**项目二报告**

**程序介绍及功能实现**

程序开发逻辑

timecount

改变频率

通过主程序直接控制方波的产生

界面上用按键跳动显示

改变输出总时间数

通过analog discovery控制方波的产生和频率

通过analog的方波界面控制方波的有无

1.直接通过程序产生控制信号：通过设置timecount参数，每当波形产生一次跳变就记一次数，1000毫秒内的总次数的二分之一就是频率。方波则有自带的函数square产生。

2.用analog discovery控制方波信号的输出频率：在analog discovery的控制界面上，第7位bit的值用来控制信号的有无，第0~4位控制频率大小。 通过analog控制Navigator产生特定频率的方波然后再由analog接收。

**功能实现**

下图所示即为软件交互界面，左侧为主画布，用于第0位bit实时显示输出方波的频率。右侧是功能选择界面。大量功能按键或是数据显示框分布其中，下面对其详细介绍。

调节频率：在右侧控制界面上方有一滑动条和输入框，通过拖动滑动条或直接输入可以改变频率。每次拖动滑动条后，可以在上面两个数据框中看到当前的频率。

控制信号：先打开waveform界面，将bit slide的第7位调节成0后，允许信号输出，而后对0~3位进行调整，得到1~15hz的频率值。此后点击控制信号按键，点击开始，即可以在analog中观察到波形。

**问题与解决方案**

频率总是设定值两倍：由于一开始用timecount计算频率是根据方波的跳变计数，所以实际的计数总数的一半才是真正的频率。将1000/timecount改为500/timecount后就没有问题了。

**思考与讨论**

如何针对不同频率的信号设置合适的周期输出点数，并分析设置周期输出点数的因素。

* 理论上，周期输出点数需要尽可能大，这样图像才光滑，但同时不能多到占用处理器太多时间，使得波形产生不流畅。

分析USB-4704 的模拟输出功能可采集信号的频率范围，若输出信号在该范围外，会出现哪些问题，并探讨可能的解决方案。

* 经过实验发现可采集的信号频率范围在1-8HZ内，若输入信号超出该范围就会发现重建的信号失真，具体表现形式是重建后的频率和原信号频率不一样，可能的解决方案就是改良程序的性能和运行速度，避免产生限制频率。