

卫星波束复用控制系统设计与实现

杨一

2025 年 5 月 26 日

摘要

本文提出并实现了一种基于Python语言和Tkinter图形界面库的简化模拟卫星移动通信系统。系统能够模拟信号输入的数模转换过程，设计了一套自创的编码系统与通信协议，可以模拟通信传输，解调与调制过程。可以实现对移动通信交换子系统的模拟，此外，还完成了模拟多波束数量配置复用、模拟功率调节、频率资源优化分配及干扰预警功能。具备较强的实时交互能力和系统可扩展性。通过Matplotlib实现了以上功能的动态可视化，为用户提供直观的波束管理界面。系统设计过程强调模块化架构和多线程事件响应机制，保证了系统的稳定运行和良好的用户体验。通过多轮实验验证，系统在波束配置准确性、协议模拟准确性，通信模块传输模拟真实性，模块调节响应速度等方面表现良好。

1 各模块设计流程简要描述

本系统设计流程严格遵循软件工程模块化开发理念，整体架构划分为四个主要模块：ADDC模拟模块、通信编码与协议实现与解译模块、模拟传输，调制解调模块、波束复用与管理模块。设计过程从用户需求分析出发，逐步细化各功能模块，确保各模块职责单一且接口清晰，利于后期维护与升级。

1.1 ADDC模拟模块设计说明

1.1.1 功能概述

该模块实现了卫星通信系统中的模拟/数字信号转换功能，主要包含以下核心功能：

- 模拟信号生成与显示（正弦波、方波、三角波）
- AD转换与数字量化
- PWM调制与解调
- 信号重建与滤波

1.1.2 系统架构设计

系统采用三区块设计结构：

1. 信号生成区

- 固定频率10kHz波形生成
- 可调节信号幅值（0.1-10.0V）
- 支持多种波形选择与参数调节

2. AD转换区

- 固定8位量化精度
- 可调采样率（0.1-5000kHz）
- 可配置参考电压（1.0-20.0V）
- PWM调制（0.05-5.0kHz）

3. 信号重建区

- 数字滤波实现
- 支持移动平均与巴特沃斯滤波
- 实时信号质量监测（RMSE与SNR）

1.1.3 关键技术实现

AD转换过程 AD转换采用8位量化，通过以下步骤实现：

$$D_{out} = round(\frac{V_{in} + V_{ref}}{2V_{ref}} \times (2^8 - 1)) \quad (1)$$

其中：

- V_{in} ：输入电压
- V_{ref} ：参考电压
- D_{out} ：输出数字量

PWM调制技术 采用基于占空比调制的PWM技术：

$$PWM_{duty} = \frac{D_{in}}{2^8 - 1} \times 100\% \quad (2)$$

其中 D_{in} 为输入数字量。

1.1.4 性能指标

系统主要性能指标如下：

- 采样率范围：0.1-5000kHz
- 信号幅值范围：0.1-10.0V
- PWM频率范围：0.05-5.0kHz
- 量化精度：8位
- 参考电压范围：1.0-20.0V

1.1.5 用户界面设计

系统提供直观的图形化界面，包含：

- 三通道实时波形显示
- 参数实时调节控件
- 信号质量实时监测
- 运行状态控制按钮

1.1.6 系统特色

1. 实时性能优越，支持50ms周期更新
2. 参数调节灵活，支持实时预览
3. 信号质量监测全面，包含RMSE与SNR指标
4. 故障容错设计，滤波器异常时自动切换备用方案

1.2 通信编码与协议实现与解译功能概述

该模块实现了卫星通信系统中的Turbo编码功能，主要包含以下核心功能：

- 随机数字信号生成（16位）
- Turbo编码实现（包含交织、RSC编码）
- 实时数据监测与显示
- 系统运行状态记录

1.2.1 系统架构设计

系统采用三层架构设计：

1. 编码器核心层

- Turbo编码器实现
- 交织器实现
- RSC编码器实现

2. 数据处理层

- 随机数据生成
- 数据格式化处理
- 编码结果组装

3. 用户界面层

- 图形界面展示
- 数据流监控
- 运行状态显示

1.2.2 关键技术实现

交织算法 采用简单周期交织方案：

$$P_{new} = (3i + 7) \bmod L \quad (3)$$

其中：

- P_{new} ：交织后位置
- i ：原始位置
- L ：数据长度

RSC编码器 采用基于状态机的递归系统卷积编码：

- 2比特状态寄存器
- 单比特奇偶校验输出
- 状态更新方程： $S_{new} = ((S_{old} \ll 1) | bit) \& 0x3$

1.2.3 性能指标

系统主要性能指标如下：

- 输入数据长度：16位
- 编码率：1/3（系统位+两组校验位）
- 状态数：4状态
- 更新周期：5秒/次

1.2.4 用户界面设计

系统提供图形化界面，包含：

- 数据流实时显示
- 编码结果可视化
- 监控日志展示（最近10条）
- 运行状态切换控制

1.2.5 系统特色

1. 自动化运行，支持5秒自动更新
2. 实时监控功能，可随时开启关闭
3. 支持手动触发数据更新
4. 监控日志滚动更新机制

1.3 模拟传输，调制解调模块说明

1.3.1 功能概述

该模块实现了卫星通信系统中的协议仿真功能，主要包含以下核心功能：

- 通信数据帧的生成与发送
- 数据帧的解析与解析
- BPSK调制与解调
- 通信过程实时监控

1.3.2 系统架构设计

系统采用五区块设计结构：

1. 传输交换模块

- 多信道管理（4信道）
- 传输延迟模拟
- 路由历史记录

2. 帧生成器模块

- 协议帧构建
- 参数配置界面
- 校验和计算

3. 帧分析器模块

- 协议帧解析
- 数据提取显示
- 错误检测

4. 监控日志模块

- 实时日志记录
- 传输状态监控
- 日志导出功能

5. 调制解调模块

- BPSK调制实现
- 波形可视化
- 信号重建

1.3.3 关键技术实现

帧格式定义 采用固定格式的通信帧结构：

$$Frame = Header_{meow} + Addr_{src} + Addr_{dst} + Type + Payload + Checksum + Footer_{mikuma} \quad (4)$$

帧结构组成：

- Header: 固定为”meow”的16进制表示
- Addresses: 4字节源目地址
- Type: 2字节帧类型标识
- Payload: 可变长度数据负载
- Checksum: 2字节校验和（可选）
- Footer: 固定为”mikuma”的16进制表示

延迟模拟 采用随机延迟模拟真实传输：

- 单跳延迟范围：0.5-1.0秒
- 双跳延迟范围：1.0-2.0秒

1.3.4 性能指标

系统主要性能指标如下：

- 支持信道数：4
- 地址空间：16位
- 帧类型数：3种
- 最大传输单元：无限制
- 校验方式：简单校验和

1.3.5 用户界面设计

系统提供图形化界面，包含：

- 多标签页界面布局
- 实时状态显示栏
- 滚动文本显示区
- 参数配置面板
- 波形显示画布

1.3.6 系统特色

1. 完整协议栈模拟实现
2. 直观的数据可视化展示
3. 灵活的参数配置选项
4. 详细的监控日志记录
5. 模块化设计便于扩展

1.4 波束复用与管理模块说明

1.4.1 功能概述

该模块实现了卫星通信系统中的波束复用控制功能，主要包含以下核心功能：

- 多波束配置管理（4/7/12波束）
- 波束功率动态调节
- 频率资源分配优化
- 干扰监测与预警

1.4.2 系统架构设计

系统采用四区块设计结构：

1. 波束管理模块

- 波束数量配置（4/7/12）
- 波束位置计算
- 波束状态管理

2. 资源控制模块

- 频率资源分配
- 功率级别调节
- 资源优化算法

3. 干扰控制模块

- 干扰阈值设置

- 波束间距检测
- 频率冲突分析

4. 可视化模块

- 波束分布图绘制
- 实时状态监控
- 交互式控制界面

1.4.3 关键技术实现

波束位置计算 采用极坐标分布算法：

$$\begin{aligned} x_i &= r \cos\left(\frac{2\pi i}{N}\right) \\ y_i &= r \sin\left(\frac{2\pi i}{N}\right) \end{aligned} \quad (5)$$

其中：

- N ：波束总数
- i ：波束编号
- r ：归一化半径（0.7）

干扰检测算法 基于欧氏距离的干扰判定：

$$D_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} < \frac{T}{50} \quad (6)$$

其中：

- D_{ij} ：波束 i 和 j 之间的距离
- T ：干扰阈值（0-100）

1.4.4 性能指标

系统主要性能指标如下：

- 波束配置：4/7/12波束可选
- 频率资源：4个频段
- 功率调节范围：0-100%
- 干扰阈值范围：0-100%
- 可视化更新率：实时

1.4.5 用户界面设计

系统提供图形化界面，包含：

- 波束分布可视化画布
- 功率控制滑动条
- 干扰阈值调节器
- 频率资源指示器
- 状态监控文本框

1.4.6 系统特色

1. 直观的波束分布可视化
2. 灵活的波束配置选项
3. 智能的频率资源优化
4. 实时的干扰监测预警
5. 完整的状态监控记录

2 整体总结

2.1 系统架构

2.1.1 整体架构

系统采用模块化设计，四个子系统相互独立但功能互补：

1. 信号处理层

- AD/DA转换模块：实现模拟/数字信号转换
- 信号调制解调功能

2. 编码传输层

- Turbo编码模块：实现信道编码
- 通信协议模块：实现数据帧处理

3. 资源控制层

- 波束管理模块：实现空间复用
- 频率分配模块：实现频率复用

2.2 系统特色

1. 全面的功能覆盖

- 从信号处理到资源管理的完整链路
- 支持多种通信场景仿真

2. 灵活的配置选项

- 参数实时调节
- 多种工作模式切换

3. 友好的用户界面

- 图形化操作界面
- 实时数据可视化

4. 完善的监控机制

- 全程运行状态监测
- 异常情况实时预警

3 实验结果

3.1 模块一结果

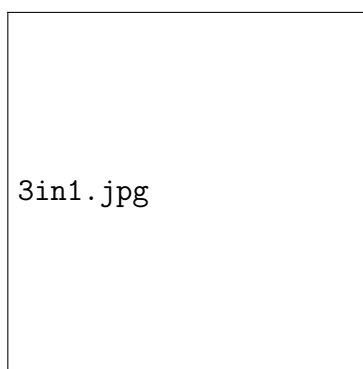


图 1: 三种模式（方波正弦波三角波）对应界面，左侧为原始信号，中间为ADC生成的数字信号，右侧为通过滤波器转换回的模拟信号

3.1.1 三角波形（Triangle Wave）

- 原始信号（左侧）：图示为频率 10 kHz，幅度 5.00 V 的理想三角波。

- **ADC转换的数字信号（中间）：**ADC采样率为 100 kHz，远高于原始信号频率 10 kHz。根据奈奎斯特采样定理，采样率 (100 kHz) 远大于信号最高频率 (10 kHz) 的两倍，因此ADC能够充分捕捉原始三角波的细节。8 bit 的量化精度提供了 $2^8 = 256$ 个量化级别，使得PWM数字信号能够较好地反映原始信号的瞬时电压。从图中可以看出，PWM的占空比随着原始三角波的电压变化而平滑地调整，体现了较好的采样和量化效果。
- **重建波形（右侧）：**重建的模拟信号与原始三角波非常接近，表明ADC/DAC过程效果优秀。显示误差仅为 0.078 V，信噪比（SNR）高达 31.35 dB。这得益于高采样率有效地避免了混叠。DAC后的低通滤波器（LPF）截止频率为 0.58 kHz，这对于平滑PWM信号（PWM频率 0.8 kHz）非常有效。尽管输入信号频率为 10 kHz，但重建后的波形保留了原始三角波的频率和基本形状，证明了在采样率足够高的情况下，即使LPF截止频率低于信号基频，也能在某种程度上重建出可辨识的波形，但这主要取决于PWM的载波频率与信号频率的关系以及LPF对这些频率成分的滤除效果。在这种高采样率下，PWM波形的平均值已经很好地反映了原始模拟信号。

3.1.2 正弦波形（Sine Wave）

- **原始信号（左侧）：**图示为频率 10 kHz，幅度 5.00 V 的理想正弦波。
- **ADC转换的数字信号（中间）：**100 kHz 的高采样率使得ADC能够精确地捕捉 10 kHz 正弦波的每一个周期内的多个采样点，从而在数字域精确地表示了原始信号。PWM信号的占空比随输入正弦波的瞬时电压变化而连续调整，波形脉冲密度也随之变化，呈现出明显的正弦波形状。
- **重建波形（右侧）：**重建的正弦波形与原始信号高度一致，其质量相当优秀。误差为 0.267 V，SNR为 22.43 dB。虽然误差略高于三角波，但从视觉上看，重建波形清晰地再现了 10 kHz 正弦波的频率和形状。LPF（0.58 kHz 截止频率）在去除PWM载波频率的同时，也保留了足够的信息来重建正弦波，这表明 100 kHz 的采样率产生了足够多的信息量，即使经过较低截止频率的LPF，也能通过平均值特性大致还原波形，尤其是在PWM频率 (0.8 kHz) 高于LPF截止频率的情况下，LPF能有效滤除高频载波。

3.1.3 方波形（Square Wave）

- **原始信号（左侧）：**图示为频率 10 kHz，幅度 5.00 V，占空比 50% 的理想方波。方波包含丰富的奇次谐波成分，其频谱带宽是无限的。
- **ADC转换的数字信号（中间）：**由于 100 kHz 的高采样率，ADC能够捕捉到方波快速上升沿和下降沿附近的大量采样点。因此，PWM信号能够准确地在数字域

表示方波的两个稳定电平以及转换过程，尽管由于量化效应和PWM的离散性，边沿看起来仍有阶梯状。

- **重建波形（右侧）：**重建的方波形误差为 0.288 V，SNR为 24.79 dB。重建信号与原始方波的形状非常接近，表现出良好的方波特征。尽管方波的边沿（上升沿和下降沿）由于LPF（0.58 kHz 截止频率）的低通特性而被平滑，无法达到理想方波的垂直跳变，但其高电平、低电平以及周期都得到了准确的重建。这再次证明了 100 kHz 的高采样率有效避免了混叠，并且DAC后的LPF成功地滤除了PWM载波成分，同时尽可能地保留了方波的主要特征。边沿的平滑是LPF的固有特性，因为方波的高频谐波被LPF滤除了，这在实际系统中是可接受的，并且通常可以通过更高截止频率的LPF或更高阶的DAC实现更陡峭的边沿。

4 实验过程中遇到的问题及其解决办法

4.1 图形界面刷新效率低下

问题：频繁重绘导致卡顿； **原因：**未复用绘图对象； **解决：**采用clear方法与局部刷新策略，避免阻塞。

4.2 干扰检测灵敏度调整难题

问题：误报或漏报多； **原因：**归一化距离不准确； **解决：**实测距离校准标准，调整阈值参数。

4.3 滑块事件响应过于频繁

问题：界面卡顿严重； **原因：**使用`onMotion`事件频率过高； **解决：**改为`onButtonRelease`事件，或结合节流机制优化响应。

4.4 中文字体显示乱码问题

问题：图形中中文乱码； **原因：**默认字体不支持中文； **解决：**加载支持中文字体如SimHei，配置默认字体。

4.5 多模块间数据同步问题

问题：数据状态不同步； **原因：**缺少集中管理； **解决：**统一数据源，使用事件驱动同步机制，必要时引入锁机制。