

---

## 如何在STM32F413/423系列器件上同步DFSDM滤波器 以及编程脉冲跳频

---

### 前言

STM32F413/423系列微控制器在DFSDM（Sigma-Delta调制器的数字滤波器）的顶层实现了专门的机制，可支持以下功能：

- 同步两个DFSDM滤波器
- 脉冲跳频

脉冲跳频是指器件可以动态调整每个麦克风的PDM（脉冲密度调制）延迟，而不必增加任何外部延迟线。使用脉冲跳频功能的音频应用包括波束成形和声源定位。

本应用笔记的目的是描述如何在STM32F413/423系列器件上同步DFSDM滤波器以及如何编程脉冲跳频。

本应用笔记详细描述：

- 如何配置脉冲跳频机制来将音频时钟和数据注入DFSDM输入
- 如何配置TIM3和TIM4定时器来生成请求的脉冲以延迟一个或多个PDM流
- 如何构建一个序列来同步两个DFSDM滤波器
- 如何构建一个序列来生成脉冲

本应用笔记假设读者熟悉STM32 MCU的DFSDM和TIMER，如意法半导体的网站[www.st.com](http://www.st.com)提供的STM32F413/423基于ARM®的高级32位MCU参考手册（RM0430）中所述。

### 相关固件

- 用于STM32F4系列产品的STM32CubeF4嵌入式软件

# 目录

<b>1</b>	<b>在DFSDM顶层实现的STM32F413/423系列机制概述</b>	<b>5</b>
<b>2</b>	<b>DFSDM滤波器同步配置</b>	<b>7</b>
2.1	DFSDM滤波器同步的一般性描述	7
2.2	DFSDM滤波器同步的编程序列	7
<b>3</b>	<b>脉冲跳频配置</b>	<b>9</b>
3.1	脉冲跳频的一般性描述	9
3.2	脉冲跳频的编程序列	10
3.3	波束成形用例中的脉冲跳频	12
<b>4</b>	<b>专用于此机制的STM32CubeF4固件API函数</b>	<b>15</b>
4.1	MCHDLY（多通道延迟）配置	15
4.1.1	HAL DFSDM函数：void HAL_DFSDM_BitstreamClock_Start(void)	15
4.1.2	HAL DFSDM函数：void HAL_DFSDM_BitstreamClock_Stop(void)	15
4.1.3	HAL DFSDM函数：void HAL_DFSDM_DisableDelayClock (uint32_t MCHDLY)	15
4.1.4	HAL DFSDM函数：void HAL_DFSDM_EnableDelayClock (uint32_t MCHDLY)	15
4.1.5	HAL DFSDM函数：void HAL_DFSDM_ConfigMultiChannelDelay (DFSDM_MultiChannelConfigTypeDef* mchdlystruct)	16
4.2	同步DFSDM滤波器	17
4.3	脉冲跳频	18
4.3.1	脉冲跳频函数：void Pulse_Skipper_Init(void)	18
4.3.2	脉冲跳频函数：void Pulse_Skipper_Bitstream_Start(void)	18
4.3.3	脉冲跳频函数：void Pulse_Skipper_Bitstream_Stop(void)	18
4.3.4	脉冲跳频函数：void Pulse_Skipper_Generate_Pulse (PulseSkipper_InitTypeDef* skipperstruct)	18
<b>5</b>	<b>结论</b>	<b>20</b>
<b>6</b>	<b>相关术语</b>	<b>21</b>
<b>7</b>	<b>版本历史</b>	<b>22</b>

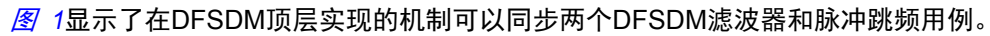
## 表格索引

表1.	文档版本历史 . . . . .	22
表2.	中文文档版本历史 . . . . .	22

## 图片索引

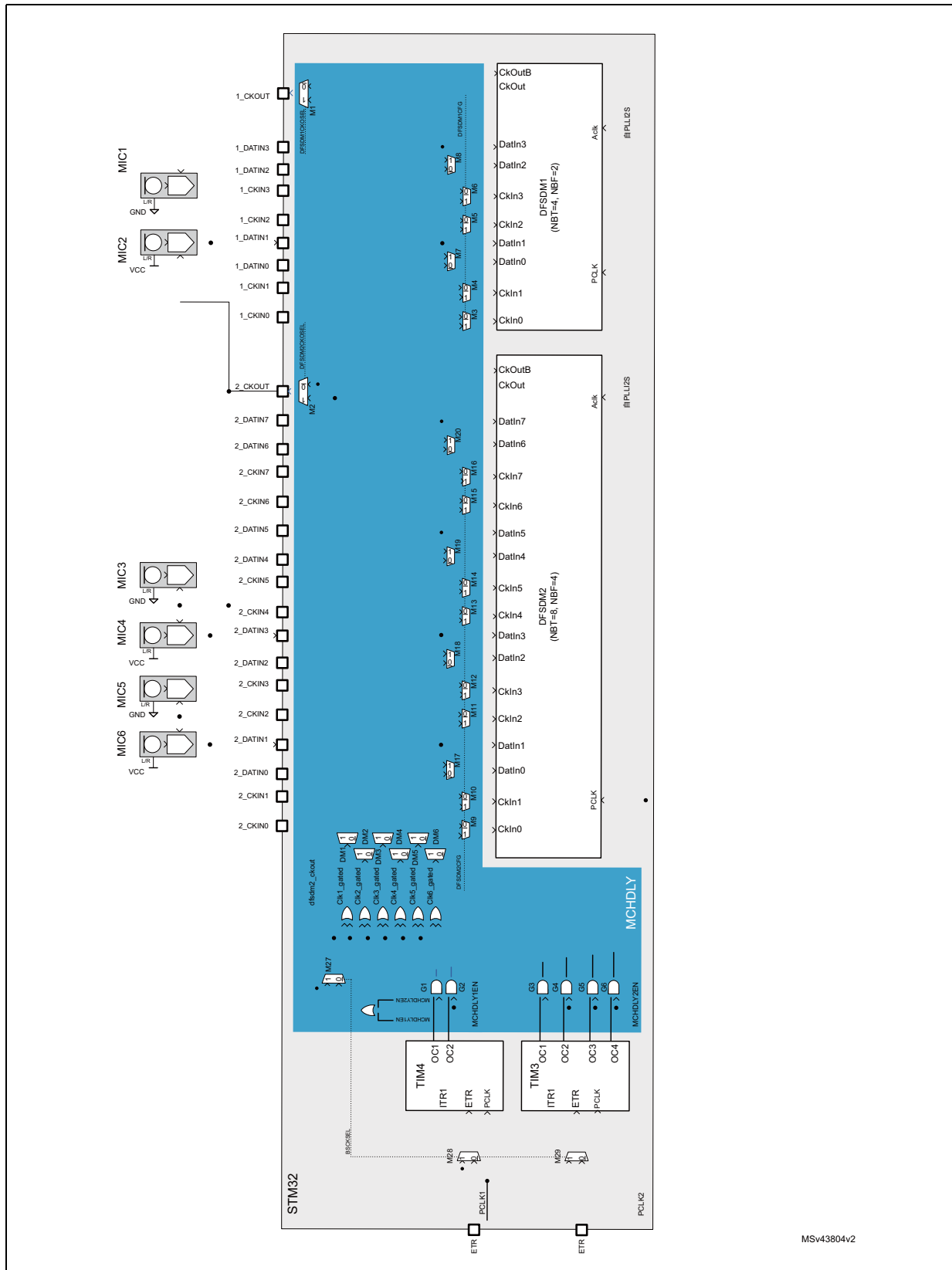
图1.	在DFSDM顶层实现的机制 .....	6
图2.	时钟门控概念 .....	10
图3.	波束成形用例 .....	13
图4.	将两个麦克风设置为1 PCM延迟所需的距离 .....	14

# 1 在DFSDM顶层实现的STM32F413/423系列机制概述

图 1显示了在DFSDM顶层实现的机制可以同步两个DFSDM滤波器和脉冲跳频用例。

如图 1所示，MCHDLY（多通道延迟）模块既是DFSDM滤波器同步的一部分，也是脉冲跳频用例的一部分。对于脉冲跳频用例，TIM3和TIM4定时器也是实现机制的一部分。

图1. 在DFSDM顶层实现的机制



MSv43804v2

## 2 DFSDM滤波器同步配置

### 2.1 DFSDM滤波器同步的一般性描述

STM32F413/423系列器件有两个DFSDM：

- DFSDM1：包括四个通道和两个滤波器
- DFSDM2：包括八个通道和四个滤波器。

每个DFSDM都可以与第一个DFSDM滤波器（DFSDM\_FLT0）同步开始转换，但有些情况需要同步DFSDM1和DFSDM2滤波器。

为同步DFSDM1和DFSDM2滤波器，用户必须遵循以下步骤：

1. 使用外部时钟源和DFSDM音频源作为输入时钟
2. 禁用DFSDM2时钟输出（dfsdm2\_ckout）
3. 对每个DFSDM进行编程
4. 开始滤波器转换
5. 启用DFSDM2时钟输出（dfsdm2\_ckout）

通过执行上述步骤，可以同步开始两个DFSDM滤波器的转换。

### 2.2 DFSDM滤波器同步的编程序列

要遵循的序列主要取决于MCHDLY模块，如下所示：

1. 禁用DFSDM2时钟输出（dfsdm2\_ckout）信号（SYSCFG\_MCHDLYCR.BSCKSEL设置为0 **M27**）
2. 启用外部DFSDM输出时钟
3. 配置DFSDM以选择外部CKINx作为输入时钟
4. 配置DFSDM以从相同通道的引脚获取通道数据
5. 输出DFSDM2外部输出时钟（dfsdm2\_ckout）至DFSDMx CKOUT焊盘（可选）：SYSCFG\_MCHDLYCR.DFSDM1CKOSEL (**M1**) 和SYSCFG\_MCHDLYCR.DFSDM2CKOSEL (**M2**) 设置为1
6. 设置DFSDM2时钟输出（dfsdm2\_ckout）路径为CKINx，具体取决于以下使用的DATINx：
  - a) 配置以下DMx开关
    - DFSDM1：SYSCFG\_MCHDLYCR.DFSDM1\_CK02SEL设置为0以注入DFSDM2时钟输出（dfsdm2\_ckout）至CKIN0 (**DM2**)
    - DFSDM1：SYSCFG\_MCHDLYCR.DFSDM1\_CK02SEL设置为1以注入DFSDM2时钟输出（dfsdm2\_ckout）至CKIN2 (**DM2**)
    - DFSDM1：SYSCFG\_MCHDLYCR.DFSDM1\_CK13SEL设置为0以注入DFSDM2时钟输出（dfsdm2\_ckout）至CKIN1 (**DM1**)
    - DFSDM1：SYSCFG\_MCHDLYCR.DFSDM1\_CK13SEL设置为1以注入DFSDM2时钟输出（dfsdm2\_ckout）至CKIN3 (**DM1**)

- DFSDM2: SYSCFG\_MCHDLYCR.DFSDM2\_CK04SEL设置为0以注入DFSDM2时钟输出 (dfsdm2\_ckout) 至CKIN0 (**DM6**)
  - DFSDM2: SYSCFG\_MCHDLYCR.DFSDM2\_CK04SEL设置为1以注入DFSDM2时钟输出 (dfsdm2\_ckout) 至CKIN4 (**DM6**)
  - DFSDM2: SYSCFG\_MCHDLYCR.DFSDM2\_CK15SEL设置为0以注入DFSDM2时钟输出 (dfsdm2\_ckout) 至CKIN1 (**DM5**)
  - DFSDM2: SYSCFG\_MCHDLYCR.DFSDM2\_CK15SEL设置为1以注入DFSDM2时钟输出 (dfsdm2\_ckout) 至CKIN5 (**DM5**)
  - DFSDM2: SYSCFG\_MCHDLYCR.DFSDM2\_CK26SEL设置为0以注入DFSDM2时钟输出 (dfsdm2\_ckout) 至CKIN2 (**DM4**)
  - DFSDM2: SYSCFG\_MCHDLYCR.DFSDM2\_CK26SEL设置为1以注入DFSDM2时钟输出 (dfsdm2\_ckout) 至CKIN6 (**DM4**)
  - DFSDM2: SYSCFG\_MCHDLYCR.DFSDM2\_CK37SEL设置为0以注入DFSDM2时钟输出 (dfsdm2\_ckout) 至CKIN3 (**DM3**)
  - DFSDM2: SYSCFG\_MCHDLYCR.DFSDM2\_CK37SEL设置为1以注入DFSDM2时钟输出 (dfsdm2\_ckout) 至CKIN7 (**DM3**)
- b) 配置以下Mx开关
- DFSDM1: SYSCFG\_MCHDLYCRDFSDM1CFG设置为1以注入DFSDM2时钟输出 (dfsdm2\_ckout) 至CKIN0、CKIN1、CKIN2和CKIN3 (**M3、M4、M5、M6**)
  - DFSDM2: SYSCFG\_MCHDLYCRDFSDM2CFG设置为1以注入DFSDM2时钟输出 (dfsdm2\_ckout) 至CKIN0、CKIN1、CKIN2、CKIN3、CKIN4、CKIN5、CKIN6和CKIN7 (**M9、M10、M11、M12、M13、M14、M15、M16**)
7. 开始转换所有使用的DFSDM1和DFSDM2滤波器
8. 释放DFSDM2时钟输出 (dfsdm2\_ckout) 信号 (SYSCFG\_MCHDLYCR.BSCKSEL设置为1 **M27**)

请注意，对于DFSDM1，DFSDM2时钟输出 (dfsdm2\_ckout) 可以注入：

- CKIN0或CKIN2
- CKIN1或CKIN3

对于DFSDM2，DFSDM2时钟输出 (dfsdm2\_ckout) 可以注入：

- CKIN0或CKIN4
- CKIN1或CKIN5
- CKIN2或CKIN6
- CKIN3或CKIN7



## 3 脉冲跳频配置

### 3.1 脉冲跳频的一般性描述

脉冲跳频的目的在于实现指定输入通道的延迟线等特性。脉冲跳频使输入串行数据流（仅串行流）中指定数量的样品在进入滤波器前被丢弃。该数据丢弃通过跳过指定数量的采样输入时钟脉冲（指定的串行数据样品未被滤波器采样）来执行。

采样时钟由脉冲跳频功能门控，以获得指定数量的时钟脉冲。指定数量的时钟脉冲被跳过时，输入数据的滤波会继续。与未跳过的数据流相比，该操作导致来自滤波器的最终输出样品（及下一个样品）从后面的输入数据计算得出。

由于是从较新的输入样品而不是“未跳过”样品计算得出，因此，最终的样品看起来是向前移动的。最终的“跳过样品”将在稍后转换，因为跳过的输入数据样品必须由随后的输入数据样品替换。

由于两个数据缓冲区都相移了，因此，最终的数据缓冲区特性（跳过和未跳过的输出数据缓冲区比较）就好像未跳过的数据流由于某种原因被延迟了。

时钟跳过的实现基于MCHDLY、TIM3和TIM4模块。MCHDLY模块由MCHDLYCR寄存器控制，所有MCHDLY控制位都如此。

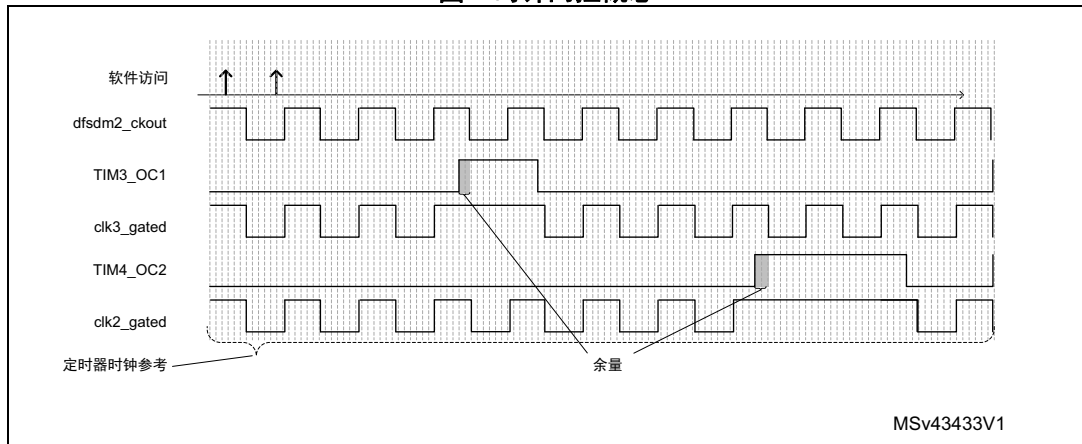
如 [图 1](#) 所示，通过使用MCHDLY模块，DFSDM可用于多达六个数字麦克风的音频应用。数字麦克风（或Sigma-Delta调制器）的时钟由DFSDM2时钟输出（dfsdm2\_ckout信号）提供。然后，该时钟输出信号被分配给：

- OR门，以通过时钟屏蔽或门控实现脉冲跳过
- TIM4和TIM3定时器的ETR（外部触发）输入，以定义必须跳过的脉冲数量，并用于同步
- 通过M1和M2多路复用器的DFSDM1\_CKOUT和DFSDM2\_CKOUT引脚输出，以在DFSDMx引脚上生成输出时钟信号

OR门用于跳过为DFSDM提供的输入串行时钟脉冲，以便在相应的输入通道上产生延迟。该时钟门控由两个定时器控制（TIM4和TIM3）。定时器以单触发模式编程，以产生具有限定长度的屏蔽脉冲，从而门控所需的时钟脉冲数。

[图 2](#) 显示了在DFSDM2的输入通道3或7（取决于所选择的输入通道）上产生一个DFSDM时钟周期延迟的情况。它还显示了在DFSDM1的输入通道0或2（取决于所选择的输入通道）上产生的两个DFSDM时钟周期另一个延迟。

图2. 时钟门控概念



MSv43433V1

### 3.2 脉冲跳频的编程序列

本节介绍在指定的输入通道上生成脉冲跳频应遵循的序列。

#### 配置MCDLY模块（如果使用的输入通道位于两个不同的DFSDM上，也可以同步DFSDM滤波器）

1. 禁用DFSDM2时钟输出（dfsdm2\_ckout）信号（SYSCFG\_MCHDLYCR.BSCKSEL设置为0 **M27**）
2. 启用外部DFSDM输出时钟
3. 配置DFSDM以选择外部CKINx作为输入时钟
4. 配置DFSDM以从相同通道的引脚获取通道数据
5. 输出DFSDM2外部输出时钟（dfsdm2\_ckout）至DFSDMx CKOUT焊盘：  
SYSCFG\_MCHDLYCR.DFSDM1CKOSEL (**M1**) 和SYSCFG\_MCHDLYCR.DFSDM2CKOSEL (**M2**) 设置为1

*请注意：在图 1 中，MIC1 通过可以从 DFSDM2 或 DFSDM1 CKOUT 信号提供时钟输出的另一个时钟输出引脚（DFSDM1\_CKOUT 引脚）接收其串行时钟。此配置仅可用于一个麦克风（MIC1）的低功耗用例，同时禁用 DFSDM2\_CKOUT 并启用 DFSDM1\_CKOUT（如用于语音检测），并将 MIC1 重新用于波束成形用例（启用 DFSDM2\_CKOUT 时）。*

6. 设置DFSDM2时钟输出（dfsdm2\_ckout）路径为CKINx，具体取决于以下使用的DATINx通道：
  - a) 配置以下DMx开关
    - DFSDM1: SYSCFG\_MCHDLYCR.DFSDM1\_CK02SEL设置为0以注入DFSDM2时钟输出（dfsdm2\_ckout）至CKIN0 (**DM2**)
    - DFSDM1: SYSCFG\_MCHDLYCR.DFSDM1\_CK02SEL设置为1以注入DFSDM2时钟输出（dfsdm2\_ckout）至CKIN2 (**DM2**)
    - DFSDM1: SYSCFG\_MCHDLYCR.DFSDM1\_CK13SEL设置为0以注入DFSDM2时钟输出（dfsdm2\_ckout）至CKIN1 (**DM1**)

- DFSDM1: SYSCFG\_MCHDLYCR.DFSDM1\_CK13SEL设置为1以注入DFSDM2时钟输出 (dfsdm2\_ckout) 至CKIN3 (**DM1**)
  - DFSDM2: SYSCFG\_MCHDLYCR.DFSDM2\_CK04SEL设置为0以注入DFSDM2时钟输出 (dfsdm2\_ckout) 至CKIN0 (**DM6**)
  - DFSDM2: SYSCFG\_MCHDLYCR.DFSDM2\_CK04SEL设置为1以注入DFSDM2时钟输出 (dfsdm2\_ckout) 至CKIN4 (**DM6**)
  - DFSDM2: SYSCFG\_MCHDLYCR.DFSDM2\_CK15SEL设置为0以注入DFSDM2时钟输出 (dfsdm2\_ckout) 至CKIN1 (**DM5**)
  - DFSDM2: SYSCFG\_MCHDLYCR.DFSDM2\_CK15SEL设置为1以注入DFSDM2时钟输出 (dfsdm2\_ckout) 至CKIN5 (**DM5**)
  - DFSDM2: SYSCFG\_MCHDLYCR.DFSDM2\_CK26SEL设置为0以注入DFSDM2时钟输出 (dfsdm2\_ckout) 至CKIN2 (**DM4**)
  - DFSDM2: SYSCFG\_MCHDLYCR.DFSDM2\_CK26SEL设置为1以注入DFSDM2时钟输出 (dfsdm2\_ckout) 至CKIN6 (**DM4**)
  - DFSDM2: SYSCFG\_MCHDLYCR.DFSDM2\_CK37SEL设置为0以注入DFSDM2时钟输出 (dfsdm2\_ckout) 至CKIN3 (**DM3**)
  - DFSDM2: SYSCFG\_MCHDLYCR.DFSDM2\_CK37SEL设置为1以注入DFSDM2时钟输出 (dfsdm2\_ckout) 至CKIN7 (**DM3**)
- b) 配置以下Mx开关
- DFSDM1: SYSCFG\_MCHDLYCRDFSDM1CFG设置为1以注入DFSDM2时钟输出 (dfsdm2\_ckout) 至CKIN0、CKIN1、CKIN2和CKIN3 (**M3、M4、M5、M6**)
  - DFSDM2: SYSCFG\_MCHDLYCRDFSDM2CFG设置为1以注入DFSDM2时钟输出 (dfsdm2\_ckout) 至CKIN0、CKIN1、CKIN2、CKIN3、CKIN4、CKIN5、CKIN6和CKIN7 (**M9、M10、M11、M12、M13、M14、M15、M16**)
7. 开始转换所有使用的DFSDM1和DFSDM2滤波器
  8. 释放DFSDM2时钟输出 (dfsdm2\_ckout) 信号 (SYSCFG\_MCHDLYCR.BSCKSEL设置为1 **M27**)

### 配置TIM3和TIM4，以便在指定的输入通道上生成脉冲跳频

1. 不需要门控时，连接到DFSDM1 CKIN的OR门上的TIM4输出 (OC[2:1]) 和连接到DFSDM2 CKIN的OR门上的TIM3输出 (OC[4:1]) 必须处于非活动模式下的低电平
2. 配置OPM（单脉冲模式）中的定时器。在该模式下，定时器可以启动计数器以响应于ETR输入的上升沿，并产生具有可编程长度的脉冲。该脉冲用于由DFSDM2产生比特流时钟时。

注：定时器使用的时钟参考可以是APB时钟或APB时钟乘以2或4。该频率越高越好。

使用TIM3和TIM4门控dfsdm2\_ckout音频时钟时，定时器参考时钟频率必须至少为DFSDM2音频时钟频率的12倍（假设比特流时钟占空比为50%）。

注：由于TIM2 OC1已连接到M27以备份DFSDM2时钟输出 (dfsdm2\_ckout) 信号，因此，其不可使用。

### 启用DFSDM1和DFSDM2的时钟门控

1. 启用DFSDM1的时钟门控：SYSCFG\_MCHDLYCR.MCHDLY1EN设置为1（G1和G2），使TIM4产生的脉冲注入相应的DFSDM1 CKINx
2. 启用DFSDM2的时钟门控：SYSCFG\_MCHDLYCR.MCHDLY2EN设置为1（G3、G4、G5和G6），使TIM3产生的脉冲注入相应的DFSDM2 CKINx

**必须发出脉冲跳频时，相应OR门的TIM3或TIM4 OC通道上的应用程序产生所需长度的脉冲。**

一次只能产生一个脉冲，因此，该应用必须等当前产生的脉冲结束才能发出下一个（如需要）。

## 3.3 波束成形用例中的脉冲跳频

[图 3](#)显示了波束成形用例的完整视图，以及在该用例中脉冲跳频何时适用。

在编程的PCM（脉冲编码调制）采样率设置为16 KHz的情况下，为避免CPU处理，麦克风距离必须设置为21 mm，这相当于两个麦克风之间有1 PCM的延迟。

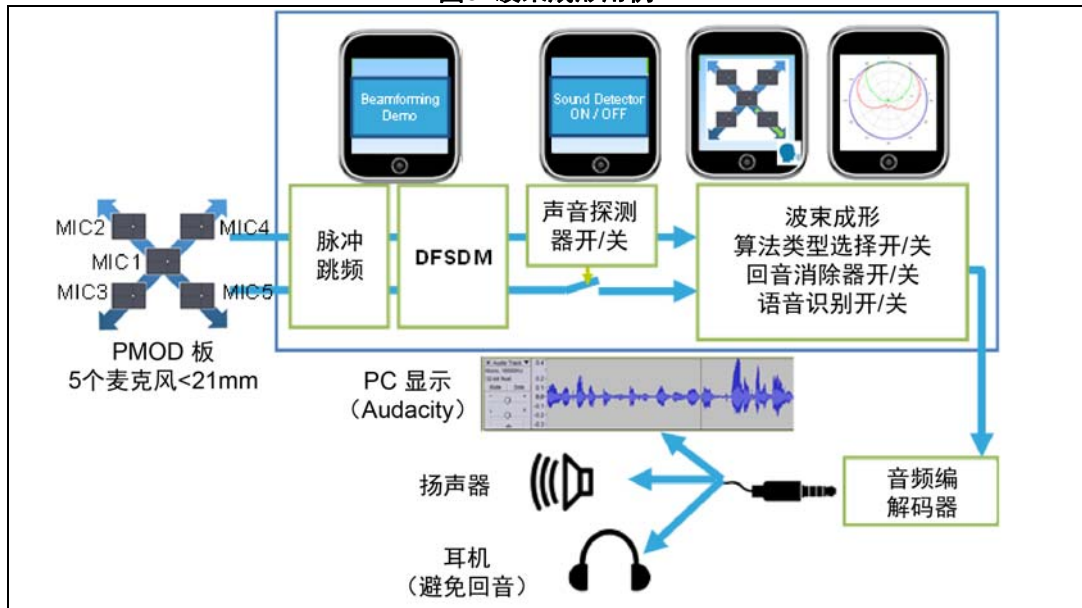
脉冲跳频功能可以释放两个麦克风之间的距离限制，并虚拟设置两个麦克风之间的一个或多个PCM延迟的距离。

在[图 3](#)中，Pmod™板包含五个与DFSDM连接的麦克风。麦克风之间的距离（M1-M2、M1-M3、M1-M4、M1-M5）小于21 mm。

波束成形用例的GUI（图形用户界面）可以让用户选择应用波束成形的方向（四个可能的方向）。前面的操作定义了要使用的两个MIC（例如MIC1和MIC2）。

由于选择的两个麦克风之间的距离小于21 mm，因此，脉冲跳频模块用于虚拟地增加该距离。此操作可以达到21 mm的距离，这相当于运行该用例所需的MIC1和MIC2之间的1 PCM延迟（见[图 3](#)）。

图3. 波束成形用例



如要计算使用的脉冲跳频周期，必须使用以下公式：

$$NB\_BITSTREAM\_CLOCK\_PERIOD = (D_{pcm} - D_{m12}) / (T \times c)$$

其中：

- $D_{pcm}$  = 相当于1 PCM延迟的两个麦克风之间的预期虚拟距离（PCM采样率设置为16KHz的距离为21mm）
- $D_{m12}$  = 两个麦克风之间的实际距离
- $T$  = DFSDM2音频时钟周期（例如 $0.510E-6$  s，假设麦克风频率= 2.048 MHz）
- $c$  = 声速340 m/s

图 4显示了将两个麦克风的距离设置为1 PCM延迟所需的距离。

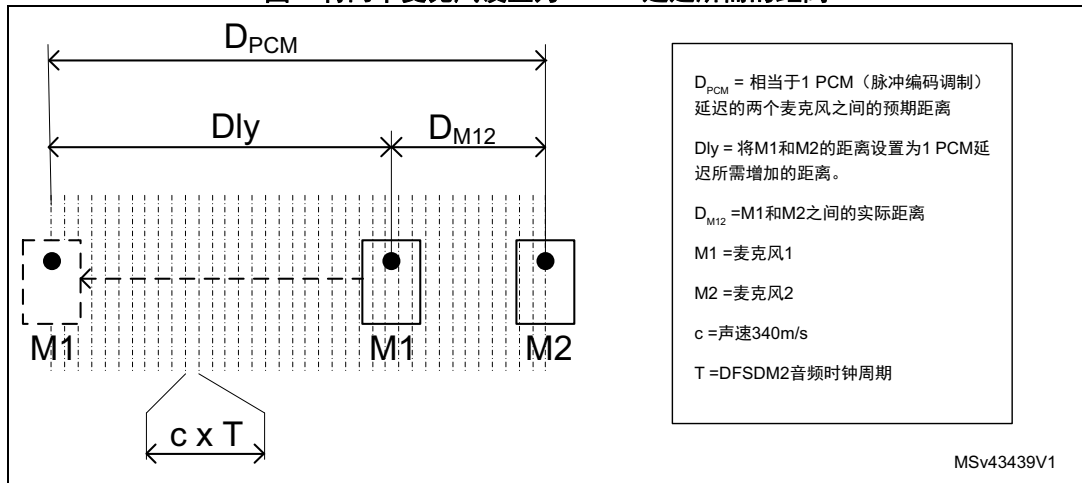
例如：

- PCM采样率设置为16 KHz：  $D_{pcm} = 21\text{mm}$ （1 PCM延迟）
- $D_{m12} = 5\text{ mm}$ （M1和M2之间的实际距离）
- PDM（脉冲密度调制）采样率设置为2.048Mhz ==>  $T = 0.510E-6\text{s}$

$$NB\_BITSTREAM\_CLOCK\_PERIOD = (21-5).10E-5 / (340 \times 0.510E-6) = 94$$

如图 2所示，该数字相当于进行门控（TIM OC脉冲持续时间）的dfsdm时钟周期。

图4. 将两个麦克风设置为1 PCM延迟所需的距离



## 4 专用于此机制的STM32CubeF4固件API函数

作为STM32CubeF4固件v1.15.0的一部分，提供了解释如何生成脉冲跳频的示例，请参见：  
STM32Cube\_FW\_F4\_V1.15.0\Projects\STM32F413H-Discovery\Examples\DFSDM。

### 4.1 MCHDLY（多通道延迟）配置

本节介绍DFSDM驱动器stm32f4xx\_hal\_dfstm.c的部分函数，以及用于配置和控制MCHDLY模块的函数。

#### 4.1.1 HAL DFSDM函数：void HAL\_DFSDM\_BitstreamClock\_Start(void)

- **函数：** void HAL\_DFSDM\_BitstreamClock\_Start(void)
- **描述：** 如果将DFSDM2时钟输出（dfsdm2\_ckout）配置为PulseSkipper\_InitTypeDef结构（参见第 4.1.5 节中的结构体定义）的一部分，则将其分配至OR门和DFSDM1/2音频输出（DFSDMx CKOUT）
- **MCHDLY影响：** SYSCFG\_MCHDLYCR.BSCKSEL设置为1 - M27。

#### 4.1.2 HAL DFSDM函数：void HAL\_DFSDM\_BitstreamClock\_Stop(void)

- **函数：** void HAL\_DFSDM\_BitstreamClock\_Stop(void)
- **描述：** 如果将DFSDM2时钟输出（dfsdm2\_ckout）配置为PulseSkipper\_InitTypeDef结构（参见第 4.1.5 节中的结构体定义）的一部分，则阻止将其分配至OR门和DFSDM1/2音频输出（DFSDMx CKOUT）
- **MCHDLY影响：** SYSCFG\_MCHDLYCR.BSCKSEL设置为0 - M27。

#### 4.1.3 HAL DFSDM函数：void HAL\_DFSDM\_DisableDelayClock (uint32\_t MCHDLY)

- **函数：** void HAL\_DFSDM\_DisableDelayClock(uint32\_t MCHDLY)
- **描述：** 阻止分配DFSDM2时钟输出（dfsdm2\_ckout）至DFSDM1/2 CkInx
- **MCHDLY影响：** 对于DFSDM1，将SYSCFG\_MCHDLYCR.MCHDLY1EN设置为0（G1和G2）和/或对于DFSDM2，将SYSCFG\_MCHDLYCR.MCHDLY2EN设置为0（G3、G4、G5和G6）

#### 4.1.4 HAL DFSDM函数：void HAL\_DFSDM\_EnableDelayClock (uint32\_t MCHDLY)

- **函数：** void HAL\_DFSDM\_EnableDelayClock(uint32\_t MCHDLY)
- **描述：** 分配DFSDM2时钟输出（dfsdm2\_ckout）至DFSDM1/2 CkInx
- **MCHDLY影响：** 对于DFSDM1，将SYSCFG\_MCHDLYCR.MCHDLY1EN设置为1（G1和G2）和/或对于DFSDM2，将SYSCFG\_MCHDLYCR.MCHDLY2EN设置为1（G3、G4、G5和G6）



#### 4.1.5 HAL DFSDM函数：void HAL\_DFSDM\_ConfigMultiChannelDelay (DFSDM\_MultiChannelConfigTypeDef\* mchdlystruct)

- **函数：** void  
HAL\_DFSDM\_ConfigMultiChannelDelay(DFSDM\_MultiChannelConfigTypeDef\* mchdlystruct)
- **描述：** 基于DFSDM\_MultiChannelConfigTypeDef结构的内容，用于通过配置以下多路复用或解复用：**M1、M2、M3、M4、M5、M6、M7、M8、DM1、DM2、DM3、DM4、DM5、DM6、M9、M10、M11、M12、M13、M14、M15、M16、M17、M18、M19、M20**来将DFSDM2时钟输出（dfsdm2\_ckout）的路径设置为DFSDM1/2 CkInx和数据输入通道
- **MCHDLY 影响：**
  - **DFSDM1：** SYSCFG\_MCHDLYCR.DFSDM1CKOSEL (**M1**) 、  
SYSCFG\_MCHDLYCR.DFSDM1CFG (**M3、M4、M5、M6**) 、  
SYSCFG\_MCHDLYCR.DFSDM1CK13SEL (**DM1**) 、  
SYSCFG\_MCHDLYCR.DFSDM1CK02SEL (**DM2**) 、  
SYSCFG\_MCHDLYCR.DFSDM1D0SEL (**M7**) 、  
SYSCFG\_MCHDLYCR.DFSDM1D2SEL (**M8**)
  - **DFSDM2：** SYSCFG\_MCHDLYCR.DFSDM2CKOSEL (**M2**) 、  
SYSCFG\_MCHDLYCR.DFSDM2CFG (**M9、M10、M11、M12、M13、M14、M15、M16**) 、  
SYSCFG\_MCHDLYCR.DFSDM2CK37SEL (**DM3**) 、  
SYSCFG\_MCHDLYCR.DFSDM2CK26SEL (**DM4**) 、  
SYSCFG\_MCHDLYCR.DFSDM2CK15SEL (**DM5**) 、  
SYSCFG\_MCHDLYCR.DFSDM2CK04SEL (**DM6**) 、  
SYSCFG\_MCHDLYCR.DFSDM2D0SEL (**M17**) 、  
SYSCFG\_MCHDLYCR.DFSDM2D2SEL (**M18**) 、  
SYSCFG\_MCHDLYCR.DFSDM2D4SEL (**M19**) 、  
SYSCFG\_MCHDLYCR.DFSDM2D6SEL (**M20**)

- **结构体定义：**

```
typedef struct
{
    uint32_t DFSDM1ClockIn;      /*!< Source selection for DFSDM1_Ckin.
    This parameter can be a value of @ref DFSDM1_CLOCKIN_SELECTION*/
    uint32_t DFSDM2ClockIn;      /*!< Source selection for DFSDM2_Ckin.
    This parameter can be a value of @ref DFSDM2_CLOCKIN_SELECTION*/
    uint32_t DFSDM1ClockOut;     /*!< Source selection for DFSDM1_Ckout.
    This parameter can be a value of @ref DFSDM1_CLOCKOUT_SELECTION*/
    uint32_t DFSDM2ClockOut;     /*!< Source selection for DFSDM2_Ckout.
    This parameter can be a value of @ref DFSDM2_CLOCKOUT_SELECTION*/
    uint32_t DFSDM1BitClkDistribution; /*!< Distribution of the DFSDM1 bitstream clock gated by TIM4
    OC1 or TIM4 OC2.
    This parameter can be a value of @ref DFSDM1_BIT_STREAM_DISTRIBUTION
    @note The DFSDM2 audio gated by TIM4 OC2 can be injected on CKIN0 or CKIN2
    @note The DFSDM2 audio gated by TIM4 OC1 can be injected on CKIN1 or CKIN3 */
    uint32_t DFSDM2BitClkDistribution; /*!< Distribution of the DFSDM2 bitstream clock gated by TIM3
    OC1 or TIM3 OC2 or TIM3 OC3 or TIM3 OC4.
```



This parameter can be a value of @ref DFSDM2\_BIT\_STREAM\_DISTRIBUTION

@note The DFSDM2 audio gated by TIM3 OC4 can be injected on CKIN0 or CKIN4

@note The DFSDM2 audio gated by TIM3 OC3 can be injected on CKIN1 or CKIN5

@note The DFSDM2 audio gated by TIM3 OC2 can be injected on CKIN2 or CKIN6

@note The DFSDM2 audio gated by TIM3 OC1 can be injected on CKIN3 or CKIN7 \*/

```
uint32_t DFSDM1DataDistribution; /*!< Source selection for DatIn0 and DatIn2 of DFSDM1.
```

This parameter can be a value of @ref DFSDM1\_DATA\_DISTRIBUTION \*/

```
uint32_t DFSDM2DataDistribution; /*!< Source selection for DatIn0, DatIn2, DatIn4 and DatIn6 of DFSDM2.
```

This parameter can be a value of @ref DFSDM2\_DATA\_DISTRIBUTION \*/

```
}DFSDM_MultiChannelConfigTypeDef;
```

- **DFSDM1ClockIn**: 用于选择DFSDM1 CkInx源, DFSDM2时钟输出 (dfsdm2\_ckout) 或CkInx PAD (**M3、M4、M5、M6**) 的外部音频时钟
- **DFSDM2ClockIn**用于选择DFSDM2 CkInx源, DFSDM2时钟输出 (dfsdm2\_ckout) 或CkInx PAD (**M9、M10、M11、M12、M13、M14、M15、M16**) 的外部音频时钟
- **DFSDM1ClockOut**: 用于选择DFSDM1时钟输出, DFSDM2音频时钟或 DFSDM1 CkOut (**M1**)
- **DFSDM2ClockOut**: 用于选择DFSDM2时钟输出, DFSDM2音频时钟或 DFSDM2 CkOut (**M2**)
- **DFSDM1BitClkDistribution**: 用于分配DFSDM2时钟音频至 DFSDM1 CkIn0或CkIn2 (**DM1**) 和/或至DFSDM1 CkIn1或CkIn3 (**DM2**)
- **DFSDM2BitClkDistribution**: 用于分配DFSDM2时钟音频至DFSDM2 CkIn3 或CkIn7 (**DM3**) 和/或至DFSDM2 CkIn2或CkIn6 (**DM4**) 和/或至DFSDM2 CkIn1 或CkIn5 (**DM5**) 和/或至DFSDM2 CkIn0或CkIn4 (**DM6**)
- **DFSDM1DataDistribution**: 用于注入DFSDM1DataIn0或DataIn1PAD至DFSDM1 DatIn0输入 (**M7**) 和/或DFSDM1 DataIn2或DataIn3 PAD至DFSDM1 DatIn2输入 (**M8**)
- **DFSDM2DataDistribution**: 用于注入DFSDM2DataIn0或DataIn1PAD至DFSDM2 DatIn0输入 (**M17**) 和/或DFSDM2 DataIn2或DataIn3 PAD至DFSDM2 DatIn2输入 (**M18**) 和/或DFSDM2 DataIn4或DataIn5 PAD至DFSDM2 DatIn4输入 (**M19**) 和/或DFSDM2 DataIn6或DataIn7 PAD至DFSDM2 DatIn6输入 (**M20**)

## 4.2 同步DFSDM滤波器

同步DFSDM滤波器没有特定函数。第 4.1 节: [MCHDLY \(多通道延迟\) 配置](#)中描述的HAL DFSDM函数应按照第 2.2 节: [DFSDM滤波器同步的编程序列](#)中的描述使用和排序。

## 4.3 脉冲跳频

MCHDLY应按照第4.1节：*MCHDLY（多通道延迟）配置*中的描述进行编程，以配置DFSDM2时钟输出（dfsdm2\_ckout）路径。

本节介绍建议的STM32F413/423探索套件固件（正在准备中，将于2017年初发布）的四个其他函数。

### 4.3.1 脉冲跳频函数：void Pulse\_Skipper\_Init(void)

- **函数：**void Pulse\_Skipper\_Init(void)
- **描述：**配置OPM（单脉冲模式）中的TIM3和TIM4。产生的脉冲的长度是pulse\_skipper.c文件的#define NB\_BITSTREAM\_CLOCK\_PERIOD部分，并作为TIM3和TIM4定时器初始化函数的参数传递。然后，DFSDM1和DFSDM2时钟门控启用（G1至G7），换句话说，DFSDM2时钟输出（dfsdm2\_ckout）信号注入到所有已使用的CkIN通道上。

### 4.3.2 脉冲跳频函数：void Pulse\_Skipper\_Bitstream\_Start(void)

- **函数：**void Pulse\_Skipper\_Bitstream\_Start(void)
- **描述：**释放DFSDM2时钟输出（dfsdm2\_ckout）信号至所有OR门（M27）。

### 4.3.3 脉冲跳频函数：void Pulse\_Skipper\_Bitstream\_Stop(void)

- **函数：**void Pulse\_Skipper\_Bitstream\_Stop(void)
- **描述：**阻止DFSDM2时钟输出（dfsdm2\_ckout）信号至所有OR门（M27）。

### 4.3.4 脉冲跳频函数：void Pulse\_Skipper\_Generate\_Pulse (PulseSkipper\_InitTypeDef\* skipperstruct)

- **函数：**void Pulse\_Skipper\_Generate\_Pulse(PulseSkipper\_InitTypeDef\* skipperstruct)
- **描述：**从选定的TIM3或TIM4输出比较通道产生具有初始化脉冲长度的脉冲（#define NB\_BITSTREAM\_CLOCK\_PERIOD）。发出新脉冲之前，必须完成先前产生的脉冲
- **结构体定义：**

```
Structuretypedef struct
{
    uint32_t DFSDM1PulseSkipperCh; /*!< Channels selection to generate pulse skipper of DFSDM1.
```

This parameter can be a value of @ref DFSDM1\_PULSESKIPPER\_CH \*/  
uint32\_t DFSDM2PulseSkipperCh; /\*!< Channels selection to generate pulse skipper of DFSDM2.  
This parameter can be a value of @ref DFSDM2\_PULSESKIPPER\_CH \*/  
}PulseSkipper\_InitTypeDef;

- **DFSDM1PulseSkipperCh**: 用于通过DFSDM1BitClkDistribution, 在DFSDM1通道0、通道1、通道2或通道3上产生脉冲, 选择为MCHDLY配置的一部分。
- **DFSDM2PulseSkipperCh**: 用于通过DFSDM2BitClkDistribution, 在DFSDM2通道0、通道1、通道2、通道3、通道4、通道5、通道6或通道7上产生脉冲, 选择为MCHDLY配置的一部分。

注: 对于DFSDM1: 可在通道0或通道2、通道1或通道3上产生脉冲。

对于DFSDM2: 可在通道0或通道4、通道1或通道5、通道2或通道6、通道3或通道5上产生脉冲。

注: 一次只能产生一个脉冲, 因此, 该应用必须等脉冲结束才能产生另一个。如果必须更新脉冲长度, TIM3或TIM4定时器必须重新初始化新的脉冲长度 (更新#define NB\_BITSTREAM\_CLOCK\_PERIOD)。

## 5 结论

本应用笔记介绍了如何配置在DFSDM顶层实现的机制，以便：

- 同步两个DFSDM滤波器
- 通过动态地调整每个麦克风的PDM延迟来应用脉冲跳频用例，无需增加外部延迟线（使用脉冲跳频功能的音频应用包括波束成形和声源定位）。

[第 2 节：DFSDM 滤波器同步配置](#)和 [第 3 节：脉冲跳频配置](#)中介绍了相关模块（MCHDLY、TIM3 和 TIM4）的配置以及应遵循的序列。

[第 4 节：专用于此机制的STM32CubeF4固件API函数](#)中介绍了专用于此机制的STM32CubeF4 固件API函数。作为STM32CubeF4 FWv1.15.0的一部分，提供了产生脉冲跳频的编码示例，请参见：STM32Cube\_FW\_F4\_V1.15.0\Projects\STM32F413H-Discovery\Examples\DFSDM。

## 6 相关术语

- PDM：脉冲密度调制。数字麦克风（或其他类型的传感器）正在提供编码到PDM中的数据。这些数据的速率由DFSDM时钟输出控制。数字麦克风的PDM样品常用速率在1和3 MHz之间。
- PCM：脉冲编码调制。从数字麦克风（或其他传感器）接收的PDM编码数据由DFSDM进行滤波和抽取。所得样品是经PCM编码的，并且提供的速率取决于抽取比率和DFSDM时钟输出频率。PCM样品的常用采样率为16 KHz。
- DFSDM1时钟输出：来自DFSDM音频时钟或系统时钟的外部DFSDM1时钟输出。
- DFSDM2时钟输出：来自DFSDM音频时钟或系统时钟的外部DFSDM2时钟输出。

## 7 版本历史

表1. 文档版本历史

日期	版本	变更
2016年12月19日	1	初始版本。

表2. 中文文档版本历史

日期	版本	变更
2017年10月25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。

© 2017 STMicroelectronics - 保留所有权利