

Budowa prostego toru transmisyjnego

1) Wybierając użyteczne fragmenty kodu pisane podczas wcześniejszych zajęć laboratoryjnych zaimplementuj prosty pseudo tor transmisyjny:

```
public void start(Stage primaryStage) throws Exception {
    S2BS s2BS = new S2BS();
    S2BSChart s2BSChart = new S2BSChart();
    String text = "P";
    Hamming hamming = new Hamming();
    int len = s2BS.stringToBinaryStream(text, isBigEndian: false).length();
    int[][] hCode = hamming.hamming74Encode(text);
    System.out.println("Text: " + s2BS.stringToBinaryStream(text, isBigEndian: false));
    System.out.println("Hamming: " + hamming.arrayToString(hCode, row: len / 4, col: 7));

    ChartDetails clock = new CLK().clk( frequency: 20, start: 0, stop: 1, stepsValue: 2000);
    ChartDetails ttl = s2BSChart.stringToBinaryStream(text, Boolean.FALSE);
    ChartDetails men = new Manchester().menchester(clock.getValues(), hamming.arrayToString(hCode, row: len / 4, col: 7));
    StringBuilder decMen = new Manchester().decManchester(men.getValues(), size: 14 * text.length());
    System.out.println("Wynik dekodowania:\n" + decMen.toString());
    makeChart(primaryStage, clock, ttl, men);

    int[][] arr = hamming.strToArr(decMen, row: len / 4, col: 7);

    int[][] hDetected = hamming.hammingDetect(arr, len);
    System.out.println("Detected: " + hamming.arrayToString(hDetected, row: len / 4, col: 3));
    int[][] hDecoded = hamming.hammingDecode(hDetected, arr, len);
    System.out.println("Decoded: " + hamming.arrayToString(hDecoded, row: 2, col: 4));
}
```

Kody używane w public void start są z poprzednich laboratoriów.

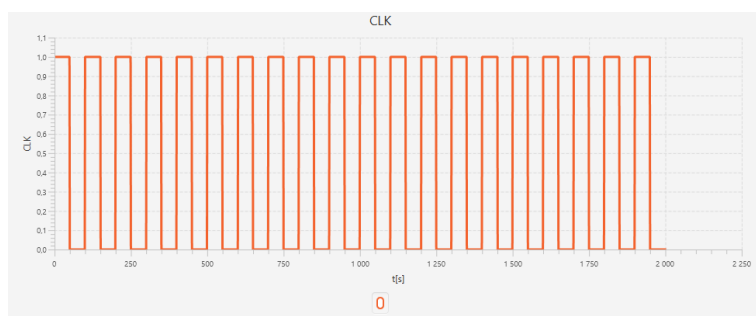
Na etapach kolejno oznaczonych numerami wygeneruj:

1. dane wejściowe
2. dane zabezpieczone kodem kanałowym
3. wykresy trzech modulacji
4. zdemodulowane dane zabezpieczone kodem kanałowym
5. zdekodowany strumień binarny

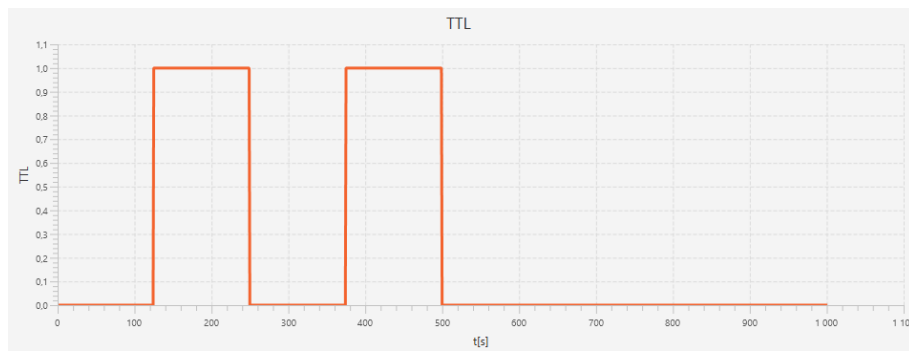
Dane:

```
Text:01010000
Hamming: 01010100000000
Demodulation:01010100000000
Detected: 000000
Decoded: 01010000
```

Wykres CLK:



Wykres TTL:



Wykres Manchester:

