[output.1 是这样生成的: 1](#_Toc401863692)

[1. 信号输出 1](#_Toc401863693)

[2. 开关量接点映射： 1](#_Toc401863694)

[3. 注意 1](#_Toc401863695)

[附件1 2](#_Toc401863696)

# output.1 是这样生成的:

## 1. 信号输出

(1) <In\_Smv .根据前缀 “220kV 陈家桥1线合并单元1(PCS221G)” ，appid="16806"

查找 IED (匹配 desc ，appid）， 将 <SMV 段 ，拷贝到， In\_Smv 段的内容中

A. 对于 每个 <ChnInfo ， 增加属性 quality ，缺省 0，

B. 对于 每个 <ChnInfo ， 增加属性 channel,

输入：IA1,IB1,IC1,IN1,VA1,VB1,VC1,VN1 ,IA2,IB2,IC2,IN2,VA2,VB2,VC2,VN2, DelayT, None

DelayT 是额定延时

判断原则，方法：

* 参考以前的 SCL，根据 chnName ， 参考 附件1（附件1对第1，2通道未判断） ；
* 利用 type 和 flag 属性 ， 但是 1,2还是 chnName来决定
* 如果是额定延时，再增加属性：delayt

C. In\_Smv 段 原来的内容删除

(2) In\_Goose , 根据appid="370", 查找

将 Goose 段，将整个 GOOSE 段（注意与 SMV有差异，多了头）拷贝到， In\_Goose 段中 , 原来的内容保留

如果有输出接点映射：

在 <goose 段最后，添加映射信息：

<MapInfo Which="1" Pos="1" />

<MapInfo Which="2" Pos="5" />

解释：将该GOOSE 的第 1 个数据（从 1 开始）映射到 接点1 （从 1 开始）

将该GOOSE 的第 5 个数据（从 1 开始）映射到 接点2 （从 1 开始）

(3) 其它同理

(4) 添加光口信息，在 <In\_Smv ，<In\_Goose ， 最后添加：portIndex="1" （光口编号从1开始）

## 2. 开关量接点映射：

（1）在 <goose 段最后，添加映射信息：

<MapInfo Which="1" Pos="3" />

<MapInfo Which="2" Pos="5" />

解释：将该GOOSE 的第 3 个数据（从 1 开始）映射到 接点1 （从 1 开始）

将该GOOSE 的第 5 个数据（从 1 开始）映射到 接点2 （从 1 开始）

没有映射，不用添加。

（2）添加光口信息，在 < goose ， 最后添加：portIndex="1" （光口编号从1开始）

（3）注意：因为程序按顺序解析，要求：开关量的映射从小到大，以顺序，不能空；

另外 多个GOOSE，小的优先。如何理解: GOOSE 0x2001 包括开关量 2 的映射，GOOSE 0x2005 包括开关量1的映射，那么，GOOSE 0x2005 在文件中要排列在 GOOSE 0x2001 前面

## 3. 注意

在 SCM 文件中 ,APPID 都是 十进制的数，以前我们是十六进制

# 附件1

bool SCL::writeIniFile(const QString iniFileName, const bool isSMV, const quint16 appid)

{

int i;

QSettings \*currentSetting = new QSettings(iniFileName, QSettings::IniFormat);

currentSetting->setIniCodec("gb2312");

QString strSection, strKey, strValue;

strSection = (isSMV ? "SMV" : "GSE") + QString::number(appid, 16).rightJustified(4, '0').toUpper();

QString str;

char buff[255]={0};

QT\_TRY

{

strKey = strSection + QString("/DESC");

currentSetting->setValue(strKey, QString::fromUtf8(currentControlBox.cbDesc));

strKey = strSection + (isSMV ? QString("/svID") : QString("/goID"));

currentSetting->setValue(strKey, QString::fromUtf8(currentControlBox.controlID));

strKey = strSection + QString("/dataset");

str = QString("%1$%2").arg(currentControlBox.reference).arg(currentControlBox.dataset);

currentSetting->setValue(strKey, str);

strKey = strSection + QString("/confRev");

currentSetting->setValue(strKey, QString::fromUtf8(currentControlBox.confRev));

if (!isSMV)

{

strKey = strSection + QString("/gocbReference");

str = QString("%1$GO$%2").arg(currentControlBox.reference).arg(currentControlBox.cbName);

// strcpy(currentControlBox.reference, str.data());

currentSetting->setValue(strKey, str);

}

for (i = 0; i < currentControlBox.m\_lstFCDA.size(); i ++)

{

strKey = strSection + QString("/CHN%1").arg(i+1);

QString phase;

QString strdoda = QString("%1.%2")

.arg(currentControlBox.m\_lstFCDA[i].doName)

.arg(currentControlBox.m\_lstFCDA[i].daName);

int coVA = 1, coD = 1, shift = 0;

float primary = 1.0, secondary = 1.0;

if (strdoda.contains("phsA")

|| QString::fromUtf8(currentControlBox.m\_lstFCDA[i].fcdaDesc).contains("A相"))

phase = "A";

else if (strdoda.contains("phsB")

|| QString::fromUtf8(currentControlBox.m\_lstFCDA[i].fcdaDesc).contains("B相"))

phase = "B";

else if (strdoda.contains("phsC")

|| QString::fromUtf8(currentControlBox.m\_lstFCDA[i].fcdaDesc).contains("C相"))

phase = "C";

else if (strdoda.contains("phsN")

|| strdoda.contains("neut")

|| QString::fromUtf8(currentControlBox.m\_lstFCDA[i].fcdaDesc).contains("零序"))

phase = "N";

else

phase = "L";

if (isSMV)

{

if (QString::fromUtf8(currentControlBox.m\_lstFCDA[i].fcdaDesc).contains(QString("延时"))

|| strdoda.contains("DelayT", Qt::CaseInsensitive))

{

sprintf(currentControlBox.m\_lstFCDA[i].type, "DelayT");

}

else if (strdoda.contains("PhV") || strdoda.contains("PPV") || strdoda.split(".").at(0).contains("Vol")

|| QString::fromUtf8(currentControlBox.m\_lstFCDA[i].fcdaDesc).contains("电压"))

{

coD = 100;

primary = 220000;

secondary = 100;

sprintf(currentControlBox.m\_lstFCDA[i].type, "ACV");

shift = calcShift(primary/secondary, true);

}

else if (strdoda.split(".").at(0) == "A" || strdoda.split(".").at(0).contains("Amp")

|| QString::fromUtf8(currentControlBox.m\_lstFCDA[i].fcdaDesc).contains("电流"))

{

coD = 1000;

primary = 1000;

secondary = 1;

sprintf(currentControlBox.m\_lstFCDA[i].type, "ACC");

shift = calcShift(primary/secondary, false);

}

else

{

sprintf(currentControlBox.m\_lstFCDA[i].type, "Other");

}

}

//fgh,2014-02-09

/\* strValue = QString("%1,%2,%3,%4,%5,%6,%7,%8")

.arg(QString::fromUtf8(currentControlBox.m\_lstFCDA[i].fcdaDesc))

.arg(currentControlBox.m\_lstFCDA[i].type)

.arg(phase).arg(primary).arg(secondary).arg(coVA).arg(coD).arg(shift);

currentSetting->setValue(strKey, strValue);

\*/

str=currentControlBox.iedName+QString::fromUtf8(currentControlBox.ldInst)+"/"+

QString::fromUtf8(currentControlBox.m\_lstFCDA[i].prefix)+

QString::fromUtf8(currentControlBox.m\_lstFCDA[i].lnClass)+

QString::fromUtf8(currentControlBox.m\_lstFCDA[i].lnInst)+"."+

QString::fromUtf8(currentControlBox.m\_lstFCDA[i].doName)+"."+

QString::fromUtf8(currentControlBox.m\_lstFCDA[i].daName);

strValue = QString("%1,%2,%3,%4,%5,%6,%7,%8,%9")

.arg(QString::fromUtf8(currentControlBox.m\_lstFCDA[i].fcdaDesc))

.arg(currentControlBox.m\_lstFCDA[i].type)

.arg(phase).arg(primary).arg(secondary).arg(coVA).arg(coD).arg(shift).arg(str);

currentSetting->setValue(strKey, strValue);

}

strKey = strSection + "/ChnCount";

currentSetting->setValue(strKey, QString::number(currentControlBox.m\_lstFCDA.size()));

strKey = strSection + "/done";

currentSetting->setValue(strKey, "true");

//2014-02-09, 写外部引用 currentControlBox.Inputs->append(QString(DataBuf));

for(i=0;i<currentControlBox.Inputs->count();i++)

{

strKey = strSection + "/input"+QString(i);

strValue=currentControlBox.Inputs->value(i);

currentSetting->setValue(strKey, strValue);

}

currentSetting->sync();

}

QT\_CATCH (...)

{

return false;

}

return true;

}