

Wi-Fi 协议栈的信号处理与 SDR 实现

王思成

Oct 13, 2026

在无线通信研究领域，软件无线电平台为理解 Wi-Fi 协议栈的物理层和 MAC 层提供了独特视角。通过 SDR 设备，我们可以直接观测并操控从比特流到射频波形的完整链路。相比商用 Wi-Fi 芯片的黑箱实现，SDR 方案虽然牺牲部分性能，却换来了可观测、可修改的开放环境。本文聚焦于 802.11a/g/n/ac/ax 系列标准中的 OFDM 物理层，结合 GNU Radio 与 Python 工具链，系统梳理信号处理流程与工程实现要点。

1 协议栈映射与帧结构

IEEE 802.11 协议栈直接对应 OSI 模型的物理层与数据链路层。其中物理层包含 PLCP 子层和 PMD 子层，负责将 MAC 协议数据单元转换为射频波形；MAC 层则实现分布式协调功能 DCF，采用 CSMA/CA 机制管理信道竞争。帧结构上，MAC 协议数据单元 MPDU 封装在 PLCP 协议数据单元 PPDU 中，后者由前导、信号字段和数据字段组成。带宽选项涵盖 20 MHz、40 MHz、80 MHz 及 160 MHz，通过信道绑定技术提升吞吐量。

2 物理层信号处理流程

2.1 扰码与前向纠错

物理层首先对 MAC 层交付的比特流进行加扰，消除连续零或连续一带来的直流分量。随后进入前向纠错阶段，802.11n/ac/ax 支持二进制卷积码 BCC 和低密度奇偶校验码 LDPC。BCC 码率包括 1/2、2/3、3/4、5/6，卷积约束长度通常为 7，生成多项式分别为 171 和 133（八进制）。LDPC 则提供更强的纠错能力，尤其在高阶调制下表现突出。

2.2 交织与星座映射

交织器将纠错编码后的比特序列重排，分散突发错误对单个 OFDM 符号的影响。交织规则遵循标准规定的比特级置换公式，确保相邻比特映射到不同子载波和不同星座位。星座映射支持 BPSK、QPSK、16-QAM、64-QAM、256-QAM 乃至 1024-QAM，每种调制方式对应特定星座点坐标。以 QPSK 为例，两个比特映射到复平面四个象限，符号能量归一化后平均功率为 1。

2.3 OFDM 调制

OFDM 调制是 802.11a/g/n/ac/ax 物理层核心。子载波间隔固定为 312.5 kHz（HE 模式下缩减至 78.125 kHz），FFT 点数随带宽变化：20 MHz 对应 64 点，40 MHz 对应 128 点，依次类推。IFFT 完成后插入循环前

缀 CP，长度为符号周期的 1/4、1/8 或 1/16，以对抗多径时延扩展。窗函数通常采用升余弦窗，降低带外辐射。

2.4 训练序列与信道估计

前导序列包含遗留短训练字段 L-STF、长训练字段 L-LTF 以及信号字段 L-SIG。在 802.11ac/ax 中还增加 VHT-SIG、HE-SIG 等专用字段。L-STF 用于自动增益控制和粗同步，L-LTF 提供信道估计样本。接收端利用 L-LTF 频域子载波的已知符号执行最小二乘信道估计，进而计算均衡系数。

3 SDR 发射链路实现

3.1 工具链与流程

GNU Radio 配合 gr-80211 模块可完成端到端 Wi-Fi 信号生成。Python 环境下，scapy 负责构造 MAC 帧，numpy 实现基带 DSP，SoapySDR 抽象层驱动 USRP 或 LimeSDR。发射流程依次为：比特流 → 卷积编码 → 交织 → QAM 映射 → IFFT → 加 CP → 加窗 → 复基带 I/Q → 射频上变频。

3.2 关键代码解析

以下 Python 片段展示 L-STF 生成过程。代码首先生成 16 点短训练序列，再在频域零填充至 64 点，最后通过 IFFT 得到时域波形。

```
1 import numpy as np
3 # 短训练序列频域符号（仅非零子载波）
   sts_freq = np.array([0,0,0,0,0,0,0,0,
5                       1+1j,0,0,0,-1-1j,0,0,0,
                       1+1j,0,0,0,-1-1j,0,0,0,
7                       0,0,0,0,0,0,0,0])
   # IFFT 并归一化
9 sts_time = np.fft.ifft(sts_freq) * np.sqrt(64/52)
   # 复制两遍形成 160 点 L-STF
11 l_stf = np.tile(sts_time, 10)
```

该实现中，频域符号仅在 ± 4 、 ± 8 子载波位置非零，符合 802.11 标准定义的 L-STF 频谱特性。IFFT 后乘以归一化因子，确保时域平均功率与后续数据符号一致。

循环前缀插入代码如下：

```
1 def add_cp(symbol, cp_len):
   """在 OFDM 符号前插入循环前缀"""
3   return np.concatenate((symbol[-cp_len:], symbol))
```

窗函数通常采用升余弦窗，公式为

$$w(n) = 0.5 \left(1 - \cos \left(\frac{2\pi n}{N_{win}} \right) \right)$$

其中 $[N_{win}]$ 为窗长度，作用于 CP 的前半部分与符号尾部，平滑符号边界。

3.3 发射参数配置

USRP 的采样率需与信道带宽匹配。20 MHz 模式下基带采样率设为 20 MS/s，数字上变频至 2.412 GHz（信道 1）。增益设置需兼顾线性范围与动态范围，通常先进行开环功率校准，再根据 EVM 指标微调。

4 SDR 接收链路实现

4.1 射频前端处理

接收端首先执行自动增益控制，将输入信号幅度稳定在 ADC 满量程的 50% 左右。直流偏移消除通过时域平均或高通滤波实现，IQ 不平衡校正则采用基于导频的 Gram-Schmidt 正交化方法。

4.2 同步与检测

L-STF 自相关检测利用其周期重复特性，相关峰阈值设为噪声方差的 8 - 10 倍。检测到峰值后进入细同步阶段，利用 L-LTF 与本地副本进行互相关，确定符号起始位置。互相关峰值位置精度可达采样级，为后续信道估计提供可靠起点。

4.3 信道估计与均衡

信道估计采用最小二乘 LS 算法：

$$\hat{H}_k = \frac{Y_k}{X_k}$$

其中 $[Y_k]$ 为接收频域符号， $[X_k]$ 为已知导频符号。均衡器直接使用 $\{\hat{H}_k\}$ 的共轭倒数进行迫零均衡。相位跟踪则利用数据符号中的导频子载波，估计并补偿残余频偏与相位噪声。

4.4 解调与译码

均衡后符号进行星座解映射，输出对数似然比 LLR 送入 Viterbi 译码器或 LDPC 译码器。Viterbi 算法采用软判决分支度量，状态数量为 $[2^{K-1}]$ （K 为约束长度）。LDPC 译码常用最小和算法，迭代次数通常设为 10 - 20 次。译码完成后执行 CRC32 校验，正确帧上送 MAC 层。

5 MAC 层在 SDR 上的实现

5.1 CSMA/CA 定时器

MAC 层核心是载波侦听多路访问/冲突避免机制。SDR 平台通过能量检测或前导检测实现物理载波侦听，阈值一般设为 -82 dBm。DCF 帧间间隔包括 SIFS（16 μ s）、PIFS（25 μ s）、DIFS（34 μ s），由高精度定时器维

护。退避计数器在每次 DIFS 空闲后启动，退避槽时长 $9\ \mu\text{s}$ 。

5.2 ACK 与 Block-ACK

接收正确数据帧后需在 SIFS 时间内发送 ACK。SDR 方案可将 ACK 生成逻辑置于 FPGA，缩短处理延迟；或采用 CPU 中断响应，适用于低速率场景。Block-ACK 机制允许一次确认多个 MPDU，bitmap 字段指示接收状态，减少 ACK 帧开销。

5.3 速率自适应

Minstrel-HT 算法周期性探测各 MCS 等级的吞吐量，维护成功率统计表。实时决策时选择吞吐量最高的 MCS，并根据信道质量动态降级或升级。机器学习方法则利用信道状态信息 CSI 作为特征，直接预测最优 MCS。

6 案例演示

6.1 真实流量抓取与回放

gr-80211 模块可配置为监听模式，捕获 2.4 GHz 信道上的 beacon 帧。捕获数据以 pcap 格式保存，Wireshark 可解析 MAC 头与信息元素。回放时将 pcap 文件重新映射为 I/Q 样本，通过 USRP 发射，验证接收端解析正确性。

6.2 自定义 Data 帧注入

利用 scapy 构造 QoS Data 帧，设置 Address1 为目标 STA MAC，Address3 为 BSSID。帧载荷包含 UDP 报文，序列号自增。注入后观察 Block-ACK bitmap 变化，可验证聚合与重传机制。

6.3 指标测量

EVM 测试需在接收端关闭均衡器，统计星座点到理想位置的均方根误差。20 MHz 模式下典型 EVM 要求为 -25 dB (16-QAM)。吞吐量测试采用 iperf3，实测 80 MHz 模式下单流 TCP 吞吐量可达 300 Mbps 以上。

6.4 安全实验

Karma 攻击通过伪造多个 SSID 的 beacon，诱导客户端主动关联。SDR 平台可实时修改 beacon 中的 SSID 字段与 RSN IE，测试 802.11w 管理帧保护 MFP 的有效性。实验中开启 MFP 后，伪造 deauth 帧将被 STA 丢弃。

7 性能优化与调试

7.1 计算瓶颈分析

FFT 是基带处理最耗时环节。Intel AVX2 可将 64 点 FFT 加速至单核 $1\ \mu\text{s}$ 以内；NVIDIA CUDA 实现可进一步降低至 200 ns。FPGA 方案如 OpenWiFi 将整个 PHY 卸载至 Zynq SoC，MAC 层仍运行于 ARM 核，端到端

延迟小于 100 μ s。

7.2 实时性保障

Linux 系统需启用 PREEMPT_RT 补丁，将中断处理移至实时线程。用户态 I/O 采用 PF_PACKET 套接字或 DPDK 框架，减少内核协议栈开销。Docker 环境需映射 /dev/bus/usb 并启用 -privileged 模式，保证 SDR 设备访问权限。

7.3 开源社区资源

gr-ieee802-11 提供完整 GNU Radio flowgraph，支持 802.11a/g；OpenWiFi 基于 FPGA+ARM 架构，已支持 802.11n 全部特性；srsRAN 则侧重蜂窝网络，但其 PHY 框架可复用于 Wi-Fi 研究。

8 未来方向

8.1 Wi-Fi 7 演进

802.11be 引入 320 MHz 带宽、4096-QAM、多链路操作 MLO。多链路设备可在 2.4 GHz、5 GHz、6 GHz 同时传输，聚合吞吐量理论峰值超过 30 Gbps。SDR 平台需扩展至 320 MHz 采样率，并实现多射频前端同步。

8.2 AI 原生物理层

深度学习信道估计网络以时域样本为输入，直接输出频域信道响应，相比 LS 算法在低 SNR 下可获得 2 – 3 dB 增益。自动调制识别模型利用 CNN 或 Transformer 架构，识别精度在 SNR > 10 dB 时超过 99%。

8.3 O-RAN 与共享频谱

6 GHz 频段引入自动频率协调 AFC 系统，Wi-Fi 与 NR-U 需共存。O-RAN 架构将 PHY-RF 接口标准化，允许第三方 SDR 设备接入同一云化基带池，实现异构网络资源调度。

9 结论

SDR 技术将原本封闭的 Wi-Fi 协议栈转化为可观测、可重构的研究平台。从比特级扰码到射频波形成形，每一个环节均可通过代码或 FPGA 逻辑精确控制。开放生态的持续演进，正逐步缩小与商用芯片的性能差距。建议读者从最简单的 BPSK beacon 发射开始，逐步扩展至完整 802.11ax 收发链路，在实践中深化对无线通信原理的理解。