Tcp3e主要还是以读文件为准，要修改的配置相对较少，主要如下：

2 //cbSet->cbData[cbCnt]->mode); /\* 2: \*/

58 // cbSet->cbData[cbCnt]->frameInd;

504 // cbSet->cbData[cbCnt]->blockSize; /\*Nominal block size\*/

504 // cbSet->cbData[cbCnt]->numInfoBits; /\*Length does not include CRC bits if crcFlag is set\*/

1 // cbSet->cbData[cbCnt]->mcount;

0 // cbSet->cbData[cbCnt]->crcFlag;

1 // cbSet->cbData[cbCnt]->inOrder;

1. // cbSet->cbData[cbCnt]->outOrder;

TCP3d的数据输入主要是以解调下来映射好的LLR值，其它配置主要如下：

tempCbConfig.mode\_sel = 1; //TCP3\_MODE

 tempCbConfig.lte\_crc\_init\_sel = 0;

 tempCbConfig.NumInfoBits = 504;

 tempCbConfig.SW0\_length = 64;

 tempCbConfig.maxst\_en = 0;

 tempCbConfig.out\_flag\_en = 0;

 tempCbConfig.out\_order\_sel = 0;

 tempCbConfig.ext\_scale\_en = 1;

 tempCbConfig.soft\_out\_flag\_en = 0;

 tempCbConfig.soft\_out\_fmt = 1;

 tempCbConfig.min\_itr = 1;

 tempCbConfig.max\_itr = 8;

 tempCbConfig.snr\_val = 14;

 tempCbConfig.snr\_rep = 1;

 tempCbConfig.stop\_sel = 0;

 tempCbConfig.crc\_iter\_pass = 1;

 tempCbConfig.crc\_sel = 0;

 tempCbConfig.maxst\_thold = 4;

 tempCbConfig.maxst\_value = 2;

 tempCbConfig.ext\_scale\_0 = 24;

 tempCbConfig.ext\_scale\_1 = 24;

 tempCbConfig.ext\_scale\_2 = 24;

 tempCbConfig.ext\_scale\_3 = 24;

 tempCbConfig.ext\_scale\_4 = 24;

 tempCbConfig.ext\_scale\_5 = 24;

 tempCbConfig.ext\_scale\_6 = 24;

 tempCbConfig.ext\_scale\_7 = 24;

 tempCbConfig.ext\_scale\_8 = 24;

 tempCbConfig.ext\_scale\_9 = 24;

 tempCbConfig.ext\_scale\_10 = 24;

 tempCbConfig.ext\_scale\_11 = 24;

 tempCbConfig.ext\_scale\_12 = 24;

 tempCbConfig.ext\_scale\_13 = 24;

 tempCbConfig.ext\_scale\_14 = 24;

tempCbConfig.ext\_scale\_15 = 24;

ptCbData->sw0LengthUsed = tempCbConfig.SW0\_length ;

ptCbData->interFlag = 0;

 ptCbData->sizeINTER = 0;

 ptCbData->inInter = NULL;

/\*我们现不需要软判决输出和状态位输出,所以现把它们设置为0\*/

ptCbData->sdFlag = 0;

 ptCbData->sizeSD = 0;

 ptCbData->sdOffset = 0;

 ptCbData->outSD = NULL;

 ptCbData->refSD = NULL;

ptCbData->stsFlag = 0;

 ptCbData->sizeSTS = 0;

 ptCbData->outSts = NULL;

 ptCbData->refSts = NULL;

**typedef** **struct** cbTestDesc

{

 cbDataDesc tCbData;

 Int32 maxNumCB;

 Int32 mode;

 Int32 doubleBuffer;

 Int32 lteCrcSel;

} T\_DRV\_TCP3D\_cbDesc;

T\_DRV\_TCP3D\_cbDesc gtTcp3dCbDesc;

gtTcp3dCbDesc.mode = CSL\_TCP3D\_CFG\_TCP3\_MODE\_MODE\_SEL\_LTE (0x00000001u);

gtTcp3dCbDesc.lteCrcSel = CSL\_TCP3D\_CFG\_TCP3\_MODE\_LTE\_CRC\_ISEL\_USE\_ZERO (0x00000000u);

gtTcp3dCbDesc.doubleBuffer = CSL\_TCP3D\_CFG\_TCP3\_MODE\_IN\_MEM\_DB\_EN\_ENABLE (0x00000001u);

gtTcp3dCbDesc.maxNumCB =1;

gtTcp3dCbDesc.tCbData.inLLR = 数据输入指针地址(64byte aligned)

/\*这无crc，因此blocksize与NumInfoBits相等\*/

gtTcp3dCbDesc.tCbData.blockSize = tempCbConfig.NumInfoBits = 504;

/\*这在输入LLR值时已把为比特解出，所以LLRoffset为504\*/

gtTcp3dCbDesc.tCbData.llrOffset = 504;

gtTcp3dCbDesc.tCbData.sizeLLR = 1512;

当SW0\_length=64时，15个寄存器的内存配置情况如下



当SW0\_length=128时，15个寄存器的内存配置情况如下

