信号发生器

1. 产生正弦波新号，频率可调：0.5Hz~10KHz，幅度可调：100mV~3V
2. 产生方波信号，频率可调：0.5Hz~1KHz，幅度可调：100mV~3V
3. 产生矩形波，频率可调：0.5Hz~1KHz, 幅度固定为3V，占空比可调：10%~90%

**一、思路**

使用一预先定义的缓冲区buffer存储正弦波一个周期的幅值，缓冲区长度定义为4096.输出正弦波时直接通过一个索引值i在缓冲区里查找并输出，这样可以节省大量的计算时间。然后通过定时器中断周期性的输出。

如何改变频率？我们可以改变索引值i的每次递增量(定义为Incval)，比如频率1k时我们让i每次加1，则一个周期输出1000个点，当频率2k时，i每次加2，输出500个点。

如何改变幅度？在定义缓冲区时我们可以定义好缓冲区的单位幅值代表多少正弦波的多少幅度，比如缓冲区中的1代表10mV。通过每次向缓冲区乘以一定的数值即可改变幅度。

改变频率的具体计算：我们设定在f<=100HZ时输出全部的1000个点，在10kHZ时输出10个点。所以当f=100HZ时，递增连i=1，f=10kHZ时，递增量i=100.所以假设频率为f，则递增量i = f / 100。

当频率小于100时，我们还是通过改变索引值i的递增量来改变频率，不过此时递增量可以是小数，比如Incval=0.9表示索引值i每次递增0.9，但是索引值只能是整数，所以我们再对i取整，这样做的效果就是缓冲区的1000个点中会有某几个点是重复输出的，这是因为递增量为小数。

幅度可调具体方法：由于DAC为12位，输出范围为0~3.3V，分辨率为3.3V/4096。缓冲区buffer的最大值我们定义为4095这是正弦函数输出为1的那点。然后当改变幅度时，比如幅度为a时，我们输出要乘以的系数为(a/3.3)\*4096，假设当前索引值为i，则取出的值为buffer[i]，则DAC输出(a/3.3)\*4096\*buffer[i].

**二、遇到的问题**

遇到的问题：上述方法是通过定时器中断实现的，当所需的正弦波频率较低时是可以的，一旦所需频率高了比如10k时，定时器中断所花费的时间就很多，在实验中我们发现当定时器中断周期设置为0.01ms时，定时器中断所花费的时间基本已经占据了整个程序的执行时间导致我们的外部中断都无法响应，所以此种方法也被我们舍弃了。

1. **改进策略**

考虑到是由于定时器中断频率过高导致超过cpu所能承载的上限，所以我们使用另一种高速传输方法：DAC+DMA传输。

DMA传输不同于定时器中断传输，这种方法直接建立存储器与外设的连接，通过DMA控制器直接把正弦波的数据从存储器送往DAC，速度非常快，使得实现高频的正弦波信号成为可能。

DAC+DMA原理：在使用DMA传输之前，我们需要预先准备好要传送的数据，然后DAC向DMA发送DMA请求，发送请求后需要向DMA发送要传输数据的首地址以及要传输数据的长度，然后DMA开始执行传输。

在STM32中，还需要使用定时器来执行DMA传输，使用定时器是因为在执行DMA传输时，定时器需要跳变一次，DAC才发送一次数据，所以我们也需要设置定时器的频率来达到我们想要的波形的频率，比如当前定时器频率为30k，波形一个周期为100个点，那么波形的频率为30k/100=300HZ

DMA传输中遇到的问题：我们使用下面这个函数执行DMA传输：

HAL\_DAC\_Start\_DMA(DAC\_HandleTypeDef \*hdac, uint32\_t Channel, uint32\_t \*pData, uint32\_t Length, uint32\_t Alignment);

在实验中我们把DMA传输放在主循环中执行，我们通过外部中断的方式改变频率以及幅值，然后在外部中断响应函数中重新计算波形数组以及所需要的点数。由于我们采用这种方式导致我们的波形在调节频率时有一定概率会出现变形的情况。

我们分析这是由于当程序在主循环执行HAL\_DAC\_Start\_DMA()输出函数时我们通过外部中断打断了这个程序的执行，而且还在外部中断里修改了与这个程序执行有关的某些参数的值，导致在中断结束后返回这个程序的时候会导致一些错误。

所以我们不在外部中断想响应程序中改变任何与HAL\_DAC\_Start\_DMA()执行有关的参数值，我们会在HAL\_DAC\_Start\_DMA()函数执行完后判断之前是否检测到由外部中断来临，然后再更改参数值。

所以本次实验启示我们在使用DMA传输时，要等到DMA一次传输完成之后再更改参数值，否则会出现意想不到的错误

1. **实验总结**

在完成信号发生器的实验中，我们总结了一下几点：

1. 在实验开始前做详细的实验文档

不管做什么项目，在开始项目之前我们都应该对这个项目做一个整体的分析，并给出一个初步的框架和思路。

然后把项目拆成一系列小的部分，然后我们以一个小部分为单位去做实验，每次完成一小部分试验后把代码保留下来备份。

比如在这次实验中就遇到在做好的代码上胡乱修改导致出不了波形

就以这次实验为例，我们应该首先把三个波形的生成分解为三个独立的部分，只要完成一个部分，其他两个部分都可以如法炮制很快的做出来。下面以正弦波为例：

在实现正弦波的时候我们可以先使用DMA输出一个特定频率的正弦波，以便验证

我们使用DMA的方式是否正确，这个特定频率的正弦波最好是极值频率，比如在本次 实验中是0.5HZ和10KHZ，此时应该把实现的代码保留下来。不然在后面的实验 中 我们是在此代码上改进的，如果出不来波形，我们就不知道是代码的问题还是示波器 的问题，这样就很耗费时间。

然后我们编写外部中断代码改变频率，并且保留这个阶段成功的代码。然后编写改变幅值的代码并保留程序。然后编写其他另外两个波形的代码。