

# NIDUC 2

## INEK00025P

### Projekt

Wydział Informatyki i Telekomunikacji	Kierunek: Informatyka Techniczna
Grupa zajęciowa: ŚR TN 9:15	Semestr: 2022/23 LATO
Nazwisko i imię: Jakub Chuchła	Nr indeksu: 264472
Nazwisko i imię: Tomasz Musz	Nr indeksu: 264474
Nazwisko i imię: Marcin Gnap	Nr indeksu: 258953
Prowadzący: Dr hab. inż. Henryk Maciejewski	

#### TEMAT:

Transmisja w systemie FEC (Forward Error Correction)

OCENA:

PUNKTY:

11 maja 2023

## Spis treści

<b>1</b>	<b>Założenia projektowe</b>	<b>3</b>
<b>2</b>	<b>Kody</b>	<b>3</b>
<b>3</b>	<b>Kanały</b>	<b>3</b>
<b>4</b>	<b>Stack technologiczny</b>	<b>3</b>
<b>5</b>	<b>Wyniki i spostrzeżenia</b>	<b>4</b>
5.1	Spostrzeżenia . . . . .	4
5.2	Wyniki . . . . .	4
5.2.1	12 bit, 10000 powtórzeń . . . . .	4
5.2.2	32 bit, 10000 powtórzeń . . . . .	4
5.2.3	100 bit, 10000 powtórzeń . . . . .	4
5.2.4	512 bit, 5000 powtórzeń . . . . .	5
5.2.5	1024 bit, 1000 powtórzeń . . . . .	5
5.2.6	16 384 bit, 100 powtórzeń . . . . .	5

## 1 Założenia projektowe

- **Matplotlib** – biblioteka do tworzenia wykresów dla języka programowania Python i jego rozszerzenia numerycznego NumPy. Zawiera ona API „pylab” zaprojektowane tak aby było jak najbardziej podobne do MATLABa.
- **NumPy** – otwartoźródłowa biblioteka programistyczna dla języka Python, dodająca obsługę dużych, wielowymiarowych tabel i macierzy.
- **Komm** - otwartoźródłowa biblioteka programistyczna dla języka Python, dodająca narzędzia do analizy i symulacji analogowych i cyfrowych systemów komunikacyjnych.
- **Pandas** - biblioteka oprogramowania napisana dla języka programowania Python do manipulacji i analizy danych. W szczególności oferuje struktury danych i operacje służące do manipulowania tabelami liczbowymi i szeregami czasowymi.

## 2 Kody

- **Hamminga**
- **Potrojeniowy**
- **BCH** (Bose–Chaudhuri–Hocquenghem)

## 3 Kanały

- **Binary symmetric**
- **Modele Gilberta-Elliota**

## 4 Stack technologiczny

- **Visual Studio Code**

## 5 Wyniki i spostrzeżenia

### 5.1 Spostrzeżenia

- Najwolniejszy kod - BCH
- Najszybszy kod - Hamminga
- Dla dłuższych ciągów bitów BCH staje się nieporównywalnie wolniejszy od pozostałych kodów
- Kod Hamming'a daje najmniej wiarygodne wyniki
- BSC daje mniej miarodajne wyniki niż kanał Gilbert'a-Eliot'a

### 5.2 Wyniki

#### 5.2.1 12 bit, 10000 powtórzeń

- Potrojeniowy, BSC - 0,0230 %
- Potrojeniowy, Gilbert-Eliot - 0,0111 %
- Hamming, BSC - 0,0895 %
- Hamming, Gilbert-Eliot - 0,0276 %
- BCH, BSC - 0,0552 %
- BCH, Gilbert-Eliot - 0,0204 %

#### 5.2.2 32 bit, 10000 powtórzeń

- Potrojeniowy, BSC - 0,0228 %
- Potrojeniowy, Gilbert-Eliot - 0,0115 %
- Hamming, BSC - 0,1007 %
- Hamming, Gilbert-Eliot - 0,0463 %
- BCH, BSC - 0,0564 %
- BCH, Gilbert-Eliot - 0,0249 %

#### 5.2.3 100 bit, 10000 powtórzeń

- Potrojeniowy, BSC - 0,0229 %
- Potrojeniowy, Gilbert-Eliot - 0,0119 %
- Hamming, BSC - 0,0963 %
- Hamming, Gilbert-Eliot - 0,0552 %
- BCH, BSC - 0,0560 %
- BCH, Gilbert-Eliot - 0,0268 %

#### **5.2.4 512 bit, 5000 powtórzeń**

- Potrojeniowy, BSC - 0,0229 %
- Potrojeniowy, Gilbert-Eliot - 0,0113 %
- Hamming, BSC - 0,0911 %
- Hamming, Gilbert-Eliot - 0,0508 %
- BCH, BSC - 0,0560 %
- BCH, Gilbert-Eliot - 0,0274 %

#### **5.2.5 1024 bit, 1000 powtórzeń**

- Potrojeniowy, BSC - 0,0229 %
- Potrojeniowy, Gilbert-Eliot - 0,0121 %
- Hamming, BSC - 0,0907 %
- Hamming, Gilbert-Eliot - 0,0507 %
- BCH, BSC - 0,0561 %
- BCH, Gilbert-Eliot - 0,0271 %

#### **5.2.6 16 384 bit, 100 powtórzeń**

- Potrojeniowy, BSC - 0,0228 %
- Potrojeniowy, Gilbert-Eliot - 0,0122 %
- Hamming, BSC - 0,0900 %
- Hamming, Gilbert-Eliot - 0,0497 %
- BCH, BSC - 0,0560 %
- BCH, Gilbert-Eliot - 0,0277 %