

**Kauno technologijos universitetas**

Elektros ir elektronikos fakultetas

**FFT**

5 Laboratorinis darbas

|  |
| --- |
|  |
| **Žygimantas Marma, EMEI-2 gr.**  Studentas |
|  |
| **Doc. Šarūnas Kilius**  Dėstytojas |
|  |

**Kaunas, 2023**

**Kaunas, 2023**

Turinys

[1. Programos analizė 4](#_Toc151845491)

[1.1. Programos veikimo algoritmas 4](#_Toc151845492)

[2. Filtrų impulsinės charakteristikos 6](#_Toc151845493)

[3. Audio efektai 7](#_Toc151845494)

[4. Išvados 8](#_Toc151845495)

[5. Priedai 9](#_Toc151845496)

**Darbo tikslas**: Susipažinti su MEMS mikrofonais ir atlikti pateiktos programos analizę.

**Laboratorinio darbo uždaviniai:**

1. Sugeneruoti poli-harmoninį signalą. Modifikuoti FFT filtravimo funkciją taip, kad:  
a. Būtų filtruojamas vienas pasirinkto dažnio signalas.  
b. Būtų filtruojami visi skirtingų dažnių signalai, paliekant didžiausios amplitudės signalą

2. Išmatuoti FFT filtro greitaveiką esant skirtingiems signalo masyvo dydžiams bei skirtingiems kompiliatoriaus optimizavimo lygiams.  
3. Sugeneruoti kintančio dažnio (Chirp) signalą bei apskaičiuoti jo dažnių spektrą.

# Poliharmoniins signalas

## Programos veikimo algoritmas

Signalo dedamosios yra:

//Test signal 1000 Hz + 3000 Hz

//InputArray[index] = ( 20000\*sin((FREQUENCY/SAMPLING\_FREQ)\*2\*PI\*index) + 10000\*sin(3\*(FREQUENCY/SAMPLING\_FREQ)\*2\*PI\*index) );

//Cos + PI/6 shift

InputArray[index] = ( 20000\***cos**((FREQUENCY/SAMPLING\_FREQ)\*2\*PI\*index + PI/6) );

1k + 3k + 5k

Ikelti laiko srityje

Ikelti dažnius

# Greitaveikos tyrimas

Kaip priklauso nuo masyvo dydžio ir optimizacijos laipsnio

# Kintančio dažnio (*Chirp*) signal analizė

Funkcija sugeneruoti Chrip:

Kodas

Gautas laiko asyje

Gautas daznis

# Išvados

1. Laboratorinio darbo metu buvo sukurti ir eksportuoti į mikrovaldiklį žemų dažnių, aukštų dažnių ir juostinis filtrai.
2. Tiek MATLAB simuliacijose, tiek realiame mikrovaldiklyje skurti filtrai veikė tinkamai ir pašalino nepageidaujamų dažnių dedamąsias.
3. Iš testavus slenkančio vidurkio filtrą buvo įsitikinta, jog šio tipo filtrui nereikalinga didelė eilė, 5 eilės filtras sugebėjo pašalinti triukšmą iš signalo.
4. Palyginus paprastą vidurkio funkciją ir FIR filtrų specialias funkcijas buvo pastebėta, jog su paprasta funkcija programa veikė 26,7% greičiau.

# Priedai

**7 lentelė.** MATLAB programa sukurtų filtrų koeficientų konvertavimui į C kodą

|  |
| --- |
| % FIR testas  % Filtro koeficientai saugomi faile    clear;  % HD - this is a generated MATLAB filter function  % Hd = FIR\_filter\_30ord; % example  Hd = lowpassFinal;  %  ylabel('Amplitude');    % Display only positive frequencies  axis([0 Fs/2 0 max(P\_y)]);  end |

**8 lentelė** pagrindinis laboratorinio darbo programinis kodas

|  |
| --- |
| #define SAMPLING\_FREQ 48000.0   //BSP\_audio\_out sampling frequency  //--- FIR related PD ---  #define TEST\_LENGTH\_SAMPLES 1024  //Test signal length                    //Turi dalintis is BLOCK\_SIZE! |

**9 lentelė.** Sugeneruoti žemų dažnių filtro koeficientai

|  |
| --- |
| #define NUM\_SECTIONS 5  float b[NUM\_SECTIONS][3] = {  {1.48120186E-01, 2.96240372E-01, 1.48120186E-01},  {1.23633875E-01, 2.47267751E-01, 1.23633875E-01},  {1.08391408E-01, 2.16782815E-01, 1.08391408E-01},  {9.94805969E-02, 1.98961194E-01, 9.94805969E-02},  {9.53591618E-02, 1.90718324E-01, 9.53591618E-02} };  float a[NUM\_SECTIONS][3] = {  {1.00000000E+00, -1.19927188E+00, 7.91752620E-01},  {1.00000000E+00, -1.00101569E+00, 4.95551188E-01},  {1.00000000E+00, -8.77603320E-01, 3.11168950E-01},  {1.00000000E+00, -8.05455930E-01, 2.03378318E-01},  {1.00000000E+00, -7.72086263E-01, 1.53522910E-01} }; |

**10 lentelė.** Sugeneruoti aukštų dažnių filtro koeficientai

|  |
| --- |
| #define NUM\_SECTIONS 4  float b[NUM\_SECTIONS][3] = {  {1.66790712E-01, -3.33581424E-01, 1.66790712E-01},  {1.33825671E-01, -2.67651342E-01, 1.33825671E-01},  {1.16241827E-01, -2.32483653E-01, 1.16241827E-01},  {1.08524666E-01, -2.17049332E-01, 1.08524666E-01} };  float a[NUM\_SECTIONS][3] = {  {1.00000000E+00, 1.06621340E+00, 7.33376244E-01},  {1.00000000E+00, 8.55483628E-01, 3.90786313E-01},  {1.00000000E+00, 7.43078504E-01, 2.08045810E-01},  {1.00000000E+00, 6.93746382E-01, 1.27845047E-01} }; |

**11 lentelė.** Sugeneruoti juostinio filtro koeficientai

|  |
| --- |
| #define NUM\_SECTIONS 5  float b[NUM\_SECTIONS][3] = {  {2.59047000E-01, 0.00000000E+00, -2.59047000E-01},  {2.59047000E-01, 0.00000000E+00, -2.59047000E-01},  {2.33543913E-01, 0.00000000E+00, -2.33543913E-01},  {2.33543913E-01, 0.00000000E+00, -2.33543913E-01},  {2.25616630E-01, 0.00000000E+00, -2.25616630E-01} };  float a[NUM\_SECTIONS][3] = {  {1.00000000E+00, 7.66866044E-01, 8.51990426E-01},  {1.00000000E+00, -2.12333336E-01, 8.39565783E-01},  {1.00000000E+00, 5.35714765E-01, 6.36464291E-01},  {1.00000000E+00, -2.47588879E-02, 6.19336420E-01},  {1.00000000E+00, 2.49164944E-01, 5.48766740E-01} }; |