QAM Transmitter with RRC Pulse Shaping Using Verilog

Bhavesh Mali

2025

Abstract

This project implements a digital communication subsystem in Verilog that performs QAM symbol generation, pulse shaping using a Root Raised Cosine (RRC) filter, and polyphase interpolation. It supports adaptive symbol timing estimation and outputs complex baseband signals. Verification is done using Python convolution.

1 Introduction

Digital modulation schemes such as QAM require symbol-level filtering to minimize inter-symbol interference (ISI). This design models a baseband transmitter pipeline featuring:

- Serial-to-parallel bit packing
- QAM symbol mapping via LUT
- Dual-path (I/Q) root raised cosine FIR filtering
- Symbol edge detection for timing recovery
- Polyphase filter access using interpolation index

2 Architecture

The system uses a bit accumulator to construct a QAM symbol which is mapped via cosine/sine ROMs. The mapped values are shifted into symbol buffers. FIR convolution is applied per phase using pre-loaded RRC taps.

3 Verilog Modules

main.v

Implements the full baseband pipeline:

- Symbol timing estimation
- QAM LUT interface
- Polyphase FIR filtering

sine_cos_rom.v

A simple ROM for cosine/sine values used in symbol mapping. Uses 8-bit outputs.

tb.v

Testbench file used to give testcase to main.v file

4 Python Verification

The ‘rrc.ipynb‘ notebook:

- Performs convolution with same RRC coefficients
- Plots waveform of real/imag outputs
- Confirms correctness of the Verilog simulation

5 Conclusion

This design demonstrates how a real-world QAM transmitter with adaptive filtering and verification can be implemented at RTL level. The use of polyphase FIR and Python-based functional verification reflects practices used in industry.