GDSII （Stream Format）格式研究

邓恒华 2016-11-04

本地： C:\DENG\OptiXera\Develop\Docs\参考资料\GDSII

远程： <http://optixera.com/hdeng/Docs/tree/master/参考资料/GDSII>

其中Record Type详细分类介绍拷屏于“GDS\_II\_Stream\_Format\_Manual\_6.0\_Feb87.pdf”。

本文主要研究GDSII （Stream Format）二进制文档的含义，以及与文本文档的转换。

共分为这些部分：

Part 1 Binary vs. ASCII

Part 2 GDSII Text File（GDSII文本格式）

Part 3 GDSII KEY File（GDSII文本格式）

其中Part 1 Binary vs. ASCII是主要部分，详细解析GDSII文档每段的含义。

相对应，GDSII文档可以有不同后缀名，比如

* GDSII二进制: **.GDS**，.GDSII，.GDS2，.DB（DataBase）， .SF， STM, .STRM（Stream Format）
* GDSII文本格式: .TXT， .KEY， .ASCII

但是后缀名并不重要，最重要是GDSII （Stream Format）二进制码的记录（Records）解析。

****

以下16进制和文本文件都是通过PSPad打开。

# **Part 1 Binary vs. ASCII**

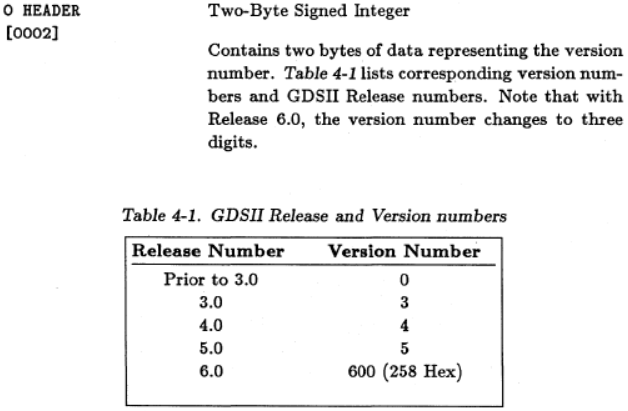
****

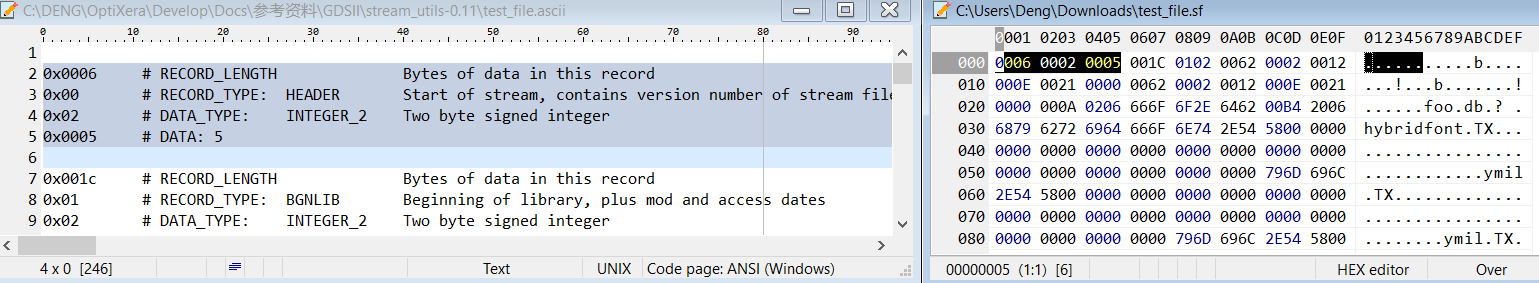
**【HEADER】 Record（文件头）**

0006 = 6个比特 （所有HEADER字节 = 0006 0002 0005）；

0002 = HEADER （此记录型号本身占掉2-byte Signed Integer）；

接下来的2-byte Signed Integer（6-2-2=2）是GDSII版本信息；

此处0005 = GDSII版本5.0。 



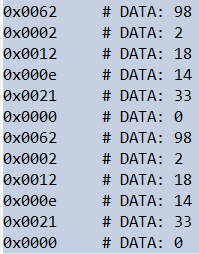
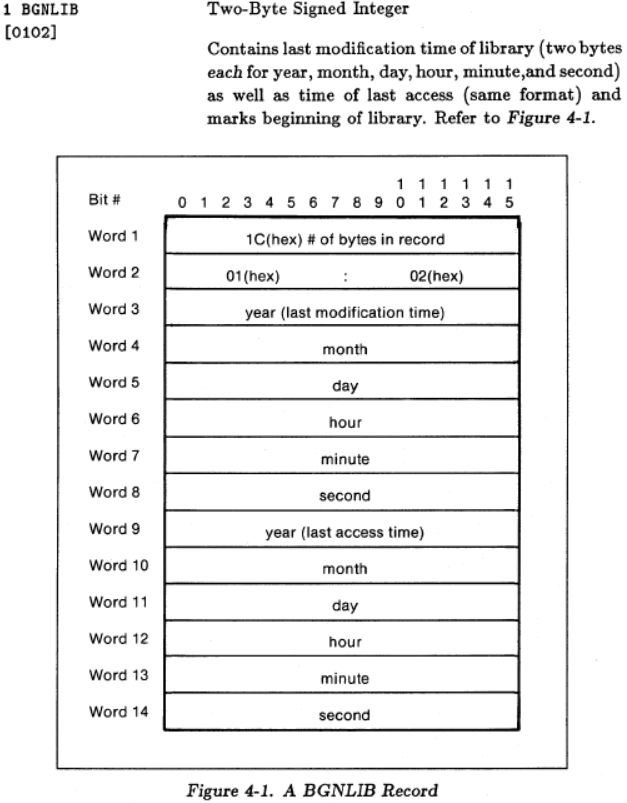
字节数

**【BGNLIB】 Record （Begin of Library，库起始）**

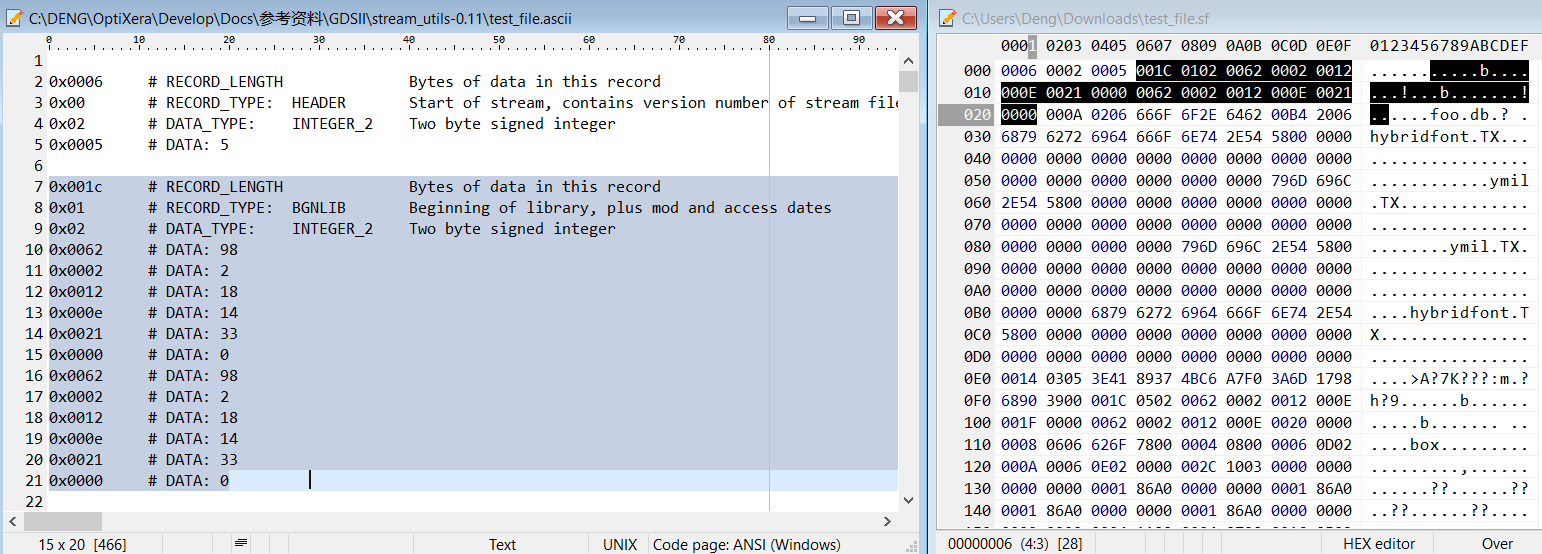
001C = 28个比特 （16进制 001C = 10进制 28。所有字节如选中所示）；

0102 = BGNLIB （此记录型号本身占掉2-byte Signed Integer）；

接下来的24比特（28-2-2=24）（12个2-byte Signed Integer）是时间信息；

此处 

= 上次修改时间 1998年2月18日14时33分00秒；上次访问时间 1998年2月18日14时33分00秒。



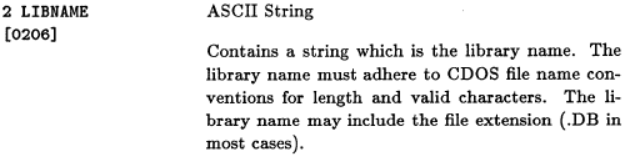
字节数

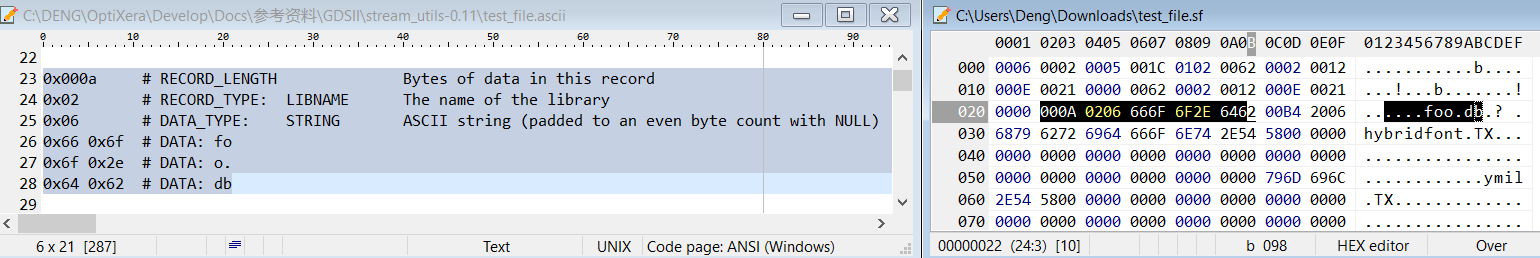
**【LIBNAME】 Record（Library Name，库名）**

000A = 10个比特 （16进制 000A = 10进制 10。所有字节如选中所示）；

0206 = LIBNAME （此记录型号本身占掉2-byte Signed Integer）；

接下来的6比特（10-2-2=6）（12个1-byte String）是Library Name信息；

此处Library Name为foo.db。 



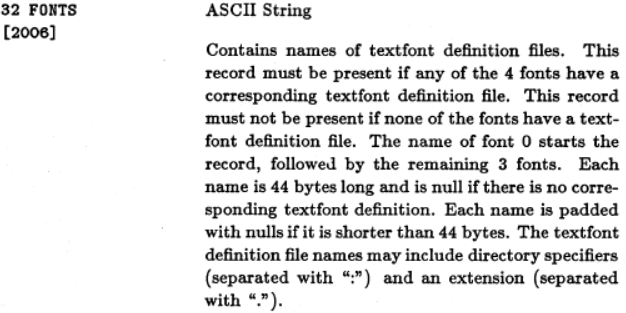
字节数

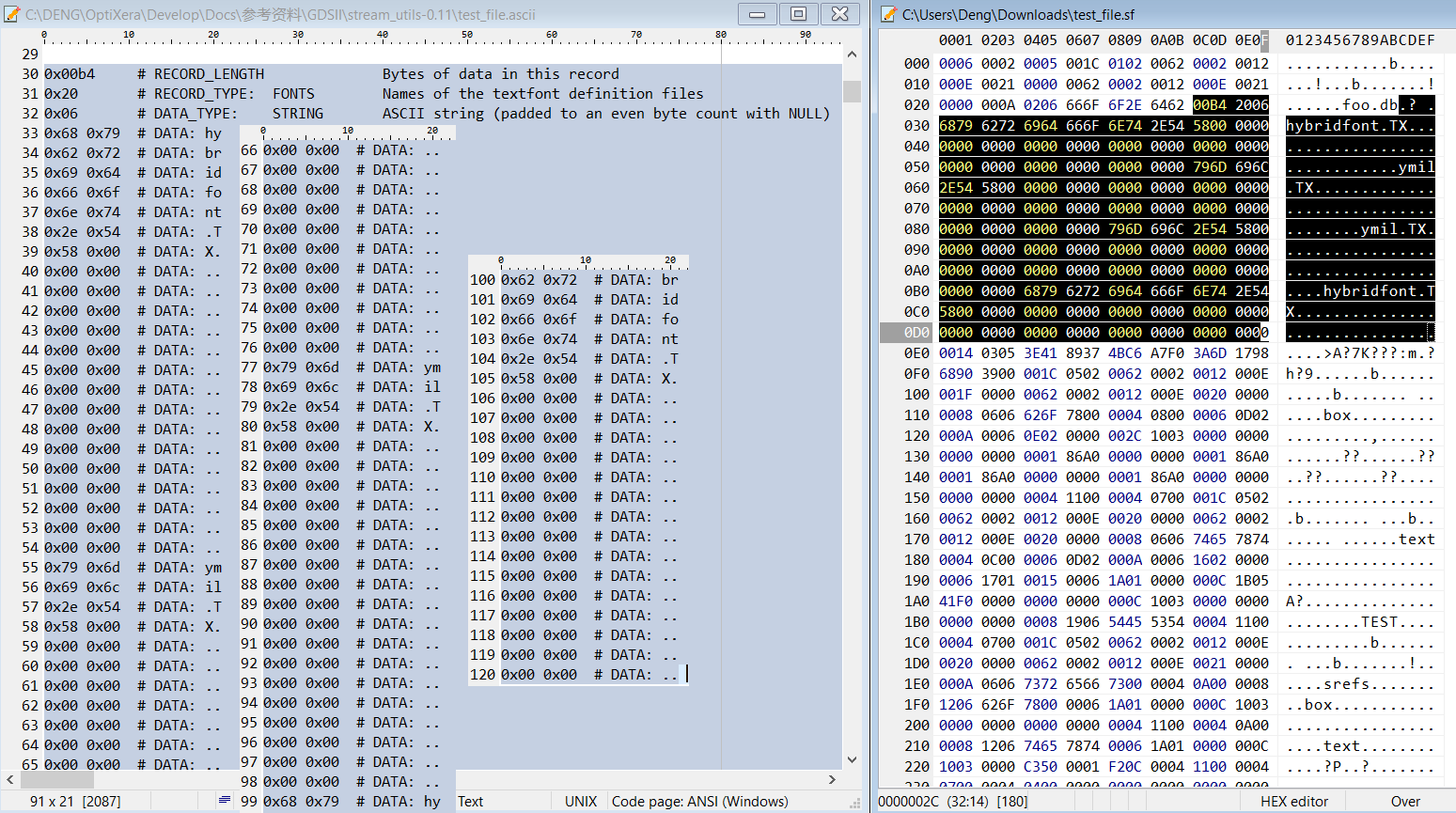
**【FONTS】 Record（字体）**

00B4 = 180个比特 （16进制00B4 = 10进制 180。所有字节如选中所示）；

2006 = FONTS （此记录型号本身占掉2-byte Signed Integer）；

接下来的174比特（180-2-2=174）（348个1-byte String）是Fonts信息；





字节数

**【UNITS】 Record（长度单位）**

0014 = 20个比特 （16进制 0014 = 10进制 20。所有字节如选中所示）；

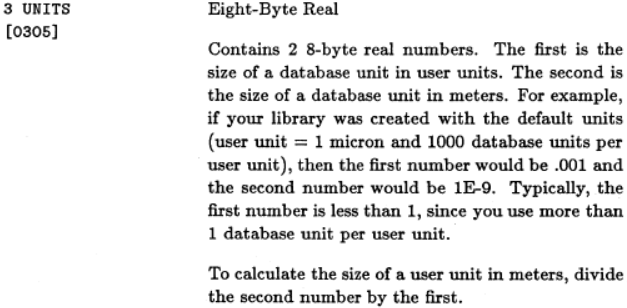
0305 = UNITS （此记录型号本身占掉2-byte Signed Integer）；

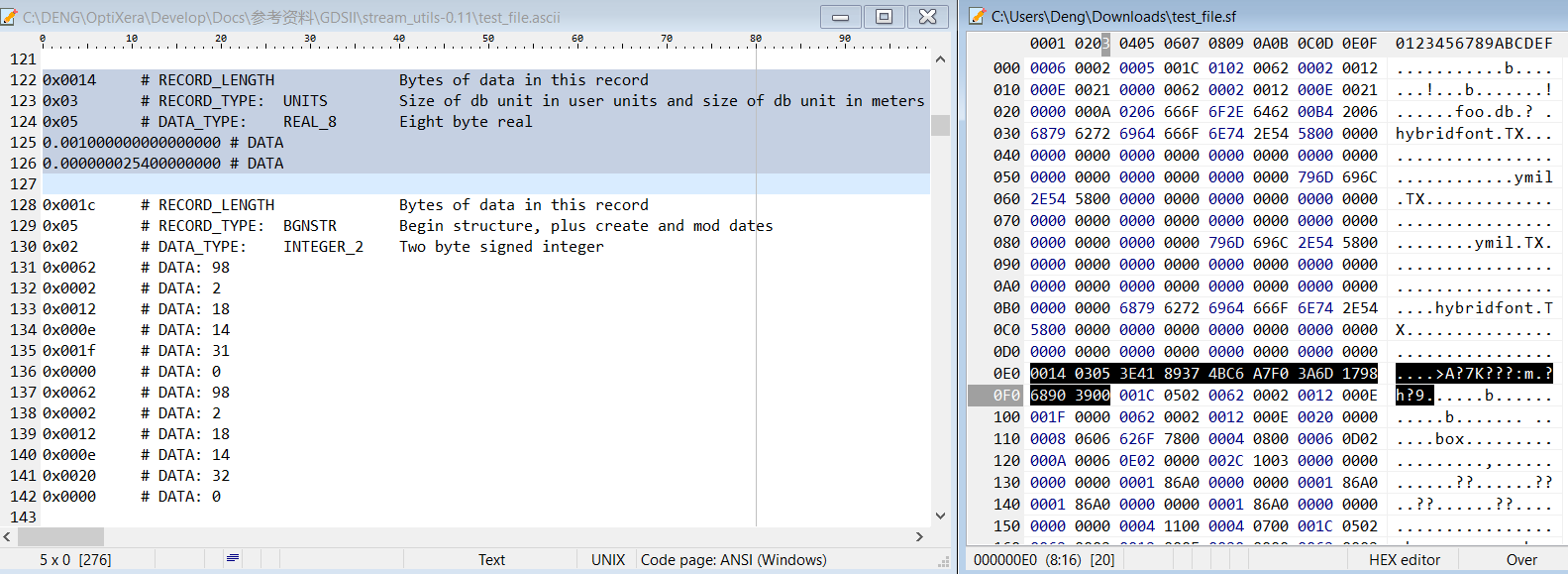
接下来的16比特（20-2-2=16）是**2个**8比特Real Number （Real\_8）存长度单位信息；

此处第一个Real\_8（8比特）是User Unit（用户长度单位），值为0.001，表示每个User Unit是1000个Database Unit，即*每个Database Unit是0.001个User Unit*。

第二个Real\_8（8比特）是Database Unit（数据库长度单位），值为0.0000000254（2.54e-8）meter （米），即25.4 nm（纳米，1e-9 m）。

该User Unit（用户长度单位）相当于 2.54e-8/0.001 = 2.54e-5 meter（米）= 25.4 um（微米，1e-6 m）= 0.001 inch （英寸）。





字节数

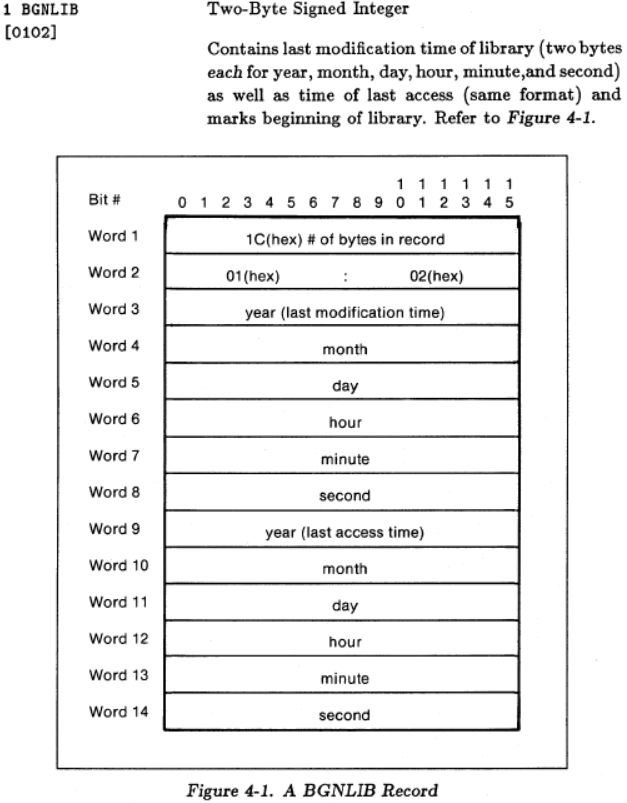
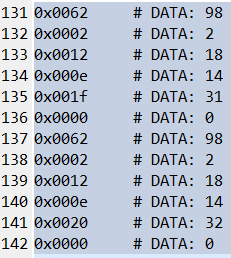
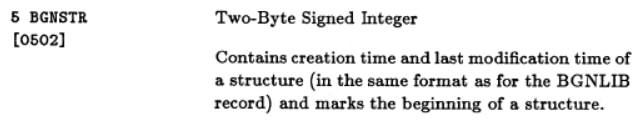
数据类型

**【BGNSTR】 Record（Begin of Structure，结构起始）**

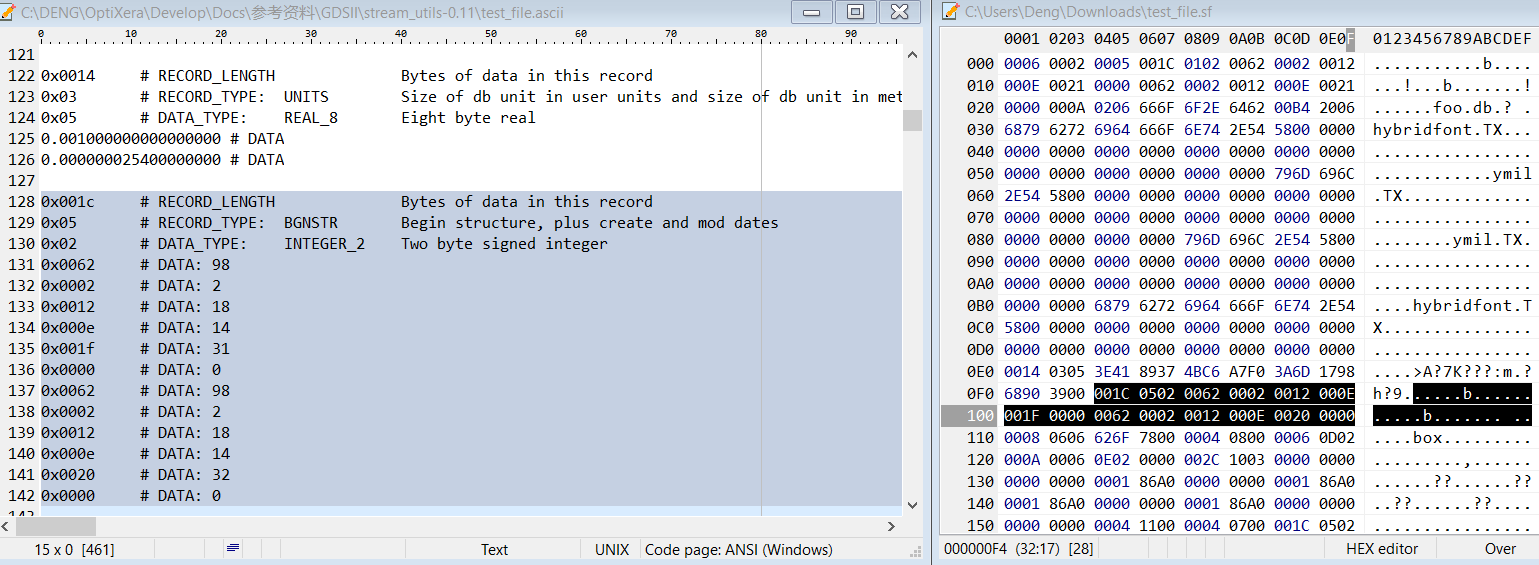
001C = 28个比特 （16进制 001C = 10进制 28。所有字节如选中所示）；

0502 = BGNSTR （此记录型号本身占掉2-byte Signed Integer）；

接下来的24比特（28-2-2=24）（12个2-byte Signed Integer）是时间信息；

= 上次修改时间 1998年2月18日14时31分00秒；上次访问时间 1998年2月18日14时32分00秒。



字节数

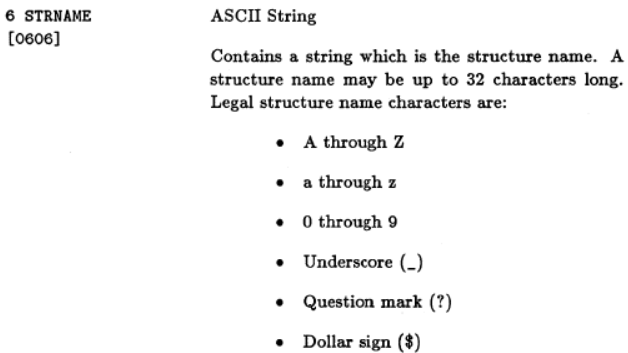
**【STRNAME】 Record（Structure Name，结构名）**

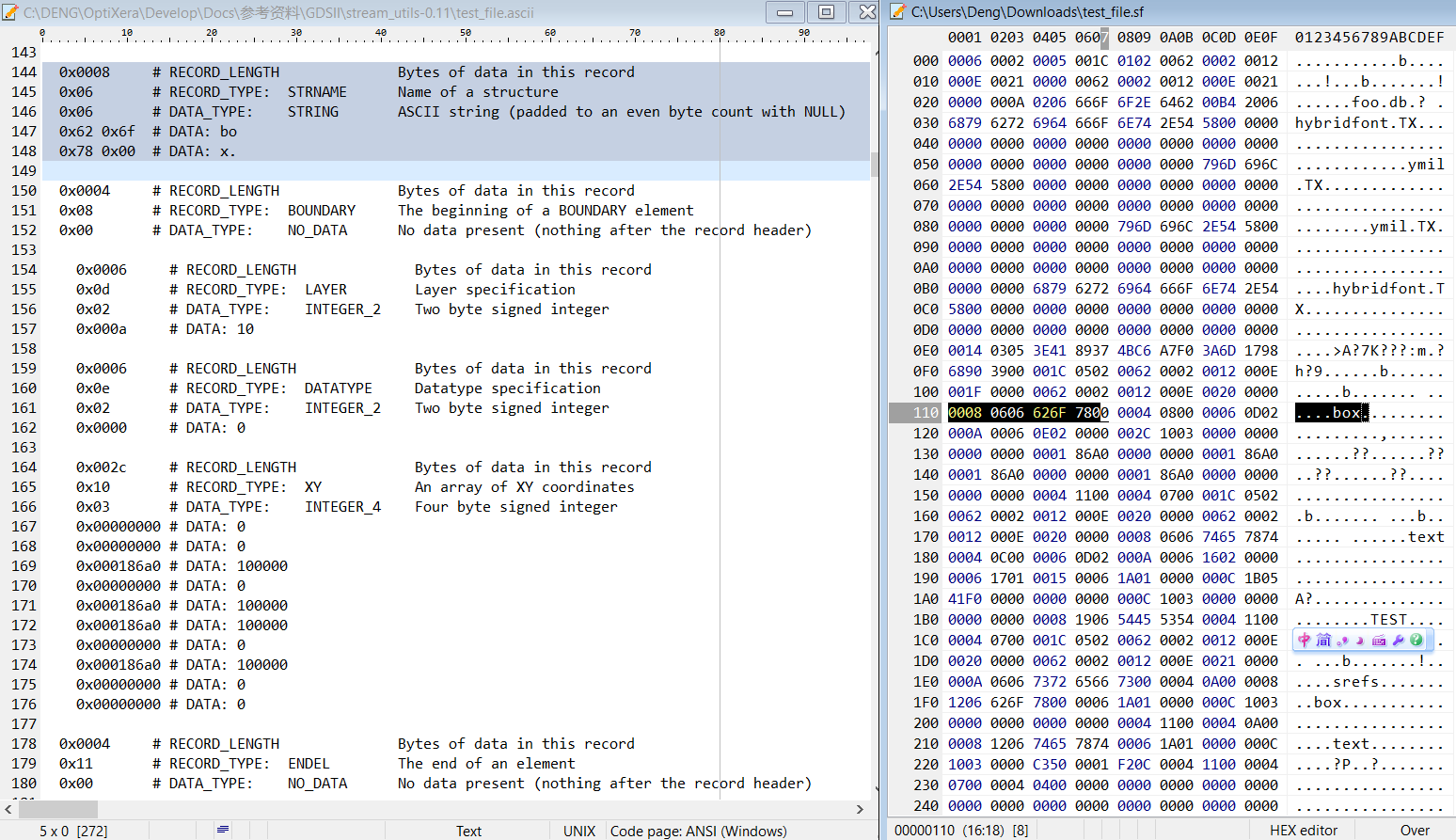
此结构名记录0008 = 8个比特 （16进制 0008 = 10进制 8。所有字节如选中所示）；

0606 = STRNAME （此记录型号本身占掉2-byte Signed Integer）；

接下来的4比特（8-2-2=4）（4个1-byte String）是结构名信息；

此处Structure Name为box。





字节数

**【BOUNDARY】 Record（界线）**

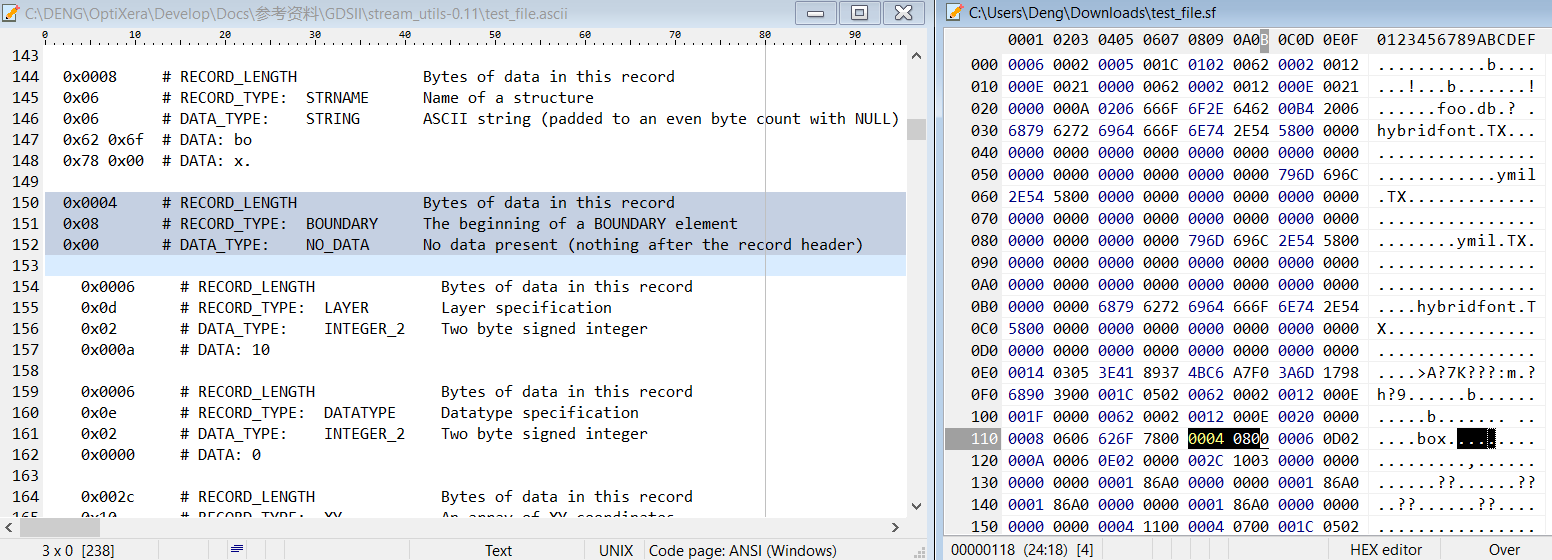
此记录0004 = 4个比特 （16进制 0008 = 10进制 8。所有字节如选中所示）；

0800 = BOUNDARY （此记录型号本身占掉2-byte Signed Integer）；

接下来的0比特（4-2-2=0）表示此界线记录头为空；

需要接下来先定义Layer和DataType才可确定在什么层上画此界线。

***（注意后文文本缩进，表示是给上面的BOUNDARY界线记录用的****。）*



字节数

**【LAYER】 Record（层名）**

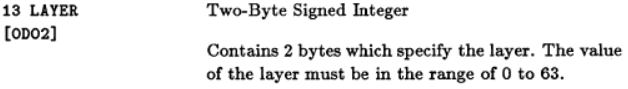
此记录0006 = 6个比特 （16进制 0006 = 10进制 6。所有字节如选中所示）；

***注意文本缩进，表示此层是给上面的BOUNDARY界线记录用的***。

0D02 = LAYER （此记录型号本身占掉2-byte Signed Integer）；

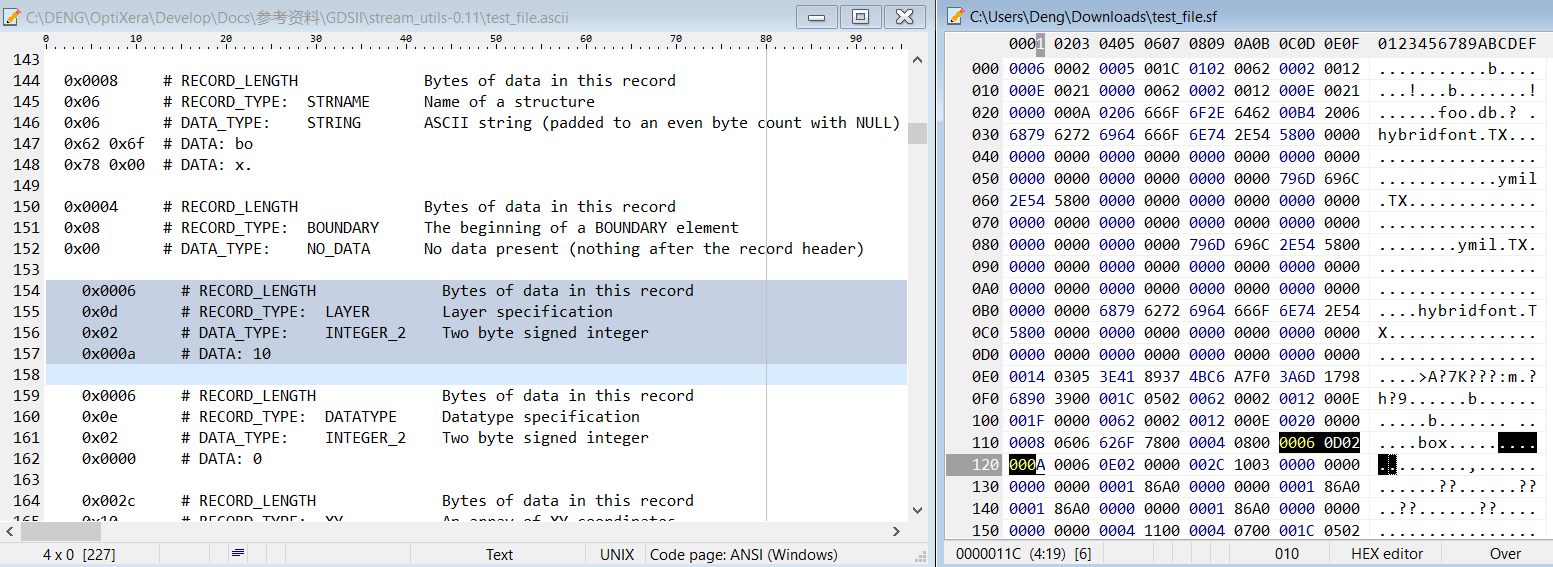
接下来的2比特（6-2-2=2）（1个2-byte Signed Integer）是层信息；

此处Layer #为10。



实际上GDSII允许256层（1个2-byte Signed Integer = 16 位，2^16 = 256。）

（扩展版本可有65536层？2个2-byte Signed Integer = 32 位，2^32 = 65536。）



字节数

**【DATATYPE】 Record（数据形式名）**

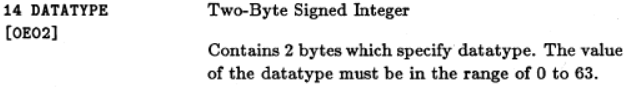
此记录0006 = 6个比特 （16进制 0006 = 10进制 6。所有字节如选中所示）；

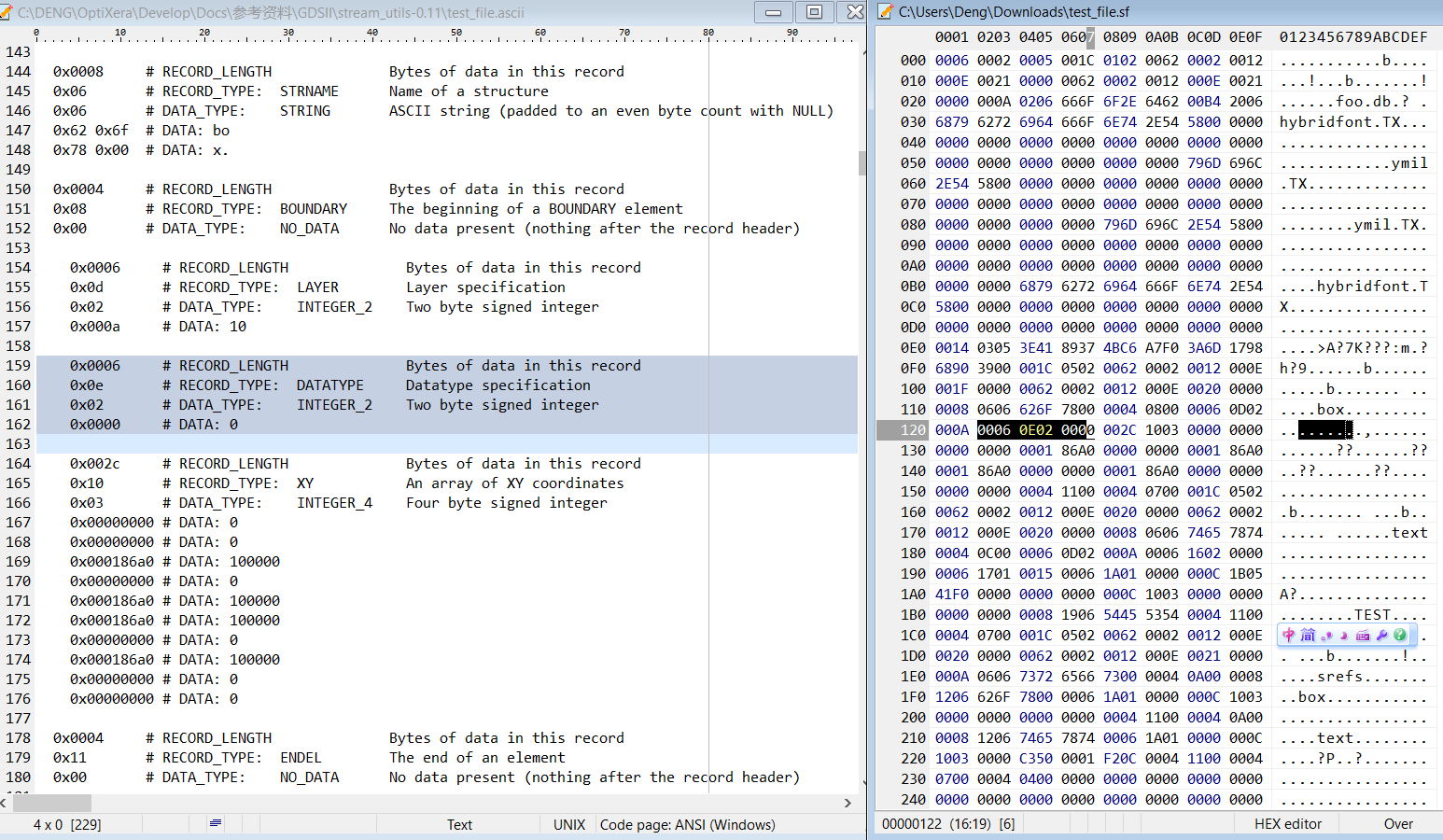
***注意文本缩进，表示此数据形式是给上面的BOUNDARY界线记录用的***。

0E02 = DATATYPE （此记录型号本身占掉2-byte Signed Integer）；

接下来的2比特（6-2-2=2）（1个2-byte Signed Integer）是数据形式信息；

此处DataType为0。





字节数

**【XY】 Record（位置坐标值）**

此层名记录002C = 44个比特 （16进制 002C = 10进制 44。所有字节如选中所示）；

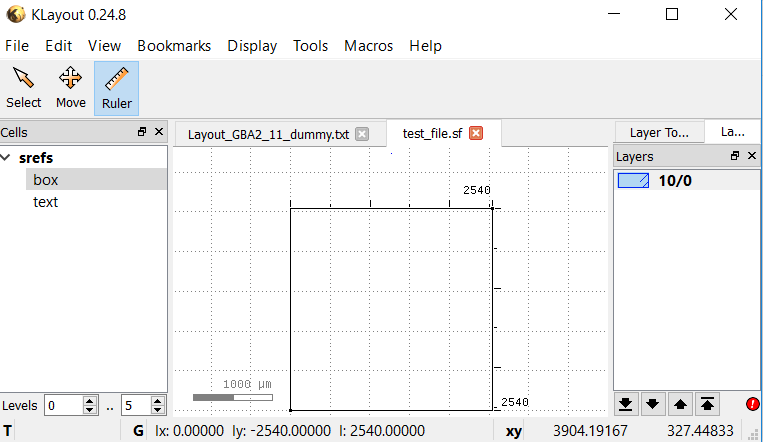
***注意文本缩进，表示此位置坐标值是给上面的BOUNDARY界线记录用的***。

1003 = XY （此记录型号本身占掉2-byte Signed Integer）；

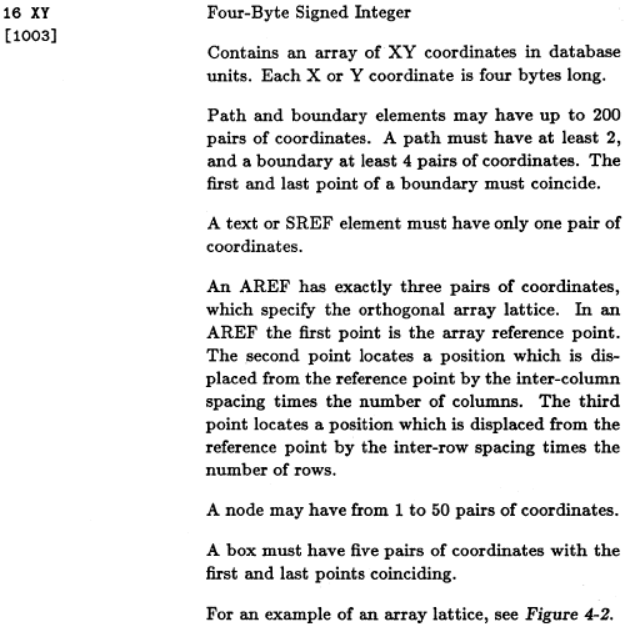
接下来的40比特（44-2-2=40）（10个4-byte Signed Integer）是(X,Y)坐标信息；

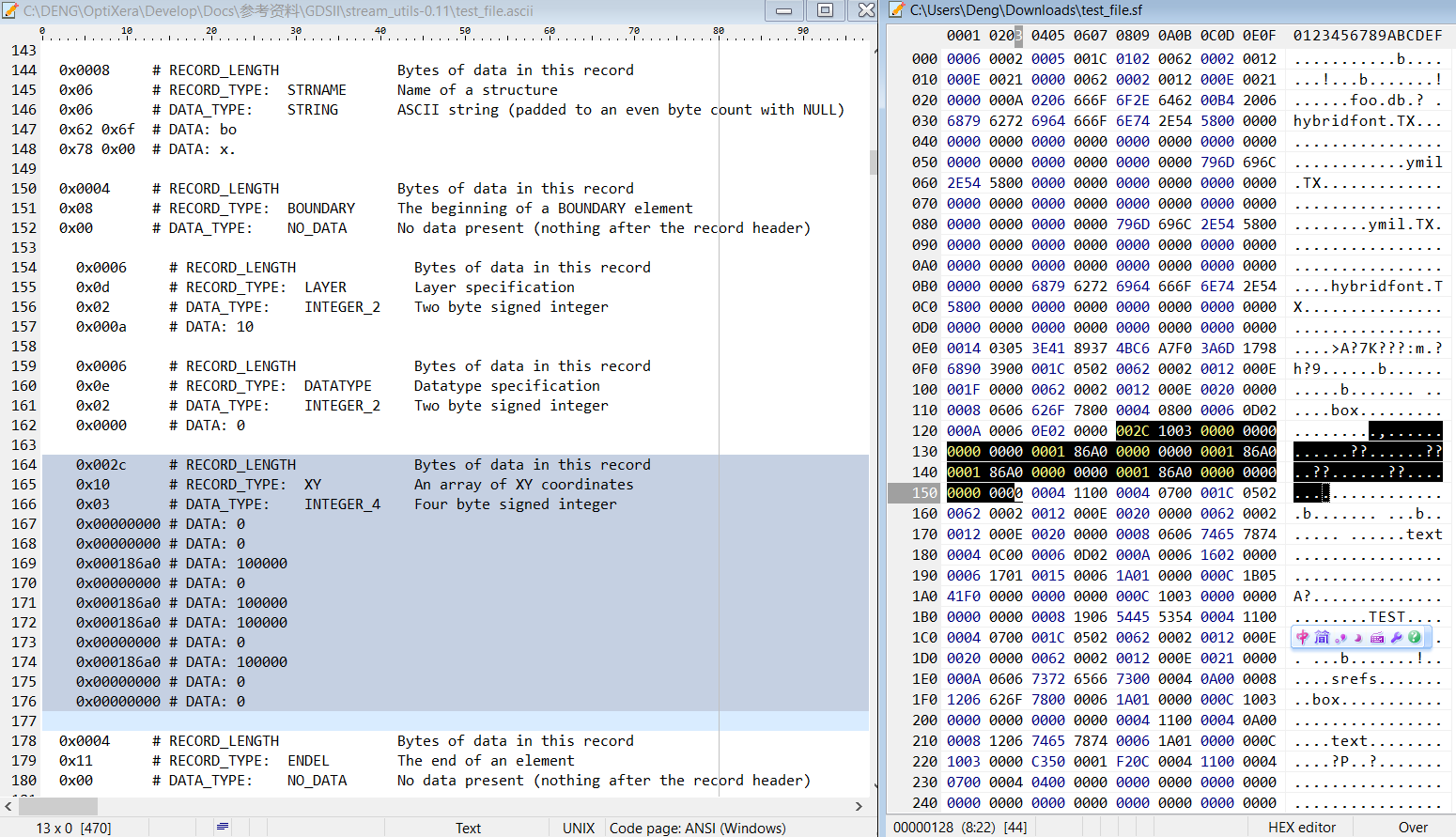
此处(X,Y)坐标组为：(0,0), (100000,0), (100000,100000), (0,10000), (0,0)。刚好表示是一个BOX。其中100000表示1e+5 \* Database Unit（数据库长度单位）。此例如第5页【Unit】所算，该BOX是边长为1e+5 \* 2.54e-8 = 2.54e-3 meter（米）= 2.540mm = 2540um。以下KLayout显示根据尺度计算，此BOX边长为2540um。

注意：对于Boundary（界线）第一个点和最后一个点坐标必须相同。对于Path（路径）可以首尾不相同。



尺度





字节数

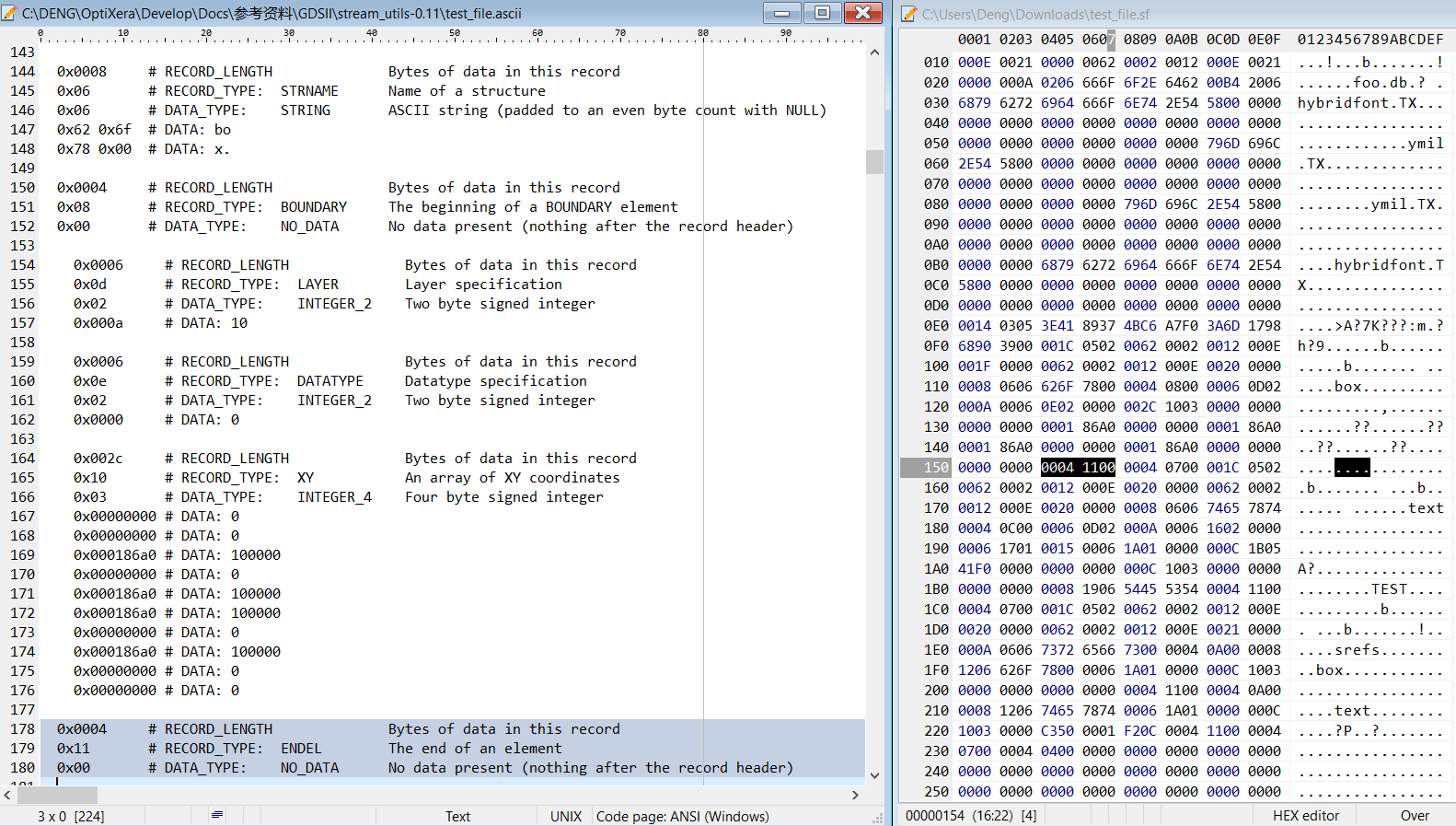
**【ENDEL】 Record（**End of Element，**单元结束）**

此记录0004 = 4个比特 （16进制 0004 = 10进制 4。所有字节如选中所示）；

***注意文本缩进此时结束，表示上面的BOUNDARY界线记录在此处结束***。

1100 = ENDEL （此记录型号本身占掉2-byte Signed Integer）；

接下来的0比特（4-2-2=0）表示此界线记录结束；



字节数

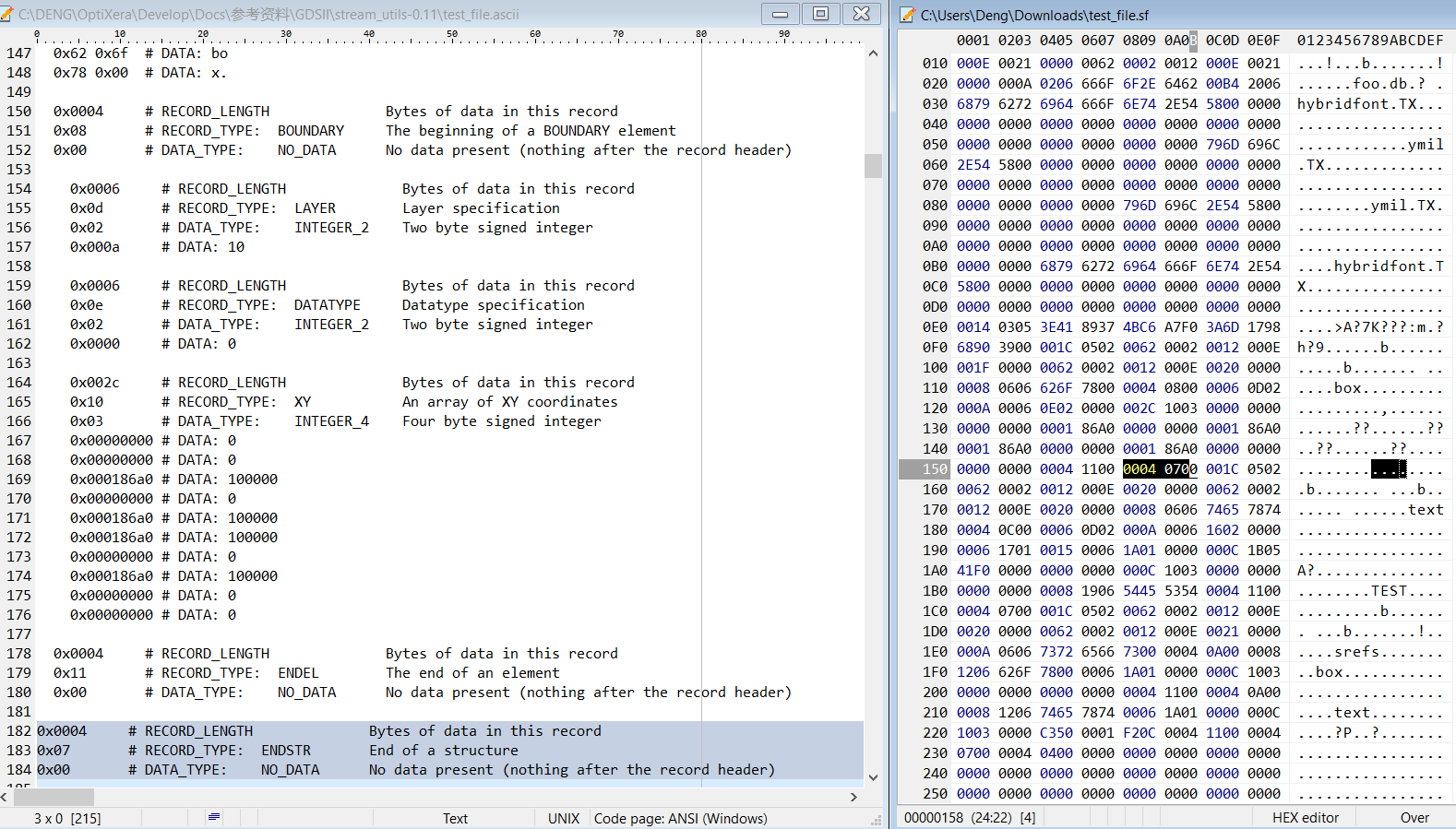
**【ENDSTR】 Record（**End of Structure，**结构结束）**

此记录0004 = 4个比特 （16进制 0004 = 10进制 4。所有字节如选中所示）；

***注意文本缩进此时结束，表示上面的Structure记录在此处结束***。此Structure仅仅包含上面一个BOX（**Boundary** Element）。

0700 = ENDSTR （此记录型号本身占掉2-byte Signed Integer）；

接下来的0比特（4-2-2=0）表示此结构记录结束；



字节数

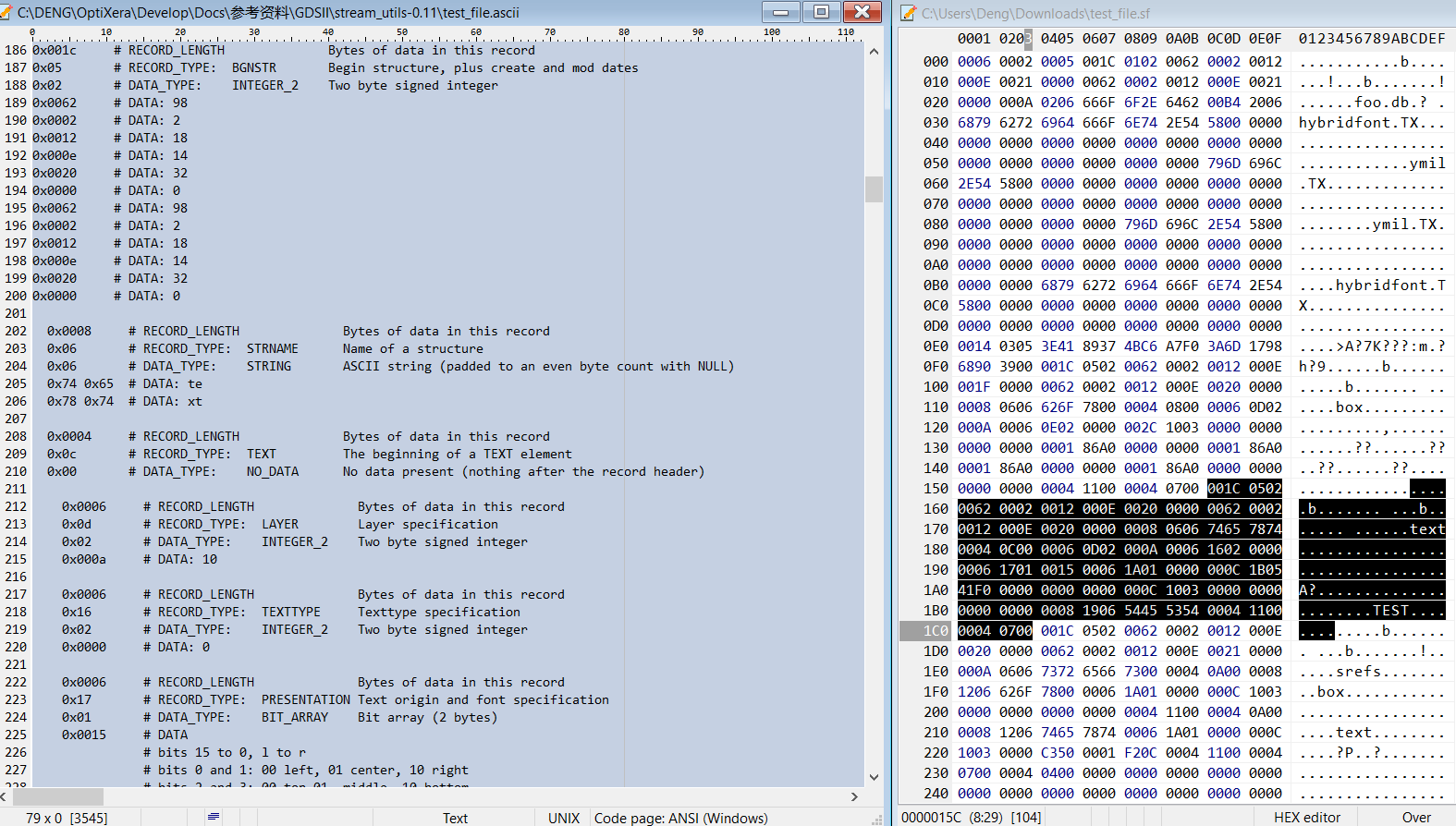
**【另外一个完整的Structure】 Records（**Text例子**）**

下一个Structure是Text例子（所有字节如选中所示）；***注意文本二层缩进***。

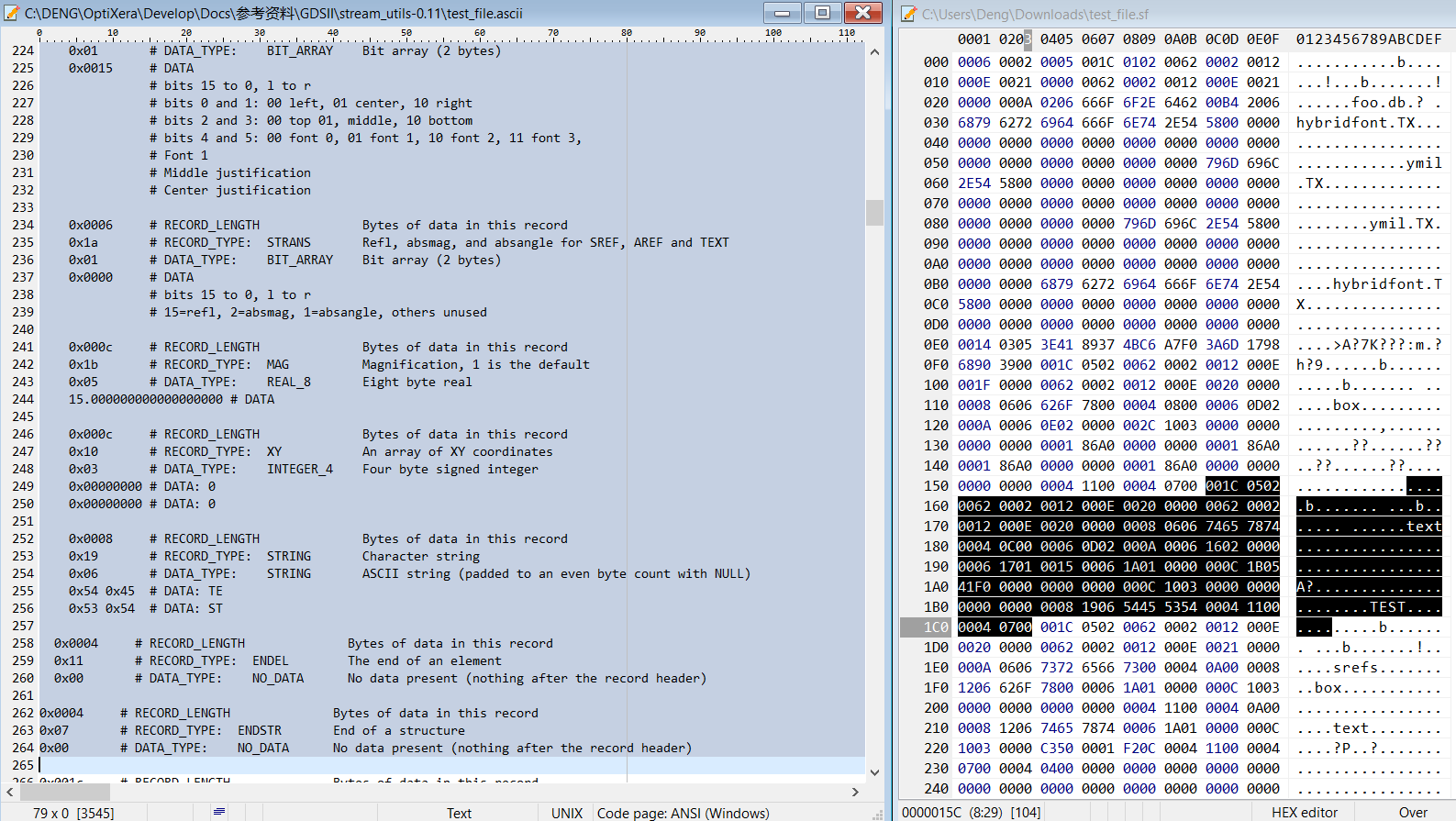
【BGNSTR】

【STRNAME】【TEXT】【LAYER】【TEXTTYPE】【PRESENTATION】【STRANS】【MAG】【XY】【STRING】【ENDEL】

【ENDSTR】



字节数



**【SREF】 Records（**Structure Reference，结构引用**）**

下一个Structure是SREF例子（所有字节如选中所示）；***注意文本二层缩进***。

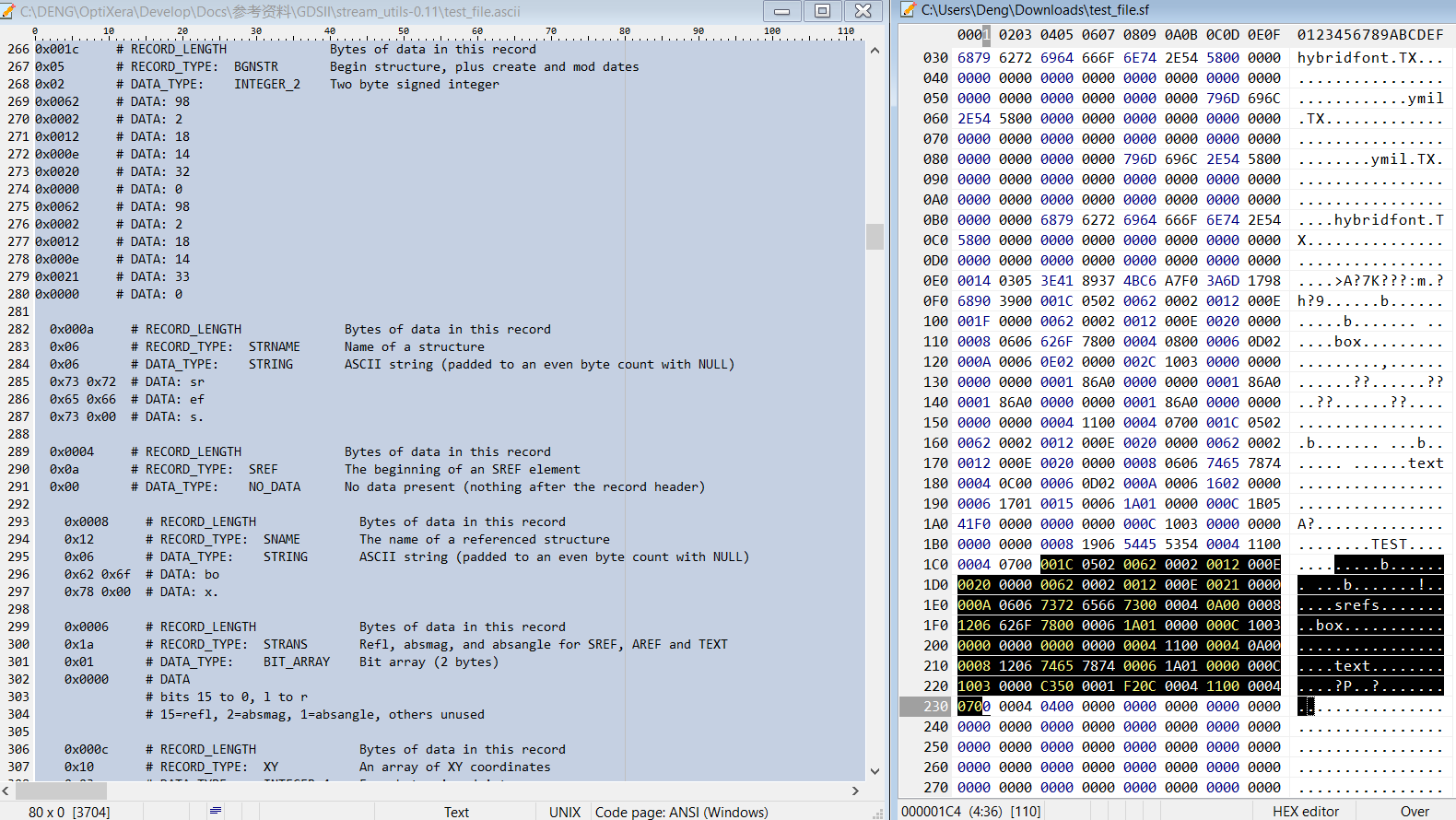
此Structure（结构）共引用了box和text两个Sub-Structure（子结构）。结构引用子结构的数目和层次并没有限制。

【BGNSTR】

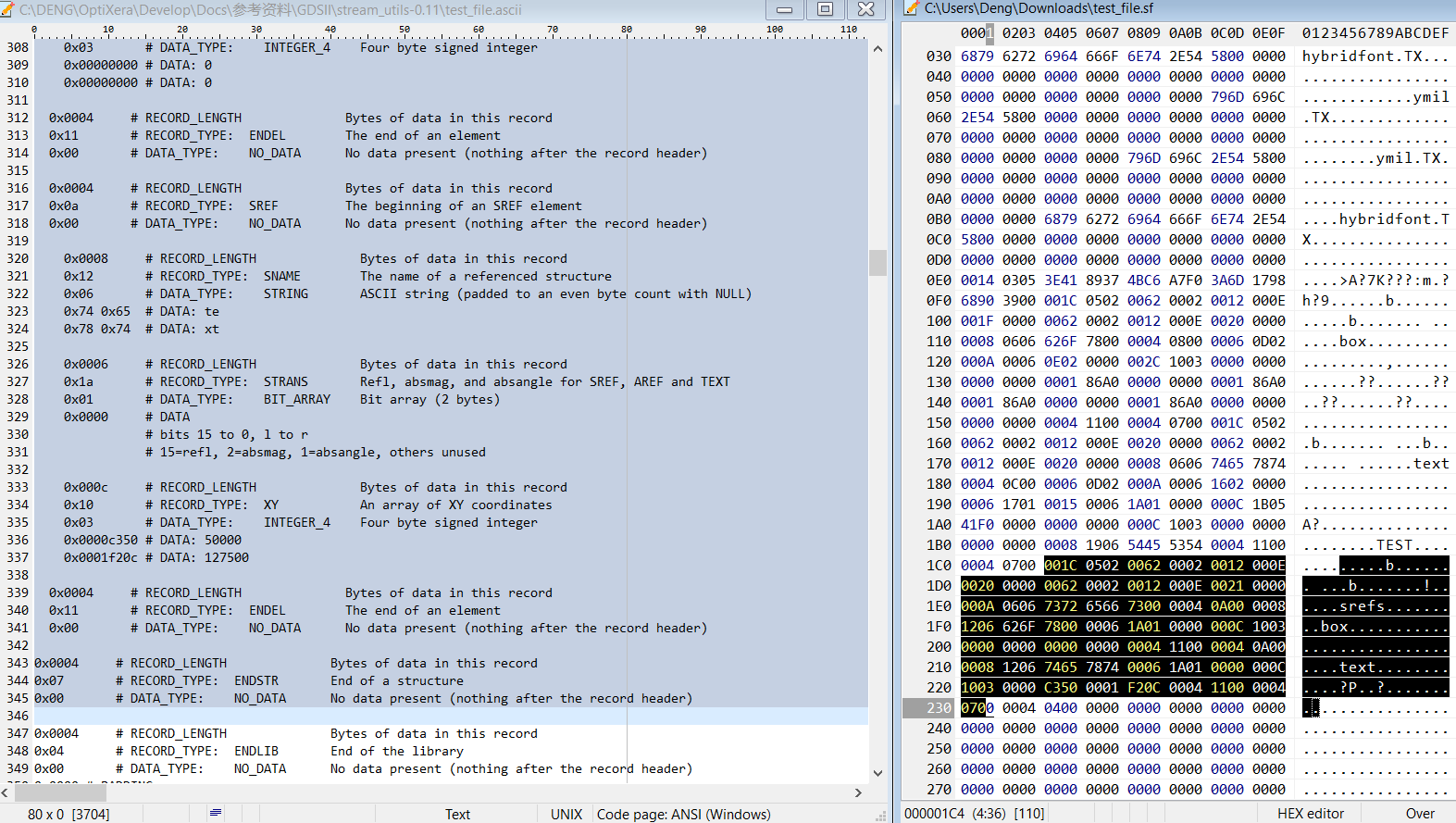
【STRNAME】 【SREF】【SNAME｛box｝】【STRANS】【XY】【ENDEL】

【SREF】【SNAME｛text｝】【STRANS】【XY】【ENDEL】

【ENDSTR】



字节数

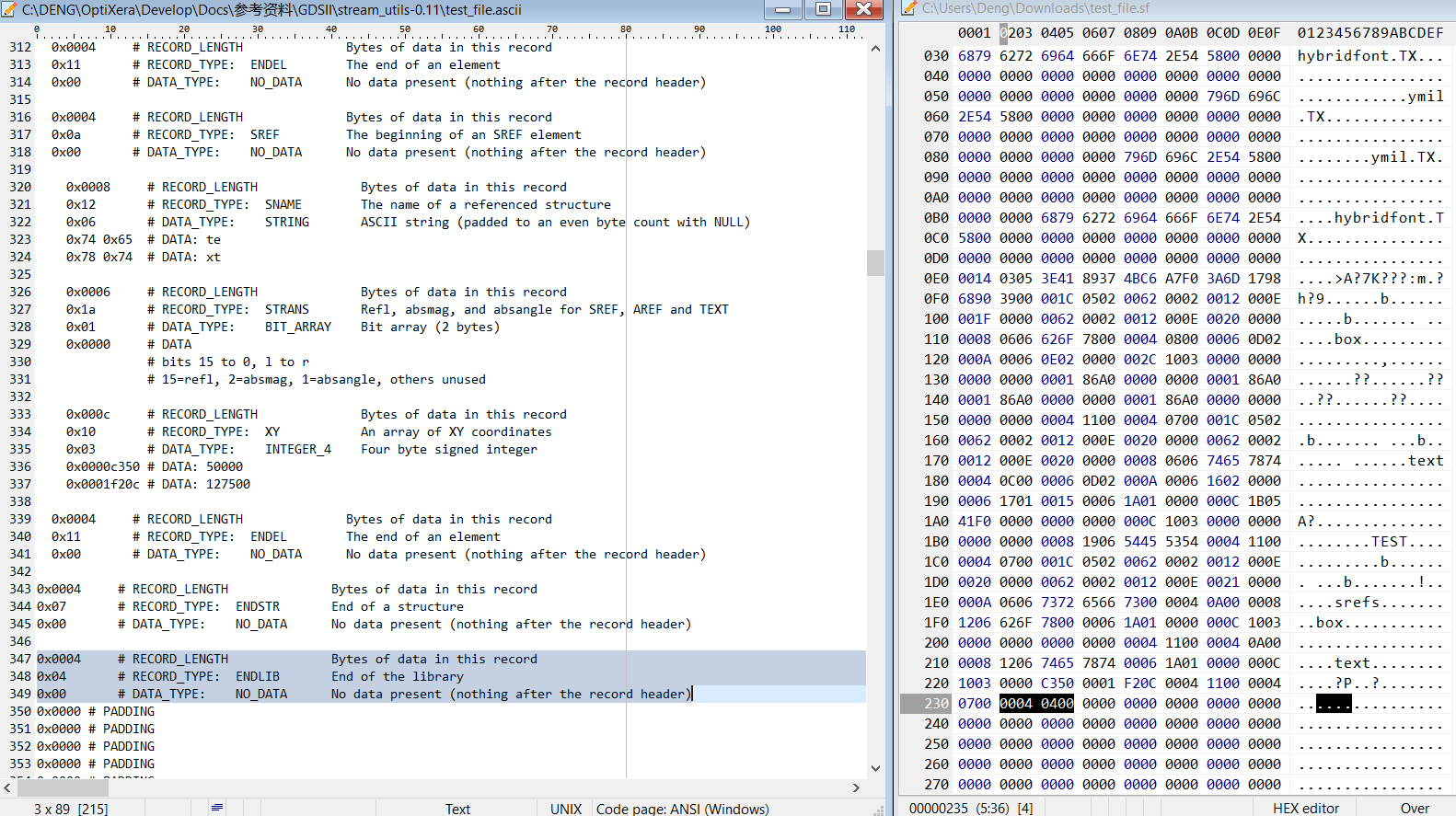


**【ENDLIB】 Record（End of Library，库终止）**

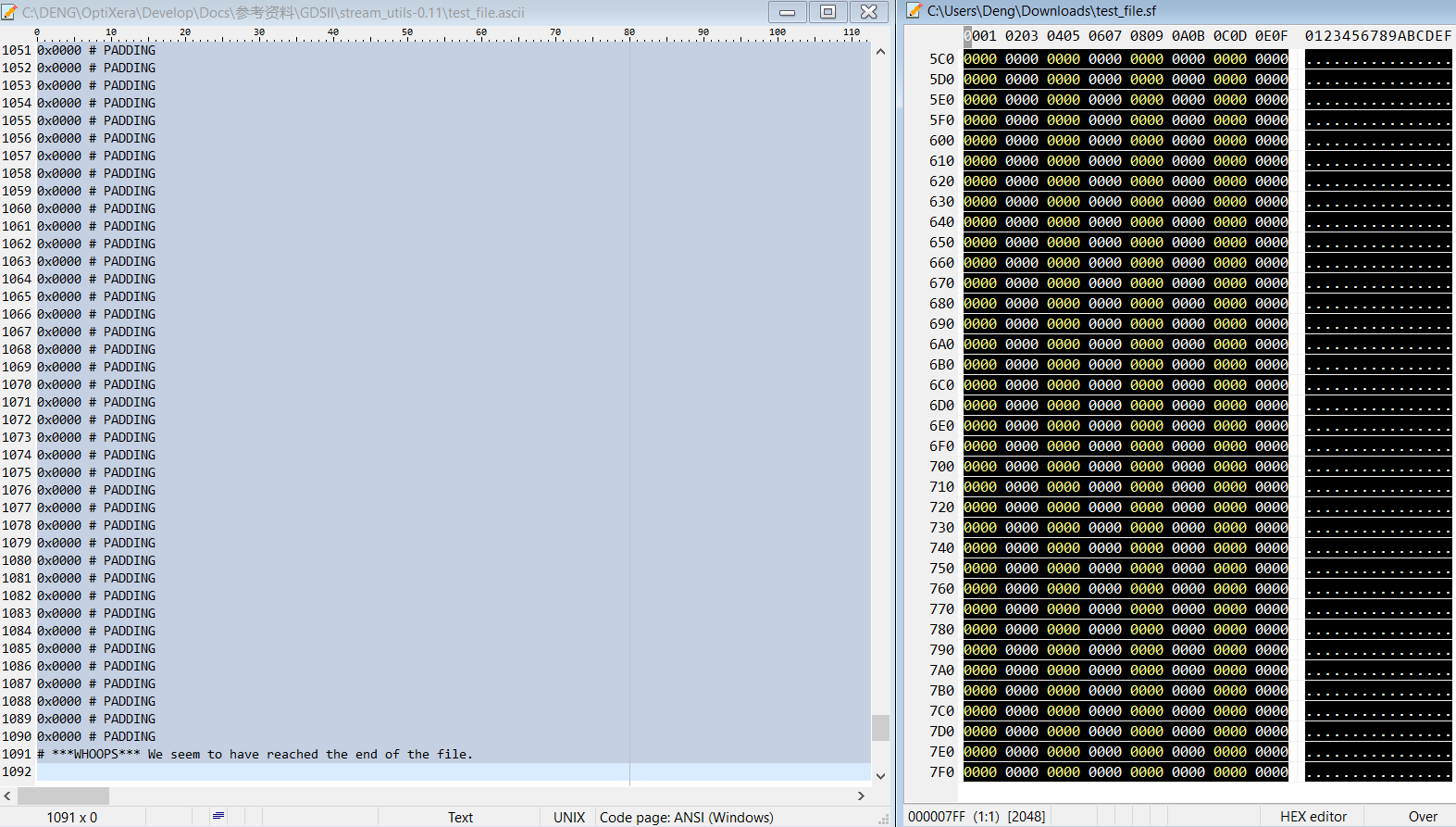
0004 = 4个比特 （所有ENDLIB字节 = 0004 0400）；

0400 = ENDLIB （此记录型号本身占掉2-byte Signed Integer）；

接下来字节全部填充为0，直到整数倍2048字节；历史原因是以前每盘磁带2048字节。



字节数



**【文件全部结束】**

# **Part 2 GDSII Text File（GDSII文本格式）**

****

很多GDSII文档实际上不用上节（Part 1 ）内ASCII格式，而是直接存成更简单的GDSII Text File（GDSII文本格式）。LinkCAD写出的GDSII文本格式可以与KLayout通用。

上节的格式转成GDSII Text File之后如下，可阅读性增强。但是上节的解析依然成立。

HEADER 600

BGNLIB

LIBNAME foo.db

UNITS 0.0254 2.54e-08

BGNSTR

STRNAME text

TEXT

LAYER 10

TEXTTYPE 0

PRESENTATION 5

STRANS 0

MAG 381

XY 0: 0

STRING TEST

ENDEL

ENDSTR

BGNSTR

STRNAME box

BOUNDARY

LAYER 10

DATATYPE 0

XY 0: 0

0: 100000

100000: 100000

100000: 0

0: 0

ENDEL

ENDSTR

BGNSTR

STRNAME srefs

SREF

SNAME box

XY 0: 0

ENDEL

SREF

SNAME text

XY 50000: 127500

ENDEL

ENDSTR

ENDLIB

# **Part ３ GDSII KEY File（GDSII文本格式）**

有些软件版本支持所谓的KEY File，其实也是**自定义文本解释格式**。

可能具体根据软件内部解释各有稍许不同，比如

* 【Klaas\_Holwerda\_Polygon\_Boolean\_Operations.pdf 】KEY files (own made extended GDSII in ascii format)，但是无法被KLayout打开；
* LinkCAD写出的GDSII文本格式可以与KLayout通用；
* L-Edit只导入二进制GDSII格式，而不支持GDSII文本格式导入。

以下是一个自定义文本格式，C:\DENG\OptiXera\Develop\Docs\参考资料\GDSII\boolean\_7\_1\boolean\apps\edit\default\data



# KEY file for GDS-II postprocessing tool

# File = C:/bool/boolean/5.6/default/data/tworec2l

# ====================================================================

HEADER 5; # version

BGNLIB;

LASTMOD {95-5-29 10:42:18}; # last modification time

LASTACC {95-5-29 10:42:18}; # last access time

LIBNAME TEMPEGS.DB;

UNITS;

USERUNITS 0.01; PHYSUNITS 1e-08;

BGNSTR; # Begin of structure

CREATION {95-5-29 10:42:18}; # creation time

LASTMOD {95-5-29 10:42:18}; # last modification time

STRNAME TOTAL;

PATH; LAYER 0; DATATYPE 0;

XY 4;

X 60000.000; Y 510000.000; X 490000.000; Y 510000.000;

X 490000.000; Y 100000.000; X 60000.000; Y 100000.000;

PROPERTY aaap;

PROPTYPE weet;

PROPVALUE nounou;

PROPERTY aaap2;

PROPTYPE weet2;

PROPVALUE nounou2;

ENDEL;

BOUNDARY; LAYER 1; DATATYPE 0;

XY 7;

X -10000.000; Y 430000.000; X 170000.000; Y 430000.000;

X 360000.000; Y 430000.000; X 570000.000; Y 430000.000;

X 570000.000; Y 170000.000; X -10000.000; Y 170000.000;

X -10000.000; Y 430000.000;

PROPERTY aaap4;

PROPTYPE weet4;

PROPVALUE nounou4;

ENDEL;

ENDSTR TOTAL;

ENDLIB;

文档结束[EOF]。