

Optički telekomunikacijski sistemi - Analiza i dizajn MIMO sistema zasnovanih
na vlaknima sa više modova

Generated by Doxygen 1.9.8

| | |
|--|-----------|
| 1 Optički telekomunikacijski sistemi - Simulacija i analiza MIMO sistema zasnovanih na vlaknima s više modova | 3 |
| 1.1 Uvod | 3 |
| 1.2 Ključni Pojmovi | 3 |
| 1.2.1 Gubici u Optičkim Vlaknima | 3 |
| 1.2.2 MIMO Sistemi | 4 |
| 1.2.3 Obrada Signala i Mreže | 4 |
| 1.3 Simulacija QPSK MIMO sistema | 4 |
| 1.3.1 Funkcionalnost | 4 |
| 1.3.2 Biblioteke | 5 |
| 1.3.3 Kako koristiti | 5 |
| 2 Hierarchical Index | 7 |
| 2.1 Class Hierarchy | 7 |
| 3 Class Index | 9 |
| 3.1 Class List | 9 |
| 4 Class Documentation | 11 |
| 4.1 QPSK_MIMO.QPSK_MIMO_GUI Class Reference | 11 |
| 4.1.1 Member Function Documentation | 13 |
| 4.1.1.1 _calculate_ber() | 13 |
| 4.1.1.2 _calculate_capacity() | 13 |
| 4.1.1.3 _demodulate() | 14 |
| 4.1.1.4 _generate_bits() | 14 |
| 4.1.1.5 _generate_noise() | 14 |
| 4.1.1.6 _get_qpsk_mapping() | 14 |
| 4.1.1.7 _plot_channel_matrix() | 14 |
| 4.1.1.8 _plot_constellation() | 15 |
| 4.1.1.9 _plot_detailed_fiber() | 15 |
| 4.1.1.10 _plot_eye_diagram() | 15 |
| 4.1.1.11 _plot_noise_impact() [1/2] | 15 |
| 4.1.1.12 _plot_noise_impact() [2/2] | 15 |
| 4.1.1.13 _plot_snr_ber() | 16 |
| 4.1.1.14 _plot_snr_capacity() | 16 |
| 4.1.1.15 _plot_tx_signal() | 16 |
| 4.1.1.16 _qpsk_modulate() | 16 |
| 4.2 QPSK_MIMO_test.TestQPSK_MIMO Class Reference | 16 |
| 4.2.1 Member Function Documentation | 17 |
| 4.2.1.1 test_ber_all_errors() | 17 |
| 4.2.1.2 test_ber_bez_gresaka() | 18 |
| 4.2.1.3 test_ber_empty_demodulated_bits() | 18 |
| 4.2.1.4 test_ber_empty_tx_and_demodulated_bits() | 18 |

| | |
|---|-----------|
| 4.2.1.5 test_ber_empty_tx_bits() | 18 |
| 4.2.1.6 test_ber_less_demodulated_bits() | 19 |
| 4.2.1.7 test_ber_long_sequences() | 19 |
| 4.2.1.8 test_ber_long_sequences_no_errors() | 19 |
| 4.2.1.9 test_ber_unequal_lengths() | 19 |
| 4.2.1.10 test_ber_with_some_errors() | 20 |
| 4.3 QPSK_MIMO.ToolTip Class Reference | 20 |
| Index | 21 |

Chapter 1

Optički telekomunikacijski sistemi - Simulacija i analiza MIMO sistema zasnovanih na vlaknima s više modova

1.1 Uvod

Ovaj projekat se bavi simulacijom i analizom MIMO (Multiple-Input Multiple-Output) sistema u optičkim telekomunikacijama, s posebnim fokusom na primjenu u vlaknima s više modova (FMF). Cilj je istražiti kako se MIMO tehnologije mogu koristiti za povećanje kapaciteta i pouzdanosti optičkih komunikacijskih sistema.

1.2 Ključni Pojmovi

1.2.1 Gubici u Optičkim Vlaknima

- **Slabljenje (Attenuation):** Gubitak snage signala tokom prijenosa kroz optičko vlakno, mjeren u decibelima (dB).
- **Apsorpcija materijala:** Gubitak optičke energije uslijed apsorpcije svjetlosti od strane materijala vlakna.
- **Gubici uslijed raspršivanja:** Gubitak snage signala uslijed raspršivanja svjetlosti izvan jezgre vlakna.
- **Nelinearni gubici:** Gubici koji se javljaju pri visokim nivoima optičke snage, uključujući stimulirano Ramanovo raspršivanje.
- **Gubici uslijed savijanja:** Gubitak snage signala uslijed savijanja optičkog vlakna.
- **Gubici uslijed spajanja modova:** Gubitak snage uslijed prijenosa energije između različitih modova u vlaknu.

1.2.2 MIMO Sistemi

- **MIMO (Multiple-Input Multiple-Output):** Tehnologija koja koristi više antena za prijenos i prijem signala, povećavajući kapacitet i pouzdanost komunikacije.
- **BER (Bit Error Rate):** Stopa grešaka u bitovima, mjera kvalitete prijenosa podataka.
- **Prostorno multipleksiranje (Spatial Multiplexing):** Tehnika koja omogućava istovremeni prijenos više tokova podataka putem različitih antena.
- **SDM (Spatial Division Multiplexing):** Metoda za povećanje kapaciteta prijenosa korištenjem više prostornih modova u vlaknu.
- **FMF (Few-Mode Fiber):** Optičko vlakno koje podržava prijenos nekoliko prostornih modova.
- **-MIMO:** Napredna varijanta MIMO tehnologije koja koristi različite valne duljine (λ) za prijenos više signala kroz jedno optičko vlakno.
- **Massive MIMO:** MIMO sistem s velikim brojem antena, ključan za buduće širokopojasne bežične mreže.

1.2.3 Obrada Signala i Mreže

- **MMSE (Minimum Mean Square Error):** Algoritam za izjednačavanje signala koji minimizira srednju kvadratnu grešku.
- **OFDM-PON (Orthogonal Frequency Division Multiplexing Passive Optical Network):** Tehnologija za prijenos podataka putem optičke mreže koja koristi OFDM modulaciju.
- **RoF (Radio over Fiber):** Tehnika prijenosa radio signala putem optičkog vlakna.
- **Hromatska disperzija (Chromatic Dispersion):** Širenje optičkih impulsa tokom prijenosa kroz vlakno, ograničava brzinu prijenosa podataka.
- **Digitalna obrada signala (DSP):** Obrada signala u digitalnom obliku, koristi se za poboljšanje kvalitete signala i kompenzaciju raznih efekata u komunikacijskom sistemu.

1.3 Simulacija QPSK MIMO sistema

Ovaj Python kod implementira grafičko sučelje (GUI) za simulaciju QPSK (Quadrature Phase-Shift Keying) MIMO (Multiple-Input Multiple-Output) sistema u višemodnom optičkom vlaknu. Koristi biblioteke `tkinter` za GUI, `numpy` za numeričke operacije, `matplotlib` za vizualizaciju i `scipy` za obradu signala.

1.3.1 Funkcionalnost

1. GUI Parametri:

- Korisnik može postaviti parametre simulacije kao što su broj bita, SNR (omjer signala i šuma), broj predajnih i prijemnih antena, broj modova, dužinu vlakna i koeficijent slabljenja.
- Kanalna matrica se može postaviti ručno ili generirati automatski.

2. Simulacija:

- Generiše se niz bitova koji se zatim moduliraju korištenjem QPSK modulacije.
- Simulira se prijenos signala kroz MIMO kanal, uključujući efekte sprezanja modova i disperzije.
- Dodaje se AWGN (Additive White Gaussian Noise) šum na primljeni signal.

- Primjenjuje se MMSE (Minimum Mean Square Error) ekvalizator za poboljšanje kvalitete signala.
- Demoduliraju se primljeni simboli i izračunava se BER (Bit Error Rate).

3. Vizualizacija:

- Prikazuje se konstelacijski dijagram odašiljanih i primljenih signala.
- Prikazuje se kanalna matrica (magnituda i faza).
- Prikazuje se eye dijagram primljenog signala prije i poslije ekvalizacije.
- Prikazuje se utjecaj šuma na signal u vremenskoj domeni.
- Prikazuje se ovisnost BER o SNR i kapaciteta o SNR.
- Prikazuje se slabljenje signala duž vlakna.

1.3.2 Biblioteke

- `tkinter`: Za izradu grafičkog korisničkog sučelja.
- `numpy`: Za numeričke operacije i manipulaciju matricama.
- `matplotlib`: Za vizualizaciju podataka i crtanje grafova.
- `scipy`: Za obradu signala i napredne matematičke funkcije.

1.3.3 Kako koristiti

Za pokretanje simulacije, potrebno je pokrenuti skriptu `QPSK_MIMO.py`. Nakon pokretanja, korisnik može unijeti željene parametre simulacije u grafičkom sučelju (GUI). Simulacija se pokreće klikom na dugme "Simuliraj". Rezultati simulacije i grafovi se prikazuju u odgovarajućim tabovima. Za brisanje svih grafova i rezultata, koristi se dugme "Resetuj". Dodatne upute o korištenju simulacije mogu se pronaći klikom na dugme "Pomoć", a detaljnije objašnjenje QPSK MIMO koncepta u višemodnom vlaknu dostupno je klikom na dugme "Objasni koncept".

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

| | |
|--|----|
| QPSK_MIMO.QPSK_MIMO_GUI | 11 |
| unittest.TestCase | |
| QPSK_MIMO_test.TestQPSK_MIMO | 16 |
| QPSK_MIMO.ToolTip | 20 |

Chapter 3

Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

| | |
|--|----|
| QPSK_MIMO.QPSK_MIMO_GUI | 11 |
| QPSK_MIMO_test.TestQPSK_MIMO | 16 |
| QPSK_MIMO.ToolTip | 20 |

Chapter 4

Class Documentation

4.1 QPSK_MIMO.QPSK_MIMO_GUI Class Reference

Public Member Functions

- **__init__** (self, tk.Tk master)
- **show_help** (self)
- **explain_concept** (self)
- **reset_simulation** (self)
- **update_channel_matrix_size** (self, event=None)
- **simulate** (self)

Public Attributes

- **master**
- **results_frame**
- **ber_label_text**
- **ber_label**
- **snr_result_label_text**
- **snr_result_label**
- **capacity_label_text**
- **capacity_label**
- **input_frame**
- **num_bits_label**
- **num_bits_entry**
- **snr_label**
- **snr_entry**
- **num_tx_ant_label**
- **num_tx_ant_entry**
- **update_channel_matrix_size**
- **num_rx_ant_label**
- **num_rx_ant_entry**
- **num_modes_label**
- **num_modes_entry**
- **channel_label**
- **channel_entry**
- **fiber_length_label**

- fiber_length_entry
- attenuation_label
- attenuation_entry
- explain_button
- help_button
- simulate_button
- reset_button
- notebook
- tx_signal_tab
- tx_signal_figure
- tx_signal_ax
- tx_signal_canvas
- tx_signal_canvas_widget
- constellation_tab
- constellation_figure
- constellation_ax
- constellation_canvas
- constellation_canvas_widget
- channel_tab
- channel_figure
- channel_ax
- channel_canvas
- channel_canvas_widget
- eye_diagram_tab
- eye_diagram_figure
- eye_diagram_ax
- eye_diagram_canvas
- eye_diagram_canvas_widget
- noise_impact_tab
- noise_impact_figure
- noise_impact_ax
- noise_impact_canvas
- noise_impact_canvas_widget
- snr_ber_tab
- snr_ber_figure
- snr_ber_ax
- snr_ber_canvas
- snr_ber_canvas_widget
- snr_capacity_tab
- snr_capacity_figure
- snr_capacity_ax
- snr_capacity_canvas
- snr_capacity_canvas_widget
- detailed_fiber_tab
- detailed_fiber_figure
- detailed_fiber_ax
- detailed_fiber_canvas
- detailed_fiber_canvas_widget
- channel_matrix_displayed
- fiber_propagation_ax
- fiber_propagation_canvas
- SNR_RANGE
- SNR_POINTS
- FIBER_LENGTH_POINTS
- RANDOM_SEED

Static Public Attributes

- tuple **SNR_RANGE** = (0, 20)
- int **SNR_POINTS** = 10
- int **FIBER_LENGTH_POINTS** = 100
- int **RANDOM_SEED** = 42

Protected Member Functions

- [_plot_noise_impact](#) (self, np.ndarray tx_signals, np.ndarray noise, np.ndarray received_symbols)
- np.ndarray [_generate_bits](#) (self, int num_bits)
- Tuple[np.ndarray, List[int]] [_qpsk_modulate](#) (self, np.ndarray bits)
- np.ndarray [_generate_noise](#) (self, np.ndarray signals, float snr_db)
- Dict[Tuple[int, int], complex] [_get_qpsk_mapping](#) (self)
- List[int] [_demodulate](#) (self, np.ndarray received_symbols, qpsk_mapping)
- float [_calculate_ber](#) (self, List[int] tx_bits, List[int] rx_bits)
- float [_calculate_capacity](#) (self, np.ndarray H, float snr_db, int num_tx_antennas, int num_rx_antennas, int num_modes)
- [_plot_tx_signal](#) (self, np.ndarray qpsk_symbols)
- [_plot_constellation](#) (self, np.ndarray received_symbols, Dict[Tuple[int, int], complex] mapping, int num_rx_antennas, int num_modes)
- [_plot_channel_matrix](#) (self, np.ndarray H, int num_tx_antennas, int num_rx_antennas, int num_modes)
- [_plot_snr_ber](#) (self, np.ndarray snr_db_range, List[float] ber_values)
- [_plot_snr_capacity](#) (self, np.ndarray snr_db_range, List[float] capacity_values)
- [_plot_noise_impact](#) (self, np.ndarray tx_signals, np.ndarray noise, np.ndarray received_symbols)
- [_plot_detailed_fiber](#) (self, float fiber_length, float attenuation)
- [_plot_eye_diagram](#) (self, np.ndarray received_symbols, int num_tx_antennas, int num_modes)

4.1.1 Member Function Documentation

4.1.1.1 [_calculate_ber\(\)](#)

```
float QPSK_MIMO.QPSK_MIMO_GUI._calculate_ber (
    self,
    List[int] tx_bits,
    List[int] rx_bits ) [protected]
```

Calculate the Bit Error Rate.

4.1.1.2 [_calculate_capacity\(\)](#)

```
float QPSK_MIMO.QPSK_MIMO_GUI._calculate_capacity (
    self,
    np.ndarray H,
    float snr_db,
    int num_tx_antennas,
    int num_rx_antennas,
    int num_modes ) [protected]
```

Calculate the channel capacity.

4.1.1.3 `_demodulate()`

```
List[int] QPSK_MIMO.QPSK_MIMO_GUI._demodulate (  
    self,  
    np.ndarray received_symbols,  
    qpsk_mapping ) [protected]
```

Demodulate the received symbols.

4.1.1.4 `_generate_bits()`

```
np.ndarray QPSK_MIMO.QPSK_MIMO_GUI._generate_bits (  
    self,  
    int num_bits ) [protected]
```

Generate random bits.

4.1.1.5 `_generate_noise()`

```
np.ndarray QPSK_MIMO.QPSK_MIMO_GUI._generate_noise (  
    self,  
    np.ndarray signals,  
    float snr_db ) [protected]
```

Generate AWGN noise.

4.1.1.6 `_get_qpsk_mapping()`

```
Dict[Tuple[int, int], complex] QPSK_MIMO.QPSK_MIMO_GUI._get_qpsk_mapping (  
    self ) [protected]
```

Return the QPSK mapping dictionary.

4.1.1.7 `_plot_channel_matrix()`

```
QPSK_MIMO.QPSK_MIMO_GUI._plot_channel_matrix (  
    self,  
    np.ndarray H,  
    int num_tx_antennas,  
    int num_rx_antennas,  
    int num_modes ) [protected]
```

Plot the channel matrix.

4.1.1.8 `_plot_constellation()`

```
QPSK_MIMO.QPSK_MIMO_GUI._plot_constellation (
    self,
    np.ndarray received_symbols,
    Dict[Tuple[int, int], complex] mapping,
    int num_rx_antennas,
    int num_modes ) [protected]
```

Plot the constellation diagram.

4.1.1.9 `_plot_detailed_fiber()`

```
QPSK_MIMO.QPSK_MIMO_GUI._plot_detailed_fiber (
    self,
    float fiber_length,
    float attenuation ) [protected]
```

Plot the detailed fiber propagation.

4.1.1.10 `_plot_eye_diagram()`

```
QPSK_MIMO.QPSK_MIMO_GUI._plot_eye_diagram (
    self,
    np.ndarray received_symbols,
    int num_tx_antennas,
    int num_modes ) [protected]
```

Plot the eye diagram.

4.1.1.11 `_plot_noise_impact()` [1/2]

```
QPSK_MIMO.QPSK_MIMO_GUI._plot_noise_impact (
    self,
    np.ndarray tx_signals,
    np.ndarray noise,
    np.ndarray received_symbols ) [protected]
```

Plot the impact of noise on the signal.

4.1.1.12 `_plot_noise_impact()` [2/2]

```
QPSK_MIMO.QPSK_MIMO_GUI._plot_noise_impact (
    self,
    np.ndarray tx_signals,
    np.ndarray noise,
    np.ndarray received_symbols ) [protected]
```

Plot the impact of noise on the signal.

4.1.1.13 `_plot_snr_ber()`

```
QPSK_MIMO.QPSK_MIMO_GUI._plot_snr_ber (
    self,
    np.ndarray snr_db_range,
    List[float] ber_values ) [protected]
```

Plot SNR vs BER.

4.1.1.14 `_plot_snr_capacity()`

```
QPSK_MIMO.QPSK_MIMO_GUI._plot_snr_capacity (
    self,
    np.ndarray snr_db_range,
    List[float] capacity_values ) [protected]
```

Plot SNR vs Capacity.

4.1.1.15 `_plot_tx_signal()`

```
QPSK_MIMO.QPSK_MIMO_GUI._plot_tx_signal (
    self,
    np.ndarray qpsk_symbols ) [protected]
```

Plot the transmitted signal.

4.1.1.16 `_qpsk_modulate()`

```
Tuple[np.ndarray, List[int]] QPSK_MIMO.QPSK_MIMO_GUI._qpsk_modulate (
    self,
    np.ndarray bits ) [protected]
```

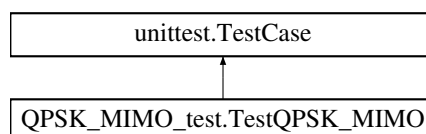
QPSK modulate the bits.

The documentation for this class was generated from the following file:

- Kodovi/QPSK_MIMO.py

4.2 QPSK_MIMO_test.TestQPSK_MIMO Class Reference

Inheritance diagram for QPSK_MIMO_test.TestQPSK_MIMO:



Public Member Functions

- **setUp** (self)
- [test_ber_bez_gresaka](#) (self)
Testira BER kada nema grešaka u prijemnim bitima.
- [test_ber_all_errors](#) (self)
Testira BER kada su svi prijemni biti pogrešni.
- [test_ber_less_demodulated_bits](#) (self)
Testira BER kada je broj demoduliranih bita manji od broja poslanih bita.
- [test_ber_empty_tx_bits](#) (self)
Testira BER kada su predajni biti prazni.
- [test_ber_empty_demodulated_bits](#) (self)
Testira BER kada su prijemni biti prazni.
- [test_ber_unequal_lengths](#) (self)
Testira BER kada su dužine predajnih i prijemnih bita nejednake.
- [test_ber_with_some_errors](#) (self)
Testira BER kada postoje neke greške u prijemnim bitima.
- [test_ber_long_sequences](#) (self)
Testira BER sa dugim nizovima bita i svim greškama.
- [test_ber_long_sequences_no_errors](#) (self)
Testira BER sa dugim nizovima bita bez grešaka.
- [test_ber_empty_tx_and_demodulated_bits](#) (self)
Testira BER kada su predajni i prijemni biti prazni.
- **tearDown** (self)

Public Attributes

- **root**
- **gui**

4.2.1 Member Function Documentation

4.2.1.1 test_ber_all_errors()

```
QPSK_MIMO_test.TestQPSK_MIMO.test_ber_all_errors (
    self )
```

Testira BER kada su svi prijemni biti pogrešni.

Testira slučaj kada su svi primljeni biti suprotni od poslanih.

@brief Testira BER kada su svi prijemni biti pogrešni.

@details Testira slučaj kada su svi primljeni biti suprotni od poslanih.

4.2.1.2 test_ber_bez_gresaka()

```
QPSK_MIMO_test.TestQPSK_MIMO.test_ber_bez_gresaka (
    self )
```

Testira BER kada nema grešaka u prijemnim bitima.

Testira slučaj kada su svi primljeni biti identični poslanim.

```
@brief Testira BER kada nema grešaka u prijemnim bitima.
@details Testira slučaj kada su svi primljeni biti identični poslanim.
```

4.2.1.3 test_ber_empty_demodulated_bits()

```
QPSK_MIMO_test.TestQPSK_MIMO.test_ber_empty_demodulated_bits (
    self )
```

Testira BER kada su prijemni biti prazni.

Testira slučaj kada nema primljenih bita.

```
@brief Testira BER kada su prijemni biti prazni.
@details Testira slučaj kada nema primljenih bita.
```

4.2.1.4 test_ber_empty_tx_and_demodulated_bits()

```
QPSK_MIMO_test.TestQPSK_MIMO.test_ber_empty_tx_and_demodulated_bits (
    self )
```

Testira BER kada su predajni i prijemni biti prazni.

Testira slučaj kada nema poslanih i primljenih bita.

```
@brief Testira BER kada su predajni i prijemni biti prazni.
@details Testira slučaj kada nema poslanih i primljenih bita.
```

4.2.1.5 test_ber_empty_tx_bits()

```
QPSK_MIMO_test.TestQPSK_MIMO.test_ber_empty_tx_bits (
    self )
```

Testira BER kada su predajni biti prazni.

Testira slučaj kada nema poslanih bita.

```
@brief Testira BER kada su predajni biti prazni.
@details Testira slučaj kada nema poslanih bita.
```

4.2.1.6 test_ber_less_demodulated_bits()

```
QPSK_MIMO_test.TestQPSK_MIMO.test_ber_less_demodulated_bits (
    self )
```

Testira BER kada je broj demoduliranih bita manji od broja poslanih bita.

Testira slučaj kada je broj primljenih bita manji od broja poslanih bita.

```
@brief Testira BER kada je broj demoduliranih bita manji od broja poslanih bita.
@details Testira slučaj kada je broj primljenih bita manji od broja poslanih bita.
```

4.2.1.7 test_ber_long_sequences()

```
QPSK_MIMO_test.TestQPSK_MIMO.test_ber_long_sequences (
    self )
```

Testira BER sa dugim nizovima bita i svim greškama.

Testira slučaj kada su svi primljeni biti suprotni od poslanih u dugom nizu.

```
@brief Testira BER sa dugim nizovima bita i svim greškama.
@details Testira slučaj kada su svi primljeni biti suprotni od poslanih u dugom nizu.
```

4.2.1.8 test_ber_long_sequences_no_errors()

```
QPSK_MIMO_test.TestQPSK_MIMO.test_ber_long_sequences_no_errors (
    self )
```

Testira BER sa dugim nizovima bita bez grešaka.

Testira slučaj kada su svi primljeni biti identični poslanim u dugom nizu.

```
@brief Testira BER sa dugim nizovima bita bez grešaka.
@details Testira slučaj kada su svi primljeni biti identični poslanim u dugom nizu.
```

4.2.1.9 test_ber_unequal_lengths()

```
QPSK_MIMO_test.TestQPSK_MIMO.test_ber_unequal_lengths (
    self )
```

Testira BER kada su dužine predajnih i prijemnih bita nejednake.

Testira slučaj kada broj poslanih i primljenih bita nije isti.

```
@brief Testira BER kada su dužine predajnih i prijemnih bita nejednake.
@details Testira slučaj kada broj poslanih i primljenih bita nije isti.
```

4.2.1.10 test_ber_with_some_errors()

```
QPSK_MIMO_test.TestQPSK_MIMO.test_ber_with_some_errors (
    self )
```

Testira BER kada postoje neke greške u prijemnim bitima.

Testira slučaj kada su neki primljeni biti različiti od poslanih.

```
@brief Testira BER kada postoje neke greške u prijemnim bitima.
@details Testira slučaj kada su neki primljeni biti različiti od poslanih.
```

The documentation for this class was generated from the following file:

- Kodovi/QPSK_MIMO_test.py

4.3 QPSK_MIMO.ToolTip Class Reference

Public Member Functions

- **__init__** (self, tk.Widget widget, str text)
- **show_tooltip** (self, event=None)
- **hide_tooltip** (self, event=None)

Public Attributes

- **widget**
- **text**
- **tooltip_window**
- **show_tooltip**
- **hide_tooltip**

The documentation for this class was generated from the following file:

- Kodovi/QPSK_MIMO.py

Index

- [_calculate_ber](#)
 - [QPSK_MIMO.QPSK_MIMO_GUI, 13](#)
 - [_calculate_capacity](#)
 - [QPSK_MIMO.QPSK_MIMO_GUI, 13](#)
 - [_demodulate](#)
 - [QPSK_MIMO.QPSK_MIMO_GUI, 13](#)
 - [_generate_bits](#)
 - [QPSK_MIMO.QPSK_MIMO_GUI, 14](#)
 - [_generate_noise](#)
 - [QPSK_MIMO.QPSK_MIMO_GUI, 14](#)
 - [_get_qpsk_mapping](#)
 - [QPSK_MIMO.QPSK_MIMO_GUI, 14](#)
 - [_plot_channel_matrix](#)
 - [QPSK_MIMO.QPSK_MIMO_GUI, 14](#)
 - [_plot_constellation](#)
 - [QPSK_MIMO.QPSK_MIMO_GUI, 14](#)
 - [_plot_detailed_fiber](#)
 - [QPSK_MIMO.QPSK_MIMO_GUI, 15](#)
 - [_plot_eye_diagram](#)
 - [QPSK_MIMO.QPSK_MIMO_GUI, 15](#)
 - [_plot_noise_impact](#)
 - [QPSK_MIMO.QPSK_MIMO_GUI, 15](#)
 - [_plot_snr_ber](#)
 - [QPSK_MIMO.QPSK_MIMO_GUI, 15](#)
 - [_plot_snr_capacity](#)
 - [QPSK_MIMO.QPSK_MIMO_GUI, 16](#)
 - [_plot_tx_signal](#)
 - [QPSK_MIMO.QPSK_MIMO_GUI, 16](#)
 - [_qpsk_modulate](#)
 - [QPSK_MIMO.QPSK_MIMO_GUI, 16](#)
- Optički telekomunikacijski sistemi - Simulacija i analiza MIMO sistema zasnovanih na vlaknima s više modova, [3](#)
- QPSK_MIMO.QPSK_MIMO_GUI, [11](#)
 - [_calculate_ber, 13](#)
 - [_calculate_capacity, 13](#)
 - [_demodulate, 13](#)
 - [_generate_bits, 14](#)
 - [_generate_noise, 14](#)
 - [_get_qpsk_mapping, 14](#)
 - [_plot_channel_matrix, 14](#)
 - [_plot_constellation, 14](#)
 - [_plot_detailed_fiber, 15](#)
 - [_plot_eye_diagram, 15](#)
 - [_plot_noise_impact, 15](#)
 - [_plot_snr_ber, 15](#)
 - [_plot_snr_capacity, 16](#)
 - [_plot_tx_signal, 16](#)
 - [_qpsk_modulate, 16](#)
- QPSK_MIMO.ToolTip, [20](#)
- QPSK_MIMO_test.TestQPSK_MIMO, [16](#)
 - [test_ber_all_errors, 17](#)
 - [test_ber_bez_gresaka, 17](#)
 - [test_ber_empty_demodulated_bits, 18](#)
 - [test_ber_empty_tx_and_demodulated_bits, 18](#)
 - [test_ber_empty_tx_bits, 18](#)
 - [test_ber_less_demodulated_bits, 18](#)
 - [test_ber_long_sequences, 19](#)
 - [test_ber_long_sequences_no_errors, 19](#)
 - [test_ber_unequal_lengths, 19](#)
 - [test_ber_with_some_errors, 19](#)
- [test_ber_all_errors](#)
 - [QPSK_MIMO_test.TestQPSK_MIMO, 17](#)
- [test_ber_bez_gresaka](#)
 - [QPSK_MIMO_test.TestQPSK_MIMO, 17](#)
- [test_ber_empty_demodulated_bits](#)
 - [QPSK_MIMO_test.TestQPSK_MIMO, 18](#)
- [test_ber_empty_tx_and_demodulated_bits](#)
 - [QPSK_MIMO_test.TestQPSK_MIMO, 18](#)
- [test_ber_empty_tx_bits](#)
 - [QPSK_MIMO_test.TestQPSK_MIMO, 18](#)
- [test_ber_less_demodulated_bits](#)
 - [QPSK_MIMO_test.TestQPSK_MIMO, 18](#)
- [test_ber_long_sequences](#)
 - [QPSK_MIMO_test.TestQPSK_MIMO, 19](#)
- [test_ber_long_sequences_no_errors](#)
 - [QPSK_MIMO_test.TestQPSK_MIMO, 19](#)
- [test_ber_unequal_lengths](#)
 - [QPSK_MIMO_test.TestQPSK_MIMO, 19](#)
- [test_ber_with_some_errors](#)
 - [QPSK_MIMO_test.TestQPSK_MIMO, 19](#)