

目录

前言.....

一、 DAC 主要特性.....

 1.1 双通道模式.....

 1.2 数据格式.....

 1.3 触发选择.....

 1.4 DMA 功能.....

 1.4.1 单通道 DMA.....

 1.4.2 双通道 DMA.....

 1.4.3 DMA 下溢错误.....

 1.5 白噪声发生器.....

 1.5.1 定义.....

 1.5.2 典型应用.....

 1.6 三角波发生器.....

 1.6.1 定义.....

 1.6.2 典型应用.....

 1.7 缓冲的输出.....

二、 使用 DAC 生成正弦波形.....

 2.1 说明.....

 2.2 生成正弦波原理.....

 2.3 生成正弦波数据表.....

 2.4 正弦波频率计算.....

联系我们.....

前言

本应用笔记举例介绍了使用数模转换器（DAC）外设生成各类输出信号的过程，该 DAC 外设内嵌在 ACM32F4xx，ACM32FP4xx 微控制器系列产品中。

数模转换器（DAC）是一种与模数转换器功能相反的器件，可以将数字形式的数据转换为相应的模拟电压信号。

ACM32F4 和 ACM32FP4xx 系列芯片的 DAC 模块是 12 位字转换器，带有两个输出通道，可分别输出目标电压，白噪声，三角波，正弦波等信号。

DAC 可用于多种音频应用中，例如：安全警报、蓝牙耳机、发声玩具、答录机、人机接口以及低成本的音乐播放器

ACM32 DAC 还可实现许多其他模拟用途，如模拟波形产生和控制工程。

本应用笔记主要包括两部分内容：

- 第 1 部分介绍 DAC 模块的主要特性。
- 第 2 部分介绍 DAC 用于生成正弦波形的示例。

一、DAC 主要特性

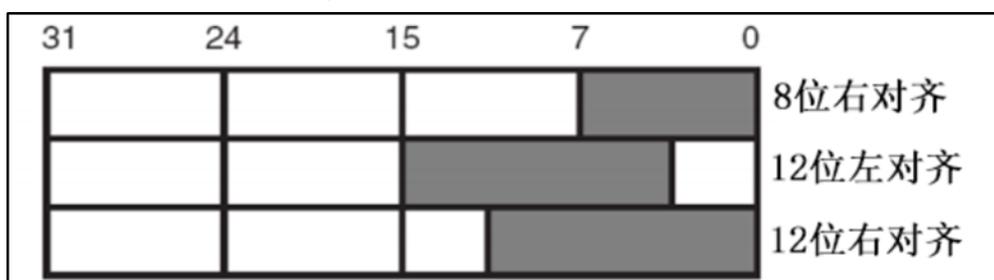
1.1 双通道模式

DAC 有两个输出通道，每个通道各有一个转换器。在双 DAC 通道模式下，转换可以单独进行，也可以同时进行。当 DAC 通道由同一个触发源触发后，两个通道将组合在一起同步执行更新操作，转换也会同时进行。

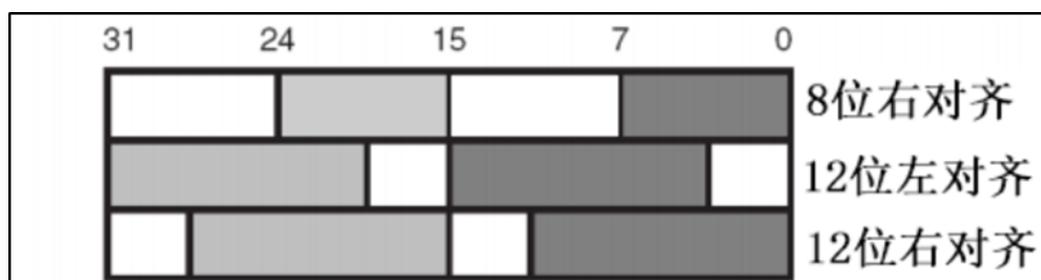
1.2 数据格式

DAC 可以使用以下三种整型格式的数据：8 位右对齐、12 位右对齐以及 12 位左对齐。12 位值的范围在 0x000 到 0xFFF 之间，其中 0x000 为最小值，而 0xFFF 为最大值。8 位值范围在 0x00 到 0xFF 之间，三种数据格式在数据寄存器中的储存格式如图。

单 DAC 通道模式的数据寄存器



双 DAC 通道模式的数据寄存器



1.3 触发选择

DAC 转换可以由某外部事件触发(定时器计数器、外部管脚信号)以及软件触发，

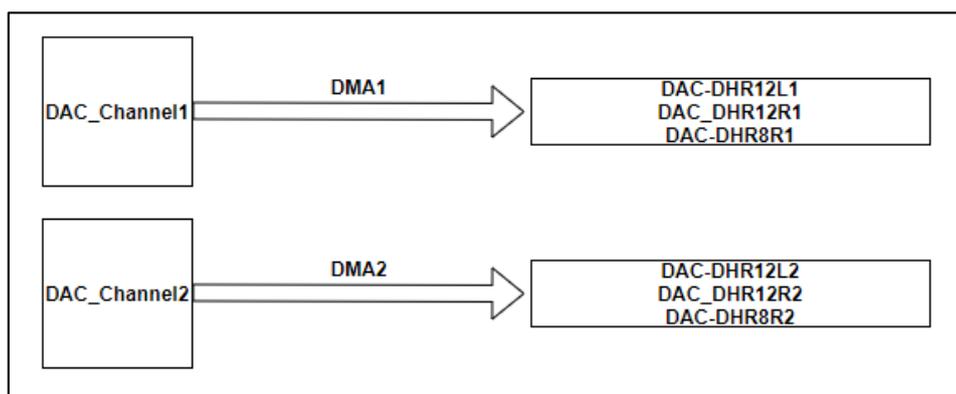
当 DAC 选择外部事件触发时，通道检测到来自选中的定时器 TRGO 或者 EXTI9 触发信号，存放在数据寄存器中的值会自动传入输出寄存器，开始转换。

当 DAC 选择软件触发时，置位软件触发标志，存放在数据寄存器中的值会自动传入输出寄存器，开始转换。

1.4 DMA 功能

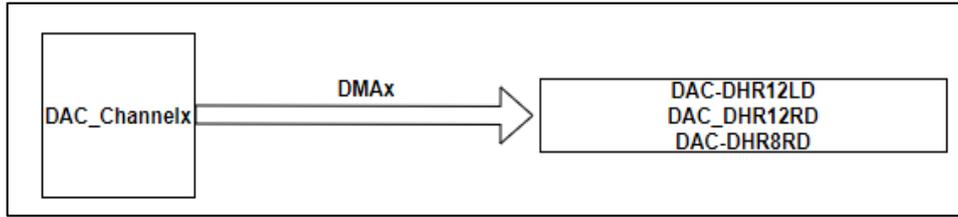
1.4.1 单通道 DMA

每个 DAC 通道都具有 DMA 功能。2 个 DMA 通道可分别用于 2 个 DAC 通道的 DMA 请求。一旦有外部触发(而不是软件触发)发生，则产生一个 DMA 请求,通过 DMA 可将要输出到 DAC 的数据搬运至通道数据寄存器中。在单 DAC 模式下，可使能对应通道的 DMA 请求，完成 DMA 传输，原理如图。



1.4.2 双通道 DMA

而在双 DAC 模式下，如果 2 个通道都使能 DMA，如上图一般，则会产生 2 个 DMA 请求。如果实际只需要一个 DMA 传输，则应只选择使能其中一个 DAC 通道的 DMA，利用一个通道的 DMA 将数据搬运至双通道数据寄存器中。这样，程序可以在只使用一个 DMA 请求，一个 DMA 通道的情况下，处理工作在双 DAC 模式的 2 个 DAC 通道，原理如图。



1.4.3 DMA 下溢错误

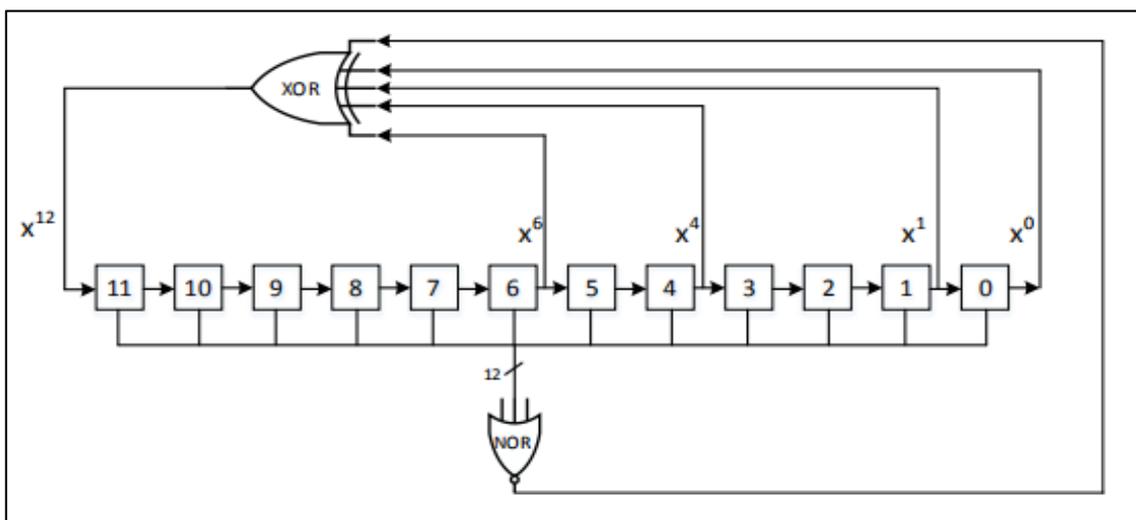
DAC DMA 请求没有缓冲队列，因此，如果第二个外部触发到达时尚未收到第一个外部触发的应答，将不会发出新的请求，并将对应通道的“DMA 下溢错误”标志置 1，DAC 通道仍将继续转换旧的数据。

因此，DMA 向 DAC 提供模式波形时，有时会出现 DMA 传输速度比 DAC 转换速度快快的情况。此时，DAC 会检测到部分模式波形遭到忽略而不予转换。所以在采用 DMA 请求时，应修改 DAC 的触发转换频率或减轻 DMA 的工作负荷，以避免发生下溢错误。

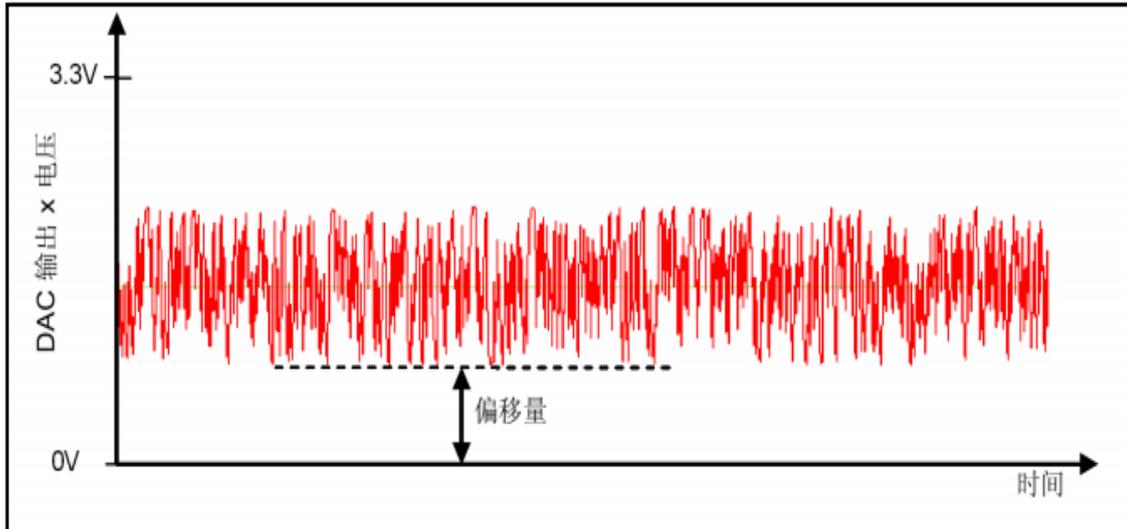
1.5 白噪声发生器

1.5.1 定义

ACM32 微控制器 DAC 为用户提供了一个伪随机码发生器。根据移位寄存器上使用的节拍数，在序列重复前，可生成具有最多 $2^n - 1$ 个数的序列。



由噪声发生器生成的噪声具有均匀的频谱分布，可将这些噪声视为白噪声。不过，白噪声分布均匀，不具备高斯输出特性。



1.5.2 典型应用

白噪声发生器可用于产生电子音乐，通常直接产生或作为滤波器的输入来形成其它类型的

噪声信号。白噪声发生器广泛用于音频合成，通常用于再现钹等打击乐器的效果，这些乐器在其频域内具有较高的噪声量。

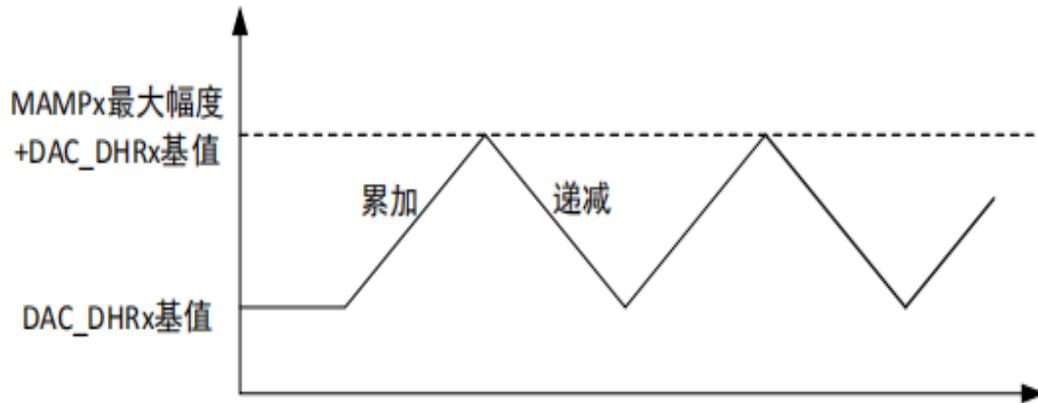
白噪声发生器可用于控制工程，以及放大器和电子滤波器的频率响应测试。

白噪声是一种通用的合成噪声源，用于通过耳鸣掩蔽器进行声掩蔽。

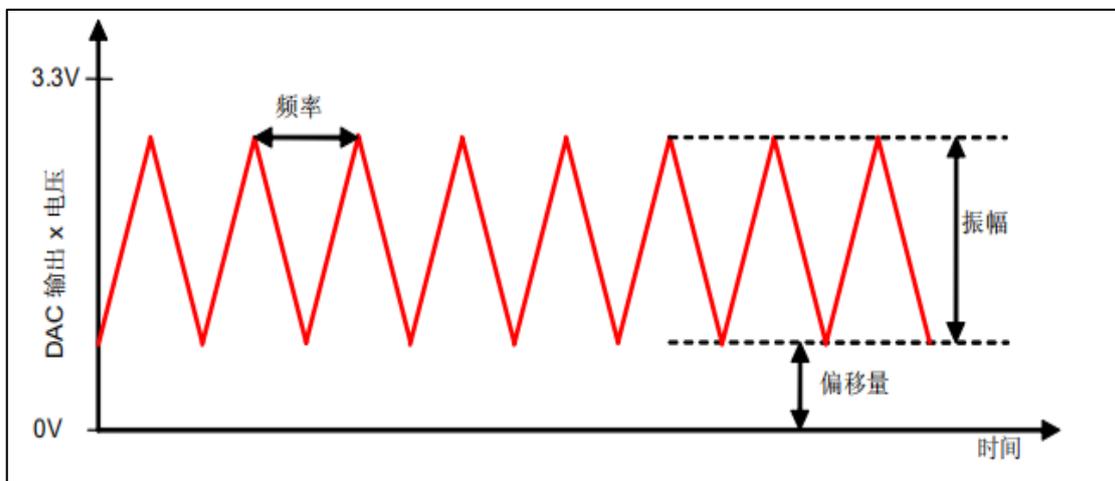
1.6 三角波发生器

1.6.1 定义

ACM32 DAC 为用户提供了具有灵活的偏移量、振幅和频率的三角波形发生器。理论上说，三角波形是一种由无限组奇次谐波组成的波形，可以使用 DAC_CR 寄存器中的 MAMPx 位修正三角波形的振幅。



其中，三角波形频率与触发源的频率相关，三角波的幅值与基值以及 MAMP_x 的设置有关。



1.6.2 典型应用

由于三角波发生器的高次谐波下降速度更快，因此其音色比方波柔和，故而三角波发生器经常用于声音合成。

三角波发生器电路还可用于多种调制解调器电路应用。

1.7 缓冲的输出

为了在不使用外部运算放大器的情况下驱动外部负载，DAC 通道内嵌一个输出缓冲器，可以根据用户应用情况进行使能和禁止。

如果未对 DAC 输出进行缓冲，当用户应用电路中存在负载时，实际电压输出会低于预期电压输出。启用缓冲器后，实际电压输出会与预期电压输出非常接近

二、使用 DAC 生成正弦波形

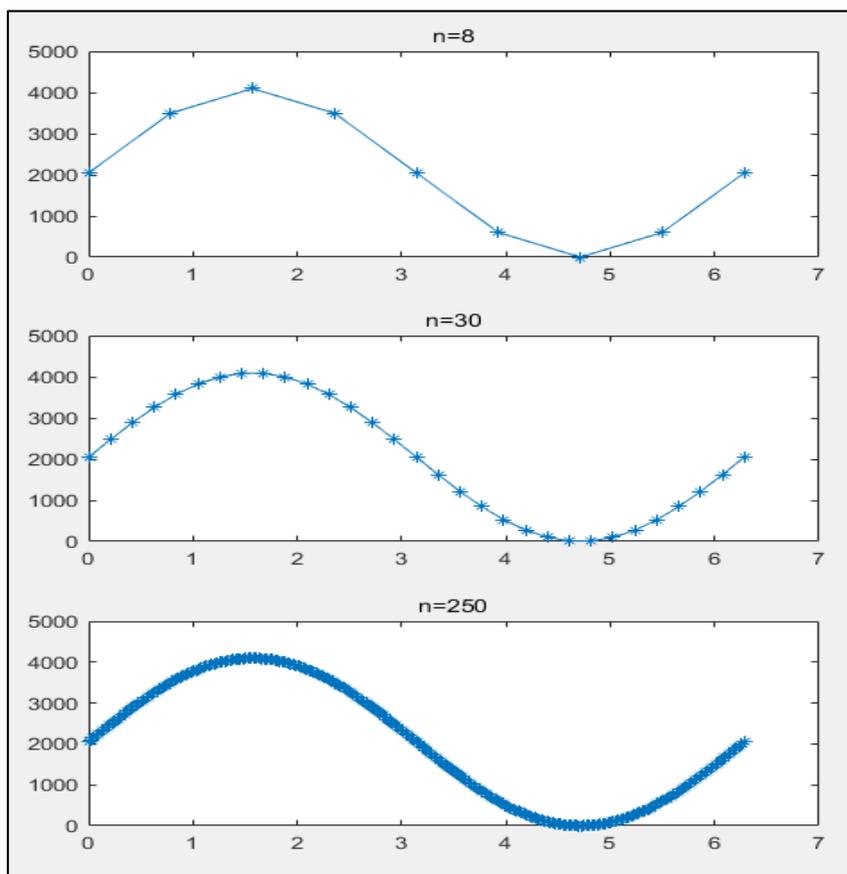
2.1 说明

本示例分步说明如何生成正弦波形。

正弦波形也称作单频正弦波音，它是一种纯音或绝对音。在确定听觉系统的各种响应时，正弦音一般用作刺激源。

2.2 生成正弦波原理

要输出正弦波，实质是要控制 DAC 以 $V=\sin(t)$ 的正弦函数关系输出电压，其中 V 为电压输出， t 为时间。而由于模拟信号连续而数字信号是离散的，所以使用 DAC 产生正弦波时，只能按一定时间间隔输出正弦曲线上的点，在该时间段内输出相同的电压值，若缩短时间间隔，提高单个周期内的输出点数，可以得到逼近连续正弦波的图形，见图，点数越多，越能更好的还原正弦波，若在外电路加适当的电容滤波，可得到更完美的图形。



2.3 生成正弦波数据表

由于正弦曲线是周期函数，所以只需要得到单个周期内的数据后按周期重复即可，而单个周期内取样输出的点数又是有限的，所以为了得到呈 $v=\sin(t)$ 函数关系电压值的数据通常不会实时计算获取，而是预先计算出函数单个周期内的电压数据表，并且转化成以 DAC 寄存器表示的值。

$\sin x$ 函数值的范围为 $[-1: +1]$ ，而 DAC 输出电压范围为 $[0\sim 3.3]V$ ，按 12 位 DAC 分辨率表示的方法，可写入寄存器的最大值为 $0xFF F = 4095$ ，即范围为 $[0:4095]$ 。所以，

实际输出时，需进行如下处理：

- 抬升 \sin 函数的输出为正值： $V = \sin(t)+1$ ，此时， V 的输出范围为 $[0:2]$ ；
- 扩展输出至 DAC 的全电压范围： $V = 3.3*(\sin(t)+1)/2$ ，此时， V 的输出范围为 $[0:3.3]$ ，

正是 DAC 的电压输出范围，扩展至全电压范围可以充分利用 DAC 的分辨率；

- 把电压值[0:3.3]转换为[0,4095]以 DAC 寄存器的形式表示： $Reg_val = 4095 * V / 3.3$ ，此时，存储到 DAC 寄存器的值范围为[0:4095]；
- 实践证明，在 $\sin(t)$ 的单个周期内，至少取 30 个点进行电压输出才能较好地还原正弦波形，所以在 $t \in [0:2\pi]$ 区间内等间距根据上述 Reg_val 公式运算得到 n 个寄存器值，利用 matlab 或 python 等脚本即可得到正弦波表；
- 控制 DAC 输出时，每隔一段相同的时间从上述正弦波表中取出一个新数据进行输出，即可输出正弦波。改变间隔时间的单位长度，或正弦波的样本点数，可以改变正弦波曲线的周期。

采样(n)	数字采样值	模拟采样值（电压）
1	2048	1.65
2	3649	2.94
3	4045	3.25
4	2937	2.36
5	1159	0.93
6	51	0.04
7	447	0.36
8	2048	1.65

2.4 正弦波频率计算

有了 2.2 中的准备，就可以生成正弦波信号了，只需要在一定时间内，循环的将正弦波数据搬运至 DAC 数据寄存器中，即可生成正弦波信号了。

为了更好的控制正弦波的频率，本应用中采用了 Timer+DMA+DAC 输出正弦波的方案，利用定时器得 TRGO 信号触发 DAC 得 DMA 请求搬运正弦波数据，则正弦波的频率跟定时器的 TRGO 触发时间和正弦波样本数量 n 相关，计算方法如下：

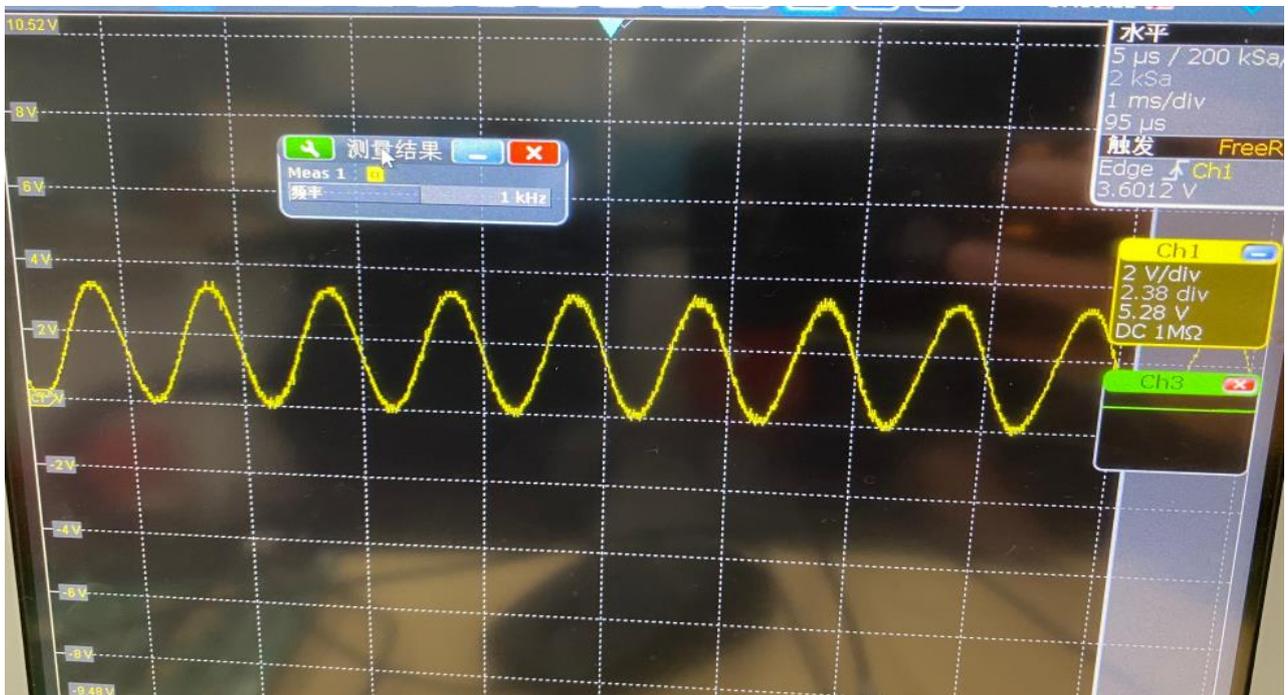
- 如系统时钟 120M，PCLK=60M，TimCLK=PCLK*2=120M

- 配置定时器分频系数 Prescaler, 则分频后的定时器频率=120M/(Prescaler+1)
- 配置定时器的计算值 Period, 则定时器触发 TRGO 信号的时间

$$=(\text{Period}+1)/ (120\text{M}/(\text{Prescaler}+1))$$
- 根据正弦波单个周期的样本数 N, 可得正弦波得频率为:

$$F=120\text{M}/((\text{Prescaler}+1)*(\text{Period}+1)*N)$$

在本应用中, 正弦波样本数 N=250 个, Prescaler=1-1 (不分频), Period=480-1, 故
 $F=120\text{M}/(1*480*250)=1000\text{HZ}$ 。



联系我们

公司：上海爱信诺航芯电子科技有限公司

地址：上海市闵行区合川路 2570 号科技绿洲三期 2 号楼 702 室

邮编：200241

电话：+86-21-6125 9080

传真：+86-21-6125 9080-830

Email: Service@AisinoChip.com

Website: www.aisinochip.com

版本维护

版本	日期	作者	描述
V1.0	2021-01-25	Aisinochip	初始版
V1.1	2021-01-27	Aisinochip	更新了 DAC 双通道 8 位右对齐数据格式插图以及问你描述错误； 修改了 DAC 生成正弦波的数据计算公式以及正弦波频率计算公式

本文档的所有部分，其著作权归上海爱信诺航芯电子科技有限公司（简称航芯公司）所有，未经航芯公司授权许可，任何个人及组织不得复制、转载、仿制本文档的全部或部分组件。本文档没有任何形式的担保、立场表达或其他暗示，若有任何因本文档或其中提及的产品所有资讯所引起的直接或间接损失，航芯公司及所属员工恕不为其担保任何责任。除此以外，本文档所提到的产品规格及资讯仅供参考，内容亦会随时更新，恕不另行通知。