

Vilniaus universitetas
Fizikos fakultetas
Radiofizikos katedra

Vaidotas Kumža

**Terahercinės spinduliuotės vaizdinimo sistema su rezonansiniu mažos
triukšmų galios detektoriumi**

Pagrindinių studijų baigiamasis darbas

Telekomunikacijų fizikos ir elektronikos
studijų programa

Studentas
Darbo vadovas
Recenzentas
Katedros vedėjas

Vaidotas Kumža
dokt. Justinas Zdanevičius
doc. Vytautas Jonkus
prof. Jūras Banys

Vilnius 2016

Turinys

Įvadas	4
1. Terahercinė spinduliuotė	5
1.1 Terahercinės spinduliuotės charakteristikos	5
1.2 THz vaizdinimas	6
1.2 THz vaizdinimo pritaikymo būdai	6
2. Lauko tranzistoriai kaip terahercinės spinduliuotės detektoriai	9
3. Žingsniniai varikliai	11
3.1. Žingsninių variklių tipai.....	11
3.1.1 Reaktyvieji žingsniniai varikliai.....	11
3.1.2 Žingsniniai varikliai su nuolatiniu magnetu.....	12
3.1.3 Hibridiniai žingsniniai varikliai	13
3.2 Žingsninių variklių charakteristikos	15
3.2.1 Sukimo momentas.....	15
3.2.2 Dvipoliai ir vienpoliai varikliai	15
3.2.3 Vieno žingsnio atsakas	15
3.4 Žingsninių variklių panaudojimas.....	16
4. Žingsninio variklio sistema	17
4.1 Mikrovaldiklis.....	17
4.2 Žingsninis variklis su MT-55 transliaciniu staleliu	19
4.3 A3967 tvarkyklė	20
4.4 Serijinė sąsaja	26
5. Programavimo eiga	27
6. Matavimo metodika	29
7. Rezultatai ir jų aptarimas	32
Išvados	37

Santrauka	38
Summary	39
Literatūra	40
Priedas	42
Mikrovaldiklio programos kodas	42
Labview programos blokinė diagrama.....	48

Ivadas

Aplink mus yra pilna įvairiausių bangų. Kai keičiame televizijos kanalus nuotoliniu pulteliu, kalbamės telefonu ar šildomės maistą mikrobangų krosnelėje – mes naudojame bangas. Priklausomai nuo jų ilgio, panaudojimo sritis gali skirtis, todėl nenuostabu, kad stengiamasi iširti ir pritaikyti žmogaus naudai visus bangų tipus. Taigi didelio dėmesio susilaukia ir terahercinė spinduliuotė. Terahercinė spinduliuotė yra elektromagnetinė spinduliuotė, kurios dažnių spektras yra tarp mikrobangų ir infraraudonųjų bangų. Nors ir natūralios prigimties terahercinė spinduliuotės yra pilna aplink mus kiekvieną dieną, tačiau ši elektromagnetinio spektro dalis lieka mažiausiai ištyrinėta, vien dėl techninių sunkumų gaminant efektyvius ir kompaktiškus terahercinės spinduliuotės šaltinius bei detektorius. Tačiau per pastaruosius du dešimtmečius intensyvių tyrimų dėka pavyko sukurti terahercinės spinduliuotės detektavimo ir atvaizdavimo sistemas, kurios jau komerciškai prieinamos.

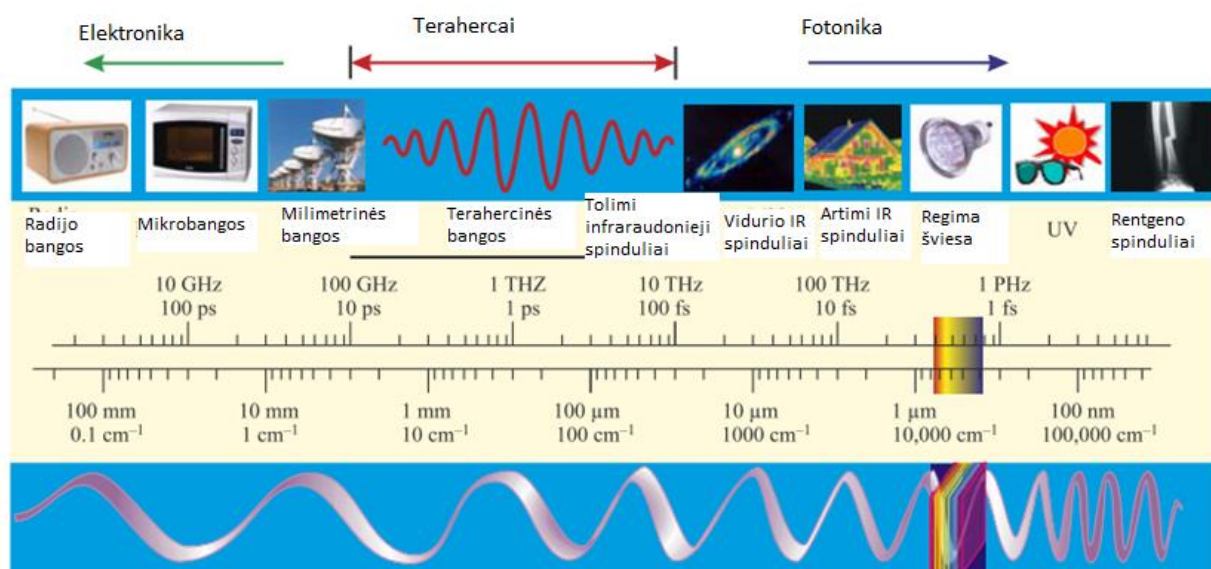
Žmonių poreikiai vis didėja, taigi tyrinėti ir vis tobulinti šią sritį yra itin svarbu. Ypač kai atsiranda vis daugiau terahercinės spinduliuotės pritaikymų, tokių kaip: itin greitas bevielis ryšys, paslėptų objektų ir defektų radimas, 2D ar 3D vaizdinimas.

Kadangi terahercinė spinduliuotė nėra jonizuojanti, jos vaizdinimo pritaikymo galimybės yra plačios. Jos gali būti panaudotos dielektrinių ir keramikinių medžiagų tankio ir storio nustatymui, dujų nuotėkio, chemikalų ir branduolinių medžiagų radimui, paslėptų sprogmenų ar ginklų oro uostuose ieškojimui. Taipogi, terahercinės spinduliuotės sugertis sukuria unikalų spektrinį modelį, iš kurio galime identifikuoti medžiagą. Taigi, turėti tokią vaizdinimo sistemą būtų labai pravartu.

Šio darbo tikslas yra susikurti terahercinės spinduliuotės vaizdinimo sistemą panaudojant žingsninius variklius ir silicio KMOP detektorius bei atvaizduoti kelis pasirinktus objektus.

1. Terahercinė spinduliuotė

Terahercinė spinduliuotė – dar žinoma kaip submilimetrinė spinduliuotė ar terahercinės bangos – tai elektromagnetinės bangos, kurių dažnių juosta driekiasi nuo 100GHz iki 10THz, o bangų ilgiai nuo 3mm iki 30 μ m. Šių bangų dažnių spektras yra tarp mikrobangų ir infraraudonųjų spindulių bei pasižymi unikalėmis savybėmis.



1pav. Terahercinių bangų juostos plotis [12]

1.1 Terahercinės spinduliuotės charakteristikos

Terahercinė spinduliuotė yra charakterizuojama šiais unikaliais bruožais:

- THz gali prasiskverbti pro daugelį dielektrinių medžiagų, tokiu kaip audiniai, popierius, plastikas, oda ir medis.
- THz nėra jonizuojanti ir turi minimalių poveikį žmogaus kūnui.
- Vanduo yra vienas stipriausių THz slopintuvų.
- Metalai nepaprastai gerai atspindi THz.

Taip pat, dauguma organinių molekulių turi absorbcijos ir dispersijos charakteristikas terahercinių bangų juostoje, todėl terahercinė medžiagų spektroskopija turi didelį kiekį fizinės ir cheminės informacijos. Šios charakteristikos yra unikalios, todėl tai kaip medžiagų pirštų anspaudai.

1.2 THz vaizdinimas

Pastaruosius dešimtmečius terahercinės spinduliuotės generavimo ir detektavimo technologijos nuolat vystėsi. Pagal naudojamą šaltinio tipą, terahercinės sistemos gali būti suskirstytos į vakuuminės elektronikos, kieto kūno elektronikos, lazerių ir mechaninio sužadavimo [5]. Terahercų šaltinio pasirinkimą nulemia naudojama vaizdinimo technika. Žiūrint iš praktinės pusės, labai svarbu žinoti, kad artėjant iki vidurinės terahercų srities (1 THz) išėjimo galia daugelyje įrenginių smarkiai nukrenta. Šiuo metu daugelis terahercinių bangų generavimo technikų jau yra komerciškai pasiekiami, įskaitant tokius kaip: optiškai pulsuojantis lazeris, kvantinis kaskadinis lazeris, atbulinės bangos osciliatoriai.

Visos terahercinės spinduliuotės detektavimo sistemos gali būti suskirstytos į dvi grupes: plačiajuostes ir siauros juostos.

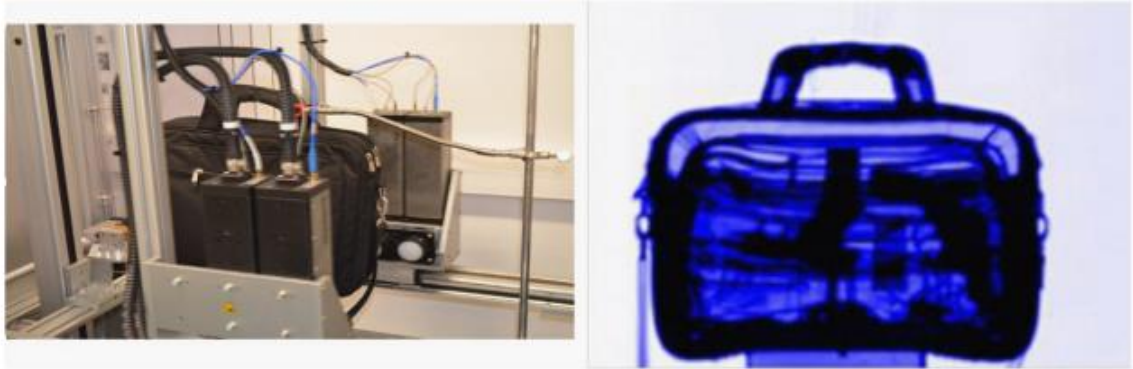
1.2 THz vaizdinimo pritaikymo būdai

Dėl anksčiau išvardintų terahercinių bangų charakteristikų, atsiranda daugybė THz vaizdinimo pritaikymo sričių. Kadangi terahercinė spinduliuotė nėra jonizuojanti, ji yra mažiau kenksmingesnė gyvam audiniui nei rentgeno spinduliai. Tai suteikia daug įdomių pritaikymo galimybių medicinos srityje, tokių kaip vežinių ląstelių diagnozavimas [6]. Taip pat buvo įrodyta, kad THz vaizdinimas gali būti panaudotas kaip alternatyva kitoms diagnostikos technologijoms, pavyzdžiui, pavaizduojant dantų struktūrą ir identifikuojant dantų ėduonį [7].

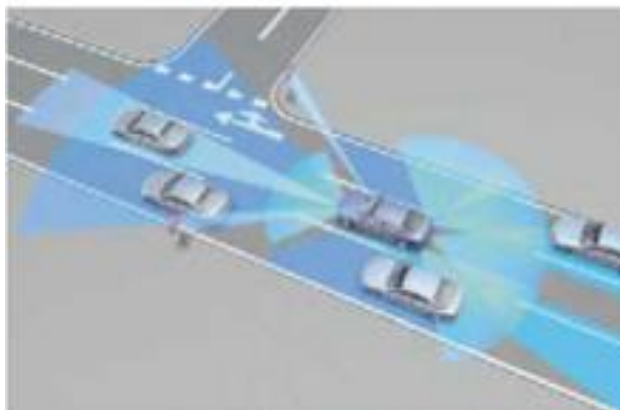
Didelis vandens absorbavimas leidžia terahercinę spinduliuotę panaudoti ieškant vandens augaliniuose audiniuose [8]. Dauguma cheminių medžiagų charakteringus spektrinius pirštų anspaudus terahercinių bangų srityje, taigi tai gali būti panaudojama farmacijos industrijoje, surandant ir identifikuojant nepageidaujamus objektus vaistų gamyboje [9].

Gynybos ir apsaugos sferoje, terahercų technologija taip pat intensyviai tyrinėjama. Terahercų spektroskopija gali būti panaudojama ieškant sprogstamųjų medžiagų, taip ir paslėptų ginklų.

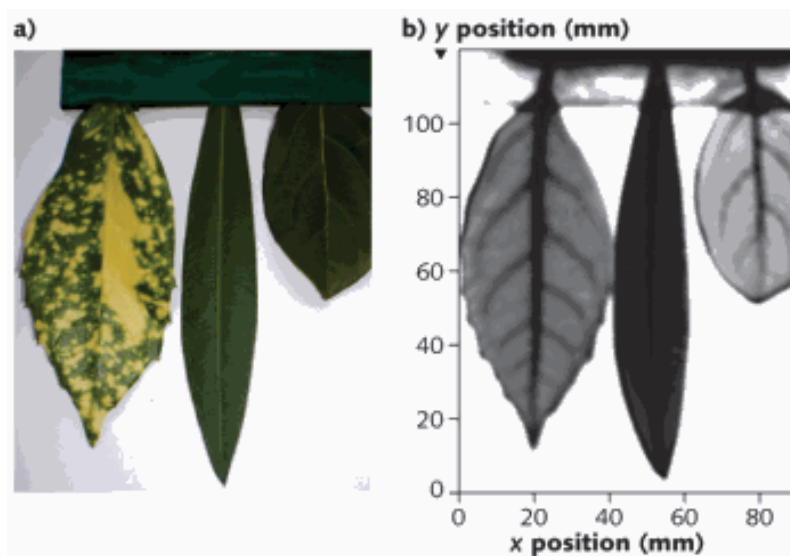
Senėjančių infrastruktūrų, tokių kaip naftos ir dujų vamzdynų, išlaikymas ir apžiūra yra svarbi problema. Tačiau pasinaudojus terahercine vaizdinimo sistema, po visomis izoliacinėmis medžiagomis ant vamzdynų paviršiaus, galima surasti korozijos vietas [10].



2pav. Krepšių skenavimas terahercine spinduliuote [12]



3pav. Pramoninis automobilių radaras



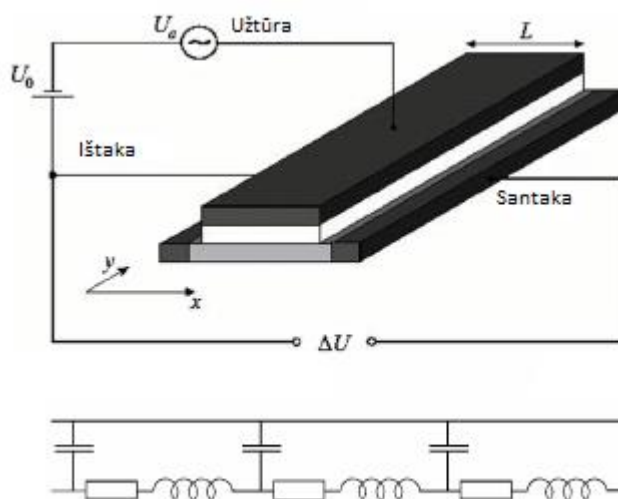
4pav. Lapai matomi šviesoje (kairėje) ir jų THz vaizdinimas (dešinėje)

2. Lauko tranzistoriai kaip terahercinės spinduliuotės detektoriai

Lauko tranzistorių kanalas gali elgtis kaip plazminių bangų rezonatorius. Šio rezonatoriaus dažnis priklauso nuo jo matmenų, taigi esant užtūros ilgiui tarp mikrometrų ir nanometrų galima pasiekti ir terahercų sritį.

Galimybė lauko tranzistoriui detektuoti terahercinę spinduliuotę atsiranda dėl tranzistoriaus netiesinių savybių, kurios lemia krintančios spinduliuotės sukeltos kintamos srovės lyginimą. Dėl to susikuria nuolatinė įtampa tarp santako ir ištako, kuri yra proporcinga spinduliuotės galiai. Akivaizdu, kad tarp ištako ir santako reikia šios tokios asimetrijos.

Viena iš jų tai ištakos ir santakos ribinių sąlygų skirtumas dėl išorinių parazitinių talpų. Kita asimetrija susidaro dėl ateinančios spinduliuotės patekimo būdo į tranzistorių, kuris gali būti realizuojamas panaudojant specialią anteną arba suprojektuojant asimetrines ištakos ir santakos kontaktines aikšteles. Galiausiai asimetrija gali natūraliai iškilti jei tiesioginė srovė pratekės pro ištaką ir santaką, taip sukuriant elektronų tankio išsekimą kanalo ištakos pusėje [1].



5 pav. Lauko tranzistoriaus kaip THz detektoriaus schema (aukščiau) ir ekvivalentinė grandinė (žemiau) [13]

Bendrai, lauko tranzistoriai gali būti aprašyti ekvivalentine grandine, pavaizduota 5 paveikslėlyje. Paskirstyta užtūros į kanalą talpa ir kanalo varža priklauso nuo užtūros įtampos per elektronų koncentraciją kanale:

$$en = CU$$

Čia e – elementarusis krūvis, n – elektronų koncentracija kanale, C – užtūros į kanalą talpa paskirstyta ploto vienetu ir U – užtūros į kanalą įtampa.

Vieni iš tokių lauko tranzistorių, kurie gali detektuoti terahercinę spinduliuotę kambario temperatūroje – GaAs/AlGaAs ir GaInAs/GaAs didelio elektronų judrio tranzistoriai [3]. Paskui buvo ištirta, kad ir Silicio KMOP tranzistoriai yra efektyvūs terahercinės spinduliuotės detektoriai kambario temperatūroje [4].

3. Žingsniniai varikliai

Žingsninis variklis – tai elektromechaninis įrenginys, kuris elektrinius impulsus paverčia į pavienius mechaninius judesius. Kai elektriniai impulsai tam tikra tvarka yra nusiunčiami į variklį, velenas pasisuka tam tikru diskrečiu žingsniu. Kryptis į kurią suksis velenas priklauso nuo elektrinių impulsų sekos, tuo tarpu greitis, kuriuo sukasi velenas, priklauso nuo impulsų dažnio. Valdant žingsninį variklį kompiuteriu galima pasiekti itin tikslią poziciją ir norimą greitį.

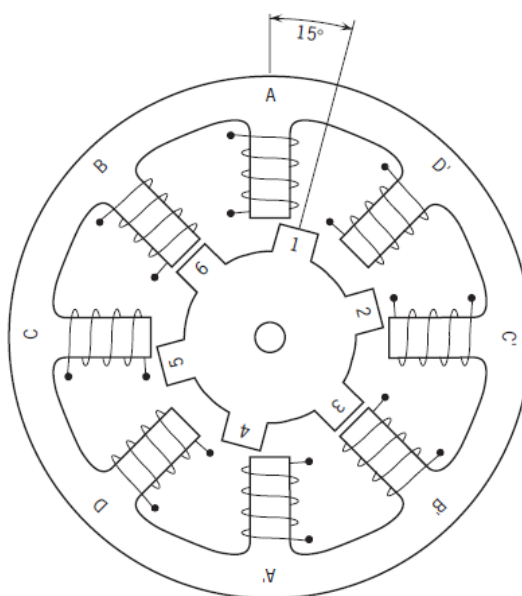
3.1. Žingsninių variklių tipai

Yra trys pagrindiniai žingsninių variklių tipai: reaktyvieji, su nuolatiniu magnetu ir hibridiniai.

3.1.1 Reaktyvieji žingsniniai varikliai

Šis žingsninių variklių tipas sudarytas iš minkštosios geležies krumpliūoto rotoriaus ir statoriaus, pagaminto iš daugybės plieno sluoksnių bei turinčio tam tikrą skaičių vienodai nutolusių išsikišusių stulpų, kurių kiekvienas apvyniotas žadinimo rite. 6 paveikslėlyje pavaizduotame žingsniniame variklyje rotorius sudarytas iš šešių krumplių, kurių plotis yra lygus statoriaus stulpų, kurių pavaizduoti aštuoni, pločiui. Statoriaus ritės yra sugrupuotos po dvi, taip gaunant keturias atskiras grandines, dar vadinamomis fazėmis: A, B, C ir D, kurios turi atskirus jungiklius. Kiekviena fazė gali būti suaktyvinta tiesioginės srovės impulsais. Aktyvavus fazę, apvijos įsimagnetina ir viena iš jų tampa šiauriniu magnetiniu poliumi, kita – pietiniu ir priverčia rotorį pajudėti vienu žingsniu į priekį (arba atgal) į tokią poziciją, kuri sudarys mažiausios magnetinės varžos kelią su įmagnetintais statoriaus dantimis. Kad rotorius suktųsi be perstojo, reikia kiekvieną fazę suaktyvinti paeiliui.

Esant šešiams rotoriaus krumpliams ir keturioms fazėms, vienos fazės aktyvavimo metu, rotorius pasisuka 15 laipsnių. Naudojant daugiau rotoriaus krumplių ir statoriaus išsikišusių stulpų galima sukonstruoti variklį su ypatingai mažu žingsniokampu.



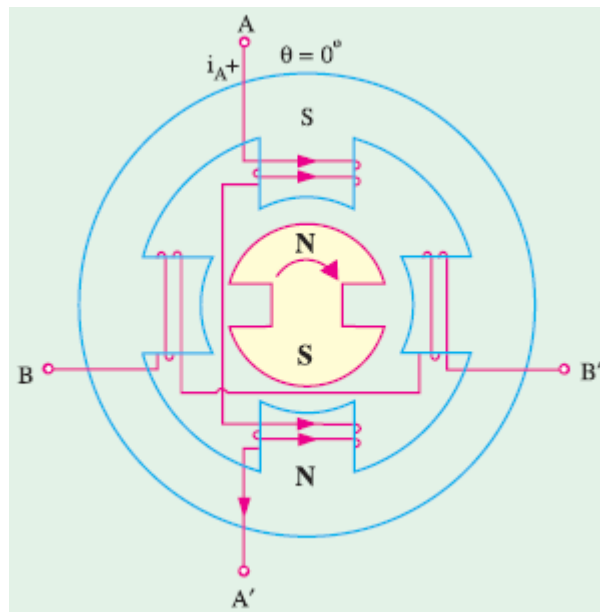
6pav. Reaktyviojo žingsninio variklio skersinis pjūvis [14]

3.1.2 Žingsniniai varikliai su nuolatinio magnetu

Šio žingsninio variklio statoriaus konstrukcija yra panaši į reaktyviojo žingsninio variklio, tačiau rotorius sudarytas iš nuolatinio magneto. Kaip pavaizduota 7 paveikslėlyje, statorius turi išsikišusius stulpus, bet rotorius yra cilindro formos ir turi radialiai įmagnetintus nuolatinis magnetus.

Kai tam tikra statoriaus fazė yra aktyvuojama, rotoriaus magnetiniai poliai susilygiuoja su aktyvuotais statoriaus poliais. Statoriaus apvijos gali įgauti bet kurį poliškumą, priklausomai nuo srovės tekėjimo krypties. Iš pradžių suaktyvuojama A fazė, taip kad šiaurinis rotoriaus polis susilygiuotų su A apvija, tada aktyvuojama B fazė, kad šiaurinis polius susilygiuotų su B' apvija, taip rotorius pasisuka 90 laipsnių kampu. Tada vėl aktyvuojama A fazė ir A' apvija susilygiuoja su šiauriniu rotoriaus poliu. Galiausiai vėl aktyvuojama B fazė ir rotorius atlieka pilną 360 laipsnių apsisukimą.

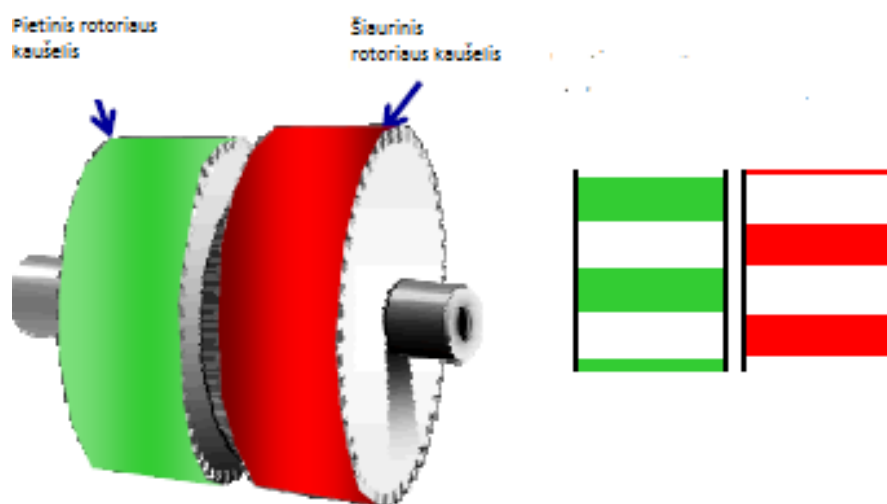
Kadangi nuolatiniai magnetai nereikalauja išorinės sužadinimo srovės, tokie varikliai nereikalauja daug galios, tačiau turi didelį fiksatoriaus sukimo momentą. Šie varikliai turi didelę inerciją, tai reiškia lėtesnę akceleraciją. Pagaminti mažą nuolatinio magneto rotorių su dideliu polių skaičiumi yra pakankamai sunku, dėl to tokių variklių žingsnio kampas yra palyginti didelis – nuo 30 iki 90 laipsnių.



7pav. Žingsninio variklio su nuolatinio magneto skersinis pjūvis [15]

3.1.3 Hibridiniai žingsniniai varikliai

Hibridiniai žingsniniai varikliai sujungia reaktyviųjų ir nuolatinio magneto žingsninių variklių bruožus. Šio variklio rotorius yra krumpliukas bei turi ašies kryptimi, apie veleną, įmagnetintą koncentrinį magnetą. Rotoriaus krumpliai suteikia kelią, kuris padeda orientuoti magnetinį srautą į reikiamas vietas oro tarpuose. Statorius kaip ir kituose varikliuose, susidaro iš kelių suporuotų stulpų su apvijomis. Veikimo principas panašus, skiriasi tik jo charakteristikos.



8pav. Hibridinio žingsninio variklio rotoriaus šoninis vaizdas ir krumplių išsidėstymas [16]

3.2 Žingsninių variklių charakteristikos

3.2.1 Sukimo momentas

Sukimo momentas žingsniniuose varikliuose yra suformuojamas kai rotorius ir statoriaus magnetiniai srautai pakeičia poziciją vienas kito atžvilgiu. Statorius būna pagamintas iš didelės skvarbos magnetinės medžiagos. Tai ir yra priežastis dėl kurios magnetinis srautas yra izoliuojamas statoriaus viduje. Taip visas magnetinis srautas sukoncentruojamas statoriaus stulpuose. Sukuriamas sukimo momentas yra proporcingas magnetinio srauto intensyvumui, sukuriamam kai apvijos yra suaktyvintos.

3.2.2 Dvipoliai ir vienpoliai varikliai

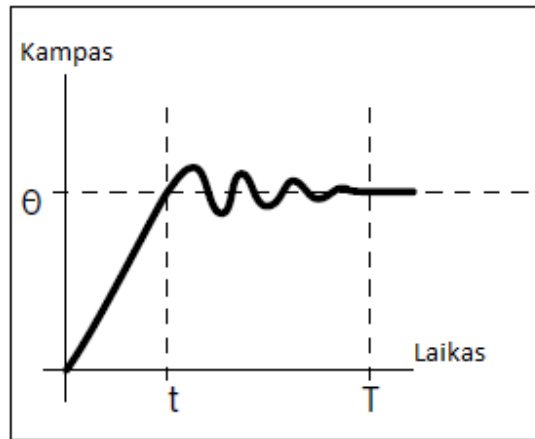
Dažniausiai žingsniniai varikliai turi dvi fazes, bet trijų ar daugiau fazių variklių taip pat egzistuoja.

Dvipolis žingsninis variklis, kaip ir vienpolis, turi dvi apvijas. Tačiau vienpolis turi ir centrinės atšakos. Šių centrinių išvadų turėjimas nėra į naudą, kadangi srovė vienu metu teka ne per visą, o tik per pusę apvijos. Tuo tarpu srovės tekėjimas dvipoliuose varikliuose yra dvikryptis. Dėl šios priežasties tokio pat dydžio dvipoliai varikliai sukuria didesnę sukimo momentą nei vienpoliai.

3.2.3 Vieno žingsnio atsakas

Kai vienas impulsas paduodamas į žingsninį variklį, rotorius elgesys atitinka 4 paveikslėlyje pavaizduotą kreivę. Žingsnio laikas t – tai laikas per kurį variklio velenas pasisuka vieno žingsnio kampą nuo tada, kai paduodamas impulsas. Šis laikas ypatingai priklauso nuo sukimo momento ir inercijos santykio. Kai judama dideliu žingsnių kiekiu iš karto, yra sukuriamas didelis sukimo

momentas, vadinasi ir didelis pagreitis. Tai ir gali sukelti pavaizduotus virpesius. Nusistovėjimo laikas T parodo laiko tarpą, per kurį šie svyravimai baigsis.



9pav. Vieno žingsnio rotoriaus atsakas laike [17]

3.4 Žingsninių variklių panaudojimas

Kai reikalingas reguliuojamas judėjimas, žingsninis variklis yra vienas iš geriausių pasirinkimų. Jie gali būti panaudoti kai reikia kontroliuoti sukimosi kampą, greitį, poziciją ar vienalaikiškumą. Dėl išvardintų privalumų, žingsniniai varikliai turi daugybę panaudojimo sričių, tokių kaip: spausdintuvai, kreivių braižymo įrenginiai, aukštos klasės biuro įranga, kietieji diskai, medicininė įranga, savaeigės mašinos ir daug kitų.

4. Žingsninio variklio sistema

Žingsnio variklio sistemą sudaro trys įrenginiai: valdiklis, tvarkyklė ir variklis. Kiekvieno iš jų savybės turi poveikį kitiems.

Valdiklio paskirtis yra siųsti komandas, kuriomis sugeneruojami impulsai. Keisdami šių impulsų dažnį galime liepti varikliui didinti greitį, judėti pastoviu greičiu ar įsakyti sustoti. Žingsninio variklio tvarkyklės funkcija yra kontroliuoti srovės dydį ir jos tėkmės kryptį į variklio apvijas. Tvarkyklė gauna impulsus iš valdiklio, kurie ir nulemia kaip ir kada apvijos turėtų būti suaktyvintos. Apvijos privalo būti suaktyvintos tam tikra tvarka, kad susidarytų judėjimas.

4.1 Mikrovaldiklis

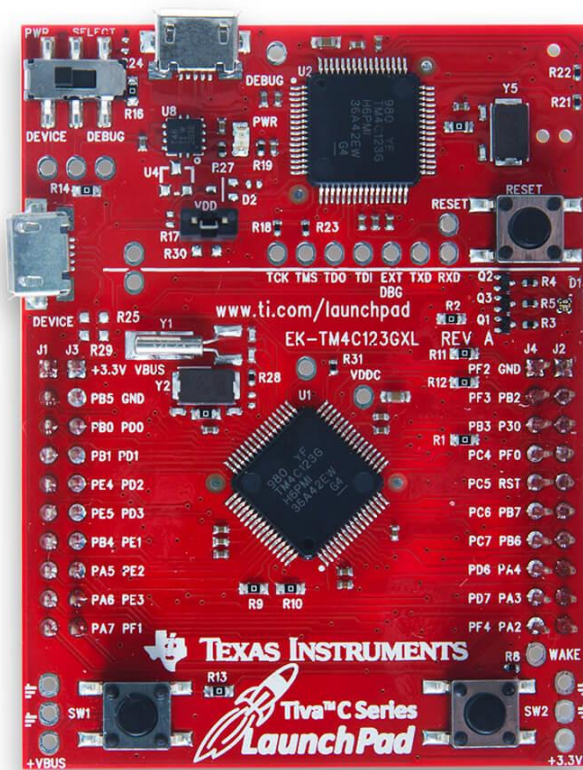
Kuriamoje sistemoje valdiklio poziciją užims Tiva C Serijos TM4C123GXL LaunchPad, kuris pavaizduotas 5 paveikslėlyje. Jis pasižymi borto emuliacija, kas leidžia programuoti ir taisyti projektus be jokių papildomų įrankių.

Šiame rinkinyje yra TM4C123GH6PM mikrovaldiklis, du universalūs vartotojo jungikliai, pakartotinės kelties mygtukas, maitinimo šviesos diodas (LED) ir vartotojo programuojamas RGB LED. Taip pat jame įrengti du, dvilypiai 20-ties kojelių pagrindai, kurie suteikia galimybę prisijungti norimus įrenginius, tokius kaip ekranus, bevielę sąsają, jutiklius ir daug kitų. Prie jų mes prisijungsime žingsninį variklį.

Pagrindiniai TM4C123GH6PM mikrovaldiklio bruožai:

- 32-bitų ARM Cortex-M4 80-MHz procesoriaus branduolys su Sistemos Laikrodžiu (SysTick), integruotu vektoriniu pertraukčių kontrolieriu (NVIC), pabudimo pertraukčių kontrolieris (WIC), atminties saugojimo sekcija (MPU), vieneto tikslumo slenkančio kabelio sekcija (FPU) ir sistemos valdymo blokas (SCB)
- Du kontrolierio srities tinklo (CAN) moduliai, naudojantis CAN protokolo 2.0 versiją
- Universaliosios jungties (USB) kontrolieriai su pilno (12Mbps) ir žemo (1.5Mbps) greičio veikomis
- Analoginis palaikymas: du 12-bitų analoginio į skaitmeninį signalo konverteriai (ADC)

- Pažengusi judesio kontrolė: aštuoni impulso pločio moduliacijos (PWM) generacijos blokai, kiekvienas su 16-bitų skaitikliu, dviem PWM komparatoriais, PWM signalo generatoriumi ir pertraukčių/ADC-trigerio selektoriumi.
- Iki 43 bendros paskirties įėjimų ar išėjimų (GPIO)
- Daugybė laikrodžių šaltinių mikrovaldiklio sistemos laikrodžiui: tikslumo osciliatorius (PIOSC), pagrindinis osciliatorius (MOSC), 32.768-kHz išorinis osciliatorius žiemojimo moduliui ir vidinis 30-kHz osciliatorius.



10pav. Tiva C Serijos TM4C123GXL LaunchPad

Bendros paskirties įėjimo/išėjimo modulis yra sudarytas iš šešių fizinių GPIO blokų, kurių kiekvienas atitinka individualią GPIO prieigą (Port A, Port B, Port C, Port D, Port E, Port F). GPIO modulis gali palaikyti iki 43 programuojamų įėjimo/išėjimo kojų, priklausomai nuo naudojamų periferijų skaičiaus.

Šiame mikrovaldiklyje taip pat yra aštuoni universalūs asinchroniniai gavėjai/siuntėjai (UART). Jo paskirtis paimti duomenų baitus ir tam tikra tvarka išsiųsti individualius bitus. Paskyrimo vietoje

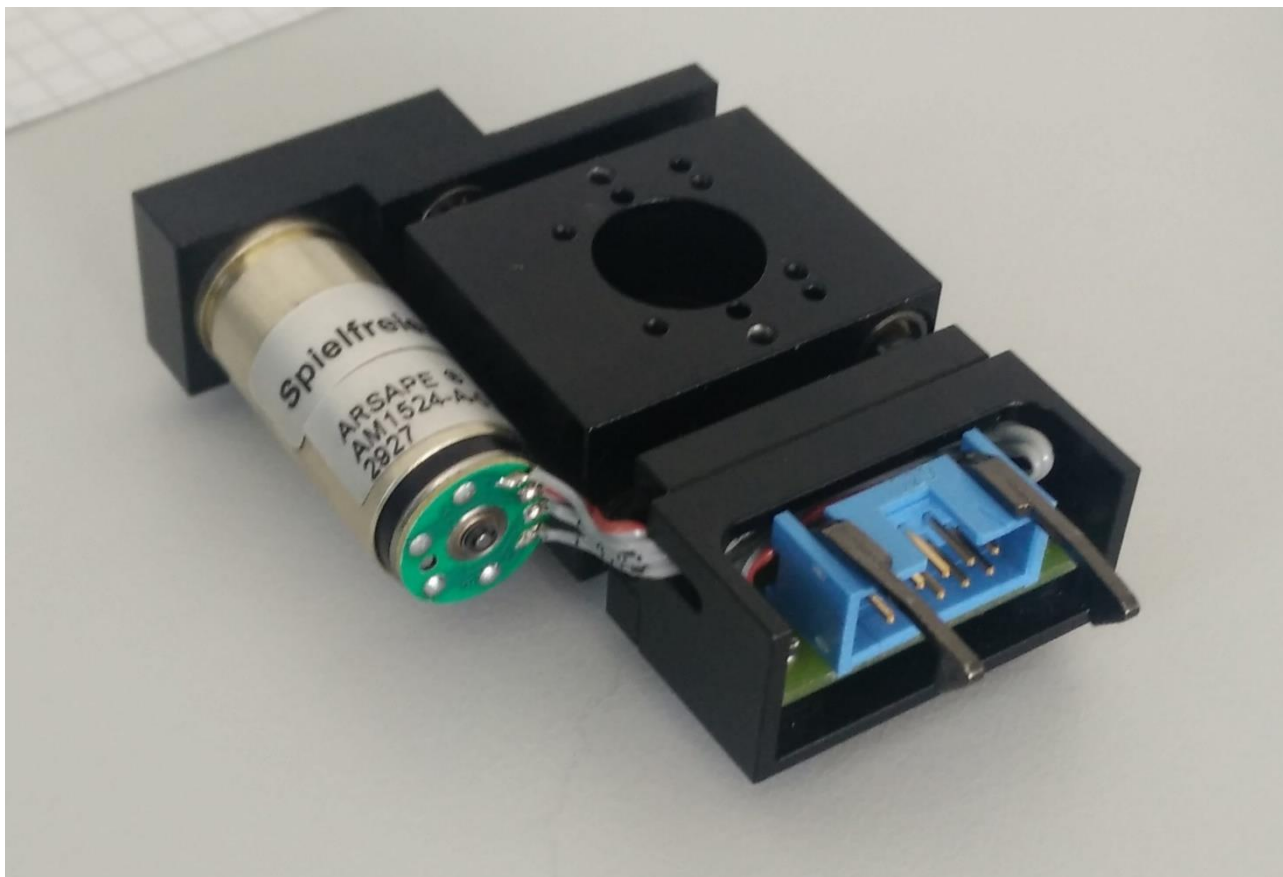
kitas UART surenka gautus bitus į pilnus baitus. Paprastai sakant, jo dėka kompiuteris gali „kalbėtis“ su prie jo prijungtais serijiniais įrenginiais, kaip šis mikrovaldiklis.

4.2 Žingsninis variklis su MT-55 transliaciniu staleliu

Žingsninį variklį su MT-55 transliaciniu staleliu geriausiai tinka įkomponuoti į ribotos erdvės aplinkas. Stalelis yra aprūpintas rutuliniu guoliu be jokios laisvosios eigos ir preciziniu švininiu sraigtu su 0.25mm nuolydžiu. Taip pat yra įmontuoti du mygtukai, kurie apriboja stalelio važinėjimą, taip jį apsaugodami nuo netyčinio nuvažiavimo iš ribų. Pagrindinės jo savybės yra tokios:

- Keliavimo atstumas lygus 5mm
- Didžiausias greitis – 0.25mm per sekundę
- Krovinio dydis iki 0.8kg
- Integruoti ribojimo jungikliai
- Centrinės skylės diametras – 15mm
- Didelė rezoliucija
- Nominali srovė – 0.25mA

Tai - dvipolis variklis, taigi jis turi dvi apvijas ir keturi laidus. Tam, kad galėtume tinkamai jį prijungti prie tvarkyklės, mums reikia identifikuoti į kurias kojeles šie laidai nueina. Tai padaroma multimetrom pagalba.

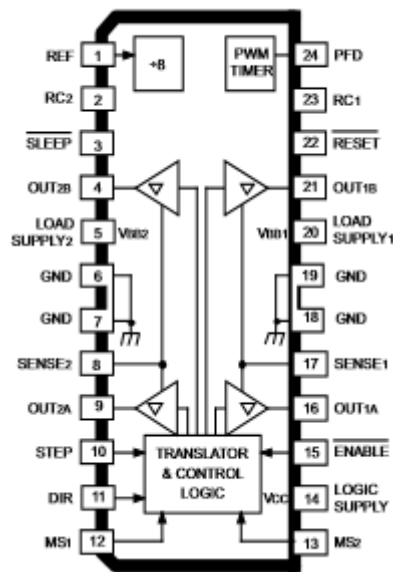


11pav. Žingsninis variklis su MT-55 transliaciniu staleliu

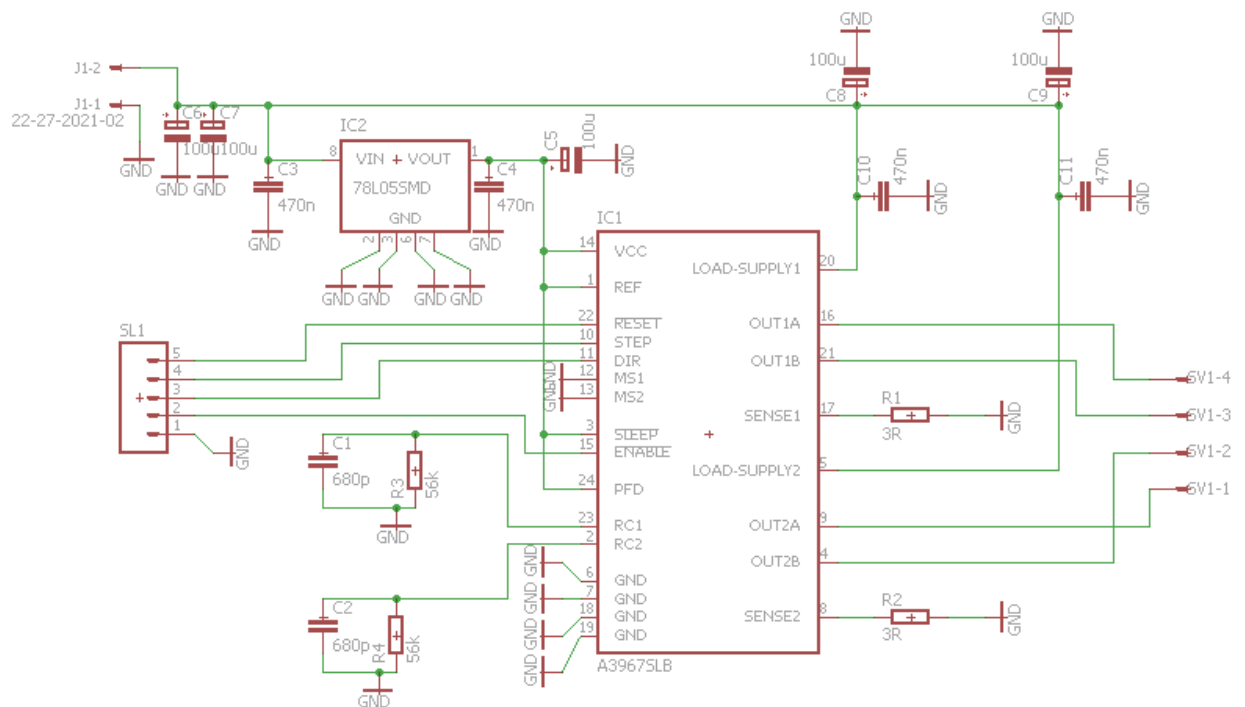
4.3 A3967 tvarkyklė

Tam, kad sukurti impulsai patektų ten kur reikia, reikalinga turėti žingsninio variklio tvarkyklę. Ji pasirenkama atsižvelgiant į naudojamo žingsninio variklio charakteristikas. Kadangi yra naudojamas dvipolis variklis, kurio nominali srovė 0.25mA, buvo pasirinkta kuo mažesnės išėjimo srovės tvarkyklė palaikanti žingsnio ir žingsnio krypties funkcijas dvipoliams varikliams. Tokia ir yra A3967 tvarkyklė.

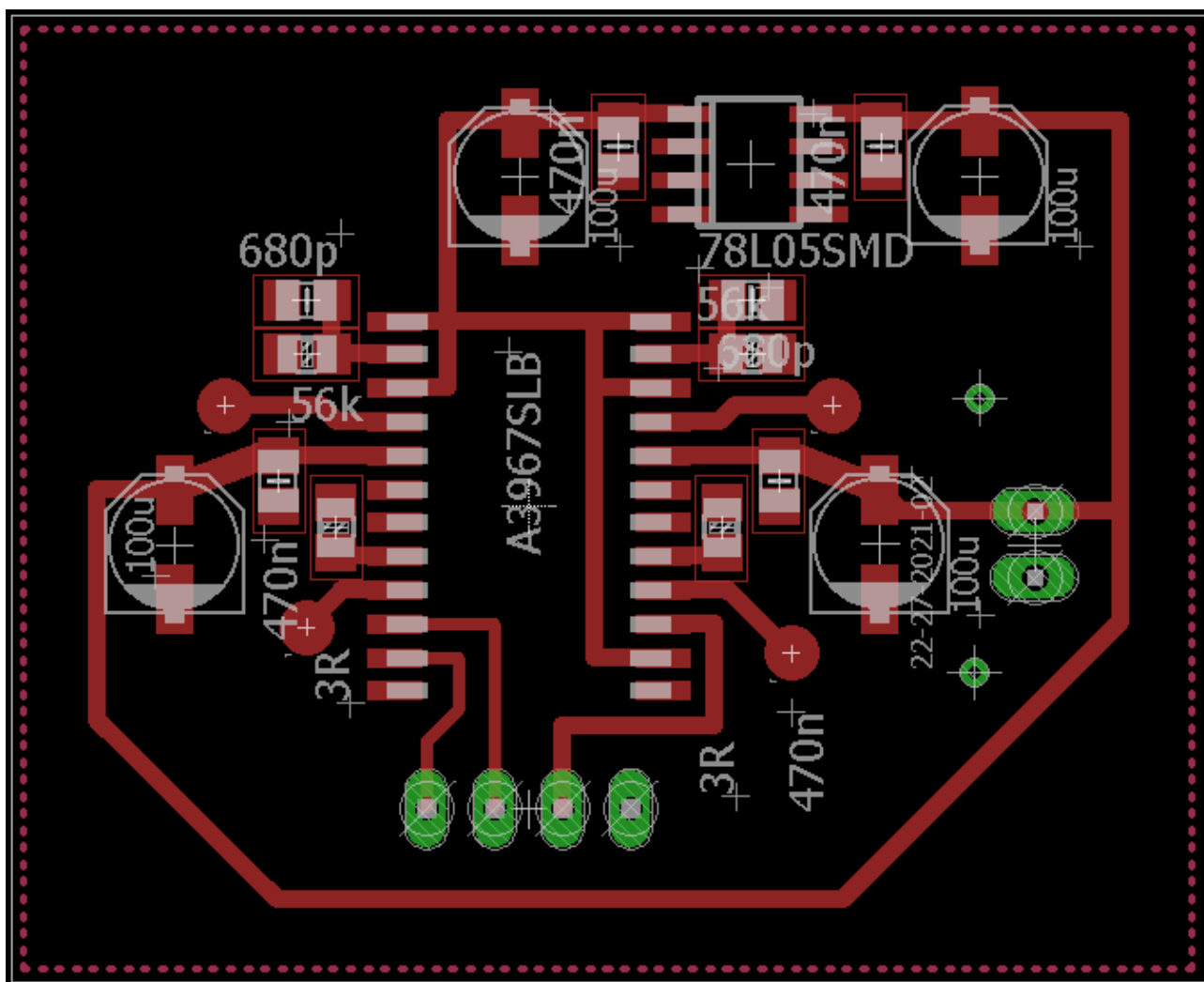
A3967 yra universali mikrožingsninė variklio tvarkyklė su įmontuotu transliatoriumi. Ši tvarkyklė yra sukurta valdyti dvipolius žingsninius variklius pilno, pusės, ketvirčio ar aštuntadalio žingsnio režimais, kurio išėjimas gali siekti 30V ir 750mA. A3967 turi srovės reguliatorių, kuris gali dirbti lėtais, greitais ir mišriais srovės kritimo režimais. Ši srovės kritimo kontrolė sumažina girdimą variklio triukšmą, padidina žingsnių tikslumą ir sumažina galios iššvaistymą.



12pav. A3967 kojelių diagrama [18]



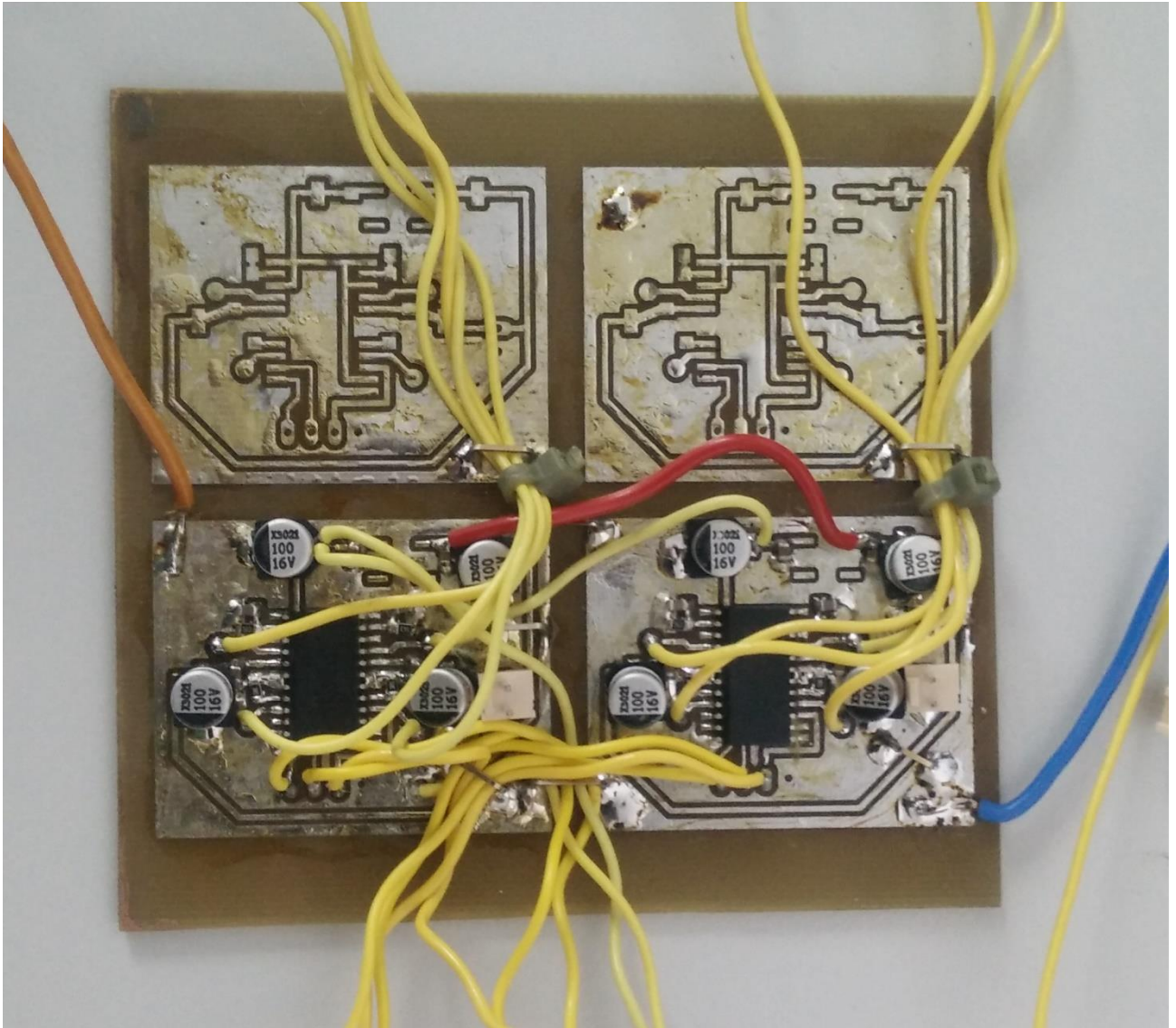
13pav. Žingsninio variklio tvarkyklės suprojektuota elektrinė schema



14pav. Žingsnio variklio tvarkyklės suprojektuota plokštė

Transliatorius – tai pagrindinė priežastis kodėl A3967 tvarkykle galima lengvai valdyti žingsninius variklius. Užtenka paduoti vieną impulsą į žingsnio įvestį ir variklis pasisuks pilnu žingsniu. Nereikia jokių fazės eiliškumo lentelių, aukšto dažnio valdymo linijų ar beprotiškai sudėtingų programų.

A3967 yra pristatomas kaip 24-kojelių mažo kontūro integruota grandinė (SOIC), kurio viduje yra sulydytos keturios kojelės tam, kad sustiprintų šiluminį išsisklaidymą. Kojelės yra žemės potencialo, todėl izoliavimas nereikalingas.



15pav. Sulituotos žingsninių variklių tvarkyklės

Kiekvienas pilnas tiltas yra kontroliuojamas impulso pločio moduliacijos (PWM) srovės valdoma grandine, kuri riboja srovę iki mums reikiamos vertės (I_{TRIP}). Iš pradžių, šaltinio ir nuotėkio išvadų įstrižinės poros yra įgalinamos ir srovė teka per variklio apvijas ir R_S . Kai įtampa ant srovę jaučiančios varžos pasidaro lygi DAC išvesties įtampai, srovei jautrus komparatorius atstato PWM skląstį, taip išjungdamas lėto ar greito kritimo režimus.

Didžiausia srovės ribojimo vertė yra nustatoma pasirenkant R_S varžos ir V_{REF} įvesties įtampos vertes. Apytikslė formulė būtų tokia :

$$I_{TRIP} (max) = V_{REF} / (8 * R_S)$$

Pasiėmus $V_{REF} = 5V$, apskaičiuojama varžos R_S vertė, su kuria mūsų didžiausia srovė bus lygi žingsninio variklio nominalinei srovei, tai yra 0,25A.

$$0,25 = 5 / (8 * R_S) \rightarrow 8 * R_S = 20 \rightarrow R_S = 2,5 \Omega$$

Kad būtume dar labiau užtikrinti jog nesudeginsime žingsninio variklio, naudosime $R_S = 3 \Omega$.

Vidinės PWM srovės valdomos schemos naudoja vieną šūvį nustatant laiką, kurį tvarkyklė liks išjungta. Vieno šūvio neveikimo laikas t_{off} yra nustatomas pasirenkant išorinės varžos R_T ir kondensatoriaus C_T vertes. Apytikslė išraiška yra tokia:

$$t_{off} = R_T * C_T$$

C_T reikšmė taip pat nustato ir komparatoriaus tuštinimo laiką: $t_{BLANK} = 1400 * C_T$

Imsimė vidutines R_T ir C_T reikšmes – $R_T = 56k\Omega$ ir $C_T = 680pF$.

Chronometražo reikalavimai:

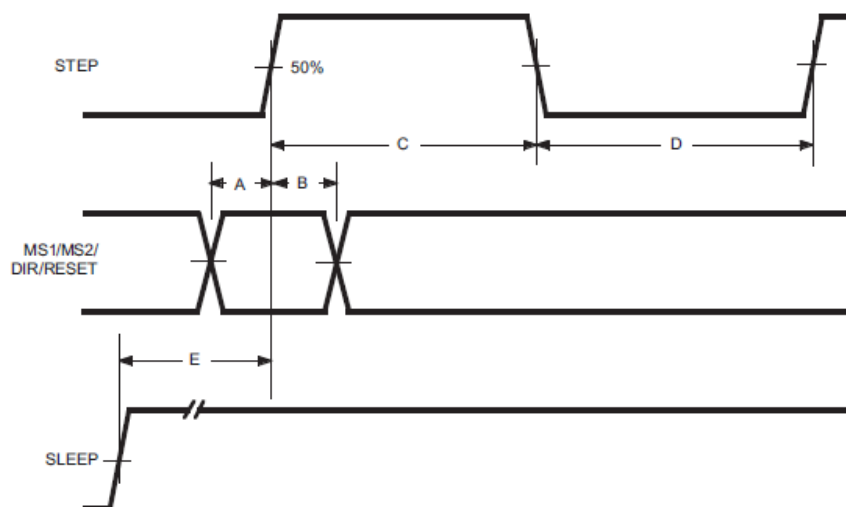
A – Mažiausias komandos aktyvumo laikas prieš žingsnio impulsą – 200ns

B – Mažiausias komandos aktyvumo laikas po žingsnio impulso – 200ns

C – Mažiausias žingsnio impulso plotis - 1μs

D – Mažiausias žingsnio impulso žemo lygio laikas - 1μs

E – Didžiausias prabudimo laikas – 1ms



16pav. Chronometražo reikalavimai [20]

Darbe naudojamas 12V RS-25-12 maitinimo šaltinis, bei 5V įtampos stabilizatorius.

Atspausdinta montavimo schema turi didelę žemės plokštumą. Tam, kad išlaikyti optimalias elektrines ir šilumines savybes, tvarkyklė yra prilituojama tiesiogiai ant plokštės. Maitinimo terminalas nuo tvarkyklės yra atskiriamas 100μF elektrolitiniu kondensatoriumi.

1lentelė. Terminalų sąrašas

Terminalo pavadinimas	Terminalo apibūdinimas	Terminalo numeris
REF	Etaloninė įvestis	1
RC2	Analoginė įvestis fiksuotam neveikimo laikui – tiltui 2	2
SLEEP	Loginė įvestis	3
OUT2B	H tilto 2 išvestis B	4
LOAD SUPPLY2	Tilto 2 maitinimas	5
GND	Žemė	6,7
SENSE2	Jutimo varža tiltui 2	8
OUT2A	H tilto 2 išvestis A	9
STEP	Loginė įvestis	10
DIR	Loginė įvestis	11
MS1	Loginė įvestis	12
MS2	Loginė įvestis	13
LOGIC SUPPLY	Loginė maitinimo įtampa	14
ENABLE	Loginė įvestis	15
OUT1A	H tilto 1 išvestis A	16
SENSE1	Jutimo varža tiltui 1	17
GND	Žemė	18, 19
LOAD SUPPLY1	Maitinimas tiltui 1	20
OUT1B	H tilto 1 išvestis B	21
RESET	Loginė įvestis	22
RC1	Analoginė įvestis fiksuotam neveikimo laikui – tiltui 1	23
PFD	Mišraus kritimo nustatymas	24

4.4 Serijinė sąsaja

Tam, kad individualios grandinės apsiikeistų informacija, jos turi turėti tą patį bendravimo protokolą. Visi šie protokolai gali būti suskirstyti į du tipus: lygiagretų (angl: parallel) ir serijinį (angl: serial). Lygiagreti sąsaja perduoda daugybę bitų iš karto, tuo tarpu serijinė sąsaja – vieną bitą vienu metu. Dėl to, kad lygiagrečioje sąsajoje reikia labai daug įėjimo/išėjimo linijų, dažniausiai naudojama serijinė sąsaja. Ji gali būti suskirstyta į dvi grupes: asinchroninius ir sinchroninius. Sinchroninė serijinė sąsaja visada sugrupuoja duomenų linijas su laikrodžio signalu, tad visi įrenginiai naudoja tą patį laikrodį. Asinchroninėje sąsajoje duomenys yra siunčiami išorinio laikrodžio signalo pagalbos.

Asinchroninis serijinis protokolas turi keletą įmontuotų mechanizmų, kurie padeda garantuoti stiprų ir be klaidų duomenų perdavimą. Šie mechanizmai yra:

- Duomenų bitai (angl: data bits)
- Sinchronizacijos bitai (angl: synchronization bits)
- Lyginumo bitai (angl: parity bits)
- Bodo rodiklis (angl: baud rate)

Bodo rodiklis parodo kaip greitai duomenys yra persiunčiami per serijinę liniją. Dažniausiai išreiškiami bitais per sekundę (bps). Invertuodami bodo rodiklį galime sužinoti kiek laiko užtruks perduoti vieną bitą. Bodo rodiklis gali būti bet koks skaičius, vienintelis reikalavimas yra, kad abu įrenginiai dirbtų tuo pačiu dažniu. Tačiau yra ir limitas, dažniausiai šis greitis neviršys 115200.

Kiekvienas duomenų blokas yra perduodamas bitų paketais. Šie paketai sukuriama pridedant sinchronizacijos ir lyginumo bitus prie mūsų duomenų. Sinchronizacijos bitai – tai du ar trys specialūs bitai paketo pradžioje ir pabaigoje, kurių tikslas nurodyti paketo pradžią bei pabaigą. Lyginumas – tai paprasčiausia klaidų tikrinimo forma, kuri remiasi duomenų bitų sumos skaičiavimu. Jei suma nelyginė, tai lyginumo bitas nustatomas į 1.

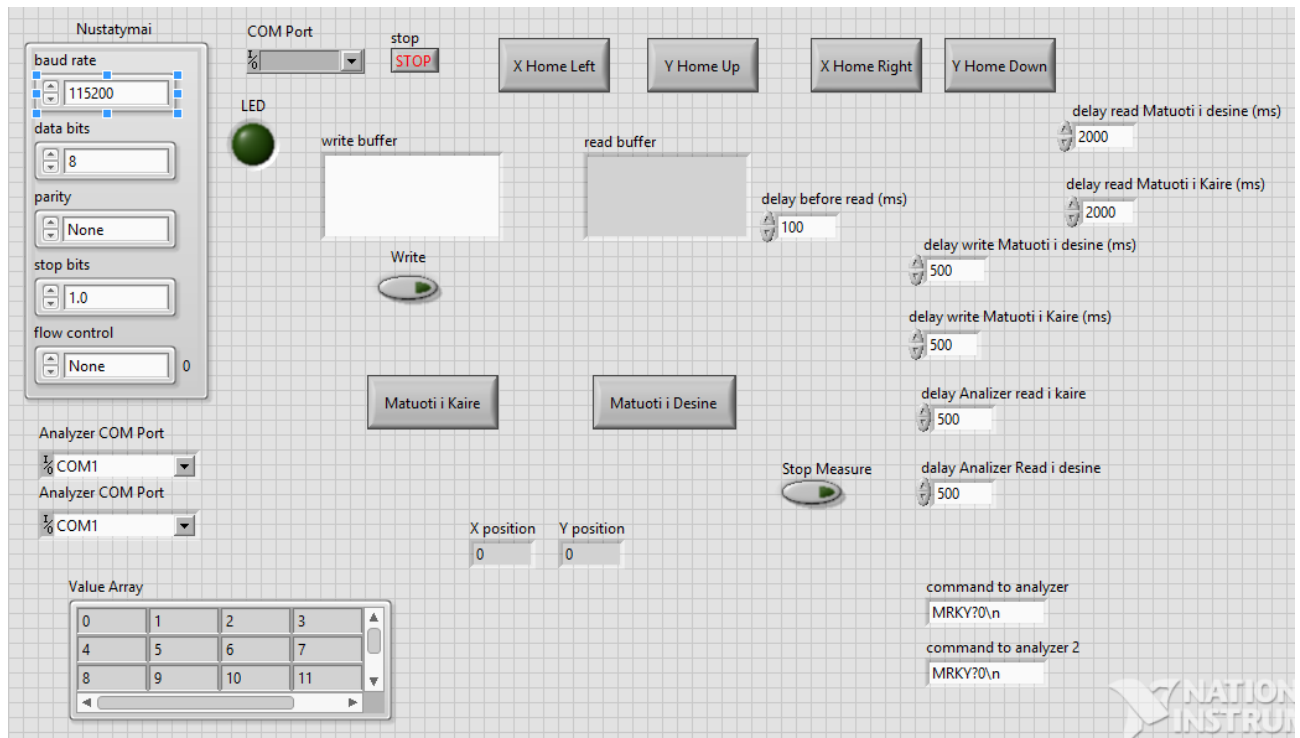
5. Programavimo eiga

Šiame darbe reikėjo suprogramuoti 2 dalykus: mikrovaldiklį, C# kalba, ir pasidaryti aplikaciją, kuri bendraus su mikrovaldikliu, siųs, priims ir apdoro duomenis iš įvairių prietaisų.

Programa mikrovaldikliui buvo rašoma su Code Composer Studio. Iš pradžių įgalinami reikiami periferijos blokai, paskui nusistatoma kurios kojelės yra įėjimai, o kurios - išėjimai. Tada įgalinamas UART, priskiriant jam siuntimo(TX) ir gavimo(RX) kojeles, bei įgalinamos siuntimo ir gavimo pertrauktys. Tai padarius jau galima „kalbėtis“ su mikrovaldikliu ASCII komandomis, pasinaudojant koku nors terminalo emulatoriumi, pavyzdžiui, PuTTY. Kai jau galima priimti ir išsiųsti simbolius, galima pradėti kurti funkcijas, reikalingas žingsninio variklio darbui. Iš pradžių sukuriamos keturios funkcijos, kurių paskirtis sukurti žingsnio impulsą, bei jo kryptį. Prie jų prisegami keturi simboliai, kuriais šios funkcijos bus paleidžiamos.

PuTTY yra nemokamas atviro kodo terminalo emuliatorius, kuriuo galime tik išsiųsti komandas ir nuskaityti gautas. Tačiau mums reikėtų ko nors sudėtingesnio, su kuo galėtume gautus duomenis apdoroti, ir priklausomai nuo jų duoti atitinkamas komandas. Čia pasitarnauja programa LabView. Joje naudojama G programavimo kalba, kuri yra labai intuityvi. Tikslas dirbant su šia programa yra sukurti aplikaciją, kuri siųstų žingsnio komandas į mikrovaldiklį, bei tuo pačiu priimtų duomenis iš analizatoriaus ir viską įrašytų į lentelę.

Iš pradžių yra inicializuojama COM jungtis, nustatomi jos parametrai. 11 paveikslėlyje parodyta aplikacijos sąsaja. Įrašius komandas į „write buffer“ ir paspaudžius mygtuką „write“, jos yra nusiunčiamos į mikrovaldiklį. „read buffer“ rodo gautus simbolius iš mikrovaldiklio. Suprogramuoti keturi „Home“ mygtukai, kurie nuvaro žingsninius variklius į kraštines padėtis. Paspaudžius mygtuką „Matuoti į dešinę“ arba „Matuoti į kairę“ prasideda matavimas, kas 500 žingsnių programa išsaugos analizatoriaus rodmenis ir įrašys į masyvą. Pasiekus žingsninio variklio kraštą, įsijungs ribojimo mygtukas, žingsninis variklis pajudės į apačią 500 žingsnių ir matavimai tęsis į kitą pusę. Pasiekus apačios ribojimo mygtuką matavimai baigsis, ir bus išmesta duomenų išsaugojimo vietos pasirinkimo lentelė.



17pav. Suprogramuotos aplikacijos su LabView priekinė panelė

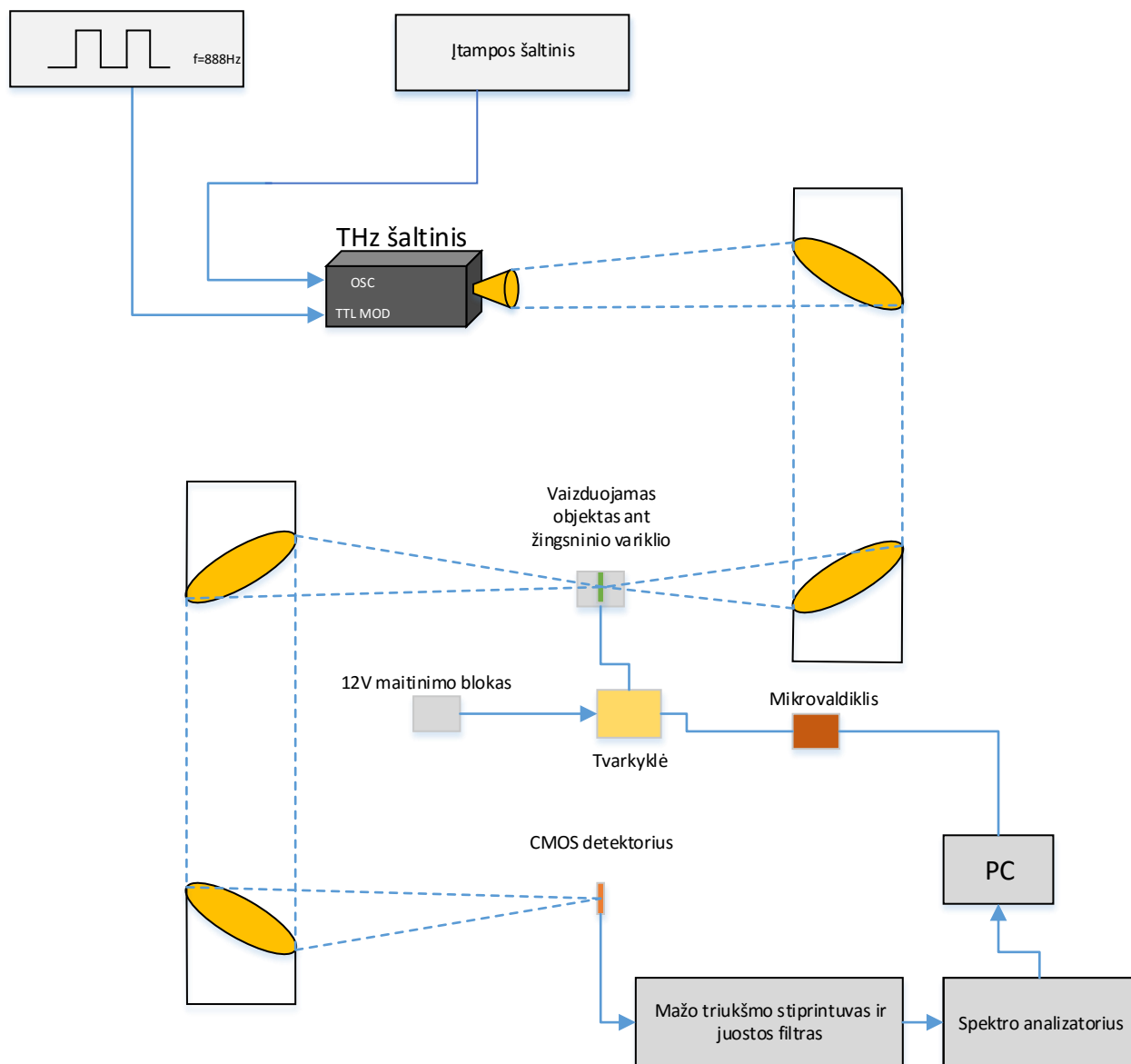
6. Matavimo metodika

Vaizdinimo schema susideda iš terahercinės spinduliuotės šaltinio, keturių sferinių veidrodžių, THz detektoriaus bei žingsninio variklio sistemos.

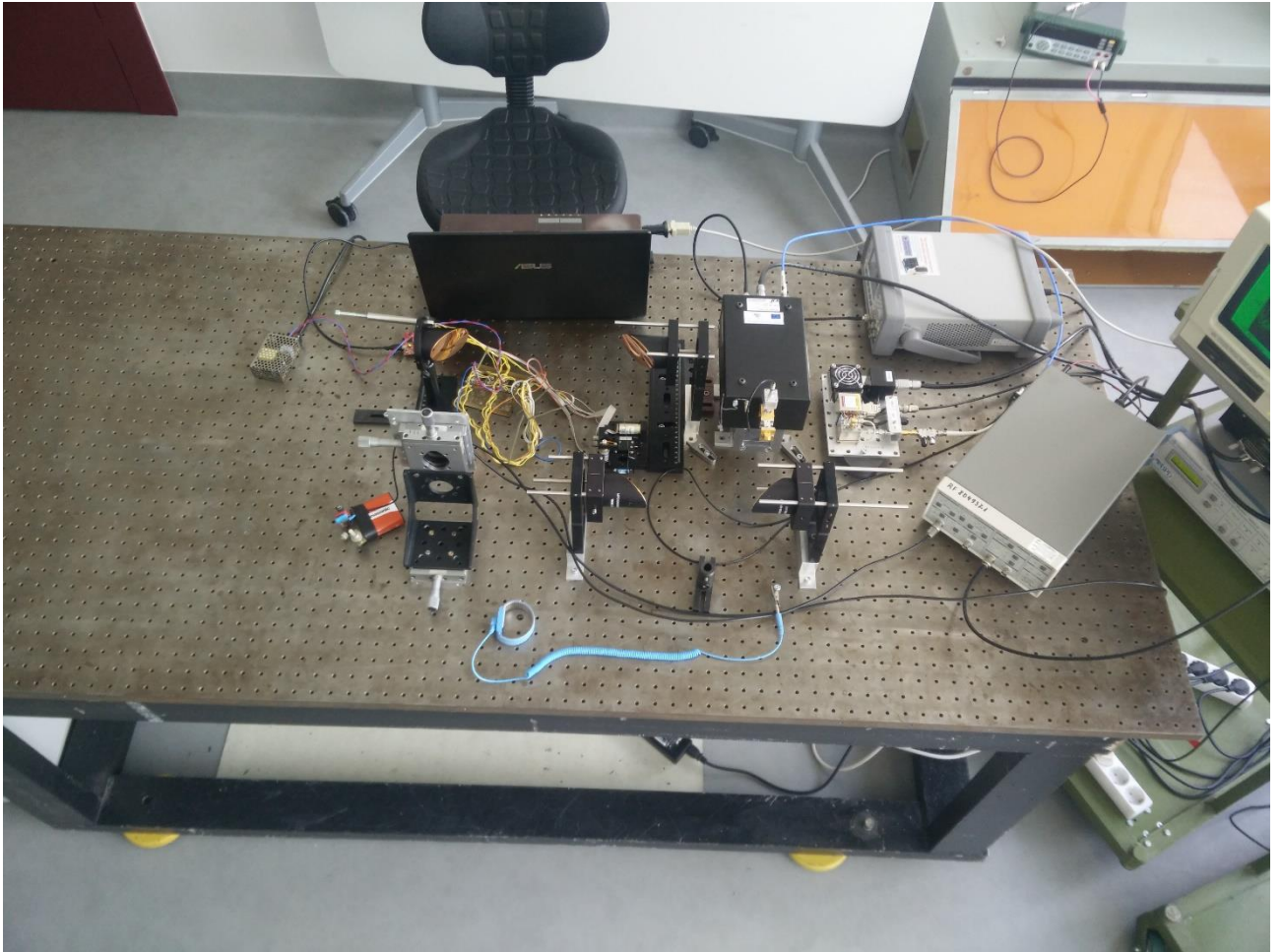
Terahercinės spinduliuotės dažnis priklauso nuo paduotos įtampos dydžio, tačiau atliekant vaizdinimą tereikia rasti dažnį ties, kuriuo detektoriaus rodmenys didžiausi. Išėjusi iš šaltinio spinduliuotė patenka į pirmąjį sferinį veidrodį, po kuriuo spinduliuotė jau keliauja lygiagrečiai. Antrasis sferinis veidrodis sufokusuoja gautą spinduliuotę į tašką židinio nuotolio atstumu. Ypatingai svarbu rasti šį tašką, nes ten turime pastatyti tiriamą objektą. Iš šio taško spinduliai keliauja taip lyg tai būtų taškinis šaltinis, todėl vėl yra panaudojami du sferiniai veidrodžiai, kurių vienas padaro pluoštą lygiagretų, o kitas sufokusuoja į detektorį. Kai detektorį pasiekia terahercinė spinduliuotė, tarp ištako ir santako susidaro įtampa. Ji keliauja į mažų triukšmų stiprintuvą, kur yra sustiprinama. Po to signalas keliauja į spektro analizatorių, kur mes jį jau matome.

Matuojant detektoriaus charakteristikas, matavimo schema buvo panaši – nereikėjo dviejų paskutinių veidrodžių ir detektorius buvo statomas vietoj vaizduojamojo objekto. Taip pat buvo naudojama Golėjaus celė, kad sukalibruotume detektorį.

Surasti fokusą yra sudėtinga ir tai užtrunka daug laiko, ypač kai yra tiek daug kintamųjų, todėl iš pradžių fokusas ieškomas su dviem veidrodžiais, vietoj bandinio įstatant detektorį. Kai fokusas randamas, detektorius perkliamas už keturių veidrodžių ir procesas kartojamas.



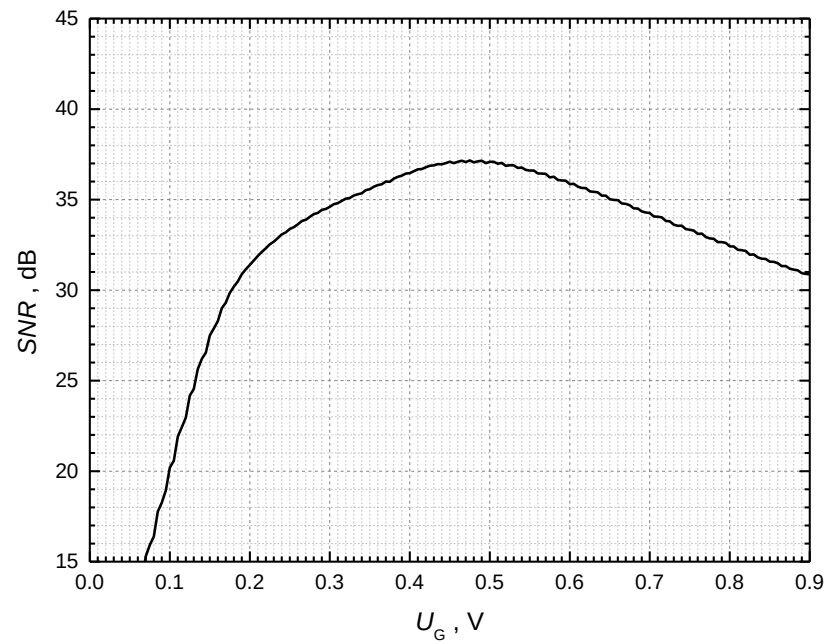
18pav. Principinė matavimo metodikos schema



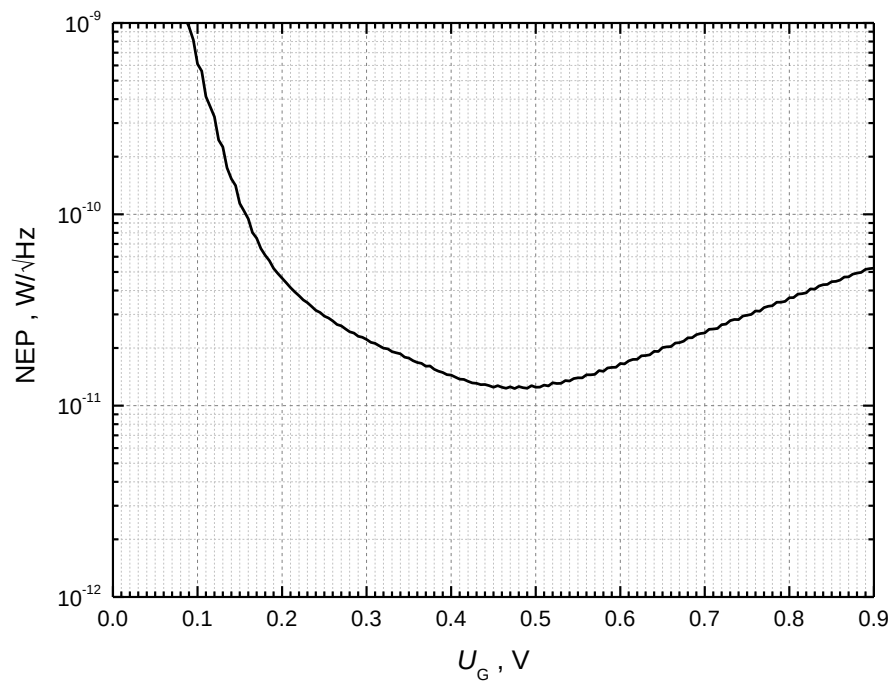
19pav. Matavimo schemos nuotrauka

7. Rezultatai ir jų aptarimas

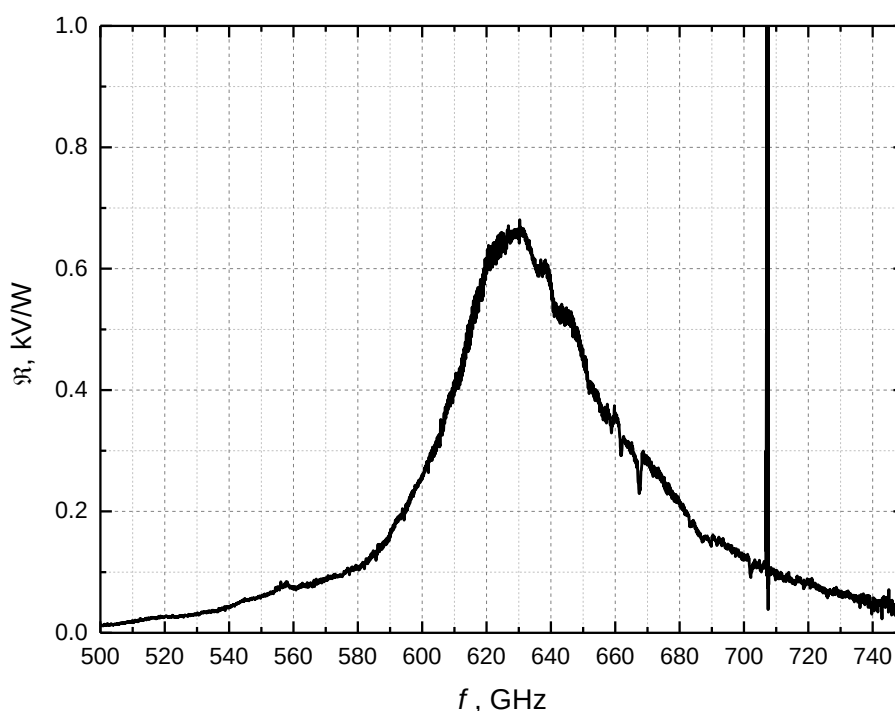
Iš pradžių buvo ištirtos silicio KMOP THz detektoriaus charakteristikos. Buvo tiriama jo galios atsakas keičiant užtūro įtampą ir spinduliuotės dažnį.



20pav. Signalų ir triukšmo santykio priklausomybė nuo užtūro įtampos



21pav. Ekvivalentinės triukšmų galios priklausomybė nuo užtūros įtampos



22pav. Silicio KMOP THz detektoriaus atsako priklausomybė nuo dažnio

Keičiant užtūro įtampa nustatyta, kad SNR (signalų ir triukšmo santykis) didėja kol pasiekama slenkstinė tranzistoriaus įtampa. Artėjant prie slenkstinės įtampos, kanalo varža diverguoja į begalybę ir tranzistorius elgiasi kaip įtampos šaltinis su labai didele vidine varža. Ten kur SNR yra didžiausias, NEP (angl: noise equivalent power) – mažiausias. Taip yra dėl to, kad NEP parodo kokią galią sukuria triukšmai, taigi kuo signalo su triukšmais santykis didesnis, tuo triukšmų sukurta galia mažesnė. Iš grafiko matyti, kad naudojamo detektoriaus slenkstinė įtampa yra ~ 0.47 V.

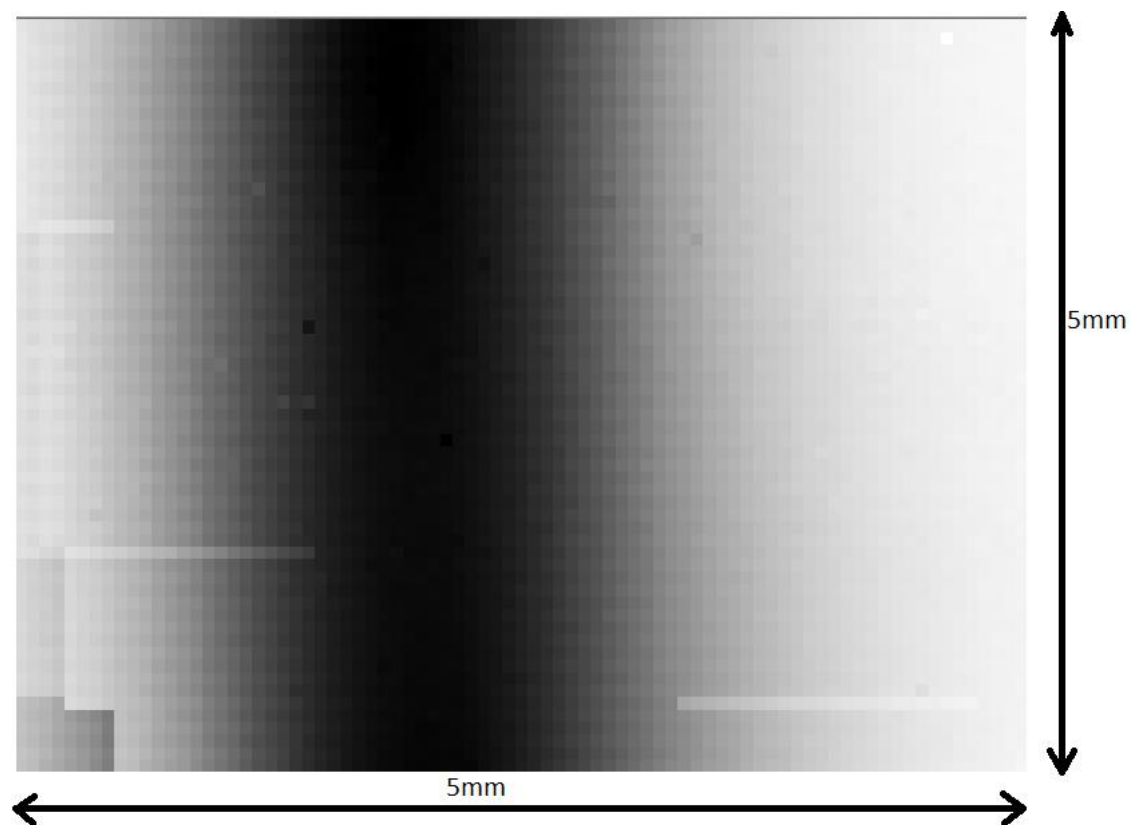
Išmatavus detektoriaus jautrio priklausomybę nuo dažnio gavome, kad yra vienas išsišokimas. Taigi, mūsų detektorius – rezonansinis, kurio rezonanso dažnis 630GHz. Ties šiuo dažniu detektoriaus jautris didžiausias, todėl vaizdinimas pavyks geriausiai esant šiam dažniui.

Buvo atvaizduoti du objektai: lydmetalio gabalas ir šviežiai nuskintas medžio lapas. Analizatoriumi buvo matuojama susidariusi detektoriuje įtampa ir kiekviename taške įrašoma į lentelę. Gauti duomenys sunormalizuoti ir atvaizduoti 24 ir 26 paveikslėliuose. 24 paveikslėlyje juodas stulpas centre vaizduoja lydmetalio gabalą, taigi pro jį THz spinduliuotė sunkiai praeina. 26 paveikslėlyje dešinėje pusėje matyti šviežiai nuskinto lapo gijoje esantys vandens pėdsakai.

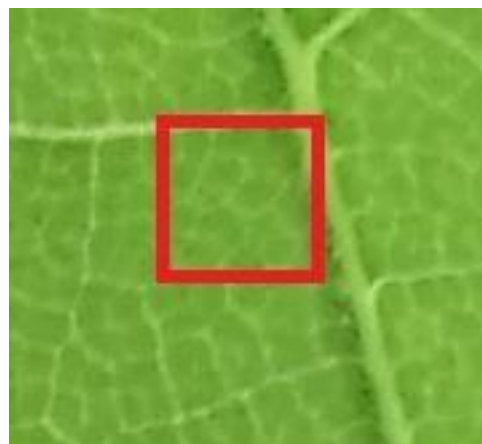
Kadangi žingsninių variklių stalelių kelionės atstumas yra tik 5mm, atvaizduoti didelius objektus yra sunku, todėl ši sistema geriausiai tinka mažų objektų vaizdinimui. Tačiau įsigijus didesnio žingsnio nuotolio žingsninius variklius bei didesnio kelionės atstumo stalelius galima nesunkiai juos integruoti ir į šią vaizdinimo sistemą.



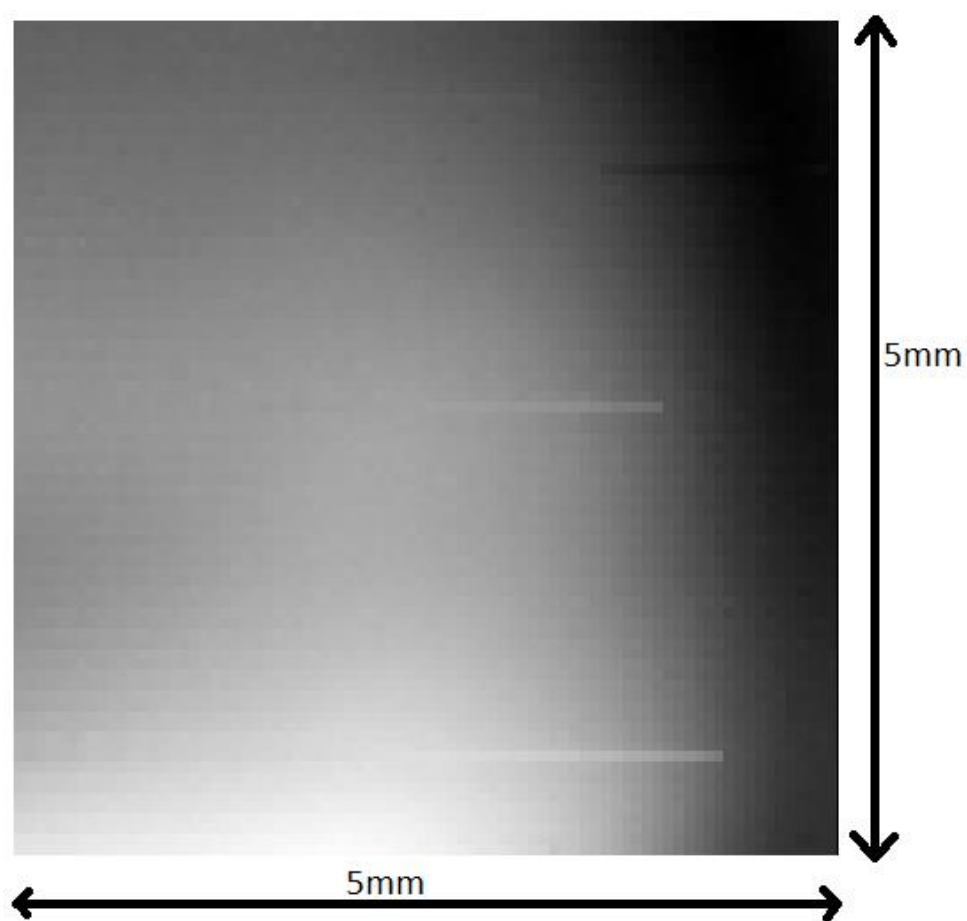
23pav. Lydmetalio vaizdas regimojoje šviesoje



24pav. Lydmetalio THz vaizdas



25pav. Šviežiai nuskinto medžio lapo vaizdas regimojoje šviesoje



26pav. Šviežiai nuskinto medžio lapo fragmento THz vaizdas

Išvados

- Sukurta pozicionavimo sistema nesukuria trikdžių, kurie įtakotų terahercų spinduliuotės detekcijos sistemą
- Silicio KMOP technologijoje pagamintas terahercų spinduliuotės detektorius su lopo tipo antena, kurio rezonansinis dažnis yra 630GHz, yra tinkamas metalinių objektų aptikimui
- Jutiklio rezonansinis dažnis nesutampa su vandens garų sugerties linija, todėl jutiklis gali būti pritaikytas daug vandens turinčių biologinių objektų atvaizdavimui

Santrauka

Vaidotas Kumža

Terahercinės spinduliuotės vaizdinimo sistema su rezonansiniu mažos triukšmų galio detektoriumi

Terahercinė spinduliuotė pasižymi unikaliomis savybėmis. Ji nėra jonizuojanti, gali prasiskverbti pro daugelį dielektrinių medžiagų. Vanduo vienas geriausių jos sugėriklių, o metalai gerai ją atspindi. Dėl to atsiranda daugybė jos pritaikymo sričių.

Šio darbo tikslas yra susikurti THz vaizdinimo sistemą panaudojant žingsninius variklius ir silicio KMOP detektorių bei ištirti kelis pasirinktus objektus.

Tam, kad tiksliai galėtume keisti bandinio padėtį, buvo naudojamas žingsninis variklis. Sujungus du tokius variklius galėjome keisti objekto padėtį dviem ašimis. Norint jį valdyti kompiuteriu, vien žingsnių variklių neužtenka, reikia turėti mikrovaldiklį bei tvarkyklę. Gaminantis tvarkyklę reikia žinoti žingsninio variklio charakteristikas.

Norint gauti vaizdinimą, reikia sufokusuoti terahercų spinduliuotę į vieną tašką ant bandinio ir išmatuoti detektoriaus atsaką. Paėmus daug taškų gaunamas objekto atvaizdavimas terahercine spinduliuote.

Eksperimento rezultatai parodė THz spinduliuotės perėjimo per tam tikras medžiagas ypatybes.

Summary

Vaidotas Kumža

THz imaging using low-NEP high-resonance detector

Terahertz radiation has some unique properties. THz is not ionizing, can go through most dielectric materials. Water absorbs it extremely well and metals reflect it good too. These are the reasons why THz has many applications.

Purpose of this work is to make a THz imaging system using stepper motors and silicon CMOS detector and to image few chosen objects.

In order to change the position of object precisely, a stepper motor is needed. Connecting two of these stepper motors together lets us to change the position of object in a plane. But in order to be able to control stepper motor, we also need a microcontroller and driver. To make a compactible driver we need to know stepper motor characteristics.

Before we get THz imaging, we need to focus terahertz radiation to a point on our object and read the response of detector. Measuring response in many points allows us to get THz imaging.

Experiment showed some features of THz radiation transition through selected materials.

Literatūra

- [1] W. Knap, J. Łusakowski, T. Parenty, S. Bollaert, A. Cappy, and M. S. Shur, Appl. Phys. Lett. 84, 2331 (2004).
- [2] M. I. Dyakonov and M. S. Shur, IEEE Transactions on Electron Devices 43, 380 (1996).
- [3] W. Knap, J. Łusakowski, T. Parenty, S. Bollaert, A. Cappy, and M. S. Shur, Appl. Phys. Lett. 84, 2331 (2004).
- [4] W. Knap, F. Teppe, Y. Meziani, N. Dyakonova, J. Łusakowski, F. Boeuf, T. Skotnicki, D. Maude, S. Rumyantsev, and M.S. Shur, Appl. Phys. Lett. 85, 675 (2004).
- [5] Lewis R.A.: A review of terahertz sources, Journal of Physics D: Applied Physics, 2014.
- [6] Yu C., Fan S., Sun Y., & Pickwell-MacPherson E.: The potential of terahertz imaging for cancer diagnosis: A review of investigations to date. Quantitative Imaging in Medicine and Surgery, 2012.
- [7] Leiss-Holzinger E., Wiesauer K., Stephani H., Heise B., Stifter D., Kriechbaumer B., Spachinger S.J., Gusenbauer C., Withalm G.: Imaging of the inner structure of cave bear teeth by novel non-destructive techniques, Palaeontologia electronica, 2015, vol. 18.
- [8] Freebody M.: Challenges Not Insurmountable for Terahertz Spectroscopy, <http://www.photonics.com>.
- [9] Walther M., Fischer B. M., Ortner A., Bitzer A., Thoman A., Helm H., Chemical sensing and imaging with pulsed terahertz radiation, Analytical and Bioanalytical Chemistry, 2010, vol. 397.
- [10] Zomega Terahertz Corporation, The Terahertz Wave eBook June 2012 Edition, <http://dl.z-thz.com>.
- [11] <http://www.ti.com/lit/ds/symlink/tm4c123gh6pm.pdf>
- [12] Alvydas Lisauskas, Terahercų elektronika.pptx

- [13] W. Knap, M. Dyakonov, D. Coquillat, F. Teppe, N. Dyakonova, J. Łusakowski, K. Karpierz, M. Sakowicz, G. Valusis, D. Seliuta, I. Kasalynas, A. El Fatimy, Y. Meziani, T. Otsuji
„Field Effect Transistors for Terahertz Detection: Physics and First Imaging Applications“
- [14] Introduction to Step Motors.pdf
- [15] B. L. Thareja and A. K. Thareja, Electrical Technology, S. Chand & Sons., 23rd Edition, Chapter 39
- [16] Mark McComb, „Introduction to Stepper Motors“
- [17] Industrial Circuits Application Note, Stepper Motor Basics.pdf
- [18] A3967 datasheet.pdf

Priedas

Mikrovaldiklio programos kodas

```
#include <stdint.h>
#include <stdio.h>
#include <stdbool.h>
#include "inc/tm4c123gh6pm.h"
#include "inc/hw_ints.h"
#include "inc/hw_memmap.h"
#include "inc/hw_types.h"
#include "driverlib/gpio.h"
#include "driverlib/interrupt.h"
#include "driverlib/pin_map.h"
#include "driverlib/sysctl.h"
#include "driverlib/uart.h"
#include "driverlib/timer.h"
#include "examples/boards/ek-tm4c123gx1/drivers/buttons.h"

unsigned char ucDelta = 0;
unsigned char ucState = 0;
int statusLeft;
int statusRight;
int N = 500;
int delay = 20000;

void XHomeLeft(void)
{
    delay = 20000;
    GPIOPinWrite(GPIO_PORTD_BASE, GPIO_PIN_1, 0); //
direction
    while(GPIOPinRead(GPIO_PORTE_BASE, GPIO_PIN_2) == 0)
    {
        GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_3,
GPIO_PIN_3); // Green LED
        GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_4,
GPIO_PIN_4); // Step
        SysCtlDelay(delay);
        GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_3, 0);
        GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_4, 0);
        SysCtlDelay(delay);
        if( delay > 4000) {delay--; delay--;}
    }
    if(GPIOPinRead(GPIO_PORTE_BASE, GPIO_PIN_2) == GPIO_PIN_2)
UARTCharPut(UART0_BASE, 'L');
}

void XHomeRight(void) //blue
{
    delay = 20000;
    GPIOPinWrite(GPIO_PORTD_BASE, GPIO_PIN_1, GPIO_PIN_1); //
direction
    while(GPIOPinRead(GPIO_PORTE_BASE, GPIO_PIN_1) == 0)
    {
```

```

        GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_2, GPIO_PIN_2);
        // Blue LED
        GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_4, GPIO_PIN_4);
        // Step
        SysCtlDelay(delay);
        GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_2, 0);
        GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_4, 0);
        SysCtlDelay(delay);
        if( delay > 4000) {delay--; delay--;}
    }
    if(GPIOPinRead(GPIO_PORTE_BASE, GPIO_PIN_1) == GPIO_PIN_1)
UARTCharPut(UART0_BASE, 'P');
}

void YHomeDown(void) //white
{
    delay = 20000;
    GPIOPinWrite(GPIO_PORTB_BASE, GPIO_PIN_7, 0);
// direction
    while(GPIOPinRead(GPIO_PORTE_BASE, GPIO_PIN_4) == 0)
    {
        GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_2 | GPIO_PIN_3
|GPIO_PIN_1, GPIO_PIN_2 | GPIO_PIN_3 |GPIO_PIN_1); // White LED
        GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_5, GPIO_PIN_5);

        // Step
        SysCtlDelay(delay);
        GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_2 | GPIO_PIN_3
|GPIO_PIN_1, 0);
        GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_5, 0);
        SysCtlDelay(delay);
        if( delay > 4000) {delay--; delay--;}
    }
    if(GPIOPinRead(GPIO_PORTE_BASE, GPIO_PIN_4) == GPIO_PIN_4)
UARTCharPut(UART0_BASE, 'K');
}

void YHomeUp(void) //red
{
    delay = 20000;
    GPIOPinWrite(GPIO_PORTB_BASE, GPIO_PIN_7, GPIO_PIN_7); //
direction
    while(GPIOPinRead(GPIO_PORTE_BASE, GPIO_PIN_5) == 0)
    {
        GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1, GPIO_PIN_1);
        // Red LED
        GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_5, GPIO_PIN_5);

// Step
        SysCtlDelay(delay);
        GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1, 0);
        GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_5, 0);
        SysCtlDelay(delay);
        if( delay > 4000) {delay--; delay--;}
    }
    if(GPIOPinRead(GPIO_PORTE_BASE, GPIO_PIN_5) == GPIO_PIN_5)
UARTCharPut(UART0_BASE, 'O');
}

```

```

void StepToTheLeft(void)    //green
{
    GPIOWrite(GPIO_PORTD_BASE, GPIO_PIN_1, 0);           // direction
    int i;
    for(i=1; i <= N; i++)
    {
        if(GPIOPinRead(GPIO_PORTE_BASE, GPIO_PIN_2) == 0)
        {
            GPIOWrite(GPIO_PORTF_BASE, GPIO_PIN_3, GPIO_PIN_3); // Green
            GPIOWrite(GPIO_PORTC_BASE, GPIO_PIN_4, GPIO_PIN_4); // Step
            SysCtlDelay(20000);
            GPIOWrite(GPIO_PORTF_BASE, GPIO_PIN_3, 0);
            GPIOWrite(GPIO_PORTC_BASE, GPIO_PIN_4, 0);
            SysCtlDelay(20000);
            if(i == N) {UARTCharPut(UART0_BASE, 'a'); UARTCharPut(UART0_BASE,
'+' );}
        }
    }

    if(GPIOPinRead(GPIO_PORTE_BASE, GPIO_PIN_2) == GPIO_PIN_2)
{UARTCharPut(UART0_BASE, 'a'); UARTCharPut(UART0_BASE, '-');}

}

void StepToTheRight(void) //blue
{
    GPIOWrite(GPIO_PORTD_BASE, GPIO_PIN_1, GPIO_PIN_1);           // direction
    int i;
    for(i=1; i<=N; i++)
    {
        if(GPIOPinRead(GPIO_PORTE_BASE, GPIO_PIN_1) == 0)
        {
            GPIOWrite(GPIO_PORTF_BASE, GPIO_PIN_2, GPIO_PIN_2); // Blue LED
            GPIOWrite(GPIO_PORTC_BASE, GPIO_PIN_4, GPIO_PIN_4); // Step
            SysCtlDelay(20000);
            GPIOWrite(GPIO_PORTF_BASE, GPIO_PIN_2, 0);
            GPIOWrite(GPIO_PORTC_BASE, GPIO_PIN_4, 0);
            SysCtlDelay(20000);
            if(i == N) {UARTCharPut(UART0_BASE, 'd'); UARTCharPut(UART0_BASE,
'+' );}
        }
    }

    if(GPIOPinRead(GPIO_PORTE_BASE, GPIO_PIN_1) == GPIO_PIN_1)
{UARTCharPut(UART0_BASE, 'd'); UARTCharPut(UART0_BASE, '-');}

}

void StepDown(void) //white
{
    GPIOWrite(GPIO_PORTB_BASE, GPIO_PIN_7, 0);
    // direction
    int i;
    for(i=1; i<=N; i++)
    {

```

```

        if(GPIOPinRead(GPIO_PORTE_BASE, GPIO_PIN_4) == 0)
        {
            GPIOPinWrite(GPIO_PORTF_BASE,      GPIO_PIN_2      |      GPIO_PIN_3
|GPIO_PIN_1, GPIO_PIN_2 | GPIO_PIN_3 |GPIO_PIN_1); // White LED
            GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_5, GPIO_PIN_5);

                                                    // Step
            SysCtlDelay(15000);
            GPIOPinWrite(GPIO_PORTF_BASE,      GPIO_PIN_2      |      GPIO_PIN_3
|GPIO_PIN_1, 0);
            GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_5, 0);
            SysCtlDelay(15000);
            if(i == N) {UARTCharPut(UART0_BASE, 's'); UARTCharPut(UART0_BASE,
'+');}
        }
    }
    if(GPIOPinRead(GPIO_PORTE_BASE,      GPIO_PIN_4)      ==      GPIO_PIN_4)
{UARTCharPut(UART0_BASE, 's'); UARTCharPut(UART0_BASE, '-');}
}

void StepUp(void) //red
{
    GPIOPinWrite(GPIO_PORTB_BASE, GPIO_PIN_7, GPIO_PIN_7); // direction
    int i;
    for(i=1; i<=N; i++)
    {
        if(GPIOPinRead(GPIO_PORTE_BASE, GPIO_PIN_5) == 0)
        {
            GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1, GPIO_PIN_1); // Red LED
            GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_5, GPIO_PIN_5); // Step
            SysCtlDelay(15000);
            GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1, 0);
            GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_5, 0);
            SysCtlDelay(15000);
            if(i == N) {UARTCharPut(UART0_BASE, 'w'); UARTCharPut(UART0_BASE,
'+');}
        }
    }
    if(GPIOPinRead(GPIO_PORTE_BASE,      GPIO_PIN_5)      ==      GPIO_PIN_5)
{UARTCharPut(UART0_BASE, 'w'); UARTCharPut(UART0_BASE, '-');}
}

void UARTIntHandler(void)
{
    uint32_t ui32Status;
    ui32Status = UARTIntStatus(UART0_BASE, true);

    UARTIntClear(UART0_BASE, ui32Status);

    while(UARTCharsAvail(UART0_BASE))
    {
        int symbol = UARTCharGetNonBlocking(UART0_BASE);

        if(symbol == 0x61) StepToTheLeft(); //a
        if(symbol == 0x64) StepToTheRight(); //d
        if(symbol == 0x73) StepDown(); //s
        if(symbol == 0x77) StepUp(); //w
        if(symbol == 108) XHomeLeft(); //1
    }
}

```

```

        if(symbol == 0x70)      XHomeRight();           //p
        if(symbol == 111)      YHomeUp();              //o
        if(symbol == 107)      YHomeDown();            //k
    }

}

int main(void) {

    SysCtlClockSet(SYSCTL_SYSDIV_4 | SYSCTL_USE_PLL | SYSCTL_OSC_MAIN |
SYSCTL_XTAL_16MHZ);

    SysCtlPeripheralEnable(SYSCTL_PERIPH_UART0);
    SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOA);
    SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOF);
    SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOD);
    SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOC);
    SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOE);
    SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOB);

    IntMasterEnable();

    GPIOPinConfigure(GPIO_PA0_U0RX);
    GPIOPinConfigure(GPIO_PA1_U0TX);
    GPIOPinTypeUART(GPIO_PORTA_BASE, GPIO_PIN_0 | GPIO_PIN_1);

    GPIOPinTypeGPIOOutput(GPIO_PORTF_BASE, GPIO_PIN_3 | GPIO_INT_PIN_2 |
GPIO_PIN_1);
    GPIOPinTypeGPIOOutput(GPIO_PORTD_BASE, GPIO_PIN_0 | GPIO_INT_PIN_1 |
GPIO_PIN_2 | GPIO_PIN_3);
    GPIOPinTypeGPIOOutput(GPIO_PORTC_BASE, GPIO_PIN_4 | GPIO_INT_PIN_5 |
GPIO_PIN_6 | GPIO_PIN_7);
    GPIOPinTypeGPIOOutput(GPIO_PORTB_BASE, GPIO_PIN_5 | GPIO_PIN_7);

    GPIOPinTypeGPIOInput(GPIO_PORTF_BASE, GPIO_PIN_4 | GPIO_PIN_0);
    GPIOPinTypeGPIOInput(GPIO_PORTE_BASE, GPIO_PIN_1 | GPIO_PIN_2 | GPIO_PIN_4 |
GPIO_PIN_5);

    GPIOPadConfigSet(GPIO_PORTE_BASE, GPIO_PIN_1 | GPIO_PIN_2 | GPIO_PIN_4 |
GPIO_PIN_5, GPIO_STRENGTH_12MA, GPIO_PIN_TYPE_STD_WPU);

    ButtonsInit();

    UARTConfigSetExpClk(UART0_BASE, SysCtlClockGet(), 115200,
(UART_CONFIG_WLEN_8 | UART_CONFIG_STOP_ONE | UART_CONFIG_PAR_NONE));

    IntMasterEnable(); //enable processor interrupts
    IntEnable(INT_UART0); //enable the UART interrupt
    UARTIntEnable(UART0_BASE, UART_INT_RX | UART_INT_RT); //only enable RX and
TX interrupts

    UARTCharPut(UART0_BASE, 'E');
    UARTCharPut(UART0_BASE, 'n');
    UARTCharPut(UART0_BASE, 't');
    UARTCharPut(UART0_BASE, 'e');
    UARTCharPut(UART0_BASE, 'r');
    UARTCharPut(UART0_BASE, ' ');

```

```
UARTCharPut(UART0_BASE, 'T');
UARTCharPut(UART0_BASE, 'e');
UARTCharPut(UART0_BASE, 'x');
UARTCharPut(UART0_BASE, 't');
UARTCharPut(UART0_BASE, ':');
UARTCharPut(UART0_BASE, ' ');

GPIOPinWrite(GPIO_PORTD_BASE, GPIO_PIN_0, 0);           // enable #1
GPIOPinWrite(GPIO_PORTB_BASE, GPIO_PIN_5, 0);           // enable #2

while(1)
{
}
}
```

Labview programos blokinė diagrama

