BNY转CIT操作设计文档

# BNY转CIT操作设计

BNY转CIT操作是指创建一个新的CIT文件，将BNY文件中的数据写入到CIT文件中，需要指定的文件头文件信息和通道定义信息。

整个过程是，指定好CIT文件的文件头信息和通道定义信息后，

（1）调用CIT操作的类库，创建文件头信息和通道定义信息，

（2）调用BNY操作类获取BNY的通道数据，

（3）调用CIT操作的类库，将BNY的通道数据写入到CIT文件中。

## 类设计

### 实体类

实体类分为文件信息类、通道定义类以及里程标类，详情参考《Cit操作算法.docx》文档中的实体类。

### 常量类

这里包含了BNY文件的一个字节的大小（4个字节）和BNY文件的通道个数（18个通道），和一个采样点的大小（一个字节的大小\*通道个数）。

### 操作类

操作类主要是指获取BNY文件中的通道数据、获取BNY文件的长度、获取BNY文件的开始里程、结束里程、获取BNY文件指定里程的位置等等。

**1、获取BNY文件的长度**

根据传入的BNY的文件的路径，读取文件流中的长度，将长度返回即可。

**2、获取BNY文件的开始里程**

（1）根据传入的BNY的文件的路径，读取文件流中的第一个采样点

（2）然后将采样点中的里程数据转换成数值类型。

**3、获取BNY文件的结束里程**

（1）根据传入的BNY的文件的路径，读取文件流中的最后一个采样点，

（2）然后将采样点中的里程数据转换成数值类型。

**4、获取BNY文件指定里程的位置**

获取BNY文件指定里程的位置，是为了取一段里程范围内数据的时候用到的，根据指定的里程，将指针定位到文件流的位置。

（1）根据传入的BNY的文件的路径，读取文件流，

（2）获取到BNY文件流的长度，计算出总的采样点个数，

（3）遍历采样点，读出每一个里程点数据，

（4）如果是增里程，里程点数据大于指定的里程点数据，当前文件流的位置减去一个采样点的大小即是该里程的位置，

（5）如果是减里程，里程点数据小于指定的里程点数据，当前文件流的位置减去一个采样点的大小即是该里程的位置。

（6）返回位置结果。

**5、获取BNY文件的数据**

（1）获取到BNY文件流的长度，计算出采样点个数，

（2）定义List<float[]>，List集合的大小就是通道个数的大小，float[]中存放的是通道的数据，

（3）遍历采样点个数，读取每一个采样点，并把采样点中的通道数据对应存放到float[]数组中，

（4）返回List<float[]>。