



OFDM Transmitter and Receiver with Software Defined Radios

Berker Acir
berker.acir@metu.edu.tr

Wireless Systems, Networks and Cybersecurity Laboratory
Department of Computer Engineering
Middle East Technical University
Ankara Turkey

July 8, 2021

Outline of the Presentation

- 1 Introduction
- 2 Implementation
- 3 Test and Results
- 4 Discussion
- 5 Future Work
- 6 Conclusion

Agenda

- 1 Introduction
- 2 Implementation
- 3 Test and Results
- 4 Discussion
- 5 Future Work
- 6 Conclusion

GNU Radio

GNU Radio

Open-source software development framework that provides signal processing functions for implementing software-defined radios. This framework also offers a graph design approach, namely GNU Radio Companion (GRC).

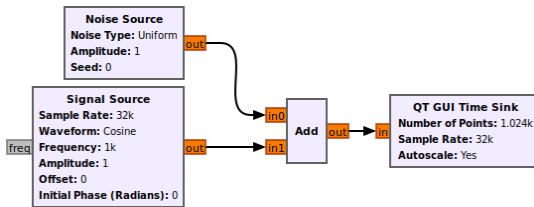


Figure: Simple flow-graph in GRC

Complex Float	64
Complex Float	32
Complex Integer	64
Complex Integer	32
Complex Integer	16
Complex Integer	8
Float	64
Float	32
Integer	64
Integer	32
Integer	16
Integer	8
Bits (unpacked byte)	
Async Message	
Bus Connection	
Wildcard	

Figure: Data types in GNU Radio

In-Phase and Quadrature (I/Q) Components of Signals

From the trigonometric identity:

$$\sin(\alpha + \beta) = \sin(\alpha)\cos(\beta) + \cos(\alpha)\sin(\beta)$$

every sinusoid can be expressed as the following:

$$\begin{aligned}x(t) &= A(t)\sin(\omega t + \phi) = A(t)\sin(\phi + \omega t) \\&= [A(t)\sin(\phi)]\cos(\omega t) + [A(t)\cos(\phi)]\sin(\omega t) \\&= \mathbf{I(t)}\cos(\omega t) + \mathbf{Q(t)}\sin(\omega t)\end{aligned}$$

GRC Example of Signal Generation with I/Q Components

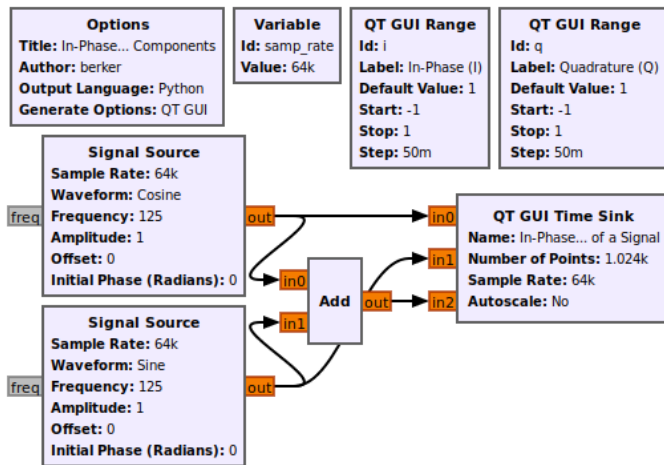


Figure: GRC flow-graph for generating I/Q signal

GRC Example of Signal Generation with I/Q Components

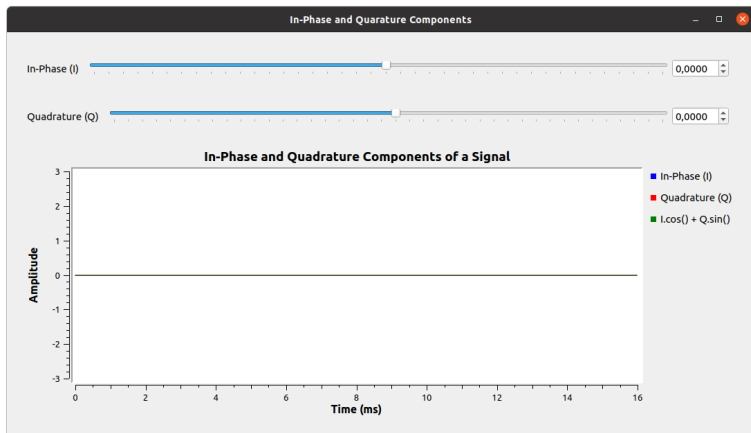


Figure: $I = 0$, $Q = 0$

GRC Example of Signal Generation with I/Q Components

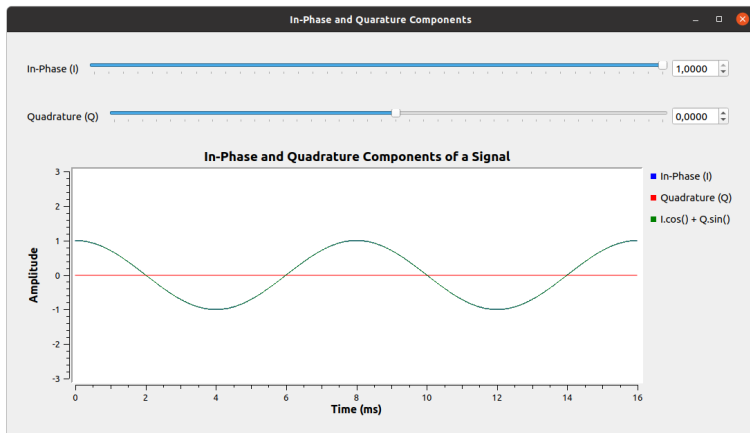


Figure: $I = 1$, $Q = 0$

GRC Example of Signal Generation with I/Q Components

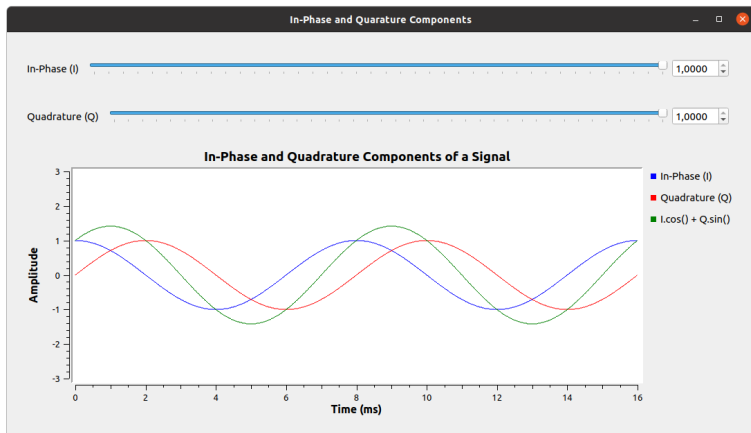


Figure: $I = 1$, $Q = 1$

Agenda

- 1 Introduction
- 2 Implementation**
- 3 Test and Results
- 4 Discussion
- 5 Future Work
- 6 Conclusion

Variables

Variable Id: samp_rate Value: 20M	Variable Id: carrier_count Value: 26	Variable Id: fft_len Value: 64	Variable Id: pilot_carriers Value: (-21, -7, 7, 21)	Variable Id: occupied_carriers Value: [-26, -2...24, 25, 26]	Variable Id: pilot_symbols Value: (1, 1, 1, -1)
Variable Id: packet_len Value: 7	Variable Id: packet_len_key Value: packet_len	Variable Id: cp_len Value: 16	Variable Id: pilot_carriers0 Value: -21, -7, 7, 21	Variable Id: occupied_carriers0 Value: [c for c in range(-...	Variable Id: pilot_symbols0 Value: 1, 1, 1, -1
Variable Id: packet_count Value: 1	Variable Id: frame_len_key Value: frame_len	Variable Id: bits_per_byte Value: 8	Variable Id: sync_word1 Value: [0., 0., 0., 0., 0....	Variable Id: sync_word2 Value: [0, 0, 0, 0, 0, 0, ...	
Variable Id: payload_modulation Value: <constellation QPSK>	Variable Id: payload_equalizer Value: <OFDM eq... simplifiedfe>	Variable Id: header_equalizer Value: <OFDM eq... simplifiedfe>	Variable Id: rolloff Value: 0		
Variable Id: header_modulation Value: <constellation BPSK>	Variable Id: header_format Value: <gnuradi...6b9930c0> >	Variable Id: header_formatter Value: <packet_header_ofdm>	Variable Id: center_freq Value: 2.5G		

Figure: Variables used in OFDM Transmitter and Receiver

OFDM Transmitter

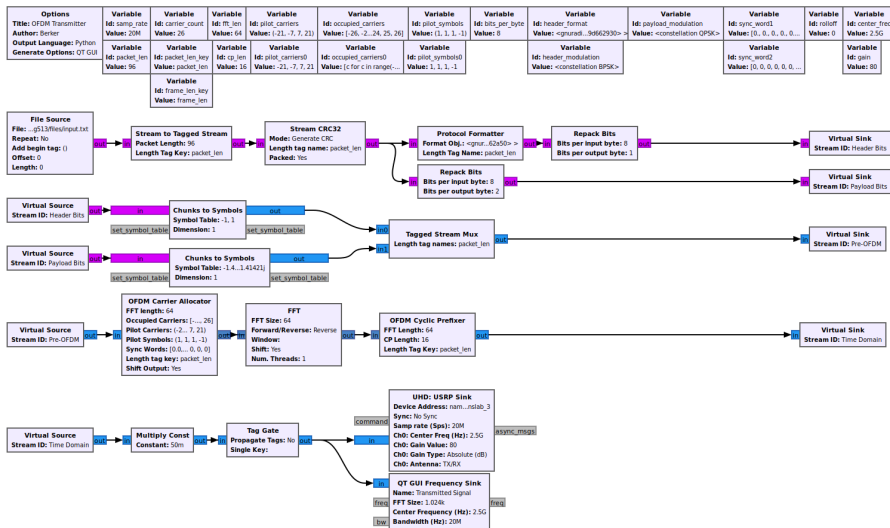


Figure: Complete GRC flow-graph of OFDM transmitter

OFDM Receiver

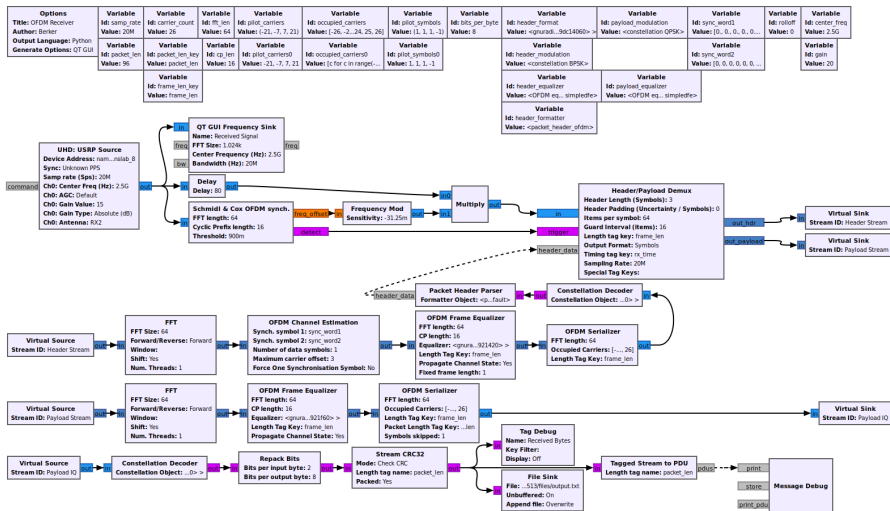


Figure: Complete GRC flow-graph of OFDM receiver

Agenda

- 1 Introduction
- 2 Implementation
- 3 Test and Results**
- 4 Discussion
- 5 Future Work
- 6 Conclusion

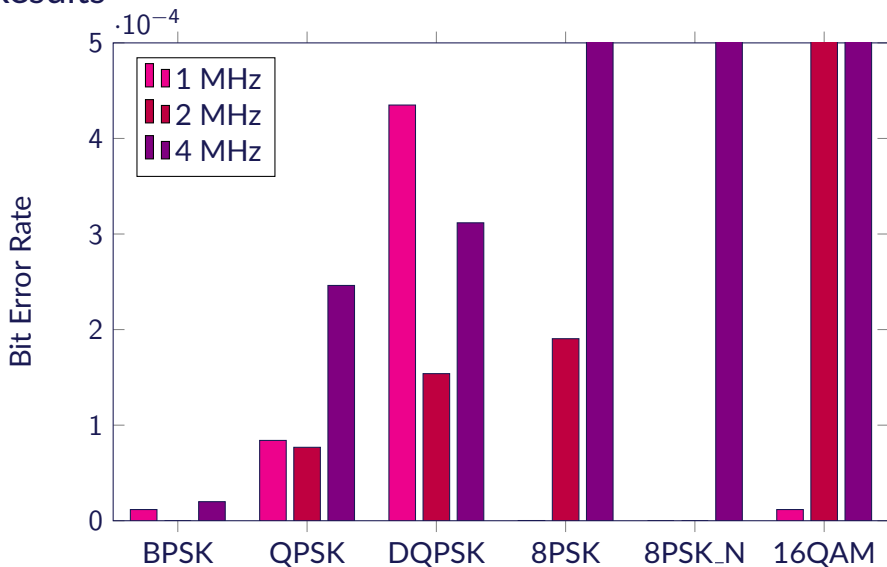
Test

Packet of size 96 bytes with known payload data is transmitted $N = 10000$ times. At the receiver side, received data and console output is written into files and these files are used for calculating:

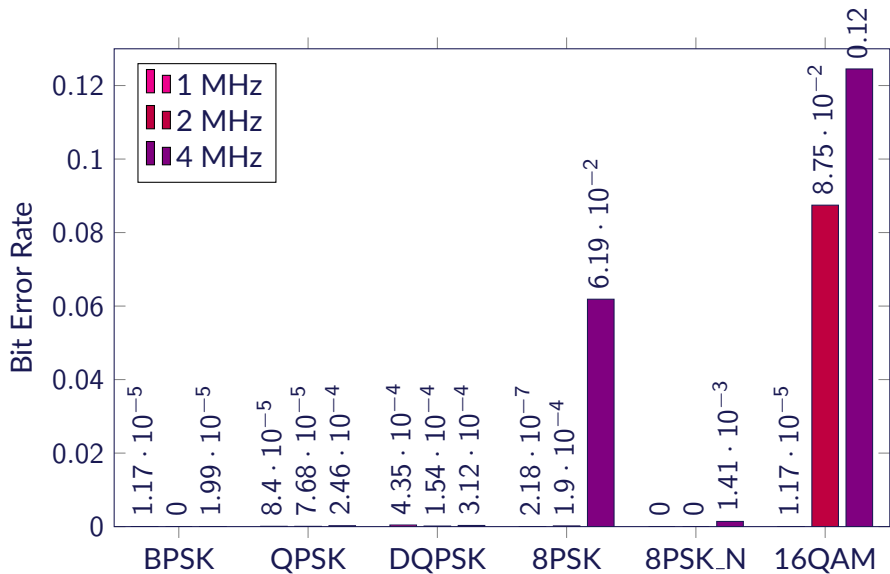
1. **Bit Error Rate**
2. **Undetected Packet Count**
3. **Reception Time**

32-bit CRC is disabled for tests and the system is tested with different OFDM parameters: Bandwidth (1 MHz, 2 MHz, 4 MHz), Payload Modulation (BPSK, QPSK, DQPSK, 8PSK, natural 8PSK, 16QAM). Tests are run 3 times per configuration and the calculations are averaged. Also, all test are done with the same transmitter and receiver antennas gain values.

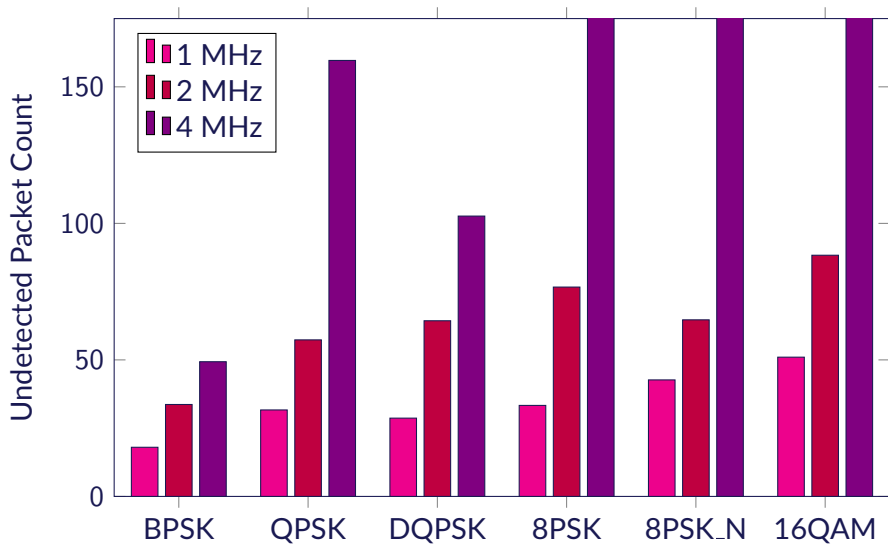
Results



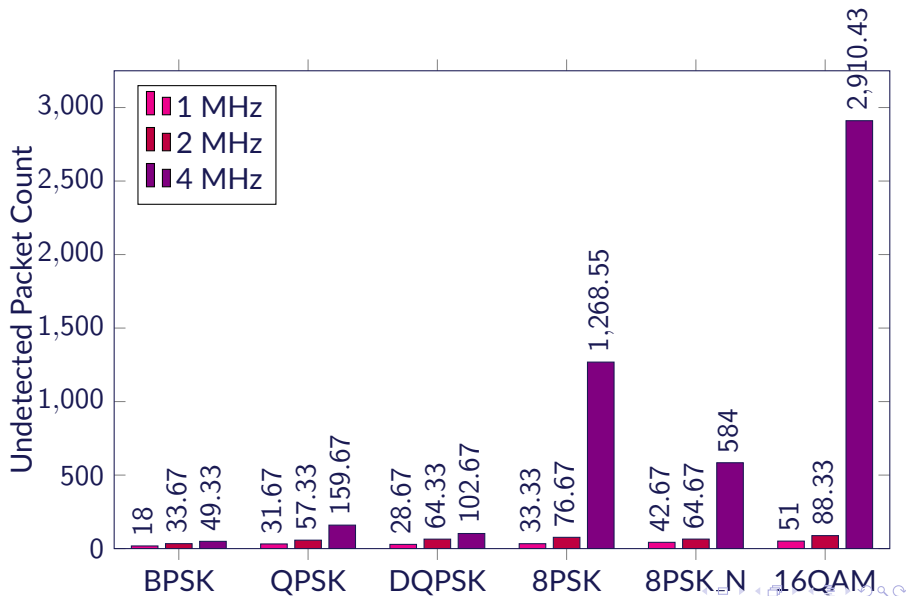
Results



Results



Results



Agenda

- 1 Introduction
- 2 Implementation
- 3 Test and Results
- 4 Discussion**
- 5 Future Work
- 6 Conclusion

Discussion

- Increasing the bandwidth increases bit error rate and undetected packet count.
- Dense modulation schemes are more prone to bit errors and undetected packets.
- Transmitter and receiver might require fine-tuning for different configuration settings. For example, the transmitter and receiver system with current parameters cannot detect incoming frames when working with 20 MHz bandwidth.

Agenda

- 1 Introduction
- 2 Implementation
- 3 Test and Results
- 4 Discussion
- 5 Future Work**
- 6 Conclusion

Future Work

- Reliable packet transmission with BEC and FEC mechanisms
- Testing the system with different OFDM parameters such as FFT length, CP length etc.
- Implementing self-adjusting system by controlling OFDM parameters such as modulation schemes, FFT length, CP length etc.
- Training a channel classification machine learning model from collected I/Q samples and deployment of the model

Agenda

- 1 Introduction
- 2 Implementation
- 3 Test and Results
- 4 Discussion
- 5 Future Work
- 6 Conclusion**

Conclusion

- OFDM transmitter and receiver design in GNU Radio software
- Real-time data transmission using USRP SDRs
- Testing of the system with different OFDM parameters
- Possible future works on the project

Questions

THANK YOU

OFDM Transmitter and Receiver with Software
Defined Radios

presented by Berker Acir



July 8, 2021

