最终生成如下四个结构,保证MMI组可以用复制粘贴功能提取到这些结构.

1：

{

001, 01, TXT\_NET\_TEST, PLMN\_TEST, //测试卡

460, 00, TXT\_NET\_CMCC, PLMN\_CMCC, //中国移动

460, 01, TXT\_NET\_CUCC, PLMN\_UNICOM, //中国联通

460, 02, TXT\_NET\_CMCC, PLMN\_460\_02, //中国移动

460, 03, TXT\_NET\_CTCC, PLMN\_460\_03, //中国电信

460, 05, TXT\_NET\_CTCC, PLMN\_460\_05, //中国电信

460, 06, TXT\_NET\_CUCC, PLMN\_460\_06, //中国联通

460, 07, TXT\_NET\_CMCC, PLMN\_460\_07, //中国移动

460, 9, TXT\_NET\_CUCC, PLMN\_460\_09, //中国联通

460, 20, TXT\_NET\_CTT, PLMN\_460\_20, //中国铁通

466, 01, TXT\_NET\_FAREASTONE1, PLMN\_FAREASTONE1, //远传电信

...

}

规则：

001 == MCC

01 == MNC

TXT\_NET\_TEST == **PPCI&N.Length > 14 ？ Abrev. Net. Name : PPCI&N**

**(宏定义对应见结构3,如果该宏对应的字符串相同，则该宏也相同。)**

PLMN\_TEST == (MCC MNC) 该宏定义要唯一，对应MCC MNC如果相同，则PLMN\_TEST\_01

2.

typedef enum

{

PLMN\_TEST,

PLMN\_CMCC,

PLMN\_UNICOM,

PLMN\_460\_02,

PLMN\_460\_03,

PLMN\_460\_05,

PLMN\_460\_06,

PLMN\_460\_07,

PLMN\_460\_09,

PLMN\_460\_20,

…

}

规则：

按结构1的顺序罗列即可

3. 对于TEXT定义也要生成如下string,每个text一行,第一列为表一中的text id名,第二列是对应String内容

RES\_ADD\_STRING(TXT\_NET\_T\_MOBILE," T-Mobile")

RES\_ADD\_STRING(xxxxxx," xxx")

规则：

TXT\_NET\_T\_MOBILE：自行定义，保证和" T-Mobile"一一对应。

T-Mobile ：**PPCI&N.Length > 14 ？ Abrev. Net. Name : PPCI&N**

4。凡遇到如下表中的mcc, mnc,要单独生成如下的表。

{

310, 20 , TXT\_NET\_WWC\_1 , PLMN\_WWC\_1 ,

310, 26 , TXT\_NET\_T\_MOBILE , PLMN\_WWC\_7 ,

310, 30 , TXT\_NET\_CENTENNIAL , PLMN\_USA30 ,

310, 90 , TXT\_NET\_AT\_T\_WIRELESS\_1 , PLMN\_310\_90 ,

334, 02 , TXT\_NET\_TELCEL , PLMN\_334\_02 ,

334, 03 , TXT\_NET\_TELCEL , PLMN\_334\_03 ,

334, 04 , TXT\_NET\_334\_04 , PLMN\_334\_04 ,

334, 05 , TXT\_NET\_334\_04 , PLMN\_334\_05 ,

334, 20 , TXT\_NET\_TELCEL , PLMN\_334\_20 ,

334, 30 , TXT\_NET\_MOVISTAR , PLMN\_334\_30 ,

334, 50 , TXT\_NET\_334\_050 , PLMN\_334\_050 ,

338, 50 , TXT\_NET\_DIGICEL , PLMN\_338\_050 ,

338, 70 , TXT\_NET\_706\_01 , PLMN\_338\_070 ,

363, 20 , TXT\_NET\_DIGICEL , PLMN\_363\_020 ,

405, 25 , TXT\_NET\_TATADOCO , PLMN\_405\_25 ,

405, 26 , TXT\_NET\_TATADOCO , PLMN\_405\_26 ,

405, 27 , TXT\_NET\_TATADOCO , PLMN\_405\_27 ,

405, 28 , TXT\_NET\_TATADOCO , PLMN\_405\_28 ,

405, 29 , TXT\_NET\_TATADOCO , PLMN\_405\_29 ,

405, 30 , TXT\_NET\_TATADOCO , PLMN\_405\_30 ,

405, 31 , TXT\_NET\_TATADOCO , PLMN\_405\_31 ,

405, 32 , TXT\_NET\_TATADOCO , PLMN\_405\_32 ,

405, 33 , TXT\_NET\_TATADOCO , PLMN\_405\_33 ,

405, 34 , TXT\_NET\_TATADOCO , PLMN\_405\_34 ,

405, 35 , TXT\_NET\_TATADOCO , PLMN\_405\_35 ,

405, 36 , TXT\_NET\_TATADOCO , PLMN\_405\_36 ,

405, 37 , TXT\_NET\_TATADOCO , PLMN\_405\_37 ,

405, 38 , TXT\_NET\_TATADOCO , PLMN\_405\_38 ,

405, 39 , TXT\_NET\_TATADOCO , PLMN\_405\_39 ,

405, 40 , TXT\_NET\_TATADOCO , PLMN\_405\_40 ,

405, 41 , TXT\_NET\_TATADOCO , PLMN\_405\_41 ,

405, 42 , TXT\_NET\_TATADOCO , PLMN\_405\_42 ,

405, 43 , TXT\_NET\_TATADOCO , PLMN\_405\_43 ,

405, 44 , TXT\_NET\_TATADOCO , PLMN\_405\_44 ,

405, 45 , TXT\_NET\_TATADOCO , PLMN\_405\_45 ,

405, 46 , TXT\_NET\_TATADOCO , PLMN\_405\_46 ,

405, 47 , TXT\_NET\_TATADOCO , PLMN\_405\_47 ,

410, 34 , TXT\_NET\_UFONE , PLMN\_410\_34 ,

704, 30 , TXT\_NET\_MOVISTAR , PLMN\_704\_30 ,

706, 40 , TXT\_NET\_MOVISTAR , PLMN\_706\_40 ,

708, 1 , TXT\_NET\_CLARO , PLMN\_708\_01 ,

708, 20 , TXT\_NET\_744\_04 , PLMN\_708\_20 ,

714, 20 , TXT\_NET\_MOVISTAR , PLMN\_714\_20 ,

740, 10 , TXT\_NET\_CLARO , PLMN\_740\_10 ,

};

规则：

在结构1中，发现上述MCC MNC数据，则另外抽取出来放在结构4，结构1中仍然保持这些数据。

有事找 baokun.cao