

**Kauno technologijos universitetas**

Elektros ir elektronikos fakultetas

**Vokoderio realizavimas Rapsberry Pi bei duomenų perdavimas panaudojant TCP**

Kursinis projektas

|  |
| --- |
|  |
| **Žygimantas Marma, EMEI-2 gr.**  Studentas |
|  |
| **Doc. Mindaugas Knyva**  Dėstytojas |
|  |

**Kaunas, 2023**

**Kaunas, 2023**

Turinys

[Įvadas 3](#_Toc136428870)

[1. Teorija: 4](#_Toc136428871)

[1.1. Vokoderis 4](#_Toc136428872)

[1.2. CODEC2 4](#_Toc136428873)

[1.3. TCP/IP 5](#_Toc136428874)

[2. Darbo aplinka 6](#_Toc136428875)

[3. Raspberry Pi programa 8](#_Toc136428876)

[3.1. Python programa siunčianti duomenis TCP protokolu 8](#_Toc136428877)

[3.2. Sukurtas Shell script 10](#_Toc136428878)

[4. Duomenų apdorojimas Linux serveryje 12](#_Toc136428879)

[4.1. Python programa priimanti ir sauganti gaunamus duomenis 12](#_Toc136428880)

[4.2. Sukurtas Shell script 13](#_Toc136428881)

[5. Atlikto tyrimo rezultatai 15](#_Toc136428882)

[6. Išvados 16](#_Toc136428883)

[Literatūros sąrašas 17](#_Toc136428884)

Įvadas

Šiais laikais yra tvarkingą ir klaidas tikrinantį duomenų paketų persiuntimą naudojant IP tinklus.

GAL BSK TEORIJOS

**Tikslas**: Šio darbo tikslas yra įgyvendinti vokoderį Raspberry Pi prietaise ir persiųsti užkoduotą įrašą naudojant TCP protokolą.

**Laaboratorinio darbo uždaviniai:**

1. Apžvelgti kas yra vokoderiai ir TCP protokolas.
2. Paruošti darbo aplinką skirtą siųsti ir gauti žinutes per TCP protokolą.
3. Sukurti Raspberry Pi programą, kuri su vokoderiu koduotų ir siųstų žinutes per TCP.
4. Sukurti programą, kuri priimtų ir dekoduotų gautus failus.
5. Atlikti gautų rezultatų tyrimą.

# Dažnių persidengimas (angl. aliasing)

Patobulinkite pateiktame pavyzdyje esančią sinusinio signalo skaičiavimui skirtą funkciją, įtraukdami kintamajį arba konstantą dažniui nurodyti.

## Praktinė dalis

Matematiškai sinuso banga yra aprašoma šia forma:

,

kur A yra amplitudė, nustatanti bangos maksimalią reikšmę.

f yra dažnis, nustatantis, kiek ciklų atsiranda per vienetą laiko.

t yra laiko kintamasis.

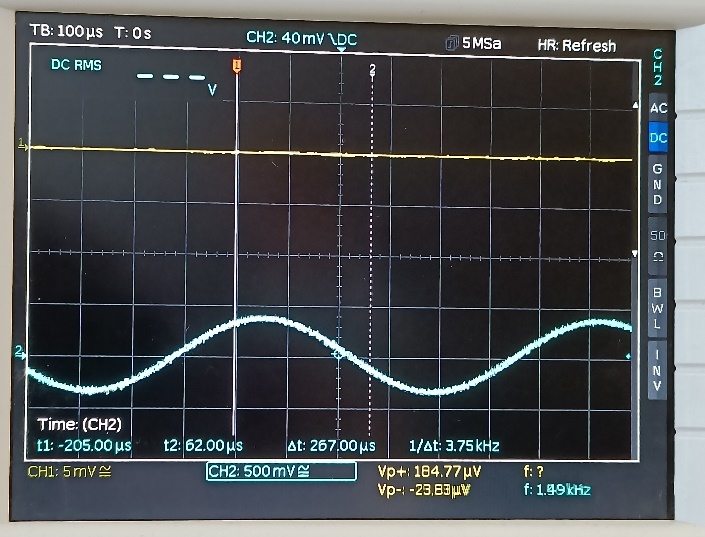
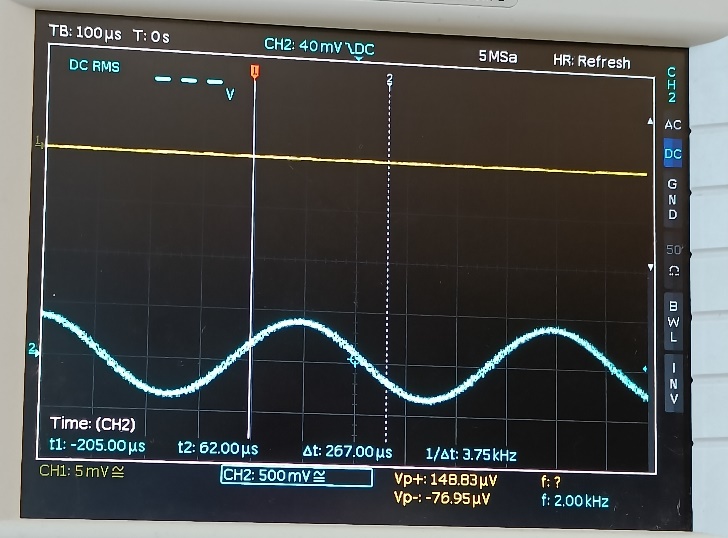
ϕ (fi) yra fazinė kampas, kuris nurodo horizontalų poslinkį bangos laiko ašyje. Ši funkcija realizuota 1 lentelėje pateiktu programiniu kodu.

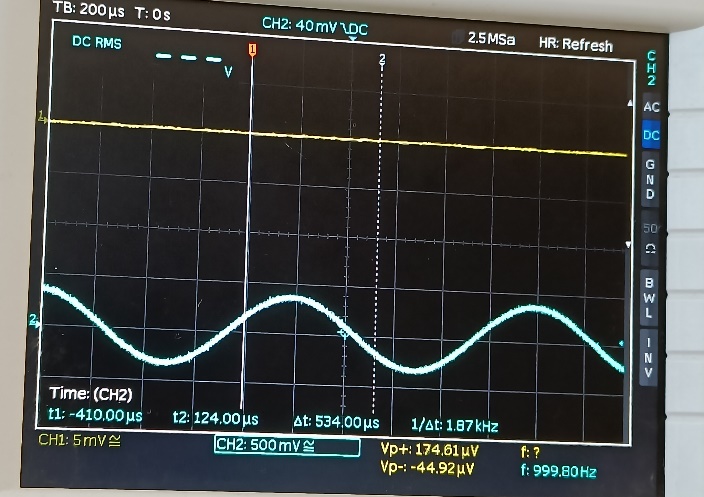
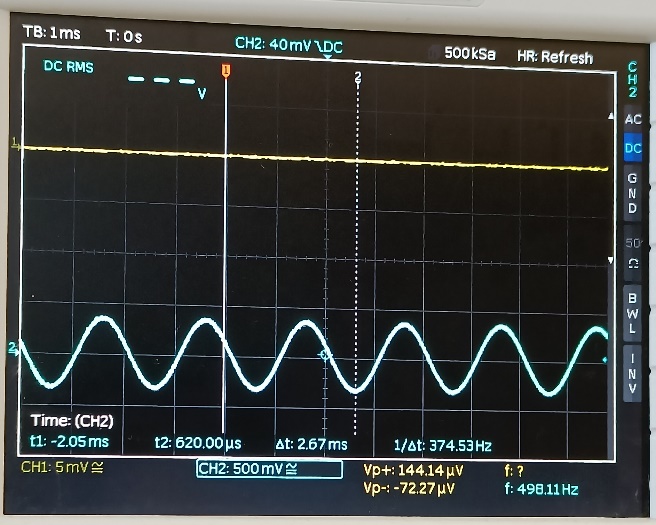
1 lentelė. Programinis kodas generuoti sinusinį signalą

|  |
| --- |
| #define FREQUENCY 1000.0 //Generated signal frequency  #define SAMPLING\_FREQ 8000.0  ...  /\*\*\* Test Signal generation \*\*\*/  for(int index = 0; index < SAMPLES\_QTY; index++)  {  double phase = 2 \* PI \* FREQUENCY \* index / (2 \* SAMPLING\_FREQ);  Left\_out\_buffer[index] = (int16\_t)(30000 \* sin(phase));  Right\_out\_buffer[index] = (int16\_t)(30000 \* sin(2 \* phase));  }  /\*\*\* Fill Output Buffer \*\*\*/  for(int i = 0; i < SAMPLES\_QTY; i++)  {       OutputBuffer[i<<1] = Left\_out\_buffer[i];       OutputBuffer[(i<<1)+1] = Right\_out\_buffer[i];  } |

2 lentelė. Dažnių persidengimo tyrimo rezultatai

|  |  |
| --- | --- |
| **Nurodytas signalo dažnis (Hz)** | **Analoginio išėjimo signalo dažnis (Hz)** |
| 1500 | 1500 |
| 2000 | 2000 |
| 3500 | 3500 |
| 4500 | 3500 |
| 7000 | 1000 |
| 8500 | 500 |

1 pav. Sugeneruoti sinusiniai signalai, viršuje kairėje nustačius 1,5k Hz dažnį, viršuje dešinėje nustačius 2 kHz, apačioje kairėje nustačius 7kHz ir apačioje dešinėje nustačius 8 kHz dažnį.

## Rezultatų analizė

Kaip galima matyti ši problema susijusi su dažnių persidengimu, kuris vadinamas aliasingu. Persidengimas atsiranda, kai diskretizuojant analoginį signalą, t.y., paverčiant jį į diskretųjį laiką naudodami diskretaus signalo apribojimus. Pagrindinė aliasingo priežastis yra nepakankamas diskretizavimo dažnis. **Diskretizuojant signalą reikia užtikrinti, kad diskretizavimo dažnis būtų didesnis nei dvigubai didesnis už didžiausią signalo dažnį, kurį norite išsaugoti. Tai vadinama Nyquisto teorema.** Kitu atveju, signalas gali būti neteisingai atkuriamas dėl dažnių persidengimo.

Pagal anksčiau pateiktą kodą ir tyrimo rezultatus, galima pastebėti, kad dažnių persidengimas atsiranda, kai norima sugeneruoti signalą, kurio dažnis yra didesnis nei pusė diskretinio laiko signalo dažnio (SAMPLING\_FREQ/2).

Šiame tyrime dalis analizuotų signalų, turi didesnius dažnius nei pusė diskretinio laiko signalo dažnio: **4500** Hz ir **7000** Hz dažniai turi viršytą Nyquisto ribą ir yra persidengiantys. Signalas su **8500** Hz dažniu yra virš Nyquisto ribos, todėl jis taip pat persidengia.

# Stačiakampės formos signalo tyrimas

Sugeneruokite stačiakampės formos 1 kHz periodinį signalą su šiomis signalo amplitudės reikšmėmis: {20000, 20000, 20000, 20000, -20000, -20000, -20000, -20000}.

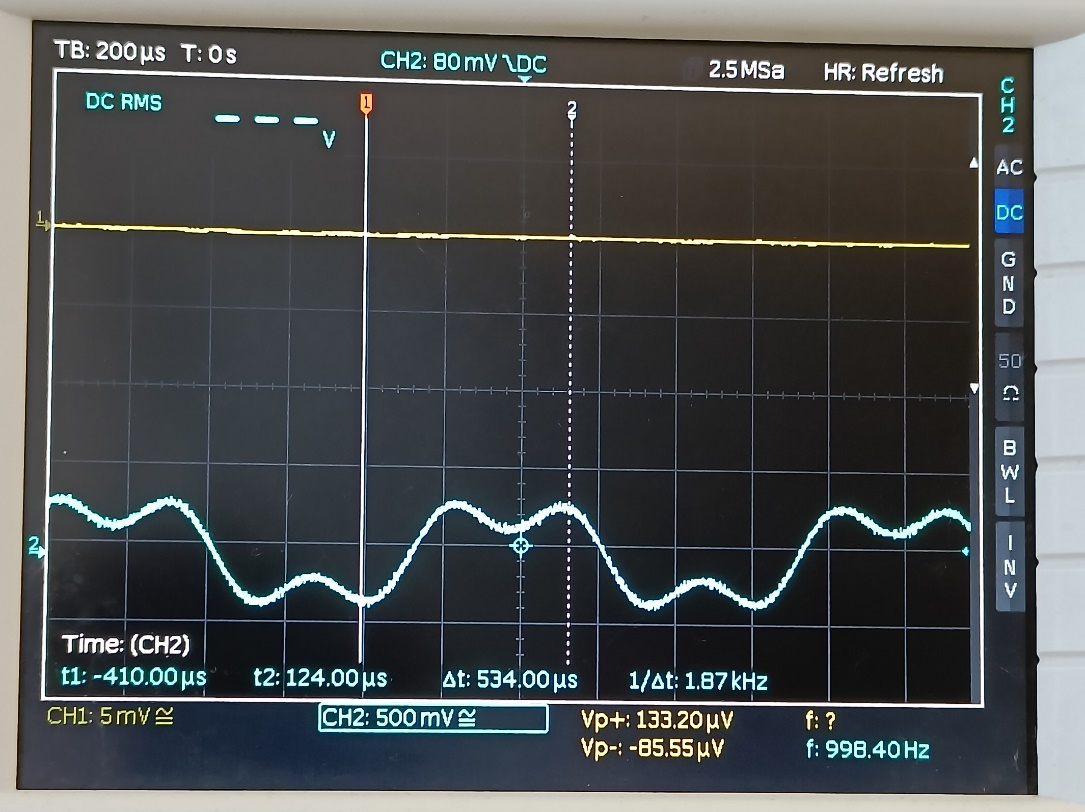
## Praktinė dalis

Stačiakampio signalo generavimas yra pateikiamas 3 lentelėje.

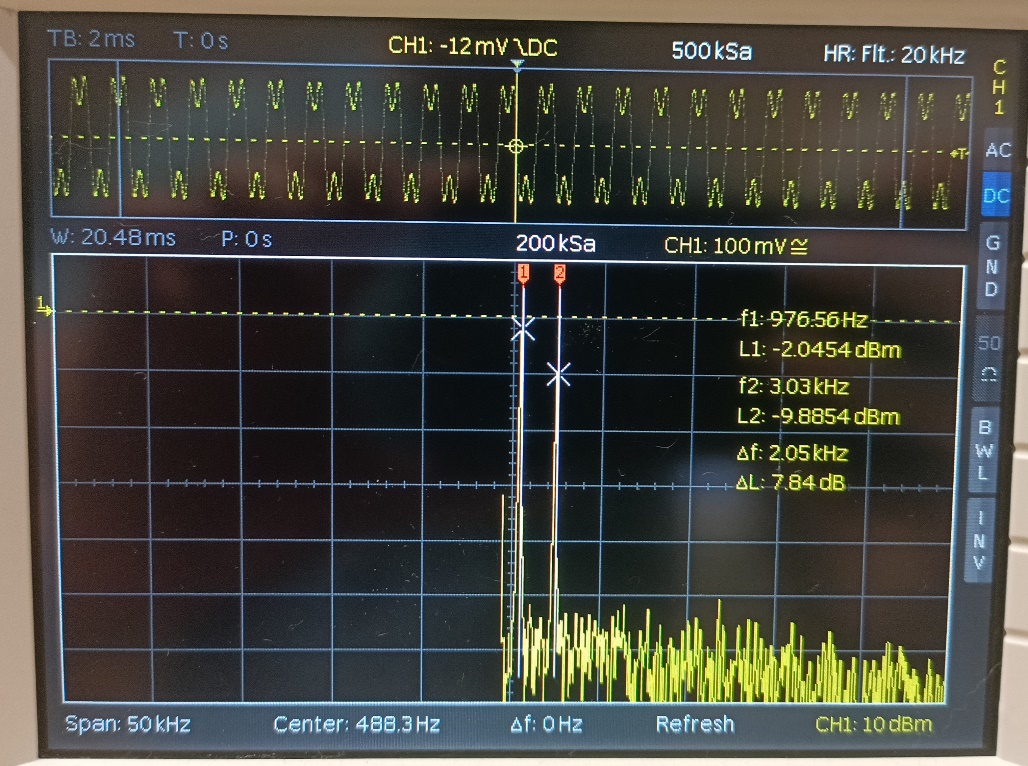
3 lentelė. Programinis kodas generuoti stačiakampį signalą

|  |
| --- |
| #define SAMPLING\_FREQ 8000.0 //BSP\_audio\_out sampling frequency  #define SAMPLES\_QTY 8 //Signal buffer for one channel  int16\_t Square\_wave[SAMPLES\_QTY];  ...  for (int index = 0; index < SAMPLES\_QTY / 2; index++)  {      Square\_wave[index] = 20000;      Square\_wave[index + SAMPLES\_QTY / 2] = -20000;  }  /\*\*\* Test Square Wave generation \*\*\*/  for (int index = 0; index < SAMPLES\_QTY; index++)  {     OutputBuffer[index << 1] = Square\_wave[index];     OutputBuffer[(index << 1) + 1] = Square\_wave[index];  } |

Sugeneruotas signalas matomas 2 paveikslėlyje. Nors signalas yra panašus į stačiakampį, jame vis dar galima matyti bangavimų. Tai lemia per mažas signalo masyvo reikšmių kiekis – šiuo atveju tik 8 elementai.



2 pav. Stačiakampis signalas, kurio dažnis lygus 1kHz

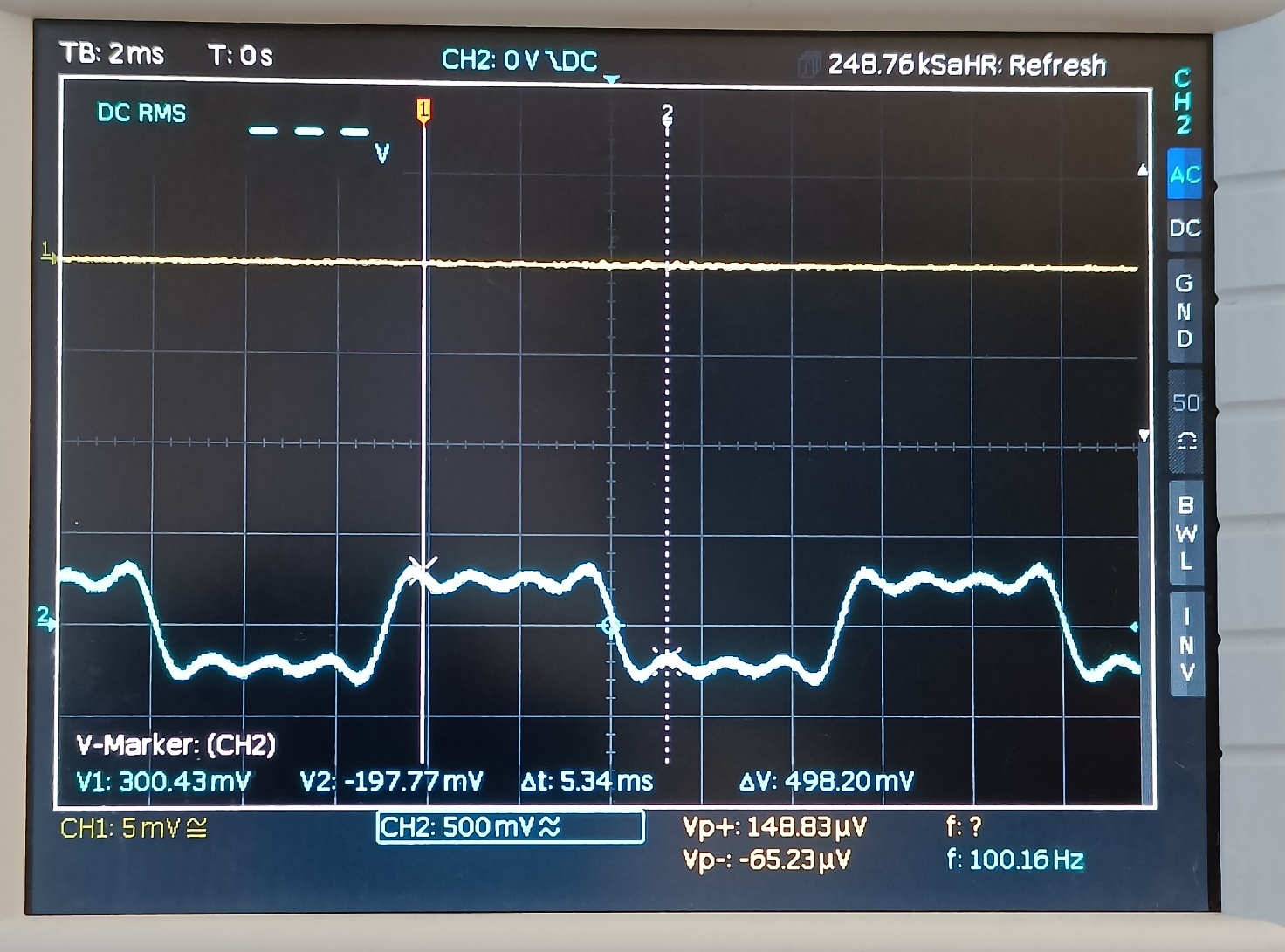


3 pav. Stačiakampio 1 kHz dažnio signalo spektras

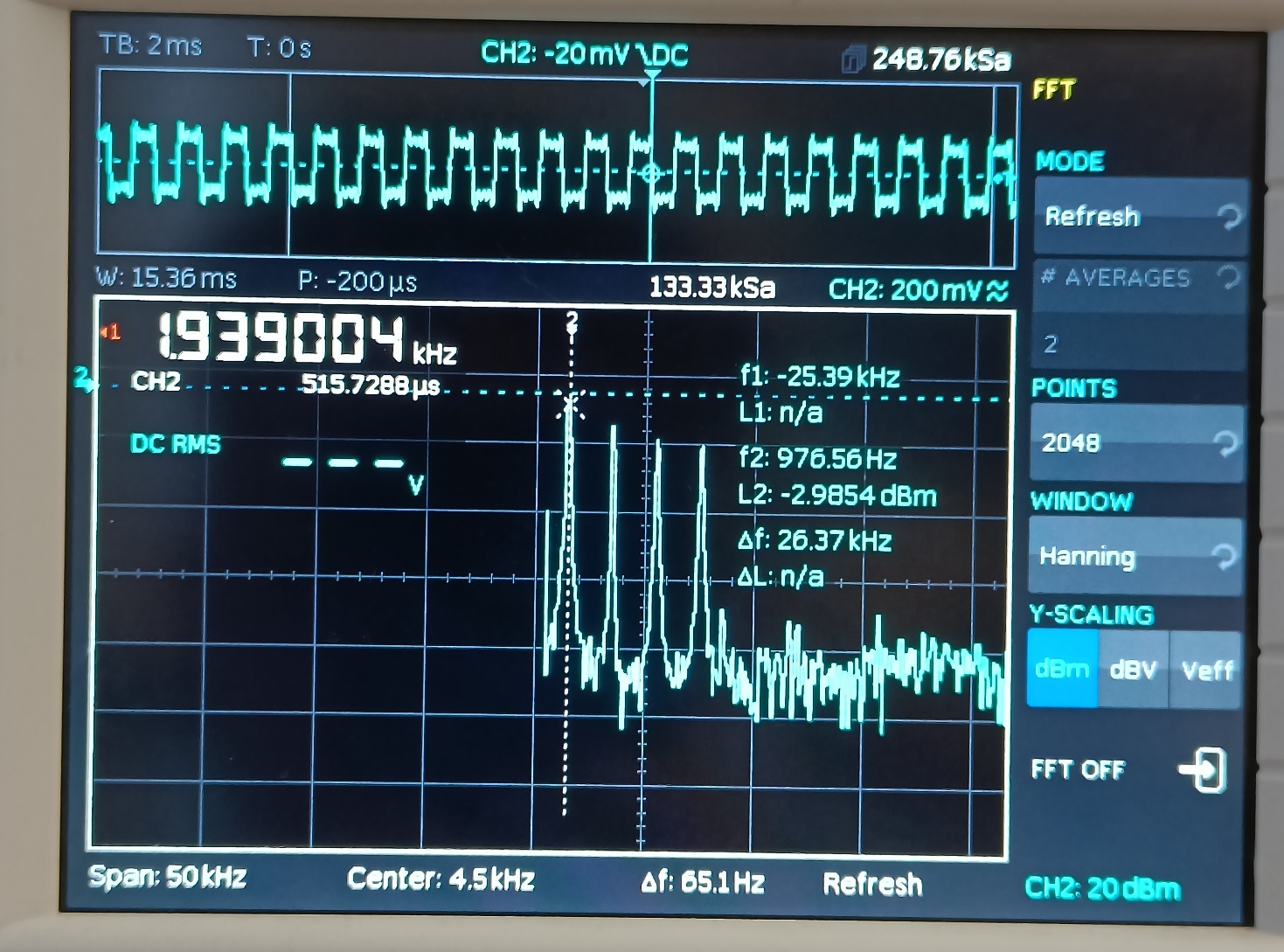
Iš signalo spektro (3 pav.) galima matyti, jog svarbiausios yra harmonikos nelyginės harmonikos pasireiškiančios ties 1kHz, 3kHz, 5kHz.

## Tyrimas esant 16 kHz diskretizavimo dažniui ir 16 masyvo reikšmių

Atlikus šį tyrimą buvo pakeistos diskretizavimo dažnio reikšmę į 16‘000 ir reikšmių keikis į 16. Rezultatus galima matyti 4 ir 5 paveikslėliuose. Galima matyti, jog signalas yra labiau panašus į stačiakampį.



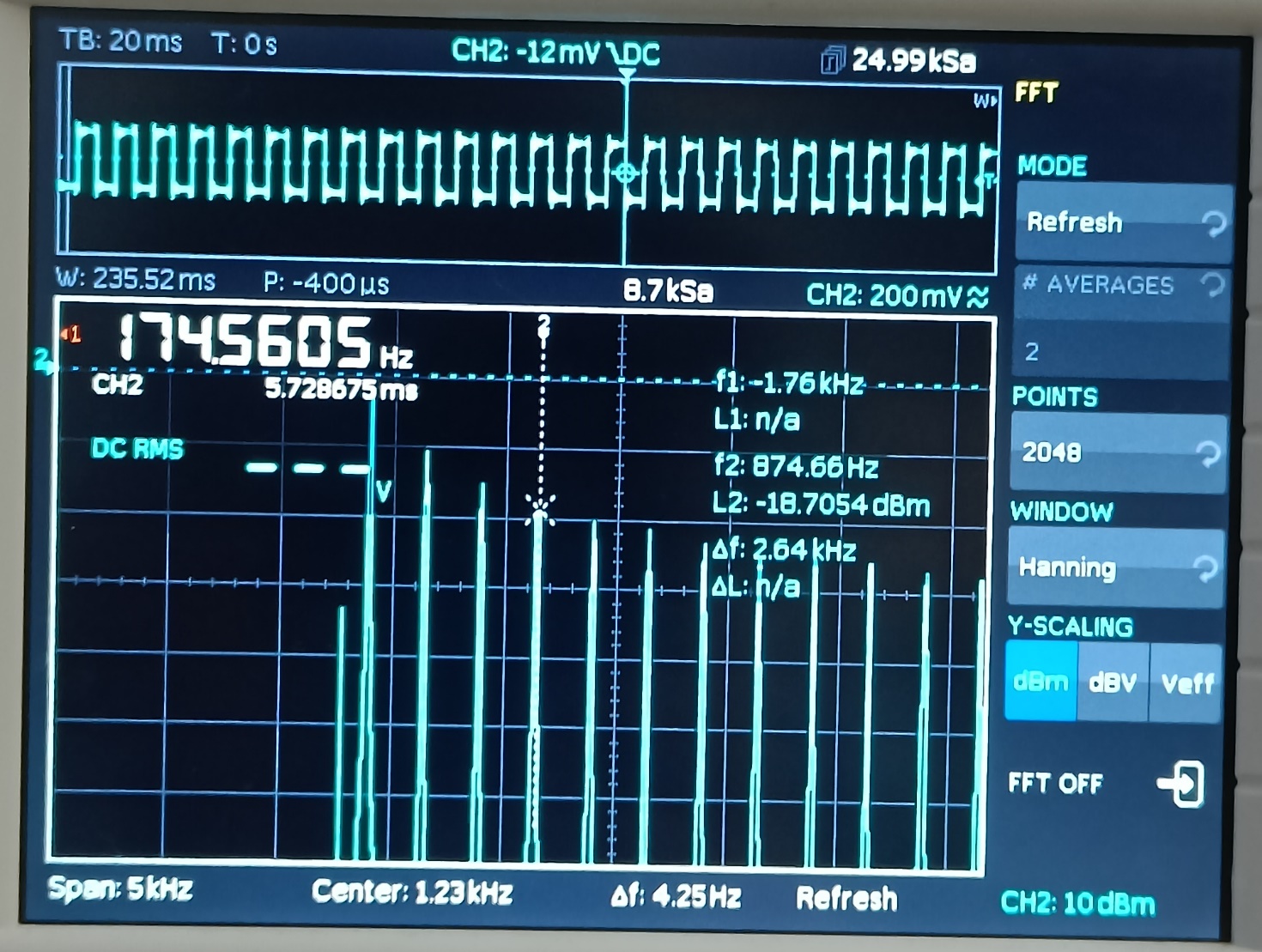
pav. pav.

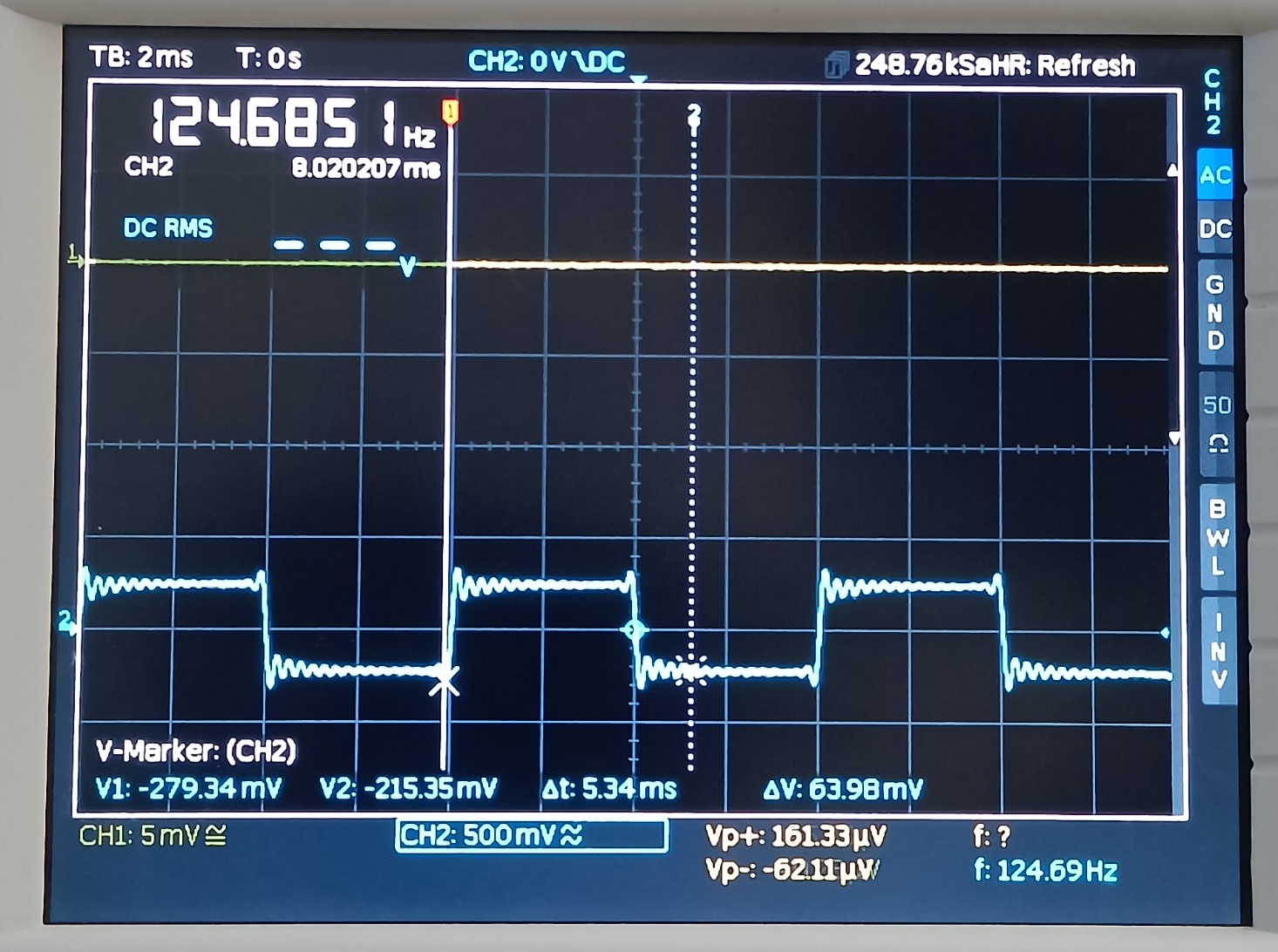


Palyginus matomos tos pacios harmonikos tik antru atveju mažiau triukšmo

## Filtro atsako tyrimas

Sugeneruokite stačiakampės formos signalą, diskretizavimo dažnis 8 kHz, reikšmių kiekis – 64, pirmosios pusės reikšmių amplitudė lygi 20000, likusios pusės amplitudė lygi -20000





# Išvados

1. Raspberry
2. Atlikus r
3. Kadangi
4. Vokoderio.

# Priedai

Ikelti visa main.c koda

Literatūros sąrašas

1. M. R. Schroeder, "Vocoders: Analysis and synthesis of speech," in Proceedings of the IEEE, vol. 54, no. 5, pp. 720-734, May 1966, doi: 10.1109/PROC.1966.4841.
2. Vikipedija apie vokoderius, prieiga per internetą: https://en.wikipedia.org/wiki/Vocoder.
3. M. Knyva, „Skaitmeninis signalų apdorojimas įterptinėse bevielio ryšio sistemose“, T121M161  
   mokomoji medžiaga.