

# Sprawozdanie z laboratorium 5. z przedmiotu TRA prowadzonego w semestrze 24Z

Piotr Sienkiewicz 324 887

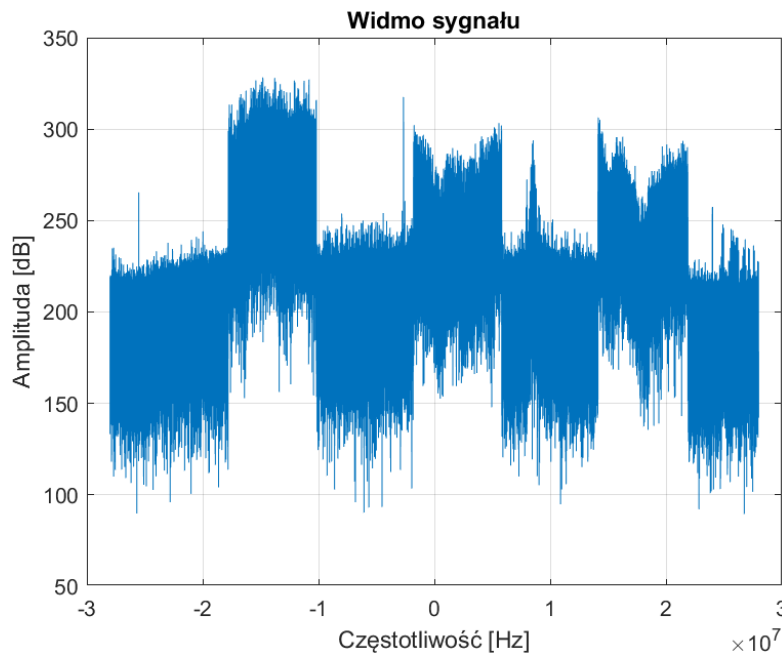
## 1 Zadanie 5a

Te laboratorium miało na celu zademonstrowanie praktycznego wykorzystania dotychczasowej wiedzy zdobytej na przedmiocie Techniki algorytmów cyfrowego przetwarzania sygnałów. Pierwszym zadaniem było wczytanie danych z zamieszczonego na stronie przedmiotu workspace'u zawierającego dwa wektory danych zebranych przez odbiornik DVB-T, który wykorzystuje mechanizm zwielokrotniania z ortogonalnym podziałem częstotliwości. Wektory te zawierają wartości odpowiadające części rzeczywistej  $I$  oraz urojonej  $Q$ . Dane te zostały spróbkowane z częstotliwością 56MHz. Następnie na podstawie odczytanych danych zostało narysowane widmo sygnału za pomocą poniższego skryptu:

Listing 1: Skrypt MATLAB do zadania 5a.

```
1 load("iio_535M_56M_1848.mat");
2 I = cf_ad9361_lpc_voltage0;
3 Q = cf_ad9361_lpc_voltage1;
4
5 S = I + 1j * Q;
6
7 % Parametry
8 fs = 56e6; % Czesotliwosc probkowania
9 N = length(S); % Dlugosc sygnalu
10
11 % Obliczenie widma
12 S_fft = fft(S, N);
13 S_fft_shifted = fftshift(S_fft);
14 f = (-N/2:N/2-1) * (fs/N);
15
16 % Wyswietlenie widma
17 figure;
18 plot(f, 20*log(abs(S_fft_shifted)));
19 xlabel('Czesotliwosc [Hz]');
20 ylabel('Amplituda [dB]');
21 title('Widmo sygnalu');
22 grid on;
```

Na widmie sygnału widać dominujący kanał zajmujący zakres częstotliwości od około -18 do -10MHz, dlatego też wybrano go jako sygnał wykorzystywany w kolejnych zadaniach.



Rysunek 1: Widmo zespolone odebranego sygnału DVB-T

## 2 Zadanie 5b

Aby dokonać właściwego przemodulowania sygnału, na początku odczytano z widma wygenerowanego w poprzednim zadaniu numer próbki, w której zaczyna się sygnał wybranego kanału, oraz numer próbki, w której się on kończy. W ten sposób, obliczając średnią obu tych liczb, uzyskano środek pasma danego kanału wynoszący 262204 (w próbkach). Modulacja polega na przemnożeniu sygnału przez sygnał:

$$e^{-j2\pi f_c t} \quad (1)$$

czyli nośną o częstotliwości:

$$f_c = \frac{n_{ch}}{N} \cdot f_s = \frac{262204}{1048576} \cdot 56 \text{ MHz} \approx 14 \text{ MHz} \quad (2)$$

gdzie  $N$  to ilość próbek sygnału,  $n_{ch}$  to odczytany wcześniej numer próbki odpowiadający środkowi pasma sygnału, a  $f_s$  to częstotliwość próbkowania.

Jak wiadomo, po zmieszaniu dwóch sygnałów rzeczywistych otrzymuje się sygnał, którego widmo składa się z różnicy oraz sumy częstotliwości sygnałów wejściowych. W tym przypadku jednak mnoży się sygnał w dziedzinie czasu (taką samą operację dokonuje mieszacz) z sygnałem zespolonym, który działa na widmo niesymetrycznie. W ten sposób uzyskuje się przesunięcie widma sygnału w lewo w dziedzinie częstotliwości, gdyż przemnożono przez ujemną częstotliwość. Przesuwając tak widmo, dokonuje się także jego przealiasowanie, przez co uzyskuje się sygnał, w którym pasmo pożądanego kanału jest symetryczne względem 0Hz, a także będący odbiciem lustrzanym, co jednak nie przeszkadza w przeprowadzaniu dalszych przekształceń sygnału.

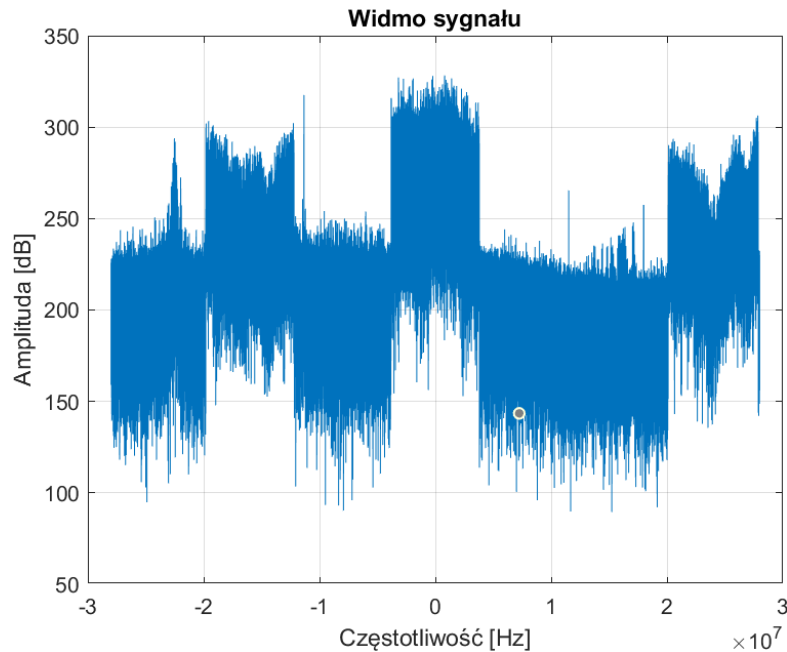
Używając poniższego skryptu przeprowadzono operację opisaną powyżej i narysowano widmo nowo otrzymanego sygnału:

Listing 2: Skrypt MATLAB do zadania 5b.

```

1 load("iio_535M_56M_1848.mat");
2 I = cf_ad9361_lpc_voltage0;
3 Q = cf_ad9361_lpc_voltage1;
4
5 S = I + 1j * Q;
6
7 % Parametry
8 fs = 56e6; % Czystotliwosc probkowania
9 N = length(S); % Dlugosc sygnalu
10
11 % Przemodulowanie sygnalu
12 Y = S' .* exp(-1j*2*pi*262204/N*(0:N-1));
13
14 % Obliczenie widma
15 Y_fft = fft(Y, N);
16 Y_fft_shifted = fftshift(Y_fft);
17 f = (-N/2:N/2-1) * (fs/N);
18
19 % Wyświetlenie widma
20 figure;
21 plot(f, 20*log(abs(Y_fft_shifted)));
22 xlabel('Czystotliwosc [Hz]');
23 ylabel('Amplituda [dB]');
24 title('Widmo sygnalu');
25 grid on;

```



Rysunek 2: Widmo zespolone odebranego sygnału DVB-T po przemodulowaniu

### 3 Zadanie 5c

Pasmo OFDM w przypadku telewizji DVB-T w Europie wynosi 8MHz, z czego pasmo użyteczne wynosi 7,61MHz, a reszta to tak zwane pasmo ochronne (guard interval) zabezpieczające przed interferencjami między kanałami. Aby uzyskać z odebranego sygnału dany kanał, należy dokonać jego filtracji. Jako że pasmo kanału ma 8MHz i jest symetryczne względem 0Hz, zastosowano filtr dolnoprzepustowy FIR o częstotliwości granicznej 4MHz, co odpowiada częstotliwości znormalizowanej:

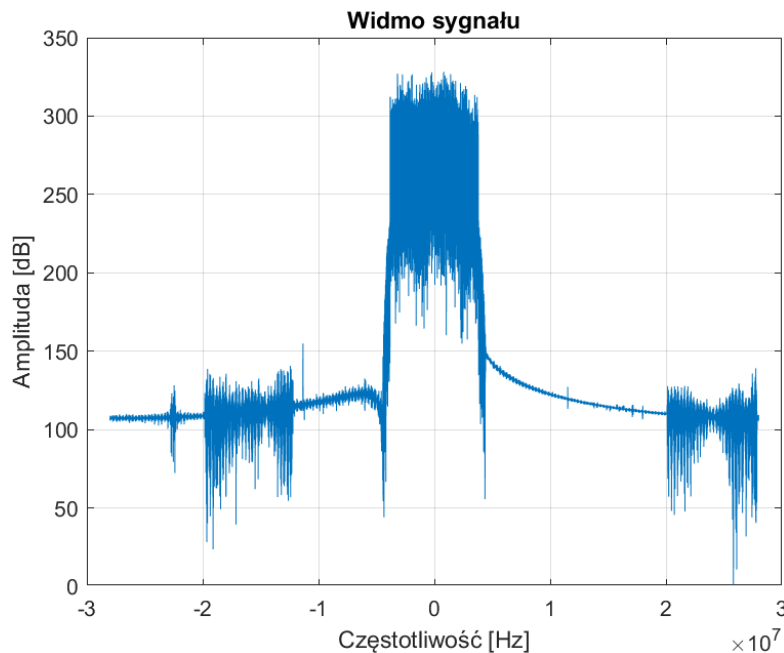
$$f_{norm} = \frac{f_g}{f_{Nyquist}} = \frac{f_g}{\frac{f_s}{2}} = \frac{4MHz}{\frac{56MHz}{2}} = 0.142857 \quad (3)$$

Na podstawie poprzedniego programu, napisano kolejny skrypt w MATLAB, który służy do wyfiltrowania danego kanału oraz wyświetlenia widma otrzymanego sygnału:

Listing 3: Skrypt MATLAB do zadania 5c.

```
1 load("iio_535M_56M_1848.mat");
2 I = cf_ad9361_lpc_voltage0;
3 Q = cf_ad9361_lpc_voltage1;
4
5 S = I + 1j * Q;
6
7 % Parametry
8 fs = 56e6; % Czestotliwosc probkowania
9 N = length(S); % Dlugosc sygnalu
10
11 % Parametry filtra
12 f_cutoff = 4e6; % Czestotliwosc odciecia
13 f_norm = f_cutoff / (fs/2); % Normalizacja
14
15 % Przemodulowanie sygnalu
16 Y = S' .* exp(-1j*2*pi*262204/N*(0:N-1));
17
18 % Filtrowanie
19 b = fir1(200, f_norm);
20 Y_filtered = filter(b, 1, Y);
21
22 % Obliczenie widma
23 Y_fft = fft(Y_filtered, N);
24 Y_fft_shifted = fftshift(Y_fft);
25 f = (-N/2:N/2-1) * (fs/N);
26
27 % Wyświetlenie widma
28 figure;
29 plot(f, 20*log(abs(Y_fft_shifted)));
30 xlabel('Czestotliwosc [Hz]');
31 ylabel('Amplituda');
32 title('Widmo sygnalu');
33 grid on;
```

Niegdyś w odbiornikach radiowych i telewizyjnych używano rezonansowych filtrów LC, które miały bardzo strome charakterystyki. Dlatego podobnie w tym przypadku zastosowano wysoki rząd filtru, aby uniknąć wpływu sąsiednich kanałów. Dla rzędu filtra równego 200 otrzymano na wyjściu sygnał o następującym widmie:



Rysunek 3: Widmo zespolone wyfiltrowanego kanału DVB-T

## 4 Zadanie 5d

Na poprzednim wykresie widma sygnału widać, że oprócz użytecznego sygnału pożądanego kanału znajduje się na nim spora ilość innych bezużytecznych sygnałów. Dalsze ich przetwarzanie byłoby bez sensu, dlatego dokonano decymacji, która obcina dalsze części widma, oczywiście symetrycznie względem 0Hz.

Do poprzedniego skryptu dodano kolejną modyfikację, uzyskując w ten sposób skrypt przedstawiony poniżej. Istotny jest w nim fakt, że dokonując decymacji, zmniejsza się praktyczną częstotliwość próbkowania. Informacja o niej jest wykorzystywana do wygenerowania odpowiednich podpisów osi OX wykresu, a więc w tym przypadku informacja ta byłaby przekłamana. FFT jest nadal wykonywana na  $N$  próbkach, a więc ilość próbek w jej wyniku się nie zmienia, natomiast zmienia się ich interpretacja częstotliwościowa, w więc dopasowano do niej nowe wartości osi OX. Dlatego w kodzie wylicza się wartość `fs_decimated`, na podstawie której jest generowany nowy wektor wartości częstotliwości używany przy generowaniu wykresu.

Listing 4: Skrypt MATLAB do zadania 5d.

```

1 load("iio_535M_56M_1848.mat");
2 I = cf_ad9361_lpc_voltage0;
3 Q = cf_ad9361_lpc_voltage1;
4
5 S = I + 1j * Q;
6
7 % Parametry
8 fs = 56e6; % Czestotliwosc probkowania
9 N = length(S); % Dlugosc sygnalu
10
11 % Parametry filtra
12 f_cutoff = 4e6; % Czestotliwosc odciecia

```

```

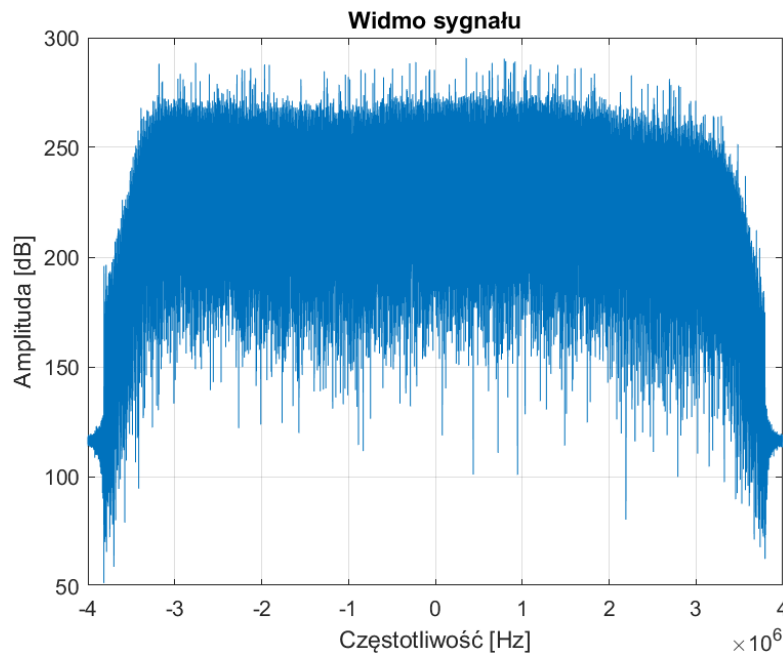
13 f_norm = f_cutoff / (fs/2); % Normalizacja
14
15 decimation = 7;
16
17 % Przemodulowanie sygnału
18 Y = S'.*exp(-1j*2*pi*262204/N*(0:N-1));
19
20 % Filtrowanie
21 b = fir1(200, f_norm);
22 Y_filtered = filter(b, 1, Y);
23
24 % Decymacja
25 Y_decimated = decimate(double(Y_filtered), decimation);
26 fs_decimated = fs/decimation; % nowa czestotliwosc probkowania
27
28 % Obliczenie widma
29 Y_fft = fft(Y_decimated, N);
30 Y_fft_shifted = fftshift(Y_fft);
31 f_decimated = (-N/2:N/2-1) * (fs_decimated / N);
32
33 % Wyświetlenie widma
34 figure;
35 plot(f_decimated, 20*log(abs(Y_fft_shifted)));
36 xlabel('Czestotliwosc [Hz]');
37 ylabel('Amplituda [dB]');
38 title('Widmo sygnału');
39 grid on;

```

Jako że częstotliwość próbkowania wynosi 56MHz, a pasmo kanału 8MHz, łatwo jest otrzymać odpowiedni współczynnik decymacji:

$$decimation = \frac{56MHz}{8MHz} = 7 \quad (4)$$

Dzięki decymacji uzyskano węższe widmo sygnału bez zbędnych składowych częstotliwościowych, które powodowałyby jedynie zwiększenie liczby obliczeń potrzebnych do analizy sygnału. Teraz widmo zawiera tylko sygnał użyteczny, który na poprzednim wykresie był widoczny jako szeroki pik wartości wokół częstotliwości 0Hz.



Rysunek 4: Widmo zespolone wyfiltrowanego kanału DVB-T po decymacji

## 5 Zadanie 5e

W zadaniu tym nie generowano już widma sygnału. Zamiast tego wygenerowany został inny bardzo użyteczny wykres - wykres autokorelacji, który jest bardzo pomocny przy próbie oszacowania długości ramki oraz prefiksu. Autokorelację dokonuje się z użyciem funkcji `xcorr()`, która wylicza korelację dwóch sygnałów: podstawowego oraz jego kopii, która zostaje za każdą iteracją przesunięta o jedną próbkę w zakresie  $-N + 1$  do  $N - 1$ . W ten sposób dla zerowego przesunięcia otrzymuje się maksymalny pik korelacji, co jest oczywiste, gdyż porównywany jest sygnał z samym sobą. Następnie wraz ze wzrostem przesunięcia wartość korelacji gwałtownie maleje aż do momentu, gdy prefiks ramki podstawowej zrówna się z kopią prefiksu w przesuniętej ramce. Wtedy spora część porównywanych próbek jest prawie taka sama, co powoduje kolejny pik wartości korelacji. W ten sposób można określić ilość próbek dzielących prefiks i jego kopię w jednej ramce.

Listing 5: Skrypt MATLAB do zadania 5e.

```

1 load("iio_535M_56M_1848.mat");
2 I = cf_ad9361_lpc_voltage0;
3 Q = cf_ad9361_lpc_voltage1;
4
5 S = I + 1j * Q;
6
7 % Parametry
8 fs = 56e6; % Czestotliwosc probkowania
9 N = length(S); % Dlugosc sygnalu
10
11 % Parametry filtra
12 f_cutoff = 4e6; % Czestotliwosc odciecia
13 f_norm = f_cutoff / (fs/2); % Normalizacja
14

```

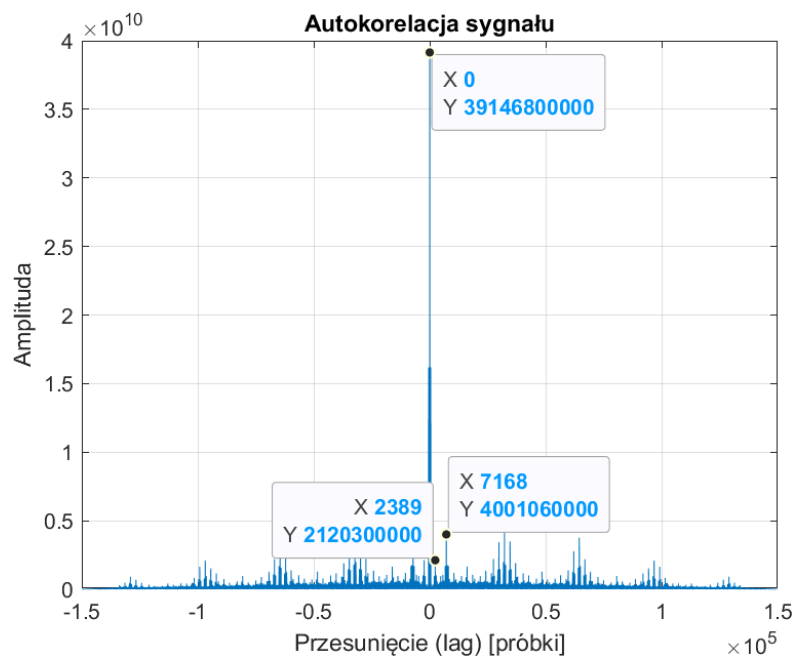


```

15 decimation = 7;
16
17 % Przemodulowanie sygnału
18 Y = S'.*exp(-1j*2*pi*262204/N*(0:N-1));
19
20 % Filtrowanie
21 b = fir1(200, f_norm);
22 Y_filtered = filter(b, 1, Y);
23
24 % Decymacja
25 Y_decimated = decimate(double(Y_filtered), decimation);
26
27 % Autokorelacja
28 Y_correlated = abs(xcorr(Y_decimated));
29
30 % Wyświetlenie autokorelacji z przesunięciem w próbkach
31 figure;
32 plot(-length(Y_decimated)+1:length(Y_decimated)-1, Y_correlated);
33 xlabel('Przesunięcie (lag) [próbki]');
34 ylabel('Amplituda');
35 title('Autokorelacja sygnału');
36 grid on;

```

Wykonanie powyższego kodu skutkowało narysowaniem wykresu korelacji przedstawionego poniżej:



Rysunek 5: Wykres autokorelacji sygnału wyfiltrowanego kanału DVB-T w funkcji przesunięcia o liczbę próbek

Naniesiono na niego trzy wskaźniki: dla pików głównego (dla którego przesunięcie jest równe 0) oraz dwóch najbliższych leżących pików. Dla pierwszego z pobocznych pików wartość przesunięcia wyniosła 2389, co nie zgadza się z dopuszczalnymi długościami ramek dla standardu DVB-T (które wynoszą 2048, 4096 lub 8192). Dla drugiego pików wartość przesunięcia wynosiła 7168, co tym razem zgadza się ze standardem, mimo że nie jest to przecież wartość 8192. Mniejsza wartość odczytanego przesunięcia wynika z faktu, że pik ten na

wykresie odpowiada takiemu przesunięciu, dla którego zwrótną się ze sobą prefiks i jego kopia. Można na tej podstawie obliczyć długość prefiksu:

$$\frac{8192 - 7168}{8192} = 0,125 \quad (5)$$

co zgadza się idealnie z dopuszczalną długością prefiksu wynoszącą 1/8 długości całej ramki.

## 6 Zadanie 5f

Zadanie te polega na znalezieniu indeksów próbek, w których rozpoczyna się kolejna ramka w sygnale DVB-T. Użyta została do tego funkcja `LM_estimator()` dostępna na stronie przedmiotu. Jako argumenty przyjmuje ona:

- **sygnał**, jako który należy podać odebrany sygnał poddany przemodulowaniu, filtracji i decymacji;
- **długość ramki**;
- **długość prefiksu**;
- **parametr  $\rho$**  pozwalający dopasować wykres do ilości szumów w sygnale;

Istotną sprawą jest to, że podanie wyznaczonej wcześniej długości ramki wynoszącej 8192 próbek nie przyniesie rezultatów. W tym przypadku wykres będzie przypominał zaszumiony sygnał. Wynika to z tego, że nadajnik próbkuje z częstotliwością  $64/7\text{MHz} \approx 9,142857\text{MHz}$ , podczas gdy odbiornik posiada częstotliwość próbkowania równą  $56\text{MHz}$ , co po dokonaniu 7-krotnej decymacji daje częstotliwość  $8\text{MHz}$ . Jest to przyczyną niezgodności parametrów ramki, dlatego jej długość należy przeskalować:

$$\text{frame\_length} \cdot \frac{\text{decimated\_frequency}}{\text{transmitter\_sampling\_frequency}} = 8192 \cdot \frac{8\text{MHz}}{9,142857\text{MHz}} \approx 7168 \quad (6)$$

Tak więc skrypt odpowiadający za rysowanie wykresu przedstawiającego momenty wykrycia początków ramek wygląda następująco:

Listing 6: Skrypt MATLAB do zadania 5f.

```

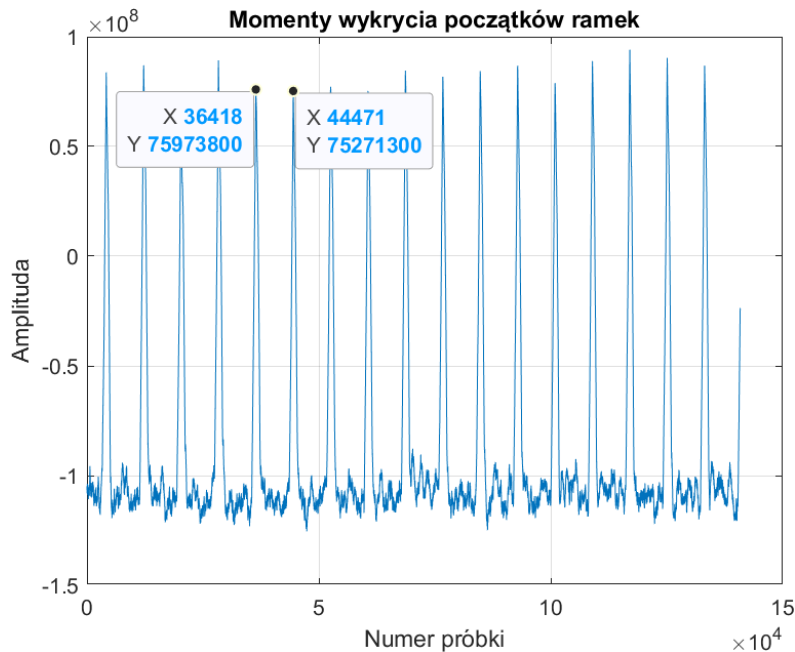
1 load("iio_535M_56M_1848.mat");
2 I = cf_ad9361_lpc_voltage0;
3 Q = cf_ad9361_lpc_voltage1;
4
5 S = I + 1j * Q;
6
7 % Parametry
8 fs = 56e6; % Czesotliwosc probkowania
9 N = length(S); % Dlugosc sygnalu
10
11 % Parametry filtra
12 f_cutoff = 4e6; % Czesotliwosc graniczna filtru
13 f_norm = f_cutoff / (fs/2); % Normalizacja
14
15 decimation = 7;
16
17 % Przemodulowanie sygnalu
18 Y = S' .* exp(-1j*2*pi*262204/N*(0:N-1));
19
20 % Filtrowanie
21 b = fir1(200, f_norm);

```

```

22 Y_filtered = filter(b, 1, Y);
23
24 % Decymacja
25 Y_decimated = decimate(double(Y_filtered), decimation);
26
27 % Autokorelacja
28 Y_correlated = abs(xcorr(Y_decimated));
29
30 % Wyznaczanie początków symboli
31 [theta, epsilon] = LM_estimator(Y_decimated, 7168, 0.125, 1);
32
33 % Wyświetlenie wykresu momentów detekcji początków ramek
34 figure;
35 plot(theta);
36 xlabel('Numer próbek');
37 ylabel('Amplituda');
38 title('Momenty wykrycia początków ramek');
39 grid on;

```



Rysunek 6: Wykres przedstawiający momenty wykrycia początków ramek sygnału DVB-T

Napisany skrypt wygenerował wykres zaprezentowany powyżej, na którym bardzo wyraźnie widać piki odpowiadające momentom wykrycia początków kolejnych ramek w odebranym sygnale DVB-T. Patrząc na zaznaczone wskaźniki, można obliczyć, że różnica ich wartości wynosi 8053 próbki, czyli trochę mniej niż wartość założona w standardzie. Jednak należy pamiętać, że dane do zadania pochodzą z rejestracji rzeczywistego sygnału. Co prawda nie wiadomo, z którego multipleksu były odbierane dane, a więc nie można sprawdzić, gdzie znajdowała się stacja nadawcza, ale można podejrzewać, że była ona na Pałacu Kultury i Nauki, a więc ze względu na niewielką odległość od nadajnika moc sygnału była zapewne bardzo dobra, co też widać na wykresach, gdzie piki sygnałów odróżniały się bardzo dobrze od tła szumów. Jednakże transmisja na obszarze miejskim jest szczególnie narażona na problemy związane z odbiciami fal od budynków itd., co zwane jest zjawiskiem wielodrogowości. Prawdopodobnie to może być powodem, dlaczego momenty wykrycia początków

kolejnych ramek nie są odległe od siebie o dokładnie 8192 próbki. Jednakże widać, że nie ma problemów z rozróżnieniem sygnałów z wyjścia detektora od szumów. Można powiedzieć, że algorytmy filtracji oraz detekcji działają bardzo dobrze. Mając dane pozyskane od nich, można już wycinać z sygnału kolejne ramki i poddawać je dekodowaniu.