行微调，必须选择PLL（同时提供乘法和除法因数）作为时钟源。对于需要精确采样频率的音频应用而言，这是最适当的选择。

DFSDM外设本身（收发器、滤波器、附加功能）使用APB时钟或系统时钟进行数字处理。该“处理”频率应比输入采样频率快4倍（如果使用曼彻斯特编码，则快6倍）。

为了节省功耗，采样频率和处理频率都应配置为应用所需的最小值。

4.5 DFSDM功耗优化

DFSDM功耗取决于：

- 使能的组件：
 - 使能的通道数
 - 使能滤波器数量
 - 使能的功能数量（模拟看门狗、短路检测器）
- 时钟速度：
 - 收发器
 - 数字滤波器（+附加功能）

为了优化功耗，仅在需要时使能所需的组件（收发器、滤波器、附加功能），应使用最小时钟速度来实现所需的输入和输出数据速率。

4.5.1 睡眠模式下的功耗优化

DFSDM可用于睡眠模式中。此时，DFSDM通常用于监控模拟信号（使能模拟看门狗时），如果达到模拟阈值，则用于唤醒CPU。

此类应用的示例包括婴儿监控或碎玻璃检测，其中，CPU几乎始终处于睡眠模式，DFSDM通过模拟监视器来监控声级。如果达到声音阈值，则DFSDM模拟看门狗中断唤醒CPU，并收集采集到的数据，而CPU对DFSDM收集的数据进行分析（以评估此数据是否为婴儿声音或碎玻璃声）。在睡眠模式期间，DFSDM不需要收集数据，只应打开模拟看门狗功能，用于优化功耗。

以下是用于低功耗操作（声音检测）的DFSDM和MCU配置示例：

- 时钟：
 - 禁用除DFSDM之外的所有外设
 - 以最低频率使用系统时钟，用于执行DFSDM操作。（例如：4 MHz，使用内部MSI振荡器，旨在减少更多功耗）
 - 将AHB和APB时钟降至最低，因为在睡眠模式下无需访问外设。使用高AHB和APB预分频器（例如：AHB预分频器= 512，AHB时钟= $4 \text{ MHz}/512 = 7.8 \text{ kHz}$ ）。
 - DFSDM使用系统时钟（例如：4 MHz），这是因为，在数字滤波器处理中，APB时钟太慢。
 - 数字MEMS麦克风的采样时钟应小于DFSDM时钟/4（例如：麦克风操作适合使用1 MHz）。
- DFSDM：
 - 配置滤波器，获得适当的输出数据速率（例如：16 kHz）。
 - 设置模拟看门狗，用于监控输出数据，设置适当的声音检测阈值。
 - 禁用最终数据的溢出错误，因为该数据不是从DFSDM收集的（仅模拟看门狗处于工作模式）。
- 唤醒后（模拟看门狗中断）：
 - 设置回AHB、APB时钟（预分频器），用于执行高速操作，加速DFSDM通信和数据传输（例如：AHB预分频器= 1，AHB时钟= 4 MHz）。
 - 可选择重新配置DFSDM，提高输出数据速率，从而实现更出色的声音识别分析（例如：44.1 kHz）。
 - 启动从DFSDM至存储器缓冲区的数据收集（例如：DMA数据传输）。
 - 对每个数据缓冲区执行声音识别（例如：FFT分析）。
 - 如果检测到关键声音，请执行必要的操作（调用警报）。如果声音不重要（或者在一段时间后并未检测到此声音），则再次进入睡眠操作，等待模拟看门狗再一次唤醒。

随着上述DFSDM情境的持续，（由于DFSDM活动而导致的）微控制器功耗的增加因数被限制在2倍左右（相对于没有任何DFSDM活动、且具有相同系统时钟频率的严格睡眠模式下的功耗）。

5 DFSDM外设配置教程

DFSDM功能类似于具有外部模拟前端部件的A/D转换器。DFSDM具有极大的灵活性，但需要一些方法才能在所有设置之间切换。以下教程帮助用户根据A/D转换的应用程序要求来配置DFSDM。

5.1 配置介绍

A/D转换的基本应用参数是：

- 输入模拟范围（ $\Sigma\Delta$ 调制器属性）
- 最小输出数据速率
- 数据速率精度设置
- 最小输出数据分辨率
- 收集的通道数（每个ADC）
- 附加功能：
 - 模拟看门狗
 - 短路检测器
- 数据传输类型：DMA、中断、轮询。

影响DFSDM配置的应用程序相关属性：

- 系统时钟范围
- 电源电压
- 功耗限制
- 采用的 $\Sigma\Delta$ 转换器（或数字麦克风）属性：时钟速度范围、输入模拟范围、 $\Sigma\Delta$ 调制器的阶数。

在以下章节中，本教程将介绍如何配置各种DFSDM部件，以满足应用程序的要求。

5.2 时钟配置

根据数据要求，指导DFSDM时钟的选择：

- 最小输出数据速率
- 数据速率精度
- 最小输出数据分辨率

我们的目标是设置滤波器参数和输入采样时钟速率，以达到输出数据速率。

输入采样时钟选择由 $\Sigma\Delta$ 调制器（或数字麦克风）的特性驱动：

请参见下文 [表 2](#)，了解适用于两种不同应用的时钟配置示例。

5.3 收发器

5.3.1 串行收发器配置

串行收发器将输入引脚（DFSDM_DATIN和DFSDM_CKIN）上的数据进行转换，并提供给滤波器。收发器的配置取决于连接到DFSDM（ $\Sigma\Delta$ 调制器或数字麦克风）的外部设备的特性。

下文 [表 2](#) 介绍了适用于两种不同应用的DFSDM配置示例：

应用1：

- 应用：温度测量，包括测量Pt100传感器上的电压和电流，并确定Pt100传感器电阻/温度。计时器每1秒触发一次温度测量。
- 外部设备：双外部 $\Sigma\Delta$ 调制器STPMS2，用于电压和电流通道测量。
- 物理连接：
 - STPMS2仅使用一条SPI线（引脚：CLK、DAT）。
 - STPMS2 CLK引脚由DFSDM_CKOUT引脚驱动
 - STPMS2 DAT引脚连接到DFSDM_DATIN1（通道1引脚上的串行数据输入，通道0输入重定向到通道1引脚，而后，DATIN1上的信号连接到通道0和通道1）。
 - 不使用DFSDM_CKIN1引脚（通道1上的串行时钟输入）。使用来自CKOUT引脚的内部重定向时钟对数据进行采样（SPICKSEL [1:0] = 1）。
- 后期处理：
 - 来自通道0（U，电压）的数据由滤波器0处理。
 - 来自通道1（I，电流）的数据由滤波器1处理。
 - Pt100传感器的电阻在固件中确定为 $R = U/I$ ，用于推断Pt100传感器温度。

应用2：

- 应用：在48 kHz数据速率下录音。
- 外部设备：MEMS数字麦克风MP34DT01-M。
- 物理连接：
 - MP34DT01-M有一条数据线（引脚：CLK、DOUT）。
 - MP34DT01-M CLK引脚由DFSDM_CKOUT驱动。
 - MP34DT01-M DOUT引脚连接到DFSDM_DATIN0（通道0上的串行数据输入）。
- 后期处理：
 - 来自通道0（麦克风）的数据由滤波器0处理。
 - DMA以连续模式，将数据速率为48 kHz的输出流存储在RAM缓冲区中。



表2. DFSDM应用示例

参数		应用1：温度测量		应用2：音频采样	
说明	选项	设置 ⁽¹⁾	注释	设置 ⁽¹⁾	注释
串行协议选择 (SITP[1:0])	SPI类型	x	STPMS2使用SPI通信	x	MP34DT01-M使用SPI类型协议。
	曼彻斯特类型	-	-	-	-
主时钟	DFSDM ⁽²⁾	x	STPMS2具有时钟输入引脚	x	MP34DT01-M需要时钟输入。
	外部器件 ⁽³⁾	-	-	-	-
有效采样沿	上升沿	-	电流流在上升沿有效	x	MP34DT01-M的引脚“L/R”处于GND电平 - 上升沿有效。
	下降沿	-	电压流在下降沿有效	-	-
时钟源	APB 时钟 系统时钟 PLL 时钟	系统时钟 (f _{HSI16})	f _{CKIN} 要求 2 MHz ⁽⁴⁾	PLL 时钟	f _{CKIN} 要求 3.072 MHz ⁽⁵⁾
输入时钟频率	f _{CKIN} /f _{CKOUT}	2 MHz	f _{CKIN} 配置： f _{CKIN} = f _{HSI16} / (CKOUTDIV + 1)，其中 f _{HSI16} = 16 MHz CKOUTDIV = 7	3.072 MHz	f _{CKIN} 配置： f _{CKIN} = f _{HSE} * N / (Q * (CKOUTDIV + 1))，其中 f _{HSE} = 16 MHz (晶体 - 精确时钟) N = 48 Q = 2 CKOUTDIV = 124
通过一条串行线路发送的数据流数量	一个流 ⁽⁶⁾	-	STPMS2也支持一个流，但需要额外一条线/一个引脚	x	使用单声道麦克风。
	两个流 ⁽⁷⁾	x	本例中，电流和电压信号共用一条线。只需要一个引脚/电线。	-	-
时钟精度	精度低	x	无需精确时钟 - 定时器每秒触发一次温度测量。基于DFSDM时钟（16MHz）的时钟信号：CKOUTSRC = 0	-	-
	高精度	-	-	x	要求精确的48 kHz数据速率（音频标准）。

1. 'x'=有效设置，'- '=无效设置。
2. 设备需要外部时钟（时钟输入到设备）。
3. 设备使用内部时钟（由设备输出的时钟）。
4. STPMS2时钟输入范围：1至4.915 MHz。
5. f_{CKIN} = 数据速率 * FOSR * IOSR，其中，数据速率 = 48 kHz，FOSR = 64，连续模式。

6. 发送一个流（仅一条采样沿有效）。
7. 发送两个流（第一个流的数据在上升时钟沿有效，第二个流的数据在下降时钟沿有效）。



5.3.2 并行收发器

并行收发器是一种替代方案，它直接从内部微控制器数据源为数字滤波器提供16位宽数据。

并行收发器包含一组32位（2x16位）并行输入寄存器，用于为数字滤波器供电。每个串行输入通道都有一个32位并行输入寄存器（分为2x16位）。DFSDM_CHyCFGR1寄存器的DATMPX[1:0]位允许从串行输入或16位并行输入中选择滤波器输入。如果DATMPX[1:0]=0，则数据来自串行收发器（作为1位串行流），否则数据来自并行收发器（作为16位带符号的数据）。

数据可以由CPU、DMA(DATMPX[1:0]=2)或ADC(DATMPX[1:0]=1)写入并行寄存器。根据嵌入式ADC功能，部分微控制器在ADC和并行输入寄存器之间具有专用的快速内部16位总线。

每个通道均配备32位并行寄存器，此寄存器被分为两个16位寄存器。这样一来，可以对这些寄存器执行32位访问，并在每次写访问时写入2个输入样本。根据DATPACK[1:0]字段定义的所选“数据打包工作模式”，可在32位寄存器中使用高16位和低16位样本：

- 标准模式：仅使用低16位样本
- 交替模式：两个样本用于相同的通道（如同具有2个样本的FIFO缓冲区）
- 双重模式：低16位样本用作给定通道的输入，而高16位样本用作下一个通道的输入。

将一个16位样本写入并行输入寄存器，自动为数字滤波器生成一个采样时钟信号，而后，该信号自动对并行输入寄存器进行采样，作为下一个待处理的输入。因此，最终输出数据速率取决于输入数据速率（CPU或DMA或ADC数据传输速度）。

并行收发器允许在没有CPU干预的情况下执行快速低通滤波。以下介绍3个实际示例：

- ADC样本的后期处理
ADC用作数据源。
ADC配置为与DFSDM一起使用。
DFSDM配置流程如下：
 - 从ADC中选择并行输入：DATMPX[1:0]=1。
 - ADC通过16位内部总线，直接将其输出数据发送到并行寄存器（仅使用低16位部件）。在ADC数据输入的情况下，执行数据打包并无任何意义，因此，建议设置DATPACK[1:0]=0。
 - DFSDM根据滤波器参数（FOSR、FORD、IOSR）的设置处理数据，然后以较低的数据速率产生最终的输出数据样本。示例：对ADC的数据取平均值。
- 对存储在RAM存储器缓冲区的数据执行后期处理（例如，软件高通滤波后，对ADC数据执行低通滤波）
从并行输入寄存器中选择并行输入：DATMPX[1:0]=2。
CPU可以从缓冲区直接将数据写入并行寄存器。或者，DMA可用于将数据加载到并行输入寄存器中。
应根据应用优化数据打包：
 - 如果数据以16位格式存储在缓冲区中，则可以使用DATPACK[1:0]=2设置来提高性能。此时，可以在一次32位传输（FIFO缓冲区）中写入2个连续采样。
 - 如果数据以16位格式存储在缓冲区中，且每个第二个样本均来自于另一个数据源（例如，对应于音频L通道的奇数样本和对应于音频R通道的偶数样本），则可以使用DATPACK[1:0]=3设置来提高性能。此时，可以在一次32位传输中写入一个样本对（将2个样本写入2个不同通道的并行输入寄存器）。
- 对来自另一个通信外设的数据进行后期处理：模拟数据从外设直接传输到给定的并行输入寄存器（DMA），并由DFSDM处理，以生成最终的滤波样本。例如，外部ADC可以连接到微控制器SPI接口，DFSDM用于低通滤波。此时，DFSDM的设置与第二种情况相似。

5.4 滤波器

5.4.1 Sinc滤波器

数字滤波器对串行或并行收发器接收的数据执行数字信号处理。数据可以是1位格式（串行收发器）或16位带符号的整数格式（并行收发器）。收发器接收的每个样本均由数字滤波器自动采样，以便于后期处理。根据数字滤波器配置参数，滤波器可能需要若干个输入样本才能提供第一个输出数据。

在DFSDM中实现的滤波器是Sinc类型的低通滤波器（亦称为梳状滤波器）。该滤波器的基本参数如下：

- FOSR：滤波器过采样率 定义在滑动平均中同时处理的样本数量。FOSR范围是1..1024。
- FORD：滤波阶数 - 定义滑动平均的迭代次数。FORD范围是1..5。

滤波器的配置取决于应用程序要求，详细内容将在接下来的4个小节中讨论：

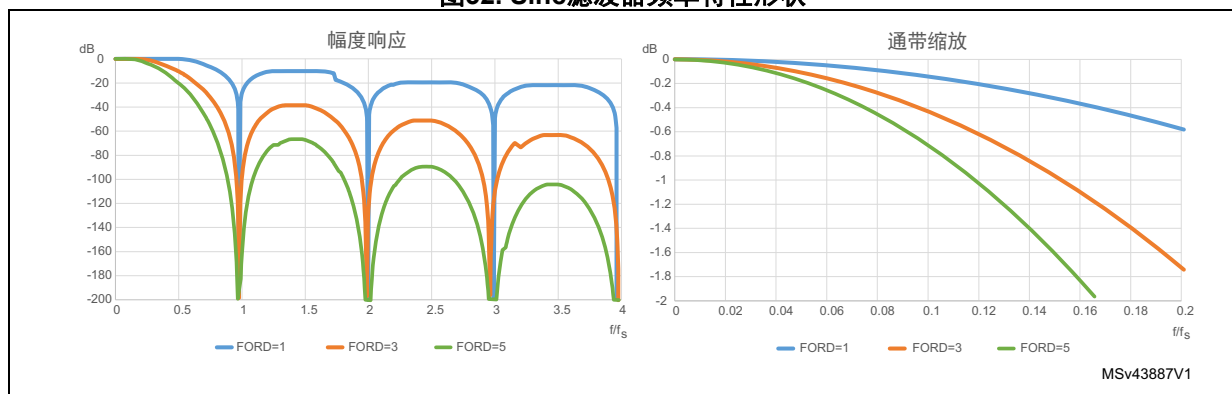
低通滤波特性

滤波器传输频率特性对于应用而言至关重要。[图 32：Sinc滤波器频率特性形状](#)提供当FORD = 1、3和5时，以及当固定FOSR = 8时，Sinc滤波器的频率特性（[图 19](#)介绍了更常见的图）。[图 32](#)中的幅度响应曲线表示在滤波器通带之外，应用于信号高频分量的抑制量（对于FORD=1，抑制量约-20dB，对于FORD=5，抑制量约-100dB）。滤波器阶数越高，高频抑制效果越好（低通滤波器）。使用高滤波器阶数可能会影响有用滤波器通带内的信号。可参见[图 32](#)的“通带缩放”曲线。在通带内，滤波器不平坦，而是会产生轻微衰减，随着频率和滤波阶数的增加而增加。根据模拟输入信号的有效频率带宽，这可能会影响滤波信号的质量。如有必要，可在最终输出数据上，应用软件补偿滤波器，来补偿这种衰减。

陷波频率是指当信号衰减达到最大值时的离散频率（参见[图 32：Sinc滤波器频率特性形状](#)）。Sinc滤波器的第一个陷波频率（及其所有谐波）与FORD无关，仅取决于采样时钟和选定的FOSR： $f_{\text{notch}} = f_{\text{sampling}} / \text{FOSR}$ 。降低FOSR会导致陷波频率之间的步长增大。

在一些应用中，人们可以利用陷波频率来更好地抑制噪声源，并将能量集中在特定频率。例如，如果应用场合对50 Hz电源频率及其谐波敏感，建议调整滤波器，使第一个陷波频率等于50 Hz。利用这种配置和滤波器调谐，可以在相对高噪声的环境中检测到非常低的信号。

图32. Sinc滤波器频率特性形状



输出数据分辨率

Sinc滤波器操作（滑动平均）的结果是增加采样信号的分辨率（以FOSR为因数增加）。多重平均可以进一步提高分辨率。随后得到的输出信号总分辨率（以LSB为单位）为：

$$\text{分辨率}_{\text{out}} = \text{分辨率}_{\text{in}} * \text{FOSR}^{\text{FORD}}。$$

- 分辨率_{in}是指输入数据的分辨率（在串行数据输入的情况下为2，在并行数据输入的情况下更宽，例如，在12位并行输入时为4096）。
- 务必留意，不能将分辨率增加超过32位，因为滤波器以32位分辨率执行所有内部操作。

输出数据频率

Sinc滤波器操作（滑动平均）会相对于输入采样率降低输出数据速率。连续信号转换的输出数据速率仅取决于FOSR：

$$\text{数据速率}_{\text{out}} = f_{\text{sampling}} / \text{FOSR} \quad (\text{等于第一个} f_{\text{notch}} \text{频率})。$$

由于数据速率_{out}仅取决于FOSR，而分辨率_{out}取决于FOSR和FORD，因此建议首先调整FOSR以达到所需的输出数据速率，然后调整FORD以达到所需的输出分辨率。

延迟时间

延迟表示从启动滤波器（具有有效输入样本）到得到第一个有效输出样本之间经过的时间。在启动滤波器（第一个样本）之后以及恢复滤波器处理之后（如果输入样本不连续，且使用触发器重新启动转换），必须考虑延迟。滤波器包含定时逻辑链（加法器和寄存器，滑动平均原理），需要使用先前输入样本的有效数据进行初始化。如果输入样本流程已停止，输入数据的历史记录丢失，则必须等待滤波器逻辑重新填充有效数据，输出数据才会再次变得有意义。滤波器延迟由下式给出：

$$T_{\text{latency}} = T_{\text{sampling}} * [(\text{FOSR} * \text{FORD}) + (\text{FORD} + 1)]$$

注：理论上，最小延迟是 $T_{\text{latency}} = T_{\text{sampling}} * (\text{FOSR} * \text{FORD})$ ，但由于滤波器的实现和优化，需要更多的周期，请参见公式中的术语（FORD + 1）。

模拟

[\[TUTORIAL\]](#)中的数字滤波器模拟允许：

- 定义输入信号：带符号的1位数据流或带符号的16位数据流
- 配置滤波器参数（FOSR、FORD和IOSR）
- 计算最终输出信号并验证滤波器参数的影响。[图 18：滤波模拟](#)提供了一个模拟结果示例。

5.4.2 积分器

积分器具有一个配置参数，即IOSR（积分器过采样率）。积分器处理是指从滤波器输出中累加（求和）IOSR连续样本。此操作还会降低输出数据速率（以FOSR为因数降低）。

设置IOSR需要考虑的要求是：

- 所需的输出数据速率
- 从滤波器生成一个积分器输出样本所需的样本数。

从[TUTORIAL]提取模拟结果的示例请参见图 18：滤波模拟。

5.5 模拟看门狗

嵌入在DFSDM外设内的模拟看门狗的配置参数请参见表 3。

表3. DFSDM模拟看门狗参数

参数	说明
AWLT	模拟看门狗低阈值。有符号的24位格式。
AWHT	模拟看门狗高阈值。有符号的24位格式。
AWFSEL	模拟看门狗快速模式选择。 0：模拟看门狗监控输出最终的24位数据（在数字滤波器之后） 1：模拟看门狗根据输入的1位串行数据样本观察数据 ⁽¹⁾

1. 当AWFSEL=1时，串行输入数据由专用模拟看门狗滤波器滤波，该滤波器的配置应与主数字滤波器相同：
 - AWFORD：模拟看门狗滤波阶数
 - AWFOSR = 模拟看门狗过采样率
 由于减少了模拟看门狗滤波器选项（AWFOSR=1...32，AWFORD=1...3），此滤波器信号的最大分辨率为16位。此时，高（AWHT[23:0]）和低（AWLT[23:0]）阈值电平的分辨率限制为高16位（AWHT[23:8]，AWLT[23:8]）。

每个输入通道都配有自身的比较器，用于比较模拟看门狗数据（来自模拟看门狗滤波器）和模拟看门狗阈值（AWHT/AWLT）。当选择多个通道（DFSDM_FLTxCR2寄存器中的AWDCH[]）时，可以同时接收多个比较请求。在这种情况下，首先管理编号最小的通道请求，然后继续管理编号较大的所选通道。可以使用不同的标志记录每个通道的结果（DFSDM_FLTxAWSR寄存器中的AWHTF[], AWLTF[]）。一次比较需要1个DFSDM时钟周期。为了能够比较所有选定的通道，用户必须根据输入采样时钟速度和DFSDM频率正确配置观察通道数和模拟看门狗滤波器参数。

DFSDM模拟看门狗可以产生中断和刹车信号。在软件干预下，中断可用于触发CPU。刹车信号（BKAWH[3:0]）用于与其他外设的快速硬件交互（例如，在电机应用中，通过停止定时器，停止其产生PWM）。

5.6 短路检测器

每个通道都有一个短路检测器，可通过SCDEN位使能。使用SCDT[7:0]为每个通道配置短路检测器阈值（范围0-255）。

短路检测器具有向上计数的计数器，可统计串行数据接收器输出上连续的0或1。每次在数据流中检测到逻辑转换（从1到0或从0到1）时，计数器都会重新启动。如果计数器达到SCDT[7:0]中存储的值，则调用短路事件。

以下2个小节将介绍短路检测器功能的使用类型。

真正的短路检测

此时，我们的目标是，在超过允许的溢出状态持续时间的时间段内，检测DFSDM输入信号处于溢出状态的情况（连续超过外部 $\Sigma\Delta$ 调制器满量程范围规范）。

根据应用要求，短路状态的最大持续时间定义为 t_{shortmax} 。将此时间转换为连续0或1的最大数量，以配置SCDT[7:0]：

$$\text{SCDT}[7:0] = t_{\text{shortmax}} \times f_{\text{sampling}}$$

每当短路检测器在SCDT[7:0]位期间检测到输入信号稳定时，将报告错误（输入信号在最大允许时间内处于饱和状态）。

短路检测处理

有两种方法可处理短路检测事件：

- 软件中断产生。微控制器应用软件（中断程序）应对给定的短路事件执行所需的操作（反馈）（例如，在电机控制应用中停止PWM生成）。
- 刹车信号。可以在没有任何软件干预的情况下，将通过BKSCD[3:0]字段配置的刹车信号传输到其他外设（例如，刹车信号可用于控制定时器外设，在电机控制应用中停止PWM生成）。

5.7 脉冲跳跃

DFSDM可用于波束整形应用，其中，可根据入射角与传感器阵列选择输入信号源的位置。该原理如 [第 4.4.2 节：支持波束整形](#) 中所述。波束整形通常与音频应用中的数字麦克风阵列结合使用。

在配置用于波束整形的DFSDM时，需要计算应用于阵列中各个麦克风的相应延迟（参见 [图 29：波束整形原理](#)）。根据首选的接收角度和输入采样频率，计算每个麦克风的延迟，并将其转换为串行输入信号采样频率的多个时钟脉冲。在配置延迟之前，必须同步开始接收从相应DFSDM通道到每个麦克风的输入位流，这可以通过DFSDM全局使能信号（DFSDMEN）来处理。在开始DFSDM转换后，必须在每个通道的每个相应PLSSKP[5:0]字段中，对计算得到的每个麦克风通道的延迟进行一次编程。配置PLSSKP[5:0]字段后，软件应回读PLSSKP[5:0]字段，检查数值是否已复位为0，才能确定已按所需延迟处理DFSDM的最终输出数据。

在初始配置之后，可以在不停止音频数据接收的情况下再次更新延迟，只需累加已经编程的延迟即可。软件必须记住连续写入每个麦克风通道的每个PLSSKP[5:0]字段，并将所有连续写入的累积值视作有效应用的延迟。

示例：

假设已经同步启动了3个麦克风通道，且使用与首选角度 φ 相应的延迟值0、5和10对PLSSKP[5:0]字段进行编程。如果首选角度更改为 $-\varphi$ ，则应使用相应的值20、10、0，在PLSSKP[5:0]字段中执行第二次写入，得到对应通道20、15、10的累积总延迟（这是确定首选角度的延迟之间的差值，而不是延迟绝对值）。

如果DFSDM停止（来自麦克风的所有连续流都停止），则在新的DFSDM同步启动（DFSDMEN = 1）时，所有麦克风的累积延迟将复位为零。

5.8 使用[TUTORIAL]进行配置

[TUTORIAL]提供了一个可配置的模拟器，它基于本应用笔记中的原理和公式，以及DFSDM外设的模型。

为了便于解释，此处假设DFSDM模拟器的接口是标准ADC。用户可以根据特定的模数转换应用来配置输入信号的特性和DFSDM参数（采样频率、模式选择、滤波器参数.....）。用户可以将输入参数收敛到预期的最终转换属性（输出数据速率和分辨率），或使用模拟器来验证预定义配置。第二种方法（更具确定性）涉及初步分析，包括借助本应用笔记，将应用要求转换为连接至DFSDM架构的数字信号处理场景。分析并非总是直截了当，用户有时可能需要考虑和权衡（如分辨率与数据速率）。

图33. [TUTORIAL]中的DFSDM配置

T36						
1	A	B	C	D	I	J
2		DFSDM calculator				
3						
4		Input parameters (conversion)	unit	option		
5		Serial/parallel mode		serial		
6		Sampling frequency	[Hz]	3 500 000		
7		Parallel data resolution	[+/-bits]	12		
8		Continuous mode		no		
9		Regular/injected mode		injected		
10		Number of scanned channels		8		
11		Trigger		Timer		
12		Trigger frequency	[Hz]	64 000		
13		Filter order (FORD)		5		
14		Filter oversampling ratio (FOSR)		9		
15		Integrator oversampling ratio (IOSR)		1		
16		Right bit shift	[bits]	1		
17		Offset correction		0		
18						
19						
20						
21						
22		Output parameters (conversion)	unit	option		
23		Max output data rate	[samples/s]	68 627.45		
24		Real output data rate (trigger)	[samples/s]	64 000.00		
25		Real output data rate per channel	[samples/s]	8 000.00		
26		Internal resolution	[+/- range]	59 049		
27		Internal resolution	[bits]	16.85		
28		Internal data overflow alert!		OK		
29		Trigger overflow alert!		OK		
30		Output resolution	[+/- range]	29 525		
31		Output bits resolution	[bits]	15.85		
32		Output bits effective resolution	[bits]	15.85		
33		Output data overflow alert!		OK		
34						

6 结论

DFSDM外设嵌入了先进（有时复杂）的功能，这些功能在涉及模数转换的多种应用中非常实用和有益。本文档介绍了DFSDM的多种不同应用领域，使用此外设的优势和配置示例，以及指南。[\[TUTORIAL\]](#)允许快速试验各种DFSDM参数的影响，并验证软件架构选择和相关配置是否能够满足应用要求。理解DFSDM处理的基本原理以及如何使用适当的工具和方法实现这些原理（本文档的目标）是确保成功实现基于DFSDM应用的关键因素。

7 版本历史

表4. 文档版本历史

日期	版本	变更
2018 年 3 月 30 日	1	初始版本。

表5. 中文文档版本历史

日期	版本	变更
2018 年11月21 日	1	中文初始版本。

重要通知 - 请仔细阅读

意法半导体公司及其子公司（“ST”）保留随时对 ST 产品和 / 或本文档进行变更、更正、增强、修改和改进的权利，恕不另行通知。买方在订货之前应获取关于 ST 产品的最新信息。ST 产品的销售依照订单确认时的相关 ST 销售条款。

买方自行负责对 ST 产品的选择和使用，ST 概不承担与应用协助或买方产品设计相关的任何责任。

ST 不对任何知识产权进行任何明示或默示的授权或许可。

转售的 ST 产品如有不同于此处提供的信息的规定，将导致 ST 针对该产品授予的任何保证失效。

ST 和 ST 徽标是 ST 的商标。所有其他产品或服务名称均为其各自所有者的财产。

本文档中的信息取代本文档所有早期版本中提供的信息。本文档的中文版本为英文版本的翻译件，仅供参考之用；若中文版本与英文版本有任何冲突或不一致，则以英文版本为准。

© 2018 STMicroelectronics - 保留所有权利