

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

Приладобудівний факультет

Кафедра автоматизації та систем неруйнівного контролю

«На правах рукопису»

УДК 681.5

До захисту допущено:

Завідувач кафедри

_____Юрій КИРИЧУК

«__» _____2022 р.

Магістерська дисертація

на здобуття ступеня магістра

**за освітньо-професійною програмою «Комп'ютерно-інтегровані системи та
технології в приладобудуванні»**

зі спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології»

**на тему: «Оптимізація алгоритму опрацювання сигналів в
системі автоматизованого контролю іонізуючого
випромінювання»**

Виконав:

Студент II курсу, групи ПК-11мп

Мельник Андрій Вікторович _____

Керівник:

Доцент, кандидат технічних наук, доцент

Лисенко Юлія Юріївна _____

Консультант з розробки стартап-проекту:

Завідувач кафедри економічної кібернетики,

Доктор економічних наук, професор

Бояринова Катерина Олександрівна _____

Рецензент:

Професор, доктор технічних наук

Защепкіна Наталія Миколаївна _____

Засвідчую, що у цій магістерській дисертації
немає запозичень з праць інших авторів без
відповідних посилань.

Студент _____

Київ – 2022 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Приладобудівний факультет

Кафедра автоматизації та систем неруйнівного контролю

Рівень вищої освіти – другий (магістерський)

Спеціальність – 151 «Автоматизація та комп'ютерно-інтегровані технології»

Освітньо-професійна програма «Комп'ютерно-інтегровані системи та технології в приладобудуванні»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____Юрій КИРИЧУК

«__» _____ 2022 р.

ЗАВДАННЯ

на магістерську дисертацію студенту
Мельнику Андрію Вікторовичу

1. Тема дисертації «Оптимізація алгоритму опрацювання сигналів в системі автоматизованого контролю іонізуючого випромінювання», науковий керівник дисертації Лисенко Юлія Юріївна, доцент, затверджена наказом поуніверситету від «9» листопада 2022р. № 4110-с
2. Термін подання студентом дисертації

3. Об'єкт дослідження: процес вимірювання іонізуючого випромінювання
4. Вихідні дані: Об'єкт дослідження – алгоритм опрацювання сигналів в системі автоматизованого контролю іонізуючого випромінювання β - і γ -випромінювання
5. Перелік завдань, які потрібно розробити: оглянути методи вимірювання іонізуючого випромінювання, проаналізувати та обрати метод вимірювання з датчиком, оглянути алгоритми обробки сигналів, обрати методи оптимізації алгоритмів, розробити автоматизовану систему, розробити програму для обробки та відображення результатів.
6. Орієнтовний перелік графічного (ілюстративного) матеріалу: 3 плакати

7. Орієнтовний перелік публікацій:

- Development Of Automated Eddy Current System For Aircraft Structure Inspection / I. Lysenko, Y. Kuts, V. Petryk, V. Malko, A. Melnyk – прийнято до розгляду в журналі «Transactions on Aerospace Research, Institute of Aviation», Poland.
- Учасник у виставці стартап-проектів «Sikorsky Challenge 2021: Україна і світ»

8. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Розробка стартап-проектів	Завідувач кафедри економічної кібернетики, Доктор економічних наук, професор Бояринова Катерина Олександрівна		

9. Дата видачі завдання _____

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1	Формулювання завдання магістерської дисертації	1 тиждень	
2	Огляд методів вимірювання іонізуючого випромінювання	2 тижні	
3	Розробка приладу дозиметричного контролю іонізуючого випромінювання	1 тиждень	
4	Огляд алгоритмів обробки сигналів з приладів дозиметричного контролю	2 тижні	
5	Розробка системи автоматизації для процесу дозиметричного контролю іонізуючого випромінювання	2 тижні	
6	Огляд бібліотек та можливостей мови програмування C# для створення програм, обробки та відображення результатів вимірів	2 тижні	
7	Оптимізація алгоритму обробки сигналів та розробка програмного забезпечення для відображення результатів вимірів	2 тижні	
8	Розробка стартап-проекту	1 тиждень	
9	Формування висновків та оформлення пояснювальної записки та презентації	1 тиждень	

Студент

Андрій МЕЛЬНИК

Науковий керівник

Юлія ЛИСЕНКО

РЕФЕРАТ

Актуальність теми

Тема магістерської дисертації пов'язана ефективним визначенням рівня забрудненості навколишнього середовища, що безпосередньо впливає на якість життя. Рівень радіоактивного фону – це один із показників, які потребують постійного моніторингу. Вплив іонізованих частинок на організм людини може проявитися через тривалий час, і тоді контроль може бути вже не актуальним. Тому розробка недорогої, зручної, портативної та діджиталізованої системи реєстрації іонізуючого випромінювання на сьогодні є дуже актуальною.

У зв'язку із відкриттям нових технологій – з'являється можливість для удосконалення існуючих систем, зменшення вартості пристрою, підвищення рівня точності, підвищення рівня швидкодії.

В даній магістерській дисертації спроектовано систему на базі лічильника Гейгера, що реалізує розроблений алгоритм обробки сигналу по відокремленню імпульсів та вирахування дози випромінювання. Система передбачає реєстрацію випромінювання, передачу даних на смартфон, де у спеціально розробленому додатку проходить аналіз даних, вираховується значення доз та візуалізується отримана інформація. Розроблена система визначає накопчену дозу, сповіщає користувача, якщо значення еквівалентної дози перевищено. Удосконалений алгоритм обробки сигналів дозволяє зменшити час сповіщення користувача при збільшенні дози випромінювання.

Мета і задачі дослідження

Мета дослідження – покращення методу алгоритму обробки сигналів та оптимізація алгоритму підрахунку дози для підвищення швидкодії.

На шляху досягнення поставлених цілей були вирішені наступні **задачі**:

- 1) аналіз всіх існуючих методів реєстрації іонізуючого випромінювання
- 2) удосконалення алгоритму підрахунку значення дози за вимірними імпульсами

3) аналіз алгоритму обробки сигналів.

4) розробка автоматизованої системи реєстрації потужності еквівалентної дози іонізуючого випромінювання.

Об'єкт дослідження – формування та опрацювання сигналів в системах автоматизованого контролю іонізуючого випромінювання.

Предмет дослідження – методи вимірювання іонізуючого випромінювання та методи оптимізації алгоритму опрацювання сигналів.

Методи дослідження базуються на теорії іонізуючого випромінювання, обробці отриманих сигналів з приладу, алгоритми обрахунку значень дози відносно зареєстрованих імпульсів

Наукова новизна одержаних результатів

1) Удосконалено алгоритм обрахунку дози випромінювання відносно зареєстрованих імпульсів з лічильника Гейгера.

Практичне значення результатів дисертації

1) Реалізовано автоматизований прилад для реєстрації рівня іонізуючого випромінювання і програма для обробки та візуалізації даних

Ключові слова

Вимірювання рівня іонізуючого випромінювання, газові лічильники, автоматизація процесу, алгоритм підрахунку дози.

ABSTRACT

Currently, the development of an inexpensive, convenient, portable, and digitized device for recording ionizing radiation is very relevant.

The topic of the master's thesis is related to the effective determination of the level of environmental pollution, which directly affects the quality of life. The level of radioactive background is one of the indicators that require constant monitoring. The impact of ionized particles on the human body may appear after a long time, then control may no longer be relevant.

In connection with the discovery of new technologies - appears to improve existing systems, reduce the cost of the device, increase the level of accuracy, and increase the level of speed.

In this project, a device based on a Geiger counter and a signal-processing algorithm for pulse separation and radiation dose calculation was developed. The device registers radiation, and transmits the data to a smartphone, the developed application analyzes the data, calculates the values of doses, and visualizes the information. The developed system determines the accumulated dose and notifies the user if the value of the equivalent dose is exceeded. The improved algorithm of signal processing allows for reducing the time of notification of the user in the event of an increased radiation dose.

The purpose and objectives of the research

The purpose of the research is to improve the method of the signal processing algorithm and optimize the dose calculation processing algorithm to increase the speed

On the way to achieving the set goals, the following tasks were solved:

- 1) Analysis of all existing methods of recording ionizing radiation
- 2) Improvement of the algorithm for calculating the dose value based on the measured pulses
- 3) Analysis of the signal processing algorithm.

4) Development of an automated system for registering the power of the equivalent dose of ionizing radiation.

The object of the research is the optimization of the signal processing algorithm in the system of automated monitoring of ionizing radiation.

The subject of research is methods of measuring ionizing radiation.

Research methods are based on the theory of ionizing radiation, processing of received signals from the device, algorithms for calculating dose values relative to registered pulses

The scientific novelty of the obtained results

1) The algorithm for calculating the radiation dose relative to registered pulses from a Geiger counter has been improved.

The practical significance of the results of the dissertation

1) An automated device for recording the level of ionizing radiation and a program for data processing and visualization have been implemented

Keywords

Measurement of the level of ionizing radiation, gas meters, process automation, and dose calculation algorithm.

ЗМІСТ

ВСТУП.....	11
РОЗДІЛ 1.АНАЛІТИЧНИЙ ОГЛЯД	12
1.1. Детектори радіації	12
1.2. Реєстрація іонізуючого випромінювання.....	12
1.3 Класифікація детекторів	18
1.4. Огляд лічильників Гейгера	26
1.4.1. Ефективність лічильника Гейгера:	28
1.4.2. Час відновлення	29
1.4.3. Корекція для часу відновлення:	31
Висновки до розділу	34
РОЗДІЛ 2. РОЗРОБКА ТА ПІДБІР ВУЗЛІВ СТРУКТУРНОЇ СХЕМИ	35
РОЗДІЛ 3. РОЗРОБКА ПРОГРАМНИХ АЛГОРИТМІВ.....	40
3.1 Опис роботи програми	40
3.2 Дослідження обробки сигналів за допомогою Google Colaboratory.....	44
3.2.1 Проведення дослідження	46
3.2.2. Опис алгоритму вирахування миттєвої дози	47
3.3 Дослідження імпульсів в реальних вимірах	48
3.4 Опис роботи додатку	53
3.5 Подальший розвиток проєкту	65
3.6 Приклад роботи програми	67
Висновки до розділу	69

РОЗДІЛ 4. РОЗРОБКА СТАРТАП-ПРОЄКТУ «АВТОМАТИЗОВАНА СИСТЕМА ДОЗИМЕТРИЧНОГО КОНТРОЛЮ ІОНІЗУЮЧОГО ВИПРОМІНЮВАННЯ»	70
4.1 Опис технологічний аудит ідеї проєкту	70
4.2 Аналіз ринкових можливостей запуску стартап-проєкту	73
4.3 Розроблення ринкової стратегії продукту	81
4.4 Розроблення маркетингової програми стартап-проєкту	83
Висновки до розділу	92
ВИСНОВКИ	93
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	94

Перелік основних скорочень

НК – неруйнівний контроль;

АЦП – аналого-цифровий перетворювач;

ЦАП – цифро-аналоговий перетворювач;

ПК – персональний комп'ютер;

ПЗ – програмне забезпечення.

БД – база даних

ВСТУП

Радіація - це явище, присутнє в нашому повсякденному житті, що походить із природних і штучних джерел. Живі організми сильно страждають від клітинного пошкодження, спричиненого радіацією, що загрожує як здоровим, так і хворим тканинам. Тривалий вплив іонізуючого випромінювання на людський організм може спричинити значні та незворотні пошкодження. Таким чином, вимірювання рівня іонізуючого випромінювання навколишнього середовища в потенційно небезпечних зонах може допомогти зменшити вплив радіації на організм людини [1].

Дозиметр іонізуючого випромінювання — це система, яка вимірює випромінювання та фіксує потужність еквівалентної чи поглинутої дози.

Результат вимірювання — числове значення дозиметричної величини у відповідних одиницях.

Потужність дози — це біологічний вплив на організм від радіаційного опромінення.

Прилад можна назвати радіаційним дозиметром — коли він вимірює одну чи кілька фізичних властивостей дозиметричних величин з належним калібруванням.

Неможливо одним пристроєм покрити усі аспекти вимірювання радіаційного фону, тому проектувати такі пристрої потрібно під конкретні задачі з конкретними діапазонами випромінювання. Для вимірювання іонізуючого випромінювання навколишнього середовища не потрібно надточної системи, але вона має чітко та швидко реагувати на змін фоновий рівень. В даній ситуації доцільно обирати датчик, який здатен реєструвати очікуване іонізуюче випромінювання. Для обробки сигналів потрібно підібрати правильний алгоритм, аби про зміну рівня радіаційного фону система сповіщала як найшвидше [1].

Пристрій реєстрації іонізуючого випромінювання в повсякденному житті має бути сучасним, інформативним, мати калібрування системи.

РОЗДІЛ 1. АНАЛІТИЧНИЙ ОГЛЯД

1.1. Детектори радіації

У людини немає органів чуття, здатних розпізнавати іонізуюче випромінювання. Тому вони повністю покладаються на прилади для виявлення та вимірювання радіації [3].

Прикладом інструментів, які використовуються у фізиці є:

I. Прилади вимірюють частинки:

- а. Лічильники Гейгера-Мюллера.
- б. Сцинтиляційний лічильник.

II. Прилади вимірюють накопичені дози:

- а. Плівкові значки.
- б. Кишенькові дозиметри.
- в. Термолюмінесцентні дозиметри.

III. Прилади вимірюють дозу та потужність дози:

- а. іонізаційно-камерні.

1.2. Реєстрація іонізуючого випромінювання

Терміни та визначення

Іонізуюче випромінювання можна виявити за його взаємодією із середовищем, що виражається створенням іонів різних знаків. Фотонне випромінювання вступає в процес взаємодії із речовиною шляхом фотоелектричного поглинання, утворення пар та комптонівського розсіювання. Кожен із перелічених процесів утворює електрони та іони, що в свою чергу можуть бути виміряні та визначені. Відношення кількості зафіксованих іонів одного знаку (dn), що створені іонізуючими частинками на елементарному шляху (dh), до цього шляху називаються *лінійною іонізацією* і визначаються як $R = dn / dh$. На лінійну іонізацію найбільше впливають характер іонізуючої речовини та енергія зарядженої частинки чи фотону [3].

Детектор іонізуючого випромінювання - це чутливий елемент конкретного

засобу вимірювання, котрий призначений для перетворення енергії іонізуючого випромінювання в інший вид енергії, що можна зручно реєструвати для подальшої обробки та вимірювання величин, які можуть дати характеристику впливу на детектор випромінювання. В ідеальному випадку детектор повинен реагувати на кожну іонізуючу частинку, що взаємодіє з ним та давати точну інформацію про значення її енергії, положення в просторі та безпосередньо час взаємодії. Також ідеальний детектор має забезпечувати обробку достатньо великої кількості фотонів, щоб мати хорошу роздільну здатність [3].

Наразі не існує ідеальних детекторів, які б задовольняли усім переліченим потребам: не всі фотони, що потрапили на детектор ефективно взаємодіють із ним.

Квартовий хід детектора Q – значення, що характеризує долю падаючих випромінюючих частинок на детектор, що при взаємодії утворюють величину, яку можна виміряти (фотоелектрон, імпульс струму, почорніння плівки чи інше). Вихідне значення квантового сигналу, що рівне одиниці, не завжди відноситься до ідеального детектора.

Для прикладу розглянемо два сцинтиляційних детектора:

P – реальний та **I** – ідеальний, вихідні сигнали значення яких формуються шляхом інтегрування світла. В ідеалі значення квантового виходу детектора **I** рівне одиниці. Він сконструйований так, що реакція сцинтилятора на кожне збурення дає чітко однаковий світлових вихід. Такий детектор дає змогу визначити чітку кількість фотонів.

P-детектор також має квантовий вихід, що дорівнює одиниці, а світловий вихід кожної сцинтиляції в середньому такий самий, як і детектор **I**, але виражається середнім значенням. Реєстрація таких сцинтиляцій не дає підрахувати точну кількість зареєстрованих первинних фотонів, хоча можна дати оцінку кількості їх загальних параметрів, оцінюючи якість детекторів, узагальнений квантовий вихід системи d запису, що визначається звітом [1].

$$QD = [\text{ВСШ}_{\text{вих}} / \text{ВСШ}_{\text{вх}}]^2 \quad (1.2.1)$$

Де:

$\text{ВСШ}_{\text{вих}}$ – відношення значень сигналу до шум на вихідному каналі;

$BSH_{\text{вх}}$ – відношення значень сигналу до шум вхідному каналі.

Види випромінювань

Іонізуючим випромінюванням називається потік частинок або квантів електромагнітного випромінювання, взаємодія яких із речовиною призводить до іонізації атомів. Такі випромінювання виникають в атомних та ядерних процесах – мимовільних чи вимушених перетвореннях електронних оболонок чи ядер атомів. Крім того, вони можуть бути отримані за допомогою прискорювачів або бути присутнім у космічних променях [4].

Дані випромінювання в залежності від маси та заряду можна підрозділити на чотири групи:

1. важкі заряджені частки;
2. легкі заряджені частки;
3. фотонне випромінювання: рентгенівське та гамма-випромінювання;
4. нейтронне випромінювання.

Для фахівців з розробки та експлуатації дозиметричної апаратури найбільший практичний інтерес становить інтервал енергій випромінювань від кількох кілоелектронвольт до 10–20 (MeV). Винятком будуть нейтрони, енергія яких може зменшуватись до теплових значень порядку сотих часток електронвольта, проте енерговиділення при взаємодії таких нейтронів з ядрами речовини потрапляє у вказаний діапазон енергій. Цей діапазон можна умовно назвати «ядерним», оскільки основні процеси ядерних перетворень відповідають саме цьому інтервалу енергій.

Звичайно, існують випромінювання і менших, і більших енергій. Методи та пристрої їх реєстрації можуть відрізнятися дуже суттєво.

У *першу групу* в основному входять альфа-частинки, а також протони, багатозарядні іони та уламки поділу. Основним механізмом взаємодії із речовиною таких частинок є іонізаційні втрати енергії. У «ядерному» діапазоні

енергій ці частинки будуть нерелятивістськими, що зумовлює високу щільність іонізаційних втрат і, відповідно, малі значення пробігів таких частинок у речовині. До першої групи також можна віднести нерелятивістські мюони та заряджені пі-мезони вторинних космічних променів, проте для більшості практичних вимірювань такі частки суттєвої ролі не відіграють.

До *другої групи* належать тільки електрони та позитрони, у тому числі і бета-частинки радіонуклідів. Основним механізмом взаємодії з речовиною у вказаному діапазоні енергій для них є іонізаційні втрати, однак радіаційні втрати зростають із збільшенням енергії та можуть відігравати істотну роль. Оскільки в «ядерному» діапазоні енергій ці частинки можуть бути релятивістськими, щільність їх іонізаційних втрат зменшується, а пробіги у речовині збільшуються. На відміну від частинок першої групи у разі легких заряджених частинок кулонівське розсіювання надає сильний вплив на напрямок їхнього руху. Для позитронів, як для античастинок, є особливий варіант взаємодії - анігіляція, що призводить до зникнення позитронів та появи анігіляційного гамма-випромінювання.

Перші дві групи частинок представляють безпосереднє іонізуюче випромінювання, оскільки безпосередньо іонізують атоми середовища при зіткненнях, якщо їхня кінетична енергія достатня для іонізації. Вони вибивають орбітальні електрони з атомів при кулонівській взаємодії. [5].

До *третьої групи* – фотонного іонізуючого випромінювання – відносяться рентгенівське та гамма-випромінювання різної природи:

1) гамма-випромінювання, короткохвильове електромагнітне випромінювання з довжиною хвилі $\lambda < 10\text{--}12$ м, яскраво вираженими корпускулярними та слабо вираженими хвильовими властивостями, що є потоком частинок – гамма-квантів (фотонів) з енергією $\varepsilon_\gamma = h\nu$ (ν – частота випромінювання, h – постійна Планка), більшою 100 (кеВ), що виникає при зміні енергетичного стану атомних ядер, при ядерних перетвореннях чи анігіляції частинок;

2) гальмівне випромінювання, що виникає при гальмуванні

високоенергетичних заряджених частинок у кулонівському полі ядер у речовині, що характеризується суцільним енергетичним спектром, з максимальною енергією ($\epsilon\gamma$), що дорівнює енергії зарядженій частки;

3) характеристичне випромінювання з дискретним енергетичним спектром, що виникає за зміни енергетичного стану електронів атома;

4) рентгенівське випромінювання, що складається з гальмівного та/або характеристичного випромінювання.

Рентгенівське випромінювання – електромагнітні хвилі, що займають спектральну область між гамма- та ультрафіолетовим випромінюваннями, що відповідає довжинам хвиль від 10^{-12} (м) до 10^{-7} (м), енергія кванта від 10 (eV) до десятка мегаелектронвольт. Енергетичні діапазони рентгенівського та гамма-випромінювань перекриваються у широкій області енергій. Чітка межа між гамма- і рентгенівськими випромінюваннями не визначена. Обидва типи випромінювання мають електромагнітну природу і за однакової енергії квантів еквівалентні, тому такий поділ умовний. Термінологічна відмінність полягає лише у способі виникнення: рентгенівське випромінювання утворюється за участю вільних електронів або електронів в атомах, а гамма випромінювання випромінюється в результаті переходів ядер із збуджених станів, при перебігу ядерних реакцій [5].

Основними механізмами взаємодії у «ядерному» діапазоні енергій фотонного випромінювання є фотоефект, комптонівське розсіювання та утворення електрон-позитронних пар. На ймовірний вибір варіанта взаємодії визначальний вплив мають як енергія квантів, так і порядковий номер атомів речовини, з якою відбувається взаємодія. Сенс поняття "пробіг" для частинок третьої групи змінюється і означає середню відстань, що проходить квантом до чергового одноразового акту взаємодії. У всіх актах взаємодії виникають частки, віднесені до другої групи.

Четверту групу частинок становлять нейтрони різних енергій. На відміну від перших трьох груп випромінювань основна взаємодія з речовиною для них відбувається не електромагнітним способом, а за допомогою особливої ядерної

взаємодії. Одноразові акти взаємодії нейтронів з ядрами атомів речовини призводять до двох варіантів: розсіювання нейтрону або ядерного процесу з вильотом різних частинок, віднесених до першої, третьої чи навіть четвертої групи. Поняття «пробіг» для нейтронів аналогічне пробігу для взаємодії гамма-квантів і залежить від їх енергії та нуклонів, що входять до складу атомів речовини.

Реєстрацію інших елементарних частинок не розглядатимемо з двох причин: вони або мають дуже малий час життя і в результаті розпаду перетворюються на частинки вже перерахованих чотирьох груп, або є стабільними (наприклад, нейтрон), але дуже слабо взаємодіють із речовиною і тому не знаходять практичного застосування [5].

Непряме іонізуюче випромінювання складається з нейтральних частинок з нульовим електричним зарядом – нейтронів, нейтральних мезонів чи квантів рентгенівського та гамма-випромінювань, що створюють безпосередньо іонізуюче випромінювання та/або викликають ядерні перетворення. Енергія цих частинок внаслідок взаємодії із середовищем передається спочатку зарядженим частинкам (електрону, протону, альфа-частці, ядру віддачі та ін.), а потім ці вторинні частинки вже виробляють іонізацію атомів та/або викликають ядерні реакції [5].

До *корпускулярного випромінювання* відноситься іонізуюче випромінювання, що складається з частинок з масою спокою, відмінною від нуля, тобто альфа- та бета-частинки, нейтрони, протони, мезони та ін.

Під *первинним іонізуючим випромінюванням* розуміється випромінювання, яке у аналізованому процесі взаємодії із середовищем є вихідним або приймається за вихідне.

Вторинне іонізуюче випромінювання виникає внаслідок взаємодії первинного випромінювання з даною речовиною.

Енергія, втрачена випромінюванням у речовині та віднесена до одиниці шляху у грамах на 1 (см²), практично не залежить від агрегатного стану речовини (газ, рідина або тверде тіло) [4].

Поглинання енергії будь-якого виду випромінювань у речовині призводить до ряду явищ, що використовуються для реєстрації випромінювань.

1.3 Класифікація детекторів

За необхідності підкреслити тип реєструючого іонізуючого випромінювання, назву детектора вибирають з додаванням терміну елемента, що вказує на види випромінювання. Приклади назв: альфа-детектор, рентгенівський детектор, нейтронний детектор.

Детектори поділяють за формою конструкції: точкові, лінійні та матричні.

За формою відображення вимірювальної інформації детектори іонізуючого випромінювання поділяють на аналогові та дискретні, а за формою залежності вихідного сигналу детектору від значення фізичної величини, що вимірюється: пропорційні та не пропорційні.

За станом речовини чутливого елемента детектори поділяють на газові, твердотілі та рідинні.

За методом реєстрації іонізуючого випромінювання детектори (таблиця 1.3) ділять на(табл. 1.3):

сцинтиляційні — радіюлюмінесцентні детектори, в яких використовується сцинтилююча речовина, що випускає кванти світла під дією іонізуючого випромінювання. Він оптично підключений безпосередньо або через світлодіод до фоточутливого пристрою.

Іонізаційні детектори -принцип дії яких заснований на використанні іонізації в речовинах світлочутливого елемента детектора.

Напівпровідникові - іонізуючі детектори, в яких електричне поле використовується для збирання носіїв заряду різного знаку, створених іонізуючим випромінюванням у напівпровідниковому матеріалі чутливого елемента детектора. Вони, в свою чергу, поділяються на охолоджуючі та неохолоджуючі

Таблиця 1.3.Ефекти іонізації, які використовуються при детектуванні і вимірюванні полів іонізуючого випромінювання [6]

Речовина чутливого елемента об'єкту	Тип системи реєстрації	Ефекти іонізації
Газ	Іонізаційна камера	Електричний
Газ	Пропорційний лічильник	Електричний
Газ	Лічильник Гейгера-Мюллера	Електричний
Напівпровідник	Напівпровідниковий перетворювач	Електричний
Сцинтилятор	Сцинтиляційний перетворювач	Світловий
Фотоемульсії	Рентгенівська плівка	Хімічний
Аморфний кремній	Багаторазові пластини з фото стимулюючою пам'яттю	Хімічний

Один детектор представляє інформацію про інтенсивність випромінювання. Для отримання інформації про розподіл інтенсивності випромінювання в просторі, детектори об'єднуються в ряди або матриці. У цьому випадку вони перетворюють радіаційне зображення в зображення іншого типу, яке називається *перетворювачем випромінювання*. Якщо радіаційне зображення перетворюється на оптичне зображення, то це називають *радіаційно-оптичним перетворювачем*. Зображення також можна отримати за допомогою детекторів з одним скануванням. Детектор рухається в просторі радіаційного зображення і дані зберігаються послідовно в усіх точках. Такий збір даних виключає можливість отримання динамічних помилок на радіаційних зображеннях.

Напівпровідниковий детектор

Кристали германію та кремнію є напівпровідниками, які зазвичай використовуються як чутливий об'єкт. Напівпровідниковий детектор діє як

твердотільна іонізаційна камера. Іонізуюча частинка (бета-, альфа-частинка тощо) взаємодіє з атомами в чутливому об'єкті детектора, утворюючи електрони шляхом іонізації.

Реєстрація цих іонів призводить до вихідного імпульсу. Енергія іонізації 30–35 (eV) для частинок в газі, що утворюють ефект іонізації в напівпровідниковому детекторі(кремнії). Заряд, що утворений іонізуючою частинкою в напівпровідниковому (кремнієвому) детекторі, в десять разів перевищує заряд, утворений цією ж частинкою в газовому детекторі. Такі значення допомагають забезпечити високу роздільну здатність та точне визначення енергії іонізуючих частинок [5].

Типи:

а. Кремнієві детектори: використовуються для виявлення альфа частинок і протонів.

б. Германієвий детектор: використовується для виявлення бета і гамма частинок.

Ефективність детектора

Ефективність детектора – це відношення кількості імпульсів, що виникають до кількості іонних частинок, які мали бути зареєстровані (рисунок 1.3.1).

а. Абсолютна ефективність: відношення кількості імпульсів вироблених детектором, до кількості гамма-променів, випромінюваних джерелом (у всіх напрямках).

б. Внутрішня ефективність: відношення кількості імпульсів, вироблених детектором, до кількості гамма-променів, що падають надетектор.

г. Відносна ефективність: ефективність одного детектора відносно іншого;

д. Повна енергетична пікова (або фотопікова) ефективність: Ефективність створення лише пікових імпульсів повної енергії, а не імпульсу будь-якого розміру для гамма-променів [5].

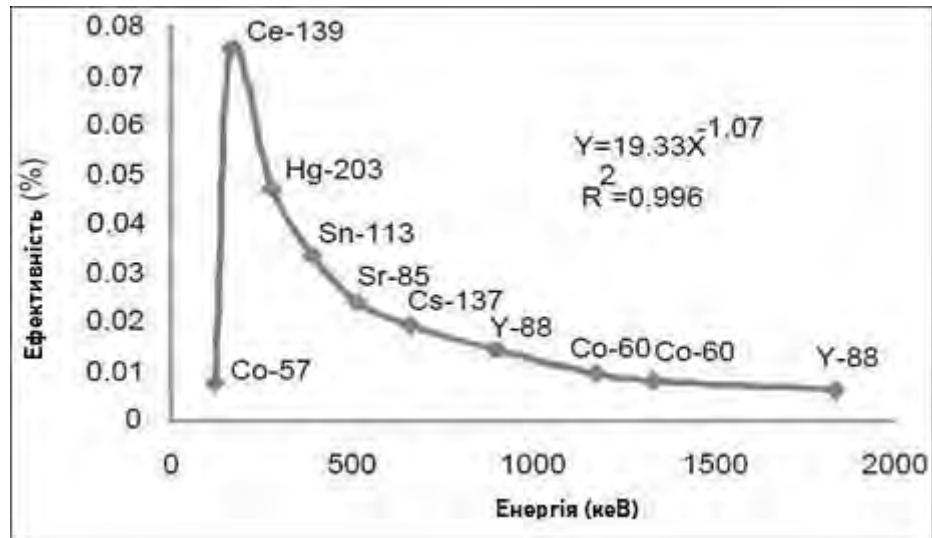


Рисунок 1.3.1. Калібрування ефективності напівпровідникового детектора

Роздільна здатність детектора - це міра ширини (повна ширина половини максимуму) одного енергетичного піку при певній енергії частинки. Роздільна здатність дозволяє системі чіткіше розділяти піки в спектрі (рисунок 1.3.2).

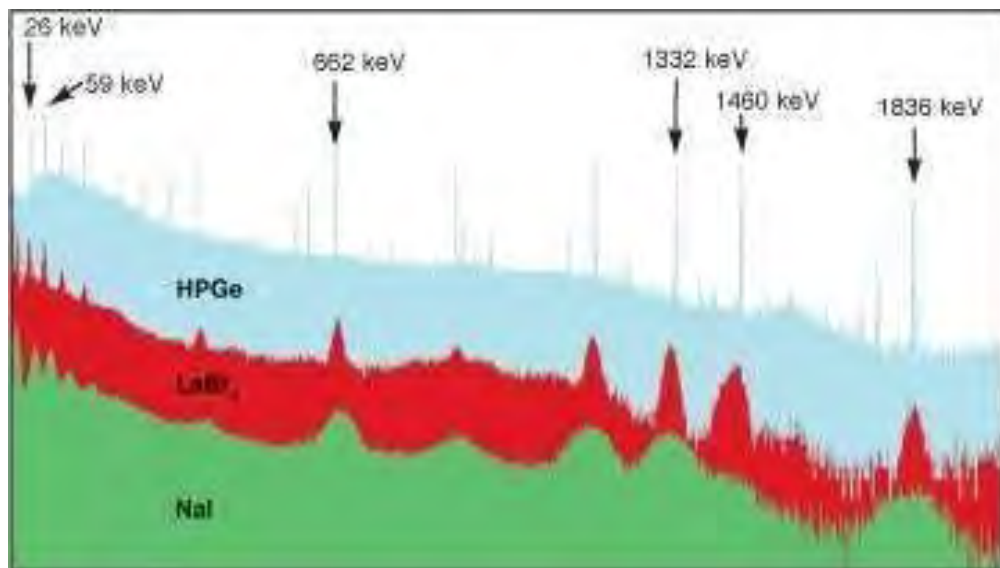


Рисунок 1.3.2. Роздільна здатність напівпровідникового детектора

На малюнку показано два спектри, зібрані з одного джерела, один з використанням чутливого елементу Йодид натрію (NaI), а другий з використанням Германію високої частоти (HPGe). Піки, представлені (NaI), мають не чіткі межі, тоді як піки HPGe чітко розділені [5].

Іонізуючі камери

В іонізуючій камері до об'єму газу, між дома електродами, прикладається

різниця потенціалів. В дозиметричних приладах камера часто має циліндричну форму: газ закупорений в циліндричному катоді, через який походить аксіально розташований стрижневий анод (рисунок 1.3.3, рисунок 1.3.4)

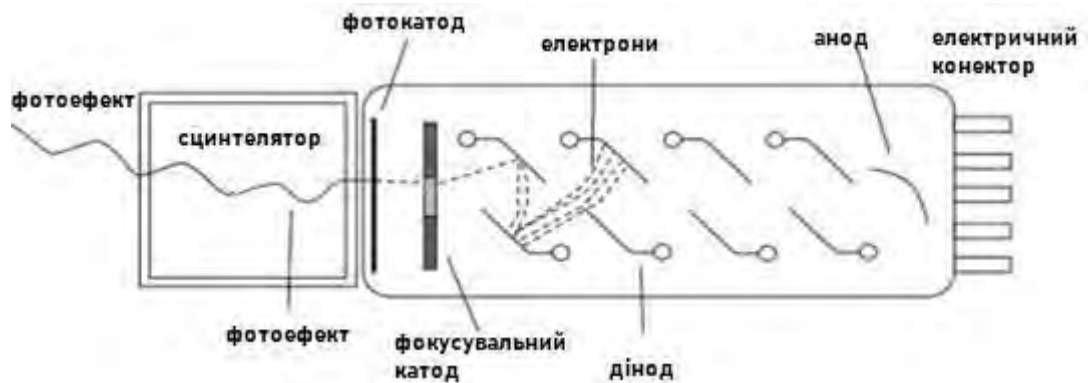


Рисунок 1.3.3. Схема роботи сцинтиляційного датчика



Рисунок 1.3.4. Сцинтиляційний детектор

В основному застосовують полііонізаційні камери (рисунок 1.3.5), скомпоновані у виді лінійних чи двовірних матриць.

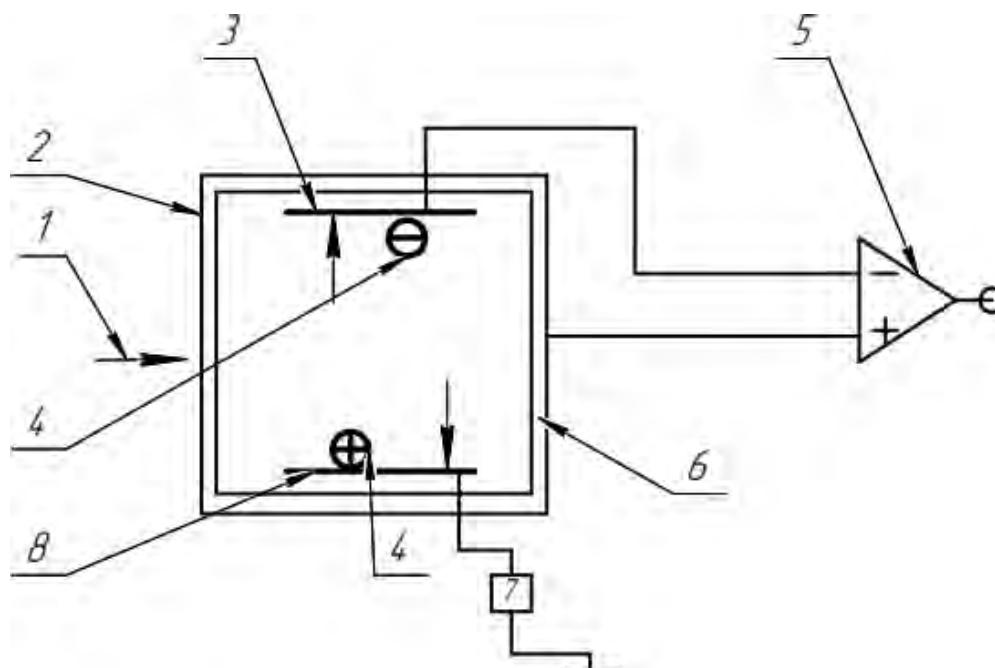


Рисунок 1.3.5 Схема пустої іонізаційної камери: 1 – іонізуюче випромінювання; 2 – вхідне вікно; 3 – анод; 4 – електрони та іонний газ; 5 – попередній підсилювач; 6 – стінки камери; 7 – джерело живлення; 8 – катод

Іонізуючі частинки проходять через чутливий об'єм камери та іонізуючий газ, що знаходиться в середині. При прикладанні до електродів різниці потенціалів – іони дрейфують вздовж ліній електричного поля, створюючи іонізуючий струм. В нормальних умовах електрони дрейфують зі швидкістю 10^4 (м/с). Швидкість дрейфу іонів набагато порядків менше [7].

Газові лічильники

Базова конфігурація газо-іонізаційного детектора показана (рисунок 1.3.6)

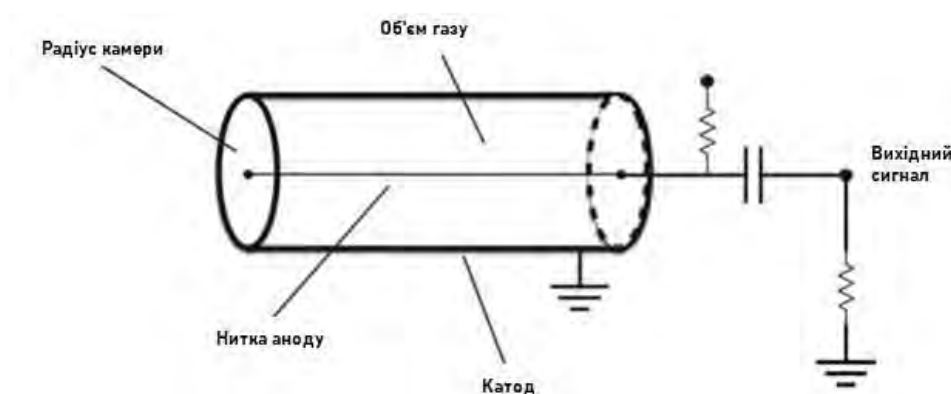


Рисунок 1.3.6. Схема іонізаційної камери

Циліндричний об'єм заповнений благородним газом, обраний через низький потенціал іонізації і спорідненість до електрона. Різниця потенціалів прикладена між центральним проводом і корпус створює радіальне електричне поле. Внаслідок прикладеної різниці потенціалів між анодом та катодом, будуть відбуватися імпульси струму на клемі трубки. На діаграмі (рисунок 1.3.7) показані загальні характеристики напруги для газоіонізаційного детектору [7].

Газоіонізаційні детектори можна охарактеризувати ефектами, створеними різною напруженістю поля між електродами, що збирають заряд. Зв'язок між розміром виробленого імпульсу та потенціалом, прикладеним до електродів газоіонізаційного детектора, показаний на рисунку 1.3.6.

Розмір імпульсу залежить від напруженості поля, а також від типу випромінювання, яке потрапляє в об'єм детектора та створює іони.

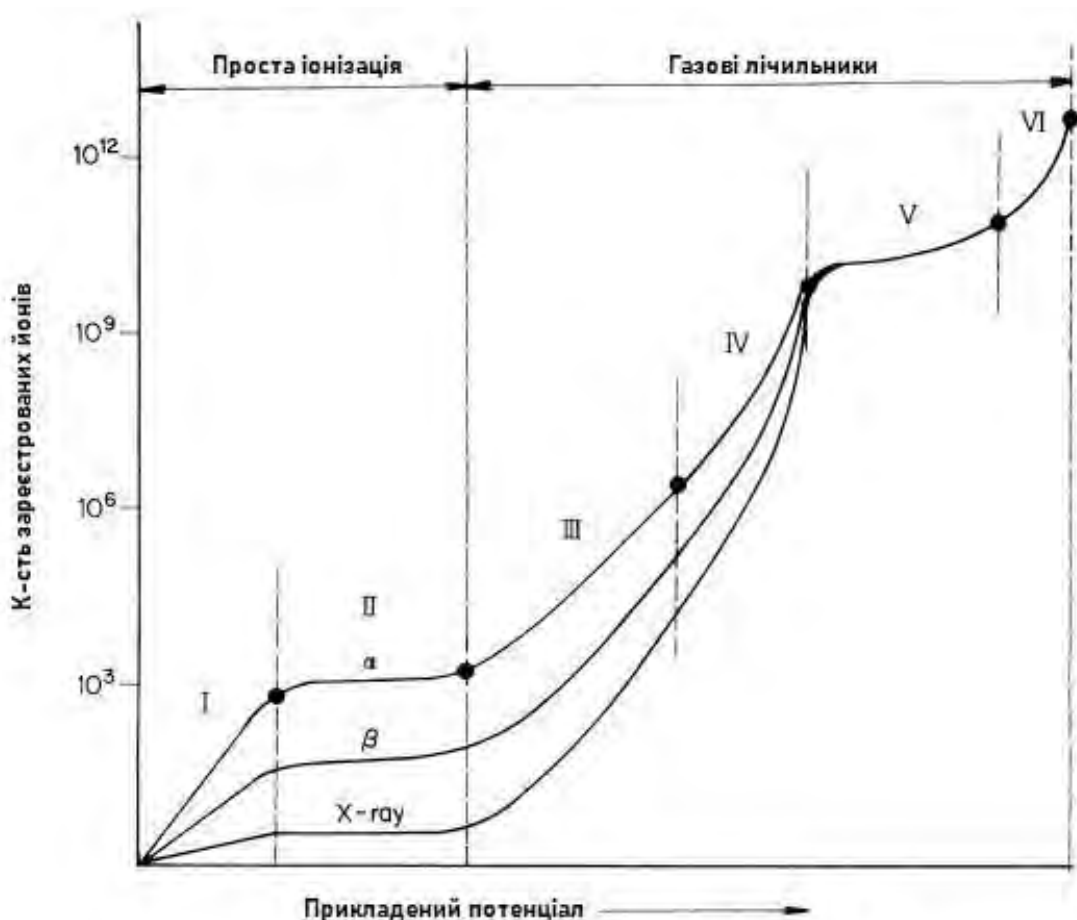


Рисунок 1.3.7. Зв'язок між розміром виробленого імпульсу та потенціалом, прикладеним до електродів газової іонізаційної камери, підданої альфа-, бета- та гамма-випромінюванню.

Різні області позначені римськими цифрами наступним чином:

область I - область рекомбінації;

область II - область простої іонізації; область III - пропорційна область;

область IV - обмежена пропорційна область;

область V - область Гейгера–Мюллера;

область VI - область безперервного розряду.

В області I електричне поле недостатньо сильне і розряд в середині трубки практично не відбувається. При низькій напруженості поля багато іонних пар, що повільно мігрують, все ще мають можливість рекомбінувати. Ця область рекомбінації не використовується для детекторів радіоактивності

Зі збільшенням напруги на електроді збирається більше іонів і електронів, утворених іонізуючим випромінюванням. Нарешті досягається напруженість поля, при якій іони, що швидко мігрують, не мають можливості рекомбінувати. Таким чином, досягається область насичення, де утворюється максимально можлива кількість іонів. Початок ділянки плато позначено областю II і безпосередньо за областю рекомбінації (I), напруга достатня, щоб дозволити зібрати всі вивільнені заряди, які дрейфують до анода внаслідок чого катод виробляє невеликий сигнал, пропорційний самому вивільненому заряду. Це називається *режим роботи іонізаційної камери* [8].

В області III напруга достатньо висока, щоб викликати подальшу іонізацію прискореними електронами біля дроту. У просторі кількох міліметрів від дроту розвивається лавина пар електронів, причому кількість утворених таким чином пар пропорційна кількості виявлених електронів. Це, фактично, підсилення заряду та робочий режим пропорційного лічильника з коефіцієнтами підсилення, які можуть досягати значень порядку 10^6 .

В області IV в газі відбувається розряд, тобто ланцюгова реакція багатьох лавин навколо всього анода. У цьому режимі кожна реакція з іонами видає однакову амплітуду вихідного сигналу незалежно від початкової вкладеної енергії. Розряд викликаний головним чином фотонами, випромінюваними в результаті збудження атомів або молекул. Детектори, які працюють у цій області,

називаються лічильниками Гейгера. Для напруги вище регіону IV, розряд відбувається безперервно з падаючим випромінюванням або без нього.

Іонізаційні камери та лічильники Гейгера сьогодні широко використовуються для виявлення випромінювання.

Загальні міркування, які керують вибором відповідного детектора іонізації газу, включають низький потенціал іонізації та мінімізацію процесу рекомбінації електронів (Вибір газу для детектору відбувається з міркувань мінімізації можливості утворення процесу рекомбінації)

Розряд пригнічується додаванням вуглеводню (такого як етан, метан або CO_2).

З іншого з боку, швидкість дрейфу, яка впливає на мертвий час, зайнятість детектора і має вирішальне значення в деяких програмах, може бути оптимізована за допомогою вибору спеціальної присадки газів. Нарешті, додавання алкоголю в невеликих відсотках, знижує накопичення залишків на анодному дроті. Фактичні гази, які використовуються для іонізації є відповідною сумішшю вищевказаних компонентів, залежно від конфігурації детектора та об'єктів його вимірювання.

1.4. Огляд лічильників Гейгера

Лічильник Гейгера – це пристрій, що використовується для виявлення і вимірювання всіх видів випромінювання: α -, β - та γ -випромінювання. В основному він складається з пари електродів, оточених газом. Між ними протікає висока напруга. Зазвичай використовується газ: гелій або аргон. Коли радіоактивні частинки діють на трубку, то газ, що в середині може вступати в реакцію іонізації. Іони (і електрони) притягуються до електродів і виробляється електричний струм. Лічильник підраховує імпульси струму, і кожен отриманий імпульс означає, що випромінювання іонізує газ. (рисунок 1.4.1)

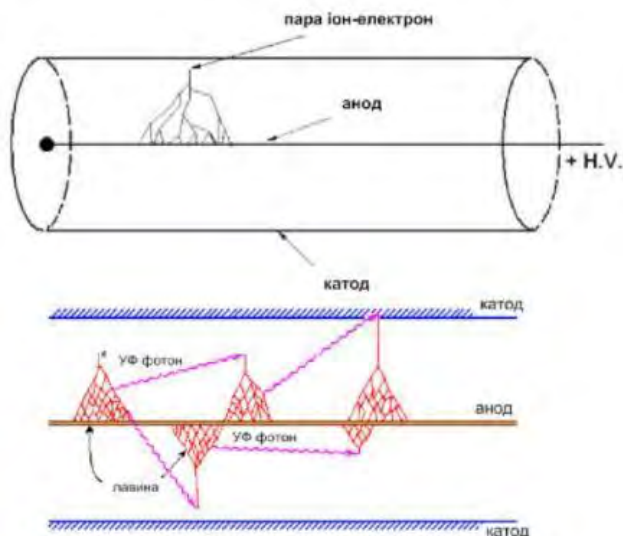


Рисунок 1.4.1. Принцип дії газорозрядних лічильників

Апарат складається з двох частин: трубки та (лічильник + джерело живлення). Трубка Гейгера-Мюллера, як правило, циліндрична, з дротом по центру. Лічильник + джерело живлення, мають регулятори напруги та опції таймера [9].

Коли іонізуюче випромінювання, таке як α -, β - чи γ - частинка, потрапляє в трубку, він може іонізувати деякі молекули газу в трубці. В процесі іонізації електрон вибивається з атома, тому атом стає позитивно зарядженим іоном. Висока напруга створює електричне поле всередині трубки. Електрони, які були вибиті з атома, притягуються до позитивного електрода, а позитивно заряджені іони притягуються до негативного електрода. Це створює імпульс струму в проводах, що з'єднують електроди. Після чого ці імпульси підраховуються, заряджені іони нейтралізуються, а лічильник Гейгера -готовий записати ще один імпульс.

Для того, щоб лічильник Гейгера відновився швидко в початковий стан після вступу випромінювання, в трубку додають газ. Для правильного використання лічильника Гейгера потрібно мати відповідну напругу. Якщо напруга занадто низька, то електричне поле в трубці буде занадто слабке, що призведе до низького імпульсу струму, тому лічильник його не зможе порахувати. Якщо напруга занадто висока, то всередині трубки розряди можуть виникати без упину, і трубка може бути пошкоджена. Трубки більшого діаметру, вимагають

більшої напруги для отримання необхідного електричного поля всередині [9].

Діапазон напруги при якому швидкість реєстрації лічильником імпульсів струму вийде на стабільний рівень - називається "діапазон плато". Якщо подавати занадто високу напругу більше 400В, то в результаті буде безперервний розряд, що унеможливить детектування іонізуючого випромінювання, оскільки результати будуть некоректними. Правильна робота детектора - це коли напруга знаходиться в області плато кривої. Для найкращої роботи, напругу слід вибирати досить близько до порогової напруги, тобто в першій 1/4 шляху області плато. Насправді, нахил кривої в районі плато - це показник якості лічильника Гейгера. Для хорошого лічильника, крива повинна зростати зі швидкістю менше 10 відсотків на 100 вольт. (рисунок 1.4.2)

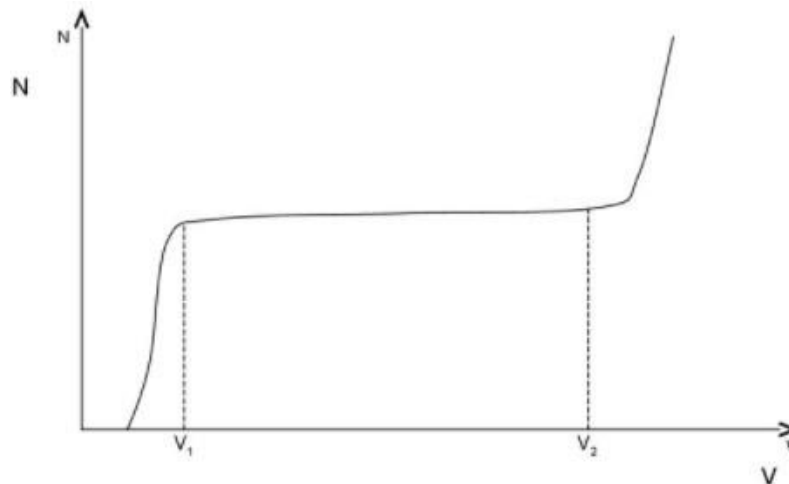


Рисунок 1.4.2. Область Плато

$$\frac{N(V_2) - N(V_1)}{(N(V_1) + N(V_2))} \cdot \frac{2}{V_2 - V_1} * 100\% \quad (1.4.1)$$

Це, для зміни на 100 вольт ((швидкість підрахунку) / (середня швидкість підрахунку) повинна бути меншою ніж 0,1. Ідеальна трубка Гейгера повинна мати нахил кривої в області плато до 3% на 100 вольт.

1.4.1. Ефективність лічильника Гейгера:

Ефективність детектора задається відношенням:

$$\varepsilon = \frac{\text{кількість випромінюваних частинок, які виявлено}}{\text{кількість частинок випромінювання}} \quad (1.4.2)$$

На практиці доведено, що лічильник Гейгера ефективний не для всіх видів випромінювань, і зараз ми розглянемо чому

Причина низької ефективності трубки GM для реєстрації α -частинок полягає в тому, що для поглинання енергії використовується газ. Оскільки α -частинки не дуже енергійні, вони просто не зможуть пройти через зовнішню оболонку трубки, яка містить газ, і вступити в реакцію іонізації. Тому такий чутливий елемент недостатньо чутливий до α -частинки. Але β -частинки досить енергійні, тому вони проходять через зовнішню оболонку і потрапляють у трубку, вони мають найвищий шанс викликати іонізацію. Лічильник Гейгера також нечутливий до γ -променів, так як вони занадто енергійні і навіть не встигають вступити в реакцію іонізації [5].

Отже, хоча лічильник Гейгера може виявляти всі три типи випромінювання, він найбільш ефективний для β -частинок і не дуже ефективний для γ -променів.

Деякі з переваг використання лічильника Гейгера:

- А) Він відносно недорогий;
- б) Довговічний і легко інтегрований;
- с) Він може виявляти всі типи випромінювання.

Основні недоліки лічильника Гейгера:

- а) Він не може розрізнити тип випромінювання, що виявляється;
- б) Він не може бути використаний для визначення точної енергії виявленого випромінювання;

Але для вирішення проблеми необхідно: виміряти загальний фон навколишнього середовища, розрахувати накопичену дозу, стежити за еволюцією радіаційного фону в часі.

1.4.2. Час відновлення

Розглянутий механізм розряду в лічильнику показує, що після кожного розряду лічильник деякий час залишається нечутливим до заряджених частинок. У міру просування іонів до катода електричне поле поблизу анода зростає, і, нарешті, коли вони досягають відстані r_c від центру лічильника, поблизу анода

знову можливий розряд. Час, від попереднього розряду, до моменту, коли іони досягнуть радіусу $r = r_c$, називають *мертвим часом*. Просторовий заряд іонів впливає на поле поблизу анода і за відстані $r > r_c$. Поки цей вплив помітний, імпульси від частинок, що потрапили в лічильник у цей час, мають менші амплітуди. Час, протягом якого частки реєструються, але амплітуди імпульсів мають менші значення називають *часом відновлення лічильника* [5].

Час руху іонів від анода до $r = r_c$ в лічильниках, що самогасяться, близько 250 (мкс) . Осцилограма імпульсів лічильника показана на (рисунку 1.4.3).

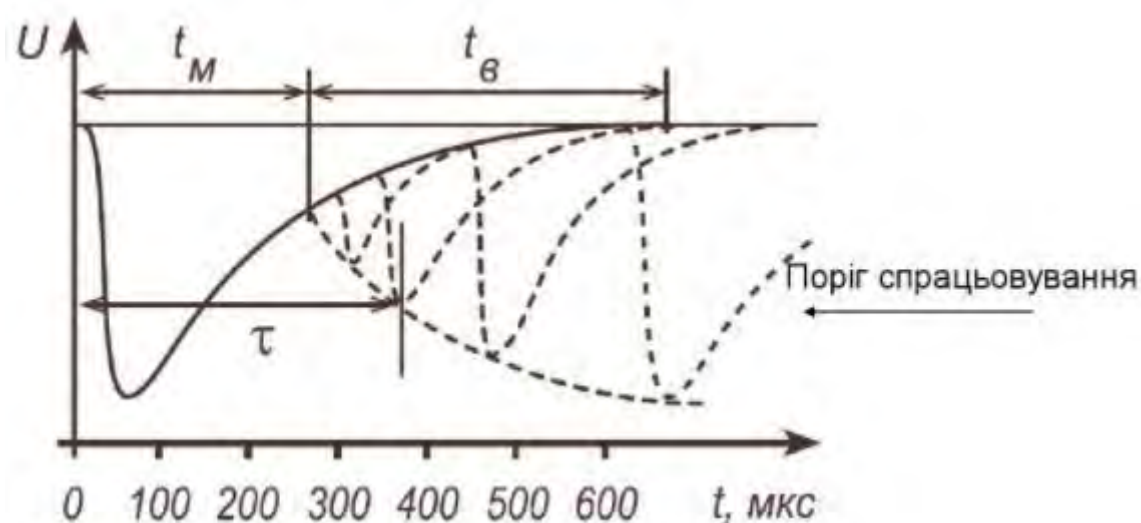


Рисунок 1.4.3. Мертвий час лічильника газорозрядного типу

З рисунка видно, що протягом мертвого часу t_m немає накладання імпульсів. Мертвий час лічильника залежить від кількості розрядів, що відбуваються в ньому за одиницю часу. Зі зростанням швидкості рахунку, мертвий час зменшується і може досягати приблизно 10 (мкс). Це можна пояснити так: чим більша швидкість рахунку, тим більше розрядів починається протягом часу відновлення, тим більше імпульсів мають меншу амплітуду. Частинки, що потрапили в лічильник протягом часу відновлення, створюють меншу щільність заряду, і, отже, іонам потрібно пройти меншу відстань від анода, щоб поблизу анода знову була напруженість поля, достатня у розвиток розряду. Значення t_m визначається також рівнем дискримінації схеми реєстрації. Чим нижчий рівень дискримінації, тим менше значення t_m може бути за більших швидкостей рахунку.

1.4.3. Корекція для часу відновлення:

Для корекції часу відновлення необхідно оперувати наступними параметрами:

t_m – мертвий час детектора або час відновлення;

t_r – фактичний час роботи детектора, що не залежить від мертвого часу детектора;

t_b – час реєстрації активних частинок детектором.

Дане значення – це фактично час обробки одного імпульсу (розряду), на який впливає час відновлення детектора.

C – кількість розрядів, які ми зафіксували.

n - підрахована швидкість лічби,

$$n = \frac{C}{t_r} \quad (1.4.3.1)$$

N - реальна швидкість лічби,

$$N = \frac{C}{t_b} \quad (1.4.3.2)$$

Співвідношення між підрахованим та реальним значенням буде виглядати так $\frac{n}{N}$ дорівнює:

$$\frac{n}{N} = \frac{C/t_r}{C/t_b} = \frac{t_b}{t_r} \quad (1.4.3.3)$$

Інакше кажучи – це співвідношення фактичного часу роботи до часу реєстрації детектора. Фактично це співвідношення – це час запуску підрахунку детектора. Щоб не аналізувати значення, було виведено апроксимаційний вираз:

$$t_b = t_r - C * t_m \quad (1.4.3.4)$$

$C * T$ - це час, протягом якого детектор не реєструє іонізуюче випромінювання. Ми можемо виразити N за допомогою значень n і T , для початку поділимо друге рівняння на значення t_r :

$$\frac{t_r}{t_b} = 1 - \frac{C * t_m}{t_r} = 1 - n * t_m \quad (1.4.3.5)$$

Підставивши перше рівняння - визначимо:

$$\frac{n}{N} = 1 - n * t_m \quad (1.4.3.6)$$

Виразимо N:

$$N = \frac{n}{1-n \cdot t_m} \quad (1.4.3.7)$$

Виходячи з виразу бачимо, що реальна швидкість лічби завжди більша за підраховану.

Також зауважимо, що добуток $n \cdot t_m$ є ключовим параметром, який визначає, наскільки збільшується фактична швидкість лічби порівняно з виміряною швидкістю лічби. При малих значеннях добутку $n \cdot t_m$ дробове збільшення буде незначним, тому $N \approx n$. Для значень продукту $n \cdot t_m < 0,01$ похибка становить менше 1%, тому нею можна знехтувати. Мертвий час починає значно змінювати вимірне значення похибки швидкості рахунку на 5%, коли добуток $n \cdot t_m \geq 0,05$ [10].

Використання методу залежної лічби для вимірювання інтенсивності дози

Загальноприйнятим способом вимірювання інтенсивності дози іонізуючого випромінювання, є підрахунок кількості імпульсів, що виходять з блоку детектування протягом постійного інтервалу часу. Однак тоді відносна похибка вимірювання коливається в широкому діапазоні, що не завжди виправдано.

Використання методу залежних чисел дозволяє позбутися цього недоліку. Суть методу полягає в тому, що вимірюється час, протягом якого на виході блоку детектування реєструється певна задана кількість імпульсів. Це значення визначає середнє значення відносної похибки вимірювання. Потім за допомогою необхідних математичних операцій знаходять значення інтенсивності дози в заданих одиницях вимірювання. Враховуються значення чутливості блоку детектування і його мертвий час. Остання обставина дає змогу значно розширити динамічний діапазон дозиметрів.

Час вимірювання при використанні методу залежних чисел обмежується двома факторами. З одного боку, це мінімальний час, необхідний для зчитування результатів вимірювання індикатора. З іншого боку, це максимально необхідний час, протягом якого забезпечується задана похибка при мінімально

можливих (фонових) значеннях потужності дози [5].

На рисунку 1.4.3 наведено один із варіантів структурної схеми дозиметричного приладу, що складається з блоку детектування ДБ, генератора дискретної частоти ГЗЧ, процесора Пр і блоку індикації БІ .

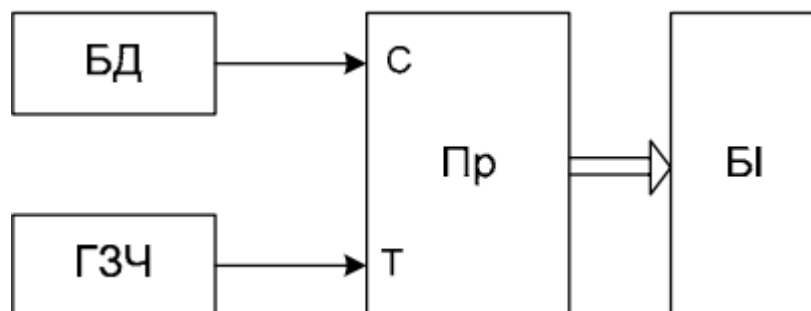


Рисунок 1.4.3. Структурна схема пристрою

Висновки до розділу

В даний час дійсно існує глобальна загроза радіаційного забруднення для простих громадян. Контролювати ситуацію забруднення навколишнього середовища завдяки використанню смартфона та компактних і багатофункціональних сенсорів дуже зручно та доречно.

Необхідно вирішити задачу із створенням портативного дозиметру із удосконаленим алгоритмом реєстрації, обробки та візуалізації сигналу.

Перед вибором детектору реєстрації випромінювання, було визначено цілі та задачі, які має вирішувати розроблений пристрій. Після чого було розпочато аналіз існуючих детекторів реєстрації іонізуючого випромінювання. Аналіз показав, що трубка Гейгера СБМ-20 дозволяє задовольнити точність та роздільну здатність, а також допомагає вирішити поставлені задачі. Точність виміру радіоактивних частинок, яка притаманна даному детектору нам підходить, оскільки немає цілі виміряти точне значення радіації, а потрібно мати розуміння про рівень зміни фону в навколишньому середовищі. Даний датчик допомагає ефективно реєструвати β – частинки та γ -промені.

Обравши датчик було проведено дослідження його мертвих зон та часу відновлення, щоб врахувати ці значення при розробці алгоритму обробки сигналів. Врахувавши та дослідивши всі особливості, можна написати програмне забезпечення, яке буде обробляти шуми отримані з датчику та компенсувати значення мертвої зони. Також можливо створити алгоритм для підрахунку доз випромінювання з обробленого сигналу, який буде забезпечувати високу швидкість реакції при вищих значеннях доз іонізованого випромінювання, ніж при низьких.

В майбутньому, доукомплектувавши пристрій ще одним чи кількома датчиками, можна розширити діапазон вимірювання та підвищити значення точності вимірюваних значень.

РОЗДІЛ 2. РОЗРОБКА ТА ПІДБІР ВУЗЛІВ СТРУКТУРНОЇ СХЕМИ

Датчик Гейгера-Мюллера СБМ-20 здатний до реєстрацій жорсткого β - і γ -випромінювання та широко застосовується в галузі радіаційного контролю.

Деякі параметри СБМ-20:

- чутливість лічильника: $56 \div 70$ (імп/мкР);
- значення робочої напруги: 400 (В);
- мертвий час після імпульсу: 190 (мкс);
- значення частоти імпульсів: 5263 (Гц).



Рисунок 2.1 Трубка Гейгера СБМ-20

Робоча напруга – 400 (В)

Інтервал робочої напруги – 350 – 475 (В)

Граничні значення експозиційних доз для γ -променів:

$$P_{min} = 0.004 \text{ (мкР/сек)}$$

$$P_{max} = 40 \text{ (мкР/сек)}$$

Інертність трубки становить не більше 10 (імп/хв)

Рівень амплітуди мінімального імпульсу – 50 (В)

Значення струму при розряді становить – 20 (мкА)

Ресурсна характеристика роботи трубки – $2 \cdot 10^{10}$ (імп)

Розрахункові значення дози відносно зареєстрованих значень імпульсів

В документації трубки Гейгера СБМ-20, надані загальні значення для калібрування лічильника відносно конкретних радіоактивних речовин:

Для елементу Ra-226 , значення зареєстрованих кількості імпульсів складає - 29 (імп/с) = 1(мР/год);

Чутливість до випромінювання кобальту Co-60 становить 22 (імп/с) = (1мР / год).

Нижче приведена формула для розрахунку значення накопиченої дози в $\frac{\text{мкЗв}}{\text{год}}$ відносно зареєстрованої кількості імпульсів

$$[\text{кількість імпульсів}] \text{ імп} = \left[\frac{\text{мкЗв}}{\text{год}} \right].$$

Значення коефіцієнта поглинання, яке слід використовувати, становить 8,77 для повітря:

$$1 \frac{\text{мР}}{\text{год}} = 8.77 \frac{\text{мкЗв}}{\text{год}}$$

Відповідно до відомих значень виразимо рівень радіації у мкЗв:

$$29 \frac{\text{імп}}{\text{с}} = 1 \frac{\text{мР}}{\text{год}}$$

Розрахуємо кількість значень за годину:

$$29 * 60 = 1740 \frac{\text{імп}}{\text{хв}} = 1 \frac{\text{мР}}{\text{год}}$$

Перетворимо рівень радіації в мкЗв:

$$\frac{1740}{8.77} = 198 \frac{\text{імп}}{\text{хв}} = 1 \frac{\text{мкЗв}}{\text{год}}$$

Отже, $1 \frac{\text{імп}}{\text{хв}}$ рівнозначний значенню дози радіації в $\frac{\text{мкЗв}}{\text{год}}$ буде:

$$1 \frac{\text{імп}}{\text{хв}} = \frac{1}{198} = 0,00504 \frac{\text{мкЗв}}{\text{год}}$$

Отже:

$$\frac{\text{імп}}{\text{хв}} = \text{імп} * 0,00504 \frac{\text{мкЗв}}{\text{год}}$$

Калібровка пристрою відносно кобальту Co-60:

$$22 \frac{\text{імп}}{\text{с}} = 1 \frac{\text{мР}}{\text{год}}$$

Значення за хвилину:

$$22 * 60 = 1320 \frac{\text{імп}}{\text{хв}} = 1 \frac{\text{мР}}{\text{год}}$$

Перетворимо рівень радіації в мкЗв:

$$\frac{1320}{8.77} = 150 \frac{\text{імп}}{\text{хв}} = 1 \frac{\text{мкЗв}}{\text{год}}$$

Отже, $1 \frac{\text{імп}}{\text{хв}}$ рівнозначний дозі радіації в $\frac{\text{мкЗв}}{\text{год}}$:

$$1 \frac{\text{імп}}{\text{хв}} = \frac{1}{150} = 0,006644 \frac{\text{мкЗв}}{\text{год}}$$

Отже:

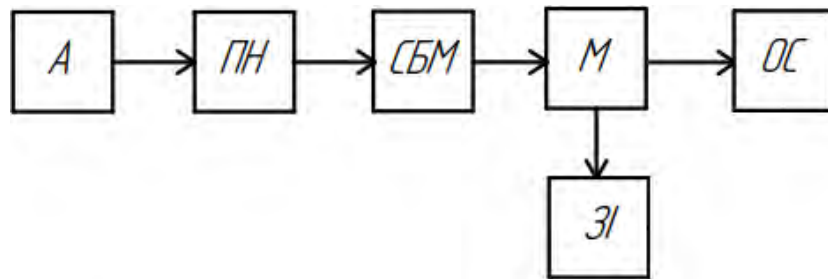
$$\frac{\text{імп}}{\text{хв}} = \text{імп} * 0,006644 \frac{\text{мкЗв}}{\text{год}}$$

Для лічильника Гейгера із датчиком СБМ-20 коефіцієнти перерахунку становлять: $\frac{\text{імп}}{\text{хв}} = 0,00504 \frac{\text{мкЗв}}{\text{год}}$ відносно радію Ra-226 та $\frac{\text{імп}}{\text{хв}} =$

$0,00664 \frac{\text{мкЗв}}{\text{год}}$ відносно кобальту Co-60. Дані виведені значення будуть враховуватися при реалізації алгоритму реєстрації дози випромінювання.

Дана система реєструє рівень випромінювання, передає отримані значення на смартфон, де встановлений додаток, який аналізує сигнал, обраховує та візуалізує значення дози випромінювання. При перевищенні допустимого значення дози – додаток сигналізує про наявність небезпеки [10].

Структурна схема автоматизованого пристрою моніторингу зображена на рисунку 2.2



A – Акумулятор
ПН – Перетворювач Напруги
СБМ – Трубка СБМ-20
М – Модулятор
ОС – Операційна Система Android
ЗІ – Зовнішні Індикатори

Рисунок 2.2. Структурна схема автоматизованого пристрою моніторингу випромінювання

Автоматизований пристрій включає в себе три основні системи:

- система реєстрація випромінювання;
- система обробки даних;
- система візуалізації оброблених даних.

Функціональна схема автоматизованого пристрою зображена на рисунку 2.3.

Схема реалізованого пристрою складається з простих компонентів, які допомагають зняти показання імпульсів із трубки Гейгера. Отримані значення передаються на смартфон, де проходять первинну обробку. Швидкість оновлення інформації про рівень радіації залежить від її рівня: чим вищий рівень радіації – тим частіше оновлення даних в додатку. Цьому сприяє удосконалений алгоритм підрахунку дози. Пристрій виходить компактний та недорогий при створенні.

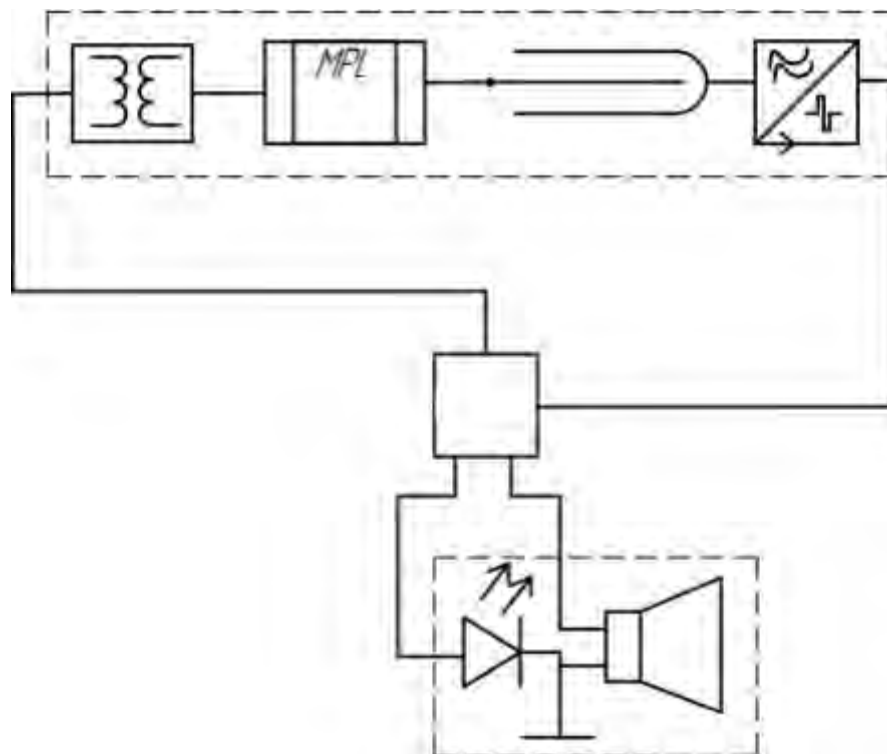


Рис. 2.3. Функціональна схема автоматизованого приладу

РОЗДІЛ 3. РОЗРОБКА ПРОГРАМНИХ АЛГОРИТМІВ

3.1 Опис роботи програми

При розробці магістерської дисертації написано програмне забезпечення, яке допомагає в обробці отриманих сигналів та візуалізації інформативної складової. Дане ПЗ допомагає зчитати аналоговий сигнал імпульсів струму, що надходить від детектору реєстрації іонізуючого випромінювання. В подальшому цей сигнал за допомогою АЦП телефону перетворюється в цифровий. Після чого застосовується оптимізований алгоритм обробки отриманих даних та перетворення у відповідні значення доз випромінювання. Результати вимірів візуалізуються на екрані смартфона та відправляються на віддалений сервер для більш глибокої обробки та аналізу даних.

При розробці програмного забезпечення було застосовані мова програмування C#, технологія розробки мобільних додатків Xamarin.

Xamarin.Forms представляє платформу, яка націлена на створення кросплатформових додатків під Android, iOS та Windows. Є певні статистичні дані, що значна частина мобільних програм створюється більш ніж для однієї платформи, наприклад, для Android та iOS. Однак неминуче розробники стикаються з такими труднощами:

- Відмінність у підходах побудова графічного інтерфейсу. Розробники змушені підлаштовувати програму під вимоги до інтерфейсу на конкретній платформі;
- різні API - відмінність у програмних інтерфейсах і реалізаціях тих чи інших функціональностей також вимагає від програміста врахування цих специфічних особливостей;
- різні платформи для розробки. Наприклад, щоб створювати програми для iOS нам необхідне відповідне середовище - Mac OS X і ряд спеціальних інструментів типу XCode. А як мову програмування вибирається Objective-C або Swift. Для Android ми можемо використовувати найрізноманітніший набір

середовищ - Android Studio, Eclipse і т.д. [12].

- а для створення програм під Windows використовується Visual Studio, та мови - C#, F#, VB.NET, C++.

Такий діапазон платформ, засобів розробки та мов програмування не може позитивно позначатися на термінах створення додатків, і, зрештою, на коштах, що виділяються на розробку. Було б дуже ефективно мати один інструмент, який дозволяв би легко і просто створювати програми відразу для всіх платформ. І саме таким інструментом і є платформа Xamarin.

Xamarin дозволяє створювати одну єдину логіку програми із застосуванням C# і .NET відразу для всіх трьох платформ - Android, iOS, UWP.

Переваги використання Xamarin.Forms:

- У процесі розробки створюється єдиний код для всіх платформ;
- Xamarin надає прямий доступ до нативних API кожної платформи;
- При створенні додатків ми можемо використовувати платформу.

Xamarin Forms підтримує декілька платформ.

Основні платформи: Android, iOS, UWP, Tizen.

Додаткові платформи, що розвиваються спільнотою: MacOS, WPF, GTK# (рисунок 3.1).

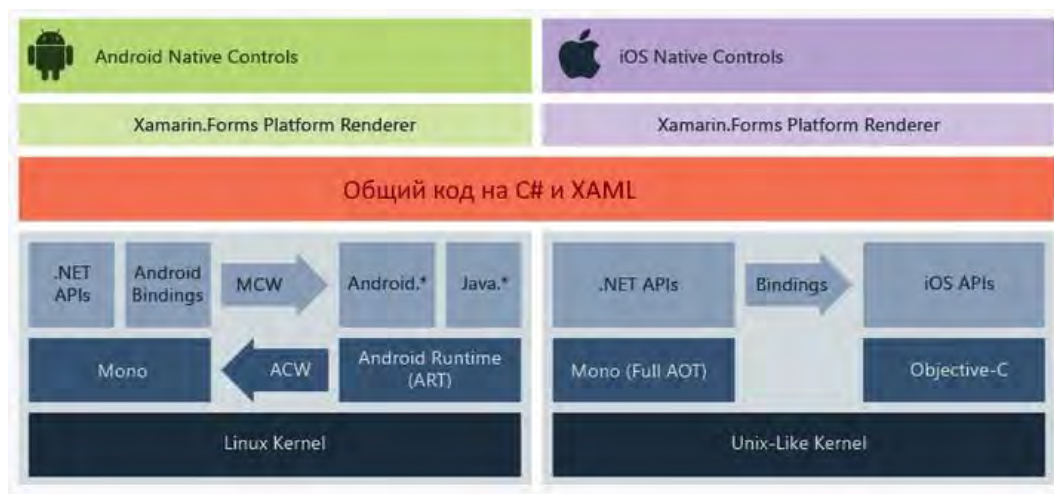


Рисунок 3.1. Взаємодія платформ в Xamarin

Хamarin працює поверх фреймворку Mono, який надає opensource-реалізацію .NET Framework. Mono може працювати поверх різних платформ – Linux, MacOS тощо.

На рівні кожної окремої платформи Xamarin покладається на низку субплатформ. Зокрема:

- Xamarin.Android - бібліотеки для створення програм на ОС Android
- Xamarin.iOS - бібліотеки для створення програм для iOS

Ці субплатформи відіграють велику роль - через них програми можуть надавати запити до прикладних інтерфейсів на пристроях під керуванням ОС Android або iOS.

За допомогою Xamarin.Android код C# з використанням Xamarin компілюється в Intermediate Language, який після запуску програми компілюється в нативну збірку. Xamarin-програми запускаються в середовищі виконання Mono. Безпосередньо код не може звертатися до API Android. Для цього треба звернутися до функціональності просторів імен Android і Java, які надаються віртуальною машиною Android Runtime. Спеціальний прошарок Managed Callable Wrappers дозволяє транслювати виклики managed-коду в нативні виклики та звертатися до функціональності просторів імен Android та Java.

І навпаки, коли Android Runtime звертається до програми з кодом Xamarin, всі виклики проходять через обгортку Android Callable Wrappers.

Програми Xamarin.iOS на відміну від Xamarin.Android, який використовує JIT-компіляцію, застосовують AOT-компіляцію коду C# в нативний ARM-код. Xamarin використовує проміжний шар Selectors для трансляції викликів коду Objective-C код на C# і шар Registrars для трансляції коду C# Objective-C. У результаті шари Selectors і Registrars загалом представляють переміжний шар, який на ілюстрації вище позначений як "bindings" і який дозволяє взаємодіяти коду Objective-C з кодом C#.

У результаті, завдяки цим платформам ми можемо створювати окремо додатки для Android, окремо для iOS, але найбільш важливою особливістю

Хamarin є можливість створювати кросплатформні додатки – тобто одна логіка для всіх платформ. Дана можливість представлена технологією Xamarin.Forms і яка працює рівнем вище Xamarin.Android і Xamarin.iOS. Тобто за допомогою Xamarin.Forms ми один раз можемо визначити візуальний інтерфейс, один раз до нього прив'язати якусь логіку на С# і все це буде працювати на Android, iOS та Windows. Потім Xamarin.Forms за допомогою рендерерів– спеціальних об'єктів для зв'язку контролів на XAML/С# з нативними контролами транслюють візуальні компоненти Xamarin.Forms у графічний інтерфейс, специфічний для кожної платформи.

Опис мови програмування С# обраної для реалізації додатку

На даний момент мова програмування С# являється однією з найпотужніших мов. Зараз на цій мові пишуться різні програми: від маленьких desktop програм до масштабних веб-порталів і веб-сервісів, якими користуються щодня мільйони користувачів.

С# - це мова, яка пройшла великий шлях, як і вся платформа .NET і стала оптимізованим інструментом.

С# - це мова із Сі-подібним синтаксисом та схожа у цьому відношенні до С++ та Java. С# - це об'єктно-орієнтована мова і в цьому плані багато запозичила у мов Java та С++. Таким чином, С# підтримує поліморфізм, наслідування, статичну типізацію. Об'єктно-орієнтований підхід дає можливість вирішити завдання у побудові великих, але в цей же час дуже гнучких, масштабованих додатків, що розширяються. Мова С# продовжує активно розвиватися та в кожній новій версії з'являється більше цікавих функціональностей.

Опис платформи

Коли хтось говорить С#, часто в контексті мають на увазі різноманітні технології платформи .NET (Windows Forms, ASP.NET, Xamarin та інші). Та навпаки, коли хтось говорить .NET, мають на увазі С#. Проте хоч і ці поняття пов'язані, ототожнювати їх не варто. Мова програмування С# була створена спеціально для роботи з фреймворком .NET, але саме поняття .NET ширше.

Можна виділити такі характеристики платформи .NET:

Підтримка кількох мов програмування. Основою платформи є середовище виконання Common Language Runtime. При компіляції будь-якою з доступних мов компілюється у складання загальною мовою Common Intermediate Language (CIL) - свого роду такий собі асемблер платформи .NET. Тому ми можемо зробити окремі модулі однієї програми використовуючи різні мови програмування.

Кросплатформність. .NET – це платформа, яка виконується на різних ОС (Windows, MacOS, Linux). Застосовуючи різні технології на платформі .NET, ми можемо розробляти програми на мові С# для різноманітних платформ - Windows, Linux, Android, MacOS,.

Потужна бібліотека класів. .NET являє собою одну єдину всім підтримуваним мовам бібліотеку класів. Який би додаток ми не розробляли на С# - текстовий редактор, калькулятор, чат або складний веб-сайт – в будь-якому випадку ми використовуємо бібліотеку класів .NET.

Різноманітність технологій. Середовище виконання CLR та основна бібліотека класів являється основою цілого стеку технологій, які можуть використовуватися розробниками при написанні тих чи інших додатків. Для роботи з базами даних даного стеку технологій призначено інструменти Entity Framework Core і ADO.NET. Для побудови кросплатформових десктопних та мобільних додатків використовують технології Xamarin/MAUI.

Продуктивність. Проведено багато тестів програм написаних на .NET 7 за багатьма показниками сильно випереджають програми, побудовані завдяки іншим технологіям. Програми на .NET 7 характеризуються високою продуктивністю.

Також можна викремити особливість мови С# і фреймворку .NET, як автоматичне видалення сміття. CLR сама контролює процес збирання сміття та очистить пам'ять.

3.2 Дослідження обробки сигналів за допомогою Google Colaboratory

Google Colaboratory – це хмарний сервіс від Google Research. Це IDE, яка дозволяє будь-якому користувачеві писати вихідний код у редакторі та запускати його з браузера. Зокрема, він підтримує мову програмування Python і

орієнтований на завдання машинного навчання, аналіз даних, освітні проекти тощо.

Ця методологія, заснована на Jupyter Notebook, доступна абсолютно безкоштовно за обліковим записом GMail, і не вимагає налаштування, а також не потрібно буде завантажувати або встановлювати Jupyter. Google Colaboratory надає обчислювальні ресурси для редагування та тестування коду. Очевидно, даний сервіс, як і всі безкоштовні сервіси надає обмежені ресурси, але для звичайного аналізу даних наданої потужності вистачає [13].

Коли ми отримуємо доступ до Colab зі своїм обліковим записом, ми отримуємо віртуальну машину, на якій ми можемо запускати свій код ізольовано від інших користувачів та ресурсів. Це означає, що якщо ми виконуємо певний код на віртуальній машині і закриваємо браузер, машини будуть видалені після певного періоду бездіяльності для звільнення ресурсів. Однак, якщо зберегти створену програму до GDrive, то можна з легкістю відтворити її навіть після тривалого часу відсутності взаємодії (рисунк 3.2).

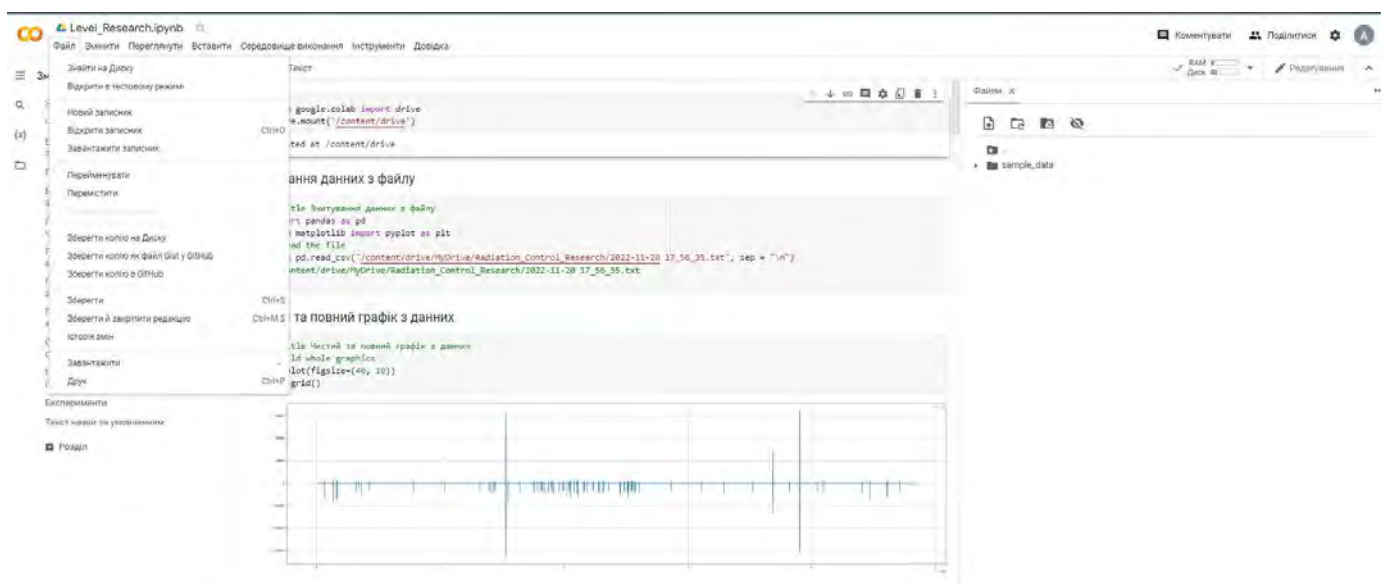


Рисунок 3.2. Візуалізація середовища розробки Google Colaboratory

Найбільшими перевагами Google Colaboratory є:

- Редагування та запуск коду на Python;
- Збереження своїх проектів на диску Google (GDrive), щоб не втратити їх;

- Завантаження коду з GitHub;
- Ділитись блокнотами (текст, код, результати та коментарі);
- Імпортування записників Jupyter або IPython.

3.2.1 Проведення дослідження

Проблема більшості схем дозиметрів на логічних мікросхемах, як аматорських, так і промислових – залежність частоти імпульсів лічильника Гейгера від рівня гамма-фону, приймається лінійно у всьому діапазоні вимірів.

Тобто опорним параметром є час виміру. Це означає, що протягом певного статично заданого часу програма реєструє імпульси, які надходять з чутливого елемента. На базі цього співвідношення, рахується значення доз випромінювання. Даний підхід не є доцільним, оскільки час зміни значень радіації – статичний, а отже людина не відразу дізнається, як радіаційний фон підвищився, а лише через певний статичний проміжок часу. Здебільшого пристрої мають час заміру 30 с, відповідно до формул значення зареєстровані за такий час переводять у кількість значень за хвилину та підставляють в формулу розрахунку дози.

Алгоритм обробки сигналів на базі нашого дозиметру базується не на часі виміру, а на кількості імпульсів. Тобто замість того, щоб дочікуватися статичного значення часу ми дочікуємося статичного значення кількості імпульсів. Даний підхід забезпечує швидше реагування на підвищення рівня радіації.

Наприклад, якщо значення 5-ти імпульсів зареєстровано протягом 7-8 секунд - це показує, що значення радіаційного фону знаходиться в межах норми, приблизно 0,3 (мкЗв/год). А от значення при 5-ти імпульсів за 2,2 секунди показує,

що рівень радіації збільшився приблизно до 0,9 (мкЗв/год), що є суттєвим перевищенням значення норми. Відповідно використовуючи алгоритм з прив'язкою до статичної кількості імпульсів відносно часу – ми збільшимо швидкодію при фіксації підвищеного рівня радіації, що в свою чергу допоможе більш завчасно попередити користувача [14, 15].

3.2.2. Опис алгоритму вирахування миттєвої дози

Спершу прилад підключається через вхід 3.5 (мм) до мобільного телефону, далі вся обробка отриманого сигналу відбувається за допомогою створеного ПЗ на мові програмування C#.

Розберемось як саме ПЗ обробляє вхідний сигнал:

Спершу відбувається перетворення вхідного аналогового сигналу в цифровий, для можливості подальшої його обробки.

Далі ПЗ робить прохід по кожному значенню сигналу, шукаючи ті, які будуть перевищувати порогове значення імпульсу, інакше кажучи шукає в сигналі значення вищі за шуми, таким чином ідентифікуючи імпульс. Порогове значення

імпульсу було знайдено експериментальним шляхом, а саме здійснено низка вимірів, та візуалізувавши сигнали з цих вимірів побачили, яке саме значення потрібно поставити пороговим, щоб відсіяти постійні значення шумів та зберегти при цьому інформативні значення імпульсів. В даному випадку рівень порогового сигналу рівний 2000. Таким чином, наша програма вміє ідентифікувати імпульси.

Далі, при старті процесу вимірювання фіксується значення часу. В даному випадку час фіксується за допомогою отримання параметру в C# `DateTime.Now.Ticks`, він повертає нам теперішній час в форматі так званих «тіків». В одній секунді міститься десять мільйонів таких «тіків». Що надає нам можливість максимально точно отримувати інформацію про момент часу початку реєстрації першого значення сигналу більшого, за порогову величину. Завдяки таким можливостям, імовірність погіршення роздільної здатності відсіюється.

Далі, прилад має ідентифікувати 5 імпульсів. Значення кожного імпульсу реєструється за допомогою ПЗ і після приходу 5-го імпульсу функція знову фіксує момент часу, і вираховує різницю між зафіксованими значеннями при старті виміру та отриманням 5-го імпульсу. Таким чином, ми отримали період часу за який відбулось 5 імпульсів.

Отримавши ці дані, ПЗ вираховує частоту, яку далі по формулі перетворює в значення миттєвої дози. Але так як іонізуючі частинки, літають в хаотичному порядку, і іноді буває таке, що на детектор за досить малий проміжок часу

потрапляє 2-3 таких частинки, а буває що довгий час не фіксується ні одна. Таким чином, якщо ми будемо виводити значення не оброблено зафіксованих імпульсів – то отримаємо у висновку різкі, іноді хаотичні перепади значень дози.

Щоб позбутися такого ефекту, отримані значення потрібно групувати та округлювати. Можна було б збільшити кількість імпульсів для підрахування, але це збільшило б час реакції пристрою на зміну радіоактивного фону, що в кінцевому випадку пагубно вплинуло б на швидкість відображення зміни дози.

Тому краще застосувати алгоритм плаваючого середнього. Він полягає в тому, що фільтр накопичує в собі значення миттєвих доз, їх кількість відповідна до порядку фільтра, і віддає їх середнє. Коли фільтр наповнений, а приходить нове значення, то за допомогою спеціального алгоритму програма замінює найстаріше значення на нове [16, 17]. Таким чином швидкодія реакції на зміну радіоактивного фону залишається високою і зміна значень згладжується, що надає більш угруповані значення. Розраховане значення сприймається за значення миттєвої дози та виводиться на екран.

3.3 Дослідження імпульсів в реальних вимірах

Зібравши пристрій та написавши програму, яка мінімально може реєструвати сигнал та записувати його до файлу, було розпочато процес дослідження.

Знайшовши об'єкт, який випромінює іонізуючі частинки було проведено вимір за алгоритмом: на відстані, біля джерела та віддаляючись від джерела. Файл із зареєстрованим сигналом було завантажено до Google Colaboratory та проведено попередню обробку даних за допомогою мови програмування Python для можливості її візуалізації на графіку. Нижче приведені самі графіки та короткі пояснення до них.



Рисунок 3.3.1. Візуалізація не оброблених даних з файлу за один сеанс виміру

На даному графіку ми спостерігаємо чистий (не оброблений) цифровий сигнал з частотою дискретизації 44100 (Гц), отриманий з нашого дозиметру. На осі X – час, по осі Y – значення амплітуди сигналу. Різкі скачки амплітуди говорять нам про утворення розряду в лічильнику Гейгера-Мюллера, що говорить нам про проходження радіоактивної частинки (рисунок 3.3.1).

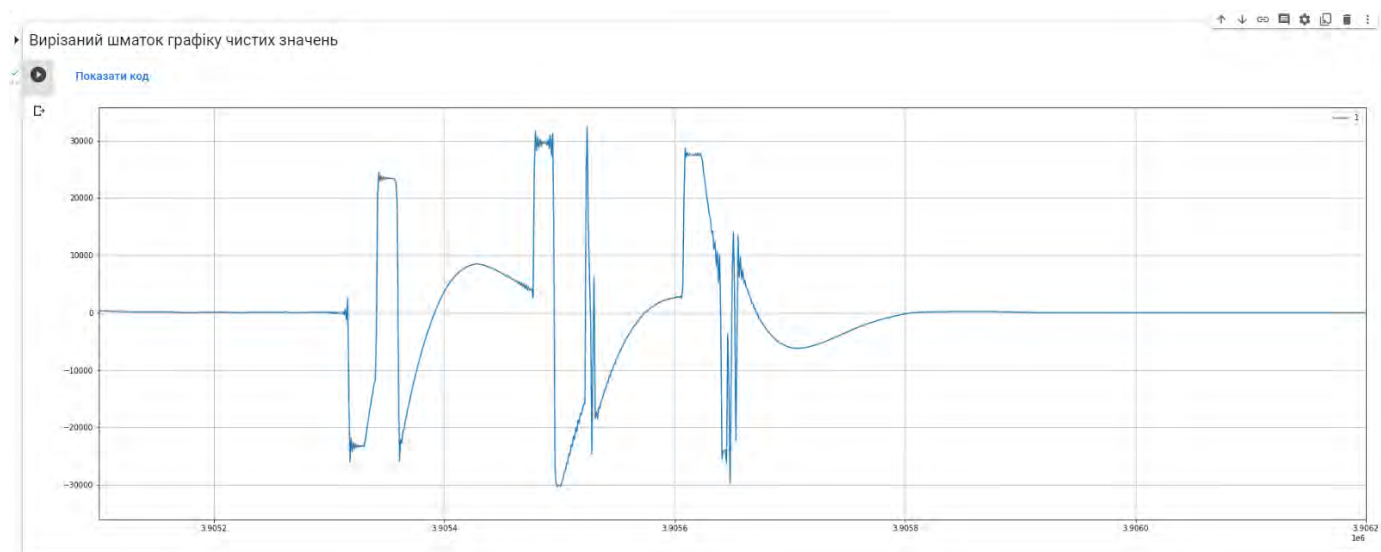


Рисунок 3.3.2. Фрагмент графіку з різкими скачками амплітуди

На даному графіку представлено збільшений розмір фрагменту графіку, де присутні декілька розрядів. Спостерігаємо не чітко зрозумілу картину, що до можливості розрізнення окремих розрядів (рисунок 3.3.2).

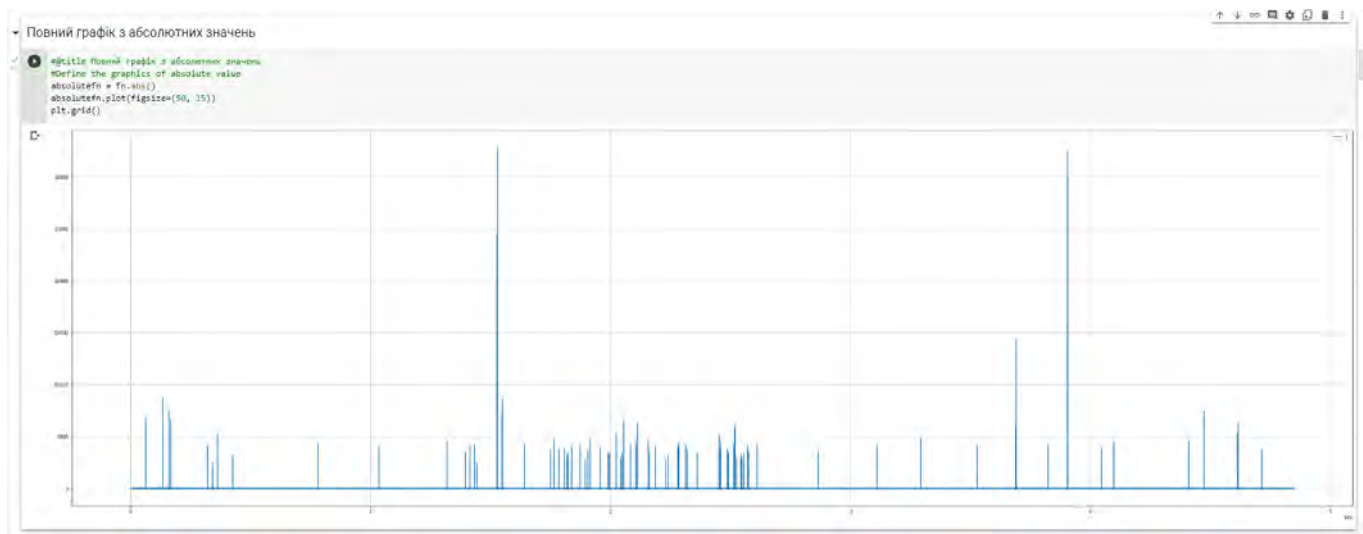


Рисунок 3.3.3. Візуалізація різких стрибків амплітуди в абсолютних значеннях

Для подальшого аналізу сигналу переводимо значення в абсолютні величини (рисунок 3.3.3).

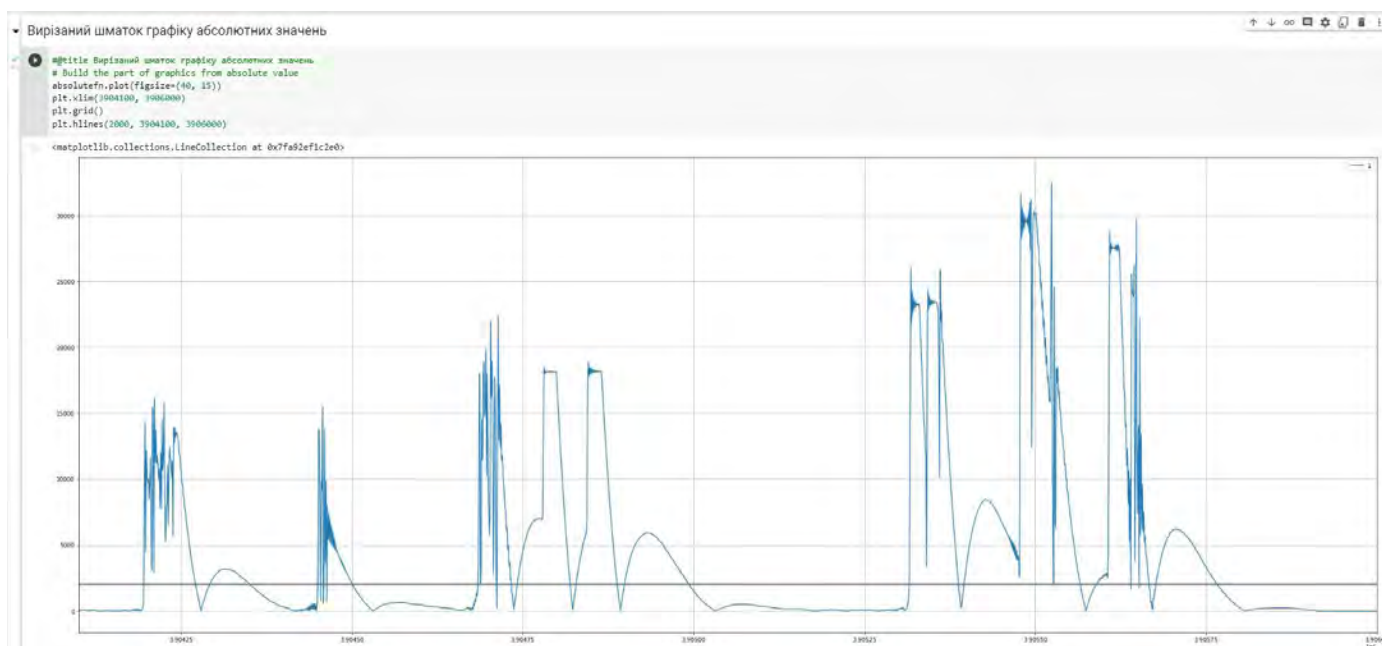


Рисунок 3.3.4. Фрагмент графіку з різкими скачками амплітуди в абсолютних значеннях

На даному графіку представлено, де присутні декілька розрядів. Значення взяті по модулю для кращого розуміння характеристики сигналу (рисунок 3.3.4).

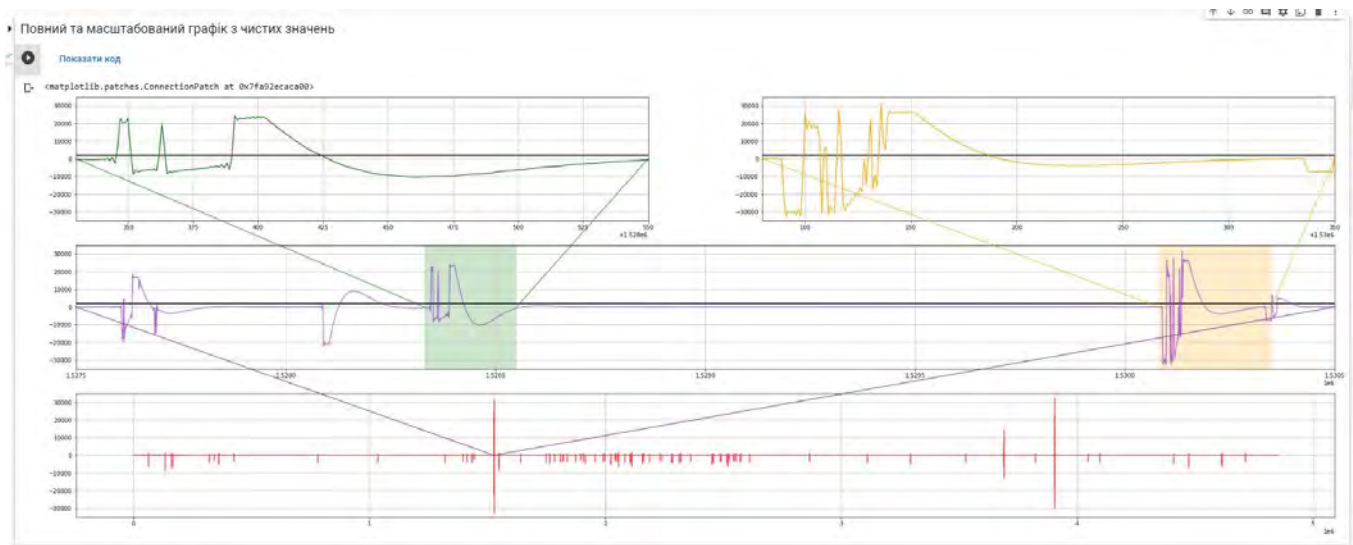


Рисунок 3.3.5 Повний графік з масштабуванням для паралельного дослідження декількох імпульсів



Рисунок 3.3.6 Повний графік з масштабуванням імпульсів в абсолютних значеннях

Даний графік був зроблений для зручності розпізнавання імпульсів при візуальному аналізі вимірів. На найнижчому графіку зображено повний загальний графік сигналу з абсолютних значень. З цього графіку зроблено виноску в центральний графік фрагменту, який ми хочемо дослідити. Та на найвищі графіки винесені окремі значення імпульсів для детального спостереження (рисунок 3.3.5).

Також було написано допоміжні функції, для швидкої обробки даних за

певним алгоритмом, для подальшого перевикористання цих функцій. Зокрема, серед створених функцій: функція застосування до сигналу алгоритму плаваючого середнього значення та функція виокремлення часових проміжків імпульсів з сигналу (рисунок 3.3.6).

Графік сигналу обробленого методом плаваючого середнього

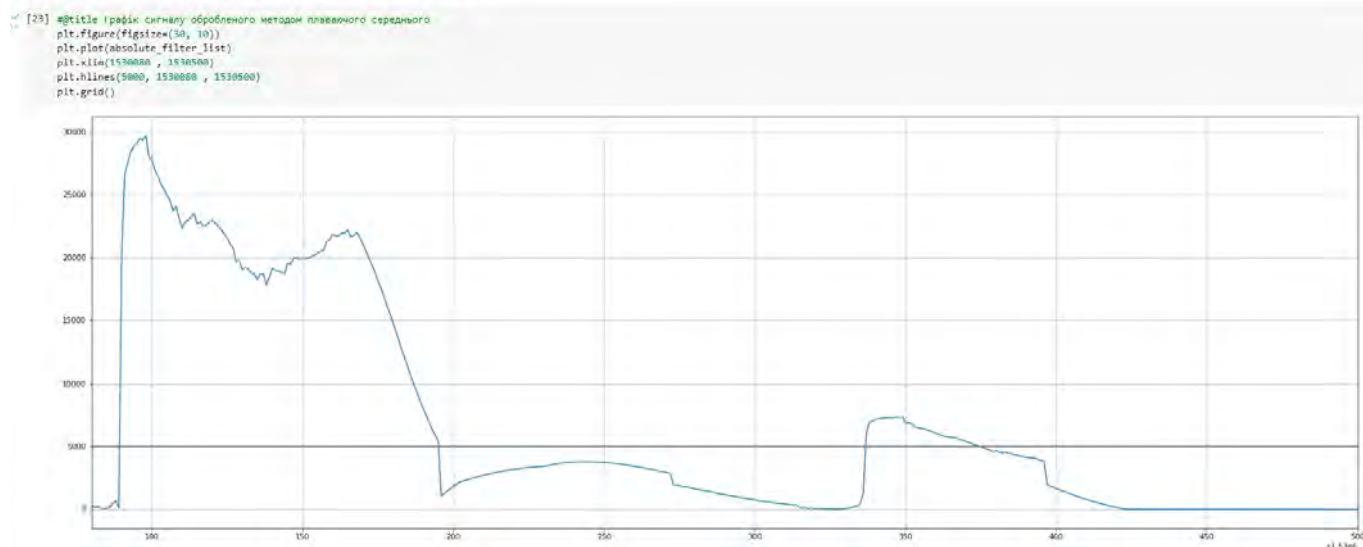


Рисунок 3.3.7. Фрагмент сигналу обробленого алгоритмом плаваючого середнього.

На даному графіку продемонстровано фрагмент імпульсів після застосування алгоритму плаваючого середнього значення. Порівнюючи даний фрагмент з фрагментами вище - сигнал став менш різким. І тепер простіше фіксувати один імпульс, оскільки амплітуда не скаче, а стала більш пологою (рисунок 3.3.7).

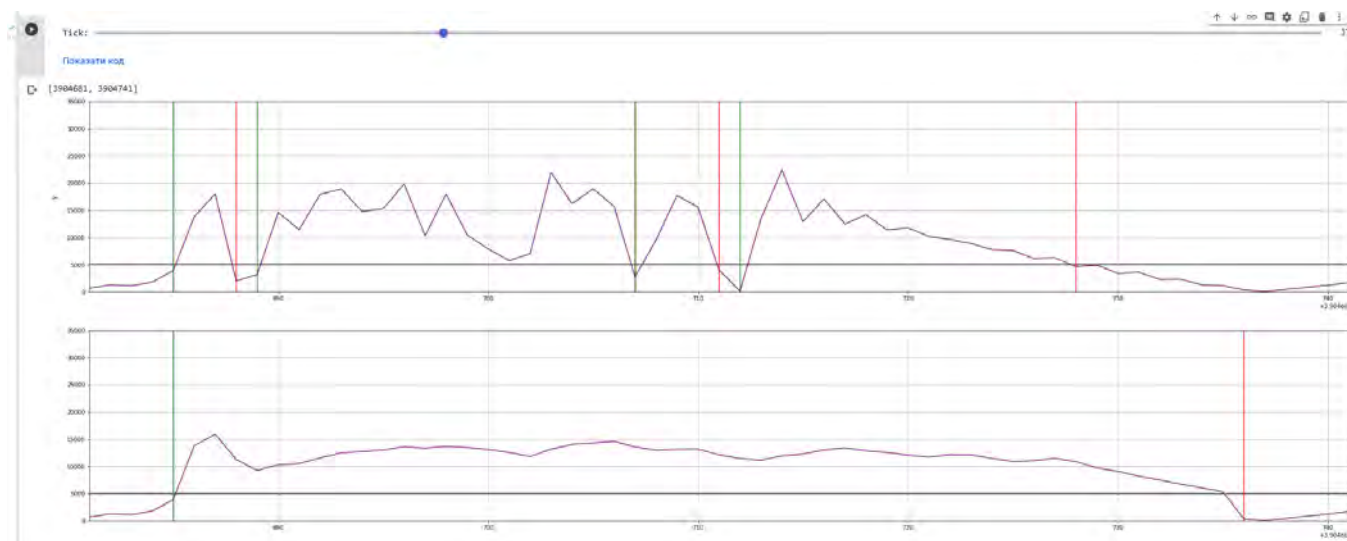


Рисунок 3.3.8. Візуалізація фрагменту обробленого сигналу.

На даному графіку приведено візуалізацію імпульсів до та після обробки алгоритмом плаваючого середнього (рисунок 3.3.1). Наглядно видно, що графік став більш пологим, що дозволяє розрізнити початок та кінець імпульсу. Застосувавши фільтр 15-го порядку, ми змогли зберегти інформативну складову сигналу та виразити граничні значення імпульсів. Тривалість імпульсу рівна:

$$T = \frac{50}{44100} = 0,0011 \text{ (с)}$$

що вказує на середньостатистичну тривалість імпульсу відповідно до документації. Отже розуміємо, що роздільна здатність не втратилася при використанні алгоритму [18, 19, 20].

3.4 Опис роботи додатку

Розберемо, як працює алгоритм роботи ПЗ по схемі вище. Наш дозиметр підключається до мобільного пристрою через порт 3.5 (мм) і передає через нього аналоговий сигнал з дозиметру (рисунок 3.4.1).

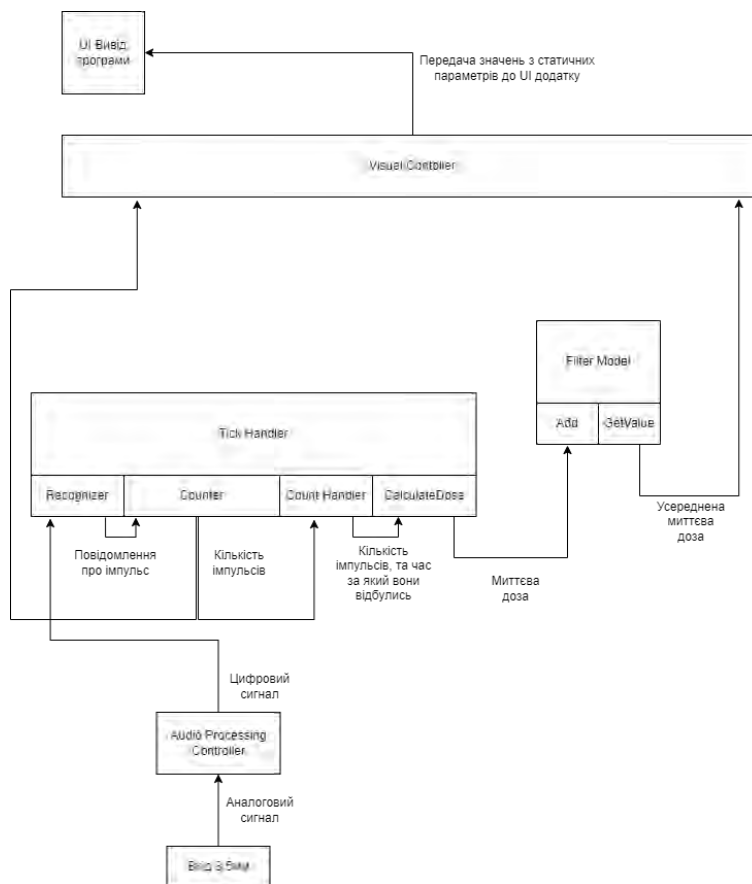


Рисунок 3.4.1 Схематичне зображення алгоритму роботи ПЗ

Далі, сигнал опрацьовує наше ПЗ. Перший контролер - це Audio Processing Controller, він відповідає за зчитування сигналу через порт 3.5 (мм), а також його оцифровування, далі оцифрований сигнал передається наступному керуючому блоку з назвою Tick Handler.

Tick Handler – це клас який відповідає за всю обробку імпульсів, а саме розпізнавання з цифрового сигналу імпульсів, їх підрахунок за допомогою лічильника, знаходження періоду часу між певною кількістю імпульсів, і врешті-решт обрахування самої миттєвої дози. Так ось, Audio Processing Controller, передає Tick Handler оцифрований сигнал, а саме за допомогою функції Recognizer, яка відповідає за це, і за розпізнавання імпульсів в цьому сигналі. Як тільки імпульс був розпізнаний, викликається метод Counter, який відповідає за підрахунок кількості імпульсів.

Як тільки цей метод викликається, він додає до лічильника один імпульс, і далі викликає функцію CountHandler, задача якої складається в перевірці чи вистачає нам імпульсів для обрахування миттєвої дози, якщо так, то вона бере дані про теперішній час, і знайшовши різницю між часом початку виміру, і теперішнім, визначає період часу, за який пройшла певна кількість імпульсів, яку ми задали в конфігураційному файлі. Далі, викликається функція CalculateDose, якій передаються як параметри кількість імпульсів, так і період часу за який вони пройшли. Ця функція відповідає за математичний обрахунок миттєвої дози. Після чого дані відправляються в FilterModel - це модель фільтра з плаваючим середнім, тобто він бере середнє значення з декількох останніх вимірів, а коли приходить новий вимір, найстаріший що є в фільтрі замінюється новим, і від цих значень береться середнє. А це середнє значення відправляється в VisualController - це контролер, який відповідає за дані які виводяться в графічну частину ПЗ для користувача.

Опис коду програми

Для початку розглянемо файлову структуру програми

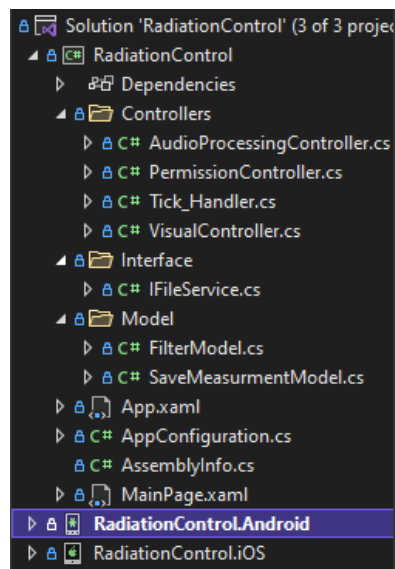


Рисунок 3.4.2 Файлова структура програми

На рисунку можемо побачити файлову структуру, на початку йде рішення, яке містить в собі три проєкти: RadiationControl, RadiationControl.Android, RadiationControl.IOS (рисунок 3.4.2). Перший проєкт це ядро нашої програми в якому реалізована основна логіка роботи, в других двох проєктах реалізовується специфічні речі для якоїсь певної платформи. Таким чином додатки що для Android, що для IOS, будуть мати однакову логіку, і код доведеться писати лише один раз, і не дублювати весь код двічі, що сильно спрощує розробку. В даному ПЗ нічого особливого для якоїсь з платформ реалізовано не було, тому все написане містить в проєкті RadioationControl.

В цьому проєкті ми можемо побачити три папки Controllers, Interface, Model.

В папці Controllers містяться основні контролери(керуючі файли) програми. AudioProcessingController – це контролер що відповідає за прийняття й опрацювання сигналу з дозиметру.

```

namespace RadiationControl.Controllers
{
    2 references
    internal static class AudioProcessingController
    {
        private static short[] audioBuffer = new short[AudioRecord.GetMinBufferSize(AppConfiguration.Freq, AppConfiguration.Channel, AppConfiguration.Encoding)];
        private static AudioRecord audioRecord = new AudioRecord(AudioSource.Mic, AppConfiguration.Freq, AppConfiguration.Channel, AppConfiguration.Encoding, audioBuffer.Length);
        private static Thread th;

        1 reference
        public static void StartMeasurment() {
            th = new Thread(Measurment);
            th.Start();
        }

        1 reference
        private static void Measurment()
        {
            Tick_Handler handler = new Tick_Handler();
            audioRecord.StartRecording();
            while (true)
            {
                audioRecord.Read(audioBuffer, 0, audioBuffer.Length);
                handler.Recognizer(audioBuffer, handler.Counter);
            }
        }

        1 reference
        public static void StopMeasurment()
        {
            audioRecord.Stop();
            th.Abort();
        }
    }
}

```

Рисунок 3.4.3 Код AudioProcessingController

Як ми бачимо контролер має декілька статичних параметрів, перший з яких audioBuffer, це масив числових значень куди буде відбуватись запис сигналу, розмір масиву визначається за допомогою функції AudioRecord.GetMinBufferSize, якому передаються значення частоти дискретизації, каналу, та кодування сигналу, на базі цих значень масив повертає мінімальний розмір масиву, для якісного і безперебійного зчитування сигналу (рисунок 3.4.3). Другий параметр audioRecord, саме за допомогою нього буде відбуватись зчитування сигналу. І третій параметр th, це потік, так як наш застосунок буде працювати в багатопотоковому режимі, це потрібно щоб головний потік працював стабільно, так як він відповідає за вивід на екран інформації, і обробкою натискань на кнопки. Саме в додатковому потоці буде проходити вимірювання. Якщо б вони проходили в основному потоці, то після запуску вимірювань в синхронному порядку почалось би виконуватись безкінечна функція, що заблокувало б всю програму, і вона б працювала в безкінечному циклі. Всі три параметри класу є приватними, тобто доступ до них можна отримати тільки з об'єктів цього класу. Також контролері присутні три функції. Перша StartMeasurment, яка починає вимірювання а саме створює новий потік, який буде виконувати функцію Measurment, і запускає його. Друга функція Measurment в цій функції відповідно і йде процес вимірювання, а саме ми створюємо новий об'єкт класу TickHandler, який відповідає за знаходження і обробку імпульсів. Запускаємо зчитування сигналу, і потім в безкінечному циклі зчитує значення в буфер, і передає цей

буфер в розпізнавач імпульсів, за допомогою `handler.Recognizer(audioBuffer, handler.Counter)`, другим параметром передається функція, яка буде викликатись при розпізнанні імпульсу. Третя функція `StopMeasurment` зупиняє процес вимірювання, а саме спочатку зупиняє зчитування сигналу, а потім потік в якому відбувалось вимірювання.

Другий контролер `PermissionController` – він відповідає за перевірку доступів, а також запитів по доступ до якогось функціонала телефону. В даному випадку він потрібен для перевірки й отримання доступу до мікрофона.

```
namespace RadiationControl.Controllers
{
    2 references
    internal static class PermissionController
    {
        2 references
        public static async Task<bool> CheckAndRequestPermission<T>()
            where T: Permissions.BasePermission, new()
        {
            var micPermission = await Permissions.CheckStatusAsync<T>();
            if (micPermission != PermissionStatus.Granted)
            {
                micPermission = await Permissions.RequestAsync<T>();
                return micPermission == PermissionStatus.Granted;
            }
            else
            {
                return true;
            }
        }
    }
}
```

Рисунок 3.4.4 Код контролера `PermissionController`

Цей контролер складається з однієї функції `CheckAndRequestPermission<T>()`, як ми бачимо в виклику присутній Generic `<T>`, саме за допомогою нього в функцію передається який саме доступ нам потрібно перевірити чи програма має, а якщо немає та зробити запит на отримання, функція після відпрацювання повертає булеве значення, `true` якщо доступ був, або його надали або `false` якщо доступу не надали (рисунок 3.4.4). Наступний контролер це `TickHandler` так званий оброблювач імпульсів він вміє розпізнавати, підраховувати імпульси, а також вираховувати з цього миттєву дозу.

```

namespace RadiationControl.Controllers
{
    [Serializable]
    internal class Tick_Handler
    {
        [Serializable]
        public Tick_Handler()
        {
            TickCounter = 0;
            TickContinuous = false;
            StartMeasurementTime = DateTime.Now.Ticks;
            StartPeriodMeasurementTime = DateTime.Now.Ticks;
        }

        private FilterModel<double> filter = new FilterModel<double>(AppConfiguration.Filter_Order);
        [Serializable]
        private int _tickCounter { get; set; }
        [Serializable]
        private int TickCounter
        {
            get { return _tickCounter; }
            set
            {
                _tickCounter = value;
                VisualController.UIController.TickCounter = value.ToString();
            }
        }

        public delegate void On_Tick();
        [Serializable]
        private bool TickContinuous { get; set; }
        [Serializable]
        private long StartMeasurementTime { get; }
        [Serializable]
        private long StartPeriodMeasurementTime { get; set; }

        [Serializable]
        public void Recognizer(short[] buffer, On_Tick operation)
        {
            for (int i = 0; i < buffer.Length; i++)
            {
                if (Math.Abs(buffer[i]) > AppConfiguration.Min_Amplitude)
                {
                    if (!TickContinuous)
                    {
                        TickContinuous = true;
                        operation();
                    }
                }
                else
                {
                    if (TickContinuous) { TickContinuous = false; }
                }
            }
        }

        [Serializable]
        public void Counter()
        {
            TickCounter++;
            this.CountHandler();
        }

        [Serializable]
        private void CountHandler()
        {
            if (TickCounter % AppConfiguration.Tick_Counts_For_Measur == 0)
            {
                long periodInTick = DateTime.Now.Ticks - StartPeriodMeasurementTime;
                StartPeriodMeasurementTime = DateTime.Now.Ticks;
                Dose_Formula(AppConfiguration.Tick_Counts_For_Measur, periodInTick);
            }
        }

        [Serializable]
        private void Dose_Formula(int tickCount, long timeInTick)
        {
            double period = (double)timeInTick / 10000000;
            double frequency = tickCount / period;
            double dose = 0.808889569 * Math.Pow(frequency, 3) - 0.868384 * Math.Pow(frequency, 2) + 41.81 * frequency;
            filter.Add(dose);
            double AverageDose = filter.GetValue();
            VisualController.UIController.Frequency = frequency.ToString();
            VisualController.UIController.Dose = AverageDose.ToString("F2");
        }
    }
}

```

Рисунок 3.4.5 Код контролеру Tick Handler

В даному контролері є конструктор класу, який присвоює деякий властивостям даного класу значення, далі йдуть оголошення самих властивостей контролеру, перше властивість це екземпляр класу фільтра плаваючого середнього, усереднення отриманих значень миттєвої дози. Наступний параметр `_tickCounter` це приватний параметр для зберігання кількості імпульсів. Далі йде параметр `TickCounter` це оболочка до попереднього параметру, для специфічної

реалізації присвоєння значення в цю властивість, а саме як ми бачимо в функції `set` цієї властивості прописано збереження властивості, і одночасно з цим замінює нове значення в контролері який відображає дані в UI частині, таким чином ми отримуємо що при оновлені даної властивості вона відразу буде оновлена в графічній частині користувача. Наступний параметр `TickContinious` це так званий прапорець, який вказує на те, чи йде зараз імпульс чи ні, при ініціалізації об'єкту задається `false`. Далі `StartMeasurmentTime` і `StartPeriodMeasurmentTime` перший фіксує час початку вимірювання, а другий час початку періоду вимірювання, тобто друга властивість змінюється кожного разу як ми спіймали необхідну кількість імпульсів для вимірювання, і фіксується заново, з самого початку значення даним властивостям задаються в конструкторі. Далі йдуть функції даного контролеру. Перша функція `Recognizer`, першим параметром є буфер в якому знаходиться сигнал, другим параметром приймає `callback` функцію. Процес розпізнавання відбувається наступним чином, ми перебираємо всі значення з буферу, тобто оцифровані значення сигналу, якщо сигнал перевищив поріг шумів, який ми задали як параметр в конфігураційному файлі, то ми ідентифікуємо це як імпульс, і викликаємо `callback` функцію, одночасно з цим міняємо значення властивості `TickContinious` на `true`, що буде означати що імпульс ще проходить, це потрібно для того, щоб якщо імпульс триває не один вимір сигналу, а наприклад десять, то ми ідентифікували його як один імпульс. Як тільки сигнал опуститься знов на рівень шумів властивість `TickContinious` зміниться на `false`, і функція буде очікувати на наступний імпульс. Друга функція `Counter` вона відповідає за підрахунок імпульсів, тобто коли її викликають, вона збільшує лічильник імпульсів на один. І далі викликає наступну функцію `CounterHandler`, це оброблювач кількості імпульсів, його задача полягає перевірити, чи вистачає нам кількості імпульсів для виділення періоду вимірювання, він це робить за наступним алгоритмом, перевіряє чи при діленні на задану на в конфігурації кількість тіків на вимір є остача, якщо остачі немає, означає що пройшло потрібна кількість вимірів, і можна виділити період часу, взявши даний час, і `StartPeriodMeasurmentTime`, і знайшовши з них різницю це і буде період часу, за

який відбулось та кількість імпульсів, що ми задали в конфігурації. Далі тривалість періоду і кількість імпульсів передаються в функцію CalculateDose, задача якої обрахувати по заданій формулі миттєву дозу, додати її в фільтр з плаваючим середнім, взяти значення середнього з фільтру, і передати його на UI частину застосунку.

Наступний контролер, це VisualController, його суть в зберіганні, передаванні і оновленні даних в UI частині застосунку, він є зручним так як за допомогою нього ми можемо змінювати значення в графічній частині, яку бачить користувач з любого місця застосунку, це так званий посередник.

```
namespace RadiationControl.Controllers
{
    [Serializable]
    public class VisualController : INotifyPropertyChanged
    {
        public static VisualController UiController = new VisualController();

        #region Frequency Label
        [Serializable]
        private string _frequency { get; set; }

        [Serializable]
        public string Frequency
        {
            get { return _frequency; }
            set
            {
                _frequency = value;
                OnPropertyChanged("Frequency");
            }
        }
        #endregion

        #region Tick Counter Label
        [Serializable]
        private string _tickCounter { get; set; }

        [Serializable]
        public string TickCounter
        {
            get { return _tickCounter; }
            set
            {
                _tickCounter = value;
                OnPropertyChanged("TickCounter");
            }
        }
        #endregion

        #region Dose Label
        [Serializable]
        private string _dose { get; set; }

        [Serializable]
        public string Dose
        {
            get { return _dose; }
            set
            {
                _dose = value;
                OnPropertyChanged("Dose");
            }
        }
        #endregion

        #region Property Change Configuration
        public event PropertyChangedEventHandler PropertyChanged;

        [Serializable]
        protected virtual void OnPropertyChanged(string propertyName)
        {
            if (PropertyChanged == null)
                return;

            PropertyChanged(this, new PropertyChangedEventArgs(propertyName));
        }
        #endregion
    }
}
```

Рисунок 3.4.6. Код контролеру VisualController

Працює він наступним чином, дані які потрібно синхронізувати між частиною яка обраховує все, і графічною частиною для кожного з них створюється приватна властивість, і додаткова публічна обгортка, яка створює специфічний зберігач для цих значень, тобто при зберіганні значення, значення зберігається в відповідний приватний параметр, і викликається створена в цьому ж контролері функція `OnPropertyChanged`, і йому передається назва цього параметра, всередині ця функція викликає `event PropertyChanged`, який змінює дані в графічній частині, таким чином при будь-якій зміні даних в цьому контролері, вони миттєво зміняться в графічній частині.

Далі перейдемо до інтерфейсів, тут є один інтерфейс `IFileService`, який прописує для цього інтерфейсу обов'язкову функцію `CreateFile`, в подальшому це використовується для зберігання файлів з вимірами в `external storage`, на телефоні користувача.

```
1 reference
public interface IFileService
{
    1 reference
    void CreateFile(string textMessage);
}
```

Рисунок 3.4.7. Код інтерфейсу `IFileService`

Далі розглянемо створенні моделі. Перша модель `FilterModel`. Задача даної моделі в реалізації фільтру з плаваючим середнім. З функцією додавання нового значення, і отримання на даний момент середнього значення з тих що містяться в фільтрі. Далі наведений код програми:

```

namespace RadiationControl.Model
{
    3 references
    internal class FilterModel<T>
    {
        1 reference
        public FilterModel(int order)
        {
            Filter_Order = order;
        }
        3 references
        private int Filter_Order { get; set; }
        private int filter_iterator = 0;

        private List<T> filter_values = new List<T>();

        1 reference
        public void Add(T value)
        {
            if (filter_values.Count <= Filter_Order)
            {
                filter_values.Add(value);
            }
            else
            {
                filter_values[filter_iterator] = value;
            }
            ItteratorUp();
        }

        1 reference
        public double GetValue()
        {
            List<double> values = filter_values.Select(x => double.Parse(x.ToString())).ToList();
            return values.Average();
        }

        1 reference
        private void ItteratorUp()
        {
            if(filter_iterator < Filter_Order - 1)
            {
                filter_iterator++;
            }
            else
            {
                filter_iterator = 0;
            }
        }
    }
}

```

Рисунок 3.4.8. Код моделі FilterModel

Друга модель SaveMeasurmentModel, її задача збирати всі дані про сигнал, а потім збереження їх в файл, відповідно до цього в модель було створено дві функції, додання значень, і перетворення всіх даних в тип даних string, тобто в текстовий формат, де кожне нове значення, починається з нового рядку.

```

namespace RadiationControl.Model
{
    0 references
    internal class SaveMeasurmentModel
    {
        private List<int> measurments = new List<int>();

        0 references
        public void AddMeasurment(int[] addData)
        {
            measurments.AddRange(addData);
        }

        0 references
        public string ToString(string delimiter)
        {
            string readyToSave = String.Join(delimiter, measurments);
            return readyToSave;
        }
    }
}

```

Рисунок 3.4.9. Код моделі SaveMeasurmentModel

Далі є ще декілька важливих файлів, які зберігаються не в папках, а в корні програми, перший важливий це AppConfiguraton, в ньому зберігається конфігурація програми, а саме основні елементи які можуть впливати, на точність та швидкодію, такі як наприклад, порядок фільтра, кількість імпульсів для одного виміру, мінімальна амплітуда, перевищуючи яку, ми розуміємо що відбувається імпульс, частота дискретизації, канал для зчитування сигналу і кодування

```

namespace RadiationControl
{
    10 references
    internal static class AppConfiguraton
    {
        public const int Freq = 44100;

        public const int Tik_Counts_For_Measur = 5;

        public const int Filter_Order = 5;

        public const int Min_Amplitude = 2000;

        public const ChannelIn Channel = ChannelIn.Default;

        public const Android.Media.Encoding Encoding = Android.Media.Encoding.Pcm16bit;
    }
}

```

Рисунок 3.4.10. Код AppConfiguraton

Також є файл в якому описується графічна частина, зазвичай для кожної сторінки створюється окремий файл, але так як поки що в нас одна сторінка, то то

в нас є один файл xaml, з описом графічних компонентів, і файл C# до нього, який виступає обробником, всіх натискань на даній сторінці.

```
1 <?xml version="1.0" encoding="utf-8" ?>
2 <ContentPage xmlns="http://schemas.microsoft.com/winfx/2004/Forms"
3             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
4             xmlns:local="clr-namespace:RadiationControl"
5             x:Class="RadiationControl.MainPage">
6
7     <StackLayout>
8         <Frame BackgroundColor="#2196F3" Padding="20" CornerRadius="20" HeightRequest="560" HasShadow="True" Margin="10,10,10,20">
9             <StackLayout x:Name="MeasurPanel" IsVisible="false">
10
11                 <Label Text="{Binding Dose, Source={x:Static local:Controllers.VisualController.UIController}, StringFormat='{0} мР/ч'}" HorizontalTextAlignment="Center" VerticalTextAlignment="Center" TextColor="White" HeightRequest="40" FontSize="40"/>
12                 <Label Text="{Binding TickCounter, Source={x:Static local:Controllers.VisualController.UIController}, StringFormat='{0}'}" HorizontalTextAlignment="Start" VerticalTextAlignment="End" TextColor="White"/>
13                 <Label Text="{Binding Frequency, Source={x:Static local:Controllers.VisualController.UIController}, StringFormat='{0}'}" HorizontalTextAlignment="Start" TextColor="White" VerticalTextAlignment="End"/>
14             </StackLayout>
15             <StackLayout x:Name="ErrorPanel" IsVisible="false">
16                 <Label Text="Вимірювання не вимірює, не надає доступу" HorizontalTextAlignment="Center" VerticalTextAlignment="Center" TextColor="White" HeightRequest="40" FontSize="40"/>
17             </StackLayout>
18         </Frame>
19         <Button x:Name="b" Text="Почати вимірювання" Clicked="button_Start_Clicked" CornerRadius="30" Margin="10,0,10,0"/>
20     </StackLayout>
21 </ContentPage>
```

Рисунок 3.4.11. Код графічної частини сторінки

```
namespace RadiationControl
{
    5 references
    public partial class MainPage : ContentPage
    {
        1 reference
        public MainPage()
        {
            InitializeComponent();
        }

        2 references
        private async void button_Start_Clicked(object sender, EventArgs e)
        {
            if(await PermissionController.CheckAndRequestPermission<Permissions.Microphone>() == true)
            {
                AudioProcessingController.StartMeasurment();
                b.Text = "Зупинити вимірювання";
                b.Clicked -= button_Start_Clicked;
                b.Clicked += button_Stop_Clicked;
                ErrorPanel.IsVisible = false;
                MeasurPanel.IsVisible = true;
            }
            else
            {
                MeasurPanel.IsVisible = false;
                ErrorPanel.IsVisible = true;
            }
        }

        3 references
        private void button_Stop_Clicked(object sender, EventArgs e)
        {
            AudioProcessingController.StopMeasurment();
            b.Text = "Почати вимірювання";
            b.Clicked += button_Start_Clicked;
            b.Clicked -= button_Stop_Clicked;
        }
    }
}
```

Рисунок 3.4.12. Код обробника графічної сторінки

Також ще є автоматично створений хатарін файл App.xaml і App.xaml.cs, перший файл нам не сильно цікавий, а в другий були внесені зміни, а саме додана перевірка на доступ до мікрофону мобільного пристрою

```

namespace RadiationControl
{
    6 references
    public partial class App : Application
    {
        2 references
        public App()
        {
            InitializeComponent();
            MainPage = new MainPage();
        }

        0 references
        protected override void OnStart()
        {
            Controllers.PermissionController.CheckAndRequestPermission<Permissions.Microphone>();
            base.OnStart();
        }

        0 references
        protected override void OnSleep()
        {
        }

        0 references
        protected override void OnResume()
        {
        }
    }
}

```

Рисунок 3.4.13. Код App.xaml.cs

3.5 Подальший розвиток проєкту

В майбутньому, до проєкту планується створити API-сервер, на який буде приходити інформація про іонізуюче випромінювання зі всіх пристроїв. Таким чином, це допоможе зі збором статистики щодо іонізуючого випромінювання зі всього світу. Тобто, уявімо, деякі користувачі придбали собі такі дозиметри, для аматорських цілей, але статистика з кожного такого пристрою відправляється на сервер, і таким чином це перетворюється в глобальну статистику.

Для зберігання та візуалізації даних по кожному пристрою найкраще підходить зв'язка InfluxDB, і Grafana.

InfluxDB – це нереляційна база даних часових рядів, з великою швидкістю читання й запису.

БД часових рядів, виходячи з назви, є системами баз даних, спеціально призначеними для обробки інформації, пов'язаної з часом.

В основному, всі мають справу з реляційними базами даних (MySQL або SQL Server). Всі вони ґрунтуються на тому, що у вас є таблиці. Ці таблиці містять стовпці та рядки, кожна з яких визначає запис у вашій таблиці. Часто, ці таблиці спеціально призначені для певної мети. Одна може бути призначена для

зберігання користувачів, інша – для фотографій чи відео. Такі системи ефективні, масштабовані та використовуються безліччю гігантських компаній з мільйонами запитів на своїх серверах.

Бази даних часових рядів працюють інакше. Дані зберігаються в «колекціях», але ці колекції мають спільний знаменник: вони об'єднані з часом. Це означає, що для кожної точки, яку ви можете зберегти, у вас є пов'язана з нею тимчасова мітка.

Виникає питання: чи не можна використовувати реляційну базу даних і просто включити до неї стовпець з іменем «час»? Наприклад, Oracle має тип даних **TIMESTAMP**, який можна було б використовувати для цієї мети. Звичайно, таке можливо, але це неефективно. До того ж InfluxDB дуже зручно інтегрувати з Grafana.

Grafana – це платформа з відкритим вихідним кодом для візуалізації, моніторингу та аналізу даних. Grafana дозволяє користувачам створювати дашборди з панелями, кожна з яких відображає певні показники протягом певного періоду часу. Кожен дашборд універсальний, тому його можна налаштувати для конкретного проекту або з урахуванням потреб розробки та/або бізнесу (рисунки 3.5).

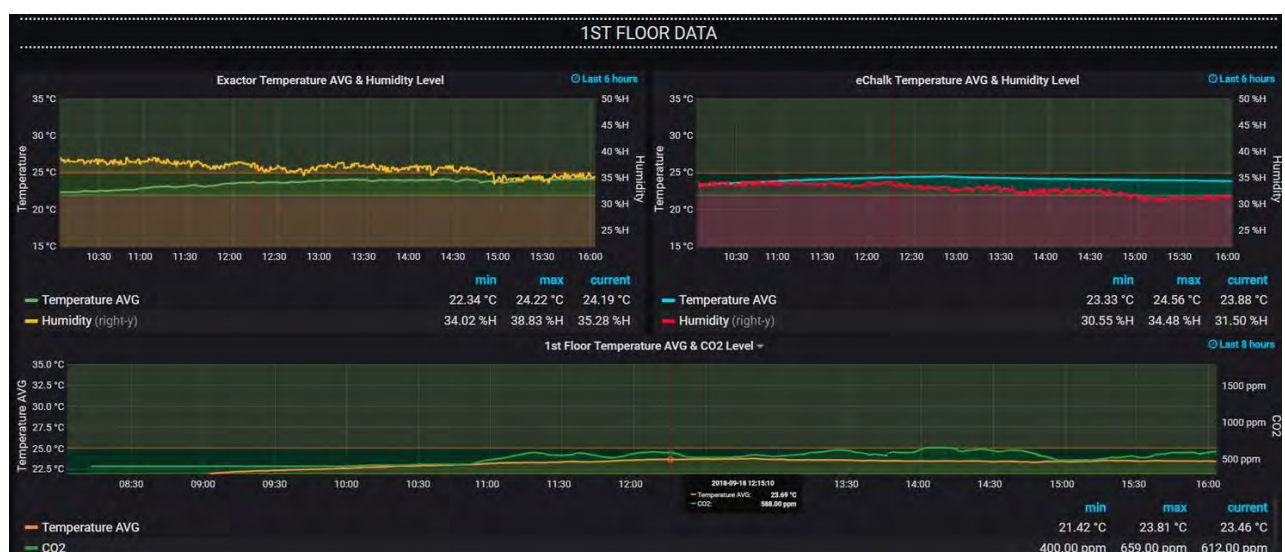


Рисунок 3.5. Приклад дашборду

На рис. Зображено приклад дашборду grafana який демонструє зміну температур в кімнатах.

Також за допомогою InfluxDB і Grafana можна налаштувати так звані алерти, а простіше кажучи попередження, тобто коли система побачить, що щось не так(параметри задаємо ми) вона сповістить нас про це через налаштований нами спосіб(пошта, телеграм, виклик на мобільний тощо)

На базі даної зв'язки технологій і нашого дозиметру з ПЗ можна реалізувати декілька систем:

- Система відслідковування зміни іонізуючого випромінювання в ризикових місцях(АЕС, кладовище радіаційних залишків тощо);
- Система моніторингу іонізуючого випромінювання по всьому світу, збираючи дані у звичайних користувачів, що купили наш прилад;
- Система моніторингу персоналу на АЕС. Наприклад, на початку зміни, персоналу видали такі дозиметри, а оператор може відслідковувати за тим який зараз фон там, де вони знаходяться, та накопичена кількість іонізуючого випромінювання, в разі перевищення норм оператору прийде сповіщення, та він зможе побачити, яке перевищення і в кого відбулось, та прийняти певні заходи.

3.6 Приклад роботи програми

Приклад відображення інформації на головній сторінці додатку:



Рисунок 3.6 Головне вікно програми

На даному зображенні приведено знімок екрану смартфона в процесі вимірювання. На інтерфейсі присутня кнопка початку та зупинки вимірювання. А також полі з підрахунком кількості імпульсів та їх частотою. І головне поле – це відображення значення еквівалентної дози в даний момент часу.

Висновки до розділу

В цьому розділі описана розробка ПЗ для широкого аналізу сигналу. В якості середовища обробки було обрано Google Colaboratory та мову програмування Python, оскільки дані інструменти дуже доцільно підходять для аналізу даних. В якості даного аналізу, було візуалізовано необроблений сигнал отриманий з пристрою. Проведено дослідження залежності рівня амплітуди сигналу до його тривалості. Розглянуто значення сигналів при звичайному фоні зареєстрованих в побуті та в безпосередній близькості до джерела випромінення енергоактивних частинок. Порівнявши значення імпульсів сигналів, було виокремлено складову шуму для подальшого врахування при автоматизованій реєстрації значень.

Також, в рамках цього аналізу було протестовано та підібрано алгоритми обробки та усереднення значень, які при їх використанні забезпечують точність та швидкодію реакції.

В рамках даного розділу описано процес створення ПЗ для смартфона, яке безпосередньо реєструє та обробляє імпульси отримані з пристрою. Для написання програми було обрана технологія створення мобільних додатків Xamarin та мова програмування C#. Було проведено аналіз бібліотек та функціональностей біля безпосередньої реалізації алгоритму обробки даних. Алгоритм обробки сигналу базується на вже попередньо досліджених підходах при первинному аналізу даних.

Проведено та описано приклад подальшого удосконалення програмного забезпечення та розширення за для більш масштабного аналізу навколишнього середовища. Приведений приклад візуалізації та відображення даних в додатку.

РОЗДІЛ 4. РОЗРОБКА СТАРТАП-ПРОЄКТУ

«АВТОМАТИЗОВАНА СИСТЕМА ДОЗИМЕТРИЧНОГО КОНТРОЛЮ ІОНІЗУЮЧОГО ВИПРОМІНЮВАННЯ»

4.1 Опис технологічний аудит ідеї проєкту

Стартап базується на ідеї портативного приладу малих габаритів для вимірювання дози іонізуючого випромінювання. Даний прилад може бути використаний на виробництві із ймовірними забрудненнями, для вимірювання накопиченої дози у працівника, а також в туристичних компаніях до зон відчуження. Також може знайти застосування для наукових цілей, оскільки буде формуватися база замірів у різних точках світу. Головною особливістю та відмінністю являється використання модулю GPS, та прив'язку до мобільного телефону, що дає змогу робити вимірювання в автоматичному режимі, Також особливістю даної системи є можливість використання різних типів приймачів іонізуючого випромінювання, що в свою чергу дає змогу застосовувати один прилад для різних завдань. Ще однією метою є створення власного ПЗ для простого та швидкого відображення даних вимірювання, та допомагає зробити продукт більш незалежним, що уточнено наведено в Таблиця 4.1.

В даному розділі буде проведено аналіз стартап-проєкту головною метою якого є визначення чи зможе продукт вийти на ринок та чи зможе він скласти конкуренцію, які вже є в цій ніші та займають пріоритетні місця в ній.

Таблиця 4.1. Опис ідеї стартап-проєкту

<i>Зміст ідеї</i>	<i>Напрямки застосування</i>	<i>Вигода для користувача</i>
Основна ідея, це розробка автоматизованого приладу для вимірювання іонізуючого випромінювання	Використання на підприємствах	- Мала вартість - Зручний формат відображення результатів - Простий в використанні
	Для наукових досліджень	

Загалом, в стартап проєкті пропонується пристрій, який являє собою

простий прилад, який має змогу змінювати датчики реєстрації випромінювання та зручно підключається до мобільного телефону, що дає змогу легко і зручно візуалізувати, обробляти та передавати інформацію на сервер.

Далі проведемо аналіз потенційних техніко-економічних переваг порівняно з пропозиціями найближчих конкурентів:

- визначимо перелік техніко-економічних властивостей та характеристик ідеї;
- визначаємо коло найближчих конкурентів (проектів-конкурентів) або товарів-аналогів, або товарів-замінників, що вже існують на ринку, та проводимо збір інформації щодо значень техніко-економічних показників для ідеї власного проекту та проектів-конкурентів відповідно до визначеного вище переліку
- проводимо порівняльний аналіз показників: для власної ідеї визначено показники, що мають а) гірші значення (W, слабкі); б) аналогічні (N, нейтральні) значення; в) кращі значення (S, сильні) (Таблиця 4.2).

Конкуренти:

Конкурент1 - Radiation Monitor Zive

Конкурент2 - Atom Tag

Таблиця 4.2. Визначення сильних, слабких та нейтральних характеристик ідеї проекту

№	Техніко-економічні Характеристики ідеї	Товари/концепції конкурентів			W (слабка сторона)	N (нейтральна сторона)	S (сильна сторона)
		Мій проект	Конкурент1 Radiation Monitor Zive	Конкурент2 Atom Tag			
1	Вартість, грн	7000	15300	11360			+
2	Вимірювання малих значень	Має швидку реакцію на зміну іонізуюч	Час реакції статичний	Час реакції статичний			+

		ого фону					
3	Можливість передачі даних на сервер	Так, за допомо гою смартф ону	Так, за допомог ою смартфо ну	Ні		+	
4	Автоматизов ане вимірюванн я	Є	Немає	Немає			+

З Таблиця 4.2 можна зробити висновки, що дана система не тільки має переваги в базових можливостях приладів конкурентів, а й велику кількість переваг виражених в можливостях збереження даних вимірювання, й можливість автоматизованого вимірювання.

В межах даного підрозділу проводимо аудит технології, за допомогою якої можна реалізувати ідею створення проекту.

Визначення технологічної здійсненості ідеї проекту передбачає аналіз складових які вказані в Таблиця 4.3.

Таблиця 4.3. Технологічна здійсненість ідеї проекту

№	Ідея проекту	Технології реалізації	Наявність технологій	Доступність технологій
1	Система автоматизованого вимірювання іонізуючого випромінювання	Вибір датчика під конкретні потреби	Наявна	Недоступна
		Самостійне виготовлення схеми датчику, та наявні електричні компоненти	Наявна	Доступна
2	Створення програмного забезпечення для обробки результатів вимірювання	Мова програмування C#	Наявна	Доступна

Обрана технологія реалізації ідеї проекту: було обрано самостійне виготовлення пристрою, та наявні електричні компоненти та мова програмування C# для створення ПЗ

Результати даного аналізу проекту показують, що за наявних технологій ми маємо можливість самостійно виготовити вимірювач, а також самостійно створити ПЗ, що зробить ціну системи значно меншою.

4.2 Аналіз ринкових можливостей запуску стартап-проекту

В даному підрозділі визначаємо ринкові можливості, які можна використати під час ринкового впровадження проекту, та ринкові загрози, які можуть перешкодити його реалізації.

Визначення ринкових можливостей дозволяє спланувати напрям розвитку, потреб потенційних клієнтів та пропозицій проектів-конкурентів.

Спочатку проведемо аналіз попиту: наявність попиту, обсяг, динаміка розвитку ринку (таблиця 4.4). У світі існує досить багато дрібних компаній, що виробляють портативні дозиметри, але значний вплив на ринок мають лише 4 із них: Radiation Monitor Zive, Pokega, Pocket PM, Atom Tag.

Таблиця 4.4. Попередня характеристика потенційного ринку стартап-проекту

№	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од	4
2	Загальний обсяг продаж, грн	Невідомий
3	Динаміка ринку	Стабільна
4	Наявність обмежень для входу	Немає
5	Специфічні вимоги до стандартизації та сертифікації	Немає
6	Середня норма рентабельності в галузі (або по ринку), %	100

Аналіз Таблиця 4.4 показує, що ринок має стабільну динаміку і хороший попит на запропонований нами продукт, тому є доцільним вихід на ринок. Визначимо потенційних клієнтів, котрим будемо пропонувати нашу систему.

Таблиця 4.5. Характеристика потенційних клієнтів стартап-проекту

<i>№</i>	<i>Потреба, що формує ринок</i>	<i>Цільова аудиторія (цільові сегменти ринку)</i>	<i>Відмінності у поведінці різних потенційних цільових груп клієнтів</i>	<i>Вимоги споживачів до товару</i>
1	Потреба в недорогому мобільному дозиметрі	Туристичні агенти, які відправляють людей до небезпечних зон; лабораторії неруйнівного контролю.	Основні критерії, що впливають на поведінку потенційних клієнтів: відповідність допускам, мобільність та автоматизованість, доступність, наявний функціонал для приваблення людей.	Надійність, малі габарити, тривалий час роботи від акумулятору, гарантійні зобов'язання.

В даній таблиці ми описали фактори, котрі формують поведінку користувачів відносно нашої системи та їх основні вимоги до продукції.

При застосуванні даної технології існують певні загрози, тож буде доцільно провести аналіз факторів ризику та можливостей при виведенні товару на ринок.

Таблиця 4.6. Фактори загроз

<i>№</i>	<i>Фактор</i>	<i>Зміст загрози</i>	<i>Можлива реакція компанії</i>
1	Введення на ринок більш інноваційних пристроїв	Поява продукції на ринку із суттєвішими перевагами	Постійна робота над вдосконаленням товару та періодичне оновлення програмного забезпечення; маркетингова діяльність.

2	Зниження попиту на продукцію	Неактуальність товару	Маркетингова діяльність, зниження ціни товару.
3	Клієнтська база	Оскільки пристрій специфічний, набрати клієнтську базу	Участь у виставках, надання тестових пристроїв для можливості проявлення переваг клієнтам над конкурентами
4	Постачання	Невчасне постачання компонентів	Пошук нових постачальників, формування резерву компонентів на складах
5	Відсутність початкового попиту	Не бажання переходити на нові технології	Приведення аргументованих доказів, що зміна технології приведе до збільшення попиту на продукт.

В Таблиця 6. ми визначили можливі фактори загроз, такі як: клієнтська база, відсутність навичок використання інноваційних систем даного типу, термін роботи, відсутність початкового попиту, конкуренція. Вони в майбутньому можуть перешкоджати ринковому впровадженню нашого проекту, а також можливу реакцію компанії на загрози та шляхи їх усунення.

Але на одному рівні з загрозами є також і можливості.

Таблиця 4.7. Фактори можливостей

<i>№</i>	<i>Фактор</i>	<i>Зміст можливостей</i>	<i>Можлива реакція компанії</i>
1	База даних отримана шляхом експлуатації пристроїв	Завдяки масовому використанню пристроїв будується велика база значень, які відображуються на карті, що дуже цікаво для клієнтів	Збільшення оборотності товару
2	Технічні конференції, виставки, семінари	Можливість представляти систему контролю шляхом написання оглядових	Написання оглядових статей в журнали, тематика яких зав'язана на неруйнівному контролі
3	Ринок збуту	Легкий шлях виходу на міжнародний ринок	Розробка та виготовлення системи у відповідності до закордонних стандартів якості
4	Наявність власного ПЗ	В процесі розробки стартапу була розроблена власна програма для обробки результатів вимірювання	Акцентування уваги на самостійності стартапу, та не потрібності покупки додаткового ПО
5	Участь в виставках	Так як продукт новий та інноваційний маємо можливість участі в виставках з найсучаснішим обладнанням	Розповсюдження рекламної продукції, даної системи

В Таблиця 4.7. були визначені фактори можливостей такі як: відсутність даного типу систем, наявність власного ПЗ, подача результатів вимірювання в форматі легкому для подальшого аналізу, участь в виставках, створення нової ніші. Вони посприяють ринковому впровадженню даного проекту.

Також доцільно буде провести аналіз ринку для подальшого формування стратегії.

Таблиця 4.8. Ступеневий аналіз конкуренції на ринку

<i>Особливості конкурентного середовища</i>	<i>В чому проявляється дана характеристика</i>	<i>Вплив на діяльність підприємства(можливі дії компанії, щоб бути конкурентноспроможною)</i>
1. Тип конкуренції: олігополія	Існує 4 прямих конкуренти, інші мають схожий функціонал, але не систему	Забезпечити користувачів
2. За рівнем конкурентної боротьби: глобальний	Конкуренти – загалом іноземні компанії, які реалізують функціонал під своїх користувачів	Забезпечити доступність в усіх країнах, де це можливо. Додання локалізації в програму та підтримку іноземних користувачів
3. За галузевою ознакою: міжгалузева	Даний пристрій може використовуватися для розповсюдження між турагентами і застосування в неруйнівному контролі	Постійне вдосконалення продукту, що неприв'язане до галузі
4. Конкуренція за видами товарів: товарно-видова	Можлива конкуренція між постачальниками схожих за функціоналом продуктів	Наявний пристрій більш автоматизований ніж конкурентні та спроможний задавати тренд використання
5. За характером конкурентних переваг: цінова та нецінова	Цінова – зробити продукт доступнішим ніж в конкурентів, нецінова – полягає в високій якості наданих послуг	Цінова – зекономити за рахунок більш оптимізованих алгоритмів обробки. Нецінова – широкі можливості застосування у різних галузях
6. За інтенсивністю: марочна	Конкурентні компанії пропонують подібний продукт	Отримання держзамовлення подальшеразширення ринку

З наведеної вище таблиці можна судити, що ситуація на ринку

характеризується вільною конкуренцією в локальних масштабах, тому немає факторів, які однозначно обмежують входження на ринок.

Більш детальний аналіз умов конкуренції наведено в Таблиця 4.9.

Таблиця 4.9. Аналіз в конкуренції в галузі за М. Портером

	<i>Прямі конкуренти</i>	<i>Потенційні конкуренти</i>	<i>Постачальники</i>	<i>Клієнти</i>	<i>Товари-замінники</i>
<i>Складові аналізу</i>	Monitor Zive, Pokega, Pocket PM, Atom Tag	Молоді стартап команди, які вносять більш інноваційний підхід	Ними являються закордонні фірми, які пропонують найкращі умови	Фактори сили: - Змінні витрати - Високий рівень чутливості до зміни цін - Контроль якості	Пристрої із більш точним датчиком та додатковим функціоналом
<i>Висновки :</i>	Пристрої конкурентів не достанькою універсальні, а ПЗ не надто інноваційне	Є гарні шанси вийти на ринок попри те, що наявні конкуренти. Термін виходу: 6-9 місяців	Доставка компонентів навіть у великій кількості не дешево задоволення, тому необхідно визначити найкращу пропозицію	Для клієнтів найбільш важлива цінова доступність послуг, захист персональних даних, якість послуг	Замінники мають розширений функціонал, але занадто високу вартість, тому вони не орієнтуються на моїх клієнтів

Провівши аналіз таблиці можна зробити висновок, що з огляду на конкурентну ситуацію на ринку можливість роботи на ринку присутня. Також були визначені характеристики, які повинен мати проект і які дії повинна проводити компанія, щоб бути конкурентоспроможною на ринку.

Після аналізу визначається та обґрунтовується перелік факторів що впливають на конкурентоспроможність. Допоки проект не впроваджено в життя, цей аналіз важко провести точно, можна дати тільки попередню оцінку конкурентоспроможності.

Основауючись на аналізі конкуренції, а також враховуючи характеристики ідеї проекту, вимоги споживачів до товару, а також фактори маркетингового середовища, можна визначити перелік факторів конкурентоспроможності. Аналіз факторів приведений в наступній Таблиця 4.10.

Таблиця 4.10. Обґрунтування факторів конкурентоспроможності

<i>№ п/п</i>	<i>Фактор конкурентоспроможності</i>	<i>Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)</i>
1	Система та методи управління фірмою	Налагоджена система та методи управління є основою успішності фірми
2	Зручне та інформативне програмне забезпечення	Завдяки задумці реалізувати ідеї відображення, зберігання перегляду інформації та багато іншого, мій продукт матиме великі переваги в зручності використання.
3	Уніфікована електрична модель	Реалізувавши один прилад з можливістю заміни датчиків для покриття всього основного діапазону випромінення ми збільшуємо наявність потенційних клієнтів. Також це стало можливим при можливості уніфікувати програму під різні датчики
4	Наявність локалізації	Надає переваги для внутрішніх клієнтів та дозволяє вийти на іноземні ринки

В даній таблиці проведено на основі Таблиця 4.9 визначили та обґрунтували фактори конкурентоспроможності даного проекту.

За попередньо визначеними факторами конкурентоспроможності проведемо аналіз слабких та сильних сторін проекту.

Таблиця 4.11. Порівняльний аналіз сильних та слабких сторін

<i>№</i>	<i>Фактор конкурентоспроможності</i>	<i>Бали 1-20</i>	<i>Рейтинг у порівнянні з конкурентами</i>						
			-3	-2	-1	0	+1	+2	+3
1	Система та методи управління фірмою	15					+		

2	Зручне та інформативне програмне забезпечення	18						+	
3	Уніфікована електрична модель	18						+	
4	Наявність локалізації	16					+		

Провівши аналіз мого стартапу у порівнянні з конкурентами, зваживши переваги та недоліки, можна зробити висновок, що проект є конкурентоспроможним. Основні переваги – зручне та інформативне ПЗ та уніфікованість електричної моделі.

Таблиця 4.12. SWOT- аналіз стартап-проекту

Сильні сторони: -інформативне ПЗ; -низька собівартість; -довговічність основних вузлів пристрою; -відслідковування зміни характеру забруднення; -наявність постійної команди, яка підтримує ПЗ; -інноваційний характер; -простота виготовлення пристрою;	Слабкі сторони: -недосвідченість команди; -можливість поломок; -вузька направленість; -знання пошуку клієнтів; -обмежений бюджет;
Можливості: -стати провідним гравцем на ринку; -залучення інвестицій на для нововведень; -удосконалення готового пристрою; -підвищення лояльності до продукту.	Загрози: -зниження активності клієнтів через різні фактори; -поява нових більш амбіційних конкурентів на ринку; -створення податків на даний вид пристроїв.

В Таблиця 4.12. проводимо перелік сильних та слабких сторін, а також ринкових загроз та ринкових можливостей, який складаємо на основі факторів загроз і можливостей, який ми складали раніше. Ринкові загрози та можливості на відміну від факторів ще не є реалізованими на ринку та мають певну ймовірність здійснення.

На основі SWOT-аналізу розробляємо альтернативи ринкової поведінки для виведення стартап-проекту на ринок та орієнтовний оптимальний час їх ринкової реалізації з огляду на потенційні проекти конкурентів, що можуть бути виведені на ринок.

Визначені альтернативи аналізуються з точки зору строків та ймовірності отримання ресурсів.

Таблиця 4.13. Альтернативи ринкового впровадження стартап-проекту

<i>№</i>	<i>Альтернатива (орієнтовний комплекс заходів) ринкової поведінки</i>	<i>Ймовірність отримання ресурсів</i>	<i>Строки реалізації</i>
1	Одиничне розповсюдження	Ймовірність низька	6 місяців
2	Розроблення програмного забезпечення, що матиме розширений аналіз та можливості.	Ймовірність висока	6 – 9 місяців
3	Пришвидшення окупності стартапу шляхом завищення ціни на продукт	Ймовірність мала	3 місяці
4	Надання товару в орендне користування	Ймовірність висока	3 місяці

4.3 Розроблення ринкової стратегії продукту

Розроблення ринкової стратегії першим кроком передбачає визначення стратегії охоплення ринку. Для цього проведемо опис цільових груп потенційних користувачів.

Таблиця 4.14. Вибір цільових груп потенційних споживачів

<i>№</i>	<i>Опис профілю цільової групи потенційних клієнтів</i>	<i>Готовність споживачів сприйняти продукт</i>	<i>Орієнтовний попит в межах цільової групи (сегменту)</i>	<i>Інтенсивність конкуренції в сегменті</i>	<i>Простота входу у сегмент</i>
1	Наукові підприємства	Середня	80%	30%	Середня

2	Турагенства, що займаються відправленням людей до зон відчуження	Середня	90%	60%	Середня
Які цільові групи обрано: Наукові підприємства, та турагенства, що займаються відправленням людей до зон відчуження					

Виходячи з результатів аналізу потенційних груп користувачів ми обираємо такі цільові групи, як наукові підприємства, та турагенства, що займаються відправленням людей до зон відчуження. Саме для них ми будемо пропонувати нашу автоматизовану систему для реєстрації іонізуючого випромінювання

Для роботи в даному сегменті ринку потрібно сформулювати базову стратегію розвитку.

Таблиця 4.15. Визначення базової стратегії розвитку

№	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку*
1	Вдосконалення	Диференційний маркетинг	Автоматизація, зручність, якість, ціна, габарити	Стратегія диференціації

За базову стратегію розвитку було взято стратегію диференціації, що передбачає надання товару важливих, з точки зору споживача, відмінних властивостей, які роблять систему, яка відрізняється від тієї, що є в конкурентів

Таблиця 4.16. Визначення базової стратегії конкурентної поведінки

№	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки*
1	Так	Компанія буде шукати нових споживачів	Не буде копіювати основні характеристики	Стратегія заняття конкурентної ніші

За базову стратегію конкурентної поведінки була прийнята стратегія зайняття конкурентної ніші, коли компанія в якості цільового ринку вибирає один або декілька ринкових сегментів малого розміру. Головне завдання компанії при цьому – це постійна підтримка та розвиток своєї конкурентної переваги.

На основі вимог споживачів з обраного сегменту до постачальника і продукту, а також в залежності від стратегії розвитку та стратегії конкурентної поведінки розробляємо стратегію позиціонування яка визначається у формування ринкової позиції, за яким споживачі мають ідентифікувати проект

Таблиця 4.17. Визначення стратегії позиціонування

<i>№</i>	<i>Вимоги до товару цільової аудиторії</i>	<i>Базова стратегія розвитку</i>	<i>Ключові конкурентоспроможні позиції власного стартап-проекту</i>	<i>Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)</i>
1	Автоматизованість, та вдосконалення в залежності від вимог споживача	Стратегія диференціації	Автоматизація, зручність, якість, ціна, габарити	Зручність, автоматизованість, вивід результатів в зручному форматі

Результатом даного підрозділу є система рішень щодо ринкової поведінки компанії, вона визначає в якому напрямі буде працювати компанія на ринку.

4.4 Розроблення маркетингової програми стартап-проекту

Під час розроблення маркетингової програми першим кроком є формулювання маркетингової концепції товару, котру отримає користувач.

Для цього підсумуємо результати попереднього аналізу конкурентоспроможності товару.

Таблиця 4.18. Визначення ключових переваг концепції потенційного товару

<i>№</i>	<i>Потреба</i>	<i>Вигода, яку пропонує товар</i>	<i>Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)</i>
----------	----------------	-----------------------------------	---

1	Ціна	Низька ціна, за рахунок власного виготовлення ключових компонентів	Збільшення кількості клієнтів
2	Автоматизація	Із-за присутності в системі GPS, не система автоматично запам'ятовує в якій координаті, яке було виміряне значення	Зручність для користувача
3	Інтеграція зі смартфоном, через власне ПЗ	Зручне передавання вимірних даних за допомогою смартфона, та інформативна подача результатів	Автоматизована обробка результатів вимірювання, і подача їх в зручному форматі

В даній таблиці формуємо комплекс асоціацій за якими споживачі будуть ідентифікувати наш торгівельний проект.

Результатом даного підрозділу є система рішень щодо ринкової поведінки компанії, вона визначає в якому напрямі буде працювати компанія на ринку.

Розробимо трирівневу маркетингову модель товару: уточняється ідея продукту та/або послуги, його фізичні складові, особливості процесу його надання (Таблиця 4.19.).

Таблиця 4.19. Опис трьох рівнів моделі товару

<i>Рівні товару</i>	<i>Сутність та складові</i>		
<i>I. Товар за задумом</i>	<i>Автоматизована система вимірювання напруженості електростатичного поля атмосфери на місцевості</i>		
<i>II. Товар у реальному виконанні</i>	Властивості/характеристики	М/Нм	Вр/Тх /Тл/Е/Ор
	1. Довговічність	Нм	Тх
	2. Гарантійний термін	Нм	Е
	3. Досконалість виробничого виконання	Нм	Тл
	4. Вартість обслуговування	М	Вр
	5. Технологічність	М	Тх
	Якість: стандарти		
	Пакування: коробка		
	Марка: назва організації розробника		
<i>III. Товар із підкріпленням</i>	До продажу: Демонстрація можливостей приладу		
	Після продажу: після продажне обслуговування		

За рахунок чого потенційний товар буде захищено від копіювання: патент, свідоцтво на авторське право

В Таблиця 4.19. ми створюємо трьохрівневу модель нашого товару що включає задум товару та його вигоди, основі характеристики готового товару, спосіб його пакування та захисту від копіювання та плагіату.

Визначимо цінові межі, якими необхідно керуватись при встановленні ціни на потенційний товар, яке передбачає аналіз ціни на товари-аналоги або товари субститути, а також аналіз рівня доходів цільової групи споживачів (Таблиця 20.). Аналіз проводиться експертним методом.

Таблиця 4.20. Визначення меж встановлення ціни

<i>№</i>	<i>Рівень цін на товаризамінники</i>	<i>Рівень цін на товари аналоги</i>	<i>Рівень доходів цільової групи споживачів</i>	<i>Верхня та нижня межі встановлення ціни на товар/послугу</i>
1	20 тис	5 тис – 30 тис	100 тис- 200 тис	2 тис – 5 тис

Таблиця 4.20. проведений аналіз ринкової ціни на товари аналоги та товари заміники, а також середній рівень доходів користувачів. За одержаними даними встановлена верхня та нижня границя на наш проект.

Визначимо оптимальну систему збуту.

Таблиця 4.21. Формування системи збуту

<i>№</i>	<i>Специфіка закупівельної поведінки цільових клієнтів</i>	<i>Функції збуту, які має виконувати постачальник товару</i>	<i>Глибина каналу збуту</i>	<i>Оптимальна система збуту</i>
1	Роздрібна продаж	Продаж клієнтам продукції	Висока	Проведення збуту силами фірм посередників, та власними силами

Було проведено розробку концепцію маркетингових комунікацій між

користувачами та виробником. В нашому випадку, дана робота з виробником та реалізація продукту через прямий продаж.

Розробимо концепцію маркетингової комунікації, котра опирається на попередньо обрану основу для позиціонування, обрану специфіку поведінки користувачів.

Таблиця 4.22. Концепція маркетингових комунікацій

<i>№</i>	<i>Специфіка поведінки цільових клієнтів</i>	<i>Канали комунікацій якими користуються цільові клієнти</i>	<i>Ключові позиції, обрані для позиціонування</i>	<i>Завдання рекламного повідомлення</i>	<i>Концепція рекламного звернення</i>
1	Пошук в інтернеті через спеціальні ресурси, та нові патенти	Інтернет, прямий зв'язок з виробниками	Автоматизація, ціна, якість	Акцентування уваги клієнтів, на широкі можливості, автоматизацію процесу	Ціна, автоматизація

Спираючись на специфіку закупівельної поведінки цільових клієнтів було обрано власну систему збуту, коли виробник безпосередньо продає товар клієнту. Ціноутворення відбувається на основі аналізу товарів – аналогів і відбувається під час фінансово-економічного аналізу проекту з залученням експертів.

Представимо команду стартап-проекту та її розвиток.

Таблиця 4.23. Команда стартап-проекту

<i>Розвиток команди</i>	<i>Учасники</i>	<i>Завдання учасників в</i>	<i>Освіта</i>	<i>Досвід роботи</i>	<i>Спеціалізовані знання</i>	<i>Витрати, тис. грн.</i>

Мінімальна команда	Мельник Андрій	Керівник; розробни; бухгалтер	Бакалавр	Інженер; розробни; електрик; програміст	Наявні певні знання в розробці інтелектуальних систем, створенні різних інженерних рішень, знання електроніки та електрики, знання теорії програмування, та мови програмування C#	10000
Разом:						10000

Дана таблиця представляє повну команду розробників системи, що являється мінімальною життєздатною командою, але дана команда без зволікань може повністю представляти інтереси стартап-проекту

Розробимо календарний графік реалізації стартап-проекту.

Таблиця 4.24. Календарний графік реалізації стартап-проекту

Зміст етапу	Період реалізації, номер місяця з початку виконання робіт													Вартість етапу, грн.
	1	2	3	4	5	6	7	8	9	10	11	12	13	
Розробка системи														2000
Визначення кошторису														5000
Проектування системи														30000
Проектування конструкції														25000
Створення програмного забезпечення														23000

Створення прототипу														10000
Тестування														7500
Виявлення та усунення недоліків														15000
Запуск реклами														10000
Запуск виробництва														50000
Всього, грн														177500

Таблиця 4.24. представляє собою календарний план-графік реалізації стартап проекту, в якій передбачені всі можливі витрати з початку розробки аж до самого виробництва, в гривнях.

Визначимо ціни на сировину, матеріали та комплектуючі виробу та визначимо постачальників

Таблиця 4.25. Сировина, матеріали та комплектуючі виробу

№	Найменування видів сировини, матеріалів та комплектуючих виробів	Постачальники	Ціна за одиницю	Примітки (місцезнаходження постачальників, умови поставок тощо)
1.	Трубка Гейгера-Мюллера	Купуємо	500	Купуємо
2.	STM32	Інвестори можуть бути постачальниками	100	Постачальники знаходяться в Китаї; поставки виконуються за замовленням
4.	Корпусні деталі		300	
5.	GPS		250	
6.	Плати		100	
Разом:			1400	

В даній таблиці прописані матеріали, що являються необхідними для розробки системи описаної даним стартап-проектом. Враховуючи те що в даній таблиці вказана роздрібна ціна, то при масовому виробництві матеріали будуть постачатись по оптовим цінам і в результаті затрати будуть декілька меншими.

Таблиця 4.26. Потреба в промислово-виробничому персоналі

<i>№</i>	<i>Посада/виконувані завдання</i>	<i>Чисельність</i>	<i>Витрати на персонал, тис. грн.</i>
1.	Керівник	1	15000
2.	Бухгалтер	1	10000
3.	Інженер	1	10000
4.	Інженер-електронщик	1	10000
6.	Програміст	1	10000
7.	Маркетолог	1	9000
8.	Робітники на виробництво	2	10000
	Разом:	9	84000

Враховуючи те, що на самому початку виробництва кількість робочого персоналу не дуже велика, тому обов'язки керівника, бухгалтера, інженера та маркетолога будуть виконуватися однією людиною. В подальшому при стабільному розвитку стартап-проекту буде проводитися розширення персоналу.

Далі необхідно визначити початкові витрати необхідні для стартапу.

Таблиця 4.27. План витрат на запуск виробництва продукції

<i>№</i>	<i>Найменування</i>	<i>Характеристика</i>	<i>Вартість, тис. грн.</i>
1.	Закупівля обладнання та устаткування	Витрати на придбання обладнання	70000
2.	Сировина, основні матеріали та комплектуючі	Вартість сировини та матеріалів для забезпечення технологічного процесу	1400
4.	Паливо та електроенергія необхідна на технологічні цілі	Витрати на електроенергію, а також на паливо, необхідні для запуску проектної потужності	15000
5.	Оплата праці персоналу	Витрати на заробітну плату	84000
6.	Освоєння та запуск виробництва	Витрати на пускові роботи, запуск виробництва	20000
	Разом		190400

В даній таблиці було визначено початковий план витрат, необхідний для запуску виробництва.

Визначимо загальні обсяги інвестицій

Таблиця 4.28. Обсяги інвестицій

<i>№</i>	<i>Вид інвестицій</i>	<i>Сума, грн</i>
1.	Першо-початкові інвестиції	190400
2.	Інвестиції на стадії виробництва	100000
3.	Інвестиції на просування стартапу	40000
Разом:		330400

В даному підрозділі було визначено загальні обсяги інвестицій необхідних для запуску стартап-проекту, в які входять ціни на сировину, матеріали, комплектуючі, заробітна плата працівникам, просування та першо- початкові інвестиції.

Висновки до розділу

Узагальнюючи проведений аналіз стартап проекту під назвою «Автоматизований прилад дозиметричного контролю іонізуючого випромінювання» можна зробити

и висновок, що ринкова комерціалізація проекту опирається на наявний попит, та рентабельність продукту на ринку. Беручи до уваги потенційні групи користувачів та клієнтів, перспектива впровадження проекту є достатньо високою, також цьому сприяє, малу кількість прямих конкурентів, так як даний прилад має більший функціонал. Це не просто вимірювач, які вже представлені на ринку, а ціла автоматизована система.. Також цьому сприяє наявність власного ПЗ для відображення результатів вимірювання.

Щоб даний прилад був на високій позиції в рейтингу в користувачів, потрібно постійно його розвивати. Можливий розвиток системи підвищення точності приладу, створення ПЗ для комп'ютерних пристроїв, додавання систем безпроводної передачі даних, покращення інформативності виводу результатів, програмне розпізнавання аномальних зон.

Продукт відразу буде націлений на міжнародний ринок. Також передбачений захист на даний прилад в вигляді, реєстрації авторських прав та створення патентів на власні технології, що використовуються в даному проекті. В даному розділі також був виконаний розрахунок інвестицій, для запускуданого стартап-проекту.

ВИСНОВКИ

В даній роботі було проаналізовано існуючі методи реєстрації іонізуючого випромінення. Детальний аналіз типів детекторів дозволив обґрунтувати вибір до застосування в системі газових лічильників. Даний тип детекторів задовольняє потреби, представленні перед створенням системи. В якості датчику було обрано лічильник Гейгера-Мюллера – трубка СБМ-20. На базі нього було прийнято рішення про розробку алгоритму, який допоможе збалансувати енергоефективність та мати високий рівень швидкодії, при реєстрації випромінення вищого за норму. Розроблено пристрій та написано ПЗ для обробки та візуалізації сигналу отриманого з датчику.

Також було розроблено додаток для мобільного засобу, який реєструє, обробляє та візуалізує сигнал з пристрою. Логіка обробки сигналу написана відповідно до потреб, які було отримано в процесі аналізу чистого сигналу. Вона поділяється на дві частини: первинна обробка сигналу для чіткого виокремлення імпульсів від шумів та логіка обрахунку дози випромінення з акцентом на статичну кількість імпульсів.

В роботі представлено можливості розширення проекту для використання не лише на підприємствах та побуті, а й на рівні країни. Також описані інструменти та можливості зручного та швидкого аналізу даних, отриманих централізовано з багатьох пристроїв вимірювання.

Список використаних джерел

- [1] Іванов Є. Радіоекологічні дослідження : навч. посібник / Євген Іванов. – Львів: Видавничий центр ЛНУ імені Івана Франка, 2004. – 149 с.
- [2] Голінько В.І. Радіаційна безпека. Навчальний посібник. – Д.: НГУ, 2018. – 153 с.
- [3] Новіков, С.Г.; Беринцев, А.В.; Алексєєв, А.С.; Сомов, А.І.; Приходько, В.В.; Гуськов, П.А.; Светухін, В.В. «Розробка пересонального переносного дозиметруа ефективної дози із каналом даних RFID» Радіопромисловість, стор. 78-85, 2018.
- [4] Б. С., «Экономиный бытовой радиомер,» Радио, т. 11, pp. 30-32, 2010.
- [5] Momose, Y.; Sakurai, T.; Nakayama, K. « Thermal Analysis of Photoelectron Emission (PE) and X-ray Photoelectron Spectroscopy (XPS) Data for Iron Surfaces Scratched in Air, Water, and Liquid Organics» pp. 11-21, 2020.
- [6] Mei, Z.; Kuts, Y.; Kochan, O.; Lysenko, I.; Levchenko, O.; Vlach-Vyhrynovska, H. Using Signal Phase in Computerized Systems of Non-destructive Testing. Meas. Sci. Rev. 2022, 22, 32–43.
- [7] I. Lysenko, V. Uchanin, V. Petryk, Y. Kuts, A. Protasov and A. Alexiev, "Intelligent Automated Eddy Current System for Monitoring the Aircraft Structure Condition," 2022 IEEE 3rd International Conference on System Analysis & Intelligent Computing (SAIC), 2022, pp. 1-5.
- [8] Batyuk, A.; Voityshyn, V.; Verhun, V. «Software Architecture Design of the Real-Time Processes Monitoring Platform. In Proceedings of the Second International Conference on Data Stream Mining & Processing (DSMP)» pp. 98-101, 2018.
- [9] Parkhomenko, A.; Gladkova, O.; Sokolyanskii, A.; Shepelenko, V.; Zalubovskiy, Y. « Implementation of reusable solutions for remote laboratory development,» iJOE, pp. 24-29, 2016.
- [10] Md. Akbar Hossain, Sayan Kumar Ray. A Smartphone-Assisted Device-to- Device Communication for Post-disaster Recovery., 2018.
- [11] Holovatyy, A.; Teslyuk, V.; Iwaniec, M.; Mashevska, M. «Development of Background Radiation Monitoring System Based on Arduino Platform. In Proceedings of the 2018 IEEE 13th International Scientific and Technical Conference on Computer S,» pp. 139-142, 2018.
- [12] Р. Романський, «Підхід до математичного моделювання та дослідження

комп'ютерних процесів на макрорівні» Математика , стор. 18-38, 2020.

- [13] Hacker E, Horsham C, Vagenas D, Jones L, Lowe J, Janda M. A mobile technology intervention with ultraviolet radiation dosimeters and smartphone apps for skin cancer prevention in young adults: randomized controlled trial., JMIR Mhealth Uhealth, 2018.
- [14] Holovaty, A.; Teslyuk, V.; Iwaniec, M.; Mashevskaya, M., «Development of a system for monitoring vibration accelerations based on the raspberry pi microcomputer and the adxl345 accelerometer.,» pp. 52-62, 2017.
- [15] Куц Ю.В. Новітні системи та технології: навчальний посібник / Ю. В. Куц, Ю. Ю. Лисенко, А.С. Момот; КПІ ім. Ігоря Сікорського. – Київ: КПІ ім. Ігоря Сікорського, 2022. – 123 с.
- [16] Баженов В.Г. Електроніка. Лабораторний практикум: навчальний посібник / В. Г. Баженов, Є. Ф. Суслов, Ю. Ю. Лисенко, А.С. Момот; КПІ ім. Ігоря Сікорського. – Київ: КПІ ім. Ігоря Сікорського, 2022. – 70 с.
- [17] Протасов А.Г. Технології теплового неруйнівного контролю [Електронний ресурс]: підручник для студ. спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології» / А. Г. Протасов, Ю. Ю. Лисенко; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 3,6 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2021. – 133 с.
- [18] Kuts, Y., Kochan, O., Lysenko, I., Huminilovych, R. (2021). Methodology for measuring phase shifts of signals using discrete Hilbert transform. In 13th International Conference on Measurement. IEEE, 18-21. <https://doi.org/10.23919/Measurement52780.2021.9446811>
- [19] Petryk V. F. SmartphoneBased Automated Non-Destructive Testing Devices / V. F. Petryk, A. G. Protasov, R. M. Galagan, A. V. Muraviov, I. I. Lysenko // Devices and Methods of Measurements. – 2020. – 11(4), PP. 272-278. doi:10.21122/2220-9506-2020-11-4-272-278.
- [20] Куц Ю.В. Спеціальні розділи математики. Курс лекцій: навчальний посібник / Ю. В. Куц, Ю. Ю. Лисенко; КПІ ім. Ігоря Сікорського. – Київ: КПІ ім. Ігоря Сікорського, 2022. – 180 с.

Додатки

Код програми надається за запитом до авторів

*Код програми
надається за запитом до авторів*

Код програми надається за запитом до авторів

Код програми надається за запитом до авторів

*Код програми надається за
запитом до авторів*

Код програми надається за запитом до авторів

Код програми надається за запитом до авторів

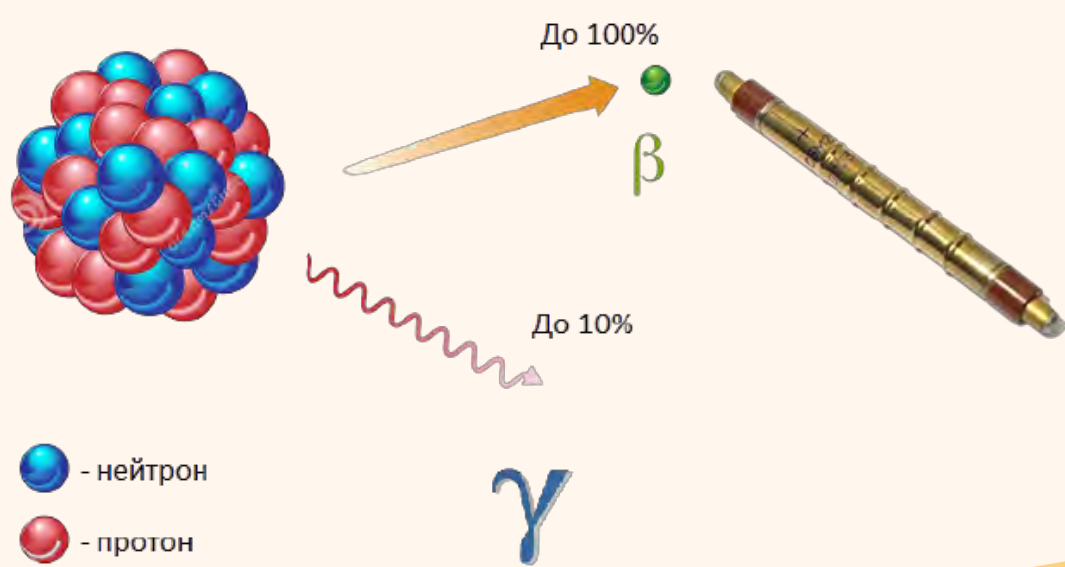


Оптимізація алгоритму опрацювання сигналів в системі автоматизованого контролю іонізуючого випромінювання



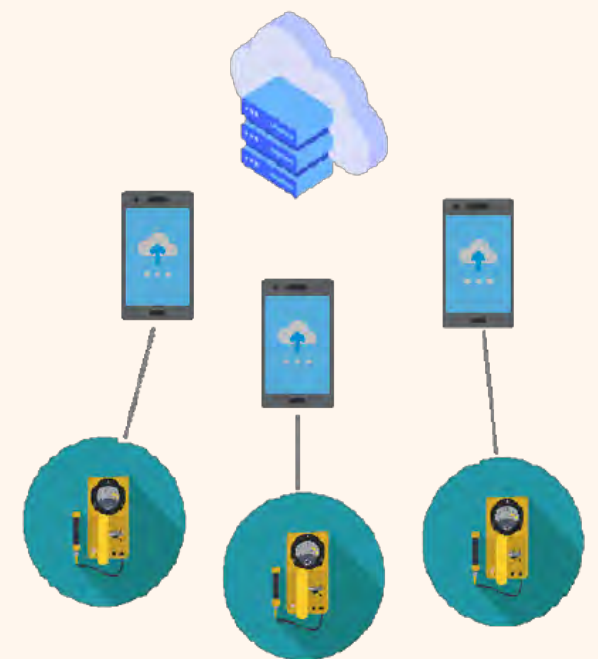
Ідея даної роботи полягає в покращенні алгоритму обробки сигналів з пристрою програмним шляхом та оптимізація алгоритму обробки та усереднення значень, які при їх використанні забезпечують достатню точність та необхідну швидкодію реакції відповідно рівня радіоактивного фону.

Поєднання недорогого датчика іонізуючого випромінювання із смартфоном дозволяє отримати компактний та зручний для використання прилад. Передбачено об'єднання подібних приладів з метою створення цілої екосистеми, пристрої якої збирають та передають інформацію про стан навколишнього середовища із прив'язкою до координат та часу до бази даних на сервер. Отримана таким чином інформація може бути використана для побудови загальної карти випромінювання підприємства/місцевості/міста/області/країни.



Основні аспекти інноваційності:

- Оптимізована швидкодія відображення результатів,
- функція інтеграції до інших пристроїв,
- функція відслідковування забруднення відвіданих користувачем місць,
- функцію синхронізації виміряних значень з мапою місцевості,
- функція сповіщення про наявність небезпеки,
- передача результатів вимірювання до загальної бази даних для подальшого аналізу.



На базі даної зв'язки даного дозиметру і ПЗ можна реалізувати декілька систем:

- Система відслідковування зміни іонізуючого випромінювання в ризикових місцях(АЕС, кладовище радіаційних залишків тощо);
- Система моніторингу іонізуючого випромінювання по всьому світу, збираючи дані у звичайних користувачів, що купили наш прилад;
- Система моніторингу персоналу на АЕС. Наприклад, на початку зміни, персоналу видали такі дозиметри, а оператор може відслідковувати за тим який зараз фон там, де вони знаходяться, та накопичена кількість іонізуючого випромінювання, в разі перевищення норм оператору прийде сповіщення, та він зможе побачити, яке перевищення і в кого відбулось, та прийняти певні заходи.



Виконав:

Студент II курсу, групи ПК-11мп
Мельник Андрій Вікторович

Науковий керівник:

Доцент, кандидат технічних наук
Лисенко Юлія Юріївна



Інфографіка

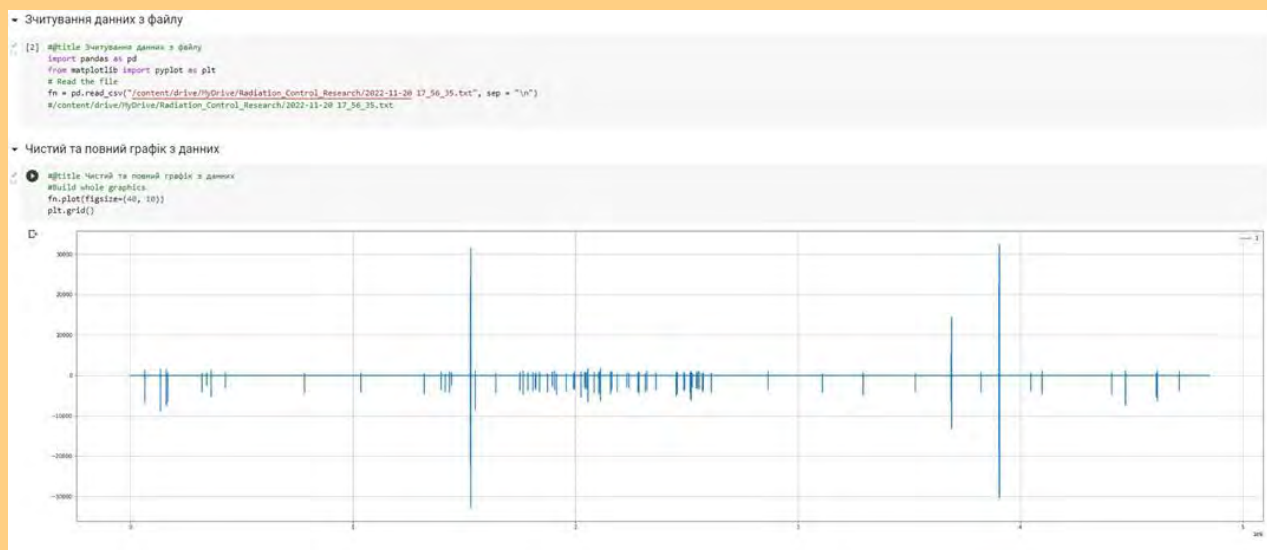


Рисунок 1. Візуалізація не оброблених даних з файлу за один сеанс виміру

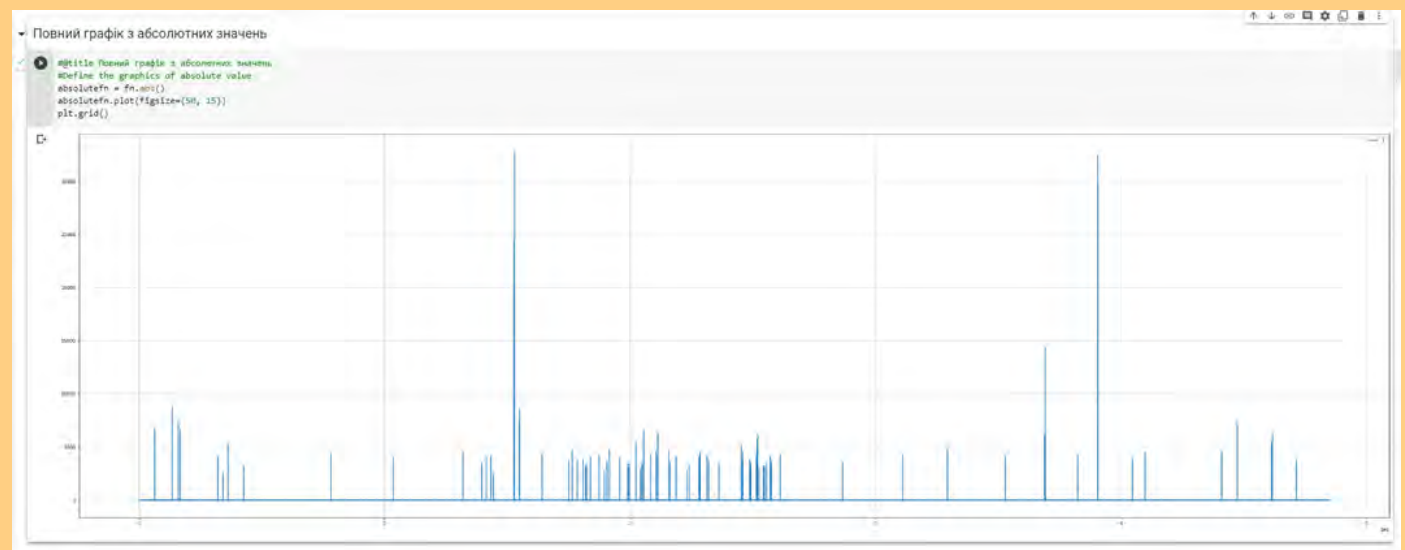


Рисунок 2. Візуалізація різких стрибків амплітуди в абсолютних значеннях

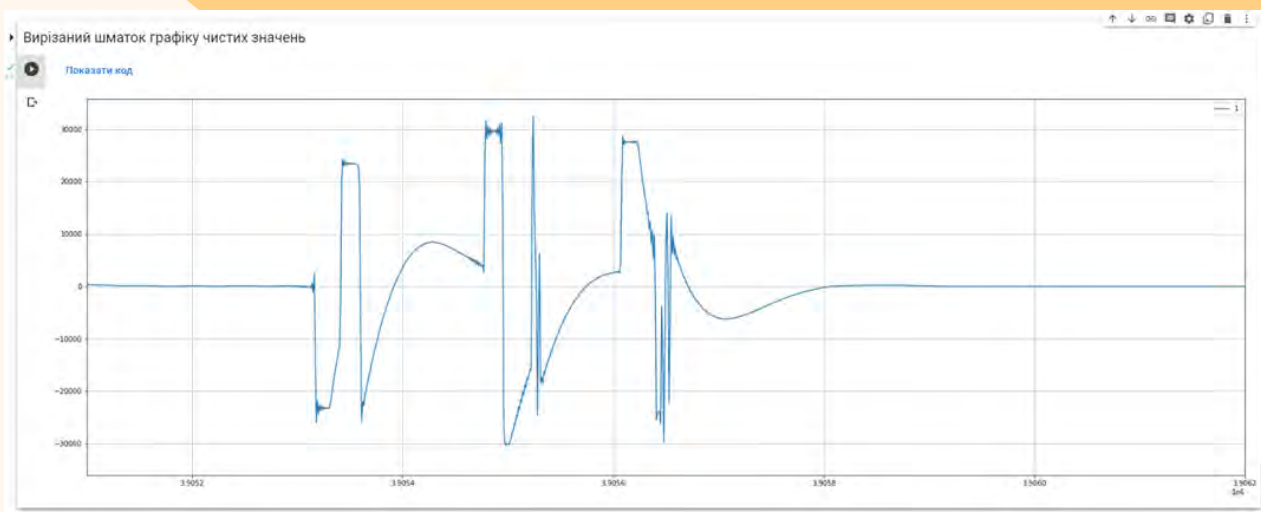


Рисунок 3. Фрагмент графіку з різкими скачками амплітуди

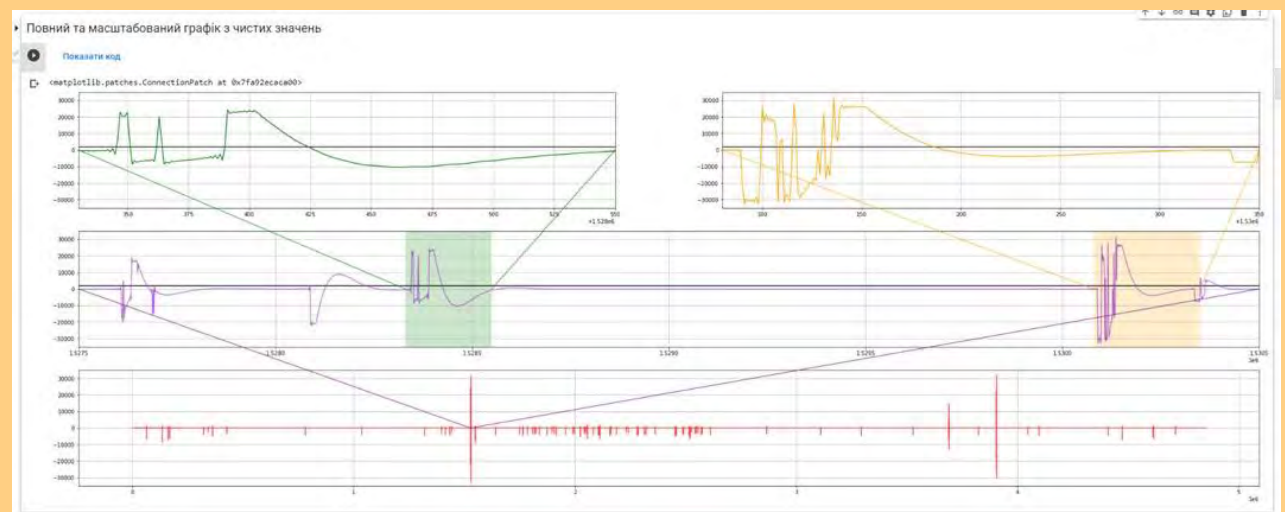


Рисунок 4. Повний графік з масштабуванням для паралельного дослідження декількох імпульсів

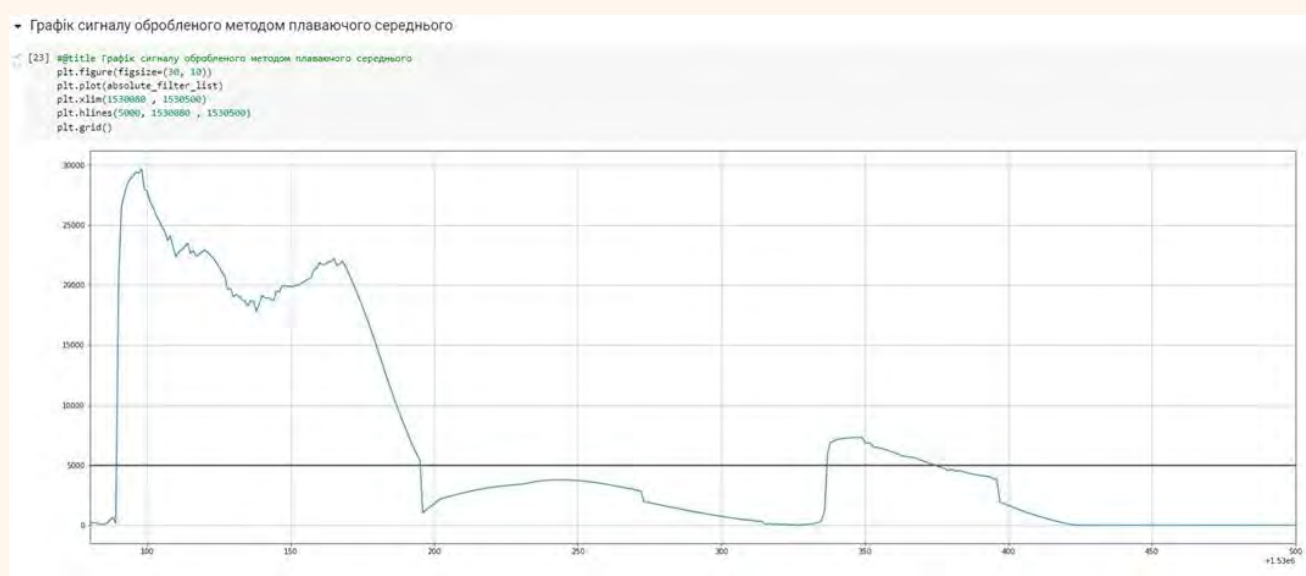


Рисунок 5. Фрагмент сигналу обробленого алгоритмом плаваючого середнього

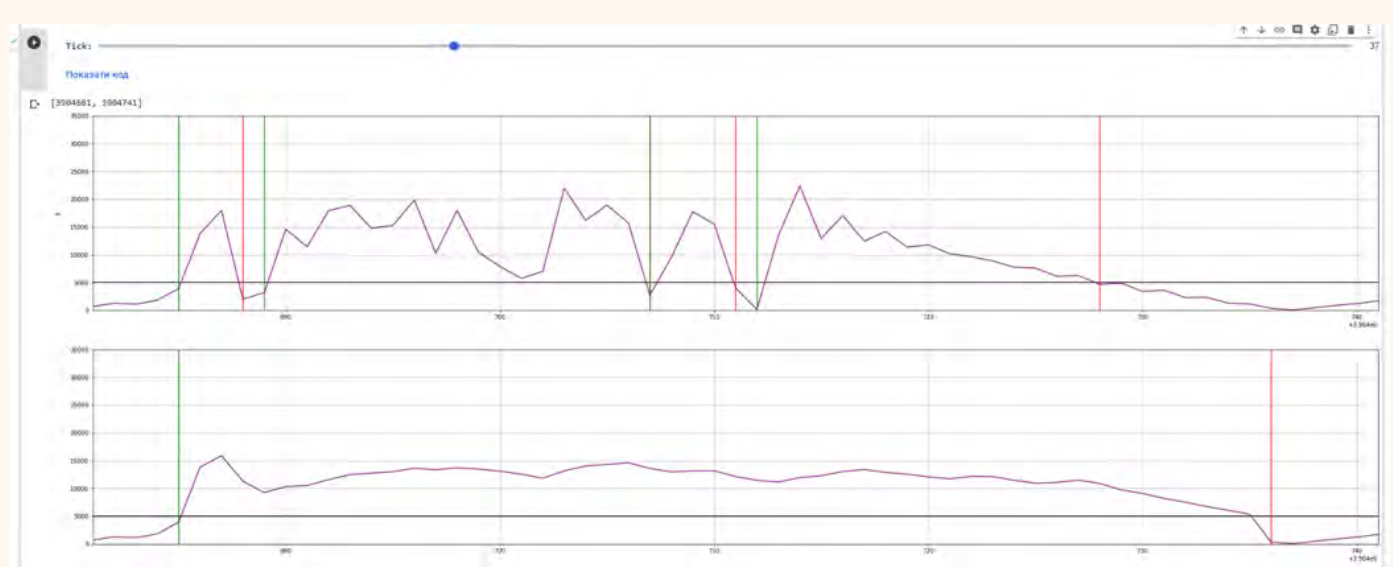


Рисунок 6. Візуалізація фрагменту обробленого сигналу

Виконав:

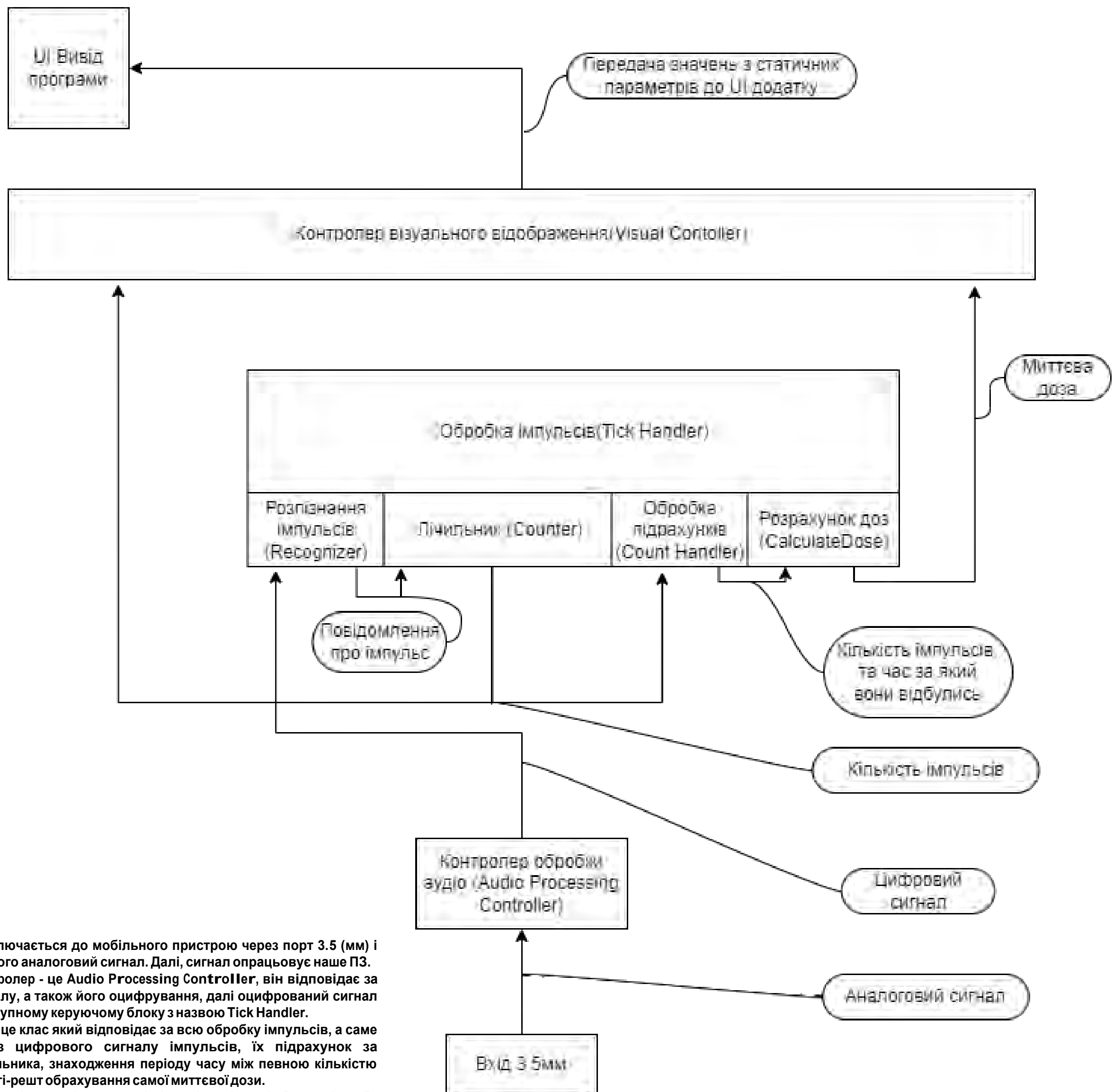
Студент II курсу, групи ПК-11мп
Мельник Андрій Вікторович

Науковий керівник:

Доцент, кандидат технічних наук
Лисенко Юлія Юріївна



Схема взаємодії ПЗ



Виконав:

Студент II курсу, групи ПК-11мп
Мельник Андрій Вікторович

Науковий керівник:

Доцент, кандидат технічних наук
Лисенко Юлія Юріївна