

Центральноукраїнський національний технічний університет  
Механіко-технологічний факультет  
Кафедра кібербезпеки та програмного забезпечення

”Допущено до захисту”  
Завідувач кафедри кібербезпеки  
та програмного забезпечення  
д.т.н., професор  
\_\_\_\_\_ Олексій СМІРНОВ  
“ \_\_\_\_ ” \_\_\_\_\_ 2024 р.

**ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА**  
**за другим (магістерським) рівнем вищої освіти**  
на тему  
**“ Дослідження та програмна реалізація системи діагностики,  
обробки та запису сигналів ЕКГ на базі флеш-накопичувачів ”**

КБПЗ - 2024

Виконав здобувач вищої освіти  
II курсу, групи КІ-23М  
ОПП «Комп’ютерна інженерія»  
спеціальності 123 «Комп’ютерна інженерія»  
\_\_\_\_\_ Литвиненко Д.О.  
« \_\_\_\_ » \_\_\_\_\_ 2024 р.

Керівник проекту  
кандидат технічних наук, доцент  
\_\_\_\_\_ Кислун О.А.  
« \_\_\_\_ » \_\_\_\_\_ 2024 р.  
Рецензент \_\_\_\_\_  
\_\_\_\_\_

Центральноукраїнський національний технічний університет  
Факультет Механіко-технологічний  
Кафедра Кібербезпеки та програмного забезпечення  
Рівень вищої освіти магістр  
Галузь знань 12 "Інформаційні технології"  
Спеціальність 123 "Комп'ютерна інженерія"  
Освітньо-професійна (освітньо-наукова) програма "Комп'ютерна інженерія"

ЗАТВЕРДЖУЮ

Завідувач кафедри

д.т.н., проф.

Олексій СМІРНОВ

« » 2024 року

## ЗАВДАННЯ НА ВИПУСКНУ КВАЛІФІКАЦІЙНУ РОБОТУ ЗА ДРУГИМ (МАГІСТЕРСЬКИМ) РІВНЕМ ВИЩОЇ ОСВІТИ ЗДОБУВАЧА ВИЩОЇ ОСВІТИ

Литвиненку Дмитру Олеговичу

(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження та програмна реалізація системи діагностики, обробки та запису сигналів ЕКГ на баз флеш-накопичувачів

2. Керівник роботи Кислун Олег Андрійович, канд. техн. наук, доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу №19-13 від 07.08.2024 року

3. Строк подання студентом роботи до захисту 02.12.2024 р.

4. Мета та завдання випускної кваліфікаційної роботи: Метою розробки є дослідження та програмна реалізація системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів

5. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1. Призначення та область використання.

6. Наукова новизна.

2. Перегляд аналогічних існуючих систем.

7. Маркетингове та економічне обґрунтування ІТ проекту.

3. Опис і обґрунтування проектних рішень.

8. Заходи з охорони праці та техніки безпеки.

4. Етапи програмування системи.

9. Висновки.

5. Впровадження системи в промислову експлуатацію

6. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

Наукова новизна

1 аркуш

Структурна схема системи

1 аркуш

Функціональна схема системи

1 аркуш

Діаграма процесів

1 аркуш

Блок-схема алгоритму роботи додатку

2 аркуша

Маркетингове та економічне обґрунтування ІТ проекту

1 аркуш

## 6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Економічний	Доренська А.О.	05.10.2024	14.11.2024
Охорона праці	Марченко К.М.	06.10.2024	16.11.2024

7. Дата видачі завдання «      » 2024 р.

## КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти	Строк виконання етапів випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти	Примітка
1.	Аналіз існуючих систем	10.10.2024р.	
2.	Постановка задачі, оформлення ТЗ	15.10.2024р.	
3.	Розробка моделі компонента	20.10.2024р.	
4.	Розробка структур даних	25.10.2024р.	
5.	Розробка алгоритмів зв'язку та відображення	30.10.2024р.	
6.	Програмування алгоритмів	10.11.2024р.	
7.	Маркетингове та економічне обґрунтування ІТ проекту	13.11.2024р.	
8.	Розрахунки з охорони праці та техніки безпеки	15.11.2024р.	
9.	Оформлення ПЗ	17.11.2024р.	
10.	Попередній захист роботи	02.12.2024р.	

Дата видачі завдання  
«      » 2024р.

Підпис керівника

\_\_\_\_\_

Кислун О.А.  
(прізвище та ініціали)

Завдання прийнято до виконання  
«      » 2024 р.

Підпис здобувача

\_\_\_\_\_

Литвиненко Д.О.  
(прізвище та ініціали)

## АНОТАЦІЯ

**Литвиненко Д.О. Дослідження та програмна реалізація системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів. 123 Комп'ютерна інженерія. Центральноукраїнський національний технічний університет. Кропивницький. 2024.**

В даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів.

Метою розробки є дослідження та програмна реалізація системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів.

Об'єктом дослідження є процес реєстрації та запису сигналів ЕКГ.

Предметом дослідження є методи та засоби реєстрації та запису сигналів ЕКГ.

Методи дослідження базуються на методах цифрової обробки сигналів, методах математичної статистики, методах розробки програмного забезпечення.

Результат роботи – програмна реалізація системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів.

В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

Розроблено зручний інтерфейс користувача. Наведені інструкції по роботі з програмними засобами.

Програма може використовуватися на ПЕОМ архітектури IBM PC з ОС Windows 10/11.

Програму розроблено в середовищі AVR Studio.

**Ключові слова:** комп'ютерна інженерія, обробка сигналів, збір інформації

## ABSTRACT

**Lytvynenko D.O. Research and software implementation of a system of diagnostics, processing and recording of ECG signals based on flash drives. 123 Computer engineering. Central Ukrainian National Technical University. Kropyvnytskyi. 2024.**

In this graduation thesis for the second (master's) level of higher education, software was developed, which is intended for the system of diagnostics, processing and recording of ECG signals based on flash drives.

The purpose of the development is the research and software implementation of a system of diagnostics, processing and recording of ECG signals based on flash drives.

The object of research is the process of recording and recording ECG signals.

The subject of research is methods and means of recording and recording ECG signals.

Research methods are based on digital signal processing methods, mathematical statistics methods, and software development methods.

The result of the work is the software implementation of the system of diagnostics, processing and recording of ECG signals based on flash drives.

In the process of working on the software model, an analysis of existing hardware and software was performed. All components of the developed software are fully described.

A convenient user interface has been developed. Instructions for working with software tools are provided.

The program can be used on PCs of IBM PC architecture with Windows 10/11 OS.

The program was developed in the AVR Studio environment.

**Keywords:** computer engineering, signal processing, information collection

№ рядку	Формат	Позначення	Найменування	Кількість аркушів	№ екз.	Примітка
			Текстові документи			
1	A4	ВКРМ-123.24.0019.00.00.ПЗ	Пояснювальна записка	87	—	
2	A4	ВКРМ-123.24.0019.00.00.ТЗ	Технічне завдання	6	—	Додаток А
3	A4	ВКРМ-123.24.0019.00.00.РП	Робоча програма	35	—	Додаток Б
			Графічні документи			
4	A4	ВКРМ-123.24.0019.00.00.НН	Наукова новизна	1	—	
5	A4	ВКРМ-123.24.0019.00.00.Е1	Структурна схема	1	—	
6	A4	ВКРМ-123.24.0019.00.00.Е2	Функціональна схема роботи системи	1	—	
7	A4	ВКРМ-123.24.0019.00.00.Д1	Діаграма взаємодії процесів	1	—	
8	A4	ВКРМ-123.24.0019.00.00.Д2	Блок-схема основної програми	1	—	
9	A4	ВКРМ-123.24.0019.00.01.Д2	Блок-схема роботи підпрограми	1	—	
10	A4	ВКРМ-123.24.0019.00.00.Д3	Маркетингове та економічне обґрунтування ІТ - проекту	1	—	

					<b>ВКРМ-123.24.0019.00.00.ВП</b>			
Вим.	Арк.	№ документа	Підпис	Дата				
Розробив		Литвиненко Д.О.			Дослідження та програмна реалізація системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів	Літ.	Аркуш	Аркуші
Перевірів		Кислун О.А.				М		1
Н. Контр.		Коваленко А.С.			ЦНТУ КІ-23М			
Затв.		Смірнов О.А.						

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ .....	3
ВСТУП.....	4
1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ .....	7
1.1 Призначення системи.....	7
1.2 Область застосування.....	8
2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ .....	10
2.1 Огляд існуючих систем, технологій, архітектур та програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти.....	10
2.2 Обґрунтування вибору засобів для побудови системи та мови програмування.....	21
2.3 Розгорнута постановка завдання .....	32
3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ .....	34
3.1 Опис функціонування системи .....	34
3.2 Розробка структурної схеми.....	38
3.3 Розробка функціональної схеми .....	40
3.4 Розробка діаграми процесів.....	42
4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ.....	45
4.1 Розробка блок-схем та опис алгоритмів функціонування системи.....	45
4.2 Захист розробленого програмного забезпечення.....	55
5 ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ .....	58
6 НАУКОВА НОВИЗНА .....	59
7 МАРКЕТИНГОВЕ ТА ЕКОНОМІЧНЕ ОБҐРУНТУВАННЯ ІТ-ПРОЄКТУ .....	60
7.1 Визначення цільової аудиторії кінцевого готового продукту .....	60
7.2 Оцінка привабливості шляхом застосування методів експертних оцінок ...	61
7.3 Вибір методу оцінки вартості ПЗ .....	63
7.4 Розрахунок економічної ефективності від впровадження реалізованого	

ПЗ як фактору його привабливості.....	64
7.5 Пропозиція алгоритму просування проєкту розробки ПЗ .....	65
7.6 Оптимізація каналів збуту та шляхів реалізації ПЗ .....	67
7.7 Визначення ключових факторів успіху конкретного проєкту.....	70
8 ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ .....	71
8.1 Вступ.....	71
8.2 Аналіз умов праці на робочому місці ІТ-фахівця .....	72
8.3 Пропозиції щодо підвищення працездатності ІТ-фахівців.....	74
8.4 Розрахунок системи загального штучного освітлення виробничого приміщення де працюють ІТ-фахівці .....	76
8.5 Висновки до розділу.....	79
9 ОСНОВНІ ВИСНОВКИ.....	81
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....	83

КБПЗ\_2024

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк. 2
Вим.	Арк.	№ докум.	Підпис	Дата		

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

ЕКГ	–	Електрокардіограма
ЕКС	–	Електрокардіосигнал
КМ	–	Кардіомонітор
АЦП	–	Аналого-цифровий перетворювач
ЦАП	–	Цифро-аналоговий перетворювач
ЦОС	–	Цифрова обробка сигналів
ФНЧ	–	Фільтр низьких частот
ОС	–	операційна система
ПЗ	–	програмне забезпечення
ПЗП	–	постійний запам'ятовуючий пристрій
USB	–	універсальна послідовна шина
AVR	–	родина 8-бітних мікроконтролерів
ФВЧ	–	фільтр верхніх частот
	–	
	–	
	–	
	–	
	–	
	–	
	–	
	–	
	–	
	–	

## ВСТУП

**Актуальність теми.** Дослідження, виявлення і діагностика захворювань серцево-судинної системи, зокрема хвороб серця, є однією з ключових задач сучасної медицини. Електрокардіографія є одним з провідних методів вивчення біоелектричної активності серця. Вона незамінна для діагностики таких станів, як інфаркт міокарда, ішемічна хвороба серця, порушення ритму і провідності, гіпертрофія передсердь і шлуночків, а також інших серцевих захворювань. Однак традиційне медичне обладнання не завжди гарантує точні діагнози, про що свідчить велика кількість помилкових висновків. У деяких випадках точний діагноз встановлюється тільки після смерті пацієнта.

У зв'язку з цим виникає потреба в розробці апаратно-програмних засобів для автоматизації процесу діагностики серця, зокрема для зняття електрокардіограми (ЕКГ), її аналізу і надання висновків. Важливо визначити оптимальний набір структурних параметрів електрокардіографічного сигналу (ЕКС), що забезпечить найточнішу діагностику, а також розробити алгоритми для виділення і обробки характерних елементів сигналу [1].

При візуальному аналізі ЕКГ часто не приймаються в увагу чи не помічаються його незначні зміни, у яких може міститися інформація про важливі нюанси в роботі серця. Ці непомітні на око відхилення від норми можуть розвиватися в патології, тому проводиться вивчення впливу незначних змін ЕКС, які не діагностувались на звичайних кардіографах, на точність діагнозів, що виставляються, і на прогнозування стану хворого.

Для автоматизації процесу кардіодіагностики необхідно також створення великої бази даних по видах електрокардіографічного сигналу при різних захворюваннях серця. У неї повинно бути внесене як можна більше (але, не занадто – для збільшення швидкодії) кількість видів сигналів з точними висновками, виставленими кваліфікованими фахівцями. Параметри вхідних у неї сигналів будуть використані для видачі попереднього висновку на основі

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		4

аналізу ЕКГ і зіставлення. Для її створення використовуються як ЕКГ конкретних хворих, реально зняті на звичайних кардіографах, так і дані, представлені у відповідних навчальних посібниках і довідниках. Важливо також забезпечити якісне відображення досліджуваного сигналу на моніторі, його кодування і збереження у відповідному запам'ятовуючому пристрої.

В даний час вже існує багато різних типів електрокардіографів, що відрізняються друг від друга контрольованими параметрами, експлуатаційними характеристиками, методами обробки і представлення інформації. Визначення їхніх характеристик, оцінка переваг і недоліків конкретних реалізацій і поліпшення параметрів, що визначають точність діагностики, також включені в проведені дослідження.

Слід зазначити, що дана система не покликана замінити собою лікаря-кардіолога, а лише повинна полегшити його роботу і збільшити точність діагностичних досліджень стану і роботи серця.

У даній роботі розглядається питання програмної реалізації системи збору інформації на базі флеш- накопичувачів.

**Мета й завдання дослідження.** Метою роботи є дослідження та програмна реалізація системи діагностики, обробки та запису сигналів ЕКГ на базі флеш- накопичувачів.

Для досягнення поставленої мети визначена програма дослідження, що складається з наступних завдань:

- Огляд існуючих систем реєстрації та запису сигналів ЕКГ.
- Дослідження системи реєстрації та запису сигналів ЕКГ.
- Програмна реалізація системи збору інформації на базі флеш- накопичувачів.

*Об'єктом дослідження є процес реєстрації та запису сигналів ЕКГ.*

*Предметом дослідження є методи та засоби реєстрації та запису сигналів ЕКГ.*

*Методи дослідження базуються на методах цифрової обробки сигналів, методах математичної статистики, методах розробки програмного забезпечення.*

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
						5
Вим.	Арк.	№ докум.	Підпис	Дата		

**Наукова новизна отриманих результатів.** У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

– Досліджено основні методи сучасної діагностики роботи серцево-судинної системи людини, а також виділено їх основні плюси та критерії застосування. Окремо виділено метод ЕКГ, та спосіб читання отриманих результатів, а саме електрокардіограми.

– Розроблено алгоритми та методи обробки сигналів та модернізовано прилад для реєстрації ЕКГ.

**Практична цінність отриманих результатів** полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі цифрової обробки сигналів ЕКГ з подальшим збереженням на флеш-накопичувачі.

**Достовірність наукових результатів** підтверджена теоретичними дослідження та математичними розрахунками, даними отриманими при комп'ютерному моделюванні, При розробці запропоновано:

- перехід від 8-розрядного АЦП до вищої розрядності (24);
- перехід від магнітної стрічки і енергозалежної пам'яті до енергонезалежної;
- перехід на більш швидкісний зв'язок по інтерфейсу USB.

Робота апробована на науково-практичній Всеукраїнській конференції on-line “Проблеми енергоефективності та автоматизації в промисловості та сільському господарстві”, яка відбулася 13-14 листопада 2024 року у ЦНТУ м. Кропивницький.

Робимо висновок, що виходячи з вищеперерахованого, дослідження та програмна реалізація системи збору інформації на базі флеш-накопичувачів, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		6

# 1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

## 1.1 Призначення системи

Електрокардіограма (ЕКГ) — це запис загального електричного потенціалу, який виникає при порушенні багатьох клітин серцевого м'яза.

ЕКГ записують електрокардіографом. Щоб уникнути технічних помилок і перешкод під час запису ЕКГ, слід звернути увагу на правильне використання електродів і їх контакт зі шкірою, заземлення приладу, амплітуду налаштування мілівольт та інші фактори, викликає спотворення кривих.

Електроди для реєстрації ЕКГ розміщуються в різних частинах тіла. Система розміщення електродів називається відведенням електрокардіографа.

У зв'язку з цим необхідним є створення апаратно-програмних засобів автоматизації процесу діагностики серця (електрокардіографія (ЕКГ), аналіз та висновок. Важливим є визначення оптимальних структурних параметрів електрокардіографічного сигналу (ЕКС), які забезпечують максимально достовірної діагностики, а також розробка алгоритмів виділення та обробки характерних структур реєстрованого сигналу [1].

При візуальному аналізі ЕКГ невеликі зміни на ЕКГ, які можуть містити інформацію про важливі нюанси серцевої діяльності, часто залишаються без уваги або ігноруються. Ці невидимі оку відхилення від норми можуть перерости в патологію, тому вплив невеликих змін на ЕКГ, які не діагностуються звичайними кардіографами, на точність діагностики та прогнозування стану пацієнта дуже чітко видно. було досліджено.

Для автоматизації кардіодіагностичного процесу необхідно створити велику базу даних типів електрокардіографічних сигналів ЕКС в різний час. захворюваннях серця та їх обробка та класифікація. У неї повинно бути внесене як можна більше (але, не занадто – для збільшення швидкодії) кількість видів сигнали з точними висновками кваліфікованих експертів. За параметрами вхідних

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		7

сигналів робиться попередній висновок на основі аналізу та порівняння ЕКГ. Він створюється з використанням ЕКГ окремих пацієнтів, записаних на звичайних кардіографах, а також інформації, представленої у відповідних навчальних посібниках і довідниках. Також важливо забезпечити якісне відображення перевіреного сигналу на екрані, його кодування та збереження у відповідному запам'ятовуючому пристрої.

В даний час вже існує багато різних типів електрокардіографів, які відрізняються один від одного контрольованими параметрами, робочими характеристиками, методами обробки та представлення даних. Проведені дослідження включали визначення їх характеристик, оцінку переваг і недоліків певних застосувань і вдосконалення параметрів, які визначають точність діагностики.

## 1.2 Область застосування

Областю використання даного приладу є одна із основних галузей медицини – кардіологія. Кардіологія (греч. kardia + logos навчання) - розділ клінічної медицини, що вивчає функцію і морфологію серцево-судинної системи в умовах норми і патології, етиологію, патогенез, клінічні прояви хвороб системи кровообігу і який займається розробкою методів їхньої діагностики, лікування і профілактики. Виділяють теоретичну кардіологію і клінічну кардіологію. Перша вивчає теоретичні основи кардіології: закономірності регулювання діяльності серцево-судинної системи, природу автоматичної діяльності серця, біофізику і біохімію серцевої діяльності. Найбільший розвиток у теоретичній кардіології одержали експериментальна кардіологія (моделювання патологічних станів у тварин) і молекулярна кардіологія (досліджує структуру клітки міокарда, роль клітинних мембран, мітохондрій, міофібрилл у функції міокарда, механізми утворення і переносу енергії в серце й ін.). Клінічна кардіологію у широкому розумінні включає і проблеми кардіохірургії, епідеміології серцево-судинних захворювань, вивчає загальні закономірності мікроциркуляції, закономірності

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		8

адаптації і компенсації в системі кровообігу.

Виходячи з наведеного, приходимо до висновку, що перевагами нової системи та програмного забезпечення є доступність, легкість у використанні, наявність інформаційної бази з правильної експлуатації, низька ринкова вартість, простота збірки. Дане програмне забезпечення дозволяє зручно й ефективно обробляти та зберігати результати проведених вимірювань.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи збору інформації на базі FLESH накопичувачів, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

КБПЗ\_2024

					VKPM-123.24.0019.00.00.ПЗ	Арк.
						9
Вим.	Арк.	№ докум.	Підпис	Дата		

## 2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ

### 2.1 Огляд існуючих систем, технологій, архітектур, програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти

Джерелом збудження підсилювача електричного сигналу (ПЕС) є біологічний об'єкт – людина, яку можна представити відповідним електричним генератором, згідно з формулою. А як відомо, характеристики будь-якого електрогенератора визначаються характером зміни ЕРС і внутрішнього опору в часі.

Електрокардіограма - це частина ЕРС серця, яка вимірюється електродами, встановленими певним чином на поверхні тіла. Закон зміни часу ЕКС можна вважати квазіперіодичним з періодом кардіокомплексу 0,1-3 с, відповідна форма кардіокомплексу наближена до трикутника з амплітудою 0-5 мс. Частотний діапазон, який сприймає серцевий комплекс, охоплює діапазон від 0,05 до 800 Гц.

Імпеданс між електродами, що включає опори переходів шкіра-електрод, відповідає внутрішньому опору джерела стимуляції і змінюється в значних межах. У технічних розрахунках зазвичай береться діапазон 5-100 КОм.

При створенні ЕКС, крім перерахованих параметрів, слід враховувати кілька важливих властивостей джерела збудження.

1. Нестабільність внутрішнього опору зумовлена зміною імпедансу спаю шкіра-електрод. При цьому слід враховувати великі значення опору між електродами і їх дисбаланс в системі розводки ЕКС.

2. Формування поляризаційних напруг на переходах шкіра-електрод, які генерують напруги до  $\pm 300$  мВ на вхідних клемах. Така напруга може викликати насичення підсилювача.

3. Повільний рух напруги зміщення та її різкі зміни при переміщенні електродів за рахунок рухів пацієнта. Скачки напруги зміщення створюють

					ВКРМ-123.24.0019.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		10

бар'єри, які важко подолати.

4. Наявність блукаючих напруг, що падають на синфазні та протифазні вхідні клеми. Бар'єри можуть мати біологічне або фізичне походження. До біологічних бар'єрів належать біопотенціали інших органів і м'язів, а до фізичних – напруги, що подаються на об'єкт із незахищених ділянок мережевих проводів, мережевих проводів інших пристроїв і провідних поверхонь (вторинне натягнення проводів). Сигнали перешкод від загальної напруги мережі, які надходять на об'єкт через ємнісний зв'язок, мають особливо високий рівень.

### Вимоги до параметрів

Імовірність передачі ЕКС в основному визначається параметрами посилення ЕКС, першою ланкою в ланцюзі обробки сигналу. Розглядаючи характеристики джерела збудження, особливості підключення підсилювача ЕКС до об'єкта та умови підключення підсилювача до пристрою обробки сигналів, обговоримо вимоги до основних параметрів підсилювача ЕКС та їх вплив. Спотворення ЕКС. Вхідна напруга  $U_{вх}$  повинна лежати в діапазоні не менш чим 0,03—5 мВ. Нижнє значення  $U_{вх}$  визначає граничну чутливість підсилювача, нижче якої реєстрація ЕКС утруднена.

На граничну чутливість впливає рівень внутрішніх шумів, приведених до входу підсилювача. Звичайні досяжні значення  $U_{ш} \leq 10—30$  мкВ.

Оптимальний вибір смуги пропускання ( $\Delta f$ ) має важливе значення. Найбільш інформативна частина ЕКС займає смугу частот  $\Delta f = 0,05—120$  Гц, але в практичній ЕКС-діагностиці використовують підсилювачі з  $\Delta f = 0,05—60$  Гц. У кардіомоніторах (КМ) у залежності від призначення тракту посилення ЕКС нормуються три значення  $\Delta f$ :

- $\Delta f$  — для лінійного виходу підсилювача ЕКС, призначеного для підключення реєстратора ЕКС;
- $\Delta f_з$  — для зображення ЕКГ на екрані КМ;
- $\Delta f_м$  — для моніторингу при великому рівні перешкод.

Типові значення параметрів АЧХ:  $\Delta f = 0,05—120$  Гц при  $\delta_f = \pm 30\%$  ( $\Delta f_з < \Delta f$

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		11

звичайно через технічні обмеження);  $\Delta f_M = 0,5—25$  Гц при  $\delta_f = \pm 30\%$ ;  $K_f \leq 6$  дБ/октаву.

Перешкодостійкість КМ стосовно синфазних сигналів визначається коефіцієнтом ослаблення синфазних сигналів  $K_{ОСС} = K_D / K_C$ , де  $K_D$  і  $K_C$  — коефіцієнти підсилення диференціального і синфазного сигналів.

Таким чином,  $K_{ОСС}$  показує здатність підсилювача розрізняти малий диференціальний (різницевий, протифазний) сигнал на тлі великого синфазного. Легко досягне значення  $K_{ОСС}$  лежить у діапазоні 70—80 дБ. Подальше збільшення  $K_{ОСС}$  до 90—120 дБ вимагає спеціальних методів і ускладнює конструкцію підсилювача ЕКС.

Повний вхідний опір  $Z_{ВХ}$  повинен бути не менш 2,5—10 МОм. При таких значеннях  $Z_{ВХ}$  можна зневажити втратами в передачі напруги ЕКС і допустити розбаланс опорів шкіра-електрод до 5—10 Ком. Напруга зсуву на вхідних затисках  $U_{сЕКС}$  не повинна зменшувати значення  $Z_{ВХ}$  і  $K_{ОСС}$ . Щоб не збільшувати напругу зсуву, необхідно обмежити постійний струм у ланцюзі пацієнта, обумовлений по вхідному струмі спокою, значенням 0,1 мкА.

### **Перетворювачі ЕКС для цифрових пристроїв. Аналого-цифрові перетворювачі**

Пристрій керування і передачі може взаємодіяти з аналоговим сигналом через АЦП, задача якого складається в перетворенні вхідної напруги в пропорційне йому число. Методи аналого-цифрового перетворення більш різноманітні, чим цифро-аналогового. Пояснюється це тим, що АЦП можна здійснити, використовуючи цілий ряд систем (рівнобіжний, із двотактним інтегруванням, послідовного наближення і т.д.).

Розглянемо принцип дії АЦ - перетворювача послідовного наближення, який найбільше часто використовується у медичних приладах, що пояснюється простотою пристрою, а також високою швидкодією і постійним часом перетворення, що не залежить від амплітуди аналогового сигналу.

Аналоговий вхідний сигнал апроксимується двійковим кодом числа з наступною перевіркою кожного біта в цьому коді доти, поки не буде досягнуте

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		12

найкраще наближення. Значення аналогового сигналу в двійковому коді зберігається в регістрі послідовного наближення (РгПП). Порозрядно РгПП з'єднаний із вхідним буферним пристроєм, що забезпечує цифровий вихід АЦП із необхідним рівнем вихідного сигналу. Уся робота АЦП тактується тактовим генератором. Після  $N$  тактів порівняння  $I_{\text{вх}}$  і  $I_{\text{ЦАП}}$  на вході ЦАП виходить  $N$ -розрядний двійковий код, що є еквівалентом аналогового сигналу. Перетворення відбувається за  $N$  тактів, тому швидкість формування  $N$ -розрядного слова завжди однакова. Установка РгПП у вихідний стан і запуск його в режим перетворення виробляється по зовнішньому логічному сигналі. По закінченні перетворення АЦП виробляє сигнал "Готовність даних".

### Цифрова фільтрація електрокардіосигналу

Цифрова попередня фільтрація ЕКС передуює алгоритмам, які виконують аналіз сигналу, і служить для виконання перетворень сигналу, які покращують умови роботи та підвищують ефективність цих алгоритмів. У найзагальнішому вигляді можна виділити три етапи фільтрації, які вирішують спеціальні завдання попередньої обробки ЕКС: низькочастотну фільтрацію, високочастотну фільтрацію та мережеве керівництво.

Передбачається, що на вхід алгоритму надходить суміш корисного сигналу з додатковими перешкодами. Основна частина продуктивності ЕКС, яка реєструється за допомогою стандартної апаратури ЕКГ, зосереджена в діапазоні частот не вище 50 Гц. Загалом, не можна робити жодних особливих припущень щодо спектру перешкод, за винятком того, що він обмежений характеристиками аналогового запису ЕКС і трактом посилення, який зазвичай має смугу пропускання від 0,1 до 100 Гц.

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		13

Перш за все, це зручніше усунути мережеве керівництво, яке можна відносно легко послабити за допомогою фільтра низьких частот. Крім того, високочастотні перешкоди пригнічуються за допомогою фільтрів низьких частот. Цю процедуру також можна інтерпретувати як обмеження спектру сигналу зверху, що в принципі дозволяє зменшити частоту зчитування порівняно з оригіналом на наступних етапах обробки через розрідження зчитування. . На останньому етапі попередньої обробки виконується високочастотна фільтрація з використанням високої частоти, що дозволяє практично повністю видалити постійну складову і зсув ізолінії за рахунок руху пацієнта і значно зменшити амплітуду Т-болу.

Сигнал, отриманий на виході цього ланцюжка фільтрів, є сумішшю корисного сигналу, в якому збережені основні частотні складові, характерні для комплексів КРС, і тієї частини перешкоди, спектр якої лежить в смузі пропускання результату. частотна характеристика використовуваних фільтрів. Подальше усунення перешкод методами цифрової фільтрації неможливе, оскільки це призведе до придушення самого сигналу. виходячи з наведеної послідовності процедур цифрової фільтрації ЕКС, розглянемо цифрові методи, за допомогою яких можна реалізувати кожен із етапів попередньої фільтрації.

### **Фільтри придушення мережного наведення**

Можна виділити три основних типи фільтрів, що знаходять застосування для придушення мережного наведення:

- режекторні неадаптивні фільтри;
- фільтри нижніх частот чи частотно-смугові фільтри, частотні характеристики яких мають нуль на частоті мережного наведення;
- адаптивні режекторні цифрові фільтри.

Фільтри першого з перерахованих типів, частотні характеристики яких мають провал на частоті мережного наведення, застосовуються для оперативної обробки ЕКС порівняно рідко, тому що є досить складними для реалізації.

Використання другого типу фільтра зазвичай спрямоване на одночасне вирішення двох або більше різних проблем фільтрації (видалення постійної

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		14

складової, придушення мережі та високочастотних перешкод). Така ідея є дуже привабливою, але разом з тим підвищення ефективності вирішення тієї чи однієї із зазначених проблем, як правило, досягається за рахунок інших. Наприклад, фільтри низьких частот з нульовою частотною характеристикою на частоті перешкод мережі, які досить легко використовувати в режимі реального часу, зазвичай мають відносно низьку частоту зрізу 20-25 Гц. Це може призвести до значного пригнічення високочастотних складових корисного сигналу, що не завжди прийнятно.

Адаптивні мережеві акомодційні фільтри характеризуються тим, що вони можуть адаптуватися до амплітуди і фази акомодції під час роботи і таким чином повністю її компенсувати. Такі фільтри, на відміну від перших двох згаданих типів цифрових фільтрів, мало впливають на сам корисний сигнал, особливо на його компоненти, спектр яких близький до частоти мережі. Крім того, адаптивні цифрові фільтри можуть поєднувати відносно простоту реалізації з високою якістю. Їх суть полягає в тому, що стабільна фільтрація можлива лише в тих випадках, коли амплітуда і фаза лінії не зазнають різких змін. Проте в реальних умовах аналізу роботи ЕКС параметри наведення зазвичай змінюються відносно повільно. Тому найбільш підходящою є адаптивна фільтрація.

### **Стискування електрокардіосигналу**

Представлення ЕКС з регулярним шаблоном запитів, що є результатом його дискретизації, часто є надлишковим. Зменшити надмірність можна за допомогою методів стиснення даних, суть яких полягає в зменшенні обсягу вихідної інформації за рахунок вибору меншої кількості істотних координат. Ці координати можна або отримати за допомогою деякого дискретного перетворення сигналу, або безпосередньо розв'язати з вихідної моделі зчитування. Найчастіше стиснення даних супроводжується деякою втратою інформації, через що вихідний сигнал не може бути точно відновлений.

Можливість досягти ефективної компресії ЕКС зумовлена тим, що високочастотні компоненти сигналу присутні у відносно коротких сегментах

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		15

серцевого циклу. Частота дискретизації розраховується в допустимих похибках дискретного представлення цих фрагментів ЕКС, тому опис низькочастотних частин сигналу при регулярній дискретизації відліків виявляється зайвим. Для усунення цієї надмірності були запропоновані різні методи стиснення, які пов'язані з вирішенням багатьох проблем зберігання, передачі та обробки ЕКС.

У цифрових системах передачі даних зменшення обсягу переданих даних зменшує вимоги до пропускної здатності каналу зв'язку, що особливо важливо для телефонних ліній зв'язку.

Для оцінки ефективності подання стисненого сигналу зазвичай використовують два показники: ступінь стиснення, що визначається відношенням числа початкових відліків сигналу до числа прийнятих координат, і похибка відновлення сигналу. Друга найбільш часто використовувана абсолютна або середньоквадратична помилка.

Підхід до вибору методу компресії та оцінки його ефективності повинен визначатися конкретною метою його застосування. У задачах зберігання і передачі даних зазвичай вказується прийнятний рівень спотворень відновленого сигналу, а вибір конкретного методу здійснюється виходячи з умов отримання найкращого значення стиснення з відомою складністю або прийнятним. реалізація алгоритму кодування та декодування сигналів.

Серед існуючих методів стиснення даних можна виділити групу методів, заснованих на розкладанні сигналу за ортогональними функціями. Використання розкладання Карунена-Льова, рядів Фур'є та перетворення Хаара для цілей стиснення дозволяє досягти високих ступенів стиснення, але потребує великої кількості обчислень. Крім того, існує також проблема попереднього вибору серцевого циклу, що ускладнює застосування цих методів у системах реального часу. Така компресія використовується для архівації та автоматизованої передачі ЕКС на відстань, коли немає твердих вимог до складності алгоритмів обробки і

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		16

швидкості обчислень.

Широко використовуються методи компресії, засновані на амплітудно-часових перетвореннях сигналу. Найпростішим є метод диференціального кодування, який зменшує надлишковість регулярного шаблону читання за рахунок зменшення обсягу кожної координати. Важливо відзначити, що цей метод забезпечує абсолютно точну реконструкцію сигналу вибірки.

Досить поширені методи стиснення сигналу, які використовують апроксимацію сигналу в різних сегментах часу відповідно до різних функцій. Апроксимаційними функціями можна вважати алгебраїчні поліноми різного ступеня, спеціальні функції, але більшість алгоритмів передбачає використання апроксимаційних функцій низького ступеня (ступенева або лінійна апроксимація). Головним чином це пов'язано з їх відносною простотою та високою швидкістю, яка є важливою для завдань передачі та обробки ЕКС у реальному часі.

Метод кодування ЕКС, який має нерегулярні підрахунки, відомий для опису сигналів спеціальних функцій. Тут розглядається задача апроксимації для визначення оптимального набору фільтрів зворотного зв'язку шляхом вибору лінійно незалежних фільтрів, які визначають кількість значущих сигналів. Завдяки цьому методу кодування можна досягти компресії порядку 15-20, залежно від складності вихідних ЕКГ. Для стиснення ЕКС успішно використовується кубічна лінійна апроксимація сигналу. Розроблений метод побудови кубічного сплайна для згладжування з виділенням адаптивних ступенів мережі вузлів зменшує обсяг даних у 3-14 разів. Зазначені методи стиснення сигналу за допомогою спеціальних функцій є перспективними для обробки ЕКС в поточному стані, але наразі вважаються складними для реалізації через великий обсяг обчислень.

### **Огляд і аналіз існуючих ресстраторів, аналогові кардіографи**

Звичайні аналогові електрокардіографи представляють собою електронно-механічні системи. Напруга, наведена на електроди в результаті серцевої

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		17

діяльності, підсилюється електронним підсилювачем і реєструється на паперовій стрічці за допомогою механічного самописця.

Структурна схема звичайного аналогового кардіографа представлена на рисунку 2.1.

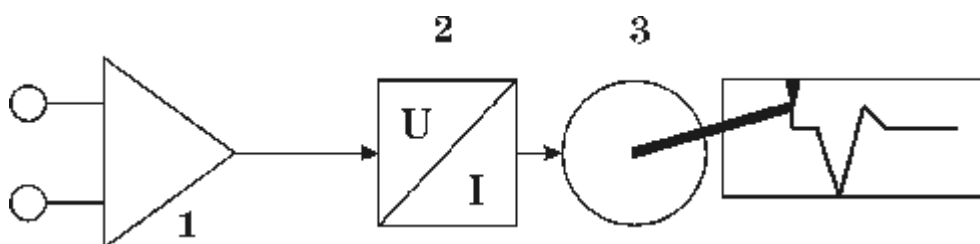


Рисунок 2.1 – Структурна схема аналогового електрокардіографа:  
1 – підсилювач; 2 – перетворювач напруга-стум; 3 – самописець.

З розвитком сучасної техніки та мікроелектроніки з'явилась база для розробки більш економічних за габаритними розмірами та по енергоспоживанню пристроїв. Тенденція розвитку мікроелектроніки обумовила новий етап в сучасній кардіології – розробка електронних електрокардіографів.

В кардіологічній практиці є багато методик дослідження та виявлення захворювань серця за допомогою електрокардіограми. Для діагностики, наприклад, деяких порушень ритму серця потрібний тривалий запис потенціалів серця.

### Холтеровські монітори

До числа знаменних подій минулого століття в області кардіології відноситься розробка Норманом Холтером системи добового моніторингу електрокардіограми. Добовий моніторинг ЕКГ за Холтером став самостійним напрямком у клінічній електрокардіології. Суть методики полягає в тривалій реєстрації ЕКГ в умовах вільної активності пацієнта, з наступним аналізом запису. Принципова схема методу не змінилася і донині. І по нині вона складається з реєстратора - пристрою, на якому здійснюється тривалий запис ЕКГ і дешифратора, що проводить аналіз отриманого запису. Оригінальний реєстратор

Нормана Холтера являв собою пристрій 19,5x9,8x4,6 см і вагою 1кг і проводив запис в одному відведенні. Можливість носити його в дамській сумочці тримати на сусідньому столику під час обіду в ресторані, представлялося, для свого часу, оптимальною по портативності системою. За 40 років існування методу технічне оснащення його разуче змінилося.

В таблиці 2.1 представлені деякі характеристики сучасних існуючих реєстраторів типа холтеровського монітора.

Таблиця 2.1 – Характеристики сучасних реєстраторів типа холтеровського монітора

Назва фірми	Країна	1.1.1.1.Модель	Розмір	АЦП біт	Частотний діапазон, Гц	Флеш-пам'ять, Мб	Маса, г	Довжина запису, год
Rozinn	США	Rozinn RZ152PM	80*150*30	нв	0.05-70	20/80	285	24/48, до 72
OXFORD	Англія	OXFORD Medilog FD4	нс	10	0,05-100	Нв	300	24/48
Schiller	Швейцарія	Microvit MT-100	110*70*90	10	0,05 40	8	140	24
MEDITECH	Угорщина	EC-GO	105*60*22	10	нв	Нв	130 (с бат.)	24
Davis Medical Electronics Inc.	США	DigiTrakPlus	85,4*53*20	10/12	0.05-60	Нв	90	24/48
Scan Tech Medical, LLC	США	IQmark Digital Holter Recorder	113*70*26	8	0.05 - 100	48-96	113	24
Національний аерокосміч. ун-т "ХАІ"	Україна	CardioSens	120*75*25	12	0,05-100	32-128	250	24/72
CUSTO MED	Німеччина	CUSTO TERA	98*65*22	нв	0,02 - 100	Нв	120 б/бат	24

Одним з основних показань до проведення холтеровського моніторингу є скарги хворих приблизно серцевого походження. До них відносяться: серцебиття, біль у грудях, запаморочення, головний біль, задишка, а так само раптова втома,

аритмії, порушення ритму, перенесений інфаркт міокарда і т.д. Особливістю цих станів є їхня непередбачена поява, рідкість виникнення і тому їх важко зафіксувати. Для цього пацієнта необхідно було б безупинно спостерігати аж до виникнення самої події, але це нереально, якщо врахувати, що деякі епізоди мають періодичність раз на місяць (рік). Однак, тільки в 0-12% дітей [9] і 6-25% дорослих з кардіогенними скаргами, холтеровський моніторинг здатен виявити порушення ритму. Основна причина низької діагностичної цінності методу - дуже короткий період спостереження.

Проблема знижується шляхом використання контролерів - реєстраторів подій (Event Recorder). Це прилади невеликих розмірів здатні робити запис ЕКГ у 1-3 відведеннях. Пристрій знаходиться в пацієнта, у момент виникнення події хворий прикладає електроди до тіла (можуть бути грудні, для кінцівок) і натисканням кнопки здійснює запис ЕКГ. Записана електрокардіограма передається лікуючому лікарю по телефону, модему, через локальну комп'ютерну мережу.

Реєстратори подій більш ефективні в діагностиці аритмій виникаючих не частіше чим 1 раз на місяць, чим холтеровський моніторинг. Дозволяють у 57% випадків виключити порушення ритму серця при скаргах пацієнтів на серцебиття дають більше діагностичної інформації і більш низьку собівартість, чим 48 годинне холтеровський моніторинг.

### **Реєстратори подій, Event - рекодер**

Основний принцип його використання вже описаний вище. Для подій тривалістю в кілька секунд чи хвилин, цей реєстратор прикладається до грудей чи до кінцівок пацієнта і записується епізод. Як правило, у них немає окремих електродів, вони убудовані в корпус самого приладу. Це дуже зручно, при виникненні події можна швидко прикласти прилад до грудей не витрачаючи час на наклеювання електродів (і гроші на їхню покупку!) чи взятися за них пальцями. У більшості комерційних моделей існує всього одна кнопка, вона служить як для запису ЕКГ, так і передачі її.

Технічне виконання event-рекодерів таке ж, як і реєстраторів холтеровських

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		20

систем. Відрізняються вони видом і обсягом пам'яті, тому що реєстратори подій не здійснюють безупинний запис ЕКГ, обсяг пам'яті в них значно менше. Дозволяють записувати фрагменти ЕКГ тривалістю від 30 до 270 секунд. Використовується як твердотільна енергозалежна пам'ять, так і флеш-пам'ять. Число записів клінічних епізодів залежить від того, на скільки відрізків розділений весь обсяг пам'яті. При заповненні всього обсягу, записані фрагменти передаються на комп'ютер чи відсилаються по телефонній лінії, де буде проаналізована лікарем і роздрукована. Деякі прилади можуть бути підключені до цифрового електрокардіографа.

## **2.2 Обґрунтування вибору засобів для побудови системи та мови програмування**

З постановки задачі до магістерської роботи дослідження та програмна реалізація системи збору інформації на базі FLESH накопичувачів системи добового моніторингу ЕКГ (ДМ ЕКГ) відомо, що джерелом вхідних даних є електричні імпульси серця. Електричні імпульси реєструються електродами, які прикріплюються до тіла людини в визначених місцях. Опір ділянки тіла людини коливається в інтервалі 5–100кОм. Аналогові сигнали, що знімаються електродами, після заданого рівня підсилення повинні надходити на вхід аналого-цифрового підсилювача.

Цифрова обробка (ЦОС) сигналів є базовим принципом для розробки функціональної структури сучасних ДМ ЕКГ. Якість ЦОС значною мірою визначається якістю виділення корисного електрокардіографічного сигналу на тлі зовнішніх перешкод і типом аналого-цифрового перетворення. При достатній розрядності і швидкості перетворення АЦП, можна відмовитися від класичної схеми побудови аналогової частини ДМ ЕКГ, що представлена на рисунку 2.1 і перейти до наступної схеми до схеми, що представлена на рисунку 2.2.

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		21

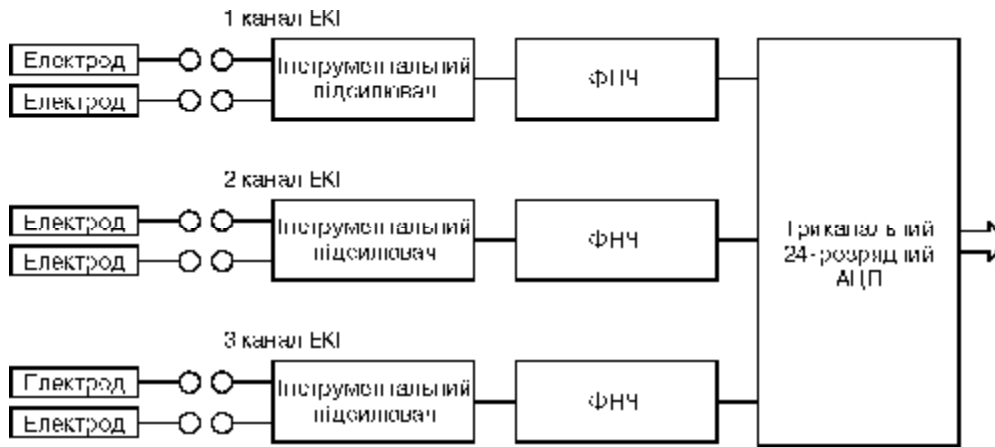


Рисунок 2.2 – Структурна схема аналогової частини ДМ ЕКГ на АЦП високої розрядності.

Необхідно використовувати інструментальні чи диференціальні підсилювачі з високим коефіцієнтом придушення синфазної перешкоди. Але завдяки високій розрядності і широкому входному діапазону схеми, ФВЧ і додаткове посилення сигналу можна не використовувати, проводячи дискретизацію сигналу ЕКГ із постійної складової. Постійна складова ЕКГ на вході приладу компенсується за рахунок додаткових біт. У результаті, різко скорочується число електронних компонентів для побудови такої схеми. Зникає необхідність застосування прецизійних великогабаритних конденсаторів з малими струмами витоку в аналоговому ФВЧ, що значно скорочує габарити. ФВЧ реалізується цифровим способом на комп'ютері з необхідною частотою зрізу, у залежності від задачі.

#### Аналогова частина приладу

Канал1, Канал2, Канал3 – канали вимірювання напруги, яка безпосередньо знімається з електродів Д. Перед тим, як надійти на аналого-цифровий підсилювач сигнал повинен бути підсилений або буферований. Це робиться за допомогою одно каскадного диференціального підсилювача на двох операційних підсилювачах.

АЦП перетворює підсилений аналоговий сигнал у цифрову форму. Коли дані на АЦП готові, сигнал DRDY повідомляє мікроконтролеру, що дані готові до

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		22

зчитування. Для зчитування даних в пристрої використовується синхронний периферійний інтерфейс SPI. Також SPI використовується для завантаження внутрішніх регістрів АЦП, тобто для конфігурації АЦП в потрібний режим роботи. Сигнал CS проводить виборку АЦП, яка ініціює використання АЦП.

### **Цифрова частина приладу**

Порт А мікроконтролера виконує функції зв'язку з пристроєм USB. Двонаправлений Порт А служить для вводу/виводу даних/адреси.

Порт В мікроконтролера виконує функції керування пристроєм та зв'язку мікро контролера по інтерфейсу USB.

Сигнали START/STOP, OUT TO USB, ERASE FLASH – це по суті керуючі сигнали, які визначають що пристрій в даний момент потрібен робити. Тобто це кнопки управління пристроєм. Сигнал START/STOP ініціює початок запису інформації або його зупинку. Сигнал OUT TO USB починає передачу даних по протоколу USB. Сигнал ERASE FLASH ініціює очищення flash-пам'яті.

Порт С мікроконтролера виконує функції керування Flash-пам'яттю та пристроєм USB.

RD – сигнал дозволу читання з пристроєм USB

WR – сигнал дозволу запису в пристроєм USB

ALE2 – сигнал замикача адресу регістра в пристрої USB

RE – сигнал дозволу читання з flash-пам'яті

ALE - сигнал замикача адресу flash-пам'яті

CLE - сигнал замикача команди flash-пам'яті

R/B – сигнал flash-пам'яті “Зайнято/Вільно”

SE – сигнал дозволу читання або запису резервної області flash-пам'яті.

Порт D мікроконтролера виконує функції зв'язку з запам'ятовуючим пристроєм(Flash-пам'ять). Порт D служить для вводу/виводу даних/адреси.

### **Опис інтерфейсу USB**

Визначення швидкісних характеристик пристрою і самого факту включення його на шину виробляється завдяки наявних у пристрої pullup-

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		23

резисторів, підключених до сигналу D+ чи D-. Підключення до сигналу D+ сигналізує підключення повношвидкостного пристрою, до D- - низькошвидкостного.

Синхроімпульси передаються закодованими разом з інформацією. Послідовність, що самосинхронізується, реалізована за схемою NRZI with bit stuffing. NRZI - Non Return to Zero Invert to ones - метод без повернення до нуля з інвертуванням для одиниць. Цей метод являє собою модифікацію простого потенційного методу кодування, названого Non Return to Zero (NRZ), коли для представлення 1 і 0 використовуються потенціали двох рівнів. У методі NRZI також використовується два рівні потенціалу сигналу, але потенціал, використовуваний для кодування поточного біта залежить від потенціалу, що використовувався для кодування попереднього біта (так називається диференціальне кодування). Якщо поточний біт має значення 1, то поточний потенціал являє собою інверсію потенціалу попереднього біта, незалежно від його значення. Якщо ж поточний біт має значення 0, то поточний потенціал повторює попередній. З опису методу NRZI видно, що для забезпечення частих змін сигналу, а значить і для підтримки самосинхронізації приймача, потрібно виключити з кодів занадто довгі послідовності нулів. Ця дія називається bit stuffing.

Поле синхронізації, що передує кожному пакету, дозволяє приймачу синхронізуватися.

### **Протокол шини USB**

USB - цілком контрольована хостом шина. У системі USB може бути тільки один майстер (керуючий елемент) - комп'ютер-хост. Будь-яка транзакція в ній здійснюється передачею до трьох пакетів. Хост, відповідно до визначеної тимчасової діаграми генерує ознаки, що керують передачею по шині. У такому пакеті ознаки міститься інформація про тип передачі, її напрямок, адресі одержувача і номері каналу (кінцевої точки в одержувачі). Пристрій на шині визначає, що пакет призначений йому, декодує відповідні адресні полюси.

Пристрою USB тільки відповідають на запити хоста і не можуть передавати інформацію один одному.

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		24

Фактично може бути тільки один випадок, коли пристрій може ініціювати передачу без активності хоста. Після приміщення хостом пристрою в режим зниженого енергоспоживання, пристрій може сигналізувати своє пробудження. Це єдиний спосіб перервати ланцюжок, відомий хостом. Все інше на шині відбувається тільки завдяки тому, що хост робить запити пристроям, і пристрою відповідають на ці запити.

### **Пристрої і канали шини USB**

Кожен пристрій на шині USB має кілька так званих кінцевих точок – endpoints(EP). Ці пронумеровані точки є кінцями логічних каналів даних між хостом і пристроєм. Таким чином, між хостом і пристроєм реалізується багатоканальна передача даних.

Ідентифікаційний номер пристрою дається йому хостом при підключенні пристрою до шини, номери ж кінцевих точок задаються при виготовленні пристрою.

Будь-який пристрій повинний підтримувати endpoint 0, тому що цей засіб конфігурування пристрою за замовчуванням у USB (Default Control Pipe) після чи включення одержання сигналу скидання по шині. Всі інші EP і канали з'являються після конфігурування пристрою хостом. DCP може так само використовуватися специфічним клієнтської ПО хоста. У цьому випадку системне ПО хоста USB є посередником між клієнтським ПО і DCP пристрою.

Кінцеві точки мають власні характеристики, їх необхідно знати клієнтському ПО для визначення типу з'єднання:

- частота передачі
- вимоги по пропускну здатності каналу
- номер EP
- вимоги по обробці помилок
- максимальний розмір пакета, що може відправляти/приймати ця EP
- тип передачі (режим)
- напрямок передачі



Існують три PID узгодження: ACK, NAK і STALL:

- ACK – означає, що дані були прийняті без помилок;
- NAK – означає - зайнято, передайте ще раз.. Це не означає помилки,

тому що помилка – це відсутність відповіді;

- STALL – означає, що відбулося щось непередбачене (можливо як результат втрати чи зв'язку неузгодженості програмного забезпечення різних рівнів). Пристрій посилає STALL, показуючи, що не зрозуміло запит.

PID PRE передує передачі на малій швидкості (1.5 Мбит). Ми його ігноруємо.

Вибір та обґрунтування окремих вузлів, блоків, елементів

### **Вибір мікроконтролера**

В даній роботі використовується мікроконтролер фірми Atmel AT90S8515

Відмінні особливості:

- AVR® - висока продуктивність і RISC архітектура з низьким енергоспоживанням

- 118 потужних інструкцій - більшість з них виконуються за один такт
- 8 Кбайт Flash- пам'яті з підтримкою внутрісистемного програмування:
- SPI послідовний інтерфейс для завантаження програмного забезпечення

- Ресурс: 1000 циклів запису/ стирання
- 512 байт EEPROM Ресурс: 100 000 циклів запису/ стирання
- 512 байт внутрішньої SRAM
- 32 x 8 робітників регістрів загального призначення
- 32 програмовані шини I/O
- Програмований послідовний UART
- SPI послідовний інтерфейс
- Напруга  $V_{CC}$ : від 2.7V до 6.0V
- Продуктивність, аж до 8 MIPS при 8 МГц
- Один 8-ми розрядний таймер/ лічильник з окремим попереднім

дільником частоти

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		27

- Один 16-ти розрядний таймер/ лічильник з окремим попереднім дільником частоти з режимами порівняння, захоплення
- Здвоєний ШІМ
- Зовнішні і внутрішні джерела переривання
- Програмувальний таймер, що стежить, з вбудованим тактовим генератором
- Вбудований аналоговий компаратор

Пристрій виробляється з застосуванням технології енергонезалежної пам'яті з високою щільністю розміщення, розробленої в корпорації Atmel. Убудована Flash - пам'ять з підтримкою внутрісистемного програмування забезпечує можливість перепрограмування програмного коду в складі системи, за допомогою SPI послідовного інтерфейсу, чи за допомогою стандартного програматора енергонезалежної пам'яті. Завдяки сполученню удосконаленого 8-ми розрядного RISC CPU з Flash- пам'яттю з підтримкою внутрісистемного програмування на одному кристалі вийшов високопродуктивний мікроконтролер AT90S8515, що забезпечує гнучке й економічно-високоєфективне рішення для багатьох пристроїв систем керування, що вбудовуються.

### **Обґрунтування вибору пристрою USB**

Пристрій сполучив у собі необхідний USB прийомопередатчик з 3.3В регулятором, механізм послідовного інтерфейсу (SIE), FIFO кінцевих точок (EP) USB, універсальний 8-розрядний інтерфейс, генератор тактових імпульсів і інтерфейс MICROWIRE/PLUS™. Підтримуються сім програмних каналів кінцевих точок: один для примусового керування кінцевою точкою і шість для підтримки переривань, масових і рівно тимчасових кінцевих точок. Кожен канал кінцевої точки має призначений для нього FIFO, 8 байтів для керування кінцевою точкою і 64 байта для інших кінцевих точок. 8-розрядний паралельний інтерфейс підтримує мультиплексовані і немультіплексовані типи шин адреси/даних CPU. Вихідна схема програмувальних переривань дозволяє сконфігурувати пристрій для різних вимог передачі сигналів переривань.

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		28

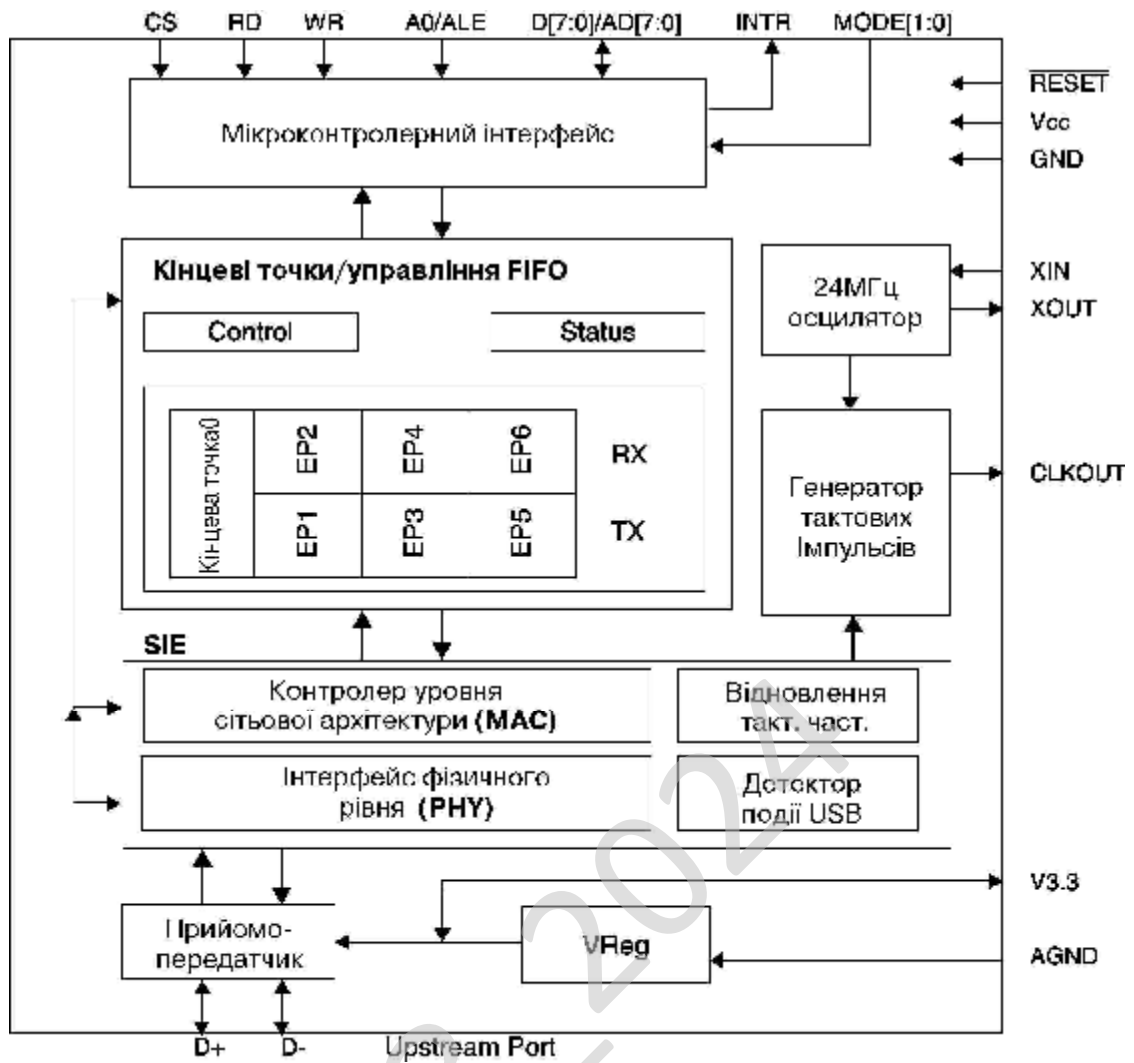


Рисунок 2.4 - Структурна схема USBN9603

### Особливості

- Швидкодіючий пристрій вузлів USB;
- Інтегрований прийомопередатчик USB;
- Пристрій підтримує 24 МГц схему генератора з внутрішньою схемою генерації на 48 МГц;
- Програмувальний генератор тактових імпульсів;
- Механізм послідовного інтерфейсу (SIE) складається з інтерфейсу фізичного рівня (PHY) і контролера рівня мережної архітектури, що здійснює мережний доступ і виявлення конфліктів (MAC), підтримує специфікацію USB 1.0 і 1.1;

- Реєстровий файл керування/стану;
- Функціональний контролер USB з родин кінцевими крапками FIFO;
- Одна двунправленна кінцева крапка керування 0 (8 байтів);
- Три кінцеві крапки передачі (64 байта кожна);
- Три кінцеві крапки прийому (64 байта кожна);
- 8-розрядний паралельний інтерфейс із двома режимами вибору;
- Немультіплексований;
- Мультіплексований (сполучимо з Intel);
- Поліпшена підтримка DMA;
- Режим автоматичного DMA (ADMA) для повної CPU-незалежної передачі великих чи обсягів ISO пакетів;
- Контролер DMA, разом з логікою ADMA, може передавати великі блоки в 64-байтних пакетах через USB;
- Автоматичне переключення/перевірка пропорційно-інтегрально-диференціального регулювання даних (PID) і відновлення пакета при відсутності підтвердження прийому (NAK) (максимум 256x64 байтів даних = 16Кб);
- Інтерфейс MICROWIRE/PLUS.

### **Вибір flash-пам'яті**

K9F4G08U0M – 4096 Мбітна (1 107 296 256) NAND flash-пам'ять з блоками по 128 МБіт. Комірка NAND забезпечує саме рентабельне рішення на ринку твердо тільних запам'ятовуючих пристроїв великої ємності. Операція програми програмує 2112- байтну сторінку за 300s, операція стирання може бути

виконана за 2ms на 128К-байтовому блоці. Дані в сторінці можуть бути прочитані за 50ns часу циклу на байт. Контакти вводу-виводу служать портами для адреси і вводу/виводу даних так само як вивід команди. Внутрішній контролер запису автоматизує всю програму і функції стирання, включаючи повторення імпульсу, де потрібно, і забезпечує внутрішню перевірку і позначення кінця даних. Навіть системи з інтенсивним використанням програмування/стирання можуть використовувати у своїх інтересах K9F4G08U0M, надійність 1000000 циклів

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк. 30
Вим.	Арк.	№ докум.	Підпис	Дата		

програмування/стирання, забезпечуючи будь-який код з виправленням помилок чи алгоритм роботи в реальному масштабі часу. Ці алгоритми були здійснені в багатьох пристроях збереження інформації великої ємності і також резерв, 16 байтів на сторінці, об'єднаної з іншими 512 байтами, можуть використовуватися системним кодом з виправленням помилок . K9F4G08U0M - оптимальне рішення до пристроях збереження великих об'ємів інформації, де використовується твердотільний тип пам'яті, що вимагає енергонезалежного збереження інформації.

Пристрій включає особливість авточитання після включення живлення, що допускає послідовний доступ до даних 1-ої сторінки без команди і введення адреси після включення живлення.

На додаток до розширеної архітектури й інтерфейсу, пристрій включає особливість зворотного копіювання програми від однієї сторінки в іншу сторінку без потреби транспортування даних в і з зовнішньої буферної пам'яті.

Середовищем розробки програмного забезпечення було обрано AVR Studio, яке є інтегрованим середовищем розробки для програмування та налагодження програм мікроконтролерів AVR і AVR32 в операційних системах Windows. Після шостої версії середовище може працювати як з контролерами AVR, так і з системами з архітектурою ARM.

Пакет програм AVR Studio розробляється компанією Atmel з 2004 року. Починаючи з версії 6.0, компанія змінила назву програми на Atmel Studio і додала можливість програмувати системи на базі архітектури ARM. Раніше у Atmel також був фірмовий асемблер (wavrasm.exe) для операційної системи Windows, який поєднував асемблер і редактор, але після випуску AVR Studio від його подальшої розробки відмовилися.

Atmel Studio містить в собі такі інструменти, як вбудований C/C++-компілятор, симулятор мікропроцесорної системи для відлагодження програм, менеджер проектів, редактор коду, модуль внутрішньосхемного відлагодження, а також інтерфейс командного рядка. Крім стандартних елементів, середовище підтримує ряд інших інструментів, таких як компілятор GCC та плагін AVR RTOS

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		31

(операційної системи реального часу). Крім C/C++, середовище дозволяє програмувати також на асемблері.

В якості мов програмування, було обрано Assembler та C++.

### 2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випускню кваліфікаційну роботу за другим (магістерським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи збору інформації на базі FLESH накопичувачів.

У процесі підготовки роботи другого (магістерського) ступеня вищої освіти необхідно виконати такі завдання:

а) проаналізувати існуючі аналогові системи з метою визначення їх позитивних і негативних якостей. Результати аналізу враховуються при подальшій розробці;

б) провести вибір та обґрунтування способу побудови системи контролю за роботою технологічного обладнання в автоматизованому режимі. Розробити функціональну та структурну схеми системи;

в) необхідна розробка системного програмного забезпечення, що дозволить реалізувати поставлену технічним завданням мету. Створення структурних схем алгоритмів і програмних підпрограм;

ж) організувати інтерфейс користувача з метою створення та відображення на екрані комп'ютера повідомлень про некоректні дії користувача та нестандартні ситуації в роботі технологічного обладнання;

г) підготовка рекомендацій щодо організаційно-методичних заходів щодо забезпечення впровадження системи у виробництво та її подальшої успішної експлуатації;

г) виконати розрахунки для визначення економічної ефективності розробленої системи;

ж) розроблення заходів з охорони праці під час впровадження та

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		32

експлуатації системи, а також розроблення заходів цивільного захисту;

з) зробити висновки про обсяги виконаної роботи та отримані результати.

КБПЗ\_2024

					ВКРМ-123.24.0019.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		33

## 3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

### 3.1 Опис функціонування системи

Для керування пристроєм використовуються 3 кнопки, які підключають напругу живлення через підтягуючі резистори до PB0, PB1, PD2. При натисненні цих кнопок встановлюється високий рівень сигналу на відповідних контактах.

Мікроконтролер – основа системи. Кварцевий генератор підключений до мікроконтролера за стандартною схемою.

Для зв'язку з 8-розрядною лінією даних/адреси flash-пам'яті використовується порт D.

Порт A використовується для зв'язку з 8-розрядною лінією даних/адреси пристрою USB.

Порт C мікроконтролера виконує функції керування вводом/виводом даних/адреси flash-пам'яті та пристрою USB:

- PC0 – виробляє строб читання даних з пристрою USB;
- PC1 – виробляє строб запису даних;
- PC2 – сигнал адресного замикача;
- PC3 – строб читання даних з flash-пам'яті;
- PC4 – сигнал адресного замикача flash-пам'яті;
- PC5 – сигнал замикача команди flash-пам'яті;
- PC6 – перевірка чи занята flash-пам'ять чи готова до поточних операцій.
- PC7 – сигнал дозволу резервної області flash-пам'яті.

Підключення кварцового генератора до пристрою USB. Виводи XIN і XOUT можуть бути підключені, щоб замкнути 24 МГц генератор із кварцовою стабілізацією частоти. Поперемінно, зовнішній 24 МГц джерело тактових імпульсів може використовуватися як вхід тактових імпульсів пристрою.

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		34

Внутрішній кварцовий осцилятор використовує основну частоту кварцу 24 МГц. Типові значення компонентів схеми кварцового генератора див. у таблиці 3.1 і на рисунку 3.1. Для використання специфічного кварцового генератора, необхідно проконсультуватися у виробника для значень компонентів, що рекомендуються.

Таблиця 3.1 – Приблизні значення компонентів

Компонент	Параметри	Значення	Допустиме відхилення
Кварцовий резонатор	Резонансна частота	24 МГц	2500 ppm (максимум)
	Тип	Кристал, зрезаний під 35 градусів до осі z (AT-Cut)	
	Максимальний послідовний опір	50 Ом	
	Максимальна шунтуюча ємність	10 пФ	
	Ємність нагрздки	20 пФ	
Резистор R1		1 МОм	±5%
Резистор R2		0	NA
Конденсатор C1		15 пФ	±20%
Конденсатор C2		15 пФ	±20%

Конденсатори C1 і C2 вибираються виходячи з ємності навантаження кварцового генератора. Ємність навантаження CL "видима" кварцовим генератором складається з C1 послідовно з C2, і паралельно з паразитною ємністю схеми. Паразитна ємність виникає через корпус мікросхеми, друкованою

платою і рознімання, і може варіюватися від 0 до 8 пф. Емпіричне правило при виборі цих конденсаторів:

$$CL = (C1 * C2) / (C1 + C2) + C_{\text{паразитне}} \quad (3.1)$$

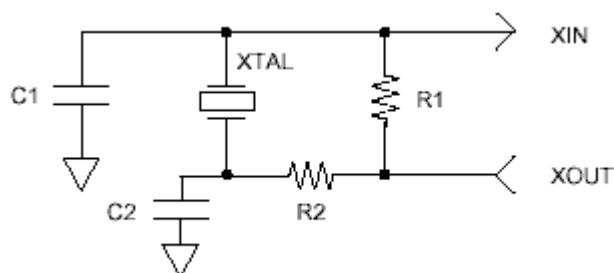


Рисунок 3.1 - Типовая схема осцилятора

Вихід прийомопередатчика пристрою USB підключений до роз'ємного з'єднання USB.

Пристрій USB працює в мультиплексорному режимі роботи. Для забезпечення мультиплексорного режиму роботи. Контакт MODE1 підключається до напруги Vcc, а контакт MODE2 до корпусу.

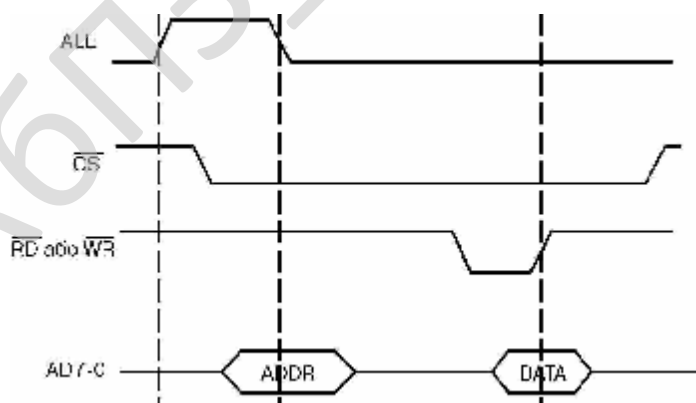


Рисунок 3.2 –Часова діаграма роботи пристрою USB в мультиплексорному режимі роботи

Мікросхема пам'яті підключена до порту D, на рисунках зображені часові діаграми циклів запису та читання flash-пам'яті.

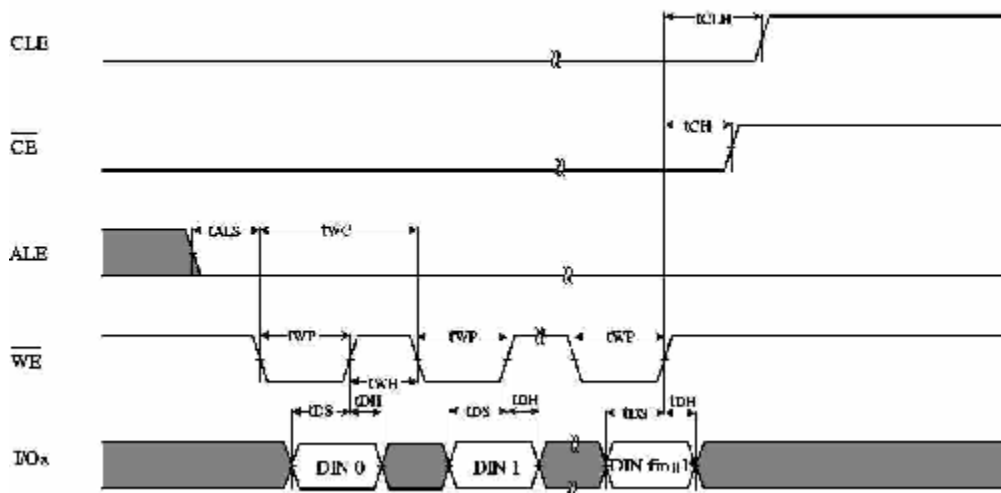


Рисунок 3.3 – Часова діаграма циклу запису flash-пам'яті

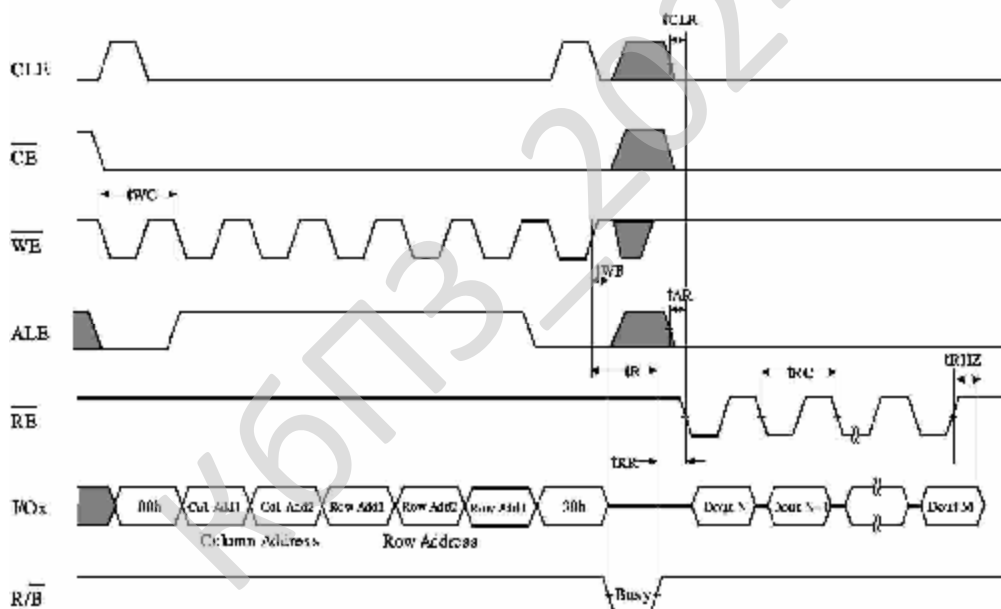


Рисунок 3.4 – Часова діаграма циклу читання flash-пам'яті

### Розрахунок часу запису

Відомо, що:

- Розрядність АЦП – 24Біт=3Байт;
- Ємкість flash-пам'яті – 1056МБіт=138412032Байт;
- Частота дискретизації – 125Гц;

- Кількість каналів – 3.
- Частота опиту АЦП

$$F_{\text{АЦП}} = K \cdot f_{\text{дискр}} \quad (3.2)$$

де  $K$  – кількість каналів;

$f_{\text{дискр}}$  – частота дискретизації.

Тому,  $F_{\text{АЦП}} = 3 \cdot 125 = 375 \text{ Гц}$

Швидкість запису розраховується за формулою

$$V_{\text{зан}} = F_{\text{АЦП}} \cdot K \cdot R \quad (3.3)$$

де  $R$  – розрядність АЦП.

Тому,  $V_{\text{зан}} = 375 \cdot 3 \cdot 3 = 3375 \text{ Байт/с}$

Час запису розраховується за формулою

$$t_{\text{зан}} = \frac{\text{flash}}{V_{\text{зан}}} \quad (3.4)$$

де *flash* – ємкість flash-пам'яті.

Тому,  $t_{\text{зан}} = \frac{553648128}{3375} \approx 164044 \text{ с} = 45,56 \text{ год.}$

Загальний час запису приладу складає 45 годин 34 хвилини.

### 3.2 Розробка структурної схеми

Можна виділити наступні основні переваги, запропонованої підходу при розробці аналого-цифрового тракту ДМ ЕКГ:

1. Немає необхідності застосування прецизійних великогабаритних конденсаторів з малими струмами витоку в аналоговому ФВЧ.
2. Аналогової сигнал не виходить за діапазон виміру АЦП при артефактах рухах чи пацієнта зовнішній перешкоді.
3. ФВЧ реалізується цифровим способом з необхідною частотою зрізу і характеристиками, у залежності від задачі.
4. Не потрібно додаткове посилення сигналу, що збільшує шуми.

5. Застосовується простий ФНЧ на R-C ланцюг для забезпечення умов теореми Котельникова.

6. Дискретизація сигналу відбувається з високою частотою безпосередньо в АЦП, без необхідності в схемах виборки-збереження і мультиплексорі.

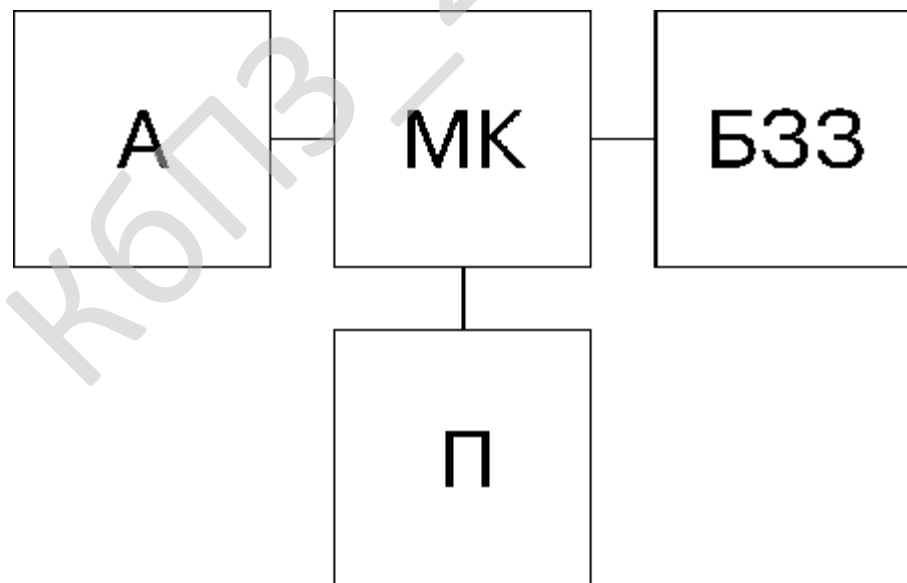
Виходячи з зазначеного, зупиняємося на запропонованій схемі аналогової частини ДМ ЕКГ.

Центральним елементом системи ДМ ЕКГ є мікроконтролер, який забезпечує зв'язок між всіма функціональними блоками.

Для зберігання не менш ніж добового обсягу інформації, що реєструється, структурна схема повинна містити енергонезалежний блок пам'яті 1Gb-32Gb.

Для зв'язку розроблюваного пристрою з комп'ютером по інтерфейсу USB (за поставленою задачею до дипломного проектування) в структурній схемі повинен бути присутній блок, який виконує функції зовнішнього інтерфейсу.

Якщо проаналізувати все, вище викладене то структурна схема пристрою буде виглядати таким чином:



Риснок 3.5 – Структурна схема приладу.

МК – мікроконтролер;

А – аналогова частина приладу;

БЗЗ – блок зовнішнього інтерфейсу USB;



На рисунку 3.7 представлена принципова електрична схема приладу ДМ ЕКТ.

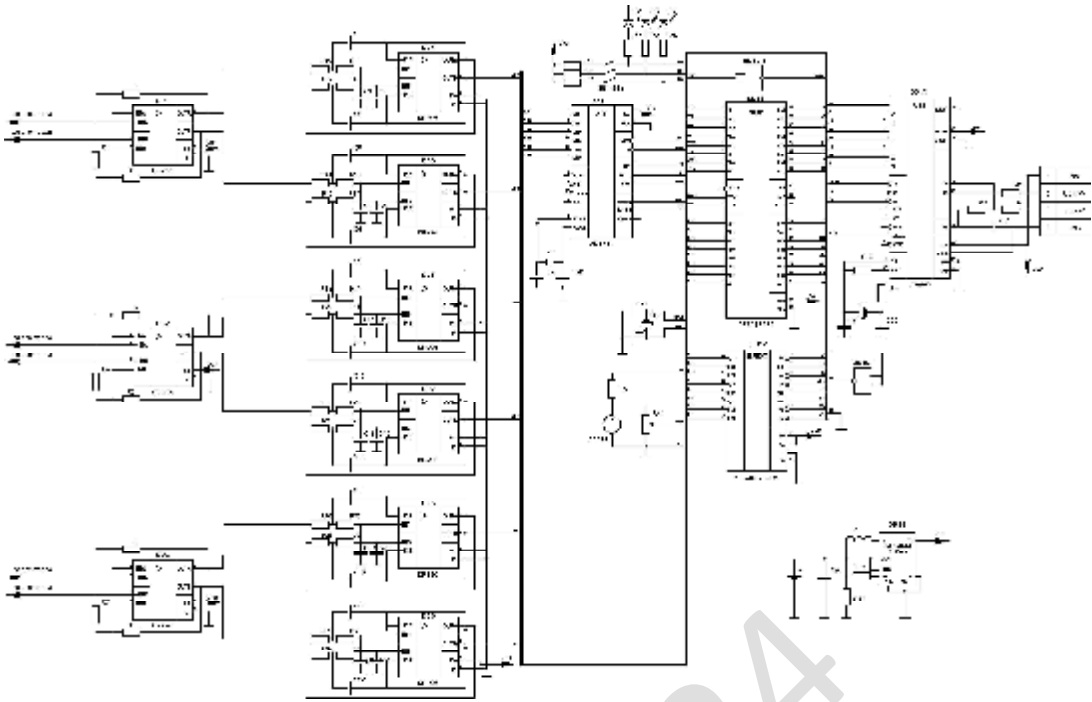


Рисунок 3.7 – Принципова електрична схема приладу ДМ ЕКТ

24-розрядний двійковий код виборки кожного кардіосигналу у послідовному коді через SPI інтерфейс поступає на входи порту В мікроконтролера SMC. З виходу мікроконтролера SMC (порт D) цей код у паралельно-послідовному 8-бітному коді передається до енергонезалежної flash-пам'яті EEPROM. Ємність накопичувача EEPROM 256-512МБ, що дозволяє без стискування зберігати добовий обсяг інформації.

При необхідності накопичена інформація може бути передана в ОЗП персонального комп'ютера для наступної обробки, аналізу і видачі результатів діагностики через контролер USB. Ця інформація через USB порт може бути також записана на портативний flash-накопичувач.

Робота пристрою управляється за допомогою трьох кнопок: START/STOP, очищення flash-пам'яті EEPROM (ERASE FLASH) та передача даних через USB порт (OUT TO USB).

Індикація в приладі виконується за допомогою трьох світло діодів, які показують стан роботи пристрою.

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		41

### 3.4 Розробка діаграми процесів

Діаграма взаємодії процесів системи, розробленої у результаті виконання магістерської роботи, наведена на рисунку 3.8 З нього видно, що процеси взаємодіють наступним чином.

Спершу завантажується процес початку роботу головної програми. Він запускає процес роботи наступних підпрограм:

- Ініціалізація мікроконтролера;

Після включення приладу до джерела живлення, всі регістри та порти мікро контролера знаходяться у вихідних значеннях. Перед початком роботи всі порти та регістри встановлюються у їх робочий стан, задається початкова адреса flash-пам'яті.

- Ініціалізація АЦП;

Після початку підпрограми зчитування даних з АЦП встановлюємо активний канал 1. Дані в регістрі даних АЦП готові до зчитування з частотою, яка раніше встановлена. Коли дані готові до зчитування АЦП сигналізує мікроконтролеру низьким рівнем сигналу DRDY, тому проводиться опитування цього контакту. При високому рівні цього сигналу, система очікує коли він досягне низького рівня. При виникненні низького рівня встановлюється лічильник байт рівний 1 та проводиться зчитування першого байту з регістру даних АЦП та запис його у flash-пам'ять. Так як результат рівний 3 байтам, інкрементуємо лічильник і зчитуємо наступний байт даних. Після зчитування третього байту та інкременту лічильника, лічильник стає більший 3. Тоді відбувається переключення каналу шляхом інкременту лічильника каналу. Процедура зчитування 3 байтового слова починається знову. Тільки оцифровані дані знімаються з каналу на одиницю більше. Коли дані зчитані з трьох каналів підпрограма закінчується і повертається в головний цикл.

- Ініціалізація роботи Flash-пам'яті;

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		42

Як повідомлялося, USB - цілком контрольована хостом шина. У системі USB може бути тільки один майстер (керуючий елемент) – комп’ютер-хост. Тому для передбачення передачі інформації на flash-накопичувач на комп’ютер потрібно два режими роботи. Режим Master – пристрій виступає, як керуючий елемент, та режим Slave – підлеглий пристрій виступає, як підлеглий. Для визначення режиму роботи, в залежності чи пристрій підключений до комп’ютера, чи до пристрою підключений flash-накопичувач головний алгоритм очікує сигналу запиту на прийом інформації IN, якщо сигнал IN не поступає за 30с, то пристрій переходить в режим Master, тобто передача проводиться на flash-накопичувач. Якщо отриманий сигнал IN, керування процесом передачі інформації передається комп’ютеру.

### **Вибір режиму**

Робота пристрою управляється за допомогою трьох кнопок: START/STOP, очищення flash-пам’яті EEPROM (ERASE FLASH) та передача даних через USB порт (OUT TO USB).

При розпізнанні натиснення кнопки “очищення” виконується підпрограма очищення flash-пам’яті. Після закінчення цієї підпрограми система знову повертається у стан очікування.

Коли натиснута кнопка “зчитування” виконується підпрограма передачі інформації, яка записана на flash-пам’ять, по протоколу USB.

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		43

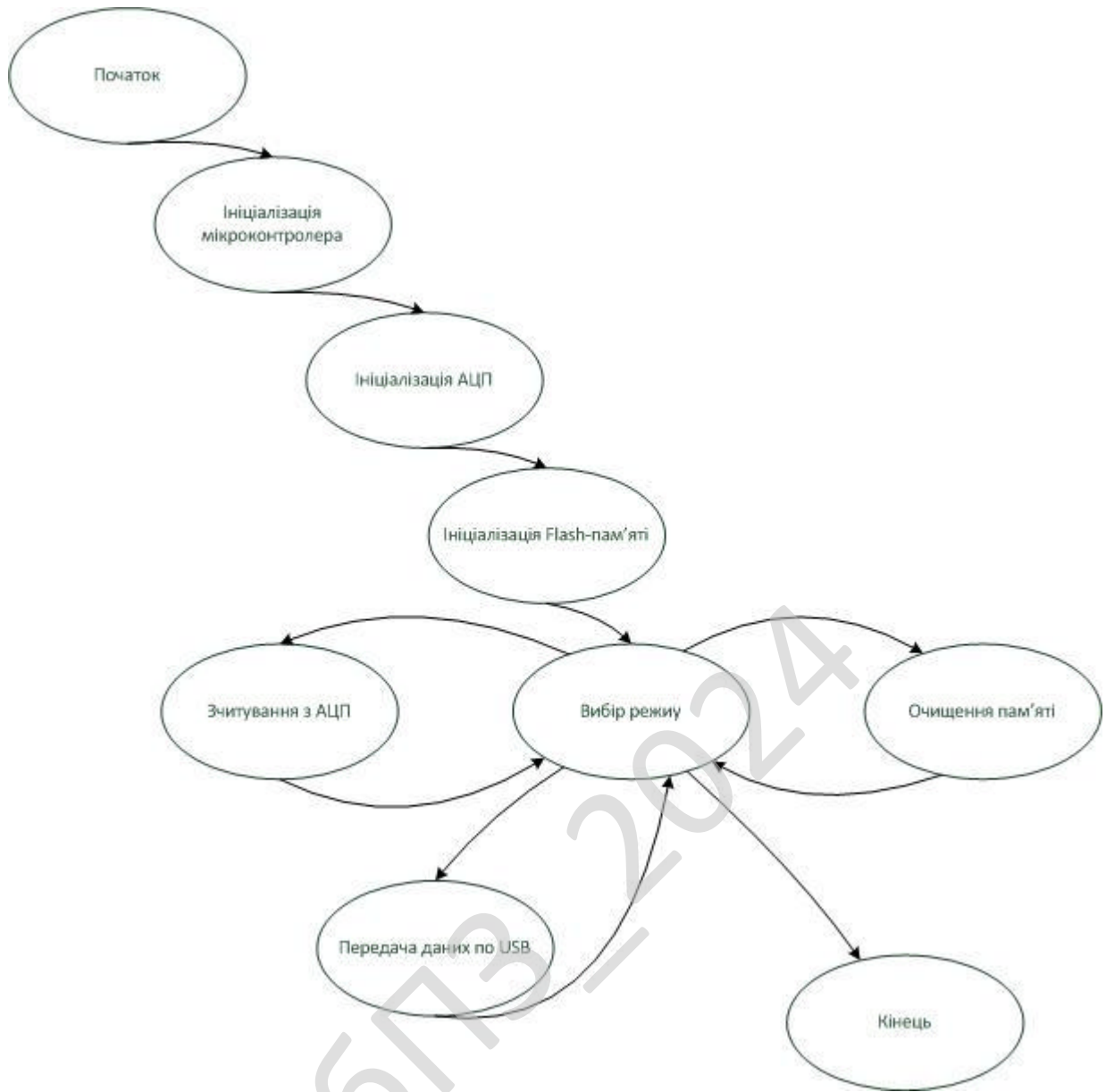


Рисунок 3.8 – Діаграма процесів системи

## 4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ

### 4.1 Блок-схеми та опис алгоритмів функціонування системи

Виходячи з розроблених функціональної та принципової електричних схем будуємо алгоритми, за якими повинен працювати прилад. З поставленої задачі до роботи маємо три кнопки управління приладом, що розроблюється.

Алгоритм головного циклу.

Після включення приладу до джерела живлення, всі регістри та порти мікроконтролера знаходяться у вихідних значеннях. Перед початком роботи всі порти та регістри встановлюються у їх робочий стан, задається початкова адреса flash-пам'яті. Після цих операцій виконується підпрограма конфігурації АЦП, яка буде описана нижче. Після завершення цієї підпрограми система виконує кругові цикли, під час дії яких вона очікує керуючого сигналу. Цей сигнал повідомляє системі, що потрібно в даний час робити. Тобто після натиснення будь-якої з трьох кнопок починають виконуватися внутрішні функціональні підпрограми. Коли натиснута кнопка “Старт/Стоп”, відбувається запуск таймера, запис початкового значення на flash-пам'ять та починається підпрограма зчитування даних з АЦП.

Після закінчення цієї підпрограми перевіряється чи не натиснута ця кнопка ще раз. Якщо вона не натиснута, то підпрограма зчитування даних з АЦП виконується знову. Цей цикл повторюється до тих пір, поки не натиснута кнопка “Стар/Стоп” ще раз. При повторному натисненні відбувається остановка таймера, запис його значення на flash-пам'ять та вихід з цього циклу і виконання повертається в головний цикл, тобто система знову очікує натиснення будь-якої з трьох кнопок для виконання подальших дій.

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк. 45
Вим.	Арк.	№ докум.	Підпис	Дата		

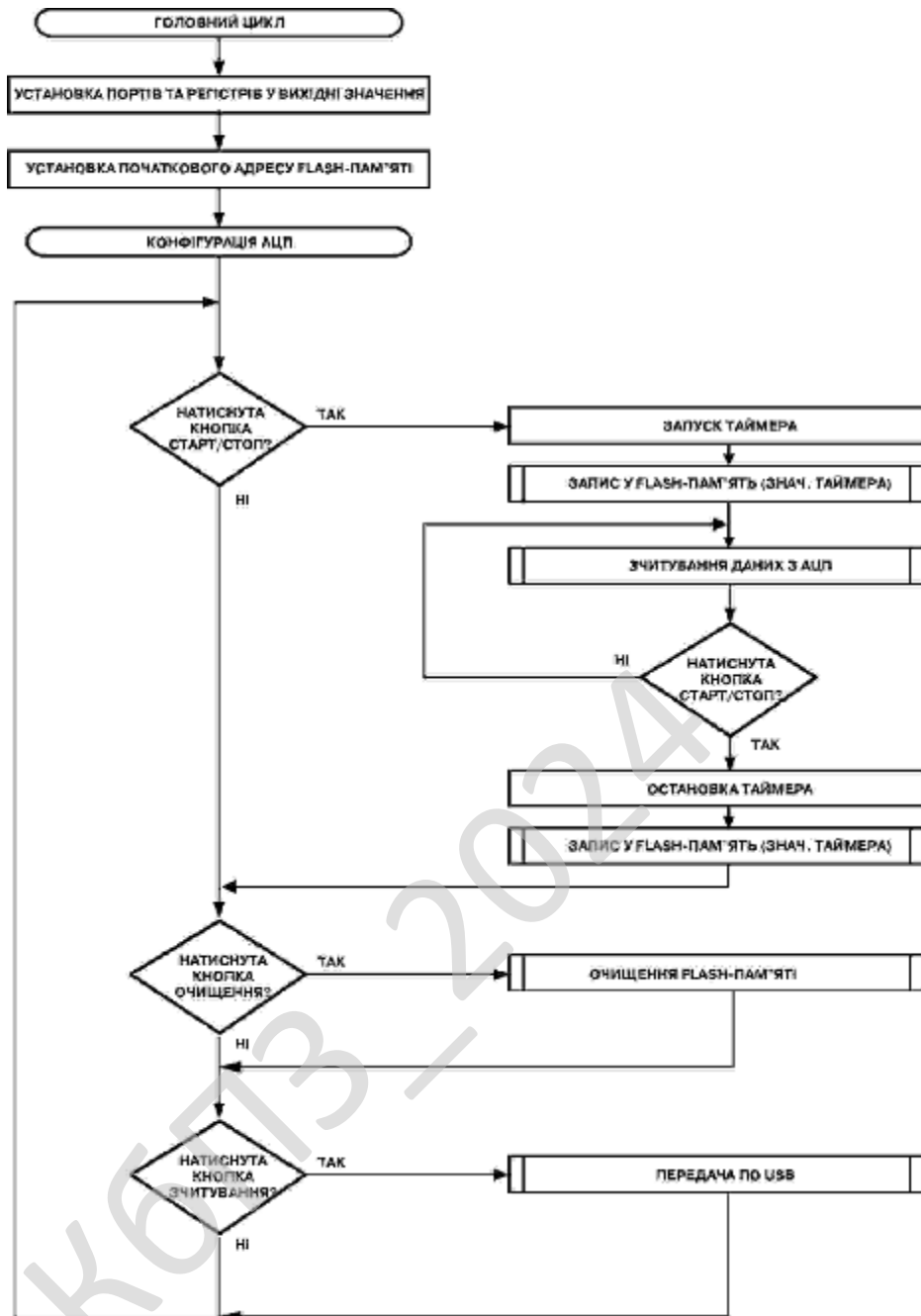


Рисунок 4.1 – Алгоритм головного циклу.

При розпізнанні натиснення кнопки “очищення” виконується підпрограма очищення flash-пам’яті. Після закінчення цієї підпрограми система знову повертається у стан очікування.

Коли натиснута кнопка “зчитування” виконується підпрограма передачі інформації, яка записана на flash-пам’ять, по протоколу USB.

## Алгоритм конфігурації АЦП



Рисунок 4.2 – Алгоритм конфігурування АЦП

Після включення АЦП до джерела живлення всі внутрішні регістри знаходяться у вихідних значеннях. Перед початком самої конфігурації АЦП у потрібний режим роботи, потрібно конфігурувати SPI інтерфейс мікроконтролера. Після конфігурації та ініціалізації SPI, по ньому проводиться конфігурація АЦП. Як відомо з опису, АЦП має вісім внутрішніх регістрів, значення яких налагоджують АЦП в потрібний режим роботи. Звертання до будь-якого з цих регістрів виконується процедурою запису у Регістр Зв'язку. Тобто записом у регістр зв'язку ми повідомляємо йому в який чи з якого з регістрів відбудуться читання чи запис. Записом у регістр зв'язку ми повідомляємо йому,

що наступний запис відбудеться у регістр Filter Low Register. Після чого записуємо потрібні дані в цей регістр. Таким же чином виконується запис в усі інші регістри АЦП. Записуємо дані у регістр Filter High Register. Ці регістри визначають частоту зміни даних в регістрі даних АЦП та частоту зрізу внутрішнього цифрового фільтру.

Записом у регістр Mode Register потрібних значень ми ініціюємо початок калібрування АЦП по заданому каналу, який задається записом у Регістр Зв'язку. Так як канали однакові за параметрами калібрування по іншому каналу не є потрібним.

### **Алгоритм зчитування з АЦП**

Як відомо має три канали з яких зчитуються дані. Довжина слова в регістрі даних АЦП – 24 біти, тобто три байти. Мікроконтролер – 8-розрядний, тому зчитування даних виконується по одному байту.

Після початку підпрограми зчитування даних з АЦП встановлюємо активний канал 1. Дані в регістрі даних АЦП готові до зчитування з частотою, яка раніше встановлена. Коли дані готові до зчитування АЦП сигналізує мікроконтролеру низьким рівнем сигналу DRDY, тому проводиться опитування цього контакту. При високому рівні цього сигналу, система очікує коли він досягне низького рівня. При виникненні низького рівня встановлюється лічильник байт рівний 1 та проводиться зчитування першого байту з регістру даних АЦП та запис його у flash-пам'ять. Так як результат рівний 3 байтам, інкрементуємо лічильник і зчитуємо наступний байт даних. Після зчитування третього байту та інкременту лічильника, лічильник стає більший 3. Тоді відбувається переключення каналу шляхом інкременту лічильника каналу. Процедура зчитування 3 байтового слова починається знову. Тільки оцифровані дані знімаються з каналу на одиницю більше. Коли дані зчитані з трьох каналів підпрограма закінчується і повертається в головний цикл.

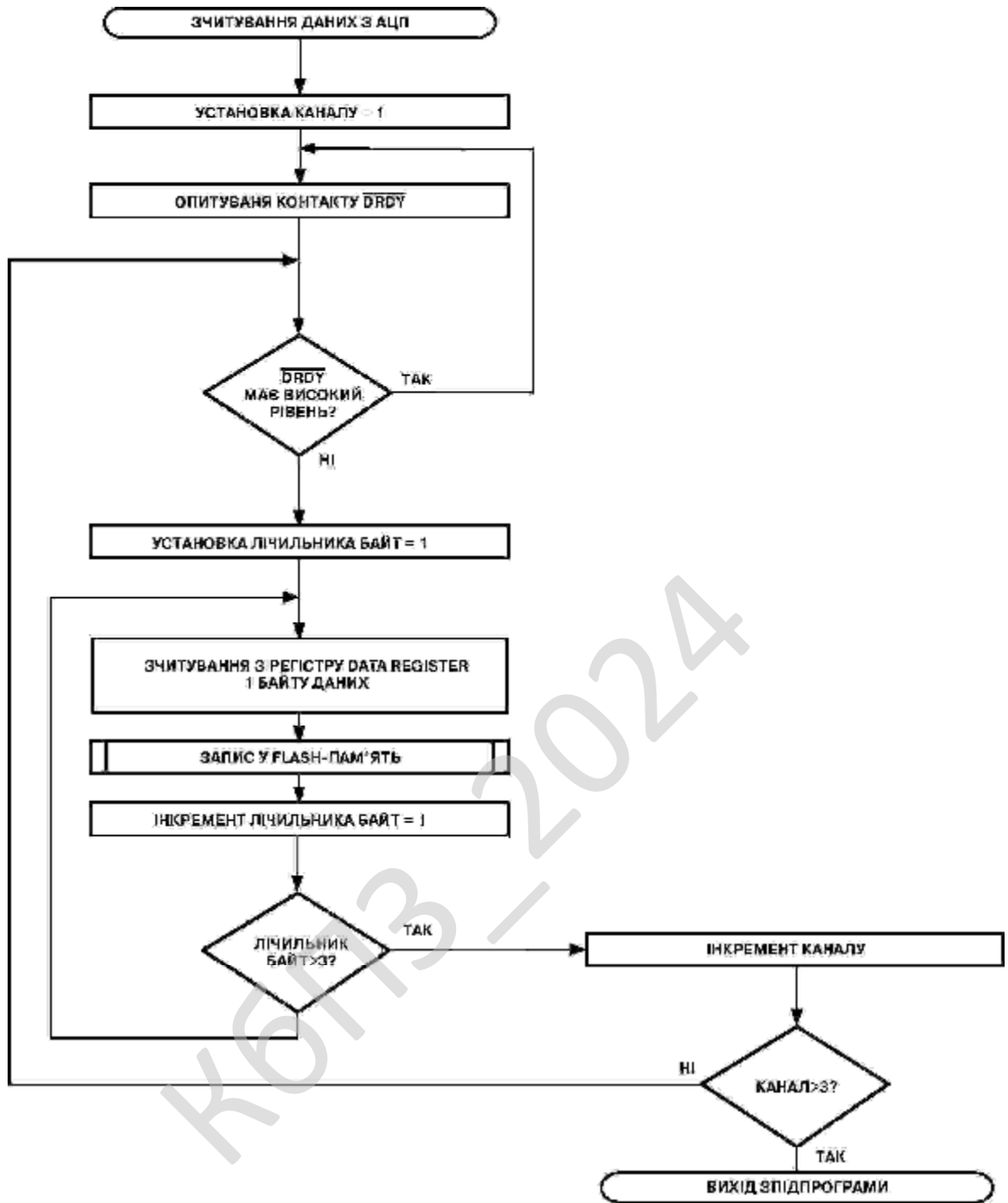


Рисунок 4.3 – Алгоритм зчитування даних з АЦП.

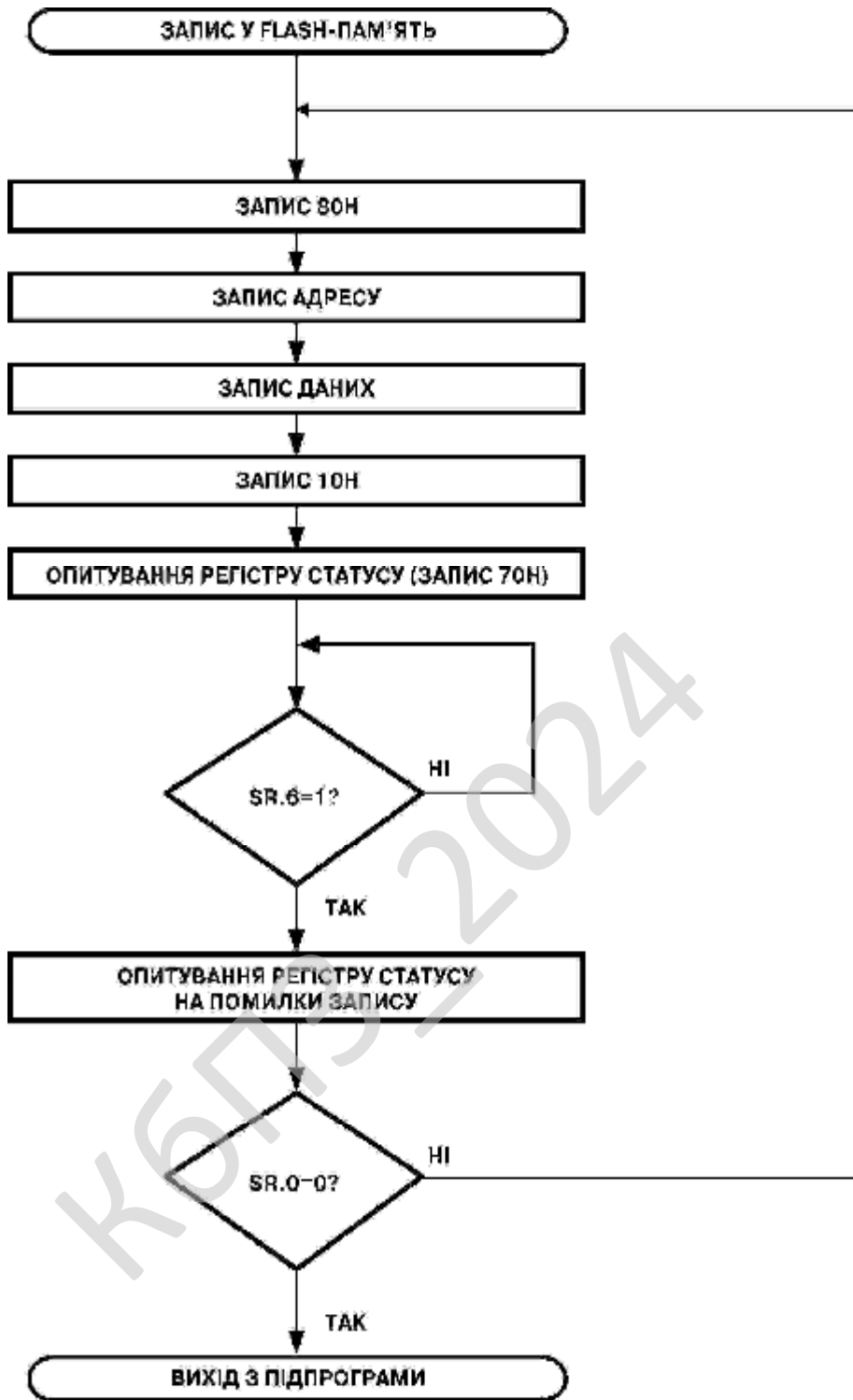


Рисунок 4.4 – Алгоритм запису даних на flash- пам'ять

### Алгоритми передачі даних по USB

Для розробки алгоритму передачі даних по інтерфейсу USB потрібно, для початку, ознайомитись з роботою шини USB.

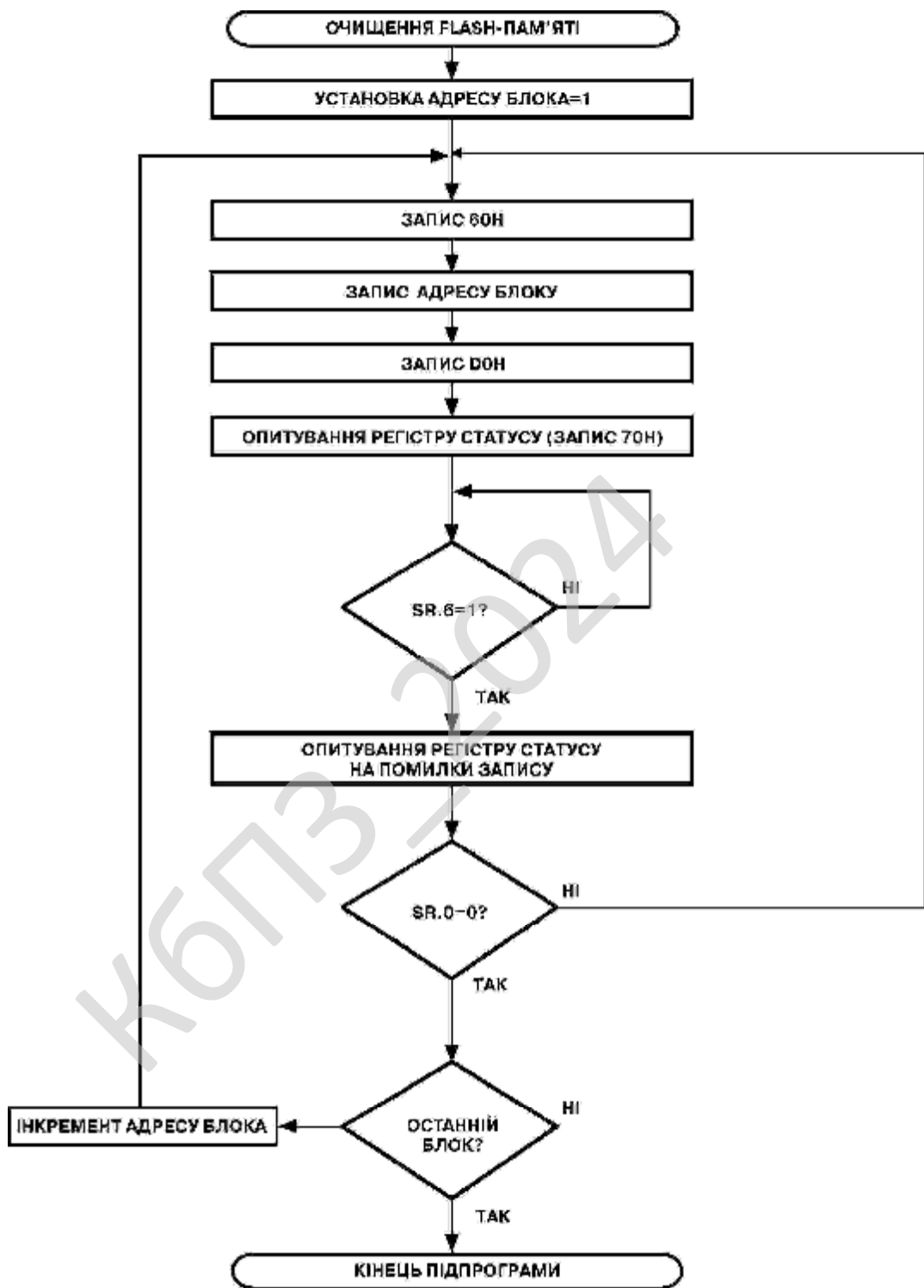


Рисунок 4.5 – Очищення flash- пам'яті

Як повідомлялося, USB - цілком контрольована хостом шина. У системі USB може бути тільки один майстер (керуючий елемент) – комп'ютер-хост. Тому для

передбачення передачі інформації на flash-накопичувач на комп'ютері потрібно два режими роботи. Режим Master – пристрій виступає, як керуючий елемент, та режим Slave – підлеглий пристрій виступає, як підлеглий. Для визначення режиму роботи, в залежності чи пристрій підключений до комп'ютера, чи до пристрою підключений flash-накопичувач головний алгоритм очікує сигналу запиту на прийом інформації IN, якщо сигнал IN не поступає за 30с, то пристрій переходить в режим Master, тобто передача проводиться на flash-накопичувач. Якщо отриманий сигнал IN, керування процесом передачі інформації передається комп'ютеру.

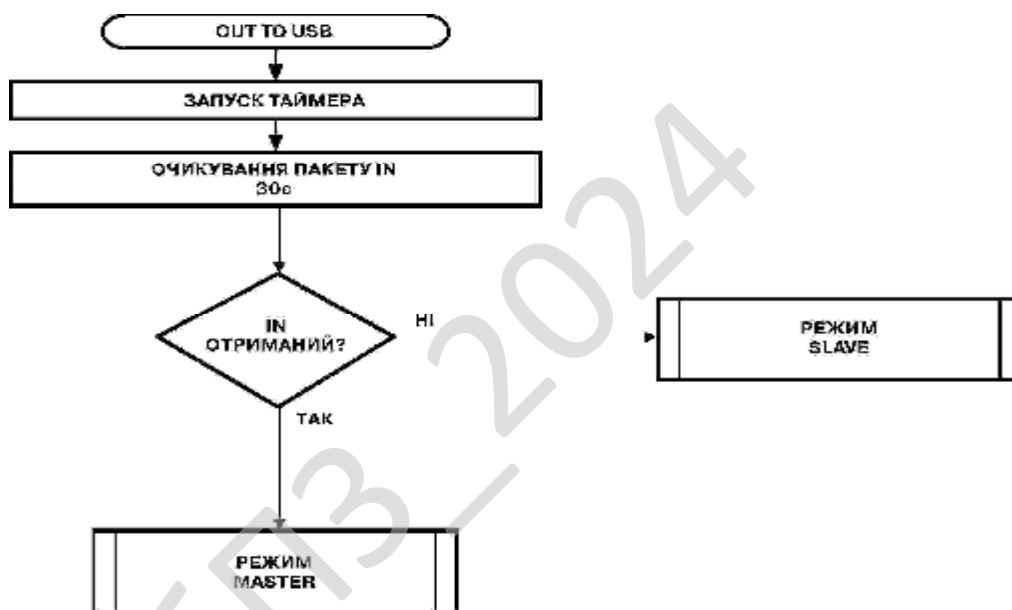


Рисунок 4.6 - Головний алгоритм передачі даних по USB

Графічне зображення алгоритму роботи в режимі Master представлено на рисунку .В режимі Master(використовується режим BULK USB – передача по 64 Байти) пристрій відправляє сигнал запиту на передачу інформації OUT, та встановлюється лічильник пакетів по 64Байт. При відсутності зв'язку з накопичувачем повертається сигнал STALL. Якщо накопичувач підключений, відбувається формування пакету 64Байт та відправка його по USB. Сигнал ACK використовується для перевірки на правильність даних, що отримані накопичувачем. При отримання сигналу STACK цей пакет відправляється ще раз.

Така відправка проходить поки всі дані не передані.

### Алгоритм режиму роботи Slave

Графічне зображення алгоритму роботи в режимі Slaver представлено на рисунку При підключення пристрою до комп'ютера пристрій повинен відповісти на запити. "Get\_Descriptor/Device", "Set\_Address", Get\_Descriptor. Та проводиться передача аналогічна режиму Master.

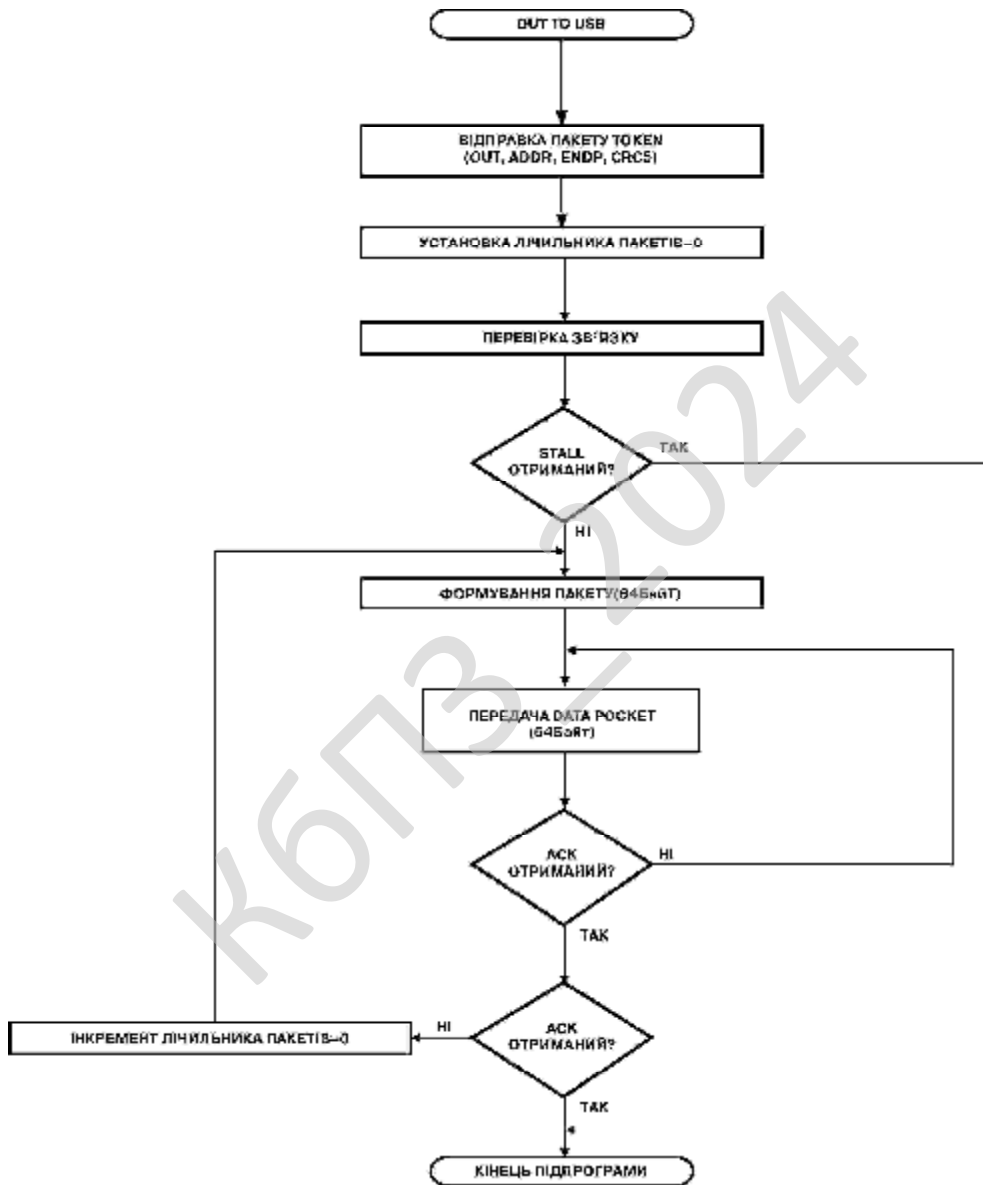


Рисунок 4.7 – Алгоритм роботи в режимі Master

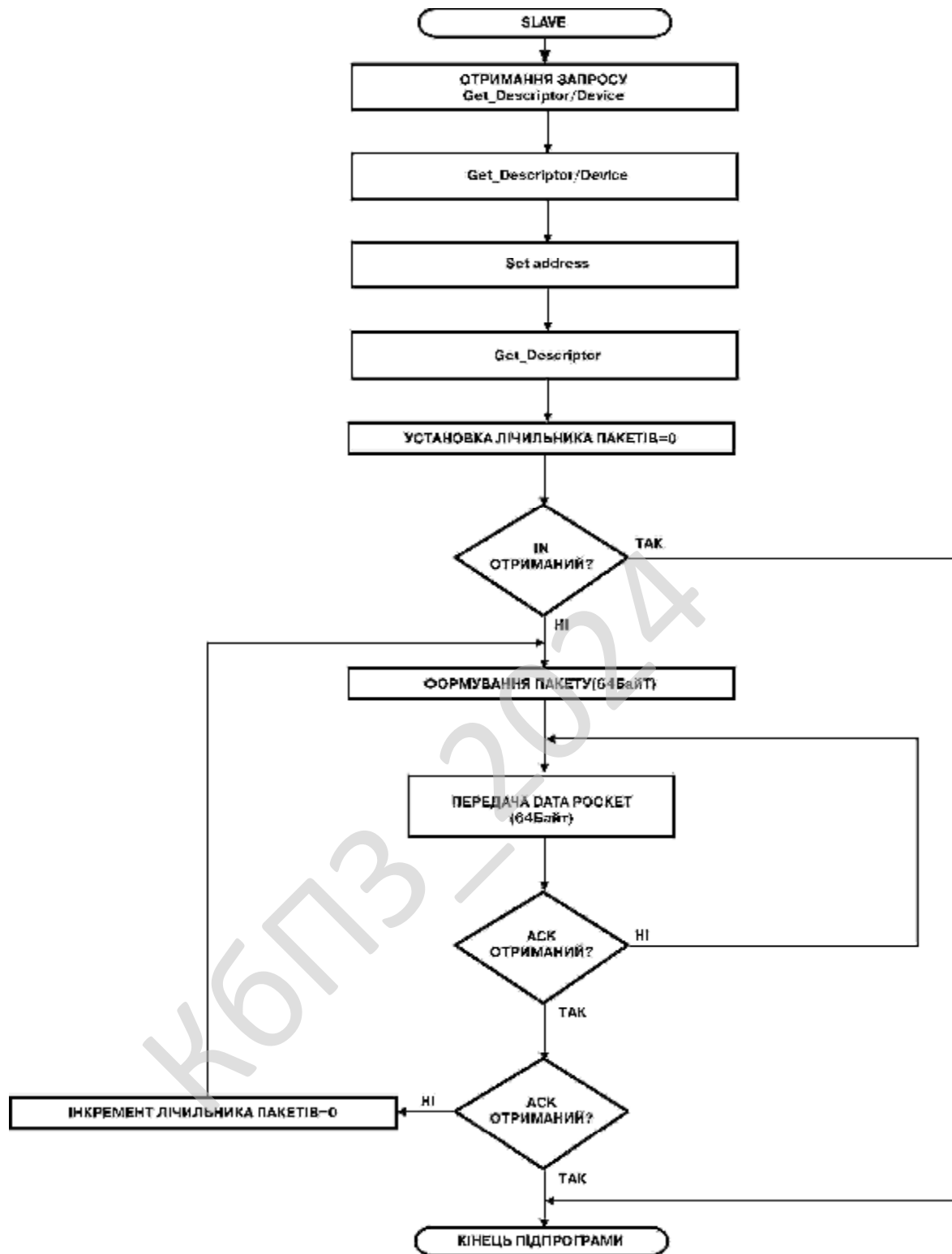


Рисунок 4.8– Алгоритм роботи в режимі Slave



## встановленні програми за допомогою:

```
procedure Sn_HDD;
uses Registry;
var reg:TRegistry;
    SerialNum,a,b:dword;
    VolumeName:array [0..255] of char;
begin
    if GetVolumeInformation(PChar('c:\'), VolumeName, SizeOf(VolumeName),
@SerialNum, a, b, nil, 0)
    then
    begin
        reg:=TRegistry.Create;
        reg.RootKey:=HKEY_LOCAL_MACHINE;
        reg.OpenKey('HardWareSN', false);
        RegKeySettDw(HKEY_LOCAL_MACHINE,'HardWareSN',SerialNum);
        reg.CloseKey;
        reg.Free;
    end;
end;
```

Багато робіт присвячено захисту даних, захисту програмного забезпечення та захисту інформацій в цілому [42. 43, 44, 45, 46, 47, 48, 49, 50.], проте просто захист від несанкціонованого копіювання роботи недоцільно. Бажано давати демонстрацію роботи програми. Наприклад, можна давати час та/або кількість відпрацювань, для чого можна використати реєстр.

```
function Beta:boolean;
uses Registry;
var regD,regK:TRegistry;
    Rd:TDateTime;
    Rk:DWORD;
begin
    regD := TRegistry.Create;
    regD.RootKey := HKEY_LOCAL_MACHINE;
    if reg.OpenKey('HardWareData', false)
    then
    begin
        Rd:=RegKeyGetDateTime(HKEY_LOCAL_MACHINE,'HardWareData');
        regK:=TRegistry.Create;
        regK.OpenKey('HardWareK', true);
        Rk:= RegKeyGetDw(HKEY_LOCAL_MACHINE,'HardWareK');
        if Rd+30>Date and Rk>100
```

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		56

```
then
Result := false
else
begin
RegKeySettDw(HKEY_LOCAL_MACHINE, 'HardWareK', Rk+1);
Result := true;
end;
end
else
begin
regD.OpenKey('HardWareData', true)
RegKeySetDateTime(HKEY_LOCAL_MACHINE, 'HardWareD', Date);
regK := TRegistry.Create;
regK.OpenKey('HardWareK', true)
RegKeySettDw(HKEY_LOCAL_MACHINE, 'HardWareK', 1);
result := true;
end;
regD.CloseKey;
regD.Free;
regK.CloseKey;
regK.Free;
end;
```

КБПЗ\_2024

					<b>БКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		57

**5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ  
В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ**

Після включення приладу до джерела живлення, всі регістри та порти мікроконтролера знаходяться у вихідних значеннях. Перед початком роботи всі порти та регістри встановлюються у їх робочий стан, задається початкова адреса flash-пам'яті. Після цих операцій виконується підпрограма конфігурації АЦП, яка буде описана нижче. Після завершення цієї підпрограми система виконує кругові цикли, під час дії яких вона очікує керуючого сигналу. Цей сигнал повідомляє системі, що потрібно в даний час робити. Тобто після натиснення будь-якої з трьох кнопок починають виконуватися внутрішні функціональні підпрограми. Коли натиснута кнопка “Старт/Стоп”, відбувається запуск таймера, запис початкового значення на flash-пам'ять та починається підпрограма зчитування даних з АЦП. Після закінчення цієї підпрограми перевіряється чи не натиснута ця кнопка ще раз. Якщо вона не натиснута, то підпрограма зчитування даних з АЦП виконується знову. Цей цикл повторюється до тих пір, поки не натиснута кнопка “Стар/Стоп” ще раз. При повторному натисненні відбувається остановка таймера, запис його значення на flash-пам'ять та вихід з цього циклу і виконання повертається в головний цикл, тобто система знову очікує натиснення будь-якої з трьох кнопок для виконання подальших дій.

При розпізнанні натиснення кнопки “очищення” виконується підпрограма очищення flash-пам'яті. Після закінчення цієї підпрограми система знову повертається у стан очікування.

Коли натиснута кнопка “зчитування” виконується підпрограма передачі інформації, яка записана на flash-пам'ять, по протоколу USB.

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк. 58
Вим.	Арк.	№ докум.	Підпис	Дата		

## 6 НАУКОВА НОВИЗНА

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів.

*Метою розробки є дослідження та програмна реалізація системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів.*

*Об'єктом дослідження є процес реєстрації та запису сигналів ЕКГ.*

*Предметом дослідження є методи та засоби реєстрації та запису сигналів ЕКГ.*

*Методи дослідження базуються на базуються на методах цифрової обробки сигналів, методах математичної статистики, методах розробки програмного забезпечення.*

**Наукова новизна отриманих результатів.** У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

– Досліджено основні методи сучасної діагностики роботи серцево-судинної системи людини, а також виділено їх основні плюси та критерії застосування. Окремо виділено метод ЕКГ, та спосіб читання отриманих результатів, а саме електрокардіограми.

– Розроблено алгоритми та методи обробки сигналів та модернізовано прилад для реєстрації ЕКГ.

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		59

## 7 МАРКЕТИНГОВЕ ТА ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ ІТ-ПРОЄКТУ

### 7.1 Визначення цільової аудиторії кінцевого готового продукту

Результати дослідження та програмної реалізації системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів можуть бути цікавими досить широкій аудиторії (рис. 7.1).

<b>Медичні заклади та клініки</b>	Лікарі, кардіологи та медичні установи можуть зацікавитися цією технологією для вдосконалення діагностики та моніторингу стану серця пацієнтів. Зокрема, компактність та зручність запису даних на флеш-накопичувачі роблять систему більш мобільною та доступною для використання.
<b>Розробники медичного обладнання</b>	Компанії, які займаються розробкою та виробництвом медичних приладів, можуть бути зацікавлені у впровадженні нових рішень для покращення точності та надійності діагностики ЕКГ.
<b>Науково-дослідницькі установи та університети</b>	Викладачі, дослідники та студенти, які займаються біомедичними технологіями, можуть використовувати результати дослідження для подальших досліджень або як основу для нових проєктів в галузі біомедицини.
<b>ІТ-спеціалісти та розробники програмного забезпечення</b>	Ця технологія може бути цікавою для розробників, які спеціалізуються на створенні медичних додатків та систем аналізу даних. Вона відкриває можливості для створення нових програмних рішень у сфері охорони здоров'я.
<b>Лабораторії медичних технологій та стартапи</b>	Компанії, які працюють у сфері медичних інновацій, можуть зацікавитися впровадженням нових методів зберігання та обробки медичних даних на флеш-накопичувачах.
<b>Сектор телемедицини</b>	Установи, які займаються розробкою рішень для віддаленого моніторингу здоров'я пацієнтів, можуть скористатися можливостями мобільного та надійного зберігання даних для передачі й аналізу ЕКГ в реальному часі.

Рисунок 7.1 – Цільова аудиторія проєкту

					ВКРМ-123.24.0019.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		60

Таким чином, технологія може бути корисною як для практичних медичних застосувань, так і для наукових досліджень та розробки нових медичних продуктів.

## 7.2 Оцінка привабливості шляхом застосування методів експертних оцінок

Оцінка привабливості програмної реалізації системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів може бути здійснена шляхом застосування методу експертних оцінок.

Спершу визначимо критерії, за якими експерти будуть оцінювати привабливість програмного рішення (рис. 7.2).



Рисунок 7.2 – Критерії експертної оцінки

Створюється група з 5-10 експертів, які мають досвід у галузі біомедичних технологій, медичного обладнання та програмного забезпечення. Приклад: кардіолог, розробник медичного ПЗ, ІТ-спеціаліст, інженер з медичного обладнання, менеджер проекту. Експерти визначають важливість кожного критерію шляхом надання вагових коефіцієнтів (від 0 до 1), де сума всіх ваг

дорівнює 1. В нашому випадку: точність діагностики (A): 0.3; зручність використання (B): 0.1; надійність зберігання даних (C): 0.15; швидкість обробки сигналів (D): 0.2; інтеграція з медичними системами (E): 0.15; вартість реалізації (F): 0.1.

Кожен експерт оцінює привабливість системи за кожним критерієм за шкалою від 1 до 10 (де 1 — мінімальна привабливість, 10 — максимальна).

Таблиця 7.1 – Зведені результати експертних оцінок

Критерій	Експерт 1	Експерт 2	Експерт 3	Експерт 4	Експерт 5	Середнє значення
A	8	9	7	8	9	8.2
B	7	8	6	7	8	7.2
C	9	8	8	9	8	8.4
D	8	8	9	8	7	8.0
E	7	8	7	6	8	7.2
F	6	7	8	6	7	6.8

Інтегральна оцінка розраховується як сума добутків середніх оцінок на вагові коефіцієнти для кожного критерію: Інтегральна оцінка =  $(A \times w_A) + (B \times w_B) + (C \times w_C) + (D \times w_D) + (E \times w_E) + (F \times w_F)$

Підставляючи дані:

Інтегральна оцінка =  $(8.2 \times 0.3) + (7.2 \times 0.1) + (8.4 \times 0.15) + (8.0 \times 0.2) + (7.2 \times 0.15) + (6.8 \times 0.1)$

Інтегральна оцінка =  $2.46 + 0.72 + 1.26 + 1.6 + 1.08 + 0.68 = 7.8$

Інтегральна оцінка 7.8 свідчить про високу привабливість системи з точки зору експертів, що вказує на потенційний інтерес до даної технології як з боку медичних закладів, так і з боку розробників медичних пристроїв та IT-фахівців.

### 7.3 Вибір методу оцінки вартості ПЗ

Для оцінки вартості програмної реалізації системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів доцільно розглянути декілька методів. Найбільш підходящим методом буде залежати від специфіки проєкту, наявних даних і мети оцінки.

Для оцінки вартості програмної реалізації системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів найбільш доцільним буде метод оцінки за аналогами (Bottom-Up Cost Estimation).

Цей метод забезпечить найбільш точні результати завдяки можливості розбити проєкт на етапи (наприклад, розробка програмного коду, тестування, інтеграція з обладнанням, підготовка користувацького інтерфейсу, забезпечення надійності зберігання даних тощо) і оцінити кожен з них окремо. Він також дозволяє врахувати специфічні вимоги проєкту та всі можливі витрати, включаючи ризики.

Процес реалізації методу оцінки за аналогами:

- складіть список всіх компонентів проєкту (розробка, тестування, підтримка);
- оцініть витрати для кожного етапу, використовуючи історичні дані, попередній досвід або залучення експертів;
- підсумуйте витрати на кожен компонент, щоб отримати загальну вартість;
- додайте резерв на ризики, щоб мати фінансовий запас на непередбачувані витрати.

Цей підхід забезпечить об'єктивність і точність, що важливо для медичних систем, де точність і надійність критично важливі.

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		63

## 7.4 Розрахунок економічної ефективності від впровадження реалізованого ПЗ як фактору його привабливості

Впровадження системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів може мати значну економічну ефективність для медичних закладів та пацієнтів. Припустимо, що система впроваджується у великій клініці, яка проводить близько 10 000 діагностичних процедур ЕКГ на рік. До впровадження нової системи клініка використовувала традиційні методи зберігання та обробки даних, що мали свої обмеження в обробці та збереженні інформації.

Таблиця 7.2 – Основні показники впровадження проекту

<b>Оцінка витрат до впровадження нової системи</b>
До впровадження нової системи клініка використовувала паперові записи та стаціонарні комп'ютерні системи для збереження даних. Річні витрати включають: Витрати на паперові матеріали та друк: \$4 000 на рік. Витрати на обслуговування стаціонарних систем зберігання даних: \$15 000 на рік. Час медичного персоналу на введення даних вручну та обробку: \$20 000 на рік (на оплату праці персоналу). Загальні витрати до впровадження: \$39 000 на рік.
<b>Оцінка витрат після впровадження нової системи</b>
Після впровадження нової системи на базі флеш-накопичувачів витрати змінилися: Витрати на обладнання та флеш-накопичувачі: \$5 000 на рік. Витрати на підтримку програмного забезпечення: \$7 000 на рік. Час медичного персоналу на обробку даних зменшився на 50% завдяки автоматизації — \$10 000 на рік (економія \$10 000). Загальні витрати після впровадження: \$22 000 на рік.
<b>Розрахунок економії</b>
Економія=Витрати до впровадження–Витрати після впровадження Економія=39000–22000=17000доларів на рік
<b>Інші вигоди від впровадження</b>

Впровадження системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів дозволяє заощадити \$17 000 на рік, забезпечуючи

## Продовження Таблиці 7.2 – Основні показники впровадження проекту

Окрім прямої економії витрат, впровадження системи має й інші економічні вигоди:

**Збільшення точності діагностики:** Менше помилок при обробці даних, що знижує витрати на повторні процедури та знижує ризики для пацієнтів.

**Підвищення ефективності медичного персоналу:** Менше часу витрачається на введення та обробку даних, що дозволяє персоналу обслуговувати більше пацієнтів.

**Зменшення витрат на фізичне зберігання даних:** Відсутність потреби в паперових архівах, економія місця в медичній установі.

**Покращення якості обслуговування пацієнтів:** Швидший доступ до результатів, зменшення часу на очікування діагнозу.

### **Показник рентабельності інвестицій (ROI)**

Для оцінки рентабельності інвестицій можна використати показник ROI (Return on Investment):

$$\text{ROI} = \frac{\text{Економія}}{\text{Вартість впровадження}} \times 100\%$$

$$\text{ROI} = \frac{17000}{50000} \times 100\% = 34\%$$

### **Період окупності**

Період окупності системи — це час, за який інвестиції у впровадження будуть повернені за рахунок економії витрат.

$$\text{Період окупності} = \frac{50000}{17000} \approx 2.94 \text{ роки}$$

період окупності близько 3 років. Окрім економії, система підвищує якість діагностики, знижує витрати на обслуговування та покращує ефективність роботи медичного персоналу, що є важливими додатковими вигодами для медичних закладів.

## **7.5 Пропозиція алгоритму просування проекту розробки ПЗ**

Алгоритм просування проекту програмної реалізації системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів подаємо на рисунку 7.3.

Алгоритм просування проекту програмної реалізації системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів включає аналіз ринку, створення унікальної торгової пропозиції, активну маркетингову стратегію та моніторинг ефективності. Головним є створення чітких та

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		65

### 1. Аналіз ринку та цільової аудиторії

- вивчення ринку медичних технологій: з'ясувати, які системи вже існують, їхні сильні та слабкі сторони.
- визначення цільової аудиторії хто є потенційними покупцями? це можуть бути великі лікарні, приватні клініки, кардіологічні центри, медичні лабораторії або амбулаторії медичні заклади.
- визначення потреб аудиторії: які проблеми може вирішити ваша система? це можуть бути, наприклад, зменшення витрат на зберігання даних, поліпшення точності діагностики, спрощення доступу до даних.

### 2. Розробка унікальної торгової пропозиції (УТП)

- визначення переваг: сформулювати основні переваги системи, такі як точність діагностики, швидкість обробки даних, надійність зберігання на флеш-накопичувачах, низькі експлуатаційні витрати.
- створення УТП: використати переваги для формулювання переконливої УТП, що буде привабливою для цільової аудиторії.

### 3. Розробка маркетингової стратегії

- визначення маркетингових каналів: вибрати, які канали будуть використовуватися для просування проекту, наприклад, спеціалізовані медичні конференції, вебінари, соціальні мережі, галузеві журнали, платформи для вебінарів та онлайн-презентацій.
- встановлення цілей просування: наприклад, залучення перших 10 клієнтів протягом 6 місяців або досягнення 500 підписувачів на вебсайті компанії.

### 4. Створення бренду проекту

- розробка бренду: створити ім'я бренду, логотип, стиль презентацій (кольори, шрифти).
- створення вебсайту: розробити інформативний вебсайт, який буде представляти систему, її можливості, переваги та переваги використання.
- створення матеріалів для просування: підготувати буклети, відео-демонстрації, інструкції, презентації, інформаційні листи для потенційних клієнтів.

### 5. Тестування системи та створення кейсів

- проведення тестових впроваджень: знайти кілька медичних установ для проведення пілотного впровадження системи, це можуть бути партнери або клієнти, які готові випробувати продукт на практиці.
- збір відгуків: збирати зворотні зв'язки від лікарів і персоналу, які використовували систему, виправляти недоліки та удосконалювати продукт.
- створення успішних кейсів: на основі тестових впроваджень створити кейси, що демонструють економічну вигоду та підвищення ефективності медичних закладів.

### 6. Активне просування системи

- публікації у спеціалізованих медичних виданнях: розмістити статті, що описують переваги системи та результати пілотних впроваджень у професійних медичних виданнях.
- участь у галузевих заходах: відвідувати конференції, виставки, медичні форуми для демонстрації системи, організувати студії та презентації для медичних фахівців.
- онлайн-просування: використовувати соціальні мережі (LinkedIn, Facebook), вебінари та онлайн-конференції для реклами продукту, запускати цільову рекламу на відповідні сегменти аудиторії.
- робота з впливовими особами (KOL): взаємодіяти з лікарями та медичними експертами, які можуть рекомендувати продукт у своїй професійній мережі.

### 7. Використання стратегій продажів

- демонстрації продукту: проводити персоналізовані онлайн- або офлайн-демонстрації для потенційних клієнтів, надавати доступ до демо-версії або проводити безкоштовні тестування на короткий час.
- оферти та знижки: надати знижки на першу покупку, безкоштовне навчання персоналу або спеціальні умови для перших клієнтів.
- підтримка клієнтів: запропонувати високоякісну технічну підтримку, навчання персоналу та консультації, забезпечити клієнтам максимальний комфорт під час впровадження системи.

### 8. Моніторинг та аналіз ефективності просування

- моніторинг показників: аналізувати кількість звернень, продажів, відвідувань сайту, взаємодію в соціальних мережах, визначити, які канали просування є найбільш ефективними.
- корекція стратегії: вносити зміни до маркетингової стратегії на основі аналізу ефективності, оптимізувати витрати на маркетинг та рекламні кампанії.

### 9. Розширення присутності на ринку

- масштабування проекту: після успішного впровадження на локальному ринку розпочати просування на регіональній або міжнародній рівні.
- розширення продукту: розробити додаткові функції та модулі, які можуть залучити ширшу аудиторію (наприклад, інтеграція з іншими медичними системами або хмарне зберігання даних).
- формування партнерських відносин: співпрацювати з іншими медичними компаніями, дистрибуторами та клініками для розширення мережі продажів.

Рисунок 7.3 – Алгоритм просування проекту

зрозумілих переваг для цільової аудиторії, активне використання всіх можливих маркетингових каналів та підтримка клієнтів на всіх етапах впровадження.

## 7.6 Оптимізація каналів збуту та шляхів реалізації ПЗ

Для оптимізації каналів збуту та шляхів реалізації проекту програмної реалізації системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів варто застосувати кілька стратегій. Вони допоможуть збільшити ефективність продажів, розширити аудиторію та підвищити впізнаваність продукту. Варто перерахувати їх:

- організувати прямі продажі через власних менеджерів з продажів, які зможуть безпосередньо контактувати з лікарнями, медичними установами, клініками та іншими потенційними клієнтами;
- забезпечити технічну підтримку та навчання для персоналу клієнтів, що допоможе створити тісні стосунки з ними;
- залучити дистриб'юторів у галузі медичних технологій, які вже мають добре налагоджені канали продажів;
- працювати з медичними постачальниками обладнання, які можуть включити вашу систему у свої портфелі рішень;
- використовувати спеціалізовані медичні онлайн-платформи для продажу програмного забезпечення та медичних систем;
- використовувати saas-платформи для надання доступу до програмного забезпечення за передплатою;
- створити вебсайт з блогом, де публікувати статті, кейси, дослідження та експертні думки, що розкривають переваги вашої системи;
- підготувати навчальні відео, інструкції та вебінари, які пояснюють використання системи, переваги та економічну вигоду для медичних установ;
- активно використовувати соціальні мережі, такі як LinkedIn, Facebook, Instagram, для таргетованої реклами на медичний персонал та адміністрацію клінік;

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		67

- створити професійні групи в LinkedIn для обговорення інновацій в медичних технологіях, де можна просувати свою систему;
- використовувати платну рекламу в Google для залучення медичних установ, які шукають рішення для діагностики та обробки ЕКГ;
- запускати рекламні кампанії в соціальних мережах для цільової аудиторії;
- брати участь у великих медичних виставках та конференціях, де можна безпосередньо демонструвати систему потенційним клієнтам та партнерам;
- організувати стенди на тематичних подіях, де відвідувачі можуть побачити систему в дії та отримати консультацію від спеціалістів;
- виступати спонсором медичних заходів, форумів або семінарів, щоб підвищити впізнаваність бренду;
- організувати доповіді чи презентації про інновації в діагностиці серцево-судинних захворювань, де можна продемонструвати переваги вашого продукту;
- налагодити співпрацю з медичними асоціаціями, які можуть стати партнерами у просуванні продукту серед своїх членів;
- запропонувати спеціальні умови для медичних установ, які готові стати партнерами та впровадити систему на ранній стадії;
- співпрацювати з медичними університетами та навчальними закладами, де можна впроваджувати систему для навчання студентів;
- створити партнерські програми з навчальними закладами, що дозволяють випробовувати нові функції та збирати відгуки;
- надати можливість потенційним клієнтам безкоштовно протестувати систему на визначений період;
- пропонувати спеціальні умови для перших клієнтів, що сприятиме швидшому впровадженню системи;

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		68

- створити програми лояльності для постійних клієнтів, пропонуючи знижки на оновлення, технічну підтримку або додаткові послуги;
- запропонувати додаткові переваги для клієнтів, які залучатимуть нових користувачів (наприклад, знижки за рекомендації);
- проводити регулярні опитування серед клієнтів для оцінки задоволеності, збирати їхні рекомендації та побажання щодо вдосконалення продукту;
- використовувати зібрані дані для вдосконалення системи, додавання нових функцій або покращення наявних;
- публікувати кейси та відгуки клієнтів на вебсайті, в блогах та соціальних мережах. це підвищить довіру до продукту та спонукатиме нових клієнтів до придбання;
- пропонувати систему як модульний продукт, що дозволяє клієнтам вибирати тільки необхідні функції або додавати нові модулі у майбутньому;
- це зробить систему доступнішою та дозволить клієнтам налаштувати її відповідно до своїх потреб;
- забезпечити можливість інтеграції з іншими медичними інформаційними системами (наприклад, EHR — electronic health record), що спростить використання та підвищить привабливість продукту для великих установ;
- постійно аналізувати ефективність різних каналів продажів, визначати, які з них працюють найкраще, та перерозподіляти ресурси;
- проводити а/в тестування різних маркетингових стратегій (наприклад, різних рекламних повідомлень, презентацій продукту) для визначення найбільш ефективних варіантів;
- регулярно переглядати маркетинговий бюджет та оптимізувати витрати, зосереджуючись на найбільш продуктивних каналах.

Оптимізація каналів збуту та шляхів реалізації проєкту програмної реалізації системи діагностики, обробки та запису сигналів ЕКГ потребує

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк. 69
Вим.	Арк.	№ докум.	Підпис	Дата		

комплексного підходу. Включення різних каналів продажів, активне використання цифрових інструментів, залучення партнерів, створення програм лояльності та проведення аналізу ефективності допоможе максимізувати охоплення цільової аудиторії та підвищити успішність впровадження системи на ринку.

### 7.7 Визначення ключових факторів успіху конкретного проєкту

Ключовими факторами успіху проєкту програмної реалізації системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів

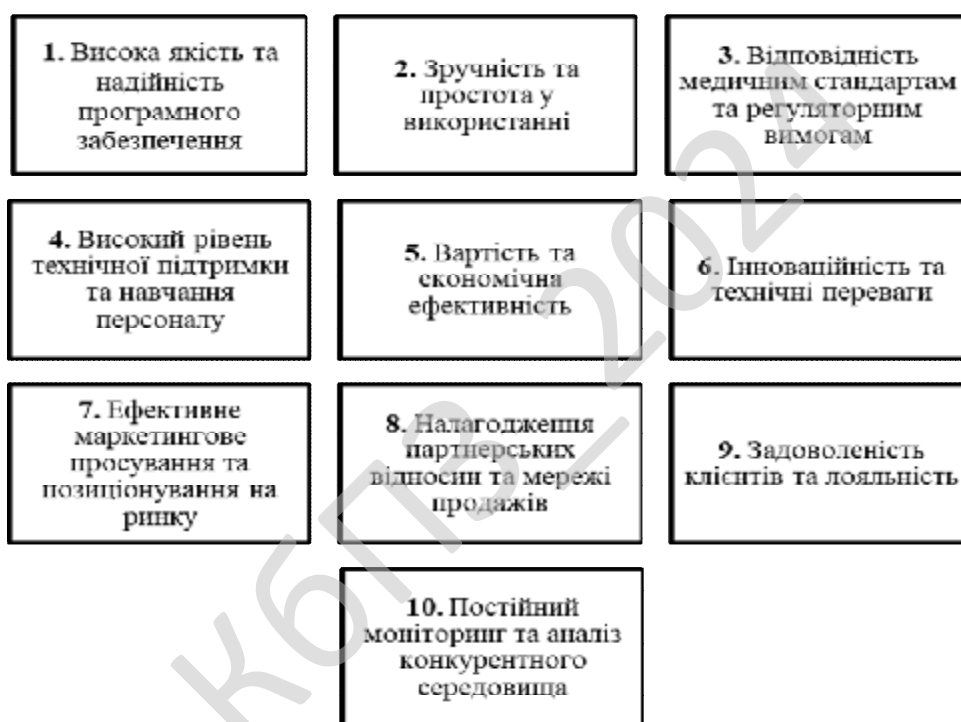


Рисунок 7.4 – Ключові фактори успіху проєкту

Успіх проєкту програмної реалізації системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів залежить від високої якості та надійності програмного забезпечення, зручності використання, відповідності медичним стандартам, надійної технічної підтримки, конкурентної ціни, інноваційності та ефективного маркетингового просування. Постійний моніторинг ринку та адаптація до потреб користувачів також відіграють важливу роль у забезпеченні довгострокового успіху.

## 8 ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ

### 8.1 Вступ

Аналізуючи умови працівників іт-сфери, на перший погляд, може здатися, що працівники сфери інформаційних технологій не схильні до ризиків на виробництві, та якщо більш глибоко розглянути умови і специфіку праці фахівців сфері іт-індустрії, можна виявити ряд факторів які будуть мати негативний вплив на стан охорони праці, та на самого іт-фахівця зокрема. Сюди можна віднести як невідповідність освітлення, так і високий рівень шуму, що негативно позначатимуться як на емоційному так і на фізичному стані фахівця, призводитимуть до зниження ефективності праці та виробничих травм. Також, важливим моментом охорони праці іт-фахівця є врахування його психологічних можливостей (швидкість реакції, особливості пам'яті та уваги, емоційний стан, тощо). Для того, щоб забезпечити ефективну роботу іт-фахівця, потрібно врахувати та максимально компенсувати такі негативні фактори як: надмірне нервово-емоційне навантаження, довготривалі статичні перевантаження, обмежена рухова активність. Всі ці чинники призводить до різноманітних відхилень у стані здоров'я, зокрема до перевтоми, зниження фізичної та розумової працездатності, неврозів, захворювань серцево-судинної системи тощо. Метою даного розділу є огляд конкретних умов праці спеціаліста у сфері іт-індустрії. Завданнями для даного розділу є: аналіз умов праці на робочому місці фахівця іт-індустрії, розробка конкретних рекомендацій щодо покращення умов праці фахівців іт-індустрії, огляд пожежної безпеки на іт-підприємстві та розрахунок системи загального штучного освітлення виробничого приміщення де працюють ІТ – фахівці.

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		71

## 8.2 Аналіз умов праці на робочому місці ІТ-фахівця

На робочому місці ІТ-фахівця (або програміста) виникають небезпечні та шкідливі для безпечної життєдіяльності фактори:

- підвищений рівень шуму;
- несприятливі мікрокліматичні умови;
- недостатній рівень освітленості;
- шкідливі речовини;
- підвищений рівень електромагнітних випромінювань радіочастот;
- висока напруга електричної мережі;
- статична електрика та інші.

Робота програміста супроводжується також підвищеним ступенем напруженості трудового процесу. При систематичному впливі виробничих факторів, які не відповідають нормативним показникам, зростає рівень професійно зумовленої захворюваності працюючих та можуть виникнути професійні захворювання органів зору, руху, нервової системи. Таким чином, вивчення умов праці на робочому місці програміста є необхідною умовою запобігання негативних наслідків впливу небезпечних та шкідливих факторів. Робоче місце, добре пристосоване до трудової діяльності інженера, правильно і доцільно організоване, щодо простору, форми, розміру забезпечує йому зручне положення при роботі і високу продуктивність праці при найменшому фізичному і психічному напруженні.

Нормування параметрів проводиться в залежності від періоду року та категорії важкості виконуваних робіт. Для постійних робочих місць, якими є робочі місця ІТ-фахівців, встановлені оптимальні параметри мікроклімату, а за неможливості їх дотримання використовують допустимі параметри. Робота ІТ-фахівця за важкістю відноситься до Іа (роботи, що виконуються сидячи і не потребують фізичного напруження) та Іб (роботи, що виконуються сидячи, стоячи або пов'язані з ходінням та супроводжуються деяким фізичним напруженням) категорій. В таблиці 8.1. наведені оптимальні параметри

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		72

мікроклімату в приміщеннях.

Таблиця 8.1 – Параметри мікроклімату для приміщень з ПК

Період року	Параметр мікроклімату	Величина
Холодний	Температура повітря в приміщенні; вологість; швидкість руху повітря	22...24°C; 40... 60%; до 0,1 м/с
Теплий	Температура повітря в приміщенні; вологість; швидкість руху повітря	23...25 °С 40...60% 0,1...0,2 м/с

Виміряні за допомогою приладів температура та вологість у приміщеннях праці ІТ-фахівців повинні відповідати зазначеним у таблиці для теплового періоду року. Слід зазначити, що для нормалізації параметрів мікроклімату слід використовувати у приміщеннях кондиціонування повітря, або забезпечити подачу свіжого повітря системами вентиляції. Норми подачі свіжого повітря наведені у таблиці 8.2.

Таблиця 8.2 – Норми подачі свіжого повітря в приміщення

Характеристика приміщення	Об'ємна витрата свіжого повітря, що подається в приміщення, м <sup>3</sup> на одну людину в годину
Об'єм до 20 м <sup>3</sup> на людину	Не менше 30
20... 40 м <sup>3</sup> на людину	Не менше 20
Більше 40 м <sup>3</sup> на людину	Може біти використана природна вентиляція

Створення сприятливих умов праці і правильне естетичне оформлення робочих місць на виробництві має велике значення як для полегшення праці, так

і для підвищення його привабливості, позитивно впливає на продуктивність праці. Забарвлення приміщень і меблів повинні сприяти створенню сприятливих умов для зорового сприйняття, гарного настрою. У службових приміщеннях, у яких виконується одноманітна розумова робота, що вимагає значної нервової напруги і великого зосередження, забарвлення повинно бути спокійних тонів – малонасичені відтінки холодного зеленого або блакитного кольорів.

При розробці оптимальних умов праці програміста необхідно враховувати освітленість. Раціональне освітлення робочого місця є одним з найважливіших факторів, що впливають на ефективність трудової діяльності людини, що попереджають травматизм і професійні захворювання. Правильно організоване освітлення створює сприятливі умови праці, підвищує працездатність і продуктивність праці. Освітлення на робочому місці програміста повинно бути таким, щоб працівник міг без напруги зору виконувати свою роботу. Стомлюваність органів зору залежить від ряду причин: недостатність освітленості; надмірна освітленість; неправильний напрям світла. Недостатність освітлення приводить до напруги зору, ослабляє увагу, приводить до настання передчасної стомленості. Надмірно яскраве освітлення викликає засліплення, роздратування і різь в очах. Неправильний напрямок світла на робочому місці може створювати різкі тіні, відблиски, дезорієнтувати працюючого. Всі ці причини можуть призвести до нещасного випадку або профзахворювань. [2]

### **8.3 Пропозиції щодо підвищення працездатності ІТ – фахівців**

Поява та впровадження нових інформаційно-комунікаційних технологій зумовлює необхідність подальшого вдосконалення охорони праці фахівців іт-індустрії. Все це потребує розробки нових нормативно-правових актів з регламентації праці та відпочинку фахівців іт-індустрії і стандартів підприємств, центрів комп'ютерної техніки, центрів інформаційних технологій, сучасних комп'ютерних класів. Для підвищення розумової працездатності то зорової роботи повинна здійснюватися ергономічна оптимізація в рамках системи

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		74

«оператор-термінал», яка сприятиме результативній фізичній та інтелектуальній працездатності і відновленню психосоматичного здоров'я фахівців it-індустрії. Всі наведені заходи щодо вдосконалення охорони праці фахівців it-індустрії повинні контролюватися службою охорони праці та комісією з охорони праці підприємства. Особливе значення у соціальному захисті цієї категорії працівників належить прийняття комплексного договору, який може забезпечити фахівців додатковими пільгами та компенсаціями.

Для більшого розуміння, пропозиції щодо підвищення працездатності it-фахівців, розіб'ємо на декілька категорій:

1 Середовище і розпорядок праці. Для мінімізації негативних ефектів, що пов'язані з перевтомленням it-фахівців, потрібно чітко прописати і реалізувати графік періодів праці-відпочинку, щоб фахівець міг можливість переключити увагу, дати можливість відпочити очам, мозку, елементарно, встати розім'яти ноги. Також потрібно зробити максимально комфортними умови мікроклімату у офісному приміщенні, де працюють it-фахівці. Мається на увазі встановлення і експлуатація, коли виникає необхідність, кондиціонерів, опалення, та системи вентиляції, задля попередження перегрівання, переохолодження it-фахівців, і подальшої неможливості ними виконувати свої функції. Також, за можливості, нами пропонується введення практики віддаленої праці it-фахівцями, якщо роботодавець не може забезпечити оптимальні і безпечні умови в офісному приміщенні, або якщо фахівця вони не влаштовують із певних причин.

2 Фізичні і психоемоційні чинники. Першим і найважливішим чинником, що впливає на працездатність it-фахівців є робоче місце, і саме тому, роботодавець має забезпечити максимальний його комфорт і безпеку. Гарантією цих факторів може слугувати сертифікація меблів, що використовуються на підприємстві it-галузі. Тому нами пропонується закупівля тільки меблів, які пошли сертифікацію на відповідність. Під психоемоційними чинниками ми розуміємо гарне самопочуття фахівців, позитивний настрій, гарний психологічний клімат у колективі, тощо. Задля того, щоб психоемоційні чинники мали

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		75

максимально позитивний ефект, керівництву слід поводити заходи, які сприятимуть укріпленню і покращенню міжособистісних стосунків у колективі, таких як психологічні тренінги, тимбілдінг, спортивні змагання і естафети. Також, сюди можна віднести розробку і впровадження системи мотивації працівників, як фінансової, так моральної і адміністративної.

#### **8.4 Розрахунок системи загального штучного освітлення виробничого приміщення де працюють ІТ-фахівці**

Приміщення з ПК повинні мати природне і штучне освітлення, яке відповідало б вимогам ДБН В.2.5-28-2006 «Природне і штучне освітлення», ДСАНПН 3.3.2.007-98 «Гігієнічні вимоги до організації роботи з візуальними дисплейними терміналами електронно-обчислювальних машин». Приміщення для роботи із ПЕОМ повинні мати природне й штучне освітлення. Віконні прорізи повинні бути орієнтовані на північ або на північний схід, забезпечувати коефіцієнт природної освітленості (К.П.О.) не менш 1,5% і мати жалюзі або штори. Віконні прорізи повинні мати регульовані пристрої для відкривання, а також жалюзі, завіски, зовнішні козирки тощо. Приміщення із ПЕОМ повинні бути обладнані системою загального рівномірного освітлення. У виробничих і адміністративно-суспільних 130 приміщеннях, де переважно ведеться робота з документами, допускається комбінована система штучного освітлення. Штучне освітлення має здійснюватися системою загального рівномірного освітлення, яка включає суцільні або такі, що перериваються лінії світильників, розташованих збоку робочих місць (переважно ліворуч), паралельно лінії зору користувачів ПК. Світильники повинні мати розсіювачі світла. У світильниках місцевого освітлення можна використовувати лампи накаливання. При розміщенні ПК по периметру приміщення лінії світильників штучного освітлення повинні розміщуватися локально над робочими місцями. Система освітлення робочого місця користувача ПК має відповідати наступним вимогам (рис. 8.1).

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		76



довжина – 7 м, висота – 2,9 м. У зазначеному приміщенні працює 7 людей.

Для того, щоб визначити потрібну кількість світильників, які повинні забезпечити нормований рівень освітленості, визначимо світловий потік, що падає на робочу поверхню за формулою [1]:

$$F=ESKZ/n,$$

де:

$F$  – світловий потік, що розраховується, Лм;

$E$  – нормована мінімальна освітленість, Лк;  $E = 300$  Лк;

$S$  – площа освітлюваного приміщення (у нашому випадку  $S=6 \times 7 = 42$  м<sup>2</sup>);

$K$  – коефіцієнт запасу, що враховує зменшення світлового потоку лампи в результаті забруднення світильників в процесі експлуатації (його значення залежить від типу приміщення і характеру робіт, що проводяться в ньому, в нашому випадку  $K = 1,5$ );

$Z$  – відношення середньої освітленості до мінімальної (зазвичай приймається рівним 1.1... 1.2, в нашому випадку  $Z = 1,1$ );

$n$  – коефіцієнт використання світлового потоку, (відношення світлового потоку, що падає на розрахункову поверхню, до сумарного потоку всіх ламп і обчислюється в долях одиниці [8]; залежить від характеристик світильника, розмірів приміщення, забарвлення стін і стелі, що характеризуються коефіцієнтами відбиття від стін ( $\rho_{стін}$ ) і стелі ( $\rho_{стелі}$ ), значення коефіцієнтів дорівнюють  $\rho_{стін} = 50\%$  і  $\rho_{стелі} = 50\%$ .

Обчислимо індекс приміщення за формулою:

$$i=S/(h(A+B)),$$

де:

$S$  – площа приміщення,  $S = 42$  м<sup>2</sup>;

$h$  – розрахункова висота підвісу,  $h = 2,9$  м (співпадає з висотою стелі, т.я. лампи освітлення закріплюються на стелі);

$A$  – ширина приміщення,  $A = 6$  м;

$B$  – довжина приміщення,  $B = 7$  м.

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		78

Підставимо всі значення у формулу та визначемо індекса приміщення:

$$i=1,4.$$

Знаючи індекс приміщення, за знаходимо  $n = 0,29$  (з табличних даних коефіцієнтів використання світлового потоку ( $n$ ) світильників з відповідним типом лампам) [8]. Підставимо всі значення у формулу, визначемо світловий потік:  $F=71689$  Лм.

Для розрахунку дудемо використовувати *світлодіодні стельові панелі Delux LED Panel 41 44Вт.*, світловий потік яких  $F_{л} = 3600$  Лм.

Число ламп визначається по формулі:

$$N=F/F_{л}$$

де:

$F$  – світловий потік,

$F_{л}$  – світловий потік однієї лампи.

Підставимо всі значення у формулу та визначемо індекса приміщення:  $N=$

$$71689 / 3600=19,9 \text{ шт.}$$

Приймаємо необхідну кількість *світлодіодних світильників* 20 шт.

## 8.5 Висновки до розділу

Дотримання всіх необхідних умов праці не лише сприяє збереженню здоров'я працівників, а також підвищує ефективність виробництва в цілому.

З цих міркувань було здійснено аналіз умов праці на робочому місці ІТ-фахівця, призначеного для праці програмістів, проведено розгляд небезпечних та шкідливих факторів, що негативно впливають на програмістів під час роботи. Виконано розрахунок штучного освітлення, як одного з ключових факторів впливу на працездатність та здоров'я програміста. Розроблено заходи з умов поліпшення охорони праці.

					ВКРМ-123.24.0019.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		79

## 9 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти, призначено для системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів.

В межах України в недостатній мірі представлені вітчизняні розробки в цій області.

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти наведені теоретичне узагальнення й рішення наукового завдання дослідження методи та засоби реєстрації та запису сигналів ЕКГ.

Рішення даного завдання полягало у вирішенні наступних задач:

- Проведено огляд існуючих систем реєстрації та запису сигналів ЕКГ.
- Досліджено системи реєстрації та запису сигналів ЕКГ.
- Виконано програмну реалізацію системи збору інформації на базі FLESH накопичувачів.

– На основі отриманих результатів досліджень створена програмна реалізація системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів.

Розроблені під час виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання системи обробки сигналів ЕКГ та збору інформації на базі флеш накопичувачів.

Проведено аналіз предметної галузі в ході якого були виявлені об'єкти, взаємодія яких носить істотний характер для функціональної діяльності предметної галузі, і їхні основні характеристики; побудована алгоритм і вибраний середовище розробки.

Розроблене програмне забезпечення має простий, дружній та зручний інтерфейс користувача, що забезпечує легкість у освоєнні роботи програмного

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		80

продукту, зручність у використанні, і не потребує особливих спеціальних знань.

При створенні програмного забезпечення було використано об'єктно-орієнтований підхід, що відповідає сучасним тенденціям у галузі розробки комерційних програмних систем.

Програма реалізована на мові Assembler та C++. Дана мова програмування дозволяє найбільш ефективно обробляти дані такого типу та працювати з обраним мікроконтроллером. Це дозволило мінімізувати строк розробки програмного забезпечення, і, як слід, зменшити витрати на його розробку. Запропоноване програмне забезпечення ділиться на загальне програмне забезпечення, що поставляється із засобами обчислювальної техніки й спеціальне програмне забезпечення, що спеціально розроблене для даної конкретної системи й включає програми, що реалізують її функції.

Програма призначена для виконання під управлінням багатозадачної операційної системи Windows 10/11.

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

Для підвищення рівня безпеки запропоновано застосовувати алгоритм шифрування.

В цілому створене програмне забезпечення підтверджує правильність використаних проектних рішень та повністю відповідає вимогам технічного завдання. Створене програмне забезпечення має потенційну можливість для подальшого вдосконалення і застосування у різних галузях.

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		81

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Бабенко М.В. Автоматизований кардіомонітор/ М.В. Бабенко, О.М. Павловський// Збірник праць XIV Всеукраїнської науково-практичної конференції студентів, аспірантів та молодих вчених “ПОГЛЯД У МАЙБУТНЄ ПРИЛАДОБУДУВАННЯ”, 18-19 травня 2021р. - К.: ПБФ, КПІ ім. Ігоря Сікорського. – 2021. – 16-18 с.
2. Електроміографія - [Електронний ресурс] - Режим доступу: <http://medbib.in.ua/elektromiografiya.html>
3. Холтер ЕКГ - [Електронний ресурс] - Режим доступу: <http://holter.com.ua/patient>
4. Digital Signal Processing: Principles, Algorithms, and Applications / John G. Proakis, Dimitris K. Manolakis - Pearson Education. 2021. – 1032 стор.
5. Biomedical Signal Processing and Signal Modeling / Eugene N. Bruce - Wiley-Interscience. 2001. – 400 стор.
6. Electrocardiography in Practice. John R. Hampton – Elsevier. 2019. – 376 стор.
7. Літовченко О.В. Використання цифрових фільтрів для ЕКГ. - Донецьк: Вид-во ДонНУ, 2021
8. Якимов В.П. Проблеми і перспективи електрокардіографічної діагностики. – Харків: Наукова думка, 2020.
9. Коваленко О.М., Механізми зниження діагностичних помилок у кардіології. – Київ: Медична академія, 2019.
10. Петров А.Н. Автоматизація діагностичних систем в кардіології. – Одеса: ОНУ, 2021.
11. Johnson J. ECG Signal Processing and Filtering. - New York: Springer, 2018.
12. Smith J. Cardiology Signal Analysis Methods. - London: Wiley, 2019.
13. Ivanov I. et al. “Digital Filters in ECG Signal Processing”, Journal of Cardiology Research, 2022.

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		82



доступу: [http://ua-referat.com/Захист\\_програмного\\_забезпечення](http://ua-referat.com/Захист_програмного_забезпечення)

27. Сайт фірми Brain [Електронний ресурс] - Режим доступу: <http://brain.com.ua>

28. Мелешко Є.В., Якименко М.С., Поліщук Л.І. Алгоритми та структури даних: Навчальний посібник для студентів технічних спеціальностей денної та заочної форми навчання. – Кропивницький: Видавець – Лисенко В.Ф., 2019. – 156 стор.

29. Власій О.О. Алгоритми та структури даних: Лабораторний практикум. – Івано-Франківськ: ДВНЗ «Прикарпатський національний університет імені Василя Стефаника», 2015. – 68 с.

30. Захист програм від злому [Електронний ресурс] - <http://easy-code.com.ua/2011/04/zaxist-program-vid-zlomu/>

31. Alexander Osterwalder, Yves Pigneur – Business Model Generation. Wiley, 2010. – 288 стор.

32. Steve Blank, Bob Dorf – The Startup Owner's Manual: The Step-by-Step Guide for Building a Great Company. K&S Ranch, 2012. – 608 стор.

33. Philip Kotler – Marketing Management (15th Edition). Pearson, 2015. – 832 стор.

34. Kenneth C. Laudon, Jane P. Laudon – Management Information Systems: Managing the Digital Firm (16th Edition). Pearson, 2020. – 688 стор.

35. Жидецький В.Ц. Основи охорони праці: підруч. 3-є вид., перероб і доп. Львів : УАД, 2006. 336 стор.

36. Босов Є.П., Жесан Р.В., Каліч В.М., Голик О.П., Зубенко В.О. Охорона праці при проектуванні систем автоматизації виробництва : навч. посіб. 2-е вид., перероб. і доп. Кропивницький : ЦНТУ, 2022. – 208 стор.

37. Конституція України. [Електронний ресурс] / Режим доступу: <https://zakon.rada.gov.ua/laws/main/254%D0%BA/96-%D0%B2%D1%80>

38. Про охорону праці : Закон України. [Електронний ресурс] / Режим доступу: <https://zakon.rada.gov.ua/laws/main/2694-12#Text>.

39. Основи законодавства України про охорону здоров'я : Закон України.

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		84

[Електронний ресурс] / Режим доступу: <https://zakon.rada.gov.ua/laws/show/2801-12#Text>.

40. Закон України «Про захист інформації в інформаційно-телекомунікаційних системах» [Електронний ресурс]. – Режим доступу до ресурсу: <http://zakon4.rada.gov.ua/laws/show/2594-15>

41. Закон України «Про охорону праці» від 14.10.1992 р. № 2694-ХІІ. – Режим доступу до ресурсу: <https://zakon.rada.gov.ua/laws/show/2694-12>

42. Зеркалов Д. В. Охорона праці в Галузі: Загальні вимоги: навч. посіб. Київ: Основа. 2011. 551 с.

43. Наказ Міністерства соціальної політики України 14.02.2018 № 207 «Про затвердження Вимог щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями». – Режим доступу до ресурсу: <https://zakon.rada.gov.ua/laws/show/z0508>

44. Охорона праці. Ч. 1. Захисне заземлення: метод. вказ. до викон. розрахунків з викор. персон. ЕОМ ІВМ сумісного типу / Кіровоград. ін-т с.-г. машинобуд.; [укл. О. В. Оришака, Є. К. Солових, В. О. Оришака]. – Кіровоград: КІСМ, 1997. – 20 с. Режим доступу до ресурсу: <http://dspace.kntu.kr.ua/jspui/handle/123456789/4358>

45. Постанова № 42 від 01.12.1999 Головного державного санітарного лікаря України «Санітарні норми мікроклімату виробничих приміщень ДСН 3.3.6.042-99. – Режим доступу до ресурсу: <https://zakon.rada.gov.ua/rada/show/va042282-99>

46. Центр післядипломної освіти та підвищення кваліфікації. – Режим доступу до ресурсу: <https://cpo.stu.cn.ua>

47. Оришака, О. В. Основи охорони праці: навч. посіб. / О. В. Оришака, Г. П. Горбачова, К. М. Марченко; М-во освіти і науки України, Центральноукраїн. нац. техн. ун-т. – Кропивницький : ЦНТУ, 2022. - 175 с. – Режим доступу до ресурсу: <http://dspace.kntu.kr.ua/jspui/handle/123456789/12161>.

48. Захист програмного забезпечення [Електронний ресурс] - Режим доступу: [http://ua-referat.com/Захист програмного забезпечення](http://ua-referat.com/Захист_програмного_забезпечення)

					<b>ВКРМ-123.24.0019.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		85

49. Захист програм від злому [Електронний ресурс] - Режим доступу: <http://easy-code.com.ua/2011/04/zaxist-program-vid-zlomu/>

50. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин: ДСанПІН 3.3.2-007-98. – Режим доступу до ресурсу: <https://zakon.rada.gov.ua/rada/show/v0007282-98>.

КБПЗ – 2024

					ВКРМ-123.24.0019.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		86

Додаток А  
(обов'язковий)

Технічне завдання

Зміст

1 Найменування та область застосування.....	2
2 Підстава для розробки.....	2
3 Мета та призначення розробки.....	2
4 Джерела розробки.....	2
5 Технічні вимоги.....	2
5.1 Вміст проекту.....	2
5.2 Показники призначення.....	3
5.3 Вимоги до функціональних характеристик.....	3
5.4 Вимоги до архітектури.....	3
5.5 Вимоги до надійності.....	3
5.6 Умови експлуатації.....	4
5.7 Вимоги до складу та параметрів технічних засобів.....	4
5.8 Вимоги до інформаційної і програмної сумісності.....	4
5.8.1 Обладнання.....	4
5.8.2 Мова програмування.....	4
5.8.3 Вхідні дані.....	5
5.8.4 Вихідні дані.....	5
6 Вимоги до програмної документації.....	5
7 Економічні вимоги.....	5
8 Вимоги щодо охорони праці.....	5
9 Перелік документів, що розробляються.....	6
10 Етапи розробки.....	6
11 Порядок контролю та приймання.....	6

					<b>ВКРМ-123.24.0019.00.00.ТЗ</b>			
Вим.	Арк.	№ документа	Підпис	Дата				
Розробив	Литвиненко Д.О.				Дослідження та програмна реалізація системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів	Літ.	Аркуш	Аркушів
Перевірів	Кислун О.А.					М	1	6
Н. Контр.	Коваленко А.С				ЦНТУ КІ-23М			
Затв.	Смірнов О.А.							



- техніко-економічне обґрунтування доцільності прийнятого до розробки програмного забезпечення;
- аналіз умов праці;
- розробка програми, що реалізує спроектовані алгоритми роботи системи.

## 5.2 Показники призначення

Система повинна забезпечувати:

- програмну реалізацію системи діагностики, обробки та запису сигналів ЕКГ на базі флеш-накопичувачів;
- цілісність даних у процесі роботи, обробки та при зберіганні;
- простий, інтуїтивно зрозумілий інтерфейс.

## 5.3 Вимоги до функціональних характеристик

Розроблене програмне забезпечення не повинно мати обмежень на версію драйверів та операційної системи.

## 5.4 Вимоги до архітектури

Компонент, що розробляється повинен використовувати системні засоби та апаратні засоби, що на даному етапі розвитку обчислювальної техніки найбільше поширені.

## 5.5 Вимоги до надійності

Програмні модулі написані по всім правилам, які стосуються стандартних викликів процедур, функцій, методів і форм, визначених технічною документацією на середовище розробки.

					<b>ВКРМ-123.24.0019.00.00.ТЗ</b>	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		3

## 5.6 Умови експлуатації

Робочі місця користувачів ПЗ повинні задовольняти наступним умовам експлуатації:

- температура повітря: 19-20 град. по Цельсію;
- відносна вологість повітря до 80%;
- атмосферний тиск 107 кПа.

## 5.7 Вимоги до складу та параметрів технічних засобів

Програмне забезпечення повинно бути реалізоване на ПЕОМ архітектури IBM PC, працювати в ОС Windows 10/11 і з сумісними з цією платформою пристроями і прикладним програмним забезпеченням.

## 5.8 Вимоги до інформаційної і програмної сумісності

Переносність програмного забезпечення повинна бути забезпечена за рахунок його реалізації стандартного інтерфейсу взаємодії з ОС, що працюють під управлінням ОС Windows 10/11.

### 5.8.1 Обладнання

Комп'ютер Intel® Celeron/8 Mb/1.2 Gb/SVGA 14" 1Mb або сумісні з ним.

### 5.8.2 Мова програмування

Середовище AVR Studio, Assembler, C++.

					ВКРМ-123.24.0019.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		2

### 5.8.3 Вхідні дані

Опис алгоритму роботи запропонованої системи.

### 5.8.4 Вихідні дані

Робоча програма.

## 6 Вимоги до програмної документації

Програмна продукція повинна бути представлена у виді опису структури даних, схем та опису алгоритму, а також текстів вихідних модулів програмного забезпечення згідно ЄСПД .

## 7 Економічні вимоги

7.1 Для ПЗ необхідно виробити функціонально-вартісний аналіз варіантів розробки.

7.2 Виконати розрахунок витрат показників економічного ефекту з урахуванням цін на 3 вересня 2024 року.

## 8 Вимоги щодо охорони праці

В частині охорони праці випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти повинен бути розглянутий аналіз умов праці на робочому місці ІТ-фахівця.

					ВКРМ-123.24.0019.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		5

## 9 Перелік документів, що розробляються

- Наукова новизна – 1 аркуш.
- Структурна схема системи – 1 аркуш.
- Функціональна схема системи – 1 аркуш.
- Діаграма процесів – 1 аркуш.
- Блок-схема алгоритму роботи програми – 2 аркуша.
- Маркетингове та економічне обґрунтування ІТ проекту – 1 аркуш.
- Пояснювальна записка – 87 аркушів.

## 10 Етапи розробки

10.1 Збір і обробка інформації по темі випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти. Постановка задачі на виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти (складання ТЗ).

10.2 Проведення досліджень або експериментальних робіт для уточнення основних положень випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти.

10.3 Розробка функціональних схем, блок схем алгоритмів роботи програмного забезпечення.

10.4 Побудова схем взаємодії даних.

10.5 Створення прототипу ПЗ.

10.6 Віднаходження ПЗ, аналіз отриманих результатів.

10.7 Робота над питанням охорони праці і техніки безпеки.

10.8 Розрахунок з техніко-економічного обґрунтування.

10.9 Оформлення пояснювальної записки і виконання робіт по графічній частині.

## 11 Порядок контролю та приймання

11.1 Подання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти на попередній захист 02.12.2024 р.

11.2 Подання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти на захист \_\_.\_\_.2024 р.

					<b>ВКРМ-123.24.0019.00.00.ТЗ</b>	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		6

Додаток Б  
(обов'язковий)

**Міністерство освіти і науки України**  
**Центральноукраїнський національний технічний університет**

**ЗАТВЕРДЖУЮ**

Керівник випускної кваліфікаційної роботи за  
другим (магістерським) рівнем вищої освіти  
\_\_\_\_\_ Кислун О.А.

Дослідження та програмна реалізація системи діагностики, обробки  
та запису сигналів ЕКГ на базі флеш-накопичувачів

Лістинг програми

Код документу 12

Носій: CD/DVD-диск / USB-флеш-накопичувач

Загальна кількість аркушів: 35

Літера: РП

Кропивницький – 2024 року

## Головний модуль

```

#include p16f873a.inc

; _BODEN_OFF - детектор зниженої напруги (Піт<4В)
; _PWRTE_ON - затримка включення
; _WDT_ON - сторожовий таймер

__CONFIG _CP_OFF & _DEBUG_OFF & _WRT_OFF & _CPD_OFF & _LVP_OFF &
_BODEN_OFF & _PWRTE_ON & _WDT_ON & _HS_OSC

RADIX dec

EXPAND

ERRORLEVEL 0, -305, -302

; *****
*****
; На початку (перші 256 інструкцій) розміщено програму МОНИТОР
(стартер), яка дозволяє
; заливати нові прошивки по RS-232, а також надає ряд простих та
потрібних функцій
; по роботі з RS-232 та доступом до флеш пам'яті даних та програми
;
; Протокол обміну з МОНИТОРОМ:
; Обмін на швидкості 57.6 Кбіт/сек для кварцу 10 МГц
; Якщо протягом 0.1 с не прийнятий байт 0xcd - запуск програми
користувача
; Прийом байта 0xcd - відкласти запуск програми користувача до
особливих вказівок
; 0xcd, 0xe6, AdrLo, AdrHi - читання ROM програми
; 0xcd, 0xe7, AdrLo, AdrHi, DataLo, DataHi - запис у ROM програм
; 0xcd, 0xe8, Adr - читання flash даних
; 0xcd, 0xe9, Adr, Data - запис flash даних
; 0xcd, 0xeb - запуск програми користувача

cblock 0x20
; змінні програми
; ці змінні після виходу з монітора зовсім не використовуються
; (Не використовуються і функцією Monitor_DetectMemoryCmd)
OutByte
CanCont
endc

org 0x0
;
; точка входу в монітор
;
STARTER_ENTRY_POINT:
clrf INTCON
clrf STATUS
clrf PCLATH
goto STARTER_ENTRY_POINT_CONTINUE
;goto USER_CODE_ENTRY_POINT
;
;
; обробник переривання
org 4
goto USER_CODE_IRQ
goto $
goto $
goto $
;
;
; ТАБЛИЦЯ СТАНДАРТНИХ ФУНКЦІЙ МОНИТОРА
; параметри кожної функції дивись на початку реалізації кожної
функції

```

```

;
org 8
;
Usart_Init: goto __Usart_Init
Usart_In_WDT: clrwdt
Usart_In: goto __Usart_In
Usart_Out: goto __Usart_Out
Mem_ReadROM: goto __Mem_ReadROM
Mem_ReadFlash: goto __Mem_ReadFlash
Mem_WriteFlash_Wait: goto __Mem_WriteFlash_Wait
Mem_WriteFlash_Write: goto __Mem_WriteFlash_Write
Monitor_DetectMemoryCmd: goto __Monitor_DetectMemoryCmd
;
;
STARTER_ENTRY_POINT_CONTINUE:
;
; конфігуруємо порти на введення
bsf STATUS, RP0
movlw b'11111111'
movwf TRISA
movwf TRISB
movwf TRISC
movlw b'11001101'; викл.рез., WDT справ 1:32, це 0.3 - 0.6 - 1.2 сек
movwf OPTION_REG
bcf STATUS, RP0
;
; вимикаємо будь-яку апаратуру, що стосується виходів
clrf T1CON
clrf T2CON
clrf CCP1CON
clrf CCP2CON
clrf SSPCON
;
; включення Usart
call Usart_Init
;
clrf CanCont
movlw 0x55
movwf OutByte
;
MainLoop:
movlw 200
call Usart_In_WDT
btfsc STATUS, C
goto MainLoop_NoData
sublw 0xcd
btfsc STATUS, Z
goto MaimLoop_CMD
sublw 0xcd
call Usart_Out
goto MainLoop
;
MainLoop_NoData:
btfss CanCont, 0
goto User_Start
movf OutByte, w
incf OutByte
call Usart_Out
goto MainLoop
;
User_Start:
movlw 0xbb
call Usart_Out
clrwdt
goto USER_CODE_ENTRY_POINT
;
;
; Спроба прийняти та декодувати команду
MaimLoop_CMD:

```

```

    bsf CanCont, 0
    call MainLoop_GetByte
    btfsc STATUS, C
    goto MainLoop
    movwf FSR
    sublw 0xeb
    btfsc STATUS, Z
    goto User_Start
    movf FSR, w
    call Monitor_DetectMemoryCmd
    goto MainLoop
    ;
    ;
    ; очікування 0.1 сек байта з Usart з попереднім обнуленням банку
MainLoop_GetByte:
    bcf STATUS, RP0
    bcf STATUS, RP1
    movlw 200
    goto Usart_In_WDT
    ;
    ; Декодування команди Монітора для доступу до пам'яті
    ; на вході: W - код команди та обраний нульовий банк
    ; якщо це код команди, виконуються її дії з обміном за Usart
    ; руйнується FSR
    ;
__Monitor_DetectMemoryCmd:
    movwf FSR
    movlw 0xe6
    subwf FSR, w
    btfsc STATUS, Z
    goto MaimLoop_CMD_ReadROM
    movlw 0xe7
    subwf FSR, w
    btfsc STATUS, Z
    goto MaimLoop_CMD_WriteROM
    movlw 0xe8
    subwf FSR, w
    btfsc STATUS, Z
    goto MaimLoop_CMD_ReadFlash
    movlw 0xe9
    subwf FSR, w
    btfsc STATUS, Z
    goto MaimLoop_CMD_WriteFlash
    return
    ;
    ;
MaimLoop_CMD_ReadROM:
    call MainLoop_GetByte
    btfsc STATUS, C
    return
    bsf STATUS, RP1
    movwf EEADR
    call MainLoop_GetByte
    btfsc STATUS, C
    return
    bsf STATUS, RP1
    movwf EEADRH
    ;
    call Mem_ReadROM
    ;
    call Usart_Out
    bsf STATUS, RP1
    movf EEDATH, w
    bcf STATUS, RP1
    goto Usart_Out
    ;
    ;
    ; Читання програмної пам'яті

```

```

; На вході EEADRH: EEADR (0x10f: 0x10d) - адреса осередку, банк -
будь-який
; На виході EEDATH::EEDATA (0x10e:0x10c) - прочитане значення
; та акумулятор W=EEDATA, банк на виході встановлюється нульовим
; __Mem_ReadROM:
bsf STATUS, RP1
bsf STATUS, RP0
bsf EECON1, EEPGD
bsf EECON1, RD
nop
nop
bcf STATUS, RP0
movf EEDATA, w
bcf STATUS, RP1
return
;
MaimLoop_CMD_ReadFlash:
call MainLoop_GetByte
btfsc STATUS, C
return
call Mem_ReadFlash
goto Usart_Out
;
;
; Читання flash - пам'яті
; На вході W - адреса осередку, банк - будь-який
; На виході EEDATA (0x10c) - прочитане значення
; та акумулятор W=EEDATA, банк на виході встановлюється нульовим
__Mem_ReadFlash:
bcf STATUS, RP0
bsf STATUS, RP1
movwf EEADR
Mem_ReadFlash_Proc: ; теж, але адреса осередку в EEADR (0x10d), банк -
будь-який
bsf STATUS, RP1
bsf STATUS, RP0
bcf EECON1, EEPGD
bsf EECON1, RD
bcf STATUS, RP0
movf EEDATA, w
bcf STATUS, RP1
return; на виході банк нульовий
;
MaimLoop_CMD_WriteROM:
call MainLoop_GetByte
btfsc STATUS, C
return
bsf STATUS, RP1
movwf EEADR
call MainLoop_GetByte
btfsc STATUS, C
return
bsf STATUS, RP1
movwf EEADRH
call MainLoop_GetByte
btfsc STATUS, C
return
bsf STATUS, RP1
movwf EEDATA
call MainLoop_GetByte
btfsc STATUS, C
return
bsf STATUS, RP1
movwf EEDATH
; перевірка адреси на допустимість
movf EEADRH, w
btfss STATUS, Z; Увага !!!, btfsc - запис тільки в монітор !!!
goto MaimLoop_CMD_WriteROM_WR
movf EEADR, w

```

```

    addlw -4
    sublw 3
    btfss STATUS, C
    goto MaimLoop_CMD_WriteIgnore
MaimLoop_CMD_WriteROM_WR:
    bsf STATUS, RP0
    bsf EECON1, EEPGD
    bsf EECON1, WREN
    movlw 0x55
    movwf EECON2
    movlw 0xaa
    movwf EECON2
    bsf EECON1, WR
    nop
    nop
    bcf EECON1, WREN
    bcf STATUS, RP0
    bcf STATUS, RP1
    ;
MaimLoop_CMD_WriteOK:
    movlw 0xAA
    goto Usart_Out
    ;
MaimLoop_CMD_WriteIgnore:
    bcf STATUS, RP0
    bcf STATUS, RP1
    movlw 0xEE
    goto Usart_Out
    ;
MaimLoop_CMD_WriteFlash:
    call MainLoop_GetByte
    btfsc STATUS, C
    return
    call Mem_WriteFlash_Wait
    bsf STATUS, RP1
    movwf EEADR
    call MainLoop_GetByte
    btfsc STATUS, C
    return
    bsf STATUS, RP1
    movwf EEDATA
    goto Mem_WriteFlash_Write
    ;
    ;
    ;
    ; Очікування закінчення запису у flash - пам'ять
    ; на вході - будь-який банк
    ; на виході - Нульовий банк, але більше нічого не руйнується
    ; необхідно викликати до запису в EEADR та EEDATA (здається)
    ;
__Mem_WriteFlash_Wait:
    bsf STATUS, RP0
    bsf STATUS, RP1
    btfsc EECON1, WR
    goto $-1
    bcf STATUS, RP0
    bcf STATUS, RP1
    return
    ;
    ; Запис у flash - пам'ять
    ; Перед викликом має стояти виклик Mem_WriteFlash_Wait
    ; На вході EEADR (0x10d) - адреса осередку,
    ; EEDATA (0x10c) - значення для запису
    ; банк - будь-який
    ; На виході Нульовий банк W руйнується, а переривання заборонені (!)
    ;
__Mem_WriteFlash_Write:
    bsf STATUS, RP0
    bsf STATUS, RP1

```

```

    bcf EECON1, EEPGD
    bsf EECON1, WREN
    bcf INTCON, GIE
    movlw 0x55
    movwf EECON2
    movlw 0xaa
    movwf EECON2
    bsf EECON1, WR
    bcf EECON1, WREN
    bcf STATUS, RP0
    bcf STATUS, RP1
    return
;
;
; налаштування стандартного режиму Usart
; на вході будь-який банк, на виході - нульовий
;
__Usart_Init:
    ; включаємо передавач USART
    bsf STATUS, RP0
    bcf STATUS, RP1
    clrf TXSTA
    movlw b'10100111'
    movwf TXSTA
    movlw 10
    movwf SPBRG
    bcf STATUS, RP0
    ;
    ; включаємо приймач USART (і передавач)
    clrf RCSTA
    movlw b'10010000'
    movwf RCSTA
    return
;
; Передати байт W. Нічого не руйнується
; Передбачається, що обрано нульовий банк
;
__Usart_Out:
    btfss PIR1, TXIF
    goto __Usart_Out
    movwf TXREG
    return
;
; Прийняти байт
; W - час тайм-ауту в одиницях по 0.5 мс
; вважається, що обрано нульовий банк
; Вихід: W - прийнятий байт, C==1, якщо була помилка або TimeOut
; Реєстр FSR руйнується, інші - ні!
__Usart_In:
    movwf FSR
Usart_In_L1:
    movlw 156
Usart_In_L2:
    btfss RCSTA, OERR
    goto Usart_In_noReset
    ;call Usart_In_Reset
    ;bcf STATUS, C
    ;btfsc RCSTA, FERR
    ;bsf STATUS, C
    ;movf RCREG, w
    bcf RCSTA, CREN
    bsf RCSTA, CREN
Usart_In_noReset:
    btfsc PIR1, RCIF
    goto Usart_In_End
    addlw 0xff
    btfss STATUS, Z
    goto Usart_In_L2
    decfsz FSR

```

```

    goto Usart_In_L1
    bsf STATUS, C
    return
;
Usart_In_End:
    bcf STATUS, C
    btfsc RCSTA, FERR
    bsf STATUS, C
    movf RCREG, w
    return
;
;
;
;
; *****
; ***** КОРИСТУВАЧИЙ КОД *****

Locked_H
    TMR1_B2
    TMR1_B3
;
    LockAct
;
    NewLock_T0: 4
    NewLock_S: 4
    NewLock_N: 4
;
    NewLockRes_T0: 4
    NewLockRes_S: 4
    NewLockRes_N: 4
;
;
;WAVE_WritePos
;WAVE_ReadPos
;WAVE_PlayValue
;
    XOUT:2
    YOUT:2
    ZOUT:2
    XYZN
;
    EmptyPos: 4
    EmptyPosCode: 4
;
;OldFile_StartAdr:4
    File_StartAdr:4
;File_WritedAdr:4
endc
;
; *****
*****
;
    cblock 0x20+0x80; змінні іншого банку
;
    WaitST__; резервовані для обробки переривань
    IntProc_Temp1__
    ACP_MemoryBuffer:0; буфер до адреси 0xff]
    WAVE_MemoryBuffer:0; а ще й буфер для гри WAVE
endc
;
;
; ; хитре макро додавання до 32-бітної змінної числа 512
;SADD512 macro Value
; local SADD512_EXIT
; movlw 2
; addwf Value+1
; btfss STATUS, C
; goto SADD512_EXIT

```

```

; incfsz Value+2
; goto SADD512_EXIT
; incfsz Value+3
; goto SADD512_EXIT
; incf Value+3; цей байт ненульовий
;SADD512_EXIT:
; endm
;
;
;
;
WAIT2 macro
    goto $+1
endm
;
WAIT3 macro
    nop
    goto $+1
endm
;
WAIT4 macro
    call WAIT4T
endm
;
WAIT5 macro
    call WAIT5T
endm
;
WAIT6 macro
    call WAIT6T
endm
;
; макро інкримент 32-бітної змінної
INC32M macro Value
    local INC32M_EXIT
    incfsz Value
    goto INC32M_EXIT
    incfsz Value+1
    goto INC32M_EXIT
    incfsz Value+2
    goto INC32M_EXIT
    incfsz Value+3
    nop
INC32M_EXIT:
    endm
;
; макро копіювання 32-бітної змінної з Src до Dest
MOVE32M macro Dest, Src
    movf Src, w; копіюємо кількість імпульсів
    movwf Dest
    movf Src+1, w
    movwf Dest+1
    movf Src+2, w
    movwf Dest+2
    movf Src+3, w
    movwf Dest+3
endm
;
; макро додавання 16-бітових змінних Arg1 = Arg1 + Arg2
ADD16M macro Arg1, Arg2
    movf Arg2, w
    addwf Arg1
;
    movf Arg2+1, w
    btfsc STATUS, C
    incfsz Arg2+1, w
    addwf Arg1+1
;
endm

```

```

;
; макро додавання байта зі знаком до 16-бітної змінної Arg1 = Arg1 +
(Arg2+128)
ADD16SigB macro Arg1, Arg2
    movf Arg2, w
    addlw 128
    addwf Arg1
    btfsc STATUS, C
    incf Arg1+1
endm
;
; макро обнулення 32-бітної змінної
CLR32M macro Value
    clrf Value
    clrf Value+1
    clrf Value+2
    clrf Value+3
endm
;
; макро обнулення 16-бітної змінної
CLR16M macro Value
    clrf Value
    clrf Value+1
endm
;
; макро додавання 32-бітових змінних Arg1 = Arg1 + Arg2
ADD32M macro Arg1, Arg2
    movf Arg2, w
    addwf Arg1
;
    movf Arg2+1, w
    btfsc STATUS, C
    incfsz Arg2+1, w
    addwf Arg1+1
;
    movf Arg2+2, w
    btfsc STATUS, C
    incfsz Arg2+2, w
    addwf Arg1+2
;
    movf Arg2+3, w
    btfsc STATUS, C
    incfsz Arg2+3, w
    addwf Arg1+3
;
endm
;
; макро віднімання 32-бітових змінних Arg1 = Arg1 - Arg2
SUB32M macro Arg1, Arg2
    movf Arg2, w
    subwf Arg1
;
    movf Arg2+1, w
    btfss STATUS, C
    incfsz Arg2+1, w
    subwf Arg1+1
;
    movf Arg2+2, w
    btfss STATUS, C
    incfsz Arg2+2, w
    subwf Arg1+2
;
    movf Arg2+3, w
    btfss STATUS, C
    incfsz Arg2+3, w
    subwf Arg1+3
;
endm

```

```

;
;
; порівняння 32-бітових змінних, прапори C та Z вуст. так, як для
Arg1 = Arg1 - Arg2
CMP32M macro Arg1, Arg2
    local CMP32M_EXIT
    movf Arg2+3, w
    subwf Arg1+3, w
    btfss STATUS, Z
    goto CMP32M_EXIT
    movf Arg2+2, w
    subwf Arg1+2, w
    btfss STATUS, Z
    goto CMP32M_EXIT
    movf Arg2+1, w
    subwf Arg1+1, w
    btfss STATUS, Z
    goto CMP32M_EXIT
    movf Arg2, w
    subwf Arg1, w
CMP32M_EXIT:
    endm
;
;
; інверсія 32-бітної змінної
COM32M macro Arg32
    comf Arg32
    comf Arg32+1
    comf Arg32+2
    comf Arg32+3
    endm
;
; виключне або з константою
XORL32M macro Arg32, ConstValue
    movlw (ConstValue)
    xorwf Arg32
    movlw (ConstValue)>>8
    xorwf Arg32+1
    movlw (ConstValue)>>16
    xorwf Arg32+2
    movlw (ConstValue)>>24
    xorwf Arg32+3
    endm
;
;
JC macro Label
    btfsc STATUS, C
    goto Label
    endm
;
JNC macro Label
    btfss STATUS, C
    goto Label
    endm
;
JZ macro Label
    btfsc STATUS, Z
    goto Label
    endm
;
JNZ macro Label
    btfss STATUS, Z
    goto Label
    endm
;
;
; ***** Специфічні (не універсальні) макроси для цієї програми
*****

```

```

;
; інкремент старшої частини та скидання прапора переповнення TMR1
IRQINCTMR1HI macro
    bcf PIR1, TMR1IF
    incfsz TMR1_B2
    goto $+2
    incf TMR1_B3
endm
;
;
;
; висновок на MMC та USART 16-бітного значення (дивись функцію
Stream_OutByte)
Stree

xorwf CodeValue+3, w
    JZ L3_NoCh
    xorwf CodeValue+3
    incf NChange
    movlw 3
    movwf FSR
L3_NoCh:
    rrf Value+2, w
    xorwf Value+2, w
    xorwf CodeValue+2, w
    JZ L2_NoCh
    xorwf CodeValue+2
    incf NChange
    movlw 2
    movwf FSR
L2_NoCh:
    rrf Value+1, w
    xorwf Value+1, w
    xorwf CodeValue+1, w
    JZ L1_NoCh
    xorwf CodeValue+1
    incf NChange
    movlw 1
    movwf FSR
L1_NoCh:
    rrf Value+0, w
    xorwf Value+0, w
    ;
    xorwf CodeValue+0; хитрий обмін змінних [CodeValue+0] <-> W
    xorwf CodeValue+0, w
    xorwf CodeValue+0
    ;
    xorwf CodeValue+0, w; та порівняння з використанням маски
FILE_GRANULA
    movf NChange
    btfss STATUS, Z
    andlw 0xff - (FILE_GRANULA-1); якщо мінялися старші байти, молодші
біти (FILE_GRANULA) ігноруємо
    andlw 0xff
    JZ L0_NoCh
    incf NChange
    clrf FSR
L0_NoCh:
    endm
;
;
;
    org 0x100
;
USER_CODE_ENTRY_POINT:
;
    clrwdt
;
    clrf INTCON

```

```

clrf STATUS
clrf PCLATH
;
movlw 0xcc
call Usart_Out
movlw 0xdd
call Usart_Out
movlw 0xee
call Usart_Out
movlw 0xff
call Usart_Out
;
;
clrf PORTA
clrf PORTC
bsf PORTA, 3
bsf PORTC, 0
clrf PORTB
;
bsf STATUS, RP0
movlw b'01001111'
movwf OPTION_REG; налаштували перший таймер
movlw b'11111111'
movwf TRISB; налаштували порт
bsf TRISC, 0
bcf TRISC, 3
bsf TRISC, 4
bcf TRISC, 5
bsf TRISC, 2
bcf TRISA, 3
bsf TRISC, 6
bsf TRISC, 7
bcf STATUS, RP0
;
movlw b'00000000'
movwf INTCON; налаштували переривання
;
; налаштування MSSP
bsf STATUS, RP0
bcf STATUS, RP1
movlw b'00000000'
movwf SSPSTAT
bcf STATUS, RP0
bcf STATUS, RP1
movlw b'00110000'; Fosc/4
;movlw b'00110001'; Fosc/16
movwf SSPCON
;
; налаштуємо АЦП та кільцевий буфер
;
movlw b'01000001'
movwf ADCON0
bsf STATUS, RP0
movlw b'10001110'
movwf ADCON1
bcf STATUS, RP0
;clrf ACP_BufPos
;clrf ACP_ReadPos
;movlw -5
;movwf ACP_NNN
;
;;; налаштуємо ШИМ
;movlw b'00000000'
;movwf T1CON; TMR1
;movlw b'00001100'
;movwf T2CON
;movlw b'00001100'
;movwf CCP1CON
;movlw b'00000000'

```

```

;movwf CCP2CON
;;
;bsf STATUS, RP0
;movlw 0x3f
;movwf PR2; період таймера 2
;bcf STATUS, RP0
;
; movlw ACP_MemoryBuffer
; movwf FSR
;SH_Init:
; movf FSR, w
; movwf INDF
; incfsz FSR
; goto SH_Init
;
;
movlw b'00000101'
movwf T1CON; налаштували другий таймер
;
movlw b'00000000'
movwf CCP1CON
;movlw b'00000111'; кожен 16-й фронт
movlw b'00000101'; кожен фронт
movwf CCP1CON; налаштували на захоплення
;
clrf TMR1H
clrf TMR1L
clrf TMR0
;
call I2C_Stop
;
;
; ініціалізація MMC
call MMC_Init
;
;goto NewLoop
;
;
call ReadEmptyPosFromEEPROM ; читаємо з EEPROM змінну EmptyPosCode та
конвертуємо в EmptyPos
;
MOVE32M File_StartAdr, EmptyPos; і це буде перший сектор нового файлу
;
clrf Shim_ST
clrf SysTime
clrf SysTime+1
clrf SysTime+2
clrf SysTime+3
;
bsf STATUS, RP0
movlw b'01000000'
movwf PIE1
movlw b'00000000'
movwf PIE2
bcf STATUS, RP0
movlw b'11000000'
movlw 0
movwf INTCON

;goto NewLoop

movlw b'11100000'
movwf INTCON
;
;
;
call StartMMCBLOCKWrite
;
MainWriteLoop:

```

```

    clrwdt; зараз напевно йде запис до SD-карти, і можна встигнути
зробити свої справи
    ;
    ; перевіряємо, чи немає команди на зупинку запису
    call TestStopCommand
    btfsc STATUS, Z
    goto MainWriteLoop_Exit
    ;
    btfss MMC_Init_Status, MMCINIT_CANWRITE
    goto MainWriteLoop_NoChangeFlash
    ;
    movf EmptyPos+0, w; дивимося, чи потрібно записувати в EEPROM змінену
змінну EmptyPos
    andlw (FILE_GRANULA-1)
    JNZ MainWriteLoop_NoChangeFlash
    ;
    ; Кодуємо EmptyPos в EmptyPosCode з перевіркою зміни: у WaitST число
байт, що змінилися, у FSR -
    ; номер байта (зміна молодших біт EmptyPosCode маскою FILE_GRANULA
може ігноруватися)
    ;
    EmptyPosCodeAndTestChangeM EmptyPosCode, EmptyPos, WaitST
    movf WaitST, w
    JZ MainWriteLoop_NoChangeFlash
    decfsz WaitST, w
    goto MainWriteLoop_CoderFatalErr
    ;
    ; Тепер FSR - номер єдиного змінного байта змінної EmptyPosCode
    ; який і потрібно зараз записати у flash пам'ять МК
    ;
    movf FSR, w; встановлюємо адресу EEPROM для запису з попередньою
call Eeprom_Write_Set_EEADR; очікуванням закінчення попереднього
запису у flash МК
    movf FSR, w; (завдяки FILE_GRANULA, очікування фактично не буде)
    addlw EmptyPosCode
    movwf FSR
    movf INDF, w
    ;
    bsf STATUS, RP1; ініціюємо запис у flash-пам'ять МК одного байта, що
змінився EmptyPos

movlw 0xfc
    call Stream_OutMMC
    ;
    ; А потім передаємо дані
    ;
    movlw 0xaa; пишемо ПСП
    call Stream_OutByte
    movlw 0x55;
    call Stream_OutByte
    movlw 0x39;
    call Stream_OutByte
    movlw 0xa7;
    call Stream_OutByte
    ;
    Stream_OutDWORD File_StartAdr ; пишемо початкову адресу файлу
    Stream_OutDWORD EmptyPos; пишемо поточну адресу
    ;
    movf SysTime+0, w; пишемо місцевий час
    call Stream_OutByte
    movf SysTime+1, w
    call Stream_OutByte
    movf SysTime+2, w
    call Stream_OutByte
    movf SysTime+3, w
    call Stream_OutByte
    ;
    ;
    movf TH_Config, w; пишемо виміряну температуру

```

```

call Streem_OutByte
movf TH_Temperature, w
call Streem_OutByte
movf TH_Temperature+1, w
call Streem_OutByte
movf TH_Counter, w
call Streem_OutByte
movf TH_Slope, w
call Streem_OutByte
movf I2C_Error, w
call Streem_OutByte
;
movf CH2_Lo, w; пишемо значення спец АЦП
call Streem_OutByte
movf CH2_Hi, w
call Streem_OutByte
;
clrf CH2_Lo; і дозволяємо ще накопичення даних у CH2
clrf CH2_Hi
clrf CH2_N
;
Streem_OutDWORD NewLockRes_T0; пишемо вимір частоти T0, S, N
Streem_OutDWORD NewLockRes_S
Streem_OutDWORD NewLockRes_N
bsf LockAct, 0; та ініціюємо захоплення наступного фронту через CCP
;
;
Streem_OutWORD XOUT; пишемо усереднені дані від акселерометра
MMA7455LT
Streem_OutWORD YOUT
Streem_OutWORD ZOUT
movf I2C_Error, w
call Streem_OutByte
movf XYZN, w
call Streem_OutByte
CLR16M XOUT; і починаємо нове накопичення даних від акселерометра
MMA7455LT
CLR16M YOUT
CLR16M ZOUT
clrf XYZN
;
;
movlw 256-8-3-1-6 - 4; в частину сектора, що залишилася, будемо
писати результати АЦП з буфера
movwf WaitST
MainWriteLoop_OutLoop:
;
Wait_ACP_Data:
movf ACP_ReadPos, w
subwf ACP_BufPos, w
btfss STATUS, Z
goto Wait_ACP_DataReady
;
; буфер АЦП порожній - можна виконати опитування та усереднення
показань акселерометра MMA7455LT
;
btfsc XYZN, 7
goto Wait_ACP_Data; накопичено вже достатньо вимірів
movlw 0x1d * 2
movwf I2C_Adress
clrf I2C_Error
call MMA7455_Init
call MMA7455_GetStatus
btfss DS1621_Byte0, 0
goto Wait_ACP_Data; дані від акселерометра ще не готові
call MMA7455_Read8
ADD16SigB XOUT, DS1621_Byte0
ADD16SigB YOUT, DS1621_Byte1
ADD16SigB ZOUT, DS1621_Byte2

```

```

    incf XYZN
    goto Wait_ACP_Data
    ;
Wait_ACP_DataReady:
    movf ACP_ReadPos, w
    andlw 0xfe
    movwf FSR
    movf INDF, w
    call Stream_OutByte
    incf FSR
    movf INDF, w
    call Stream_OutByte
    incfsz FSR, w
    goto $+2
    movlw ACP_MemoryBuffer
    movwf ACP_ReadPos
    decfsz WaitST
    goto MainWriteLoop_OutLoop
    ;
    call Stream_OutMMC; пишемо кінець пакету - CRC
    call Stream_OutMMC
    ;
    ; передаємо стан запису на SD-карту: <прапор ініціалізації, відповідь
на CMD25, Data Response Token>
    movf MMC_Init_Status, w
    call Usart_Out
    movf MMCWriteCmdR1, w
    call Usart_Out
    movf MMCWriteDataRespToken, w
    call Usart_Out
    ;
    btfss MMC_Init_Status, MMCINIT_CANWRITE
    goto MainWriteLoop
    ;
    ;
    call SPI_ReadByte; читаємо відповідь SD - карти відразу за переданим
блоком даних
    movwf FSR
    andlw 0x1f
    sublw 0x5
    btfsc STATUS, Z
    goto MainWriteLoop_WrOK
    ;
    ; виникла помилка під час запису в SD/MMC - карту
    movf FSR, w
    movwf MMCWriteDataRespToken; збережемо код помилки для візуалізації
    ;
    bcf MMC_Init_Status, MMCINIT_CANWRITE
    ;
    movlw CMD12_STOPREAD ; тільки у разі помилки даємо команду скасування
    call MMC_Cmd
    call MMC_OffCS; вимикаємо SD-картку
    goto MainWriteLoop
    ;
MainWriteLoop_WrOK:
    INC32M EmptyPos
    goto MainWriteLoop
    ;
    ;
    ;
    ;
Stream_OutByte:
    call Usart_Out
Stream_OutMMC:
    btfsc MMC_Init_Status, MMCINIT_CANWRITE
    goto SPI_OutByte
    return
    ;
    ;

```

```

; Підготувати SD-карту до запису безліч блоків даних з сектора
EmptyPos
; Якщо картка готова, то MMC_Init_Status встановлюється біт
MMCINIT_CANWRITE
; та карта включена (CS=0)
;
StartMMCBlockWrite:
    movlw 0xff
    movwf MMCWriteCmdR1
    movwf MMCWriteDataRespToken
    bcf MMC_Init_Status, MMCINIT_CANWRITE
    ;
    btfss MMC_Init_Status, MMCINIT_INITOK
    return; SD-карта не ініціалізована
    ;
    movf EmptyPos+2, w
    iorwf EmptyPos+3, w
    btfsc STATUS, Z
    return; Номер сектора < 65536, що не допустимо
    ;
    MOVE32M MMC_Adr, EmptyPos
    btfsc MMC_Init_Status, MMCINIT_HC
    goto StartMMCBlockWrite_UseHCAddress
    ;
    clrf MMC_Adr+0; Карта не HC - переведення номера сектора в номер
байта
    bcf STATUS, C
    rlf EmptyPos+0, w
;
    movlw 100
    виклик Usart_In_WDT
    btfsc STATUS, C
    перейти до NewLoop
    sublw 0xcd
    btfss STATUS, Z
    перейти до NewLoop
    movlw 100
    виклик Usart_In_WDT
    btfsc STATUS, C
    перейти до NewLoop
    movwf FSR
    sublw 12
    btfsc STATUS, Z
    перейти до CMD1
    movf FSR,w
    sublw 13
    btfsc STATUS, Z
    перейти до CMD2
    movf FSR,w
    sublw 23
    btfsc STATUS, Z
    перейти до CMD4
    movf FSR,w
    sublw 25
    btfsc STATUS, Z
    перейти до CMD5
    movf FSR,w
    sublw 11
    btfsc STATUS, Z
    перейти до EnterNewLoop
    movf FSR,w
    sublw 16
    btfsc STATUS, Z
    перейти до CMD3
    movf FSR,w
    sublw 34
    btfsc STATUS, Z
    перейти до CMDE_ReadEmptyPos

```

```

movf FSR,w
sublw 35
btfsc СТАТУС, Z
перейти до CMDE_WriteEmptyPos
;movf FSR,w
;sublw 27
;btfsc СТАТУС, Z
;перейти до CMD7
movf FSR,w
sublw 10
btfsc СТАТУС, Z
перейти до 0
;
movlw 0xee
виклик Usart_Out
перейти до NewLoop
;
;
CMD3:
викликати MMC_Init
перейти до NewLoop
;
;
; прочесть з Usart 4 байт адреса для MMS
;
CMDs_WaitUsartAdr:
movlw 200
виклик Usart_In_WDT
btfsc СТАТУС, C
повернення
movwf MMC_Adr+0
movlw 200
виклик Usart_In_WDT
btfsc СТАТУС, C
повернення
movwf MMC_Adr+1
movlw 200
виклик Usart_In_WDT
btfsc СТАТУС, C
повернення
movwf MMC_Adr+2
movlw 200
виклик Usart_In_WDT
movwf MMC_Adr+3
повернення
;
;
; КОМАНДА ЧТЕНИЯ СЕКТОРА ИЗ ММС ПО ЗАДАННОМУ АДРЕСУ
; На входе номер сектора и размер читаемых данных
; У відповідь передається 0xfe, якщо команда прийнята і будуть дані,
; або відповідь R1 або маркер помилки даних, якщо на цьому все.
; Якщо розмір не 512, замість CRC передається маркер 0xaa, 0x79
CMD1:
;
; очікуємо з Usart 4 байта адреси повідомлення з MMS
виклик CMDs_WaitUsartAdr
btfsc СТАТУС, C
перейти до NewLoop
;
; очікуємо з Usart байт розміру даних для часткового чтєння
movlw 100
виклик Usart_In_WDT
btfsc СТАТУС, C
перейти до NewLoop
movwf ReadMMCSize
;
викликати MMC_Adr_KorrHC
movlw 0x40; імітація відповіді SD - карти "помилка параметра"

```

```

JNZ CMD1_Read_Error
;
викликати MMC_OnCS_WaitEndOfWrite
movlw CMD17_READ_SINGLE_BLOCK
викликати MMC_Cmd
i lw 0xff
btfss STATUS, Z
перейти до CMD1_Read_Error
;
; очікуємо дані або може бути помилка коду
викликати MMC_WaitResp_Long
виклик Usart_Out
sublw 0xfe; якщо маркер 0xfe, то слідом будуть дані, інакше помилка
btfss STATUS, Z
перейти до CMD1_OffMMC
;
; Відповідь 0xfe - буде блок даних: DATA, 2 байта CRC, 0x55
;
викликати SPI_ReadByte; переходимо в кльову синхронізацію (апарат
SSP завжди швидше програми)
;
CMD1_ReadUsart:
; читаємо і передаємо в частині сектора Usart в ReadMMCSize*2 байт
movf ReadMMCSize, w
movwf ЗачекайтеST
CMD1_Loop:
викликати SPI_ReadAndOut_Usart
викликати SPI_ReadAndOut_Usart
decfsz ЗачекайтеST
перейти до CMD1_Loop
;
викликати CMD5_SkipData
;
movf ReadMMCSize, w
JZ CMD1_SendCRC
;
викликати SPI_ReadByte; читаємо непотрібні нам 2 байта CRC (перший
уже доступний, другий ждемо)
;
movlw 0xaa ; но в комп відправляємо спеціальний код (істинна CRC
безкоштовна)
виклик Usart_Out
movlw 0x79
виклик Usart_Out
перейти до CMD1_OffMMC
;
CMD1_SendCRC:
; а також читаємо і відправляємо в комп два байта CRC
викликати SPI_ReadAndOut_Usart
викликати SPI_ReadAndOut_Usart
перейти до CMD1_OffMMC
;
CMD1_Read_Error:
andlw 0x7f
виклик Usart_Out
CMD1_OffMMC:
викликати MMC_OffCS
перейти до NewLoop
;
;
;
; КОМАНДА ЧТЕНИЯ МНОЖЕСТВА СЕКТОРОВ, начиная с заданого адреса
; На вході: номер сектора, розмір блоку/2, код швидкості обміну (0-
usart57600 або k-115200*k)
; Відклик: 0x53, відповідь R1 на команду SD, і якщо ні, то буде
передача даних
; З цього режиму можна буде вийти тільки командою STOP
; Спочатку передається швидкість обміну видом 0xTT, де T - множник
115200, і далі передаються

```

```

; дані на новій швидкості, поки немає команди STOP або не виникає
помилка чтення
; Для виходу з бесконечного циклу необхідно послати команду STOP
; При виході з бесконечного циклу включається стандартна швидкість
обміну по Usart,
; множинне чтення змінюється (шлеться R1) і посилається стандартна
квітація на STOP.
; Команду STOP можна надіслати повторно (відповідає головний цикл),
щоб підтвердити,
; что приглашенная ранее передача данных прекращена
; Формат даних - 0xfe, дані, CRC. Якщо не 0xfe, то даних більше не
буде
; Перед входом у високошвидкісний режим пауза 1 сек, і видається код
перевірки 0x99, 0xaa, 0x12, 0x55
;
CMD5:
    bcf INTCON, GIE
    ;
    ; очікуємо з Usart 4 байта адреси повідомлення з MMS
    виклик CMDs_WaitUsartAdr
    btfsc STATUS, C
    перейти до NewLoop
    ;
    ; очікуємо з Usart байт розміру даних для часткового чтення
    movlw 200
    виклик Usart_In_WDT
    btfsc STATUS, C
    перейти до NewLoop
    movwf ReadMMCSize
    ;
    ; очікуємо з коду Usart швидкості видачі i
    ;
    ;
    ;
; КОМАНДА FILL СЕКТОРА З MMC ЗА ЗАДАНИМ АДРЕСОМ
CMD2:
    ; очікуємо з Usart 4 байти адреси запису (Fill) у MMC
    call CMDs_WaitUsartAdr
    btfsc STATUS, C
    goto NewLoop
    ;
    ; очікуємо з Usart байт для заповнення сектора
    movlw 200
    call Usart_In_WDT
    btfsc STATUS, C
    goto NewLoop
    movwf ReadMMCSize
    ;
    call MMC_OnCS_WaitEndOfWrite
    movlw 0x58; команда на запис сектора
    call MMC_Cmd
    call Usart_Out
    addlw 0
    btfss STATUS, Z
    goto CMD2_Write_Error
    ;
    call SPI_ReadByte; один байт для очікування реакції
    ;
    ; а тепер запис пакета даних
    movlw 0xfe
    call SPI_OutByte
    clrf WaitST
CMD2_Loop:
    movf ReadMMCSize, w
    ;incf ReadMMCSize
    call SPI_OutByte
    movf ReadMMCSize, w
    ;incf ReadMMCSize
    call SPI_OutByte

```

```

decfsz WaitST
goto CMD2_Loop
;
call SPI_ReadByte; пишемо кінець пакету - CRC
call SPI_ReadByte
;
call SPI_ReadByte; читаємо відповідь відразу ж за посланим блоком
;movwf ReadMMCSIZE
call Usart_Out
;
;call SPI_ReadByte; читаємо ще один про всяк випадок
;
;call MMC_WaitEndOfWrite
;
;call MMC_WaitForWriteEnd
;
;movf ReadMMCSIZE, w
;call Usart_Out
;
; ймовірно, запис виконано успішно
call MMC_OffCS; вимикаємо MMC (CS:=1);
;movlw 0x55
;call Usart_Out
goto NewLoop
;
CMD2_Write_Error:
call MMC_OffCS
movlw 0xee
call Usart_Out
goto NewLoop
;
; КОМАНДА ЗАПISУ СЕКТОРА З MMC ЗА ЗАДАНИЙ АДРЕС
CMD4:
; очікуємо з Usart 4 байти адреси для запису в MMC
call CMDs_WaitUsartAdr
btfsc STATUS, C
goto NewLoop
;
call MMC_OnCS_WaitEndOfWrite
movlw 0x58; команда на запис сектора
call MMC_Cmd
call Usart_Out
addlw 0
btfss STATUS, Z
goto CMD4_Write_Error
;
call SPI_ReadByte; один байт для очікування реакції
;
; а тепер запис пакета даних
movlw 0xfe
call SPI_OutByte
clrf WaitST
CMD4_Loop:
movlw 100
call Usart_In_WDT
call SPI_OutByte
movlw 100
call Usart_In_WDT
call SPI_OutByte
decfsz WaitST
goto CMD4_Loop
;
call SPI_ReadByte; пишемо кінець пакету - CRC
call SPI_ReadByte
;
call SPI_ReadByte; читаємо відповідь відразу ж за посланим блоком
call Usart_Out
;

```

```

; ймовірно, запис виконано успішно
call MMC_OffCS; вимикаємо MMC (CS:=1);
goto NewLoop
;
CMD4_Write_Error:
call MMC_OffCS
movlw 0xee
call Usart_Out
goto NewLoop
;
;
;
; Перевіряємо, чи немає Usart - команди на зупинку послідовного
читання (якщо так, то прапор Z = 1)
; Перевірка на цю команду в головному командному циклі має бути такою
самою
;
TestStopCommand:
btfsc RCSTA, OERR
call Usart_In_Reset
movf RCREG, w
sublw 0xcd
btfss STATUS, Z
return
movf RCREG, w
sublw 11
return
;
StopCommandSendUsart: ; посил за Usart стандартного підтвердження на
команду Stop
movlw 0x5e
call Usart_Out
movlw 0x44
call Usart_Out
movlw 0xe7
call Usart_Out
movlw 0x23
goto Usart_Out
;
;
;
CMDE_ReadEmptyPos: ; Команда запиту вільної позиції
movlw 0x71
call Usart_Out
call ReadEmptyPosFromEEPROM
movf EmptyPos, w
call Usart_Out
movf EmptyPos+1, w
call Usart_Out
movf EmptyPos+2, w
call Usart_Out
movf EmptyPos+3, w
call Usart_Out
goto NewLoop
;
CMDE_WriteEmptyPos: ; Команда встановлення вільної позиції
call CMDs_WaitUsartAdr
btfsc STATUS, C
goto NewLoop
;
EmptyPosCoderM EmptyPosCode, MMC_Adr
;
call CMDs_WaitUsartAdr
btfsc STATUS, C
goto NewLoop
;
XORL32M MMC_Adr, 0xale8bdca
CMP32M MMC_Adr, EmptyPosCode
btfss STATUS, Z

```

```

goto NewLoop
;
movlw 0
call Eeprom_Write_Set_EEADR
movf EmptyPosCode+0, w
call Eeprom_Write_EEADR_w
movf EmptyPosCode+1, w
call Eeprom_Write_EEADR_w
movf EmptyPosCode+2, w
call Eeprom_Write_EEADR_w
movf EmptyPosCode+3, w
call Eeprom_Write_EEADR_w
;
movlw 0x71
call Usart_Out
goto NewLoop
;
;
;
; Читання з EEPROM змінної EmptyPosCode та її перетворення на
EmptyPos
;
ReadEmptyPosFromEEPROM:
movlw 0
call Eeprom_Read_w
movwf EmptyPosCode+0
movlw 1
call Eeprom_Read_w
movwf EmptyPosCode+1
movlw 2
call Eeprom_Read_w
movwf EmptyPosCode+2
movlw 3
call Eeprom_Read_w
movwf EmptyPosCode+3
;
MOVE32M EmptyPos, EmptyPosCode
;
movlw EmptyPos+4; декодування EmptyPos
movwf FSR
movlw 8*4
movwf WaitST
bcf STATUS, C
ReadEmptyPosDecode1:
movf WaitST, w
andlw 7
btfsc STATUS, Z
decf FSR
movlw 0x80
btfsc STATUS, C
xorwf INDF
rlf INDF, w
rlf INDF
decfsz WaitST
goto ReadEmptyPosDecode1
;
; Робимо EmptyPos кратним FILE_GRANULA. Пам'ятай: через скорочену in
call SPI_ReadByte; а кроме того в ответ на CMD8 должны получить в ответ
эхо
call Usart_Out
iorwf WaitST
call SPI_ReadByte
call Usart_Out
iorwf WaitST
call SPI_ReadByte
call Usart_Out
sublw 1
iorwf WaitST
call SPI_ReadByte

```

```

call Usart_Out
    sublw 0xaa
    iorwf WaitST, w
    btfsc STATUS, Z
    bsf MMC_Init_Status, MMCINIT_SD2; Отже, версія специфікації SD >= 2.0
    ;
    ; посилаємо тепер CMD1 або ACMD41 достатню кількість разів, поки
SD/MMC не прокинеться
    movlw 200
    movwf WaitST
MMC_Init_L21:
    clrwdt; !!!
    movlw 50
    call Delay1; чекаємо 5 мс
    ;call SPI_ReadByte; надсилаємо байт 0xff перед кожною командою
    ;movlw 0x41
    ;call MMC_Cmd; видаємо команду CMD1
    ;movwf FSR
    ;call Usart_Out
    ;sublw 0
    ;btfsc STATUS, Z
    ;goto MMC_Init_L3; відповідь 0 - значить, карта готова до роботи
    ;
    ;
    call SPI_ReadByte; надсилаємо байт 0xff перед кожною командою
    movlw CMD55_APP_CMD
    call MMC_Cmd
    andlw 0xff-1
    btfss STATUS, Z
    goto MMC_Init_L23; не зрозуміло, чому CMD55 не прийнята
    ;
    btfsc MMC_Init_Status, MMCINIT_SD2
    bsf MMC_Adr+3, 6; якщо SD>=2.0, то включаємо біт HCS (D30, що наш
хост підтримує HC)
    ;
    call SPI_ReadByte; надсилаємо байт 0xff перед кожною командою
    movlw ACMD41_SD_SEND_OP_COND
    call MMC_Cmd
    addlw 0
    btfsc STATUS, Z
    goto MMC_Init_L3; відповідь 0 - значить, карта готова до роботи
MMC_Init_L23:
    ;
    clrf MMC_Adr+3; тоді спробуємо ініціалізувати без HC
    call SPI_ReadByte; надсилаємо байт 0xff перед кожною командою
    movlw CMD55_APP_CMD
    call MMC_Cmd
    andlw 0xff-1
    btfss STATUS, Z
    goto MMC_Init_L22; не зрозуміло, чому CMD55 не прийнята
    ;
    call SPI_ReadByte; надсилаємо байт 0xff перед кожною командою
    movlw ACMD41_SD_SEND_OP_COND
    call MMC_Cmd
    addlw 0
    btfsc STATUS, Z
    goto MMC_Init_L3; відповідь 0 - значить, карта готова до роботи
    ;
MMC_Init_L22:
    decfsz WaitST
    goto MMC_Init_L21
    ; помилка, карта все ще не готова до роботи
    call Usart_Out
    goto MMC_Init_Exit
    ;
MMC_Init_L3:
    call Usart_Out
    ;

```

```

    bsf MMC_Init_Status, MMCINIT_INITOK; *** Карта була успішно
ініціалізована ***
    ;
    call SPI_ReadByte; надсилаємо байт 0xff перед кожною командою
    movlw CMD58_READ_OCR ; прочитаємо OCR, в якому біт 30 це ознака SD HC
    call MMC_Cmd
    call Usart_Out
    movwf FSR
    ;
    call SPI_ReadByte; читаємо старший байт OCR (D30 це ознака, що карта
HC)
    call Usart_Out
    movf FSR, f; перевіряємо відповідь R1: якщо не нуль, то читання OCR
не вдале
    btfss STATUS, Z
    andlw 0
    andlw 0x40
    btfss STATUS, Z
    bsf MMC_Init_Status, MMCINIT_HC
    ;
    call SPI_ReadByte
    call Usart_Out
    call SPI_ReadByte
    call Usart_Out
    call SPI_ReadByte
    call Usart_Out
    goto MMC_Init_Exit
    ;
    ;
    ;
MMC_OnCS: ; Включення MMC низьким рівнем CS:=0
    bcf PORTA, 3
    goto SPI_ReadByte; пропускаємо 1 байт після включення CS (начебто це
важливо)
    ;
MMC_OffCS: ; Вимкнення MMC високим рівнем CS:=1
    call SPI_ReadByte; належить робити цикл після кожної команди
    bsf PORTA, 3
    goto SPI_ReadByte; і ще цикл шини, щоб було реальне вимкнення
    ;
    ;
MMC_Cmd:
    call SPI_OutByte; видаємо код команди
    movf MMC_Adr+3, w
    call SPI_OutByte; 4 байт адреси
    movf MMC_Adr+2, w
    call SPI_OutByte
    movf MMC_Adr+1, w
    call SPI_OutByte
    movf MMC_Adr+0, w
    call SPI_OutByte
    movlw 0x95
MMC_Cmd_SendCRC:
    call SPI_OutByte; та контрольну суму
    ;
MMC_WaitRespCmd:
    movlw 64
    ;
    ; очікування відповіді від MMC (першого не 0xff байта)
    ; на вході в w - число байт очікування
MMC_WaitResp:
    movwf FSR; і чекаємо на відповідь
MMC_Cmd_Wait:
    call SPI_ReadByte
    addlw -0xff
    btfss STATUS, Z
    goto MMC_Cmd_Ret
    decfsz FSR
    goto MMC_Cmd_Wait

```

```

MMC_Cmd_Ret:
    addlw +0xff
    return
    ;
    ;
    ; Тривале очікування першого не 0xff байта від MMC
    ; На вході в w - кількість блоків по 256 байт для очікування
    ;
MMC_WaitResp_Long:
    clrf WaitST
MMC_WaitResp_Long_L1:
    clrf FSR
MMC_WaitResp_Long_L2:
    call SPI_ReadByte
    addlw -0xff
    btfss STATUS, Z
    goto MMC_WaitResp_Long_Ret
    decfsz FSR
    goto MMC_WaitResp_Long_L2
    decfsz WaitST
    goto MMC_WaitResp_Long_L1
MMC_WaitResp_Long_Ret:
    addlw +0xff
    return;
    ;
    ;
    ; Включити MMC та дочекатися закінчення запису
    ;
MMC_OnCS_WaitEndOfWrite:
    call MMC_OnCS
    clrf WaitST
MMC_WaitEndOfWrite_L1:
    clrf FSR
MMC_WaitEndOfWrite_L2:
    call SPI_ReadByte
    sublw 0
    btfss STATUS, Z
    return
    decfsz FSR
    goto MMC_WaitEndOfWrite_L2
    decfsz WaitST
    goto MMC_WaitEndOfWrite_L1
    ; не вдалося дочекатися, видно якісь глюки
    return
    ;
    ;
    ;
    ; введення байта за SPI
SPI_ReadByte:
    movlw 0xff
    ; виведення байта по SPI та прийом байта у відповідь
SPI_OutByte:
    movwf SSPBUF
    bsf STATUS, RP0
SPI_OutByte_LL:
    btfss SSPSTAT, BF
    goto SPI_OutByte_LL
    bcf STATUS, RP0
    movf SSPBUF, w
    return
    ;
    ;
    ;
    ; затримка в інтервалах по 250 тактів (0.1 мс), регістр FSR
    руйнується
Delay1:
    movwf    incf EEADR
    bcf STATUS, RP1

```

```

return
;
;
; Встановлення EEADR:=W, перед установкою виконує очікування
завершення запису
;
Eeprom_Write_Set_EEADR:
    bsf STATUS, RP1
    bsf STATUS, RP0
    btfsc EECON1, WR
    goto $-1
    bcf STATUS, RP0
    movwf EEADR
    bcf STATUS, RP1
    return
;
; *****
*****
;
USER_CODE_IRQ:
    movwf IntProc_Temp1
    swapf IntProc_Temp1
    swapf STATUS, w
    clrf STATUS
    movwf IntProc_Temp2
;
    btfsc INTCON, T0IF
    goto IRQ_StartACP
;
    bcf PIR1, ADIF; ну, мабуть це переривання після закінчення АЦП
;
    incfsz ACP_NNN
    goto IRQ_Not256
;
; Тепер ACP_NNN == 256
    movlw -10
    movwf ACP_NNN
;
    bcf ADCON0, 3; вибираємо нульовий канал
;
    btfsc CH2_N, 6
    goto IRQ_Exit
;
    bsf STATUS, RP0
    movf ADRESL, w
    bcf STATUS, RP0
    addwf CH2_Lo
    movf ADRESH, w
    btfsc STATUS, C
    incfsz ADRESH, w
    addwf CH2_Hi
;
    incf CH2_N
    goto IRQ_Exit
;
IRQ_Not256:
; читаємо АЦП і додаємо дані з АЦП до ACP_Sum
    bsf STATUS, RP0
    movf ADRESL, w
    bcf STATUS, RP0
    addwf ACP_Sum+0
    movf ADRESH, w
    btfsc STATUS, C
    incfsz ADRESH, w
    addwf ACP_Sum+1
;
; дивимось, чи вже завершено накопичення від CH0
    incfsz ACP_NNN, w
    goto IRQ_Exit

```

```

;
; 2010.07.12+1 - перемикання на інший канал заблоковано (не
використовується):
;bsf ADCON0, 3; вибираємо CH2
;
;;; накопичення закінчено
;movlw -10
;movwf ACP_NNN
;
movf FSR, w
movwf IntProc_FSR
;
bcf ACP_BufPos, 0; завжди парна адреса
movf ACP_BufPos, w
addlw -ACP_MemoryBuffer
movf ACP_BufPos, w
btfss STATUS, C
movlw ACP_MemoryBuffer
movwf FSR
;
movf ACP_Sum, w
movwf INDF
incf FSR
movf ACP_Sum+1, w
movwf INDF
incfsz FSR, w
goto $+2
movlw ACP_MemoryBuffer
movwf ACP_BufPos
;
movf IntProc_FSR, w
movwf FSR
;
clrf ACP_Sum
clrf ACP_Sum+1
;
IRQ_Exit:
    swapf IntProc_Temp2, w
    movwf STATUS
    swapf IntProc_Temp1, w
    retfie
;
;
;
IRQ_StartACP:
;
; запускаємо перетворення АЦП
bsf ADCON0, GO
;
; збільшуємо лічильник часу
INC32M SysTime
;
bcf INTCON, T0IF; це ми з перер. від таймера 0 розібралися
;
;goto IRQ_Exit
;
; 24.06.2009 - обробка переповнення TMR1 та події захоплення від CCP
;
btfss PIR1, CCP1IF
goto Irq_NoLock
;
movf CCP1L, w
movwf Locked_L
movf CCP1H, w
movwf Locked_H
bcf PIR1, CCP1IF
btfss PIR1, TMR1IF
goto Irq_Lock_NoTmr1
; події захоплення та переповнення близькі

```



```

;
MMA7455_Read8:
    movlw 6; XOUT8, слідом YOUT8, ZOUT8
    movwf DS1621_Cmd
    goto DS1621_In3
;
;
; Читання статусу: D0 - дані готові, D1 - були перезаписані
;
MMA7455_GetStatus:
    movlw 9; Status Register
    movwf DS1621_Cmd
    goto DS1621_In1
;
;
;
;
;
Termo_Read:
    movlw b'10010010'
    movwf I2C_Adress
    clrf I2C_Error
    clrf TH_Slope
    ; реєстр конфігурації
    movlw 0xAC
    movwf DS1621_Cmd
    call DS1621_In1
    movf DS1621_Byte0, w
    movwf TH_Config
    ; реєстр температури
    movlw 0xAA
    movwf DS1621_Cmd
    call DS1621_In2
    movf DS1621_Byte0, w
    movwf TH_Temperature+0
    movf DS1621_Byte1, w
    movwf TH_Temperature+1
    ; реєстр Counter
    movlw 0xA8
    m    goto I2C_Stop
;

; Визначення коду швидкості передачі, що відповідає ReadMMCSpeed
; Повинно бути узгоджено з FastOut та ReadBlockSPI_And_FastOut
;
GetFastOutCode:
    btfsc ReadMMCSpeed, 3
    retlw 0x88; 921600 = 115200*8 при ReadMMCSpeed == 8
    btfsc ReadMMCSpeed, 1
    retlw 0x77; 806400 = 115200*7 при ReadMMCSpeed == 7
    btfsc ReadMMCSpeed, 2
    retlw 0x44; 460800 = 115200*4 при ReadMMCSpeed == 4
    retlw 0x00
;
;
; Виведення байта W або Usart, або Fast (RC0) залежно від
ReadMMCSpeed
;
FastOut:
    btfsc ReadMMCSpeed, 3
    goto FastOut921600Proc; 921600 = 115200*8 при ReadMMCSpeed == 8
    btfsc ReadMMCSpeed, 1
    goto FastOut806400Proc; 806400 = 115200*7 при ReadMMCSpeed == 7
    btfsc ReadMMCSpeed, 2
    goto FastOut460800Proc; 460800 = 115200*4 при ReadMMCSpeed == 4
    goto Usart_Out; інакше висновок по апаратному Usart при ReadMMCSpeed
== 0
;

```

```

;
; Виведення блоку довжиною W з SPI або Usart, або Fast (RC0)
;
ReadBlockSPI_And_FastOut:
    movwf WaitST
    btfsc ReadMMCSpeed, 3
    goto ReadBlockSPI_And_FastOut921600L; 921600 = 115200*8 при
ReadMMCSpeed == 8
    btfsc ReadMMCSpeed, 1
    goto ReadBlockSPI_And_FastOut806400L; 806400 = 115200*7 при
ReadMMCSpeed == 7
    btfsc ReadMMCSpeed, 2
    goto ReadBlockSPI_And_FastOut460800L; 460800 = 115200*4 при
ReadMMCSpeed == 4
;
; виведення блоку по апаратному Usart при ReadMMCSpeed == 0
;
ReadBlockSPI_And_FastOutUSART_L:
    call SPI_ReadAndOut_Usart
    decfsz WaitST
    goto ReadBlockSPI_And_FastOutUSART_L
    return
;
;
;
FastOut921600Proc: ; величини затримок: 3,2,3,3,3,2,3,3,2,3
    bcf PORTC, 0
    movwf FSR
    rrf FSR
    rlf PORTC
    rrf FSR
    rlf PORTC
    nop
    rrf FSR
    rlf PORTC
    nop
    rrf FSR
    rlf PORTC
    nop
    rrf FSR
    rlf PORTC
    rrf FSR
    rlf PORTC
    nop
    rrf FSR
    rlf PORTC
    nop
    rrf FSR
    rlf PORTC
    nop
    bsf PORTC, 0
    return
;
;
FastOut806400Proc: ; величини затримок: 3,3,3,3,4,3,3,3,3,3
    bcf PORTC, 0
    movwf FSR
    rrf FSR
    rlf PORTC
    nop
    rrf FSR
    rlf PORTC
    nop
    rrf FSR
    rlf PORTC
    nop
    rrf FSR
    rlf PORTC
    nop
    rrf FSR
    rlf PORTC
    nop

```

```

nop
rrf FSR
rlf PORTC
nop
rrf FSR
rlf PORTC
nop
rrf FSR
rlf PORTC
nop
rrf FSR
rlf PORTC
nop
nop
bsf PORTC, 0
return
;
FastOut460800Proc: ; величины затримок: 5,6,5,6,5,6,5,5,6,5
    bcf PORTC, 0
    movwf FSR
    ;nop
    WAIT2; nop
    rrf FSR
    rlf PORTC
    ;nop
    ;nop
    ;nop
    WAIT4; nop
    rrf FSR
    rlf PORTC
    ;nop
    ;nop
    WAIT3; nop
    rrf FSR
    rlf PORTC
    ;nop
    ;nop
    ;nop
    WAIT4; nop
    rrf FSR
    rlf PORTC
    ;nop
    ;nop
    WAIT3; nop
    rrf FSR
    rlf PORTC
    ;nop
    ;nop
    ;nop
    WAIT4; nop
    rrf FSR
    rlf PORTC
    ;nop
    ;nop
    WAIT3; nop
    rrf FSR
    rlf PORTC
    ;nop
    ;nop
    WAIT3; nop
    rrf FSR
    rlf PORTC
    ;nop
    ;nop
    ;nop
    WAIT5; nop
    bsf PORTC, 0
    ;nop

```

```

    WAIT2; nop
    return
    ;
    ;
    ; Процедури швидкісного читання блоку байт по SPI та швидкої видачі
по Fast (RC0)
    ; Розмір блоку у WaitST
    ; Вважається, що у SSPBUF вже лежать готові до відправки дані, а нові
дані з'являються через
    ; 8 (або 9) тактів після запиту SSPBUF: = 0xff
    ;
ReadBlockSPI_And_FastOut806400L:
    bcf PORTC, 0
    rrf SSPBUF, w
    movwf FSR
    rlf PORTC
    nop
    rrf FSR
    rlf PORTC
    nop
    rrf FSR
    rlf PORTC
    nop
    rrf FSR
    rlf PORTC
    movlw 0xff; запитуємо наступний байт із SPI
    movwf SSPBUF
    rrf FSR
    rlf PORTC
    nop
    rrf FSR
    rlf PORTC
    nop
    rrf FSR
    rlf PORTC
    nop
    rrf FSR
    rlf PORTC
    nop
    nop
    bsf PORTC, 0
    decfsz WaitST
    goto ReadBlockSPI_And_FastOut806400L
    return
    ;
    ;
ReadBlockSPI_And_FastOut921600L:
    bcf PORTC, 0
    rrf SSPBUF, w
    movwf FSR
    rlf PORTC
    rrf FSR
    rlf PORTC
    nop
    rrf FSR
    rlf PORTC
    movlw 0xff
    rrf FSR
    rlf PORTC
    movwf SSPBUF
    rrf FSR
    rlf PORTC
    rrf FSR
    rlf PORTC
    nop
    rrf FSR
    rlf PORTC
    nop

```

```
    rrf FSR
    rlf PORTC
    nop
    bsf PORTC, 0
    decfsz WaitST
    goto ReadBlockSPI_And_FastOut921600L
    return
;
ReadBlockSPI_And_FastOut460800L:
    bcf PORTC, 0
    movwf FSR
    nop
    rrf SSPBUF, w
    movwf FSR
    rlf PORTC
    ;nop
    ;nop
    ;nop
    WAIT4; nop
    rrf FSR
    rlf PORTC
    ;nop
    ;nop
    WAIT3; nop
    rrf FSR
    rlf PORTC
    ;nop
    ;nop
    ;nop
    WAIT4; nop
    rrf FSR
    rlf PORTC
    nop
    movlw 0xff
    movwf SSPBUF
    rrf FSR
    rlf PORTC
    ;nop
    ;nop
    ;nop
    WAIT4; nop
    rrf FSR
    rlf PORTC
    ;nop
    ;nop
    WAIT3; nop
    rrf FSR
    rlf PORTC    ;
;
END
```