传入参数：

@主档ID： @INSPECT\_DEV2ID string --去数据库找

@用户: @userName string

逻辑如下：

1. 获得检验单号和检验来源

SELECT

@INSPECT\_CODE=ISNULL(INSPECT\_DEV2.INSPECT\_CODE,'') -- 检验单号

,@INSPECT\_PUR=ISNULL(INSPECT\_DEV2.INSPECT\_PUR,'') -- 检验来源 IQC OQC

FROM INSPECT\_DEV2

LEFT JOIN INSPECT\_FLOW ON INSPECT\_FLOW.INSPECT\_FLOWID=INSPECT\_DEV2.INSPECT\_FLOWID

LEFT JOIN COLUM002 ON COLUM002.COLUM002ID=INSPECT\_DEV2.COLUM002ID

WHERE INSPECT\_DEV2ID=@INSPECT\_DEV2ID

1. 判断检验单是否已经完成

如果 检验单号 不为空 并且 检验来源 不为空 则

@动态表名=“INSPECT\_”+@INSPECT\_PUR

@动态表面ID栏位=@动态表名+“ID”

SELECT STATE FROM @动态表名 WHERE @动态表面ID栏位=@INSPECT\_CODE

如果 STATE=“已完成”

返回错误：”检验单已完成，无法再次产生检验报告” 提出API

（QMS可复用逻辑,产生或更新检验单据）

1. 执行存储过程 DEV2\_GET\_INPECT\_CODE @INSPECT\_DEV2ID,‘COC\_ATTR\_001’

存储过程会返回

如果 返回值前两位=“错误”，则 退出API，将返回值返回

1. 重新获得主档资料

SELECT

@INSPECT\_CODE=ISNULL(INSPECT\_DEV2.INSPECT\_CODE,'') -- 检验单号

,@INSPECT\_PUR=ISNULL(INSPECT\_DEV2.INSPECT\_PUR,'') -- 检验来源 IQC OQC

,@ITEMID=ISNULL(INSPECT\_FLOW.ITEMID,'') -- 物料编码 (有个低代码待办)

,@LOTID=ISNULL(INSPECT\_DEV2.LOTID,'') -- 批次号

,@DOC\_CODE=ISNULL(INSPECT\_DEV2.DOC\_CODE,'') -- 来源单号

,@COLUM002ID=ISNULL(INSPECT\_DEV2.COLUM002ID,'')

,@COC\_ATTR =ISNULL(COLUM002.COC\_ATTR,'') --特殊设定

,@INSPECT\_FLOWID=ISNULL(INSPECT\_DEV2.INSPECT\_FLOWID,'')

FROM INSPECT\_DEV2

LEFT JOIN INSPECT\_FLOW ON INSPECT\_FLOW.INSPECT\_FLOWID=INSPECT\_DEV2.INSPECT\_FLOWID

LEFT JOIN COLUM002 ON COLUM002.COLUM002ID=INSPECT\_DEV2.COLUM002ID

WHERE INSPECT\_DEV2ID=@INSPECT\_DEV2ID

1. 获得一些资料

得到 @COLUM002结果集

SELECT \* FROM COLUM002 where COLUM002ID =@COLUM002ID

得到 @第一个样本ID

SELECT TOP 1 @样本=SAMPLEID FROM INSPECT\_2D WHERE INSPECT\_DEV2ID=@INSPECT\_DEV2ID

得到 INSPECT\_2D异常结果集

SELECT \* FROM INSPECT\_2D

LEFT JOIN COLUM001 ON INSPECT\_2D.LOCATION=COLUM001.COLUM001NAME

WHERE INSPECT\_DEV2ID=@INSPECT\_DEV2ID AND SAMPLEID =@样本

AND COLUM001.COLUM002ID=@COLUM002ID AND COLUM001.COLUM001CODE IS NOT NULL

ORDER BY LOCATION

（QMS可复用逻辑,获得应检样本数）

**得到@检验批次数量**LOT\_QTY

SELECT **@检验批次数量**LOT\_QTY=ISNUL(LOT\_QTY,0) FROM @动态表名 WHERE @动态表面ID栏位=@INSPECT\_CODE

（QMS可复用逻辑,获得应检样本数）

1. 获得应检样本数

由@COLUM002ID获得应检样本数

执行存储过程：GET\_INPECT\_CNT @COLUM002ID,@检验批次数量LOT\_QTY

@应检样本数=结果集CNT列的值

1. 进一步检验

如果 INSPECT\_2D异常结果集 记录数>0 则

返回错误：“当前设备原始LOCATION资料和当前选择的检验项目的检验内容不一致”

1. 更新检验内容的CODE给 INSPECT\_2D

如果 @COLUM002结果集 记录数>0 并且 @INSPECT\_2D异常结果集 记录数>0

（通过LOCATION关联检验内容，更新设备原始记录）

UPDATE INSPECT\_2D SET

COLUM001CODE =COLUM001.COLUM001CODE

,ADD\_VALUE=COLUM001.ADD\_VALUE

FROM INSPECT\_2D

LEFT JOIN COLUM001 ON INSPECT\_2D.LOCATION=COLUM001.COLUM001NAME

WHERE INSPECT\_DEV2ID=@INSPECT\_DEV2ID

AND SAMPLEID =@样本

AND COLUM001.COLUM002ID=@COLUM002ID

AND COLUM001.COLUM001CODE IS NOT NULL

1. 将 INSPECT\_2D 的检验内容（检验位置）传递给QMS

如果 @COLUM002结果集 记录数=0 或者为NULL

执行以下逻辑

|  |
| --- |
| --根据@COLUM002ID 产生检验内容（COLUM001）  --得到第一个样本的所有检验内容  SELECT \* FROM INSPECT\_2D  WHERE INSPECT\_DEV2ID=@INSPECT\_DEV2ID AND SAMPLEID =@第一个样本ID  1.  INSERT INTO COLUM001(COLUM001CREATEDATE,COLUM001CREATEUSER,TENID,COLUM001ID,CUSTOMID,RE,MAX\_VALUE,STD\_VALUE,COLUM001NAME,INSPECT\_LEVELID,OPTIONS,COLUM001CODE,MIN\_VALUE,COLUM002ID,REMARK1,AC,INSPECT\_AQLCODE,REMARK,REMARK2,COLUM0A10,INSPECT\_PLANID) VALUES (CONVERT(VARCHAR(20),GETDATE(),120),@CREATEUSER,@TENID,@COLUM001ID,@CUSTOMID,@RE,@MAX\_VALUE,@STD\_VALUE,@COLUM001NAME,@INSPECT\_LEVELID,@OPTIONS,@COLUM001CODE,@MIN\_VALUE,@COLUM002ID,@REMARK1,@AC,@INSPECT\_AQLCODE,@REMARK,@REMARK2,@COLUM0A10,@INSPECT\_PLANID)'  @CREATEUSER='system'  ,@TENID='001'  ,@COLUM001ID=newid()  ,@CUSTOMID=''  ,@RE=''  [,@MAX\_VALUE='0.2'](mailto:,@MAX_VALUE='0.2') --INSPECT\_2D.VALUE2  ,@STD\_VALUE='1' --标准值:INSPECT\_2D.VALUE1  ,@COLUM001NAME='尺寸1' --INSPECT\_2D.LOCATION  ,@INSPECT\_LEVELID=''  ,@OPTIONS=''  ,@COLUM001CODE='A01' --第一个A01，第二个A02，依次类推  [,@MIN\_VALUE='0.1'](mailto:,@MIN_VALUE='0.1') --INSPECT\_2D.VALUE3  ,@COLUM002ID='INSPECT\_ZONE\_019' --传上文得到的@COLUM002ID  ,@REMARK1='二次元' --固定值 二次元  ,@AC=''  ,@INSPECT\_AQLCODE=''  ,@REMARK=''  ,@REMARK2=''  ,@COLUM0A10='Num' --固定值“Num”  ,@INSPECT\_PLANID='c6cae8ea-24e0-4fbe-ac6e-775843549e5b'  --固定值“c6cae8ea-24e0-4fbe-ac6e-775843549e5b”  --（32个检验样本的抽检计划ID）   1. 回写A01的编码给原始资料   UPDATE INSPECT\_2D SET COLUM001CODE=@COLUM001CODE WHERE INSPECT\_2DID=@INSPECT\_2DID |

1. **将INSPECT\_2D实际值传入QMS**

1.获得INSPECT\_2D实际测量的样本数量

SELECT @CNT=COUNT(DISTINCT SAMPLEID) FROM INSPECT\_2D WHERE INSPECT\_DEV2ID=@INSPECT\_DEV2ID

**2.**如果 @CNT>0 则

**2.1 删除QMS中的记录：@COLUM002ID @INSPECT\_CODE**

**DELETE INSPECT\_ZONE WHERE COLUM002ID=@COLUM002ID AND INSPECT\_CODE=@INSPECT\_CODE**

2.2 开始同步记录

SELECT SAMPLEID FROM INSPECT\_2D

WHERE INSPECT\_DEV2ID=@INSPECT\_DEV2ID GROU BY SAMPLEID

循环每个SAMPLEID

BEGIN

SELECT COLUM001CODE AS 检验内容编码

,MAX(VALUES) AS 检验值

FROM INSPECT\_2D WHERE INSPECT\_DEV2ID=@INSPECT\_DEV2ID

AND SAMPLEID =循环中的SAMPLEID

GROUP BY COLUM001CODE

ORDER BY COLUM001CODE

1. 循环结果集

BEGIN

@变量SELECT +=”,”+检验内容编码

@变量VALUES+=”,”+检验值

END

1. 插入检验结果

INSERT INSPECT\_ZONE (INSPECT\_ZONECREATEUSER,INSPECT\_ZONECREATEDATE,INSPECT\_ZONEID,INSPECTTYPE,COLUM002ID,CUSTOM\_ITEMID,LOTNO,INSPECTCODE,PCSCODE,COLUM001ID,ISAUTO+@变量SELECT ) VALUES (

@userName --传参@userName

,CONVERT(varchar(20),GETDATE(),120)

,newid()

,'OQC' --@INSPECT\_PUR

,''INSPECT\_ZONE\_012'' --@COLUM002ID

,''11'' --@ITEMID

,''11'' --@LOTID

,''OQC\_2025030001'' --@INSPECT\_CODE 检验单号

,'' --样本序号 (第1个SAMPLEID 给1，第2个给2,依次类推)

,’0’ --固定给 0

,@变量VALUES )

END

1. **让QMS产生随机值**
2. 如果 @应检样本数>@CNT

则 @产生样本数量=@应检样本数-@CNT

1. 产生检验随机值(会清除曾经已产生的随机值）

（以下逻辑建议额外写通用存储过程给调用，我连合并代码的功夫都省了 存储过程名称GET\_INSPECT\_RANK）

|  |
| --- |
| 传入参数：  @COLUM002ID(检验项目)  @INSPECT\_CODE(检验单号)  @产生样本数量  @INSPECT\_PUR --检验类别  @userName  通用逻辑如下：   1. 删除QMS中的随机记录：   DELETE INSPECT\_ZONE WHERE COLUM002ID=@COLUM002ID AND INSPECT\_CODE=@INSPECT\_CODE AND  AND ISAUTO=”1”  1.获得结果集A：  SELECT \* FROM COLUM001 WHERE COLUM002ID=@COLUM002ID  2.获得@已存在记录数：  SELECT @已存在记录数=COUNT(1) FROM INSPECT\_ZONE WHERE INSPECTCODE=@INSPECTCODE  3.获得@ITEMID，@LOTID  @动态表名=“INSPECT\_”+@INSPECT\_PUR  @动态表面ID栏位=@动态表名+“ID”  SELECT @ITEMID=ITEMID，@LOTID=LOTID FROM @动态表名 WHERE @动态表面ID栏位=@INSPECT\_CODE  循环次数：@产生样本数量  BEGIN  1.循环结果集  BEGIN  获得：@标准值 结果集A.STD\_VALUE  @上公差 结果集A.MIN\_VALUE  @下公差 结果集A.MAX\_VALUE  @上下公差余量=结果集A.ADD\_VALUE  得到：@实际值：  如：标准值=1.08，上公差0.01 下公差 0.03 参数上下公差余量=0.001  则随机范围是 （1.08-0.03+0.001~1.08+0.01-0.001）  BEGIN  @变量SELECT +=”,”+结果集A.COLUM001CODE  @变量VALUES+=”,”+实际值  END  END  2.插入检验结果  INSERT INSPECT\_ZONE (INSPECT\_ZONECREATEUSER,INSPECT\_ZONECREATEDATE,INSPECT\_ZONEID,INSPECTTYPE,COLUM002ID,CUSTOM\_ITEMID,LOTNO,INSPECTCODE,PCSCODE,COLUM001ID,ISAUTO+@变量SELECT ) VALUES (  @userName --传参@userName  ,CONVERT(varchar(20),GETDATE(),120)  ,newid()  ,'OQC' --@INSPECT\_PUR  ,''INSPECT\_ZONE\_012'' --@COLUM002ID  ,''11'' --@ITEMID  ,''11'' --@LOTID  ,''OQC\_2025030001'' --@INSPECT\_CODE 检验单号  ,'' --样本ID：@已存在记录数+循环次数  ,’0’ --固定给 0  ,@变量VALUES )  EMD |

1. **产生【CPK-扩展项目】的随机值**
2. 得到CPK-扩展项结果集

SELECT COLUM002ID FROM COLUM002 WHERE INSPECT\_FLOWID=@INSPECT\_FLOWID WHERE COC\_ATTR=”[COC\_ATTR\_002](javascript:;)”

1. 循环CPK-扩展项结果集

BEGIN

1. （QMS可复用逻辑,获得应检样本数）

由CPK-扩展项结果集.@COLUM002ID 获得应检样本数

执行存储过程：GET\_INPECT\_CNT @COLUM002ID,@检验批次数量LOT\_QTY(前文获取过）

@**CPK-扩展项目**应检样本数=结果集CNT列的值

1. 产生检验随机值（调用十一点提到的GET\_INSPECT\_RANK存储过程）

传入参数：

@COLUM002ID(检验项目) --CPK-扩展项结果集.@COLUM002ID

@INSPECT\_CODE(检验单号) (前文获取过）

@产生样本数 --@**CPK-扩展项目**应检样本数

@INSPECT\_PUR --检验类别 (前文获取过）

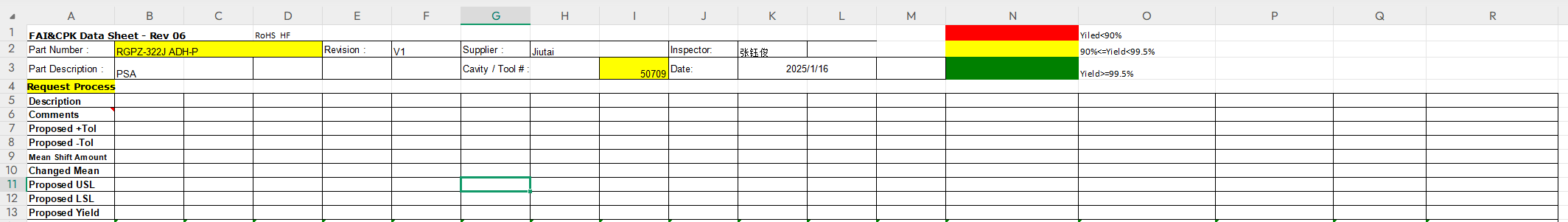
@userName

END

1. **产生CPK报告**
2. 如下图：

1.1 【B2】 =@ITEMNAME 【I3】 =@ITEMID

1.2 5~14行的列数边框数=后面逻辑给值，此处待定



2.15~63行是动态的，逻辑如下：

* 1. 这个区域内，分两个区块，区块0=A列 区块1=B~M列 区块2= N~R 列

并且 区块1和区块2的列数和行数都是动态的

区块0只有1列，区块0的行数随区块1的行数产生序号

* 1. 区块1：

2.2.1 通用获取数据方法1

（以下逻辑建议额外写通用存储过程给调用，我连合并代码的功夫都省了 存储过程名称GET\_INSPECT\_LIST）

|  |
| --- |
| 传入参数：  @COLUM002ID --需要查询的检验项目  @INSPECT\_CODE --检验单号  @INSPECT\_PUR --检验类别  通用逻辑如下：  1.获得结果集A：  SELECT \* FROM COLUM001 WHERE COLUM002ID=@COLUM002ID  2.循环结果集A  BEGIN  获得：@变量SELECT +=”,”+结果集A.COLUM001CODE  获得：@变量SELECTB +=”,”+结果集A.COLUM001CODE.REPLACE(“A”,”B”)  END  3.组装SQL  SQL=“SELECT INSPECT\_ZONECREATEUSER,INSPECT\_ZONECREATEDATE,INSPECT\_ZONEID,INSPECTTYPE,COLUM002ID  ,CUSTOM\_ITEMID AS ITEMID ,LOTNO,INSPECTCODE,PCSCODE,COLUM001ID,ISAUTO”  +@变量SELECT+@变量SELECTB  +“FROM INSPECT\_ZONE  WHERE COLUM002ID=@COLUM002ID AND INSPECT\_CODE=@INSPECT\_CODE AND INSPECTTYPE=@INSPECT\_PUR ORDER BY PCCODE”  4.执行  5.返回查询结果 |

其中：

返回的所有列名以A开头的列数量，就是区块1的列数

返回的所有列名以A开头的列名，就是区块1第15行的列名

返回的数据就是32~63行区域的内容（行数依照查询的结果展示）

2.2.2 获得标准值结果集：

SELECT COLUM001CODE

,STD\_VALUE --标准值

,MIN\_VALUE --上公差

,MAX\_VALUE --下公差

,REMARK1 --检验方法

FROM COLUM001 WHERE COLUM002ID=@COLUM002ID

其中：返回的列COLUM001CODE 与 2.2.1的结果集中以A开头的列名相对应

然后把标准值，上下公差 按照原文档的要求分别产生18~31行的记录

检验方法对应第17行

2.3 区块2：

2.3.1 得到CPK-扩展项结果集

SELECT COLUM002ID FROM COLUM002 WHERE INSPECT\_FLOWID=@INSPECT\_FLOWID WHERE COC\_ATTR=”[COC\_ATTR\_002](javascript:;)”

2.3.2 循环CPK-扩展项结果集中每个COLUM002ID

调取：GET\_INSPECT\_LIST的逻辑执行，返回的结果就是区块2的内容

呈现方式参考区块1，唯一的区别在于：CPK-扩展项的结果会有多个，如果是多个就是列叠加

2.3.1 17~31行参考区块1

2.4 前文1.2中待定的列数= 区块0+区块1+区块2的总列数

1. **产**生结束以后将文件存下文件服务器

DELETE SCANDOC WHERE INSPECT\_DEV2I=@INSPECT\_DEV2I

INSERT SCANDOC(TENID,SCANDOCID,SCANDOCCODE,SCANDOCNAME,DOCTYPE,PEOPLEID

,createdate,SCANDOC\_user,INSPECT\_DEV2ID)

VALUES ('001',@SCANDOCID,@SCANDOCID,@SCANDOCNAME,@DOCTYPE,@PEOPLEID

,CONVERT(VARCHAR(20),GETDATE(),120),@PEOPLEID,@INSPECT\_DEV2ID）

@DOCTYPE", --写死‘CPK报告’

@SCANDOCNAME --文件服务器/2D/品号/品号\_批次号\_测量时间.xls

@PEOPLEID --检验单号

@SCANDOCID --newid()

@INSPECT\_DEV2ID --整个API传递进来的ID