****

**Kauno technologijos universitetas**

Elektros ir elektronikos fakultetas

Individualus darbas

**Garso signalų įrašymas ir filtravimas**

Skaitmeninis signalų apdorojimas realaus laiko sistemose (T121M001)

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

**Kaunas, 2023**

Turinys

[Paveikslų sąrašas 3](#_Toc153019610)

[Santrumpų ir terminų sąrašas 4](#_Toc153019611)

[Įvadas 5](#_Toc153019612)

[1. Literatūros analizė 6](#_Toc153019613)

[2. Kuriamos sistemos koncepcija 11](#_Toc153019614)

[3. Sistemos realizavimas 13](#_Toc153019615)

[Išvados 20](#_Toc153019616)

[Literatūros sąrašas 21](#_Toc153019617)

Paveikslų sąrašas

[1 pav. Miego apnėjos vizualizacija 6](#_Toc153019618)

[2 pav. Įvairių elektrodų ir jutiklių, naudojamų miegui stebėti polisomnografijos metu 7](#_Toc153019619)

[3 pav. Mokslininkų sukurtos sistemos struktūrinė schema [7] 8](#_Toc153019620)

[4 pav. Mokslininkų sukurtos sistemos apnėjos detektavimui maketas 9](#_Toc153019621)

[5 pav. Mokslininkų sukurtos sistemos akselometro pagrindu schema [9] 10](#_Toc153019622)

[6 pav. „BlueCoin“ maketo mikrofono ir kitais sensoriais 11](#_Toc153019623)

[7 pav. „BlueCoin“ maketo dalis su SD kortele 12](#_Toc153019624)

[8 pav. BlueCoin prietaiso blokinė diagrama 12](#_Toc153019625)

[9 pav. Pagrindinis programos algoritmas 13](#_Toc153019626)

[10 pav. Standartinis WAVE formatą 14](#_Toc153019627)

[11 pav. Sukurtas lowpass filtras su MATLAB filter designer programa 18](#_Toc153019628)

[12 pav. Gauti audio įrašai iš sistemos SD kortelės 19](#_Toc153019629)

Santrumpų ir terminų sąrašas

**Santrumpos:**

PSG – polisomnografija (angl. P*olysomnography*);

OMA – obstrukcinė miego apnėja;

MEMS – Mikroelektromechaninė sistema (angl. *Micro-Electro-Mechanical System*).

Įvadas

Miegas yra esminis geros savijautos elementas, tačiau gerai išsimiegoti pavyksta nevisiems. Miego sutrikimai, tokie kaip miego trūkumas ir nemiga, veikia mūsų imunitetą, širdies ir kraujagyslių sistemą, gali sukeli vėžį ir yra galima depresijos atsiradimo priežastis [1]. Viena iš dažniausiai pasitaikančių miego sutrikimo formų yra obstrukcinė miego apnėja, kurią patiria maždaug 5-15% populiacijos [2].

Miego apnėjos diagnostikos problema susijusi ne tik su brangia ir nepatogi polisomnografijos procedūra, bet ir su dideliu nediagnozuotų atvejų skaičiumi. Remiantis moksliniais tyrimais, apie 80% obstrukcinės miego apnėjos atvejų lieka nediagnozuota, o tai reiškia, kad pacientai negauna būtinos gydymo ir rizikuoja susirgti kitomis sveikatos problemomis [3]. Polisomnografija (angl. *polysomnography – PSG*) yra labiausiai paplitęs metodas apnėjos diagnostikai. Atliekant šį tyrimą, pacientas guldomas per naktį miego laboratorijoje kur yra stebimi jo fiziologiniai kintamieji [4]. PSG tyrimo metu registruojama kūno padėtis, galūnių judesiai, prisotinimas deguonimi, širdies ritmas ir dažnis, kvėpavimo pastangos, smegenų veikla, akių judesiai ir miego fazės [4]. Nors PSG tyrimas gali pateikti tikslią diagnozę, jis yra brangus medicinos tyrimas ir sukelia nepatogumų pacientams.

Būtent todėl modernios sistemos, galinčios surinkti miego kokybės duomenis, o vėliau naudojant dirbtinio intelekto technologijas, analizuoti juos, gali būti inovatyvus žingsnis šioje srityje. Šios sistemos gali ne tik efektyviai nustatyti miego apnėjos požymius, bet ir sumažinti diagnozavimo ir stebėjimo sąnaudas, kurių kitu atveju reikalauja tradicinės polisomnografijos procedūros. Naujos, neinvazinės, sistemos naudojimas taip pat gali sumažinti laiką, per kurį galima nustatyti miego sutrikimus, prisidedant prie ankstyvo gydymo ir mažinant galimas sveikatos rizikas. Be to, šios technologijos gali suteikti galimybę pacientams stebėti savo miego kokybę namuose, neatsitraukiant nuo įprasto gyvenimo ritmo. Būtent dėl šios priežasties šiame kursiniame darbe yra pasiūloma apnėjos detektavimo sistemos dalis kuri yra atsakinga už garsų miego metu įrašymą ir jų išsaugojimą tolesniam apdorojimui.

**Darbo tikslas**: sukurti garso signalų įrašymas ir filtravimas sistemos maketą

**Darbo uždaviniai**:

1. Išanalizuoti rinkoje esančius sprendimus.
2. Suprojektuoti bei sukurti garsus gebančią įrašyti sistemą.
3. Atlikti audio signalų filtravimą.

# Literatūros analizė

## Miego apnėja

Obstrukcinė miego apnėja yra kvėpavimo sistemos sutrikimas, kuris turi įtakos kvėpavimui miegant. Šį sutrikimą sukelia pasikartojantys viršutinių kvėpavimo takų obstrukcijos (užsivėrimo) kurios trunka mažiausiai 10 sekundžių, bet gali trukti iki minutės [5]. Tai gali atsirasti dėl kelių priežasčių, įskaitant raumenų, palaikančių gerklės ir liežuvio minkštuosius audinius, atsipalaidavimą, dėl kurio šie audiniai gali žlugti ir užkimšti kvėpavimo takus. Taip pat, riebalinių audinių perteklius gerklėje ir liežuvyje, padidėjusios tonzilės, didelis liežuvis ar mažas žandikaulis gali prisidėti prie kvėpavimo takų obstrukcijos miego metu. Kiti veiksniai, galintys padidinti kvėpavimo takų kolapso riziką, yra nutukimas, alkoholio vartojimas, rūkymas ir miegas ant nugaros. Šio sutrikimo liustracija vizualiai pateikiama paveikslėlyje (1 pav.).

A picture containing screenshot

Description automatically generated

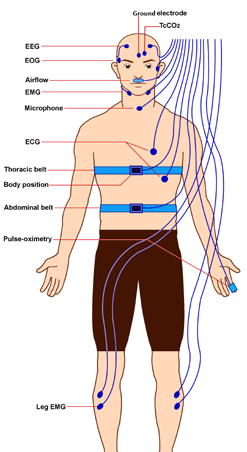
1 pav. Miego apnėjos vizualizacija[[1]](#footnote-2)

## Esantys sprendimai

### Polisomnografija

Dabartinis auksinis obstrukcinės miego apnėjos diagnozavimo standartas yra naktinė daugiakanalė polisomnografija (PSG), brangi, daug darbo reikalaujanti ir nepatogi procedūra. Detaliau polisomnograma yra procedūra, kurios metu naudojama elektroencefalograma, elektrookulograma, elektromiograma, elektrokardiograma, pulso oksimetrija, matuojamas oro srautas ir kvėpavimo pastangos, siekiant įvertinti pagrindines miego sutrikimų priežastis [6]. Įprastam PSG reikalinga visapusiška stebėjimo sistema, leidžianti registruoti miego stadijas, galūnių judesius, oro srautą, kvėpavimo pastangas, širdies susitraukimų dažnį ir ritmą, deguonies prisotinimą ir kūno padėtį. Šio tipo tyrimai atliekami miego laboratorijoje, kurioje visą tyrimo laiką dalyvauja apmokytas miego technikas. PSG pirmiausia naudojami diagnozuoti su miegu susijusius kvėpavimo sutrikimus, įskaitant OMA, centrinę miego apnėją ir su miegu susijusią hipoventiliaciją / hipoksiją. Vis dėlto miego apnėja yra labiausiai paplitęs miego sutrikimas, dėl kurio diagnostiniai tyrimai atliekami miego centruose.

Kita šio tyrimo problema yra tai, kad jis yra labai brangus ir sudėtingas diagnostikos metodas, reikalaujantis specializuotos įrangos ir medicinos personalo. Lietuvoje yra mažai PSG tyrimų centų, o klinikų, turinčių PSG įrangą, yra tik kelios. Dėl to, žmonės dažnai turi laukti ilgą laiką, kol jie gali gauti polisomnografijos tyrimą. Dažniausiai tyrimo laukimo laikas yra kelios savaitės ar net iki keleto mėnesių. Toks ilgas laukimo laikas gali būti ypač problemiškas pacientams, kuriems yra įtariama miego apnėja, kadangi jie jų būklė gali dar pablogėti. Tačiau net ir atlikus PSG tyrimą dažnu atveju liga nėra diagnupzojama, dėl neįprastų asmeniui miego sąlygų. Šis stresas dėl pasikeitusios aplinkos lemią kitokį miego režimą. To būtų galima išvengti atliekant tyrimus namuose arba kitais pažangesniais neinvaziniais metodais. Būtent tai ir yra pagrindinė šiame darbe siūlomos sistemos panaudojimo motyvacija.



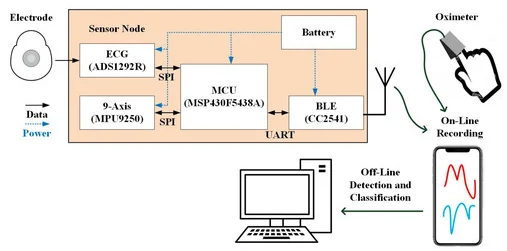
2 pav. Įvairių elektrodų ir jutiklių, naudojamų miegui stebėti polisomnografijos metu

### Naujoviškų sistemų sprendimai

Analizuojant kitų mokslininkų darbus galima pastebėti jog tobulėjant mašininio mokymosi metodams bei augant įterptinių sistemų skaičiavimų resursams vis atsiranda vis daugiau darbų tiriančių šio ligos aptikimą intymiais metodais.

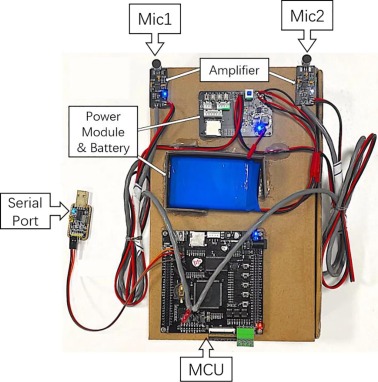
Kaip viena iš pavyzdžių galima būtų pateigti 2020 metų Taivano ir Amekiriečių mokslininkų darbą „Nešiojamojo miego apnėjos sindromo stebėjimas ir įvykių aptikimas naudojant ilgalaikę trumpalaikę atmintį pasikartojantį neuronų tinklą“ [7]. Šiame tyrime mokslininkai sprendė apnėjos aptikimo problemą sukurdami išsamų matavimo modulį. Šiame modulyje integruoti pilvo ir krūtinės ląstos triašiai akselerometrai, pulso oksimetras (SpO2) ir elektrokardiogramos (EKG) jutiklis. Norint užfiksuoti krūtinės ir pilvo judesius, buvo naudojamos pjezoelektrinės juostos, o signalai buvo registruojami 100 Hz diskretizavimo dažniu. Mokslininkai kombinavo aparatinės įrangos sprendimą kartu su naujoviška neuroninio tinklo klasifikavimo technika, kad nustatytų obstrukcinę miego apnėją.

Kad būtų galima nuolat stebėti, buvo sukurtas 27 gramų jutiklis, turintis devynių ašių akselerometrą, EKG jutiklį, Bluetooth modulį ir mikrovaldiklį. Itin mažos galios mikrovaldiklis (MSP430) valdė MPU9250 akselerometrą, kad užfiksuotų signalus, kurie per Bluetooth modulį (CC2541) buvo perduodami į mobilųjį įrenginį (išmanųjį telefoną ar planšetinį kompiuterį). EKG ir akselerometro signalo žodžio ilgis ir diskretizavimo dažnis buvo atitinkamai 12 bitų/500 Hz ir 16 bitų/50 Hz. Perdavimo sparta iš jutiklio įrenginio į iOS įrenginį buvo nustatyta 115 200 bps. Šis išsamus požiūris ne tik nagrinėjo techninius signalų gavimo aspektus, bet ir suteikė praktišką bei efektyvų sprendimą ilgalaikiam su miegu susijusių fiziologinių parametrų stebėjimui.



3 pav. Mokslininkų sukurtos sistemos struktūrinė schema [7]

Naudojantis konvoliucinių ir ilgos-trumpos atminties neuroninių tinklų kombinacija Xu Lin 2022 metų darbe pasiūlė neinvazinę apnėjos detektavimo sistemą [8]. Šiame darbe mokslininkai naudojo garsus surinktus iš 32 pacientų ir sugebėjo pasiekti 73,92 procentų tikslumą naudojant keturių klasių klasifikavimą. Knarkimo garsai buvo įrašomi su *Sony PCM A100* įrenginiu naudojant du mikrofonus. Įrašymo prietaisai buvo padėti vieno metro atstumu nuo pacientų. Garso duomenų fiksavimo dažnis buvo 44,1 kHz, o įrašytas garsas buvo išsaugotas kaip dviejų kanalų *wav* failas. Tyrime buvo naudojamos Melo filtrų banku išskirti požymiai iš knarkimo garsų. Būtent Melo filtrai buvo pasirinkti remiantis žmogaus klausos sistemos netiesinio garso dažnio suvokimo ypatybėmis. Būtent naudojant šiuos Melų filtrus tyrėjai gavo 16x16 pikselių dydžio nuotraukas, kurias naudojo kaip neuroninių tinklų įvestis. Mokslininkai sukurtą modelį testavo su *STM32 ARM* procesoriaus tipo realaus laiko įterptine sistema. Dėl įterptinių sistemų resursų ribotumo tinklo parametrų skaičius buvo sumažintas 40,27 % tuo tarpu tikslumas sumažėjo tik 0,35 %. Tai parodo potencialą sukurti išmaniuosius įrenginius, naudojamus namuose, šios ligos prevencijos tikslams

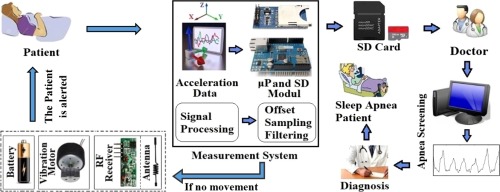


4 pav. Mokslininkų sukurtos sistemos apnėjos detektavimui maketas

Kaip kitą mokslininkų naudojančių įterptine sistema paremtą apnėjos aptikimo sistemą, galima pateikti A.H. Yüzer „Nauja nešiojama realaus laiko miego apnėjos aptikimo sistema, pagrįsta pagreičio jutikliu“ darbą sutelktą aplink pagreičio jutiklį [9]. Pagrindinis tikslas buvo greitai nustatyti apnėjos reiškinį ir atitinkamai įspėti pacientą. Pagreičio jutiklis, esantis ant diafragmos, nuolat registruoja diafragmos judesius, naudodamas ADXL345, MEMS pagrįstą 3 ašių akselerometrą, kurio matavimo diapazonas yra ±2 g. Šio jutiklio skiriamoji geba yra 13 bitų.

Per akselerometru pagrįstą sistemą aptikus apnėjos įvykį, signalas perduodamas į apyrankę. Apyrankė su vibraciniu varikliu inicijuoja vibraciją tol, kol paciento atnaujins normalų kvėpavimą. Pažymėtina, kad vibracijos variklio intensyvumą galima pritaikyti atsižvelgiant į individualius paciento veiksnius, tokius kaip miego trūkumas, ypač aprūpinant senyvus, neįgalius ar vaikus. Pagrindinės šios sistemos yra jos gebėjimas pasiekti ir saugoti atitinkamus kvėpavimo parametrus, nereikalaujant tam skirto miego kambario. Nuolatiniai diafragmos pagreičio, imami 20 Hz dažniu, yra saugomi SD kortelėje arba per USB perkeliami į kompiuterį. Kiekvienas duomenų rinkinys sudaro iš trijų ašių duomenų, todėl duomenų perdavimo sparta yra 1920 bitų per sekundę.

Siekiant patvirtinti sistemos veiksmingumą, miego ekspertai ligoninės aplinkoje tiksliai pažymėjo tikrus miego apnėjos atveju. Siūloma realaus laiko miego apnėjos aptikimo sistema įrodė savo veiksmingumą atlikdama našumo analizę, suderinama su ekspertų pažymėtinais miego apnėjos atvejais. Šis naujoviškas sprendimas ne tik pristato naują požiūrį į apnėjos apkrovą, bet ir patenkina poreikius, todėl tai yra perspektyvi miego stebėjimo technologijos plėtra.



5 pav. Mokslininkų sukurtos sistemos akselometro pagrindu schema [9]

Taigi atlikus literatūros analizę matyti, kad yra paklausa sukurti išmaniėją apnėjos detektavimo sistema, bei mikrofono ir garsais paremta sistema yra vienas iš variantų.

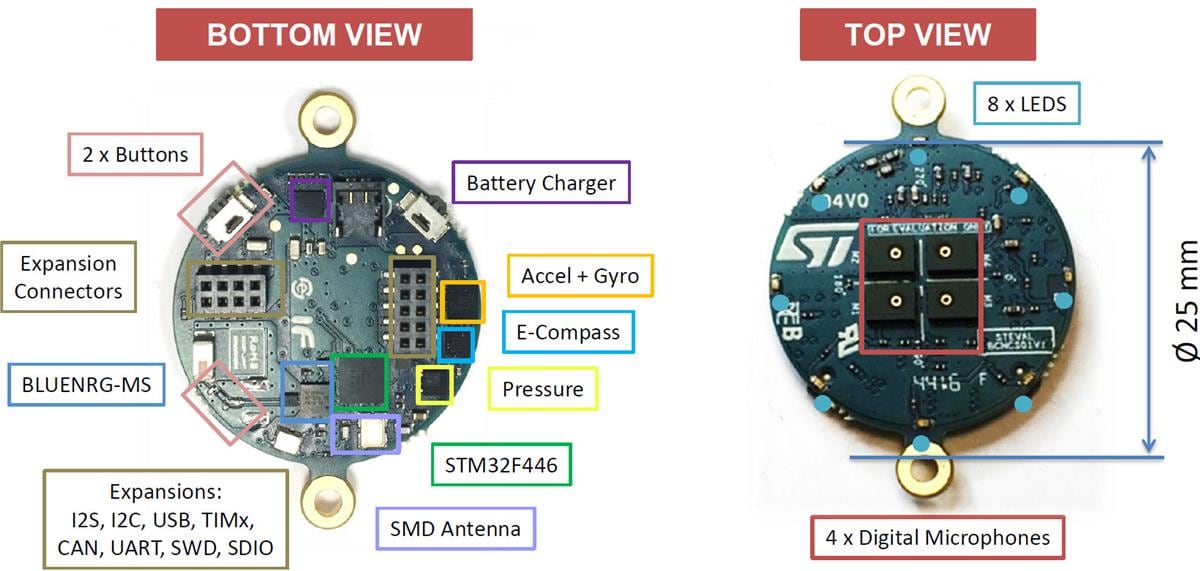
Taip pat duomenų perdavimas per blueteuht ar saugojimas į SD kortelę yra labai svarbi sistems dalsi kadangi tolimesnis apdorojimas tikriausiai bus atlikatas mokslinikų arba dirbtinio intelekto, kadangi įterpinės sistemos negali palaikyti mašininiam mokymuisi reikalingų skaičiavimo resursų.

# Kuriamos sistemos koncepcija

Šiame darbe yra siūloma apnėjos aptikimo sistemos dalis kuri gebėtų kokybiškai įrašyti garsus miego metu, kurie toliau jau bus apdorojami ir klasifikuojami medikų ar inotavių mašininių mokymosi pagrįstų sistemų ligos aptikimui ir jos stadijos klasifikavimui. Todėl norint sukurti tokį prietaisą svarbu pasirinkti tinkamą apartinę įrangą garso įrašymui ir jo saugojimui.

## STM32 „Bluecoin“ prietaisas

Taigi, kadangi norit sukurti garso analizės sistema reikalingas mikrofonas tai yra viena svarbiausių projekto dalių. Dėl greitesnės integracijos buvo pasirinktas STM32 „BlueCoin Starter kit“ prietaisas.

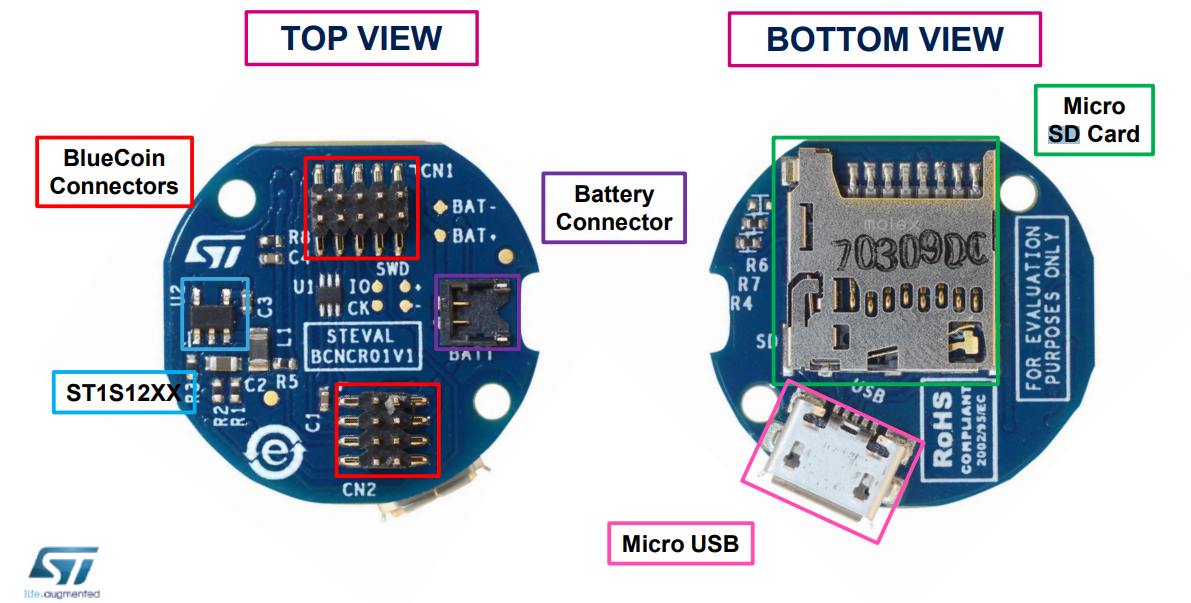


6 pav. „BlueCoin“ maketo mikrofono ir kitais sensoriais

Ši integruota testavimo ir prototipų kūrimo platforma, skirta išplėstiniam akustiniam ir judesio aptikimui daiktų interneto programoms, pagrįsta žmogaus ausies klausymosi gebėjimais. „BlueCoin“ leidžia tyrinėti pažangius jutiklius ir jų signalų apdorojimo funkcijas. Įrenginys skirtas robotikos ir automatikos programoms, naudojant 4 skaitmeninius MEMS mikrofonų rinkinius, didelio našumo 9 ašių inercinį ir aplinkos jutiklių bloką bei veikimo laiko nuotolio jutiklius.

Didelio našumo STM32F446 180 MHz CPU leidžia realiuoju laiku įgyvendinti labai pažangius jutiklių algoritmus, tokius kaip adaptyvus pluošto formavimas (angl. *adaptive beamforming*) kuris pagerina garso šaltinio lokalizavimą. Šias funkcijas naudoti yra paruoštos nemokamos bibliotekos. „BlueCoin“ taip pat turi integruotą BLE sąsają su kuria gali prisijungti per bet kurio daiktų interneto ir išmaniojo pramonės belaidžio jutiklio BlueThooth tinklo. Taip pat, naudojant šį maketą galimas neapdorotų jutiklių duomenų srautinio perdavimas per USB sąsają, duomenų registravimas į SD kortelę ar garso perdavimas.

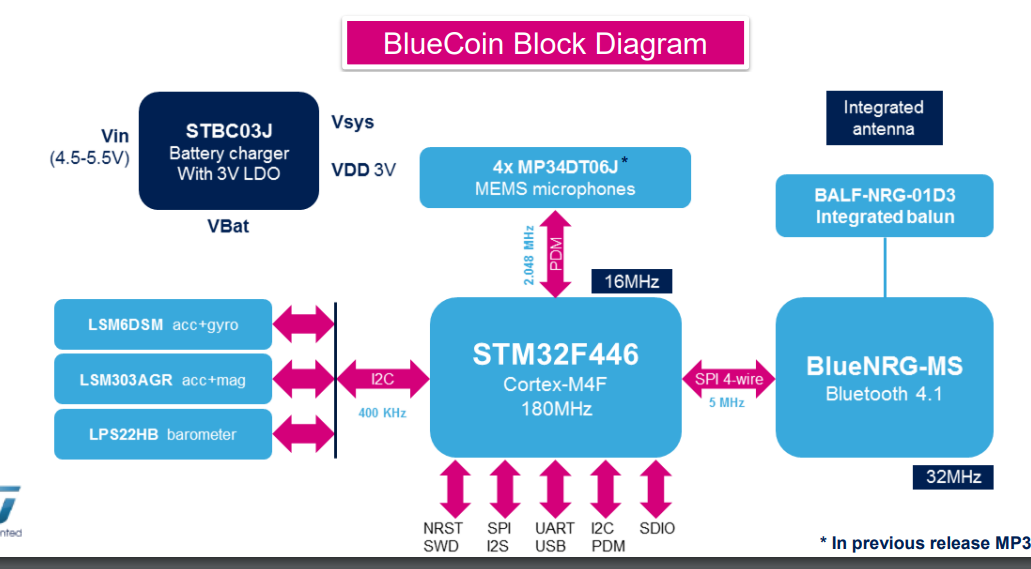
Būtent šios funkcijos ir yra reikalingos norint sukurti apnėjos detekcijos sistemą. Kadangi dažniausiai miegame ne vieni, todėl yra reikalinga atstumo kalibravimą nuo pasirinkto šaltinio lokacijos kad galima būtų identifikuoti specifinį pacientą. Taip pat SD kortelė jau yra integruota šiame įrenginyje todėl nereikia papildomos sąsajos ir prietaiso norint duomenys saugoti SD laikmenoje.



7 pav. „BlueCoin“ maketo dalis su SD kortele

## Strukturinė schema

Sistemą taip pat sudar akselerometrai, BLE ar magnetometrai, tačiau šiame darbe jie nėra naudojami. Pagrindiniai darbe naudojami MP34DT06J mikfonai duomenis perduoda per PDM sąsają (8 pav.). MP34DT06J yra itin kompaktiškas, mažos galios, įvairiakryptis skaitmeninis MEMS mikrofonas, pagamintas su talpiniu jutimo elementu ir IC sąsaja. Jutimo elementas, galintis aptikti akustines bangas, pagamintas naudojant specialų silicio mikroapdirbimo procesą, skirtą garso jutiklių gamybai. Pagrindiniai skaitmeninio mikrofono parametrai yra: akustinės perkrovos taškas 122,5 dBSPL, 64 dB signalo ir triukšmo santykis ir –26 dBFS ± 1 dB jautrumas.

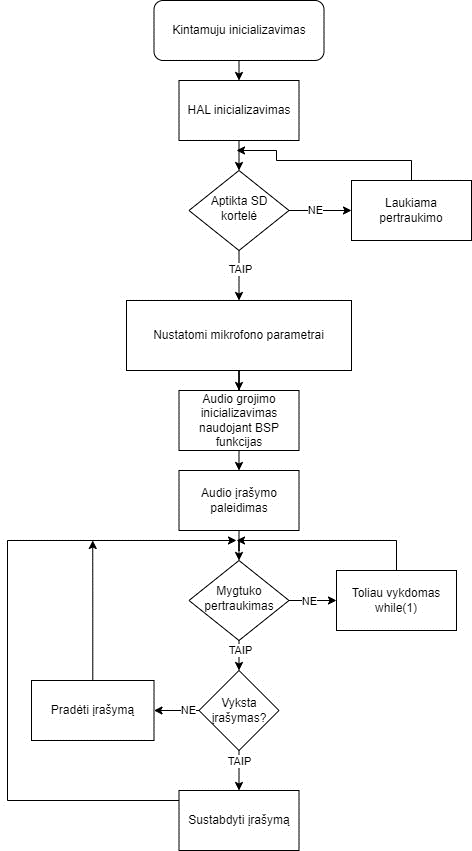


8 pav. BlueCoin prietaiso blokinė diagrama

# Sistemos realizavimas

## Programos algoritmas

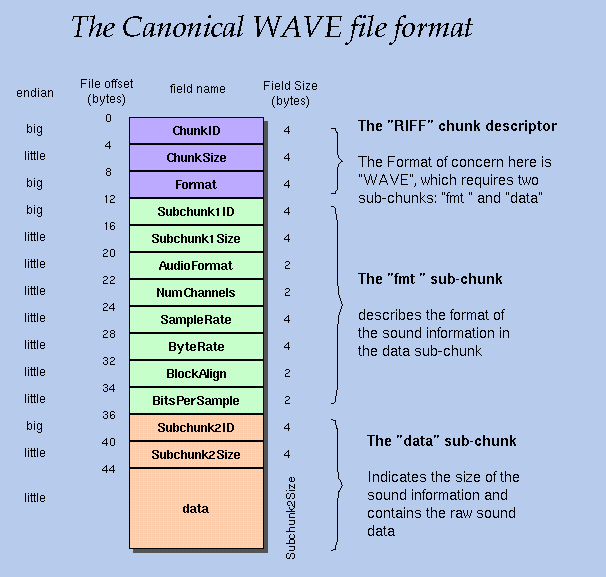
Sukurtos sistemos algoritmas susideda iš mikrovaldiklio aparatinės įrangos ir periferijų inicijacijos (HAL sluoksnio, laikmačių, LED, mygtukų, SD kortelės ir pertraukėjų). Toliau programa bando aptikti SD kortelę cikle, tam nepavykus mikrovaldiklis pereina į mažos galios režimą naudojant **\_\_WFI()** funkcija (laukti pertraukimo). Aptikus kortelę ji yra susiejama su „FatFs“ failų sistema. Konfigūruojamas garso įvesties periferinis įrenginys (mikrofonas) naudonat I2S sąsają su nurodytais parametrais. Tada yra pradedamas mikrofono audio įrašymas. Galiausiai ciklas nuolat tikrina kintamajį („BUTTONInterrupt“), rodančią mygtuko pertraukimo būseną. Įvykus paspaudimui, jei SD kortelės registravimas įjungtas, jis jį išjungia; kitu atveju įrašymas yra įjungiamas. Šis programos algoritmas grafiškai yra pavaizduotas paveikslėjyje (9 pav.).



9 pav. Pagrindinis programos algoritmas

## WAVE failų formatas

WAVE failo formatas yra „Microsoft“ RIFF (Resource Interchange File Format) specifikacijos, skirtos daugialypės terpės failams saugoti, pogrupis. RIFF failas prasideda failo antrašte, po kurios seka duomenų dalių seka. WAVE failas dažnai yra tik RIFF failas su vienu „WAVE“ gabalėliu, kurį sudaro dvi dalys – „fmt“ dalis, nurodanti duomenų formatą, ir „duomenų“ dalis, kurioje yra tikrieji duomenys.



10 pav. Standartinis WAVE formatą

Šią informaciją reikėjo įrašyti į failo pradžia norint, kad kompiuteris atpažintų .wav formato audio failą iš įrenginio. Tai padaryti nebuvo labai sunku, nes internete yra nemažai pavyzdžių tačiau taip pat reikėjo įrašyti tokius garso parametrus kaip audio kanalų skaičius, diskretizavimo dažnis bei bitų greitis. Dalis šį darbą atliekančios funkcijos pateikiama lentelėje toliau.

**1 lentelė.** Dalis .wav failo antraštės įrašymo funkcija

|  |
| --- |
| static uint32\_t WavProcess\_HeaderInit()  {    uint16\_t   BitPerSample=16;    uint16\_t   NbrChannels=AUDIO\_CHANNELS;    uint32\_t   ByteRate=AUDIO\_SAMPLING\_FREQUENCY\*(BitPerSample/8);    uint32\_t   SampleRate=AUDIO\_SAMPLING\_FREQUENCY;    uint16\_t   BlockAlign= NbrChannels \* (BitPerSample/8);      pHeader[0] = 'R';    pHeader[1] = 'I';    pHeader[2] = 'F';    pHeader[3] = 'F';   ... Neesminis kodas      /\* Write the file format, must be 'WAVE' -----------------------------------\*/    pHeader[8]  = 'W';    pHeader[9]  = 'A';    pHeader[10] = 'V';    pHeader[11] = 'E';    ... Neesminis kodas      /\* Write the number of channels, ie. 0x01 (Mono) ---------------------------\*/    pHeader[22]  = NbrChannels;    pHeader[23]  = 0x00;    pHeader[24]  = (uint8\_t)((SampleRate & 0xFF));    pHeader[25]  = (uint8\_t)((SampleRate >> 8) & 0xFF);    pHeader[26]  = (uint8\_t)((SampleRate >> 16) & 0xFF);    pHeader[27]  = (uint8\_t)((SampleRate >> 24) & 0xFF);    /\* Write the Byte Rate -----------------------------------------------------\*/    pHeader[28]  = (uint8\_t)(( ByteRate & 0xFF));    pHeader[29]  = (uint8\_t)(( ByteRate >> 8) & 0xFF);    pHeader[30]  = (uint8\_t)(( ByteRate >> 16) & 0xFF);    pHeader[31]  = (uint8\_t)(( ByteRate >> 24) & 0xFF);    /\* Write the block alignment -----------------------------------------------\*/    pHeader[32]  = BlockAlign;    pHeader[33]  = 0x00;  ... Neesminis kodas      return 0;  } |

Taip pat pabaigus garso įrašymą yra svarbu atlikti duomenų kiekio atnaujinimą failo antraštėje. Šią operacija atlieka toliau aprašyta funkcija.

**2 lentelė.** Dalis .wav headerio įrašymo funkcijos

|  |
| --- |
| static uint32\_t WavProcess\_HeaderUpdate(uint32\_t len)  {    /\* Write the file length ----------------------------------------------------\*/    /\* The sampling time: this value will be be written back at the end of the       recording opearation.  Example: 661500 Btyes = 0x000A17FC, byte[7]=0x00, byte[4]=0xFC \*/    pHeader[4] = (uint8\_t)(len);    pHeader[5] = (uint8\_t)(len >> 8);    pHeader[6] = (uint8\_t)(len >> 16);    pHeader[7] = (uint8\_t)(len >> 24);    /\* Write the number of sample data -----------------------------------------\*/    /\* This variable will be written back at the end of the recording operation \*/    len -=44;    pHeader[40] = (uint8\_t)(len);    pHeader[41] = (uint8\_t)(len >> 8);    pHeader[42] = (uint8\_t)(len >> 16);    pHeader[43] = (uint8\_t)(len >> 24);    /\* Return 0 if all operations are OK \*/    return 0;  } |

## Failų sistema

Supaprastinti failų įrašymą į SD kortelę buvo panaudota failų sistema – „FatFs“. FatFs yra bendras FAT / exFAT failų sistemos modulis, skirtas mažoms įterptinėms sistemoms. FatFs modulis parašytas laikantis ANSI C (C89) ir visiškai atskirtas nuo disko įrašymo/skaitymo (I/O) sluoksnio. Todėl jis nepriklauso nuo naudojamos platformos. FatFs sukurtas saugužiam naudojimui realaus laiko operacinėje sistemoje (angl. thread-safe) ir yra integruotas į ChibiOS, RT-Thread, ErlendOS ir Zephyr realaus laiko operacines sistemas. Minimalioje modulio versijoje programinis kodas užima nuo 2 iki 10 kB RAM atminties. Jis gali būti įtrauktas į mažus mikrovaldiklius su ribotais ištekliais, kaip pvz., 8051, PIC, AVR, ARM, Z80, RX ar kitus.

Sukurtame projekte failų sitema inicializuoja SD kortelę ir susieja „FatFs“ draiverį su nurodytu takeliu (angl. *path*). Tai yra pirmasis žingsnis norint pradėti naudoti failų sistemą. Jei inicializacija pavyksta, sistema registruoja failų sistemą ir ją priskiria „FatFs“ moduliui. Jei nepavyksta, įvykdomas klaidos apdorojimas, kuris šiuo atveju įjungia LED indikatorių ir sustabdo programos vykdymą, kaip parodyta **3 lentelė**.

**3 lentelė**. SD kortelė inizializavimo funkcija.

|  |
| --- |
| void DATALOG\_SD\_Init(void)  {    char SDPath[4];      if(FATFS\_LinkDriver(&SD\_Driver, SDPath) == 0) {      if(f\_mount(&SDFatFs, (TCHAR const\*)SDPath, 0) != FR\_OK) {        while(1) { // error          BSP\_LED\_On(LED1);          HAL\_Delay(500);          BSP\_LED\_Off(LED1);          HAL\_Delay(100);        }      }    }  } |

Taip pat projekte buvo įgyvendintas failo pavadinimo generavimas pagal skaičių, kuris padidinamas kiekvieną kartą, kai funkcija yra iškviečiama (kiekvieną kartą vėl paspaudus mygtuką įrašyti). Tada ji atidaro failą su sugeneruotu pavadinimu, inicijuoja .wav failo headerį ir įrašo jį į failą. Ši funkcija grąžina sėkmės arba nesėkmės būseną, leisdama kontroliuoti programos srautą.

**4 lentelė.** Garso .wav failo headerio įrašymo funkcija

|  |
| --- |
| uint8\_t DATALOG\_SD\_Log\_Enable(void)  {    static uint16\_t sdcard\_file\_counter = 0;      uint32\_t byteswritten; /\* written byte count \*/    char file\_name[30] = {0};      WavProcess\_HeaderInit();      sprintf(file\_name, "%s%.3d%s", "Audio\_Record\_N", sdcard\_file\_counter, ".wav");    sdcard\_file\_counter++;    if(f\_open(&MyFile, (char const\*)file\_name, FA\_CREATE\_ALWAYS | FA\_WRITE) != FR\_OK) {      return 0;    }      if(f\_write(&MyFile, pHeader, sizeof(pHeader), (void \*)&byteswritten) != FR\_OK) {      return 0;    }    return 1;  } |

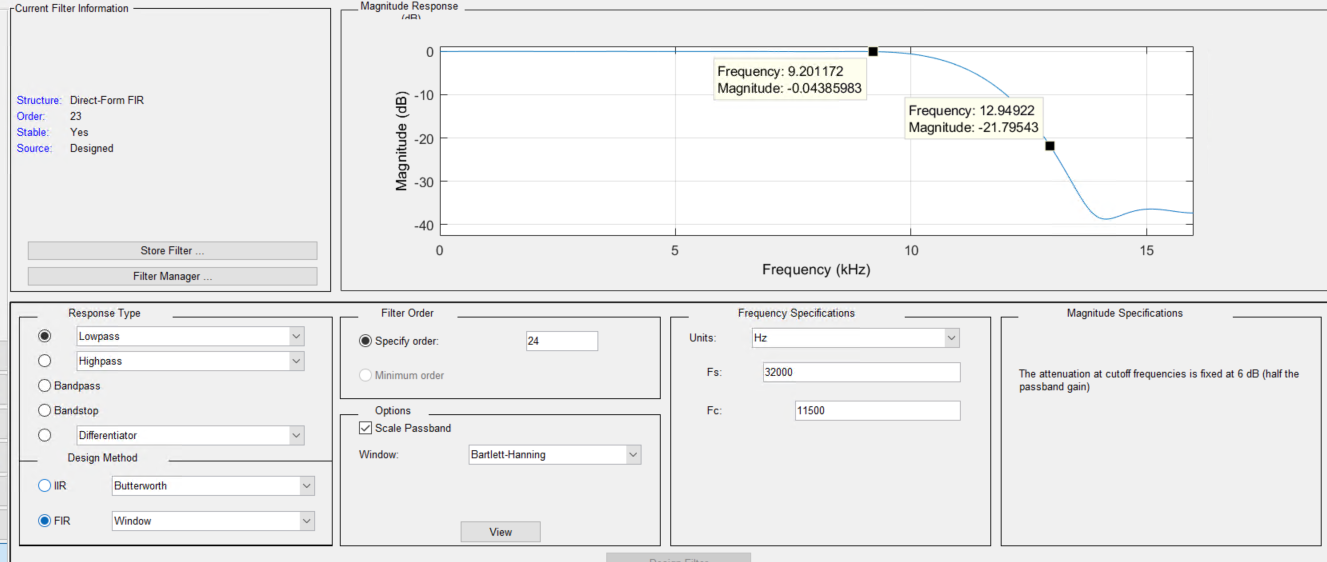
Galiausiai audio duomenys yra įrašomi pertraukiklio callback'o funkcijoje, kuri kviečiama, kai nustatomas mygtuko paspaudimas. Ji tikrina, ar įrašymas į SD kortelę įjungtas. Jei taip, ji įrašo garso duomenis į failą iš nurodyto buferio, o jei įvyksta klaida, įjungiamas LED indikatorius. Tai suteikia galimybę įrašyti garso duomenis į failą realaus laiko režime, priklausomai nuo įvykusio įvykio.

**5 lentelė.** Pertraukiklio funkcijos callbackas skirtas įrašyti garso duomenis į failą jeigu mygtukas nustauspas

|  |
| --- |
| /\*\*  \* @brief Highest priority interrupt handler routine  \* @param None  \* @retval None  \*/  void SW\_Task1\_Callback(void)  {    FRESULT s;    uint32\_t byteswritten; /\* File write/read counts \*/      /\* Check Push Button Event  \*/    if (SD\_Log\_Enabled) {      s=f\_write(&MyFile, &(((uint8\_t \*)Audio\_OUT\_Buff)[index\_buff]), SIZE\_BUFF, (void \*)&byteswritten);      if(s != FR\_OK) {        BSP\_LED\_On(LED4);      }    }  } |

## Signalo filtravimas

Paprasto žmogaus knarkimo garso dažnių diapazonas prasideda nuo 180 Hz ir pasiekia piką ties 300 Hz. Tačiau pasitaiko įvairių dažnių diapazonas kuris gali prasidėti nuo 60-130 Hz, ir pasiekti iki 1000 Hz dažnio garsą. Kuo didesnis dažnis, tuo didesnė viršutinių kvėpavimo takų obstrukcija. Aukšto dažnio triukšmą, pvz., elektroninius trukdžius ar šnypštimą, galima sumažinti įdiegus žemų dažnių filtrą, kuris sumažintų dažnius, viršijančius žmogaus kalbos diapazoną (pvz., virš 10 kHz). Būtent dėl šių priežąsčių buvo sukurtas FIR tipo lowpass filtras naudojant MATLAB filter designer įrankį.



11 pav. Sukurtas lowpass filtras su MATLAB filter designer programa

Todėl buvo sukurtas Lowpass FIR (Finite Impulse Response) tipo filtras, naudojant Bartlett-Hamming langą. Filtras yra 23 eilės, jo naudojamas atkirtos (cutoff) dažnio (Fc) vertė yra 11‘500 Hz, o signalo diskretizavimo dažnio (Fs) vertė yra 32000 Hz. Bartlett-Hamming langas buvo naudojamas kaip langas (window) filtravimo koeficientams apskaičiuoti. Teoriškai filtras leidžia praeiti signalams su mažesniu nei 11500 Hz dažniu. Tačiau galima matyti (11 pav.), kad signalai pradedami slopinti ties 9,2kHz dažnio verte, kas ir yra siekiama remiantis anksčiau aptartuose žmogaus skleidžiamuose dažniuose.

Ikelti koda is stm32

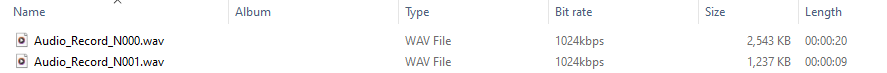
## Sistemos veikimas

Naudojantis „BlueCoin“ rinkiniu naujas kodas yra ikeliamas naudojantis Nucleo mikrovaldikliu kaip pavaizduota paveikslėlyje (xxxx)

Pati sistema gali veikti tiek maitinant per USB sąsają tiek iš išorinės baterijos.

Ikelti foto coino

Galiausiai iš įrenginio įrašyti .wav failai perkėlus SD kortelę į kompiuterį yra pasiekiami ir įrašytas garsas gali būti grojamas su kompiuterinėmis media aplikacijomis.



12 pav. Gauti audio įrašai iš sistemos SD kortelės

Pvz su filtru ar be ne FFT kazkaip galima gauti?

Audio rezultatas failo

Išvados

1. Darbe buvo apžvelgta miego apnėjos problematika ir egiztuojantis sprendimo metodai.
2. Sukurta, sistema naudojantis MEMS mikrofonais galinti įrašyti audio duomenis į SD kortelę.
3. Darbe buvo sukurtas FIR tipo žemų dažnių filtras naudojantis MATLAB filter designer įrenginį ir jo koeficientai eksportuoti į STM32 mikrovaldiklį audio signalo filtravimui.
4. Atliekant darbą taip pat buvo susipažinta su su .wav failų formatu ir FatFs failų sistema noritn patogiai įrašyti ir ateityje pasiekti garso įrašus.

Literatūros sąrašas

1. Irwin MR. Why sleep is important for health: a psychoneuroimmunology perspective. Annu Rev Psychol. 2015 Jan 3;66:143-72. doi: 10.1146/annurev-psych-010213-115205.
2. Parish, J.M. and Somers, V.K., 2004, August. Obstructive sleep apnea and cardiovascular disease. In Mayo Clinic Proceedings (Vol. 79, No. 8, pp. 1036-1046). Elsevier. doi: 10.4065/79.8.1036.
3. Kapur, V., Strohl, K.P., Redline, S., Iber, C., O'connor, G. and Nieto, J., 2002. Underdiagnosis of sleep apnea syndrome in US communities. Sleep and Breathing, 6(02), pp.049-054. doi: 10.1055/s-2002-32318.
4. Motamedi, K.K., McClary, A.C. and Amedee, R.G., 2009. Obstructive sleep apnea: a growing problem. Ochsner Journal, 9(3), pp.149-153.
5. Baran, R., Grimm, D., Infanger, M. and Wehland, M., 2021. The effect of continuous positive airway pressure therapy on obstructive sleep apnea-related hypertension. International Journal of Molecular Sciences, 22(5), p.2300. doi: 10.3390/ijms22052300.
6. Vensel Rundo, J., Downey, R. (2019). "Chapter 25 - Polysomnography." In Levin, K. H., Chauvel, P. (Eds.), Handbook of Clinical Neurology (Vol. 160, pp. 381-392). Elsevier. ISSN 0072-9752. ISBN 9780444640321. doi: 10.1016/B978-0-444-64032-1.00025-4.
7. Chang, H.-C.; Wu, H.-T.; Huang, P.-C.; Ma, H.-P.; Lo, Y.-L.; Huang, Y.-H. Portable Sleep Apnea Syndrome Screening and Event Detection Using Long Short-Term Memory Recurrent Neural Network. Sensors 2020, 20, 6067. https://doi.org/10.3390/s20216067
8. Lin, X., Cheng, H., Lu, Y., Luo, H., Li, H., Qian, Y., Zhou, L., Zhang, L. and Wang, M., 2022. Contactless sleep apnea detection in snoring signals using hybrid deep neural networks targeted for embedded hardware platform with real-time applications. Biomedical Signal Processing and Control, 77, p.103765. doi: 10.1016/j.bspc.2022.103765.
9. A.H. Yüzer, H. Sümbül, K. Polat, A Novel Wearable Real-Time Sleep Apnea Detection System Based on the Acceleration Sensor, IRBM, Volume 41, Issue 1, 2020, pp. 39-47, ISSN 1959-0318, https://doi.org/10.1016/j.irbm.2019.10.007.

1. Apnėjos vizualizacija, prieiga per: https://www.uvmhealth.org/healthwise/topic/tp12620 [↑](#footnote-ref-2)