

# Research on $\pi/4$ QPSK Modulation Communication Transmission System and FPGA Implementation

Yuhui Chu, Yunjie Yuan\*, Xiaolei Cai, Yanyan Wang, Peijie Yin

Xi'an Electronic Engineering Research Institute, Xi'an 710100, Shaanxi, China

\*Corresponding author: Yunjie Yuan, yinpeijie789@hotmail.com

**Copyright:** © 2025 Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY 4.0), permitting distribution and reproduction in any medium, provided the original work is cited.

Abstract: This paper examines the  $\pi/4$ QPSK modulation communication transmission system, analyzing the performance advantages and disadvantages of  $\pi/4$ QPSK in comparison to QPSK. It also presents a comprehensive FPGA implementation scheme for a modulation communication transmission system, integrating RS channel coding, framing, frequency conversion, and other modules. This design is based on practical research and development requirements. The Xilinx Spartan6 chip board was used for board-level verification. The  $\pi/4$ QPSK modulated signal was transmitted via D/A conversion and radio frequency, with the transmitted waveform was looped back for reception. After A/D processing, the correctness of the designed modulation transmission scheme was verified.

**Keywords:** π/4QPSK modulation; FPGA

**Online publication:** February 24, 2025

### 1. Introduction

In communication systems, the transmitter primarily consists of modules such as modulation, encoding, and framing. The design of the transmitter's signal directly affects the performance of the entire communication system. Digital modulation primarily involves mapping signals to constellation points, with different types of constellation diagrams directly affecting the system's noise resistance and the bandwidth resources it consumes.  $\pi/4$  Quadrature Phase Shift Keying ( $\pi/4$ QPSK) combines the advantages of Quadrature Phase Shift Keying (QPSK) and Offset Quadrature Phase Shift Keying (OQPSK), making it a form of quadrature phase shift keying modulation. The constellation diagram of QPSK modulation can experience phase jumps of up to  $\pi$ , which can lead to spectral expansion and larger side lobes, as well as being affected by the nonlinearity of power devices. In contrast, OQPSK limits phase jumps to  $\pi/2$ , resulting in less bandwidth usage and better anti-interference capabilities, but with higher implementation complexity.  $\pi/4$ QPSK limits phase jumps to  $\pi/4$  and  $3\pi/4$ , with minimal phase changes, making the spectrum more compact after filtering and limiting compared to the previous two. This results in better bandwidth utilization under small-scale fading conditions. Single-carrier systems have a

lower peak-to-average power ratio compared to multi-carrier systems, reducing the requirements for RF devices. Therefore, in communication systems with low throughput, such as messaging systems, or in systems with high real-time requirements, such as voice systems, single-carrier systems provide significant cost and overhead advantages. Thus, research on  $\pi/4$ QPSK communication systems is necessary.

 $\pi/4$ OPSK employs differential encoding, allowing for both coherent and non-coherent demodulation (differential demodulation). Non-coherent demodulation does not require the recovery of a local carrier, significantly reducing the impact of carrier synchronization inaccuracies on the system <sup>[1,2]</sup>. Additionally, noncoherent demodulation enables rapid synchronization, reducing the need for synchronization codes in the frame structure compared to coherent demodulation, thereby increasing the proportion of effective data and improving spectral efficiency. In fast-fading channels, differential demodulation can achieve lower bit error rates than coherent demodulation. However,  $\pi/4$ QPSK also has drawbacks, such as the need for linear power amplifiers and the potential loss of paired symbols in the event of reception errors<sup>[2]</sup>. Considering the trade-offs between the advantages and disadvantages of  $\pi/4$ QPSK modulation, it is suitable for burst wireless communication systems. In a study conducted by Zhou,  $\pi/4$ QPSK modulation was applied to ship communication systems to improve bandwidth efficiency and system flexibility <sup>[3]</sup>. As for Liu et al., a lookup table method was used to synthesize  $\pi/4$ QPSK waveforms through multiple square-root raised cosine filters, achieving simpler implementation with minimal performance loss<sup>[4]</sup>. Zhou *et al.* proposed a cross-product frequency discriminator demodulation method to address frequency offset issues in burst transmissions of  $\pi/40PSK$ <sup>[5]</sup>. The literature mentioned above primarily analyzes  $\pi/4$ QPSK from an algorithmic perspective, whereas this paper validates the system's transmission scheme through the FPGA implementation of the  $\pi/4$ QPSK transmitter.

# 2. $\pi/4$ QPSK modulation transmitter implementation scheme

As shown in **Figure 1**, the transmitter of the  $\pi/4$ QPSK modulation system mainly consists of RS encoding, framing, serial-to-parallel conversion, differential encoding, and up-conversion modules. The transmitted signal is finally output through a DAC, resulting in the RF output of the  $\pi/4$ QPSK modulated waveform.



Reed-Solomon (RS) coding is a powerful forward error correction code suitable for correcting burst and random errors. It treats the original data symbols as polynomial coefficients and attaches a generator polynomial to the original data to form the encoded codeword. Therefore, RS coding has low overhead and implementation complexity, making it suitable for  $\pi/4$ QPSK systems. As shown in **Figure 2**, the RS encoder is implemented using a linear feedback shift register structure, with a division-based structure. The input consists of *m* original data symbols representing polynomial coefficients, and the generator polynomial is defined as  $g(x) = (x-\alpha)(x-\alpha^2)...(x-\alpha^{2t})$ , where  $\alpha$  is the primitive element of the finite field  $GF(2^m)$ . After passing through this linear feedback shift register system, the output is the codeword.



Figure 2. RS encoder structure

Since synchronization and other processing are required at the receiver, preamble symbols such as frame headers and synchronization frames (used by the receiver to detect the signal) are added to the output of the RS encoder. The final framed output remains a serial bitstream.

The serial-to-parallel conversion module maps the bitstream into I and Q signals, which are then fed into the  $\pi/4$ QPSK module.  $\pi/4$ QPSK modulation uses eight different carrier phases to represent digital information, as shown in the signal space diagram in **Figure 3**. The eight phase states are  $\{0, \pi/4, \pi/2, 3\pi/4, \pi, -3\pi/4, -\pi/2, -\pi/4\}$ . These phase states can be viewed as two sets of QPSK signal phases, with the black solid points forming one set  $\{0, \pi/2, \pi, -\pi/2\}$  and the white hollow points forming the other set  $\{\pi/4, 3\pi/4, -3\pi/4, -\pi/4\}$ . The phase state transitions in  $\pi/4$ QPSK can only occur between these two QPSK phase groups, meaning the maximum phase jump in  $\pi/4$ QPSK is  $3\pi/4$ . The differentially encoded data is then up-converted and mixed before being sent to the DAC, resulting in the final transmitted  $\pi/4$ QPSK modulated signal.



Figure 3.  $\pi/4$ QPSK constellation diagram

# 3. FPGA implementation of the transmitter

This paper uses the Xilinx Spartan6 FPGA hardware platform to implement the  $\pi/4QPSK$  modulation transmitter. The FPGA platform, equipped with hardware description languages, implements signal processing algorithms by building digital circuits, offering real-time processing, parallelism, and flexibility, making it suitable for communication systems. The FPGA platform can support multiple modulation schemes, data rates, and flexible waveform configurations. The  $\pi/4QPSK$  modulation transmitter scheme designed in this paper was verified through simulation on the Vivado platform and waveform debugging using Chipscope. The simulation and measured waveform analyses are as follows:

As shown in Figure 4, the simulation results of the RS encoder's input and output are presented. The input

data (datain) and output data (dataout) are shown, with the encoded results sequentially appended to the input data. The encoded output is available in the next clock cycle, verifying the low encoding delay. The input data length is 16 bytes, and the encoded output length is 32 bytes, achieving RS(32,16) encoding.

As shown in **Figure 5**, the simulation results of the framing module are presented. The output (data\_bit) is the framed bitstream, combining the synchronization frame, frame header, and encoded data into a serial bitstream.

As shown in **Figure 6**, the simulation results of the modulation module are presented. The I and Q signals are the differentially encoded outputs, and the I and Q signals after shaping filtering and up-conversion are shown. The final synthesized I and Q signals are output as the modulated signal. The bitstream enters the modulation module, is converted into I and Q signals through serial-to-parallel conversion, and then differentially encoded. The differentially encoded I and Q signals undergo shaping filtering. A DDS generates a 25 MHz carrier, which is multiplied with the shaped I and Q signals. The resulting I and Q signals are summed to produce the final modulated output signal.

The final modulated  $\pi/4$ QPSK signal is output through a DAC and looped back into the hardware board. The waveform after A/D processing is shown in **Figure 7**, which represents the observable measured data waveform. The waveform is clear with minimal glitches, demonstrating that the designed transmitter hardware implementation meets practical application requirements.

																		14.845	000 1	us													
Name	Value	2 us		4 u	5	1	6 us		8	as		10 us		12	us		14 us	1	16	us	1	l8 us		20 us		22 us		24	us	1	26 us		28
🕌 clk	1																				1 I												ſ
🖁 rst	0																																
> 🔮 datain[7:0]	00	11	22	33	44	55	66	77	88	99	aa	bb	cc	dd	ee	55								0	0								
🕌 en	1																																
> 🗑 dataout[7:0]	e3	00 X	11 🗶	22	33	44	55	66	77	88	99	aa	ЬЬ	cc	X dd	ee	55	e3 🗸	f4	fb	da 🕺	54 f	5 X 0	9 <b>2</b> c	91	32	42	48	81	c7	22	3f X	0
H code_end	0																																

Figure 4.	RS	encoding	simu	lation	results
Figure 4.	цо	encouning	Sinna.	auon	resuits

successive and the second second																																		- 0	
Layout Help	Layout Help																																		
CH (M) (M) (M) (M) (M)	🔥 😹 ↓ ↑ 😳 😤 🗄 🗇 😳 🎤 😥 🥕 🔗 🖉 🖉 😫 🛬 🖆 👔 🏫 🍾 🖬 🐷 💌 🛊 🚱 🚱 💆																																		
						18, 86	4. 9000	100 us																											^
Name	Malua	1	110 000		110 0			110.00			110.00			119.00	· ···-		119.05			119 100			10 150		1.	0 200		14	2 250			9 200		110	250
18 clk	value	-	10,000	<u> </u>	10,0	10 us		10, 50			10, 5,						10.00			19, 100			19, 190					ta da							
16 rst	0	20000000000000		*****			SALES SALES		191111111111111	******	reserves		******	******		STRATES STRATE		nen en								******			101101010	906999999999					
counter2[9:0]	1	34	35 X	36 X 31	7 0	1	2	(3 X	4 X	5 )	6 Х	7 )	8)	• X 1	10 / 1	1 / 1	2 X 13	(14)	15	X 16	(17)	18	(19)	20 🗙	21	22	23 )	24 🔀 :	25 X 2	16 / 2	7 X 2	28 X	29 🗙 30	) X 31	X 32
counter1[9:0]	0	32	33 🔨	34 33	3 X 📃	0	$ \longrightarrow $		2 )(	3 🔨	4 X	5 (	6 )		8 (	9 🔨 1	• X 11	12	X 13	X 14	15	16	17	18 🗙	19	20	21 (	22 :	23 🗙 2	4 2	5 X 2	26	27 28	3 29	30
1🐻 ena	1																																		
en_RS	1		L																																
· 🐻 start	0																	_	_												_			_	
dataout[7:0]	00	<u>X 06</u>	K		00			(II X	<u>11 X</u>	22 X	83 X	44 X	55 X @	6 X 7	77 <u>X</u> 8	18 X 9	9 <u>X aa</u>	Хрр	X cc	X dd	X ee X	55	<u>47</u>	57 X	75	20 X	2f X 1	89 X ·		12 X 3	7 X 0	<u>6 X</u>	38 X 89	X 70	X 82
ug data_bit	0							JUUU									uпп		Л			υų									υΨ				
n																																			

Figure 5. Framing module output simulation results

🌔 project_1 - [D:/phy_code/signal_process/project	t_1/project_1.xpr] - Vivado	p 2018.3	App. surf.)	- 6 X
<u>File Edit Flow Tools Reports Wind</u>	dow La <u>v</u> out <u>V</u> iew	Run Help Q- Quic	K Access	Synthesis Complete 🗸
🖷 🔸 🔶 🗉 🐘 🗙 🕨 🛤	Φ Σ % Ø 1	8 I I 🕨 🛌	10 ms v 🖭    C	Default Layout 🗸 🗸
Flow Navigator 😤 ≑ ? 🔤 SIMUL	ATION - Post-Synthesis Sir	mulation - Functional - sim_	1 - tb_tx	? ×
✓ PROJECT MANAGER	Untitled 1* × top TX.v	x pi4qpsk mod.v	diff encode.v x	2 5 15
Settings	0 8 0 0 5	a latint lating		0
Add Sources		3   3   13   14   E	12.010100 us	
Language Templates	Name	Value	15 us 110 us 115 us 120 us 125 us 120 us 125 us 140 us 145 us	150 us 155 us
👎 IP Catalog			7 1846 86 46 7 6 7 6 7 6 7 7 7 7 7 7 7 7 7 7 7 7	AN A THE A MAN
97	> 😻 Xk[7:0]	90	N INAVAAA ILAA MAARKA AA KU KUMARAMAA AA AA MAARAA INAA INAWAAA AA INII AAA W	WWA A A M
V IP INTEGRATOR			Contracts for a state of the traction with the with the with the second state of the s	1' WW V''
Create Block Design			A tax and tA — I. t. A. a a. A than a said that a shift diffet. A An a ta aBA a dia an Ama	LAA 🗤 🗛 L
Open Block Design	> 😻 YK[7:0]	-90	EMAR AIR MA — HEAR AA AMAAAAA AMAAAAH EMMEETI WAXAAH KA AMAAN 🗎	A VEI AVAVAVEI VA
Generate Block Design			Taki Mil Ad 2000 A A Adda a a Alada a da Anna a da a addit ta A An Add	V Y Y Y Y Y Y
✓ SIMULATION			A A MARINA A MARINA A AAMATATATATA ATA JUMITA A LA ATA	han an a l
Run Simulation	w m_axis_datal[25:10	-570	, A JARAMANA IN KANARAR ATA JUU UNBARKAMUS AAANA JIKA TAMAAA JIKA KAMAA JIKA KA	АММА И.
			AA MAAAA AAAAAM AAAAAA AAAAAAAAAAAAAAA	HTTTT WWW
Y RTL ANALYSIS				
> Open Elaborated Design		10022	NT CARRY WE COLL OF CORRELATION AND CARAGE WALL AND COMPARENT OF A A MAIN AND A COMPANY AND A COMPANY AND A COMP	***\1 AA AA /
Y SYNTHESIS	/ •• III_axis_dad(25:10)	-16555	UTEMIZTUATETTI COOL MINISTAT MINISTATICA IN MALIA	
Run Synthesis				
Open Synthesized Design			der der der der eine sin der eine der eine eine der der der der der der der der der de	al all all a Milliolik
Constraints Wizard	> N x_band[23:0]	-90896		NINGNONUMBER
Edit Timing Constraints			הערבים את היה בארך בית את המנה ברבי בית את האת את את ברבי היות את ביו ביות את את את	
🕷 Set Up Debug			hain dah sam saan da muululu dan a amdah sa ara dan dah sa ara sa birdililin da dasa ada a daha du ahu an sia	hidds of the deads of the second
Timing Summary	> 😻 y_band[23:0]	2128080	ይገኙ ለቁም ቁም እ <sub>ን</sub> ም ግብ እንዲሆኑ እና እንዲሆኑ እስለ እንዲሆኑ እስለ እና ለመሆኑ እንደ መስከት እንደ እንዲሆኑ እስለ እና መሆኑ እንደ እና መ	
Report Clock Networks			n dan set da fi da he da	NUM SAME DE SUR SUR SU
Report Clock Interaction			wateriel it klud skille headended et schemmunger esterer om det skille bekalte tet i her sekaretes bekaretes be	and Recorded and
Report Methodology	> 👹 pi4qpsk_out[23:0]	-2218976		
Report DRC				
Report Utilization				~
Seport Power		< >>		>
V 1	ru console   Messages	LUG		Sim Time: 73810100 ps

Figure 6. Modulation output simulation results



Figure 7. Measured waveform after D/A output and A/D input

# 4. Conclusion

This paper primarily introduces the simulation and FPGA implementation of the  $\pi/4$ QPSK modulation system, specifically including the implementation of RS encoding, framing, serial-to-parallel conversion, differential encoding, and up-conversion. The specific implementation methods for each module are discussed and the proposed methods are realized on the FPGA platform. By utilizing debugging software to observe the output signals and the outputs of each module, satisfactory results have been achieved.

# **Disclosure statement**

The authors declare no conflict of interest.

# Author contributions

Conceptualization: Yunjie Yuan Investigation: Yuhui Chu, Xiaolei Cai Formal analysis: Yanyan Wang, Peijie Yin Writing: Yuhui Chu

# References

- [1] Yuan Y, 2012, Software Radio-Based  $\pi$ /4-DQPSK Modulation and Demodulation System, thesis, Huaqiao University.
- [2] Jiang N, 2005, Research on Digital π/4-DQPSK Modulation and Demodulation and FPGA Implementation, thesis, University of Electronic Science and Technology of China.
- [3] Zhou G, 2019, Design of NC-OFDM System Based on π/4QPSK for Ship Communication Systems. Electronic Measurement Technology, 42(23): 128–131.
- [4] Liu Z, Sun Y, Liu Y, et al., 2016, Design and Implementation of  $\pi/4$ QPSK Based on Multi-Channel Synthesis. Journal of China Academy of Electronics and Information Technology, 11(06): 642–648.
- [5] Zhou F, Zheng L, Tao W, 2017, Digital Demodulation of π/4-QPSK Signals Based on Cross-Product Frequency Discriminator. Telemetry and Remote Control, 38(02): 49–55.

#### Publisher's note

Bio-Byword Scientific Publishing remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.