# 开发问题记录

1. 故障模拟中的短路计算找不到和本仪器对应的公式
2. DSP的addsp\_i(0,num) = num+2（测试能够重现），即其中一个数为0的时候会发生问题，两个数都为0没有问题
3. DSP的缓冲区安全问题（换库解决），如果两次回送时间间隔太小而传送的数据量过大就会导致数据错误
4. 对于多通道的DSP波形生成，由于缓冲区大小的限制（30K），按照12通道来计算的话，如果回传一次，则每个通道最多生成625个样点数据，如果是30通道，则最多生成250个样点数据，如果是16通道，则每个通道最多生成468个样点数据
5. 对于采样点数据，如果每个通道发送大于或等于256个采样点的话，在DSP端计算实际值的地方如果用mpysp\_i就会卡死，而用\*则不会，搞不懂错误原因，即使mpysp\_i(1,1)也不行，代码如下：

for(i=0; i<pSmvRawData->nChannelCount; i++)

{

//缓存到缓冲区

for(j=0; j<pSmvRawData->nSmvPerCircle; j++)

{

#if 1

g\_smvCalcData.fChannelData[i][g\_nSmvCacheCount+j] =

pSmvRawData->fChannelData[i][j] \*pSmvRawData->fScaleFactor[i];

#endif

//用mpysp\_i的话，如果pSmvRawData->nSmvPerCircle>=256就会在这儿卡死

#if 0

g\_smvCalcData.fChannelData[i][10] = mpysp\_i(1, 1);

//mpysp\_i(pSmvRawData->fChannelData[i][j], pSmvRawData->fScaleFactor[i]);

#endif

}

1. 当板子刚启动完毕之后，运行程序进行数据计算时，刚开始数据接收很缓慢，大约过5秒钟后接收变为正常。但一旦运行一次之后就不会出现这个问题了。