

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
Факультет комп'ютерно-інтегрованих технологій, мехатроніки і робототехніки
Кафедра метрології та інформаційно-вимірювальної техніки

ПОЯСНЮВАЛЬНА ЗАПИСКА

до кваліфікаційної роботи
бакалавра

на тему: **Комп'ютеризована система для вимірювання частоти
електричних сигналів**

Виконав: студент 4 курсу, МТ-2
групи _____

Спеціальності: 152 «Метрологія та
інформаційно-вимірювальна техніка»

Гісса А.Д.

(прізвище та ініціали)

Керівник: доцент кафедри М та ІВТ

кандидат технічних наук, доцент

Чепюк Л.О.

(науковий ступінь, вчене звання, посада, прізвище та ініціали)

Рецензент: К.Т.Н., доцент

Добрянський О.О.

(науковий ступінь, вчене звання, посада, прізвище та ініціали)

Житомир – 2023

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
Факультет комп'ютерно-інтегрованих технологій, мехатроніки і робототехніки
Кафедра метрології та інформаційно-виміральної техніки

Спеціальність 152 «Метрологія та інформаційно-вимірально-технічна»

ЗАТВЕРДЖУЮ

Завідувач кафедри метрології та
інформаційно-виміральної техніки
д.т.н., професор Подчашинський Ю.О.

« 30 » 03 2023 р.

ЗАВДАННЯ
на кваліфікаційну роботу бакалавра

Студент Гісса Антон Дмитрович

1. Тема роботи: Комп'ютеризована система для вимірювання частоти електричних сигналів

Керівник роботи: доцент кафедри М та ІВТ кандидат технічних наук, доцент
Чепюк Л.О.

затверджена наказом по університету від « 30 » березня 2023 р. № 129/с

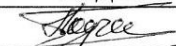
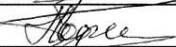
2. Термін здачі закінченої роботи на кафедру « 19 » червня 2023р.

3. Вихідні дані до роботи об'єкт розробки – комп'ютеризована система вимірювання частоти електричних сигналів: діапазон вимірюваних частот від 0.015 Гц до 10МГц; час на один вимір, від 0.1 мс до 17 мс; похибка вимірювання $\pm 0,5 \%$; джерело живлення + 5 В постійного струму.

4. Зміст пояснювальної записки (перелік питань, які підлягають розробці)
огляд та аналіз методів вимірювання огляд та аналіз методів вимірювання частоти електричних сигналів, розробка структурної схеми, розробка принципової схеми, конструкторська частина, розробка алгоритму та програми роботи системи, розрахунок метрологічних характеристик, оцінка показників надійності роботи системи, безпека життєдіяльності та охорона праці

5. Перелік графічного матеріалу (з точним визначенням обов'язкових креслень)
структурна схема, електрична принципова схема, перелік елементів, конструкція приладу, блок-схема алгоритму роботи системи.

6. Консультанти з кваліфікаційної роботи бакалавра із зазначенням розділів роботи, що їх стосуються

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
1-9	Чепюк Л. О., доцент		
			

7. Дата видачі завдання 15.02.2023р.

Календарний план

№ з/п	Назва етапів кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1	Узгодження технічного завдання	25.02.23 – 1.03.23	Виконано
2	Огляд та аналіз методів вимірювання частоти електричних сигналів	2.03.23 – 15.03.23	Виконано
3	Розробка структурної схеми	16.03.23 – 31.03.23	Виконано
4	Розробка електричної принципової схеми	1.04.23 – 24.04.22	Виконано
5	Розробка корпусу пристрою	25.04.23 – 09.05.23	Виконано
6	Розробка алгоритму та програми системи	10.05.23 – 24.05.23	Виконано
7	Розрахунок метрологічних характеристик	25.05.23 – 03.06.23	Виконано
8.	Розрахунок надійності	04.06.23 – 09.06.23	Виконано
9	Оформлення пояснювальної записки та презентаційного матеріалу	10.06.23 – 17.06.23	Виконано

Студент



Гісса А.Д.

Керівник роботи



Чепюк Л. О.

АНОТАЦІЯ

В даній кваліфікаційній роботі розроблено комп'ютеризовану систему на основі мікроконтролеру для вимірювання частоти електричних сигналів. Основним елементом системи є мікроконтролер STM32F100RB.

Цифрові методи вимірювання частоти забезпечують високу точність вимірювання частоти при порівняно простому пристрої вимірювального приладу. Такі вимірювачі дуже універсальні, здатні працювати в дуже широкому діапазоні частот, причому, без необхідності будь-яких перемикачів між вимірювань (крім особливих випадків вимірювання дуже низьких частот або надвисоких частот). У робочому діапазоні зберігається висока швидкість вимірювань і висока точність.

Ключові слова: частота, комп'ютеризована система, мікроконтролер.

ABSTRACT

In this qualification work a computerized system based on a microcontroller for measuring the frequency of electrical signals is developed. The main element of the system is the STM32F100RB microcontroller.

Digital methods provide astonishing accuracy of frequency measurement with a relatively simple device of measuring instrument. Such meters are very versatile, able to work in a very wide range of frequencies, and without the need for any switching of measurement limits (except for special cases of measuring very low frequencies or ultra-high frequencies). High measurement speed and high accuracy are maintained over the entire operating range.

Keywords: frequency, computerized system, microcontroller.

РЕФЕРАТ

Кваліфікаційна робота представлена пояснювальною запискою в обсязі 75 сторінок формату А4, записка містить вступ, основну частину, що складає дев'ять розділів, висновки, список використаних джерел та додатки. Темою роботи є розробка комп'ютеризованої системи для вимірювання частоти електричних сигналів.

Перший розділ містить основні завдання та вимоги до виробу, а саме: умови експлуатації, технічні характеристики, надійність та конструктивне оформлення.

Другий розділ містить загальний огляд методів вимірювання частоти електричних сигналів: поняття частоти періодичного сигналу, середнє значення частоти, вимірювання частоти, динамічна похибка вимірювань.

Третій розділ містить інформацію про особливості структурної схеми, також виконано розробку структурної схеми комп'ютеризованої системи для вимірювання частоти електричних сигналів.

Четвертий розділ містить вибір та опис елементної бази системи. На основі обраної елементної бази розроблено електричну принципову схему.

П'ятий розділ містить огляд розробленого компонування та трасування друкованої плати системи.

Шостий розділ містить огляд розробленого алгоритмічного та програмного забезпечення для комп'ютеризованої системи вимірювання частоти електричних сигналів.

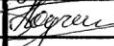
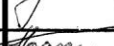
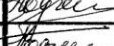
Сьомий розділ містить розрахунок показників надійності роботи системи.

Восьмий розділ містить розрахунок основних метрологічних характеристик комп'ютеризованої системи для вимірювання частоти електричних сигналів.

Останній розділ містить відомості про безпеку життєдіяльності та охорону праці.

ЗМІСТ

ВСТУП.....	8
1. ТЕХНІЧНЕ ЗАВДАННЯ	15
2. ОГЛЯД ТА АНАЛІЗ МЕТОДІВ ВИМІРЮВАННЯ	
ЧАСТОТИ ЕЛЕКТРИЧНИХ СИГНАЛІВ	16
2.1. Поняття частоти періодичного сигналу	16
2.2. Середнє значення частоти. Вимірювання частоти	21
2.3. Динамічна похибка вимірювань	25
3. РОЗРОБКА СТРУКТУРНОЇ СХЕМИ	31
4. РОЗРОБКА ПРИНЦИПОВОЇ СХЕМИ	34
5. КОНСТРУКТОРСЬКА ЧАСТИНА	42
6. РОЗРОБКА АЛГОРИТМУ ТА ПРОГРАМИ РОБОТИ СИСТЕМИ	45
6.1. Алгоритм роботи системи	45
6.2. Програма роботи для частотоміра на STM32.....	50
7. ОЦІНКА ПОКАЗНИКІВ НАДІЙНОСТІ РОБОТИ СИСТЕМИ	64
8. РОЗРАХУНОК МЕТРОЛОГІЧНИХ ХАРАКТЕРИСТИК	68
9. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ ТА ОХОРОНА ПРАЦІ	70
ВИСНОВКИ.....	72
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	73
ДОДАТКИ.....	75

					МММТ.480.003.003 – ПЗ					
Вим	Лист	№ докум.	Підпис	Дата	Комп'ютеризована система для вимірювання частоти електричних сигналів	Лім.		Лист	Листів	
Розроб.	Гісса А.Д.							8	75	
Перевір.	Чепюк Л.О.					Державний університет «Житомирська політехніка», гр. МТ-2				
Реценз.										
Н. Контр.	Чепюк Л.О.									
Затверд.	Подчашинський Ю									
					Пояснювальна записка					

ВСТУП

У наш час науково-технічний прогрес характеризується щоденною розробкою нової техніки, прискорення якої значною мірою залежить від успіхів сучасної мікроелектроніки.

Комп'ютеризована система для вимірювання частоти електричних сигналів називається частотоміром або частотомером. Це пристрій або програмне забезпечення, призначене для точного вимірювання частоти сигналів у герцах (Гц) або їх кратних одиницях.

Частотоміри можуть бути вбудованими в осцилографи, сигнальні генератори або функціональні генератори, а також існують в самостійному вигляді. Вони здатні вимірювати частоти від низькочастотних до високочастотних сигналів, залежно від їхнього діапазону роботи.

Частотоміри зазвичай працюють на основі лічильників подій. Вони виявляють періодичні сигнали, підраховують кількість періодів протягом певного часу і обчислюють частоту, використовуючи формулу: частота = кількість періодів / час.

Ці системи використовуються в багатьох галузях, де точне вимірювання частоти є важливим. Наприклад, в електроніці, телекомунікаціях, радіо, медичинській діагностиці, аудіо та відео-технологіях і багатьох інших областях, де частота сигналів має значення для правильної роботи системи чи аналізу даних.

Частотоміри можуть мати різні характеристики та можливості залежно від своїх технічних характеристик. Деякі частотоміри можуть мати широкий діапазон роботи і здатні вимірювати частоти від кількох герц до кількох гігагерц. Інші можуть мати вбудовані функції, такі як вимірювання періоду, часу затримки або амплітуди сигналів.

Крім того, деякі сучасні частотоміри можуть бути підключені до комп'ютера або мережі, що дозволяє зберігати та аналізувати дані про вимірювання. Це надає можливість зберігати результати вимірювань,

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		8

проводити довготривалі вимірювання, а також використовувати додаткове програмне забезпечення для обробки і аналізу даних.

Частотоміри є важливими інструментами в розробці, випробуванні та обслуговуванні електронних пристроїв і систем. Вони дозволяють виміряти точну частоту генерованих сигналів, перевіряти стабільність і точність роботи генераторів, аналізувати вхідні сигнали і визначати їх характеристики.

Для отримання додаткової інформації про конкретні моделі частотомірів і їх технічні можливості, рекомендується звернутися до спеціалізованих джерел, таких як каталоги виробників, технічні специфікації або наукові публікації, які детально описують принцип роботи і застосування цих пристроїв.

Існує багато комп'ютеризованих систем для вимірювання частоти електричних сигналів, включаючи програмне забезпечення та апаратне забезпечення. Ось декілька прикладів таких систем:

1. Осцилографи: Осцилографи є пристроями, які вимірюють і відображають часовий графік електричних сигналів. Багато осцилографів мають вбудовані функції для вимірювання частоти сигналів.

2. Логічний аналізатор: Логічний аналізатор використовується для аналізу цифрових сигналів. Він може вимірювати частоту сигналів на основі періоду або часу тривалості пульсу.

3. Спектральний аналізатор: Спектральний аналізатор вимірює спектр частот електричних сигналів. Він може використовуватись для визначення точної частоти сигналу або для аналізу його спектрального складу.

4. Програмне забезпечення для комп'ютерів: Існують різні програмні засоби для вимірювання частоти електричних сигналів на комп'ютерах. Ці програми можуть використовувати аудіо- або вхідно-

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		9

вихідні порти комп'ютера для зчитування сигналів і обробки їх для вимірювання частоти.

Ці системи можуть мати різні функції та можливості, включаючи вимірювання точної частоти, аналіз спектра, збереження даних і т.д. Вибір конкретної системи залежить від ваших потреб і бюджету.

Осцилограф - це пристрій, який використовується для вимірювання і візуалізації електричних сигналів у вигляді графіка, який залежить від часу. Осцилографи широко використовуються в електроніці, телекомунікаціях, медицині, наукових дослідженнях та інших галузях.

Осцилографи можуть мати аналоговий або цифровий (цифровий осцилограф, ЦО) формат. Аналогові осцилографи використовують електронні проміжні частоти (ЕПЧ) для перетворення вхідного сигналу у видимий графік на екрані. Цифрові осцилографи перетворюють аналоговий сигнал на цифровий формат з використанням аналого-цифрового перетворювача (АЦП) та зберігають цифрові дані для подальшого відображення на екрані.

Основні характеристики осцилографів включають:

- Ширина смуги: Максимальна частота сигналу, яку може виміряти осцилограф без спотворень.
- Частота дискретизації: Частота збирання дискретних вимірювань сигналу для створення цифрового зображення.
- Вертикальна роздільна здатність: Максимальна кількість дискретних значень, які можуть бути виміряні по вертикалі.
- Горизонтальна роздільна здатність: Максимальна кількість дискретних значень, які можуть бути виміряні по горизонталі.
- Запам'ятовування сигналу: Можливість зберігання та відображення даних про сигнал для подальшого аналізу.

Деякі сучасні осцилографи мають додаткові функції, такі як автоматичне вимірювання параметрів сигналу, математичні операції

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		10

(наприклад, додавання, множення, інтегрування) над сигналами, можливість зберігання даних на зовнішні носії (наприклад, USB-накопичувачі), можливість підключення до комп'ютера для аналізу даних тощо.

Вибір осцилографа залежить від конкретних потреб, бюджету та специфікацій, необхідних для вимірювань. Найбільш підходящий осцилограф варто вибирати, враховуючи потреби та характеристики вашого проекту або дослідження.

Логічний аналізатор є пристроєм, який використовується для аналізу цифрових сигналів, зокрема логічних рівнів (0 та 1). Він здатен вимірювати, відображати та аналізувати послідовність логічних рівнів на різних входних каналах одночасно. Логічні аналізатори використовуються в електроніці, програмуванні, вбудованих системах та інших галузях, де важливим є аналіз цифрових сигналів.

Основні характеристики логічних аналізаторів включають:

1. Кількість входних каналів: Це кількість паралельних входних каналів, на яких можна аналізувати логічні сигнали. Наприклад, логічний аналізатор може мати 8, 16, 32 або більше входних каналів.
2. Швидкість дискретизації: Це максимальна швидкість, з якою логічний аналізатор може зчитувати та аналізувати стани логічних рівнів на входних каналах. Вимірюється у кількості дискретних зчитувань за одиницю часу (наприклад, мегагерц).
3. Роздільна здатність: Це мінімальна одиниця виміру, з якою логічний аналізатор може розрізняти різні логічні рівні. Зазвичай виражається у вольтах або мікрвольтах.
4. Буферна пам'ять: Це обсяг пам'яті, який доступний для зберігання зчитаних даних. Більша буферна пам'ять дозволяє зберігати більшу кількість даних для подальшого аналізу.

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		11

5. Функції аналізу: Деякі логічні аналізатори мають розширені функції аналізу, такі як декодування протоколів зв'язку (наприклад, I2C, SPI, UART) або відображення послідовностей подій на логічних каналах.

Логічні аналізатори можуть бути самостійними пристроями або модулями, які підключаються до комп'ютера чи іншого пристрою за допомогою інтерфейсу, такого як USB або Ethernet. Сучасні логічні аналізатори часто мають дружній інтерфейс користувача та можуть працювати у відповідних програмних середовищах для аналізу та візуалізації цифрових сигналів.

Спектральний аналізатор є пристроєм або програмним забезпеченням, яке використовується для аналізу спектра частот електричних сигналів. Він дозволяє виміряти і візуалізувати розподіл сигналів за частотою.

Основні функції та характеристики спектральних аналізаторів включають:

1. Частотний діапазон: Це діапазон частот, на якому можна вимірювати спектральні складові сигналу. Він може бути від декількох герц до гігагерц.

2. Роздільна здатність: Це мінімальна відстань між двома частотними точками, яку може розрізнити спектральний аналізатор. Визначає точність вимірювання частот.

3. Ширина смуги: Це максимальна ширина діапазону частот, яку спектральний аналізатор може аналізувати одночасно. Відображає максимальну кількість частотних точок на візуалізаційному спектрі.

4. Вимірювання потужності: Спектральні аналізатори здатні вимірювати потужність сигналів на різних частотах. Це дозволяє оцінювати потужність або амплітуду спектральних складових сигналу.

5. Візуалізація спектру: Спектральні аналізатори можуть відображати спектр сигналу у вигляді графіка, такого як спектрограма,

амплітудний спектр або фазовий спектр. Це дозволяє виявляти основні та вторинні частотні складові сигналу.

6. Додаткові функції: Деякі спектральні аналізатори можуть мати додаткові функції, такі як аналіз модуляції сигналу, виявлення періодичних сигналів, сигнальний пошук, спектральна маска та інші.

Спектральні аналізатори широко використовуються у багатьох галузях, включаючи телекомунікації, радіозв'язок, аудіо- та звукотехніку, радіолокацію, медичну діагностику, наукові дослідження та інші області, де аналіз спектра сигналів є важливим.

На комп'ютерах існує безліч програмного забезпечення для вимірювання частоти електричних сигналів. Деякі з них є безкоштовними, а інші мають комерційну ліцензію. Ось кілька популярних програм для вимірювання частоти:

1. Audacity: Це безкоштовний аудіо-редактор, який також має функцію вимірювання частоти. Він дозволяє записувати аудіо з мікрофону або іншого джерела, а потім вимірювати частоту сигналу за допомогою вбудованих інструментів аналізу спектра.

2. MATLAB: MATLAB є потужним інструментом для наукових обчислень і аналізу сигналів. Він має вбудовані функції для обробки сигналів, включаючи вимірювання частоти. MATLAB надає широкі можливості для роботи з реальними та цифровими сигналами.

3. National Instruments LabVIEW: LabVIEW - це візуальне програмне середовище для розробки, яке включає інструменти для вимірювань і обробки сигналів. Воно може бути використане для вимірювання частоти електричних сигналів, а також для подальшого аналізу та візуалізації результатів.

4. PicoScope: PicoScope - це програмне забезпечення для осцилографів та логічних аналізаторів від компанії Pico Technology. Воно дозволяє підключати осцилографи PicoScope до комп'ютера та

використовувати їх для вимірювання частоти сигналів, а також для багатьох інших аналізів.

Це лише кілька прикладів програмного забезпечення, доступного для вимірювання частоти електричних сигналів на комп'ютерах. Вибір певної програми залежить від ваших потреб, бюджету та особистих вподобань.

					МММТ.480.003.003 – ПЗ	Арк.
						14
Змн.	Арк.	№ докум.	Підпис	Дата		

1. ТЕХНІЧНЕ ЗАВДАННЯ

1.1. Підстава для розробки, призначення та галузь застосування

Підставою для розробки є завдання на дипломне проектування.

Система призначена для вимірювання частоти електричних сигналів

Потрібно розробити структурну схему системи та вибрати елементну базу, розробити електричну принципову схему системи вимірювання частоти електричних сигналів. Розробити програмно-алгоритмічне забезпечення та оцінити точність системи вимірювання.

Початок розробки – лютий 2023 р. Кінець розробки – червень 2023 року.

1.2. Умови експлуатації

1.2.1. Температура зовнішнього середовища від 0 до +45 °С.

1.2.2. Відносна вологість не більше 80%.

1.2.3. Атмосферний тиск 600-800 мм. рт. ст.

1.2.4. Хімічно активних речовин немає.

1.2.5. Характер роботи неперервний.

1.3. Вимоги до технічних характеристик

1.3.1. Об'єкт управління – комп'ютеризована система вимірювання

1.3.2. Параметр вимірювання – частота електричного сигналу.

1.3.3. Діапазон вимірювання частоти, від 0.015 Гц до 10 мГц.

1.3.4. Похибка вимірювання частоти, %, $\pm 0,5$.

1.3.5. Відображати отримані значення на індикаторі.

1.3.6. Джерело живлення + 5 В постійного струму

1.4. Вимоги до надійності

Система є що відновлюється, не резервується, ремонтпридатною.

Наробіток на відмову складає не менше 65000 годин.

1.5. Вимоги до конструктивного оформлення

Пристрій виконати на стандартній платі і в стандартному корпусі.

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		15

2. ОГЛЯД ТА АНАЛІЗ МЕТОДІВ ВИМІРЮВАННЯ ЧАСТОТИ ЕЛЕКТРИЧНИХ СИГНАЛІВ

Частота – важливий параметр сигналу. Існує величезна безліч випадків, коли необхідно знати частоту сигналу, або мати можливість виміряти частоту з високою точністю (з метою тестування, діагностики пристроїв і пошуку несправностей; для виконання вимірювань у таких вимірювальних системах, де використовують перетворення величин у частоту сигналу тощо). Дуже часто потрібно генерувати сигнал з потрібною частотою: багато пристроїв містять у собі генератори опорного сигналу, від точності встановлення і стабільності частоти яких залежить нормальна робота пристроїв. Наприклад, дуже жорсткі вимоги висуваються до передавачів (які не повинні заважати сусідам по діапазону). Але і в приймачі частота гетеродина також має бути точно встановлена і стабільна, від цього залежить точність налаштування на передавач і якість прийому. Від якості опорного генератора безпосередньо залежить точність багатьох приладів (таких, наприклад, як цифрові частотоміри або, скажімо, звичайний годинник).

2.1. Поняття частоти періодичного сигналу

Поняття періоду і частоти можуть бути застосовні тільки до періодичних сигналів (або функцій). Функція $u(t)$ називається періодичною, якщо існує число $T > 0$ таке, що для будь-якого t .

$$u(t \pm T) = u(t),$$

значення T називають періодом функції.

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		16

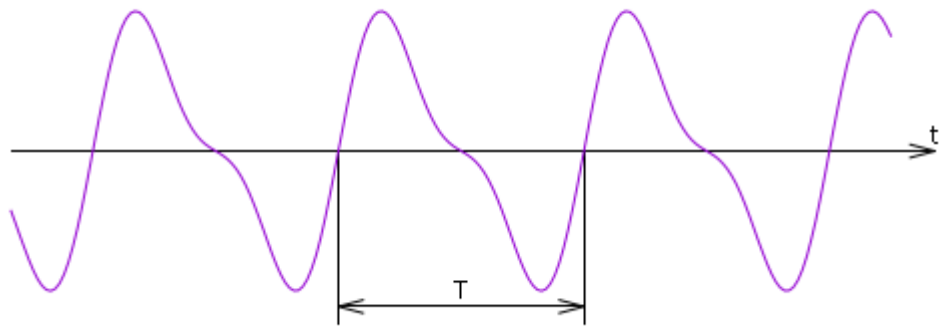


Рис 2.1 Періодичний сигнал

Зрозуміло, що за такого визначення періодична функція має нескінченно багато періодів: якщо деяка величина є періодом функції, то і будь-яка кратна їй величина також буде періодом. Але якщо періодична функція - не константа, то для неї існує найменший період (найменша позитивна величина, що є періодом). Далі під періодом будемо мати на увазі саме найменший період.

Частота періодичного сигналу (функції) - величина, обернена його періоду:

$$f=1/T.$$

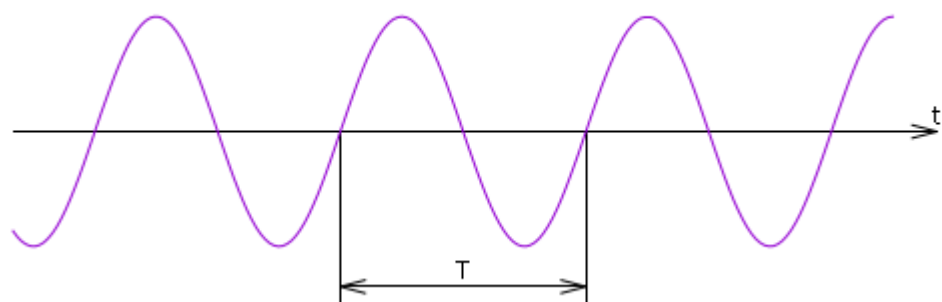


Рис 2.2 Гармонічний сигнал

Прикладом періодичного сигналу є гармонійний (синусоїдальний) сигнал. Для того щоб повністю описати такий сигнал, достатньо задати всього три параметри: амплітуду сигналу A , період T (або, що рівнозначно, частоту f) і початкову фазу φ_0

$$u(t) = A \sin\left(\frac{2\pi}{T}t + \varphi_0\right)$$

Запишемо те саме з використанням частоти:

$$u(t) = A \sin(2\pi ft + \varphi_0).$$

Поряд із частотою сигналу, також розглядають циклічну (також іноді звану круговою або кутовою) частоту $\omega = 2\pi f$, використовуючи яку, вираз для синусоїдального сигналу можна записати таким чином:

$$u(t) = A \sin(\omega t + \varphi_0).$$

Іноді слово "циклічна" опускають, якщо з контексту або позначень зрозуміло, про яку саме частоту йдеться.

Останній вираз можна записати ще простіше:

$$u(t) = A \sin \varphi(t),$$

де $\varphi(t) = \omega t + \varphi_0$ – фаза сигналу. Неважко помітити, що фаза синусоїдального сигналу лінійно зростає з часом зі швидкістю, що дорівнює циклічній частоті:

$$\frac{d\varphi}{dt} = \omega,$$

звідки можемо виразити частоту через фазу (фаза розглядається як функція часу)

$$\omega = \frac{d\varphi}{dt},$$

$$f = \frac{\omega}{2\pi} = \frac{1}{2\pi} \frac{d\varphi}{dt}.$$

Як і у випадку будь-якого періодичного сигналу, частота синусоїдального сигналу є константою.

Частота реального сигналу. Миттєва частота

Суворі визначення і формальні теоретичні підходи гарні для математики. У реальному житті, в техніці, сигнали ніколи не бувають періодичними. Насамперед тому, що жоден сигнал не може тривати нескінченно довго. Сигнал має початок і кінець, що вже порушує ідеальну періодичність. Але навіть якщо відволіктися від цього, скоріше філософського питання про скінченність існування, то і за час існування сигналу сувора періодичність недосяжна. З іншого боку, деякий ступінь регулярності та повторюваності характерний для дуже багатьох реальних сигналів.

Часто такі сигнали виявляється зручно представляти у квазігармонійній формі

$$u(t) = A(t) \sin \varphi(t),$$

де на відміну від гармонійного сигналу (тобто сигналу виду

$$u(t) = A \sin \omega t),$$

амплітуда $A(t)$ уже не обов'язково постійна, а фаза $\varphi(t)$ не зобов'язана лінійно змінюватися з часом.

Функція $A(t)$ описує поведінку огинаючої сигналу. Фаза $\varphi(t)$ визначає поведінку сигналу в межах "періоду" і "повторюваність" сигналу. Зокрема, фаза визначає моменти проходження сигналу через 0: якщо $\sin \varphi = 0$, то зрозуміло, що і $u(t) = 0$; це відбувається, коли

$$\varphi(t) = k\pi,$$

де k - ціле.

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		19

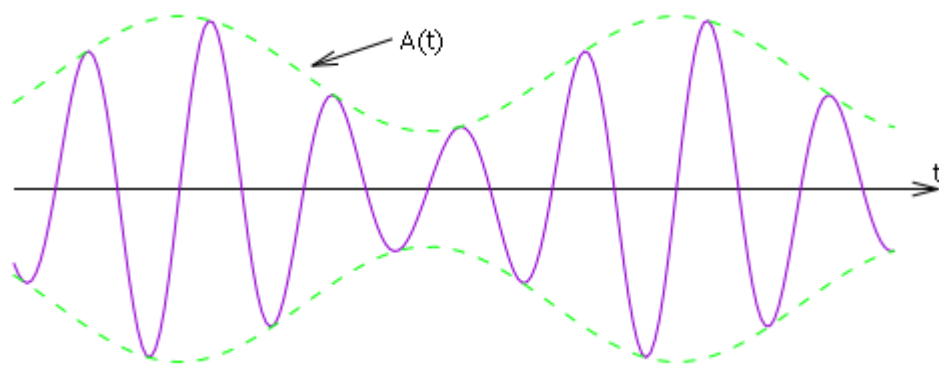


Рис 2.3 Квазігармонічний сигнал

Миттєвою частотою такого квазігармонійного сигналу, за аналогією з гармонійним, називають швидкість зміни його фази:

$$\omega = \frac{d\varphi}{dt}.$$

Миттєва частота реального сигналу - це вже не постійна величина, а функція часу.

Зауважимо, що навіть у разі періодичних, з математичної точки зору, ідеальних сигналів, іноді буває зручніше розглядати їх як "не цілком" періодичні, із змінною в часі амплітудою та/або частотою.

Найпростіший приклад - модульований за амплітудою сигнал під час модуляції його гармонійним сигналом:

$$u(t) = A(1 + m \sin \Omega t) \sin \omega t,$$

припускаємо, що $\Omega \ll \omega$. Для простоти початкові фази вважаємо рівними 0.

Очевидно, що період $T = \frac{2\pi}{\omega}$ несучого коливання вже не є періодом модульованого сигналу через множник $(1 + m \sin \Omega t)$, який зміниться через час T .

Легко показати, що якщо частота несучого сигналу кратна частоті модулюючого сигналу, тобто.

$$\omega = k\Omega,$$

k - ціле, то період модульованого сигналу виявляється таким, що дорівнює періоду модулюючого. Якщо частоти не кратні, але сумірні (їхнє відношення виражається раціональним числом), то період сигналу виявляється ще більшим, він буде в цілу кількість разів більшим за період низькочастотного модулюючого коливання. А якщо частоти несумірні (їхнє відношення не є раціональним числом), то модульований сигнал, строго кажучи, виявляється неперіодичним.

Зайве говорити, що з практичного погляду такий підхід абсолютно незручний; справжні частота і період розглянутого сигналу абсолютно не відображають його реальних властивостей. Водночас миттєва частота, яка в разі амплітудно-модульованого сигналу дорівнює частоті опорного сигналу, виявляється набагато об'єктивнішою та інформативнішою характеристикою сигналу.

Перейдемо тепер до питання про вимірювання частоти. У загальному випадку, вимірювання миттєвої частоти сигналу - досить складне завдання. Воно помітно спрощується, коли заздалегідь є інформація про характер сигналу (відомий вид функції, що описує сигнал). Тоді, відстежуючи миттєві значення сигналу й обробляючи ці дані (за допомогою аналогового ланцюга або цифровими методами), зможемо визначати миттєву частоту сигналу в будь-який момент. Отримана при цьому точність, з низки причин, часто виявляється не надто високою.

Дуже точному вимірюванню піддається середнє значення частоти сигналу, про це далі.

2.2. Середнє значення частоти. Вимірювання частоти

Нехай ми маємо сигнал виду

$$u(t) = A(t) \sin \varphi(t),$$

де $A(t) > 0$. Тоді точки, в яких сигнал проходить через нульові значення, визначаються тільки фазою. Візьмемо дві послідовні точки t_1 і t_2 такі, що в цих точках

$$\sin \varphi(t) = 0,$$

а значить і

$$u(t) = 0,$$

причому $u(t)$ проходить через нульове значення в цих точках в одному напрямку. Наприклад, від від'ємних значень до позитивних. Це означає, що $\sin \varphi$ в цих точках має однаковий знак, у нашому випадку - обидва значення позитивні.

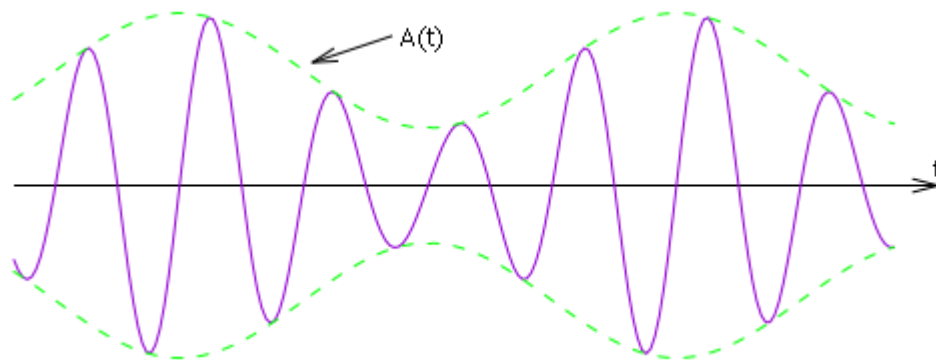


Рис 2.4 Періодичний сигнал

З урахуванням властивостей періодичності тригонометричних функцій, можемо стверджувати, що значення фази в зазначених точках відрізняються на 2π :

$$\varphi(t_2) - \varphi(t_1) = 2\pi.$$

Оскільки миттєва циклічна частота $\omega = d\varphi/dt$, то ця ж різниця фаз може бути обчислена інтегруванням миттєвої частоти:

$$\varphi(t_2) - \varphi(t_1) = \int_{t_1}^{t_2} \frac{d\varphi}{dt} dt = \int_{t_1}^{t_2} \omega dt.$$

А оскільки $\varphi(t_2) - \varphi(t_1) = 2\pi$, то і

$$\int_{t_1}^{t_2} \omega dt = 2\pi$$

Далі скористаємося тим, що інтегрування функції та знаходження її середнього - взаємопов'язані операції. За визначенням, середнє значення x на відрізка $[t_1, t_2]$ дорівнює

$$x = \frac{1}{\Delta t} \int_{t_1}^{t_2} x dt,$$

$$\Delta t = t_2 - t_1.$$

Тоді середнє значення циклічної частоти

$$\omega = \frac{1}{\Delta t} \int_{t_1}^{t_2} \omega dt = \frac{2\pi}{\Delta t}.$$

Середнє значення частоти

Для того, щоб виміряти середнє значення частоти, достатньо виміряти проміжок часу між двома моментами, коли сигнал проходить через нульове значення (в одному напрямку). Утім, цей результат цілком відповідає інтуїтивному уявленню про період реального сигналу і співвідношенню між періодом і частотою.

Якщо сигнал високочастотний, то інтервал Δt стає дуже малим і його виявляється важко виміряти з високою точністю. Тоді виявляється вигідним узяти інтервал, за який фаза сигналу зростає не на 2π , а на кратне цій величині значення, тобто $2\pi m$, де m - ціле:

$$\Delta\varphi = 2\pi m$$

Інакше кажучи, вимірюємо тривалість не одного періоду сигналу, а тривалість m періодів. Тоді середня частота за відповідний інтервал часу становитиме

$$f = \frac{1}{2\pi\Delta t} \int_{t_1}^{t_2} \omega dt = \frac{2\pi m}{2\pi\Delta t},$$

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		23

$$f = \frac{m}{\Delta t}.$$

Інтервал вимірювання, Δt як і раніше, залежить від періоду сигналу, але тепер, змінюючи m , можемо впливати на тривалість інтервалу, наближаючи його з більшою або меншою точністю до бажаного значення.

Для точного вимірювання, Δt добре підходять цифрові методи. Покажемо, як виконати вимірювання. Перетворимо вихідний сигнал у цифровий таким чином, що значенням $u(t) < 0$ відповідатиме логічний 0, а значенням $u(t) \geq 0$ - логічна 1. Тоді моментам переходу вихідного сигналу через 0 від від'ємних значень до позитивних, відповідатимуть фронти отриманого цифрового сигналу.

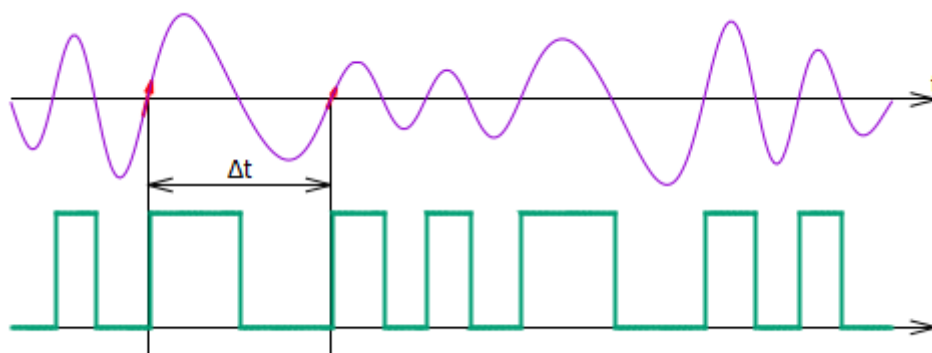


Рис 2.5 Принцип вимірювання частоти

Насправді, обов'язкова наявність гістерезису під час перетворення (поріг перемикання від 0 до 1 має бути вищим, ніж поріг зворотного перемикання). Інакше поблизу порога перемикання будемо отримувати пачки паразитних імпульсів через наявність шумів і перешкод у сигналі. Але це деталі реалізації, які не змінюють самого принципу.

Завдання визначення проміжку часу між двома заданими фронтами розв'язується дуже просто - за допомогою лічильника підраховують кількість імпульсів n еталонного генератора з частотою f_r (з періодом T_r) за цей проміжок часу (за першим фронтом сигналу лічбу запускають, за останнім - зупиняють). Тоді

$$\Delta t = nT_r = \frac{n}{f_r},$$

$$f = \frac{m}{\Delta t} = f_r \frac{m}{n}.$$

Ціле значення m задається точно. Підрахунок n виконується з точністю ± 1 , тому відносна похибка підрахунку, а отже, і похибка вимірювання f становить $1/n$ (без урахування похибки еталонного генератора). Для отримання якомога меншої відносної похибки вигідно, щоб значення n було якомога більшим. Збільшувати n можна, збільшуючи частоту еталонного генератора. Однак, на цьому шляху є обмеження, пов'язані з граничною швидкістю лічильника. Інший варіант - збільшувати тривалість інтервалу вимірювання, збільшуючи m . Цей підхід дає змогу досягти дуже високої точності вимірювань, але ціною збільшення тривалості вимірювання.

2.3. Динамічна похибка вимірювань

Існує спосіб визначення середньої частоти сигналу за деякий інтервал часу з високою точністю. Але якщо частота сигналу змінюється, середня частота дає занадто мало інформації про сигнал. Найчастіше буває необхідно знати, як у часі змінюється миттєва частота сигналу і наскільки сильно вона відхиляється від середнього значення.

Розглянемо, як співвідносяться середня і миттєва частота на прикладі сигналу, модульованого за частотою гармонійним сигналом:

$$\omega = \omega_0 + \Delta\omega \sin \Omega t.$$

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		25

Тут:

ω - миттєва частота модульованого сигналу;

ω_0 - його середня частота;

$\Delta\omega$ - найбільше відхилення миттєвої частоти від середнього значення;

Ω - частота модулюючого гармонійного сигналу.

Для простоти вважаємо, що початкова фаза модулювального впливу дорівнює 0.

Оскільки миттєва частота за визначенням $\omega = \frac{d\phi}{dt}$ то фаза модульованого сигналу описується виразом

$$\phi = \int \omega dt = \int (\omega_0 + \Delta\omega \sin \Omega t) dt = \omega_0 t - \frac{\Delta\omega}{\Omega} \cos \Omega t + \phi_0,$$

а сам сигнал може бути представлений у такій формі,

$$u(t) = A \sin(\omega_0 t - \frac{\Delta\omega}{\Omega} \cos \Omega t + \phi_0),$$

утім, у цьому випадку це не настільки важливо.

Подивимось, який результат дасть вимірювання середньої частоти сигналу

$$\omega = \frac{1}{\Delta t} \int_{t_1}^{t_2} \omega dt = \frac{1}{\Delta t} \int_{t_1}^{t_2} (\omega_0 + \Delta\omega \sin \Omega t) dt = \omega_0 - \frac{\Delta\omega}{\Omega \Delta t} (\cos \Omega \Delta t - \cos \Omega t_1),$$

де $\Delta t = t_2 - t_1$.

Враховуючи, що

$$\cos \alpha - \cos \beta = -2 \sin \frac{\alpha - \beta}{2} \sin \frac{\alpha + \beta}{2},$$

отримуємо

$$\omega = \omega_0 + \frac{2\Delta\omega}{\Omega \Delta t} \sin \frac{\Omega \Delta t}{2} \sin \frac{\Omega(t_1 + t_2)}{2}.$$

Величина

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		26

$$\frac{t_1 + t_2}{2}$$

є не що інше, як середина інтервалу вимірювання. Це значення можна перетворити до вигляду

$$\frac{t_1 + t_2}{2} = \frac{2t_2 - (t_2 - t_1)}{2} = t_2 - \frac{\Delta t}{2}.$$

Перетворили вираз таким чином, щоб цей момент часу виражався через t_2 , тобто той момент часу, коли стають відомі результати вимірювання.

Тоді

$$\omega = \omega_0 + \frac{2\Delta\omega}{\Omega\Delta t} \sin \frac{\Omega\Delta t}{2} \sin \Omega(t_2 - \frac{\Delta t}{2}).$$

Зауважимо, що якщо частота змінюється не за гармонійним законом, однаково, миттєву частоту як функцію можна розкласти на гармонійні складові та розглядати вплив операції усереднення на кожну складову окремо. Це можливо, оскільки операція усереднення є лінійною.

Розглянемо, як впливає інтервал вимірювання на результат вимірювань. Припустимо спочатку, що частота сигналу змінюється досить повільно, а інтервал вимірювання досить малий, так що,

$$\frac{\Omega\Delta t}{2} \ll 1,$$

а значить,

$$\sin \frac{\Omega\Delta t}{2} \approx \frac{\Omega\Delta t}{2}$$

і

$$\omega \approx \omega_0 + \Delta\omega \sin \Omega(t_2 - \frac{\Delta t}{2}).$$

Тобто, якщо обрано досить малий інтервал вимірювань, то результат вимірювань наближається до миттєвої частоти сигналу в момент, що відповідає середині інтервалу вимірювання (результат вимірювання запізнюється на $\frac{\Delta t}{2}$ відносно миттєвої частоти).

					МММТ.480.003.003 – ПЗ	Арк.
						27
Змн.	Арк.	№ докум.	Підпис	Дата		

Зі збільшенням інтервалу вимірювання зменшується коефіцієнт

$$\frac{2\Delta\omega}{\Omega\Delta t},$$

при цьому через властивості тригонометричних функцій,

$$\left| \sin \frac{\Omega\Delta t}{2} \right| \leq 1.$$

Так що, при

$$\Delta t \rightarrow \infty, \omega \rightarrow \omega_0.$$

На основі отриманих висновків можемо належним чином обрати інтервал вимірювання.

1. Якщо хочемо точно виміряти середню частоту, вплив відхилень миттєвої частоти на результат необхідно мінімізувати. Як було з'ясовано, для цього слід збільшувати інтервал вимірювання. Припустимо, що відносна похибка методу становить ε . Тоді, ймовірно, ми захочемо, щоб додаткова похибка, яку вносять коливання частоти, не перевищувала цієї величини:

$$\frac{2\Delta\omega}{\Omega\Delta t\omega_0} < \varepsilon,$$

звідки отримуємо

$$\Delta t > \frac{2\Delta\omega}{\Omega\omega_0\varepsilon}$$

або

$$\Delta t > \frac{\Delta f}{\pi F f_0 \varepsilon}.$$

2. Припустимо, що навпаки, потрібно відстежити зміни миттєвої частоти сигналу. Тоді має сенс вибирати малий інтервал вимірювання. Насамперед, щоб виявити зміну частоти сигналу, яка відбувається з частотою Ω , слід вибрати

$$\Delta t < \frac{2\pi}{\Omega} = \frac{1}{F},$$

тоді можна бути впевненим, що вимірювані відхилення не виявляться нульовими за рахунок множника $\sin(\Omega\Delta t/2)$

А якщо, припустимо,

$$\Delta t = \frac{1}{2F},$$

то

$$\omega^- = \omega_0 + c\Delta\omega\sin(t_2 - \frac{\Delta t}{2}),$$

де коефіцієнт $c \approx 0.64$, тобто вимірюване відхилення частоти від середнього значення становитиме не менше 60% від реального відхилення.

Якщо йдеться про досягнення якомога вищої точності у вимірюванні миттєвої частоти, то слід далі зменшувати інтервал вимірювання. Зі зменшенням інтервалу, результат дедалі більше наближається до миттєвої частоти, тобто зменшується динамічна похибка вимірювання, але водночас із цим зростає похибка методу. Це обмежує граничну точність вимірювання частоти шляхом вимірювання середньої частоти сигналу. Не має сенсу знижувати абсолютну динамічну похибку Δ_2 до значень менших за абсолютну похибку методу Δ_1 .

Похибка методу (відносна й абсолютна)

$$\varepsilon = \frac{1}{n} = \frac{1}{\Delta t_{fr}},$$

$$\Delta_1 = \varepsilon f_0 = \frac{f_0}{\Delta t_{fr}}.$$

Під динамічною похибкою розумітимемо різницю між найбільшим дійсним відхиленням ($\Delta\omega$ чи Δf) миттєвої частоти від середнього значення і

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		29

найбільшим відхиленням результату вимірювання від істинного значення середньої частоти. Для циклічної частоти похибка може бути обчислена як

$$\Delta\omega - \frac{2\Delta\omega}{\Omega\Delta t} \sin \frac{\Omega\Delta t}{2}$$

або просто для частоти:

$$\Delta 2 = \Delta f - \frac{\Delta f}{\pi F \Delta t} \sin(\pi F \Delta t).$$

За малих значень аргументу, синус може бути приблизно обчислений за формулою

$$\sin x \approx x - \frac{x^3}{6}.$$

Тоді

$$\Delta 2 \approx \Delta f - \Delta f + \frac{\Delta f}{\pi F \Delta t} \frac{(\pi F \Delta t)^3}{6},$$

$$\Delta 2 \approx \frac{\pi^2 \Delta f F^2 \Delta t^2}{6},$$

а оскільки ми вимагаємо, щоб

$$\Delta 2 \geq \Delta 1,$$

то можемо записати

$$\frac{\pi^2 \Delta f F^2 \Delta t^2}{6} \geq \frac{f_0}{\Delta t f r},$$

$$\Delta t \geq \sqrt[2]{\frac{6 f_0}{\pi^2 \Delta f F^2 f r}}.$$

Зрозуміло, що інтервал вимірювання при цьому не може бути меншим за період сигналу (що впливає із самого принципу вимірювання).

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		30

3. РОЗРОБКА СТРУКТУРНОЇ СХЕМИ

Цифровий частотомір - вимірювальний прилад, який показує значення вимірюваної частоти електричного сигналу в десятковій формі. Частотоміри здатні вимірювати значення частоти від декількох коливань за секунду (герц, Гц) до дуже високих частот порядку тисяч мегагерц (МГц). У частотомірах використовуються декадні лічильники. Принцип роботи такого частотоміра побудований на використанні лічильника, який накопичує кількість вхідних імпульсів за певний період часу. Після заданого періоду, відомого, як час затвора, значення лічильника передають на дисплей, і лічильник скидають у нуль. Структурну схему частотоміра наведено на рис. 3.1.

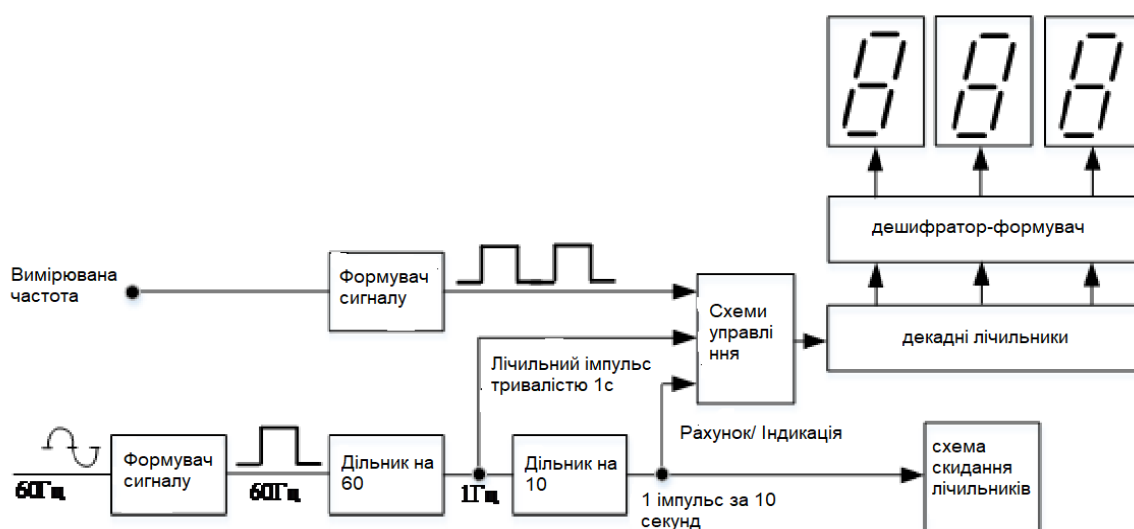


Рис 3.1 Структурна схема частотоміра

Тимчасові діаграми на рис. 3.2 дають точну ілюстрацію процесів усередині частотоміра. Діаграма А показує, що на вході керування

запуском/зупинкою лічильників спочатку (ліворуч на діаграмі) діє рівень логічного 0, потім цей вхід точно на 1 секунду переходить у стан логічної 1 і потім знову повертається до логічного 0. На діаграмі В показано безперервний ланцюжок імпульсів невідомої частоти, що надходять на вимірювальний вхід частотоміра.

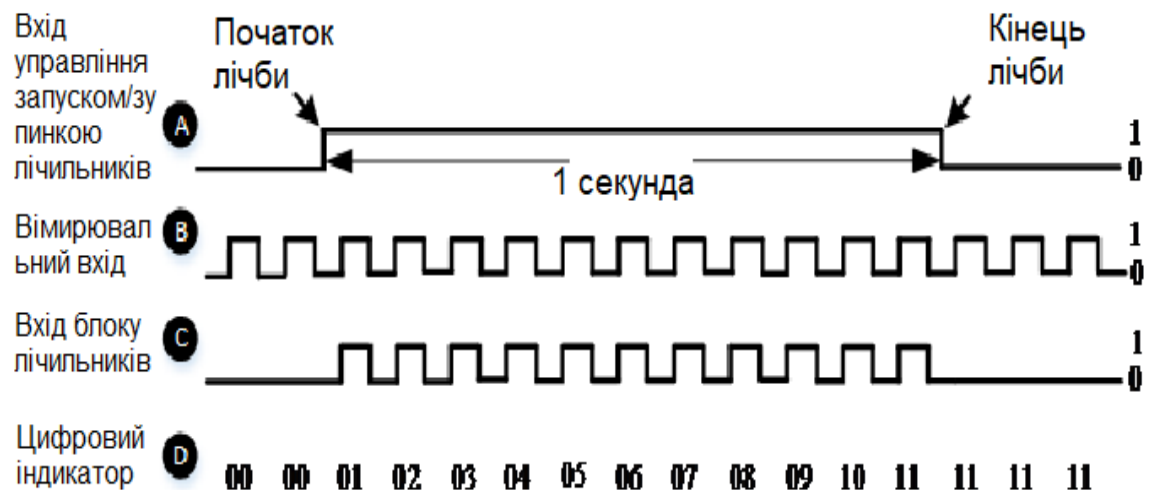


Рис. 3.3 Тимчасові діаграми для вимірювання частоти 11 Гц

Стан входу керування запуском/зупинкою лічильників і вимірювального входу пов'язані з логічною операцією "І", на рис. 3.3 представлена таблиця істинності елемента "І". На діаграмі С показано тільки ті імпульси, які пройшли через логічний елемент "І". Ці імпульси перемикають лічильники. Діаграма D дає часову розгортку десяткових чисел, що послідовно з'являються на індикаторах. Зверніть увагу, що "лічба" на індикаторах починається зі стану 00 (у вихідному стані індикатори очищені). Потім протягом 1 секунди вони

встигають "порахувати" до 11. Значення частоти вимірюваного сигналу (представленого на рисунку 3.2) дорівнює, отже, 11 Гц (11 імпульсів/с).

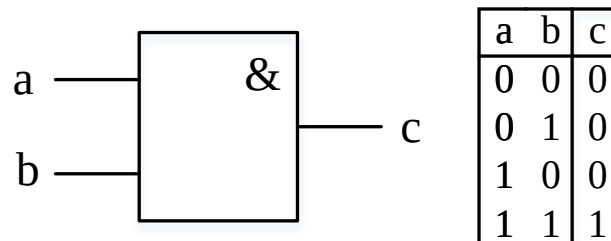


Рис 3.3 Елемент «І» та його таблиця істинності

4. РОЗРОБКА ПРИНЦИПОВОЇ СХЕМИ

Отже, для вимірювання частоти вхідного сигналу, беремо інтервал часу, що містить m періодів сигналу і, дивимось, яка кількість n імпульсів еталонного генератора з частотою F_0 укладається в цей інтервал часу. Частоту сигналу визначаємо як

$$F = \frac{1}{T} = F_0 \frac{m}{n}.$$

У процесі вимірювань потрібно виділяти один або кілька періодів вхідного сигналу. Для цього виберемо деякі характерні точки сигналу, які приймемо за початок періодів. Тоді період буде укладений між двома послідовними такими точками (між початком цього періоду і початком наступного). Для цифрового (імпульсного) сигналу природно прийняти фронт або спад сигналу за початок періоду. Домовимось, для визначеності, що далі початком періоду вважатимемо фронт сигналу (рис. 4.1).

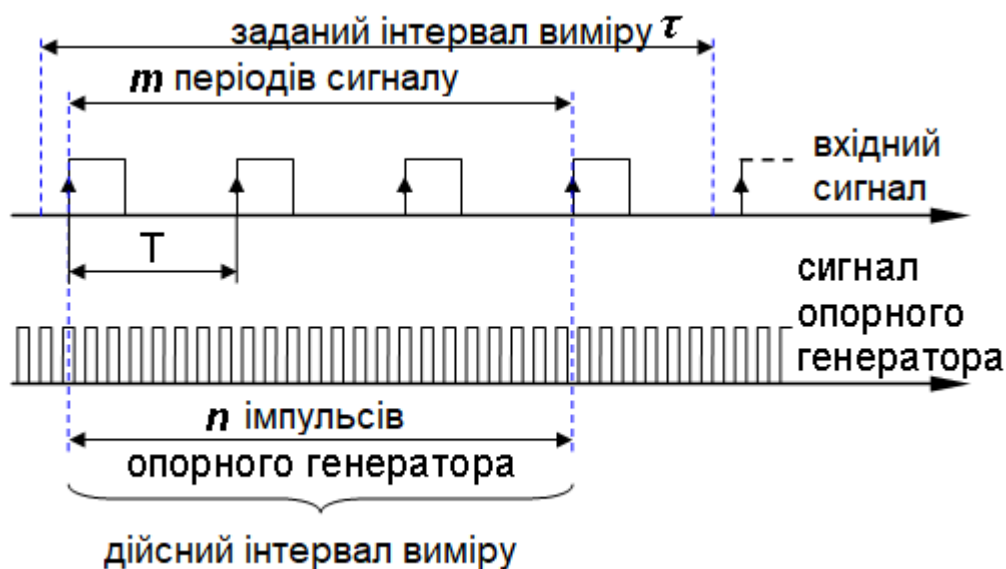


Рис 4.1. Принцип вимірювання частоти

Перед виконанням вимірювання слід задати бажаний інтервал вимірювання t , тобто час, що відводиться на один вимір. Цю величину можна змінювати в широких межах, залежно від вимог до швидкості вимірювання і точності результату. Відносна похибка вимірювання без урахування помилки, що виникає через нестабільність еталонного генератора (тільки похибка лічби) може бути оцінена як

$$\varepsilon = 1/n \approx \frac{1}{tF_0}.$$

Тут врахується, що абсолютна точність підрахунку n становить $+1$. Як видно, збільшення інтервалу вимірювання знижує відносну похибку підрахунку (цього ж можна домогтися за незмінного інтервалу, збільшуючи частоту еталонного генератора).

Також від обраного інтервалу вимірювання залежить мінімальна частота сигналу, яку частотомір здатен виміряти (нижня межа вимірювання). Для гарантовано успішного вимірювання, інтервал вимірювання має бути хоча б удвічі більшим за період вимірюваного сигналу (для більшої надійності краще, якщо умова виконується із запасом), тобто.

$$t_{\min} > 2/F_{\min}.$$

При виконанні цієї умови, незалежно від початкової фази вхідного сигналу, в інтервал вимірювання потрапить щонайменше два фронти сигналу (що дасть змогу виділити хоча б один період сигналу). Наприклад, за нижньої межі 10 Гц, інтервал вимірювання має бути не менше 0.2 с. А при виборі інтервалу в 1 мс (для проведення високошвидкісних вимірювань), нижня межа виміру складе 2000 Гц.

Особливістю методу є те, що дійсний інтервал вимірювання, впродовж якого ведеться відлік імпульсів опорного (еталонного) генератора, обирають строго кратним цілій кількості m періодів вхідного сигналу. Тому *дійсний* інтервал вимірювання не витримується точно рівним *заданому* інтервалу.

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		35

Дійсний інтервал трохи відхиляється від заданого у більший або менший бік (залежно від реалізації) і в невеликих межах може варіюватися від одного вимірювання до іншого. В ідеалі, розбіжність між заданим і дійсним інтервалом не перевищує одного періоду вхідного сигналу і найчастіше є незначною, особливо під час вимірювання високих частот.

У заданий інтервал вимірювання потрапляють 4 фронти сигналу, між якими укладено $m = 3$ повних періодів сигналу, які й утворюють дійсний інтервал.

Немає необхідності заздалегідь задавати конкретну величину для m , значення m послідовно збільшується на 1 під час вимірювання доти, доки тривалість вимірювання не виявиться приблизно рівною заданому бажаному інтервалу, після чого вимірювання завершується. Тим самим не потрібно виконувати будь-яких перемикання меж вимірювань; будь-яка частота вимірюється без жодного переналаштування приладу для вимірювання частоти, значення m визначається під час вимірювання.

Отже, після завершивши вимірювання буде визначено дві величини: кількість періодів вхідного сигналу m , що визначають час лічби, і кількість імпульсів n опорного генератора за цей час. Знаючи частоту опорного генератора, можна розрахувати частоту сигналу.

Таймери мікроконтролерів (в тому числі STM32) чудово підходять для реалізації описаної вище ідеї вимірювання частоти. Таймери загального призначення цих мікроконтролерів мають так звані канали, які здатні фіксувати поточне значення лічильника таймера за фронтом зовнішнього сигналу. Принцип вимірювання частоти за допомогою таймерів мікроконтролера наведено на рис. 4.2.

Якщо спрощено описувати процес вимірювання, то все що потрібно, це налаштувати таймер і запустити його. При цьому, залежно від налаштувань,

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		36

на рахунковий вхід таймера може бути подано як сигнал від окремого зовнішнього опорного генератора, так і тактовий сигнал з шини. Тобто, у найпростішому випадку, можна обійтися тактовим сигналом мікроконтролера для використання як сигналу еталонного генератора.

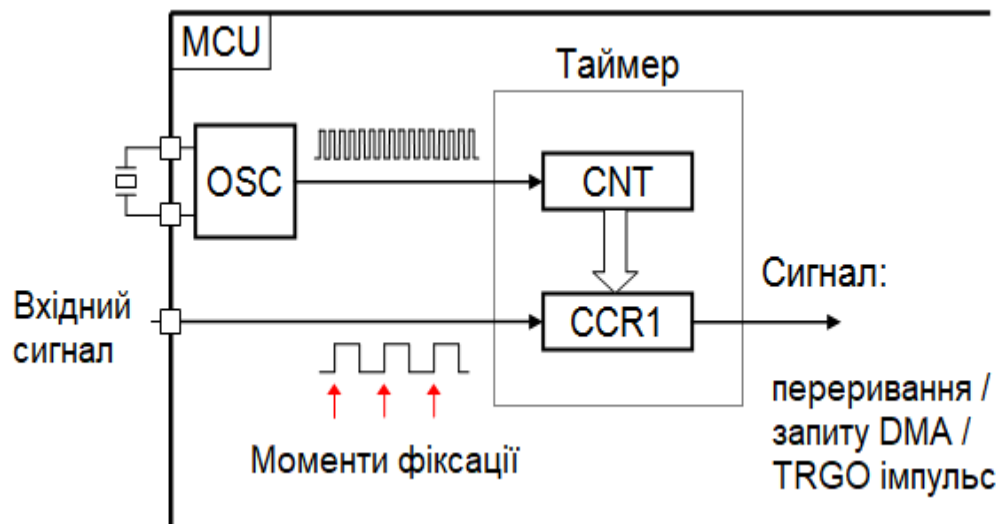


Рис 4.2. Принцип вимірювання частоти за допомогою таймерів мікроконтролера

За кожним фронтом вхідного сигналу таймер фіксуватиме поточне значення лічильника в спеціалізованому регістрі каналу. Необхідно зберегти результат першої фіксації $n1$, потім дочекатися, поки мине інтервал вимірювання, зберегти результат останньої фіксації $n2$ і загальну кількість фіксацій c . Тоді кількість імпульсів від першої до останньої фіксації становитиме $n2 - n1$. Кількість періодів вхідного сигналу між першою та останньою фіксацією дорівнюватиме $c - 1$ (оскільки кількість відрізків розбиття на 1 менша за кількість точок розбиття). Частоту сигналу обчислюємо як

$$F = F_0 \frac{c - 1}{n2 - n1}.$$

Якщо $c < 2$, значить, що не вдалося виділити повний період за обраний інтервал вимірювання (занадто низька частота вхідного сигналу для даного інтервалу вимірювання).

Розглянутий частотомір гранично будується за дуже простою схемою. Він складається з мікроконтролера і його мінімальної обв'язки, що забезпечує живлення і генерацію тактових сигналів (рис. 4.3). Вхід приладу - вивід PA1 мікроконтролера; вхід цифровий (TTL / CMOS 3.3V сумісний). Для роботи з аналоговим сигналом, слід перетворити сигнал в імпульсний, доповнивши схему блоком формування імпульсного сигналу з аналогового. Для цього достатньо скористатися будь-яким відповідним за частотними параметрами підсилювачем-обмежувачем; у деяких випадках аналоговий сигнал може подаватися на вхід PA1 безпосередньо, з урахуванням того, що на входах мікроконтролера є власні тригери Шмітта. Отже, можна вважати, що досліджуваний сигнал спочатку імпульсний або вже перетворений на імпульсний і необхідно зосередитися на цифровій частині приладу.

Якщо ж використовувати оціночну плату, яка вже містить усю мінімальну обв'язку, то схема буде зовсім простою.

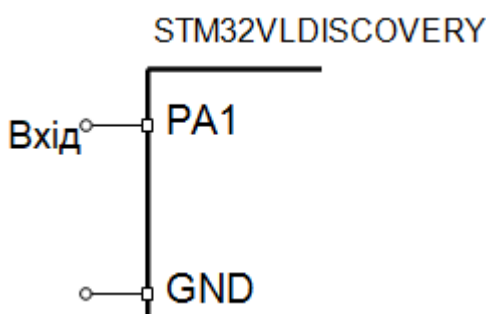


Рис 4.3. Організація підрахунку імпульсів

Більший інтерес представляє структурна схема пристрою, що враховує взаємодію використовуваних під час роботи частотоміра периферійних пристроїв мікроконтролера (рис. 4.4).

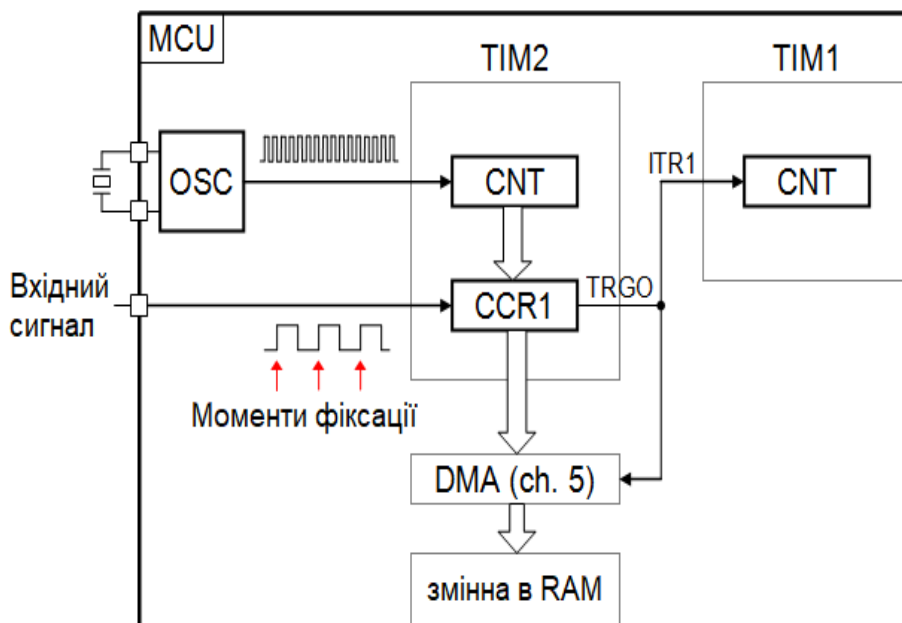


Рис 4.4. Принцип дій частотоміра

На рис. 4.4 блок OSC умовно зображує стабілізовану кварцовим резонатором систему генерації тактового сигналу мікроконтролера. Імпульси тактового сигналу підраховуються лічильником у таймері TIM2. За кожним фронтом вхідного сигналу, частота якого вимірюється, значення лічильника таймера TIM2 фіксується в регістрі CCR1 першого каналу цього таймера. Таймер TIM2 налаштований таким чином, що при кожній фіксації він формує імпульс на своєму внутрішньому виході TRGO. Таймери TIM2, TIM1 сконфігуровані для спільної роботи; TIM2 є ведучим, TIM1 - веденим. Таймер TIM1 підраховує імпульси на виході TRGO таймера TIM2, тобто кількість фіксацій у його каналі (інакше кажучи, кількість фронтів вхідного сигналу). Одночасно сигнал TRGO таймера TIM2 використовується як запит на пересилання даних за допомогою DMA. Система DMA налаштовується на

пересилання єдиного значення: перше зафіксоване в каналі значення зберігається в призначеній для цього змінній у пам'яті, після чого DMA автоматично зупиняється.

Використання зазначених таймерів не є єдино можливим; можуть бути обрані будь-які інші таймери, за умови, що таймер, керований вхідним сигналом, має щонайменше два канали (щонайменше три, якщо цей самий таймер використовується для апаратного формування інтервалу вимірювання), а інший таймер може функціонувати як підлеглий для нього.

Використовувана схема визначає верхню межу вимірювання приладу: максимальна вимірювана частота становить приблизно 10 МГц, за умови, що сигнал близький до меандру, тобто тривалість імпульсів не надто сильно відрізняється від тривалості пауз між ними, інакше кажучи, шпаруватість близька до 2 (відповідно, коефіцієнт заповнення близький до 0.5).

Обмеження пов'язане з тим, що як вхід частотоміра використовується вхід таймера, який керує каналом таймера. Вхід таймера має схему ресинхронізації, завдання якої - прив'язати моменти перемикання вхідного сигналу до тактового сигналу. Якщо проміжок часу між двома послідовними перемиканнями рівня сигналу становить менше періоду тактового сигналу таймера, то схема ресинхронізації може "не помітити" перемикання. Щоб цього не відбувалося, як тривалість вхідного імпульсу, так і тривалість паузи між імпульсами мають бути більшими за період тактового сигналу. Тоді гарантується робота без пропуску імпульсів. Наслідком зазначеної умови є те, що період вхідного сигналу не повинен бути меншим за два періоди тактового сигналу, а частота сигналу, відповідно, не повинна бути більшою за половину тактової частоти (12 МГц за тактової частоти 24 МГц, але краще із запасом, тому ми обираємо значення 10 МГц). Слід мати на увазі, що обмеження за частотою є орієнтовним і не є достатньою умовою нормальної роботи. Як було зазначено, важливо, щоб як тривалість імпульсу, так і паузи

					МММТ.480.003.003 – ПЗ	Арк.
						40
Змн.	Арк.	№ докум.	Підпис	Дата		

між імпульсами були більшими за період тактового сигналу; якщо тривалості сильно відрізняються, максимальна частота буде набагато меншою.

Збільшити верхню межу вимірювання можна, увімкнувши на вході частотоміра один або кілька послідовно з'єднаних Т-тригерів. Крім того, що кожен тригер удвічі знижує частоту, він також забезпечує вирівнювання тривалості імпульсу та тривалості паузи, принаймні для регулярного вхідного сигналу, без різких стрибків частоти.

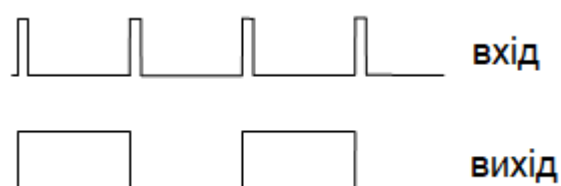


Рис 4.5. Принцип вимірювання частоти

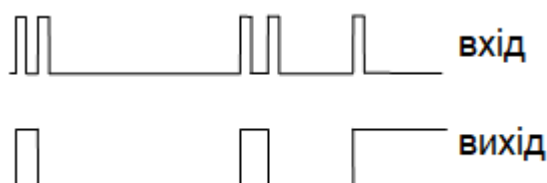


Рис 4.6. Принцип вимірювання частоти

Було б вигідно використовувати як вхід частотоміра вхід ETR таймера, на якому встановлено асинхронний керований дільник частоти. Тоді без проблем отримали б верхню межу, що дорівнює тактовій частоті мікроконтролера (24 МГц у цьому випадку), без суворих обмежень на шпаруватість імпульсів. Але вхід ETR не може керувати фіксацією в каналах. Вхід ETR все ж таки можна використовувати, але тоді доведеться задіяти

додатковий таймер, який на основі сигналу зі свого ETR входу за допомогою свого каналу формуватиме сигнал на виході, і вже частота сигналу з цього виходу вимірюватиметься так, як це було описано вище.

.

					МММТ.480.003.003 – ПЗ	Арк.
						42
Змн.	Арк.	№ докум.	Підпис	Дата		

5. КОНСТРУКТОРСЬКА ЧАСТИНА

Процеси при якому знаходять оптимальне розміщення елементів на друкованій платі отримав назву компоновання. Очевидно, що з'єднання між частинами схема є одним із найважливіших критеріїв, виконання мінімуму забезпечує мінімізацію взаємних наведень, спрощення конструкції та підвищує надійність. Компоновання у кваліфікаційній роботі виконувалось за допомогою графічних редакторів та програми Layout.

Друкована плата для підключення індикатора наведена на рис. 5.1.

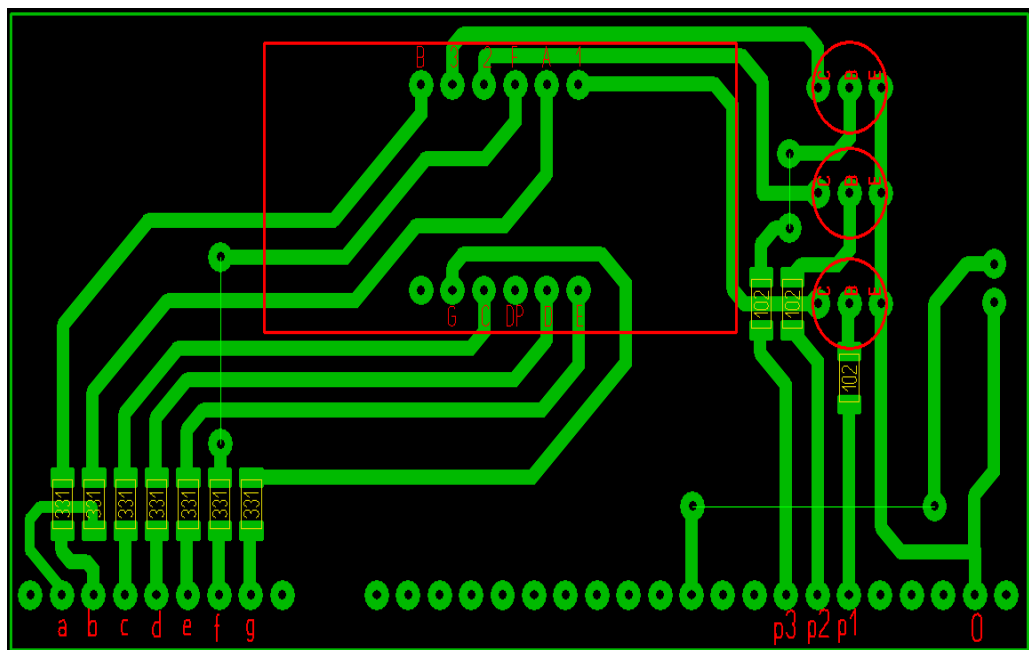


Рис 5.1 Трасування плати в Layout

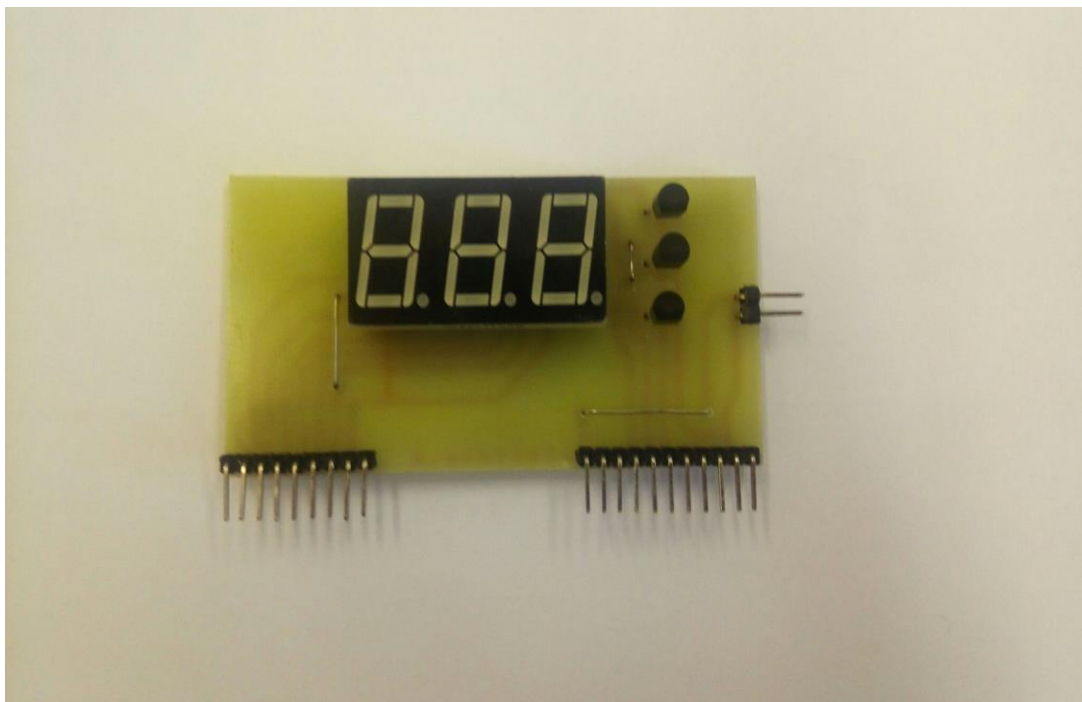


Рис 5.2 Зовнішній вигляд плати

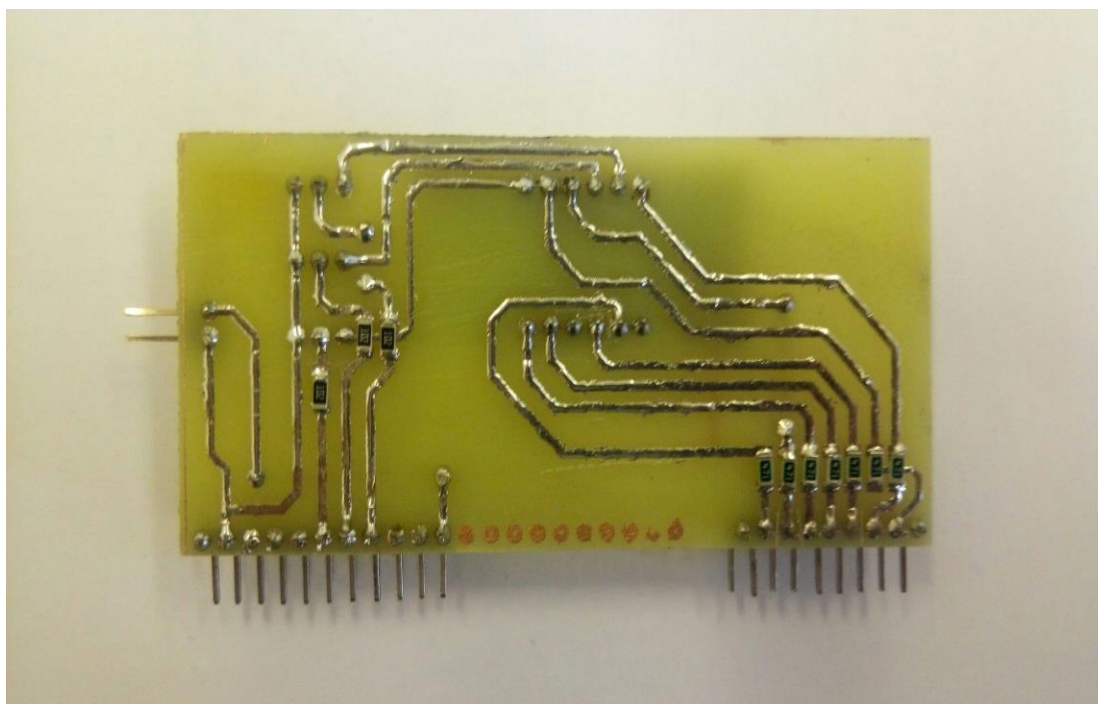


Рис 5.3 Зовнішній вигляд плати із зовнішнього боку

6. РОЗРОБКА АЛГОРИТМУ ТА ПРОГРАМИ РОБОТИ СИСТЕМИ

6.1. Алгоритм роботи системи

Алгоритм являє собою блок-схему, у якої обов'язково є початок і кінець.

Для початку роботи потрібно підключити стандартні бібліотеки. Це бібліотека повного функціоналу мікроконтролера і бібліотека функції затримок.

Потім налаштувати динамічне управління, для управління розрядами тисяч, сотень, десятків і одиниць.

Далі слід налаштувати такі параметри мікроконтролера, як вибір режиму генератора, увімкнути/вимкнути сторожовий таймер, дозволити/заборонити перемикання джерела такого сигналу, дозволити/заборонити роботу таймера увімкнення живлення, дозволити/заборонити скидання мікроконтролера за зниженням напруги живлення, дозволити/заборонити скидання мікроконтролера в разі переповнення стека, дозволити/заборонити низьковольтне програмування.

Потім потрібно ввести змінні для підрахунку вхідних імпульсів і самої частоти електричного струму.

Слідом вводимо масив для виведення цифр на екран. Кожна цифра семисегментного індикатора - це набір із семи 0 і 1.

Потім слід налаштувати підпрограму переривань, тобто визначити переривання високого рівня, дати зрозуміти мікроконтролеру куди слід переходити, якщо здійснювся прапор переривання, а так само куди повернутися після виконання переривання.

Після слід налаштувати порти введення і виведення сигналів, деякі ніжки потрібно налаштувати на введення інформації в мікроконтролер, а деякі на виведення.

Далі налаштовуємо переривання за таймером 0, для підрахунку імпульсів, що надходять у мікроконтролер протягом 1 секунди, а потім таймером 1, який через кожні 5 мс перемикає розряд динамічної індикації. Після дозволяємо глобальне переривання і на цьому алгоритм зациклюється. Алгоритм основної функції наведено на рис. 6.1.



Рис. 6.1. Алгоритм основної функції

Алгоритми переривання тісно пов'язані один з одним, тому використовуються переходи з алгоритму на алгоритм.

Алгоритм підрахунку вхідного сигналу по ніжці RB0 представлений на рис. 6.2. Якщо на ніжку RB0 прийшов сигнал, то прапор переривання скидається (програмно) і значення counter стає більшим на одиницю. Значення counter зростатиме до того моменту, поки не відбудеться переривання за таймером 0.

У таймер 0 записано значення, за якого тільки рівно через 1 секунду відбудеться переривання, після чого прапорець переривання за таймером 0 обнуляється і в нього знову записується число, яке змушує його через секунду перерватися.

Слідом ми обнуляємо змінні thousand, hundred, ten і one. Якщо значення counter менше тисячі, то переходимо нижче на два блоки. Якщо змінна counter набула значення більше 1000, то від цього значення віднімаємо тисячу і змінну thousand збільшуємо на одиницю. Потім перевіряємо, чи не стало значення змінної counter меншим чи дорівнює тисячі, якщо дорівнює або більшим, то операції, описані вище, повторюємо, до того моменту, доки значення не стане меншим за тисячу.

Коли значення counter стає меншим за тисячу, перевіряємо, чи більше воно за сотню, якщо менше, то переходимо на два блоки нижче, якщо ні, то віднімаємо від значення counter число 100, при цьому збільшуючи змінну hundred на одиницю. Це відбувається доти, доки значення counter не стане меншим за 100.

Далі перевіряємо, щоб значення counter було більшим за десять, якщо умова не виконується, то потрібно перейти на два блоки нижче, а якщо виконується, то віднімаємо від значення counter число 10, при цьому збільшуючи значення змінної ten на одиницю. Це відбувається доти, доки значення counter не стане меншим за 10.

					МММТ.480.003.003 – ПЗ	Арк.
						47
Змн.	Арк.	№ докум.	Підпис	Дата		

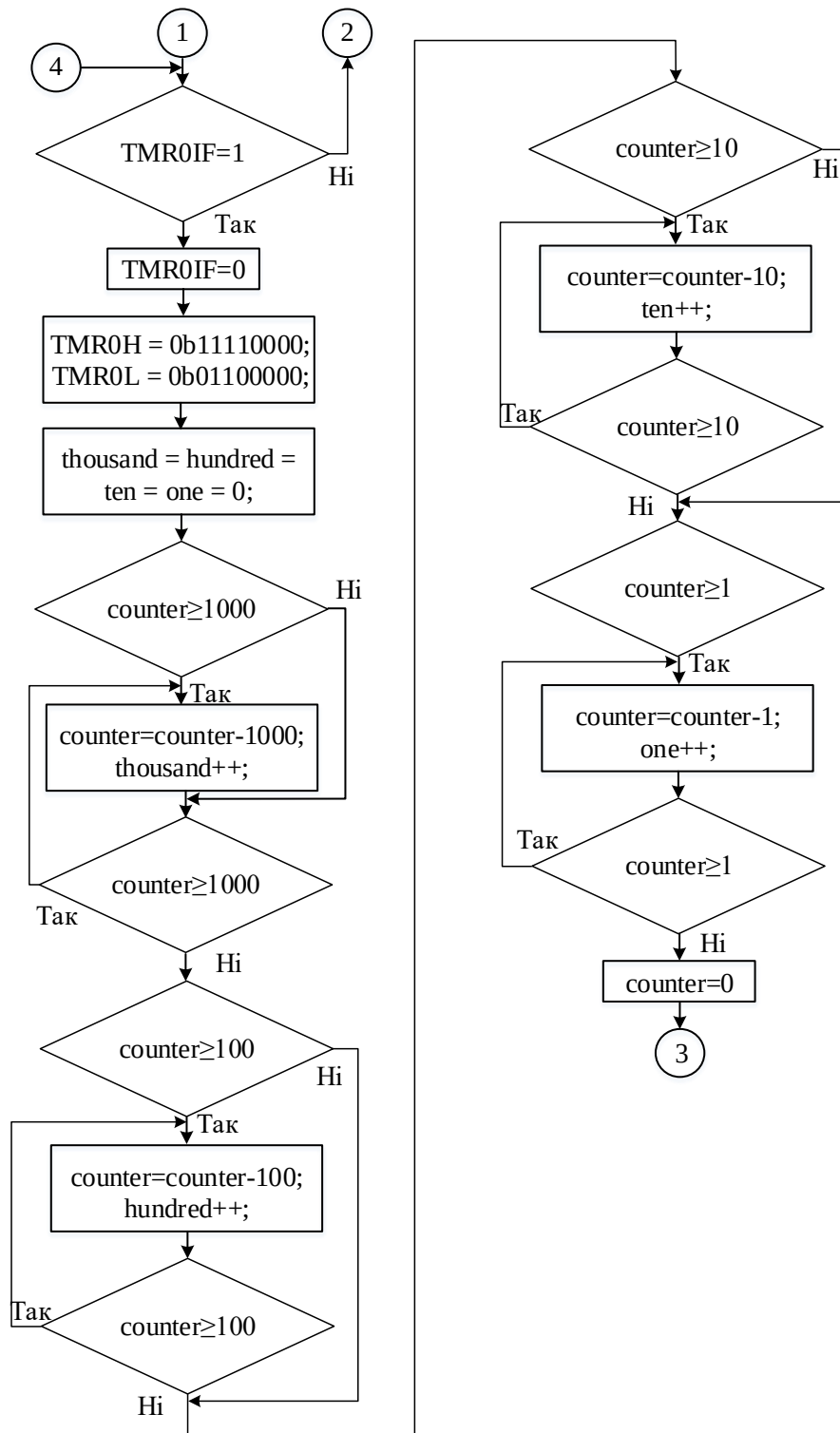


Рис 6.2 Алгоритм підрахунку вхідного сигналу по ніжці RB0

Коли це сталося, потрібно перевірити, чи більше значення counter одиниці чи ні, якщо ні, то значення counter обнуляємо і переходимо до виведення на екран частоти. Якщо більше одиниці, то рахуємо так само, як

описувалося вище. Алгоритм переривання за таймером 0, який відповідає за підрахунок розрядів динамічної індикації, представлений на рисунку 6.3.

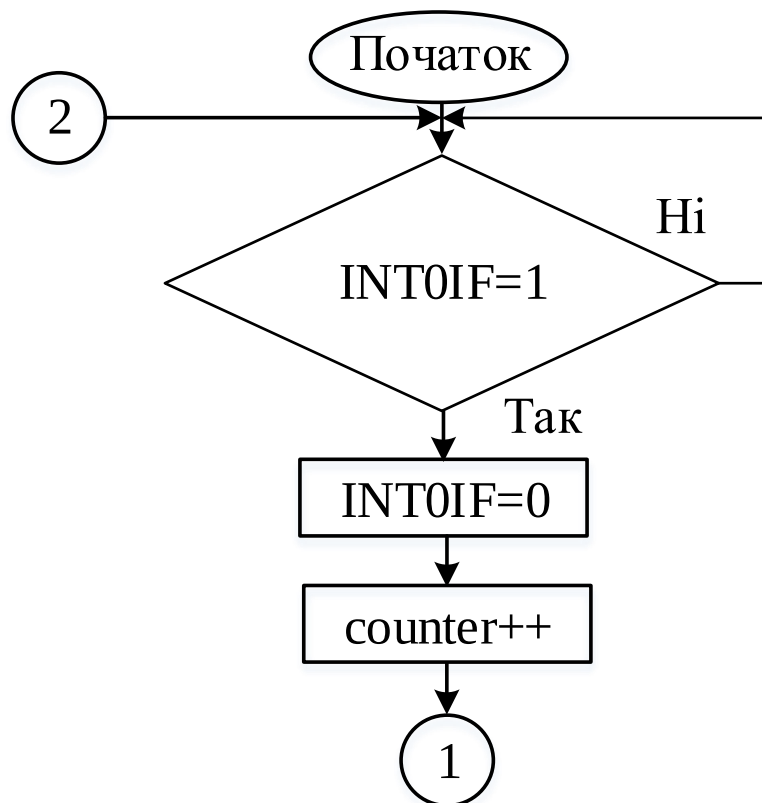


Рис 6.3. Алгоритм переривання за таймером 0

Для виведення на екран використовується таймер 1. Якщо спрацював прапор переривання за переповненням таймера 1, то прапор скидається, у таймер записується число, що дорівнює 5 мс, а потім по черзі виводяться на екран розряди thousand, hundred, ten, one. Алгоритм переривання за таймером 1, який відповідає за виведення на динамічний індикатор, представлений на рис. 6.4.

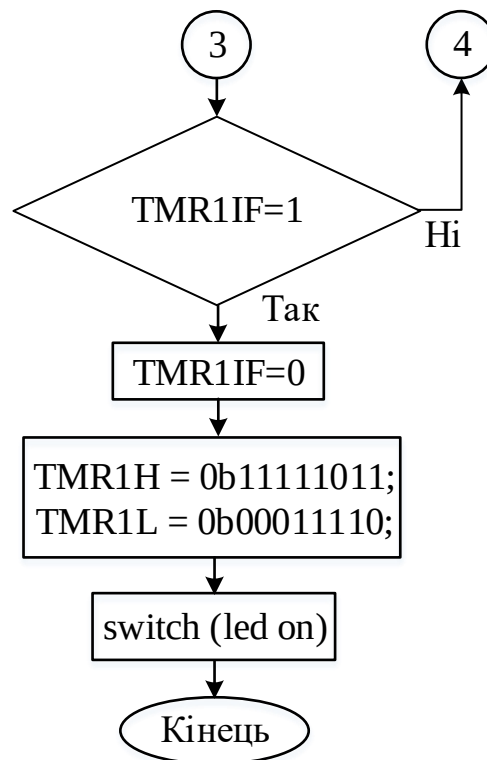


Рис 6.3. Алгоритм переривання за таймером 1

6.2. Програма роботи для частотоміра на STM32

Отже, як основний таймер, який рахує імпульси опорного генератора і виконує фіксацію значень свого лічильника за вхідним сигналом, обрано TIM2. Для керування першим каналом таймера використовується вивід PA0 або PA1. Будемо використовувати PA1 (у налагоджувальній платі PA0 зайнятий кнопкою). Зауважимо, що PA1 вивід пов'язаний із входом TI2 таймера TIM2, але сигнал TI2 за допомогою налаштувань таймера можна перенаправити на канал 1.

Вибір першого каналу для фіксації зумовлений тим, що таймер можна налаштувати на формування тригерного імпульсу на внутрішньому виході

TRGO таймера саме у відповідь на фіксацію в першому каналі (для цього слід задати бітовому полю MMS у регістрі TIM2->CR2 значення 0x3).

Другий канал таймера використовуємо для генерації додаткового переривання, що передує перериванню за переповненням лічильника. Канал працює в режимі порівняння, у його регістр CCR2 записується значення $(ARR+1)/2$, за замовчуванням 0x8000. Це означає, що додаткове переривання виникатиме під час досягнення лічильником половини від переповнюючого значення (тобто переривання виникає якраз посередині між переповненнями).

Використання таймера TIM2 і його каналу 1 для фіксації диктує вибір використовуваного каналу DMA, це буде канал 5 пристрою DMA1.

Для підрахунку кількості фіксацій у каналі 1 таймера TIM2 можна використовувати будь-який таймер, здатний працювати як ведений для TIM2. Наприклад, нехай це буде TIM1.

Початкова ініціалізація обладнання виконується у функції `fm_init`, яка викликається на початку `main`. Тут виконується налаштування NVIC (задається пріоритет використовуваних переривань і дозволяється їхнє опрацювання); вмикається тактовий сигнал для задіяних у роботі периферійних пристроїв (DMA1, TIM1, TIM2, GPIOA); виконується базове налаштування таймерів та DMA (задаються ті налаштування, що залишатимуться незмінними впродовж усієї роботи).

Перш ніж заглиблюватися в опис деталей програмної реалізації для використовуваного методу вимірювання частоти, коротко обговоримо, як запустити вимірювання й отримати результат. Це може бути корисно, якщо передбачається використовувати пропонований код як частину будь-якого проекту.

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		51

Кожне вимірювання частоти вимагає виклику двох функцій: `measure` та `get_result` (до першого вимірювання слід викликати `fm_init()` для початкової ініціалізації засобів вимірювання частоти). Виконання вимірювання у два кроки дає змогу звільнити мікроконтролер для виконання інших завдань на час вимірювання. Функція `measure` запускає вимірювання й одразу завершується, саме вимірювання відбувається у фоновому режимі. Поки триває вимірювання, можна зайняти мікроконтролер будь-якою іншою корисною роботою. Коли вимірювання завершиться, використовуємо `get_result` для зчитування результату.

Розглянемо докладніше використання кожної з цих функцій.

`boolmeasure(uint32_ticks)`

Аргумент `ticks` задає бажаний інтервал вимірювання в тактах (або періодах) сигналу на рахунковим вході таймера TIM2, тобто в періодах сигналу еталонного генератора. У цьому випадку - просто в періодах тактового сигналу мікроконтролера. Можливо, не дуже зручно те, що інтервал, заданий в одиницях часу, потрібно перетворювати на такти ($ticks = SystemCoreClock * t$, тут *SystemCoreClock* - частота тактового сигналу на рахунковий вхід таймера TIM2; *t* - заданий час вимірювання). Натомість зручно одразу оцінювати відносну похибку лічби, яка становить $1.0 / ticks$.

Наприклад, при тактовій частоті 24 МГц, щоб отримати інтервал вимірювання 1 с, слід задати `ticks = 24000000`. При цьому відносна похибка лічби виявиться не більше $5 \cdot 10^{-8}$ (при використанні типу `float` для обчислень, варто мати на увазі, що зазначена точність знаходиться вже на межі можливостей цього типу).

Значення для `ticks` за бажання можна задати дуже малим (аж до 0), але не варто при цьому розраховувати на те, що інтервал буде витримано з високою точністю. По-перше, хоча інтервал відраховується за допомогою таймера досить точно, але завершення вимірювання відбувається в

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		52

обробнику переривання. Від моменту спрацьовування таймера до моменту фактичного завершення вимірювання може минути близько 100 тактів (близько 4 мкс за тактової частоти мікроконтролера 24 МГц). На щастя, точність, з якою відміряється інтервал, не впливає на точність вимірювання (натомість на точність впливає абсолютне значення інтервалу вимірювання - що коротший інтервал, то гірша точність; не варто без особливої потреби задавати надкороткі інтервали вимірювання). По-друге, сам метод вимірювання передбачає, що дійсний інтервал вимірювання може відрізнятись від заданого (бо він завжди буде утворений цілою кількістю періодів сигналу, частоту якого вимірюємо).

Функція `measure` виконує ініціалізацію змінних, використовуваних під час вимірювання; оновлює налаштування таймерів і DMA; запускає таймери; дозволяє роботу каналів таймера TIM2. Тим самим вона запускає нове вимірювання частоти. Після цього функція одразу завершується, не чекаючи завершення вимірювання, яке відбуватиметься у фоновому режимі. Завдяки цьому, поки відбувається вимірювання частоти, можна виконувати іншу корисну роботу. Про завершення вимірювання можна дізнатися за встановленою в **true** глобальною змінною `fN2e`.

Функція `measure` повертає **true** у разі успішного запуску нового вимірювання. Якщо ж у момент виклику вже відбувається вимірювання (запущене раніше і таке, що ще не закінчилося), функція завершується, не втручаючись у поточний вимір, і повертає значення **false** (це свідчить про помилку використання функції - її було викликано несвоєчасно).

Коли вимірювання завершено і встановлено змінну-прапор `fN2e`, можна запросити результати вимірювання, для цього використовується функція

`intget_result(uint32_t&N1,uint32_t&N2).`

Функцію допускається викликати і достроково, не чекаючи завершення

					МММТ.480.003.003 – ПЗ	Арк.
						53
Змн.	Арк.	№ докум.	Підпис	Дата		

вимірювання (зокрема й одразу після виклику `measure`), але тоді вона чекатиме на установку `fN2e`, опитуючи стан змінної в програмному циклі. Якщо ми не хочемо, щоб функція призупиняла виконання програми, то маємо викликати її, тільки переконавшись, що змінна `fN2e` встановлена в `true`.

Повернуте функцією значення - код помилки:
0 - немає помилок; у `N1`, `N2` вміщено правильні значення.
-1 - не вдалося виконати вимірювання, тому що за інтервал вимірювання виявлено менше двох фронтів вхідного сигналу (для виконання вимірювання необхідно виділити хоча б один цілий період вхідного сигналу, а для цього потрібно отримати два фронти сигналу); причиною помилки є вибір надто малого інтервалу вимірювання для цієї частоти сигналу; для гарантовано успішного вимірювання частоти сигналу, обраний інтервал вимірювання повинен хоча б удвічі перевищувати максимально можливий період вхідного сигналу.

Результати вимірювання присвоюються аргументам функції `N1`, `N2`, які передаються за посиланням. У `N1` функція поміщає кількість періодів вхідного сигналу, що утворюють дійсний інтервал вимірювання, а в `N2` - кількість імпульсів опорного генератора (у цьому випадку - кількість імпульсів системного тактового сигналу) за цей інтервал.

Усе найважливіше, що стосується вимірювання, відбувається в обробнику переривання `TIM2_IRQHandler` від таймера `TIM2`, головного таймера частотоміра. Для взаємодії основної програми та обробника використовується досить велика кількість глобальних змінних

Правду кажучи, багато хто з них необхідний тільки для внутрішнього використання в обробнику і міг би бути визначений у ньому як статичні змінні. Але їм потрібна початкова ініціалізація перед запуском кожного виміру, і це простіше зробити поза обробником (обробник і без того містить

					МММТ.480.003.003 – ПЗ	Арк.
						54
Змн.	Арк.	№ докум.	Підпис	Дата		

більш ніж достатньо коду), тому змінні зроблено глобальними. Можна було б об'єднати всі ці дані й обробник в одному об'єкті, але це не скоротило б текст програми, а завдання створити універсальну бібліотеку не ставилося.

Коротко про призначення перерахованих змінних.

`volatile uint32_t N1_prev;`

Взагалі, змінні з іменами виду N1xxx використовуються для збереження результатів лічби таймера TIM1 у певні моменти часу (відповідно, змінні N2xxx належать до результатів лічби TIM2). Зокрема, N1_prev використовується для зберігання значення таймера TIM1, зчитаного під час попереднього виконання обробника переривання від TIM2. Змінна слугує для того, щоб обробник TIM2_IRQHandler міг визначити, чи відбувалися фіксації в каналі TIM2 (які й підраховує TIM1) з моменту попереднього зчитування значення лічильника TIM1 (під час попереднього виклику цього обробника). Для визначення, зчитуємо поточне значення лічильника, порівнюємо з попереднім із N1_prev. Якщо значення відрізняються - були фіксації, які потрібно врахувати в обробнику. Якщо не відрізняються - фіксацій не було. Після виконання перевірки, присвоюємо змінній N1_prev уже зчитане поточне значення, яке стане "попереднім" під час наступного виклику обробника.

Природно, під час визначення поточного результату рахунку таймера TIM1, потрібен облік переповнень лічильника. Виконується це просто: якщо поточне значення більше за попереднє, то переповнення не було і значення коректне; якщо поточне значення менше - переповнення було і потрібне коригування на коефіцієнт перерахунку лічильника в таймері TIM1, тобто на $TIM1 \rightarrow ARR + 1$ (за замовчуванням це $0xFFFFFFFF+1$). Завдяки такому алгоритму, під час виконання в обробнику фрагмента

`uint32_t N1 = N1_correction + TIM1->CNT;`

					МММТ.480.003.003 – ПЗ	Арк.
						55
Змн.	Арк.	№ докум.	Підпис	Дата		

який зчитує поточне (істинне) значення лічильника в TIM1, абсолютно не важливо, чи встиг обробник переповнення від TIM1 оновити змінну N1_correction, чи не встиг - у будь-якому разі, ми виявимо невраховане переповнення й у разі потреби скоригуємо значення.

Тепер про те, навіщо такі складнощі з виявленням фіксацій, якщо, як відомо, існує прапор, що встановлюється апаратно під час виконання фіксації (TIM2->SR, біт TIM_SR_CC1IF). Річ у тім, що, по-перше, при виконанні першої фіксації, відбувається пересилання даних засобами DMA і DMA самостійно скидає біт. Це вже вимагає введення додаткових програмних перевірок, крім перевірки прапора. По-друге, між читанням прапора і його скиданням (для виявлення наступних фіксацій), може відбутися ще одна (або навіть не одна) фіксація. Але скинемо прапор, і якщо після скидання не буде фіксацій, під час наступного виклику обробника, перевіривши прапор, помилково дійдемо висновку, що фіксацій не було. Якщо це відбувається наприкінці вимірювання (для останньої фіксації), вона не буде належним чином скоригована. Це дуже серйозна помилка, яка сильно спотворює результат, але виникає не завжди, а тому її важко виявити.

У пропонованому ж варіанті, фактично щоразу обчислюється "прапор" фіксації, тому пропуски неможливі: спочатку зчитується поточна кількість фіксацій і запам'ятовується, а потім виконуємо обробку фактично останньої. Можлива ситуація, коли після зчитування лічильника TIM1 і виявлення неврахованих фіксацій, відбуваються нові фіксації, і виконується обробка результату останньої з них. Проте під час наступного виклику обробника TIM2_IRQHandler буде знову виявлено наявність необроблених фіксацій, хоча фактично обробку було виконано. Але повторна обробка не становить жодної небезпеки, вона не змінює результату; важливо, що немає пропусків події.

volatile uint32_t N1_correction;

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		56

Змінна містить поправку для значення лічильника TIM1, що враховує переповнення лічильника (тобто в разі переповнення збільшується змінна відразу на коефіцієнт перерахунку лічильника); істинне значення кількості підрахованих імпульсів на цей момент дорівнюватиме

$N1_correction + TIM1 \rightarrow CNT;$

`volatile uint32_t N2_correction;`

Поправка для таймера лічильника TIM2; справжнє значення кількості імпульсів, підрахованих таймером, становить

$N2_correction + TIM2 \rightarrow CNT;$

`volatile uint32_t N2b_correction;`

Змінна містить коригувальне значення для результату першої фіксації (коригувальне значення для лічильника в TIM2 у той момент, коли відбувається перша фіксація). Змінна необхідна, оскільки за низькочастотного сигналу лічильник може встигнути переповнитися до того, як відбудеться перша фіксація.

`volatile uint32_t N2b_naked;`

У цю змінну система DMA поміщає результат першої фіксації, тобто значення лічильника таймера в момент, що відповідає першому фронту вхідного сигналу. Значення має бути скориговано з урахуванням переповнень лічильника. Істинне значення складе

$N2b_correction + N2b_naked;$

`volatile uint32_t N1e;`

Коли вимірювання завершується, обробник переривання TIM2_IRQHandler поміщає в цю змінну підраховану за інтервал вимірювання кількість фронтів вхідного сигналу (інакше кажучи, кількість фіксацій від першої врахованої до останньої, включно з першою та останньою). Це

					МММТ.480.003.003 – ПЗ	Арк.
						57
Змн.	Арк.	№ докум.	Підпис	Дата		

значення на 1 більше періодів вхідного сигналу, що утворюють істинний інтервал вимірювання.

```
volatile uint32_t N2e;
```

Кількість імпульсів опорного генератора, підрахованих таймером TIM2 на момент останньої врахованої фіксації (вірне значення, з урахуванням переповнень).

```
volatile bool fN2e;
```

Присвоюючи цій змінній значення true, обробник переривання TIM2_IRQHandler таким чином повідомляє основній програмі, що вимірювання завершено і результати готові для зчитування.

```
uint32_t time_slot;
```

Змінна містить заданий час вимірювання в тактах сигналу на лічильному вході TIM2.

Далі в тексті програми йде визначення функцій:

```
extern "C" void TIM1_UP_TIM16_IRQHandler();
```

```
extern "C" void TIM2_IRQHandler();
```

```
int get_result(uint32_t &N1, uint32_t &N2);
```

```
bool measure(uint32_t ticks);
```

```
void fm_init();
```

```
int main();
```

```
int dispaly_init();
```

```
int display(uint32_t fclk, uint32_t m, uint32_t n, uint32_t err);
```

```
extern "C" void TIM1_UP_TIM16_IRQHandler();
```

Функція є обробником переривань від таймера TIM1, вона просто враховує переповнення лічильника таймера, оновлюючи значення глобальної

					МММТ.480.003.003 – ПЗ	Арк.
						58
Змн.	Арк.	№ докум.	Підпис	Дата		

змінної N1_correction. Зауважимо, що змінна містить не кількість переповнень, а коригувальне значення, склавши яке з поточним значенням лічильника, одразу отримаємо істинне значення результату лічби. Оскільки значення, що переповнює лічильник, дорівнює $TIM1 \rightarrow ARR + 1$, то при кожному переповненні для оновлення змінної виконується код

```
N1_correction+=TIM1->ARR + 1;

extern "C" void TIM2_IRQHandler();
```

Функція є обробником переривань від таймера TIM2, на цю функцію покладено основну роботу з вимірювання частоти вхідного сигналу; вимірювання йде у фоновому режимі, тому одночасно з ним мікроконтролер може займатися іншими справами.

Під час налаштування NVIC для переривань від TIM2 задають вищий пріоритет, ніж для переривань від TIM1, для того, щоб у разі переповнення лічильника таймера TIM1 не переривалася робота обробника TIM2_IRQHandler.

Обробник TIM2_IRQHandler веде облік переповнень лічильника таймера TIM2, оновлюючи змінну N2_correction. Але це лише мала частина того, що робить обробник. Найголовніше його завдання - відстежувати фіксації, що вже відбулися в каналі 1 таймера, і зіставляти відповідні їм коригувальні значення, що враховують переповнення лічильника. Також обробник стежить за часом, що минув від початку вимірювання, і зупиняє вимірювання, коли заданий час спливає. Час може відміряти окремий таймер, але тут цю функцію виконує третій канал, який працює в режимі порівняння. Якщо слід зупинити вимірювання після ticks імпульсів на лічильному вході таймера, то з урахуванням переповнень, зрозуміло, що в цей момент у лічильнику буде значення $ticks \% 0x10000$ (залишок від ділення ticks на модуль перерахунку, тут воно також дорівнює $ticks \& 0xFFFFFFFF$). Тому перевіряти, чи не минув час, слід щоразу, коли значення в лічильнику стає

					МММТ.480.003.003 – ПЗ	Арк.
						59
Змн.	Арк.	№ докум.	Підпис	Дата		

рівним ticks & 0xFFFFFFFF. Для цього в регістр порівняння каналу 3 таймера, перед запуском вимірювання, поміщаємо значення ticks & 0xFFFFFFFF і, обробляючи переривання від каналу, порівнюємо час, що минув (у тактах), із заданим.

Якщо час на вимірювання минув, зупиняємо лічбу: спочатку вимикаємо канал 1 таймера, тим самим забороняючи подальше виконання фіксацій у каналі за фронтами вхідного сигналу, після чого можемо зупинити лічильники таймерів (із цього моменту можемо бути певними, що більше переповнення лічильників не відбуватимуться), опрацювати результати останньої фіксації й установити прапорець завершення вимірювання (привласнити true глобальній змінній fN2e).

Як уже зазначалося, найважливіше завдання обробника - визначати коригувальні значення (що враховують переповнення) для першої та останньої фіксації. Коригувальні значення визначаються щоразу, коли виявляється, що з моменту останнього виклику обробника змінилася кількість фронтів, підрахованих лічильником TIM1. Якщо переривання відбулося через "напівпереповнення", то, безсумнівно, останній фіксації відповідає поточне коригувальне значення. Якщо обробляється переривання за переповненням, уточнюємо, чи відбулася фіксація до переповнення, чи вже після (за величиною зафіксованого значення, як порівняти з поточним значенням лічильника). Якщо фіксація була після переповнення, за коригувальне значення для фіксації вибираємо поточне коригувальне значення, інакше - попереднє (до переповнення). Визначивши коригувальне значення, складаємо його з результатом останньої фіксації і зберігаємо результат у глобальній змінній N2e. Вимірювання може бути зупинено в будь-який момент, і остання врахована фіксація може виявитися взагалі останньою на інтервалі вимірювання. Але регулярне оновлення N2e гарантує, що змінна міститиме актуальне значення.

					МММТ.480.003.003 – ПЗ	Арк.
						60
Змн.	Арк.	№ докум.	Підпис	Дата		

Коригувальне значення першої фіксації визначається аналогічним чином (оскільки перша фіксація була виконана між попереднім викликом обробника і даними). Зрозуміло, це робиться тільки один раз за вимірювання. Обробник виявляє те, що перша фіксація відбулася за виконанням умови

$$N1_prev == 0 \ \&\& \ N1 \neq 0.$$

Для першої фіксації коригувальне значення і нескоригований результат фіксації зберігаються в окремі змінні (нескоригований результат фіксації зберігається за допомогою DMA).

```
int get_result(uint32_t &N1, uint32_t &N2);
```

Функція поміщає у свої аргументи, що передаються за посиланням:

N1 - кількість періодів вхідного сигналу, що утворюють дійсний інтервал вимірювання;

N2 - кількість імпульсів еталонного генератора (у нашому випадку - тактового сигналу мікроконтролера) за цей інтервал, утворений N1 періодами вхідного сигналу.

Повернуте значення:

0 - у разі успішного завершення;

-1 - у разі помилки, коли в заданий інтервал вимірювання не "помістився" навіть один період вхідного сигналу (період, обмежений двома послідовними фронтами); це означає, що обраний інтервал надто малий для вимірювання такої низької частоти. Для гарантії успішного вимірювання має бути

$$f_{min} > 2/\tau.$$

Якщо вимірювання не завершено, функція очікує його завершення для отримання результатів.

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		61

Значення N1, N2 функція визначає за результатами роботи обробника переривань від TIM2, які поміщаються обробником у глобальні змінні.

N1 - це просто зменшений на 1 результат лічби таймера TIM1 з урахуванням поправки на переповнення (лічильник підраховує фронти сигналу; кількість періодів, укладених між першим і останнім фронтом, буде на одиницю меншою; облік початкового значення не потрібен, оскільки лічильник від початку встановлюється на 0).

N2 визначається як різниця між результатами останньої і першої фіксації в каналі 1 таймера TIM2 (з урахуванням переповнень).

```
bool measure(uint32_t ticks);
```

Функція запускає новий вимір (якщо наразі не відбувається вимір, який ще не завершився) і одразу завершується, звільняючи мікроконтролер для виконання інших обчислень, поки триває вимір. Для запуску вимірювання, функція виконує ініціалізацію глобальних змінних, які використовуються в процесі роботи оброблювачів переривань від таймерів; виконує ті дії з ініціалізації обладнання, які потрібно виконувати щоразу наново (наприклад, щоразу потрібно задавати кількість переданих за допомогою DMA даних); дозволяє рахунок для таймерів; дозволяє роботу каналів таймера TIM2. З моменту ввімкнення каналів TIM2 вимірювання починається, а функція повертає керування програмі, що викликала.

```
void fm_init();
```

Функція відповідає за початкову ініціалізацію периферійних пристроїв мікроконтролера, задіяних у вимірюванні частоти. Викликається одноразово, до першого вимірювання. Наприклад, на початку основної функції main.

```
int main();
```

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		62

Основна функція програми. Виконує початкову ініціалізацію частотоміра шляхом виклику `fm_init`, потім починає циклічне вимірювання частоти з виведенням результатів кожного вимірювання на дисплей.

```
int dispaly_init();
```

```
int display(uint32_t fclk, uint32_t m, uint32_t n, uint32_t err);
```

Функції для початкової ініціалізації індикатора і виведення на нього результатів вимірювань. Тут вони тільки оголошені, але не визначені. Визначаються функції в окремому файлі; їхня реалізація залежить від використовуваного індикатора. Можна виводити результат на семисегментний індикатор достатньої розрядності; текстовий або графічний дисплей з інтерфейсом SPI або I2C; можна передавати дані на комп'ютер, наприклад, через UART.

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		63

7. ОЦІНКА ПОКАЗНИКІВ НАДІЙНОСТІ РОБОТИ СИСТЕМИ

При розрахунку надійності системи потрібно визначити такі показники:

- ймовірність безвідмовної роботи системи за заданий час ($P(t)$);
- параметр потоку відмов (Λ);
- середнє напрацювання до відмови ($T_{\text{серед}}$).

Надійність – властивість об'єкта виконувати задані функції, зберігаючи в часі значення встановлених експлуатаційних показників в заданих межах.

Ймовірність безвідмовної роботи – ймовірність того, що в межах вказаного періоду часу відмова пристрою не виникає.

Відмова – подія, яка полягає в порушенні працездатного стану об'єкта.

Середнє напрацювання до відмови – це математичне очікування напрацювання об'єкта до першої відмови.

Інтенсивність відмов – умовна щільність імовірності появи відмови.

Виріб використовуватиметься в таких умовах:

- температура повітря 19 °С, відносна вологість повітря 85 %;
- призначений для роботи в закритому приміщенні зі штучно регульованими кліматичними показниками.

Критерієм відмови системи вважається відмова будь-якої її складової частини, що приводить до порушення її працездатності і неможливості використання за призначенням. Відмова будь-якої складової частини системи призведе до відмови системи в цілому. Відновлення роботи системи проводиться методом заміни складової частини, що відмовила.

Розрахунок параметра потоку відмовлень виробу проводиться по формулі:

$$\Lambda = \sum_{i=1}^n N_i \lambda_i \quad (8.1)$$

де n – загальне число групи КВ, що входять до складу проектного виробу; $N_i \lambda_i$ – кількість і інтенсивність відмовлень КВ i -ої групи відповідно.

					МММТ.480.003.003 – ПЗ	Арк.
						64
Змн.	Арк.	№ докум.	Підпис	Дата		

Інтенсивність відмовлень КВ i -ої групи в робочому режимі визначалася по формулі

$$\lambda_i = \lambda_{oi} a_i \dots a_n \quad (8.2)$$

де λ_{oi} – інтенсивність відмовлень КВ в номінальному режимі і нормальних умовах; $a_i \dots a_n$ – поправочні коефіцієнти, що враховують умови роботи (кліматичні і механічні впливи в залежності від призначення й умов майбутньої роботи), електричне навантаження КВ і температуру навколишнього середовища, ступінь інтеграції інтегральних мікросхем, а також співвідношення видів відмовлень.

Конкретні формули, по яких визначалися інтенсивність відмовлень кожної групи комплектуючих виробів, приведені в таблиці 8.2. Величини значень визначилися по довідковим даним з врахуванням вихідних даних для проектного виробу.

По формулі (8.1) визначаємо потік відмовлень виробу:

$$\Lambda = 13,254 \cdot 10^{-6} \text{ 1/год.}$$

Ймовірність безвідмовної роботи виробу протягом часу t визначалася по формулі

$$P(t) = e^{-\Lambda t} \quad (8.3)$$

де Λ – параметр потоку відмовлень; t – час роботи виробу (год)

Дані розрахунку $P(t)$ приведені в таблиці 8.1.

Таблиця 8.1

Ймовірність безвідмовної роботи виробу протягом часу t

250	500	750	1000	2000	5000	8000	10000
0,9967	0,9934	0,9054	0,9868	0,9738	0,9359	0,8994	0,8759

Середній наробіток до відмовлення виробу визначався по формулі:

$$T_0 = \Lambda^{-1} \quad (8.4)$$

Підставляючи дані в формулу (8.4) одержимо

$$T = (13,254 \cdot 10^{-6})^{-1} = 75450 \text{ г.}$$

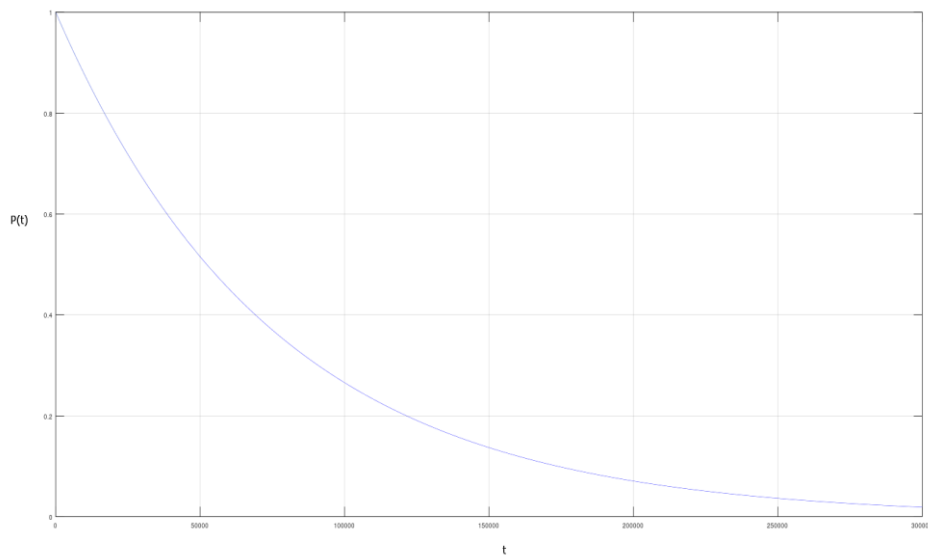


Рис. 8.1 Ймовірність безвідмовної роботи виробу протягом часу t

Таблиця 8.2

Перелік комплектуючих виробів

Поз.	Найменування	Кіл.	$\lambda_{i0} \cdot 10^6$ год ⁻¹	Формула розрахунку λ_i	Коефіцієнти				$\Sigma \lambda_i \cdot 10^6$ год ⁻¹
1	2	3	4	5	6	7	8	9	10
1	Конденсатори				a_1	a_2		a_4	
	КМ5А Н90 0.1мкФ	5	0,02	$\lambda_0 \cdot a_1 \cdot a_2 \cdot a_4$	2,1	0,8		1	0,168
	КМ5А ПЗЗ 22пФ	2	0,02	$\lambda_0 \cdot a_1 \cdot a_2 \cdot a_4$	2,1	0,8		1	0,0672
2	Резистори				a_1	a_2		a_4	
	С1-4 4.7 кОм	1	0,008	$\lambda_0 \cdot a_1 \cdot a_2 \cdot a_4$	2,1	0,8		1	0,01344
	С1-4 10 кОм	1	0,008	$\lambda_0 \cdot a_1 \cdot a_2 \cdot a_4$	2,1	0,8		1	0,01344
3	Мікросхеми				a_1	a_2	a_4	a_6	
	АТmega8	1	2,1	$\lambda_0 \cdot a_1 \cdot a_2 \cdot a_4 \cdot a_6$	2,1	1	1,3	1	5,733
	МАХ232АСРЕ	1	0,8	$\lambda_0 \cdot a_1 \cdot a_2 \cdot a_4 \cdot a_6$	2,1	1	1,3	1	2,184
6	Індикатори				a_1	a_2	a_4	a_6	
	Kingbright L-44 E D	1	0,7	$\lambda_0 \cdot a_1 \cdot a_2 \cdot a_4 \cdot a_5$	2,1	1	1,3	1	1,911
7	Кварцовий резонатор				a_1	a_2	a_4	a_6	
	НС-49U	1	0,2	$\lambda_0 \cdot a_1 \cdot a_4$	2,1			1	0,42
					a_1	a_2	a_4	a_6	
8	Перемикачі	4	0,03	$\lambda_0 \cdot a_1$	2,1				0,252
9	Пайки	171	0,0001	$\lambda_0 \cdot a_1$	2,1				0,036
10	Роз'єм	4	0,01	$\lambda_0 \cdot a_1 \cdot a_2 \cdot a_4$	2,1	1	1,3	1	0,1092
	Разом								13,254

8. РОЗРАХУНОК МЕТРОЛОГІЧНИХ ХАРАКТЕРИСТИК

Проведено десять повторних вимірювань частоти частотоміром, діапазон частот виміру згідно технічних характеристик в від 10 до 85 кГц.

Було отримано наступну вибірку:

$F = (50,05; 50,93; 50,95; 50,06; 50,07; 50,06; 50,06; 50,95; 50,96; 50,96)$ кГц.

За відомими формулами шукаємо:

1) середнє арифметичне $\bar{F}$ за (8.1):

$$\bar{F} = \frac{1}{10} \sum_{i=1}^{10} F_i \cong 50,505 \quad (8.1)$$

2) похибки повторних вимірювань ΔF_i та ΔF_i^2 (8.2):

$$\Delta F = |\bar{F} - F_i| \text{ та } \Delta F_i^2 = (\bar{F} - F_i)^2, \quad (8.2)$$

де $i = 1, 2 \dots 10$;

Результати вимірювань та обчислень заносимо у табл. 8.1.

Таблиця 8.1.

№ з. п.	F, кГц	$F T_i$, кГц	$F T_i^2$, кГц
1	50,05	0,455	0,207025
2	50,93	0,425	0,180625
3	50,95	0,445	0,198025
4	50,06	0,435	0,189225
5	50,07	0,445	0,198025
6	50,06	0,445	0,198025
7	50,06	0,445	0,198025
8	50,95	0,455	0,207025
9	50,96	0,455	0,207025
10	50,96	0,445	0,207025
сер	50,505	0,455	0,198105

Розрахуємо випадкову похибку вимірювання за (8.1). За даними табл. 8.1 знайдемо коефіцієнт Стюдента для 10 вимірювань і довірчої ймовірності 0,95 $t_{0,95}=2,26$; розрахуємо за (8.2) виправлене вибіркове середнє квадратичне відхилення:

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		67

$$\sigma_B = \sqrt{\frac{1}{10 \times 9} \sum_{i=1}^{10} (F - F_i)^2} \approx 0,148 \text{ кГц}$$

та знайдемо випадкову похибку:

$$\Delta_B F = 2,26 * 0,148 = 0,335 \text{ кГц}$$

(Результати вимірювань та обчислень заносимо у табл. 8.1.)

1) похибку приладу за (8.3):

$$\Delta_{\text{пр}} F = \frac{1}{300} t_p k F_{\text{гр}} \quad (8.3)$$

Для надійної ймовірності $P = 0,95$ з табл. 8.1 знайдемо при умові $t_{0,95}=2,0$. Враховуючи, що за умовою $F_{\text{гр}} = 85 \text{ кГц}$, а клас точності $k = 1$, отримаємо:

$$\Delta_{\text{пр}} F = \frac{1}{300} 2,0 * 80 * 1 \approx 0,53 \text{ кГц}$$

2) похибку заокруглення за (8.4):

$$F_3 = P \frac{h}{2}. \quad (8.4)$$

Для цього знайдемо інтервал заокруглення h як половину останнього розряду виміряної величини – 0,05. Отже:

$$\Delta F_3 = 0,95 * \frac{0,05}{2} \approx 0,02 \text{ кГц};$$

3) загальну абсолютну похибку (8.5):

$$\Delta F = \sqrt{(\Delta_B F)^2 + (\Delta_{\text{п}} F)^2 + (\Delta_3 F)^2} = \sqrt{0,335^2 + 0,53^2 + 0,02^2} \approx \quad (8.5)$$

$$\approx 0,63 \text{ кГц}$$

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		68

9. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ ТА ОХОРОНА ПРАЦІ

Вимоги безпеки перед початком роботи

- Вимірювання та випробування слід проводити за програмами та методиками, технічними умовами організацій-виробників або стандартами на продукцію.
- Дозвіл на проведення вимірювань та випробувань діючих електроустановок має бути оформлений відповідно до чинних Правил охорони праці.
- Випробування та вимірювання в діючих електроустановках напругою вище 1000 В виробляються за вбранням.
- Допуск за нарядами або розпорядженнями на проведення вимірювань та випробувань провадиться тільки після видалення з робочих місць інших бригад, що працюють на обладнанні, що підлягає випробуванню або вимірюванню, та здачі ними нарядів або повідомлення про закінчення робіт з розпорядження.
- До складу бригад, що проводять випробування або вимірювання, можуть бути включені працівники з числа ремонтного персоналу з групою з електробезпеки не нижче II для виконання підготовчих робіт, охорони обладнання, що випробовується, а також для роз'єднання та з'єднання шин, жил кабелю, проводів. Ремонтний персонал, включений до складу бригади, до початку випробувань або вимірювань повинен бути проінструктований виробником робіт щодо заходів безпеки при випробуваннях або вимірюваннях.

Вимоги безпеки під час виконання роботи

- Випробування електроустаткування з подачею підвищеної напруги від стороннього джерела
- Роботи електровимірювальними кліщами, вимірювальними штангами та показчиками напруги.
- Роботи з імпульсним вимірювачем ліній

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		69

- Роботи з електролічильниками та вимірювальними приладами
- Роботи з вимірювання напруженості електричного поля
- Випробування електроустаткування з подачею підвищеної напруги від стороннього джерела

Вимоги безпеки після закінчення роботи

- Вимкнути випробувальне (вимірювальне) обладнання;
- у разі повного закінчення випробувань від'єднати дроти від випробувальної установки та зняти огорожі;
- весь інструмент, пристрої, прилади та засоби захисту привести в належний порядок та розмістити у спеціальних шафах та на стелажах;
- доповісти про завершення робіт вищому оперативному (черговому) персоналу та оформити закінчення робіт розписом в оперативному журналі;
- зняти спецодяг, прибрати його та інші засоби індивідуального захисту у шафу для робочого одягу;

					МММТ.480.003.003 – ПЗ	Арк.
						70
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

У результаті виконання кваліфікаційної роботи розроблено комп'ютеризовану систему для вимірювання частоти електричних сигналів.

Основними елементами системи є мікроконтролер STM32F100RB.

Похибка вимірювального каналу 0,5 %.

Надійність розробленої системи: 75450 годин безвідмовної роботи.

Розраховані характеристики задовільняють вимоги технічного завдання.

У якості графічного матеріалу надано структурну та електричну принципу схему, топологія друкованої плати, перелік елементів, що використовуються при побудові. Щодо програмного та алгоритмічного забезпечення надано відповідний код програми для мікроконтролеру та зображено відповідні блок-схеми алгоритму.

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		71

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- 1) Є. Т. Володарський, В. В. Кухарчук, В. О. Поджаренко, Г. Б. Сердюк, «Метрологічне забезпечення вимірювань і контролю: навч. посіб», Вінниця: ВДТУ, 2001.
- 2) Кухарчук В. В. Метрологія та вимірювальна техніка. Навчальний посібник. / Кухарчук В. В., Кучерук В. Ю., Долгополов В. П., Грумінська Л. В. – Вінниця: Універсум-Вінниця, 2004. – 252 с.
- 3) Метрологія у галузі зв'язку. Книга 1. Загальні електрорадіовимірювання: Посібник / Л.В. Коломієць, П.П. Воробієнко, М.Т. Козаченко, М.Б. Налісний, Л.О. Козаченко, О.В. Грабовський. – Одеса: ТОВ «ВМВ», 2009. – 480 с.
- 4) Метрологія у галузі зв'язку, Книга І: Метрологія, стандартизація, менеджмент якості та оцінка відповідності: [підручник], / [Л. В. Коломієць, П.П. Воробієнко, М. Т. Козаченко]. - Одеса: Вид-во «Стандарт», 2006. – 430с.
- 5) Шаповаленко О.Г. Основи електричних вимірювань / О.Г. Шаповаленко, В.М. Бондар. - К.: Либідь, 2002. - 412с.
- 6) Дорожнець М. А. Основи метрології та вимірювальної техніки / М. А. Дорожнець, В. Н. Мотало, Б. С. Стадник. – Львів: Львівська Політехніка. – 2005. – 163 с.
- 7) Петрушак В. С., Поліщук А. О. Аналіз методів та засобів вимірювання частоти радіосигналу International Scientific-technical journal «MEASURING AND COMPUTING DEVICES IN TECHNOLOGICAL PROCESSES» 2019, Issue 1, p. 22-25.
- 8) Поліщук Є. С. Метрологія та вимірювальна техніка. Підручник. / Поліщук Є. С., Дорожновець М. М., Яцук В. О. та ін. 2-е видання, доповнене та перероблене.; За ред. проф. Є. С. Поліщука.– Львів. Видавництво Львівської політехніки, 2012. – 544 с.

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		72

9) Кухарчук В. В. Основи метрології та електричних вимірювань. Підручник. / Кухарчук В. В., Кучерук В. Ю., Володарський Є. Т., Грабко В. В. – Херсон: ОЛДІ-ПЛЮС, 2013. – 538 с.

10) Дорожовець М. М. Опрацювання результатів вимірювань. Навчальний посібник. – Львів: Видавництво Національного університету «Львівська політехніка», 2007. – 624 с.

11) Гуржій А.М., Поворознюк Н.І. Електричні і радіотехнічні вимірювання / А.М. Гуржій, Н.І. Поворознюк. – Київ: Навчальна книга, 2002.

12) Муратов В.Г. Метрологія, технологічні вимірювання та прилади навчальний посібник. - вид. 2-е, доп. / В.Г Муратов. - К.: Освіта України, 2016 – (Рек. МОН України).

13) Програмування вбудованих систем : метод. вказівки до виконання лабораторних робіт для студентів денної та заочної форми навчання за спеціальністю 123 “Комп’ютерна інженерія ” / уклад. Дреєва Г.М., Дреєв О.М., Денисенко О.О., Коноплицька-Слободенюк О.К. — Кропивницький: ЦНТУ, 2018. — 90 с.

14) Квашнін, В. О. Програмування та застосування мікроконтролерів STM32F4Discovery: монографія / В. О. Квашнін, А. В. Бабаш, В. В. Квашнін. – Краматорськ: ЦТРІ «Друкарський дім», 2017. – 143 с.

15) Архітектура новітніх мікроконтролерів: Програмування мікроконтролерів сімейства ARM: Навчальний посібник / А.О. Новацький. – Київ: КПІ ім. Ігоря Сікорського, 2017. –138с.

16) Microcontroller, Microprocessor and Microcomputer Interfacing for Real-Time Systems. Witold Kinsner- Winnipeg, MB: OCO Research, 2020.- 973p.

					МММТ.480.003.003 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		73

ДОДАТКИ

Лістинг програми

```

/*
*****
Частотомір з межами вимірів 0.02Гц..10МГц.
Вхід: PA8 (TIM1_CH1)

*****
#include "stm32f10x.h"
int display_init();
int display(uint32_t fclk, uint32_t m, uint32_t n);
#define CNT_OVR 0x10000
{
public:
    uint32_t n;
};
class DBuf
{
private:
    Data *p0;
    Data *p;
    uint32_t m_size;
    uint32_t cnt;
    bool full;
public:
    uint32_t c0;
    DBuf():p0(0), p(0), m_size(0), cnt(0), full(false),
        is_ready(false), c0(0) {}
    DBuf(uint32_t _size):p0(0), p(0), m_size(0), cnt(0),
        full(false), is_ready(false), c0(0)
    {
        if(_size != 0)
        {
            p = p0 = new Data[_size];
            m_size = _size;
        }
    }

    void clear() volatile
    {
        delete[] p0;
        p0 = 0;
        p = 0;
        m_size = 0;
        cnt = 0;
        full = false;
        is_ready = false;
        c0 = 0;
    }

    void reset(uint32_t _size) volatile
    {
        clear();
        if(_size != 0)
        {
            p = p0 = new Data[_size];
            m_size = _size;
        }
    }

    inline bool is_full() const volatile {return full;}

    inline uint32_t get_cnt() const volatile {return cnt;}
    uint32_t size() const volatile

```

```

    {
        return (full) ? m_size : p-p0;
    }

    inline uint32_t max_size() const volatile {return m_size;}

    void push(const Data &d) volatile
    {
        if(m_size != 0)
        {
            *p = d;
            ++p;
            if(p >= p0 + m_size)
            {
                full = true;
                p = p0;
            }
            cnt++;
        }
    }

    Data *first() volatile
    {
        if(!full)
            return (p == p0) ? 0 : p0;

        return p;
    }

    Data *last() volatile
    {
        if(!full)
            return (p == p0) ? 0 : p-1;
        Data *px = p - 1;
        if(px < p0)
            px += m_size;
        return px;
    }
};

// *****
// Глобальні змінні, що використовуються обробниками переривань
// *****
volatile uint16_t m_cnt_prev = 0;

volatile uint16_t m_capt = 0;

volatile uint16_t n_cnt_prev = 0;

volatile uint16_t n_capt = 0;

volatile uint32_t m_corr = 0;

volatile uint32_t m_capt_corr = 0;

volatile uint32_t n_corr = 0;

volatile uint32_t n_capt_corr = 0;

volatile DBuf dbuf;

void _update_fm_vars()
{
    uint32_t m_ccr1 = TIM1->CCR1;
    uint32_t n_ccr1 = TIM2->CCR1;

```

```

uint32_t m_cnt = (uint16_t)(TIM1->CNT - 1);
uint32_t n_cnt = TIM2->CNT;

bool m_ovf = m_cnt < m_cnt_prev;
bool n_ovf = n_cnt < n_cnt_prev;

m_cnt_prev = m_cnt;
n_cnt_prev = n_cnt;

bool captf = (m_ccr1 != m_capt);

if(captf)
{
    m_capt = m_ccr1;
    n_capt = n_ccr1;
    m_capt_corr = m_corr;
    n_capt_corr = n_corr;
    if(m_ovf && m_ccr1 <= m_cnt)
        m_capt_corr += CNT_OVR;
    if(n_ovf && n_ccr1 <= n_cnt)
        n_capt_corr += CNT_OVR;
}

if(m_ovf)
    m_corr += CNT_OVR;
if(n_ovf)
    n_corr += CNT_OVR;
}

extern "C" void TIM6_DAC_IRQHandler()
{
    if(TIM6->SR & TIM_SR_UIF)
    {
        TIM6->SR = ~TIM_SR_UIF;

        _update_fm_vars();
    }
}

extern "C" void TIM7_IRQHandler()
{
    TIM7->SR = ~TIM_SR_UIF;

    TIM1->CCER &= ~TIM_CCER_CC1E;

    _update_fm_vars();

    TIM1->CCER |= TIM_CCER_CC1E;

    uint32_t m = m_capt + m_capt_corr;
    uint32_t n = n_capt + n_capt_corr;

    uint32_t n0 = 0;
    if(dbuf.size() > 0)
        n0 = dbuf.first()->n;

    dbuf.push({m, n});

    const Data *p1 = dbuf.first();
    if(p1->m != 0 && dbuf.is_full())
        dbuf.is_ready=true;

    if(n0 > p1->n)
        dbuf.c0++;
}

void fm_init()

```

```

{

    const uint32_t level=(1 << __NVIC_PRIO_BITS) - 1;

    NVIC_SetPriority(TIM6_DAC_IRQn, level);
    NVIC_SetPriority(TIM7_IRQn, level);

    NVIC_EnableIRQ(TIM6_DAC_IRQn);
    NVIC_EnableIRQ(TIM7_IRQn);

    RCC->APB1ENR |=
        RCC_APB1ENR_TIM2EN |
        RCC_APB1ENR_TIM6EN |
        RCC_APB1ENR_TIM7EN;
    RCC->APB2ENR |=
        RCC_APB2ENR_IOPAEN |
        RCC_APB2ENR_TIM1EN;

    GPIOA->CRH =
        GPIOD->CRH & ~0xF | 0x8;

    TIM1->CR2 |= 3 << 4;    // MMS=011 ==> TRGO = Compare pulse
    TIM1->SMCR |=
        (5 << 4) |          // TS=101 (TI1FP1)
        7;                  // SMS=111 (External mode clock 1)
    TIM1->CCMR1 |= 1;        // CC1S=01, IC1=TI1

    // CC1S=11: IC1=TRC=TRGIO (SMCR_TS=00 ==> TRC=TRGIO)
    TIM2->CCMR1 |= 3;

    TIM6->ARR = 0x7FFF;

}

void fm_stop()
{
    TIM7->CR1 &= ~TIM_CR1_CEN;
    TIM6->CR1 &= ~TIM_CR1_CEN;
    TIM1->CCER &= ~TIM_CCER_CC1E;
    TIM1->CR1 &= ~TIM_CR1_CEN;
    TIM2->CCER &= ~TIM_CCER_CC1E;
    TIM2->CR1 &= ~TIM_CR1_CEN;

    TIM6->DIER &= ~TIM_DIER_UIE;
    TIM7->DIER &= ~TIM_DIER_UIE;
}

// k*timer_div1*timer_div2/SystemCoreClock
bool fm_start(uint32_t timer_div1, uint32_t timer_div2, uint32_t k)
{
    if(timer_div1 - 1 > 0xFFFF || timer_div2 - 2 > 0xFFFE || k == 0)
        return false;

    if(TIM1->CCER & TIM_CCER_CC1E)
        fm_stop();

    m_cnt_prev = 0;
    m_capt = 0;
    m_corr = 0;
    m_capt_corr = 0;

    n_cnt_prev = 0;
    n_capt = 0;
    n_corr = 0;
    n_capt_corr = 0;
    dbuf.reset(k + 1);
}

```

```

TIM2->CNT = 0;
TIM2->CR1 |= TIM_CR1_CEN;
TIM2->CCER |= TIM_CCER_CC1E;

TIM1->CNT = 1;
TIM1->CR1 |= TIM_CR1_CEN;
TIM1->CCER |= TIM_CCER_CC1E;

TIM6->CNT = 0;
TIM6->CR1 |= TIM_CR1_CEN;

TIM7->PSC = timer_div1 - 1;
TIM7->EGR = TIM_EGR_UG;
TIM7->ARR = timer_div2 - 1;
TIM7->CR1 |= TIM_CR1_CEN;

TIM6->DIER |= TIM_DIER_UIE;
TIM7->SR = ~TIM_SR_UIF;
TIM7->DIER |= TIM_DIER_UIE;

return true;
}

bool get_result(uint32_t &m, uint32_t &n, uint32_t &cnt, uint32_t &c0)
{
    m = 0;
    n = 0;
    cnt = dbuf.get_cnt();
    c0 = dbuf.c0;
    bool r = dbuf.is_ready;
    if(r)
    {
        __disable_irq();
        cnt = dbuf.get_cnt();
        c0 = dbuf.c0;
        volatile const Data *p1 = dbuf.first();
        volatile const Data *p2 = dbuf.last();
        m = p2->m - p1->m;
        n = p2->n - p1->n;
        __enable_irq();
    }
    return r;
}

int main()
{
    display_init();
    fm_init();

    fm_start(240, 10000, 5);

    uint32_t cnt0=0;

    while(true)
    {
        uint32_t m, n, cnt, c0;
        if(get_result(m, n, cnt, c0)&&cnt0!=cnt)
        {
            cnt0=cnt;

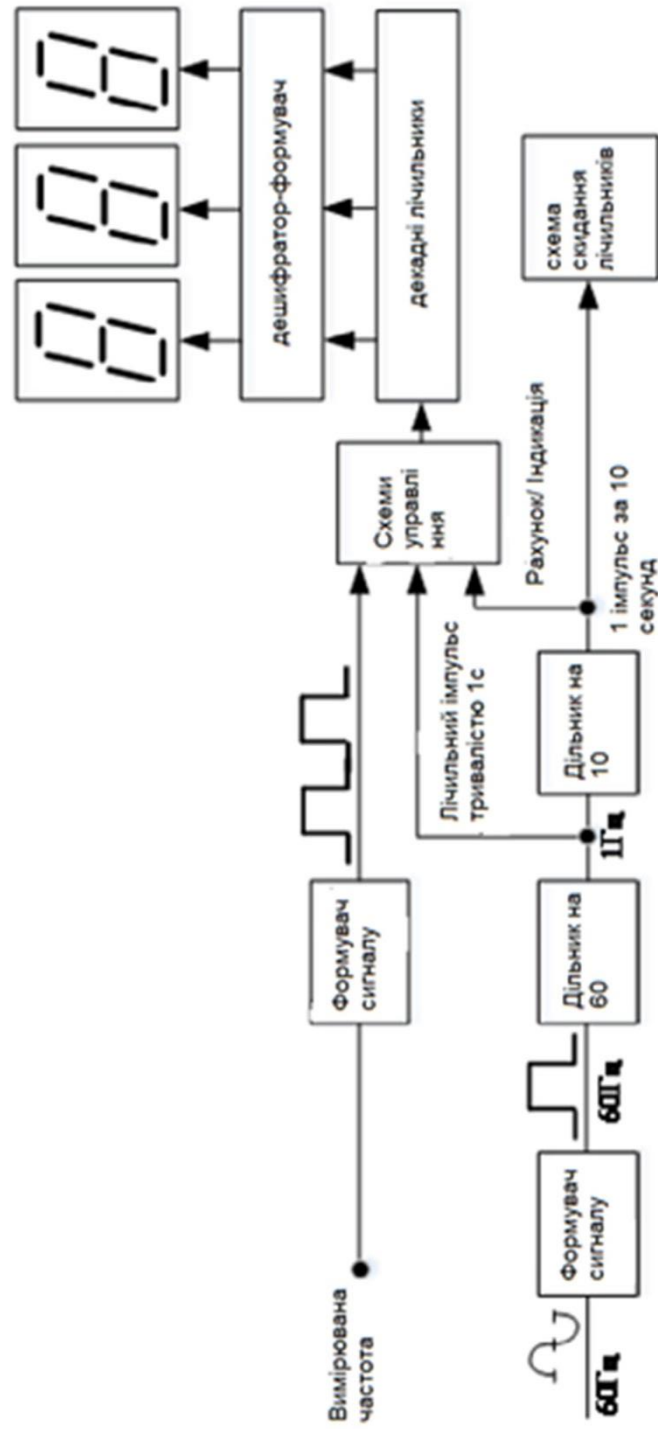
            // float f=(float)SystemCoreClock*m/n;

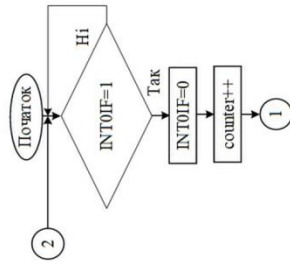
            display(SystemCoreClock, m, n);
        }
    }
    // return 0;
}

```

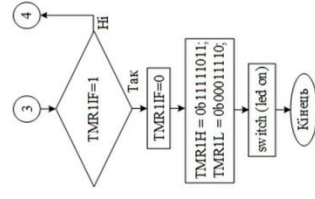
	Поз.познач.	Найменування	Кільк	Приміт.
		Конденсатори		
	C1-C8	KM5A H90 0.1 мкФ (-20+50)%	8	
	C9, C10	KM5A ПЗЗ 22 пФ 5%	2	
		Мікросхеми		
	DD1	STM32F100RBTx	1	
		Резистори		
	R1... R3	C1-4 0.125 Вт 5% 470 Ом	3	
	R5	PTV09 0.125 Вт 5% 20K	1	
	R6	C1-4 0.125 Вт 5% 330 Ом	1	
		Перемикачі		
	SA1, SA2	ESP1010 12В 0.5А (SPDT)	2	
		Роз'єми		
	XS1	D-SUB DBR 9 МА	1	
	XS2	AMPMODU 103635-4	1	
	XS3	IDC-14 MS	1	
	XS4	DTD-0020 (DJK-05D)	1	

					МММТ.480.003.003 – ПЕ								
Вим	Лист	№ докум.	Підпис	Дата									
Розроб.		Гісса А.Д.			Комп'ютеризована система для вимірювання частоти електричних сигналів Перелік елементів				Лім.	Лист	Листів		
Перевір.		Чепюк Л.О.									1	1	
Реценз.									Державний університет «Житомирська політехніка», гр. МТ-2				
Н. Контр.													
Затверд.		Подчашинський Ю											

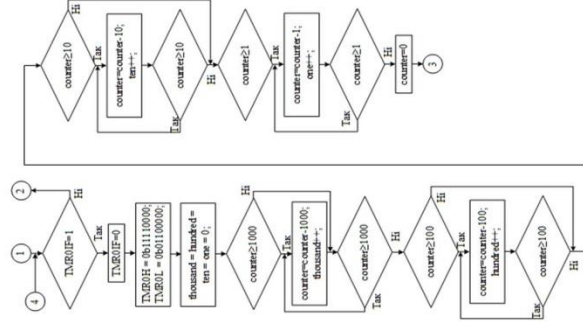
[illegible]



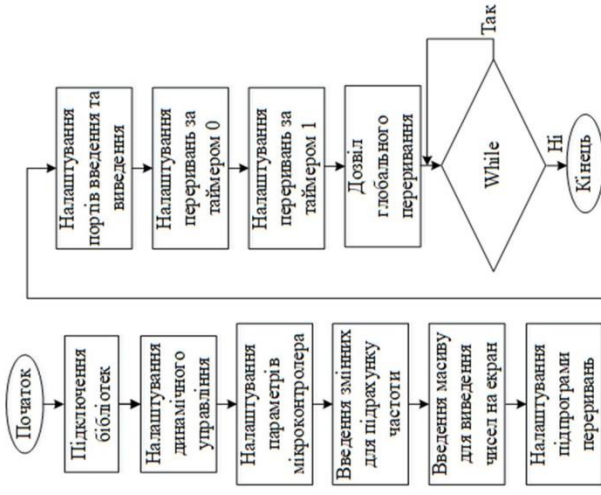
Алгоритм переривання за таймером 0



Алгоритм переривания за таймером I



Алгоритм підрахунку вхідного сигналу по ніжці RB0



Алгоритм основной функции

[illegible]

[illegible]