**语音TDMA原理设计**

## 一、概述

本文档主要描述，5.8G语音答题器在无线传输数据层上的处理原理，具体协议细节另文介绍。

方案设计要解决的问题，就是无线数据信道上怎么高效传递数据，以满足30路语音传输的速度要求。同时，针对数据量少的单包、数据量大的大批数据，上传和下载两个方向做具体处理，也就是满足原有投票系统的数据传输要求和优化。

大方向是把原来的竞争式提交数据，改成无冲突的分时复用方式。TDMA(Time Division Multiple Access) 时分多址的基本原理是把时间分割成周期性的帧(Frame)，每一个帧再分割成若干个时隙向基站发送信号，在满足定时和同步的条件下，基站可以分别在各时隙中接收到各移动终端的信号而不混扰。同时，基站发向多个移动终端的信号都按顺序安排在预定的时隙中传输，各移动终端只要在指定的时隙内接收，就能在合路的信号中把发给它的信号区分并接收下来。

在硬件平台上，我们选择了支持空中4Mbps的芯片（老芯片2Mbps），同时每包最大64字节（原32字节）可提高传输打包的效率，也是系统的选型关键。

TDMA底层需要时间的同步，这一块我们在老2.4G系统上已经有实现经验，就是以基站信标产生的中断做基准时间，调整定时器，来做同步，难度不大，会有抖动，实测调整时序让相互之间不影响即可。

原理是简单的，我们核心问题是，无线传输单次传输是不可靠的，语音数据肯定涉及到重传，怎么处理重传？然后单多包上传下载怎么处理？

最后怎么尽量兼容原有体系减少键盘、基准、SDK的工作量？（放协议设计讨论）

## 二、方案设计

关于语音的查询和查询，有固定时序重传和动态时序重传方案。

前者就是每个键盘分配固定的时间片，在里面完成传输和重传，优点是简单，缺点是每个键盘可给的重传的时间是固定的，实际上不会每个键盘都丢包（物理信道保证），不丢包的键盘还分配时间，就浪费了，信道利用率不高。

后者是先给固定的时间传输，然后有动态可调整的重发周期。重发周期是可以指定某个键盘，重新传哪些包的。系统总体上根据实际测试经验，安排可以最长的重传周期即可，这个周期会比每个键盘都分配固定时间要高效，省出的时间可以让键盘数目增加，甚至可以多给1个键盘多分配时间，这样一定情况下可减少语音的丢包率。

上述最长的重传周期的具体时间长度，根据键盘数量、实际测试经验（信道通讯质量）来安排，重传时长内没完成重传的语音数据，是可以丢弃的，先保证总体语音的流畅。丢1包大约20ms，不会很明显，丢包的数据可以用上一个20ms的数据。

所以使用动态时序重传语音数据方案。

TDMA还涉及一个组网操作，主要是基站给节点分配内部编号，和告诉基站我来了。这主要关系到TDMA的效率，如果一个键盘没开机，或者已经配对到另外基站了，基站不知道，还一直查询，那么信道资源就被占用了。

所以，一般情况下，键盘和基站之间的通讯流程是，N个数据（查询和提交）周期，加1个组网周期。如下图：

。。。。

数据周期和组网周期，或者我们约定所有周期的时间都是固定的，时长我们定义为1个时隙，这样，键盘可以严格和基站同步，中间也好做无线休眠省电。

数据周期内部，我们计划沿用1问10答的时间片模式，1问是指基站发出查询数据指令，告诉哪些键盘要传什么数据，这里是10个键盘。然后后面10个时间片，就是被询问的键盘在指定的时间片里面回答，例如第2个键盘就在第2个时间片里面回答。如图：

。。。。

1问也可以是其他指令，不限定是数据查询。然后键盘根据指令做成合适的反应。例如指令甚至可以设计支持到，第1个时间片键盘1提交数据，时间片2是基站给键盘2发送信息。

当1问是其他类型的指令时候，对应的10答时间周期就可以针对具体业务做新的处理，例如怎么传输一个大文件到键盘上做升级。

也就是说，平常N个数据周期和组网周期后，根据业务需要，后面可以插入功能周期，功能周期完成再继续数据周期。

这些功能周期，时长也尽量和数据周期等长，里面11个时间片，第1个一般是基站发指令。

这个时间片，可以暂定2.4G系统常用的2.5ms，不要太小，工程实现上不好太精准，两边要留抖动范围。根据前期实验，5.8G芯片在STM32的驱动下，读写64字节FIFO时间80us左右，64字节空中时间150us，引导头40us，合起来发送时间270us左右。这样，2.5ms内可以发7-8包64字节的数据，这是和2.4G体系区别较大，也就是，每一个时间片，基站和键盘都可以连续发多包。那么，基站发信标的时候，是可以发多条指令的，而不是2.4G时候的1条指令。

最终系统是1问10答，还是N答，可以调整。时间片到底多少ms，能发几个数据包，都根据工程实现最终确定和调整。后面原理讨论，暂时按1问10答，时间片2.5ms，发7个数据包来举例说明。

## 三、业务处理

原有投票系统，不支持语音的情况下，对数据量大小进行了单包和多包划分，单包例如单选多选这种简单数据，多包是例如一套测验题目的数据。然后根据传输的方向划分了上传和下载，这样就数据传输而言，有单包上传、单包下载、多包下载、多包上传4种数据业务。TDMA系统要对这4种业务进行实现和优化。

以上业务是数据不可丢失错误的，但实时性要求不高。语音数据对实时性要求高，对带宽效率要求高，但可以丢部分数据。

单包上传和语音数据传输，我们前一章定义为数据周期。单包下载、多包下载、多包上传，就是所谓功能周期。

所以键盘和基站的通讯流程，平常就是连续的数据周期，如果有特殊业务，就是多个功能周期和1个数据周期交替。如图：

。。。。

### 3.1语音传输（数据查询和提交）

要解决传输和重传。1问10答，每个答的时间片就可以传输语音数据。

重传周期，只要基站的指令，包含要重传的键盘SN号，以及重传哪些片段即可。重传哪些片段可以用bit位控制。

所以语音传输的过程是，30路语音的话，基站先询问第一组10个键盘，用SN号或者内部编号指定要应答的键盘，然后指定的键盘在10个时间片内分别传输语音数据，每个时间片可以传输7片64字节，语音数据带序号1到7，这样，基站就可以判断哪些序号的语音包没收到。

然后基站询问第2组10个键盘，然后询问第3组10个键盘，键盘数量少可能就只问到第2组。然后是安排1到几个重传周期。

问多少组键盘，以及安排几个重传周期，要根据测算和实际通讯效果，但合起来的时间周期，必须要小于这个周期内能传输的单个键盘语音的时间长度。

### 3.2组网

组网可以给与类似语音传输的专门周期。例如定时查询一段时间语音（比如10个语音传输周期）后插入1个组网周期。

也可以在1问10答后安排1个时间片给组网申请用，加入机会多一些，信道1/12的时间用于组网。

各有优劣。

目前先用后面方式。所以正常1周期变成了1问10答1组网。

组网要求基站发送的信标里面，能告诉键盘已经是新的一轮组网了，可以用1字节的组网序号，基站上电开关1次或者软件控制开始新一轮，组网号就变化，然后键盘就知道要重新组网申请。

（还有一种方式，独立的组网周期，组网周期只给1个时间片，时间片前面3包是基站广播现在记录的所有SN，后面4包提交申请，但基站不应答，前面SN就相当于应答。缺点是组网周期的插入会导致所有键盘无线多开机2.5ms，也不是大问题。）

### 3.3投票信标

投票信标用于告诉键盘现在的投票参数，基站1问里面可以发7片数据包，有1包用于投票信标即可。

### 3.4单包上传

单包上传用于键盘提交简单的投票数据，数据不可丢弃，是需要基站确认收到的，如果没有，键盘下一次重新发送。

单包上传可以安排独立的周期，例如1问9答1确认，但数据量少，1次数据才占用每次可发7包的1/7，太浪费。

所以考虑和语音数据包可以混合送，也就是3.1节的10答的时候，键盘如果有语音数据先提交语音数据，如果语音数据不满7片，就可以用1片来提交单包数据。

单包数据是需要基站确认的，确认包就放到基站1问的时间片里面，有7片时间，包含1个确认包即可。注意的是，确认包里面的SN号，是上一个周期，或者是上一组查询时候，提交了单包数据的SN。如果没收到单包数据，平常确认包里面SN号全部空。

### 3.5单包下载

单包下载是给指定键盘1个简短的数据，例如写1个姓名，必须保证键盘有应答收到，或者超时没响应。

一种方式是沿用2.4G老体系的方法，SDK向基站传输要单包下载的时候，基站等当前数据查询周期完成，然后暂数据查询流程，转发指令，然后监听键盘应答，有应答就向SDK转发键盘应答数据。这种方式简单，业务限制上就是不允许语音传输的时候做单包下载。

针对2.4G体系的优化可以是，由基站完成多次询问尝试，然后1个周期或者2个周期键盘没应答就自动给SDK一个超时报告，然后自动转入数据查询周期。另外一个可以优化的地方，就是可以支持超出1包的数据，例如1个144字符的短信息1次完成。

还有1种方法是，把下载指令放到基站1问的时间片里面，7片中的1片，然后键盘在应答时间片里面的7包数据里面，用1包做应答。这种方法稍微复杂一点，暂不采用。

### 3.6多包下载

多包下载就是给1个键盘或者多个键盘下载一个比较大的数据文件。

多包下载，沿用2.4G体系，基站下载的方式，SDK先把一段数据传给基站，然后基站再广播给键盘，并查询键盘是否接受完毕。一段都接收完成，然后再到下一段。

可以优化的是，数据广播也限定在一个周期内完成，就是1指令11下载（最后1个组网时间片也可用于下载），后面11个时间片都用于广播数据，每片至少可以发7包，每包可含至少48字节数据，数据吞吐量比2.4G大很多，就是下载更快。查询键盘收齐情况也可以优化到1个周期，时间片对分1半做应答的话，可以一次做20个键盘的查询。

如果时间问题，前期也可沿用2.4G方式，暂不优化。

### 3.7多包上传

多包上传就是键盘要上传一大段数据，例如一个家庭作业。

继续沿用2.4G的流程，键盘先用单包上传方式提交多包申请，告诉基站和SDK，要传输多大的数据。SDK再根据现在业务控制是否启动接收。老体系是SDK主导的询问流程，SDK每次问键盘要多片数据，直到所有数据接收完。基站只是双向数据的转发。

可以优化的是，改成基站主导询问过程，SDK只是发启动接收命令。另外一个优化就是在1周期内完成，1问11提交，11个提交时间片都是指定的那个键盘做数据提交。也可以12时间片每个时间片内都是先基站问，然后键盘发数据，由于时间片是允许发7个包的，那么第1个包用于基站问，后面6个包用于键盘应答。后一种方法可能灵活些。

### 3.8总结

空闲、投票中、语音传输中的周期（定义为数据周期）是：

第1组传输、第2组传输、第3组传输、第1批重传、第2批重传。

具体传几组，重传几次，可具体调整。

有单包下载、多包下载、多包上载时候，可以先完成这些特殊周期（定义为功能周期），然后再回到数据周期，也可以N个功能周期加1个数据周期，功能都完成后再全部数据周期。举例：

多包下载、多包下载、多包下载、数据查询、多包下载(完成)、数据查询、数据查询

每个数据周期组成：

1基站询问时隙+10个键盘应答时隙+1个组网时隙

基本上每个功能周期都是：

1个基站指令时隙+11个基站下载或者键盘上传数据时隙

基站询问时隙里面可以发7个包，考虑到键盘接收的可靠性，尽量发满7次，可以安排询问和重传指令3次，发投票信标2次，确认包2次。

在10个键盘应答时隙，每个键盘都在指定的时间片里面做应答提交数据。键盘可以提交单包数据，也可以提交语音数据，最多可能7包，也可以不提交任何数据。对于语音数据，基站要判断是否有丢包，以便在重传周期时候安排重发。对于单包数据，基站要记录SN，以便在下一个周期的询问时隙里的确认包发送里面发送。

功能周期里面，键盘根据基站指令时隙里面的指令，做具体反应，接收基站数据或者发送特定数据。

具体到键盘和基站SDK协议细节，尽量保持兼容老体系。

为减少基站处理时间（TDMA要求无线数据尽快读完），可能取消基站的结果打包，会把每个键盘的原始数据都直接传给SDK，给SDK做结果解析。

### 3.9笔迹数据传输（不能丢）

笔迹数据和语音数据的区别是，笔迹数据不能丢。

要双方知道是否丢包，笔迹数据包要带编号，这样，后面包号来了能知道前面是否丢包。加上本次说要上传几个包，也能知道丢包。

数据不能丢，最终是要键盘知道哪些包丢了，下次询问的是否重传，所以确认包里面要带基站收到哪些信息。由于一次上传最大7包，所以确认包可以用bit位表示哪些包收到了，键盘再细分那个包里面哪些笔迹数据小包没成功，下次询问再传。但笔迹小包必须是依次序号组织的，也就是SDK那边能按顺序重排，因为笔迹数据是带先后的，不能乱。原来的用于单包数据提交的确认包，格式需要改一下。

考虑到数据不能丢，所以语音流程里面的查询重传机制可以不用，重传机制保障了本次数据更有效，但占用了时间的，相当于降低了速率。另外本时间段重传，是不能保证这次全部收齐，还是得用确认包机制。

笔迹数据一般比语音数据量少，所以每个周期上传7片数据里面，可以重复，提高基站接收端成功率。

笔迹数据又是动态增加的，不像语音数据，每个时间片都一样长度。所以最好还支持动态长度。当然也可按语音时间片段处理，有多少传多少。

### 3.10题型数据、单题题干、答案下发

这些数据的特定是数据量有一定量，但广播一段时间可以确保每个键盘都能收到，不需要像多包下载需要每个键盘都查询核对是否收齐。

还是按教育协议的2.4章节基站广播信息信标、2.2.13节执行。

## 四、针对60路通讯的设计

由于5.8G系统实现上最终单路只支持30路同时通讯，所以硬件上增加一路5.8G。信息的汇总可以在基站主控MCU汇总，统一传给USB。但业务处理上也需要增加处理如下：

### 4.1组网

组网应答里面要告诉键盘是在主频点还是副频点，基站可能还需要平均分配，例如主频点1个键盘，下一个键盘分配到副频点，也可以超出30路后再分配到副频点，甚至30路后再启用副频点，平常不占用。

### 4.2信标广播、单包上传

信标广播和单包上传，由于要明显支持60路，所以肯定是2路5.8G同时执行。

包括题型数据控制的信标广播，也是2路同时执行。

### 4.3单包下载、多包下载、多包上传

单包下载，没有必要通知键盘换到主频点，所以，单包下载，基站根据SN号查询在哪一路5.8G，单独执行，也可以直接双路同时广播。应该是双路同时广播简单一些。

多包下载，例如下载升级包，可以先通知副频点键盘切换到主频点来，但貌似还麻烦一些。还是先2路同时广播数据，询问键盘的时候，也同时询问2路。有数据包键盘没收到的，也同时2路广播。

多包上传，特别是A1键盘要求支持家庭作业。前面已经说尽量基站来取多包（以前基站没实现），基站来取包前期还是只实现2路同时询问。

如果提升效率，多包下载和多包上传的询问和重发，2路通讯可以分开执行，效率提升理论1倍，例如收作业速度。如果从线程的角度看，2路包括发信标确实都可以不同步，只是按自己的状态机单独执行。例如单包下载处理，有新的单包下载指令下来，判断SN在哪一路，然后往哪一路的命令队列里面扔。另外一路没有事情就做正常的流程。

前期看，2路同时执行相同的事情，实现简单一些。

## 【版本修改说明】

V0.5 基础版本

V0.6 增加3.9节笔迹数据怎么不丢；3.10节基站信息广播信标，以支持200题型、单题题干信息、测验答案下发；增加第四章节60路怎么处理