

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»**

*Спеціальність: 123 «Комп'ютерна інженерія»*

*Освітньо-професійна програма: «Обслуговування  
комп'ютерних систем і мереж»*

*Група: 4КС-57*

# **Дипломний проект**

**здобувача освіти денної форми навчання  
КС.57.14.000.ДП**

***НІЗЯЄВА  
ІВАНА РОМАНОВИЧА***

**м. Одеса  
2024 р.**

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

Спеціальність: 123 «Комп'ютерна інженерія»

Освітньо-професійна програма: «Обслуговування комп'ютерних систем і мереж»

Група: 4КС-57

**ПОЯСНЮВАЛЬНА ЗАПИСКА**

до дипломного проекту на тему:

**Розробка додатку для розрахунку стабілізаторів напруги живлення**

Проектний матеріал складається з пояснювальної записки на 75 сторінках та графічного (презентаційного) матеріалу на 14 аркушах (слайдах)

Дипломник  (Нізяєв І.Р.)

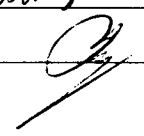
Керівник  (Кривченко Ю.В.)

**Консультанти:**

з економічного розділу  (Іванченков В.С.)

з розділу охорони праці та техніки безпеки  (Чорновол Н.І.)

з нормоконтролю  (Петрашова В.І.)

старший консультант  (Кривченко Ю.В.)

**До захисту допущений**

Голова циклової комісії  (Кривченко Ю.В.)

Завідувач відділення  (Скорнякова О.В.)

Захист «19» 06 2024 р. Протокол ЕК № 3

Оцінка ЕК 4/90б/00 885

Секретар ЕК 

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

Відділення комп'ютерних систем Комісія КТ та ПІ  
Спеціальність 123 «Комп'ютерна інженерія»  
Освітньо-професійна програма «Обслуговування комп'ютерних систем і мереж»

ЗАТВЕРДЖУЮ:  
Заст. дир. з НВР Беркань І.В.  
" 15 " 01 2024 р.

### ЗАВДАННЯ

#### на дипломний проект

Здобувачеві освіти Нізяєву Івану Романовичу  
(прізвище, ім'я, по батькові)

1. Тема проекту Розробка додатку для розрахунку стабілізаторів напруги живлення

затверджена наказом по коледжу від "02" листопада 2023 р. № 244-А2-ОЗ

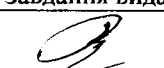
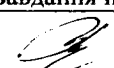
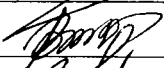
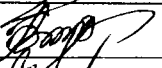
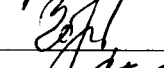
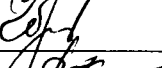
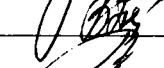

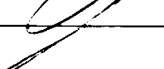
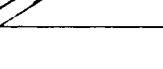
2. Термін здачі закінченого проекту 10.06.24р

3. Вихідні данні до проекту Схеми ключових стабілізаторів напруги; Перелік ключових параметрів стабілізаторів напруги різних типів; Програма має розраховувати числові значення параметрів елементів ключових стабілізаторів напруги; Передбачити відповідні попередження про некоректні початкові дані та точність обчислення; Програма повинна мати візуальний інтерфейс з можливістю необхідних налаштувань за допомогою списків вибору та перемикачів

4. Зміст розрахунково-пояснювальної записки (перелік питань, які необхідно розробити) Аналіз методів розрахунку параметрів стабілізаторів напруги; Аналітичний огляд програмного забезпечення для математичного моделювання електронних схем; Короткий аналіз і вибір програмних засобів для розробки; Реалізація алгоритмів розрахунку стабілізаторів напруги; Реалізація візуального інтерфейсу та тестування розробленого застосунку

5. Перелік графічного (презентаційного) матеріалу (з точним зазначенням обов'язкових креслень, кількості слайдів) Структурна джерела вторинного електроживлення з безперервною стабілізацією напруги; Структурні схеми безтрансформаторних джерел з нерегульованим і з регульованим перетворювачем; Структурні схеми джерел на основі перетворювача і стабілізатора напруги, ключового стабілізатора; Схеми ключових стабілізаторів напруги; БСА розрахунку стабілізаторів напруги; Візуальний інтерфейс застосунку для розрахунку схем стабілізатора напруги; Порівняння результатів ручного та програмного розрахунків



6. Консультанти по проекту, із зазначенням розділів проекту, що їх стосується

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
Основний розділ	Кривченко Ю.В.		
Економічний розділ	Іванченков В.С.		
Розділ охорони праці	Чорновол Н.І.		
Нормоконтроль	Петрашова В.І.		
Старший консультант	Кривченко Ю.В.		

7. Дата видачі завдання 15.01.24

Керівник Кривченко Ю.В.


Завдання прийняв до виконання Нізяєв І.Р.

  
(підпис)  
  
(підпис)

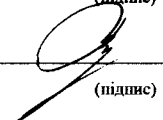
КАЛЕНДАРНИЙ ПЛАН

№ з/р	Назва етапів дипломного проекту	Термін виконання етапів дипломного проекту (роботи)	Відмітка про виконання
1	Вступ. Постановка мети та задач проектування	29.04.24	Виконав
2	Дослідження роботи блоків живлення та їх параметрів	1.05.24	Виконав
3	Загальний опис реалізації моделі	4.05.24	Виконав
4	Аналіз схем з безперервною стабілізацією напруги	7.05.24	Виконав
5	Розрахунок стабілізаторів напруги	11.05.24	Виконав
6	Вибір і опис інструментарію для реалізації моделі	14.05.24	Виконав
7	Огляд програмних засобів розрахунку електронних схем	16.05.24	Виконав
8	Розробка візуального інтерфейсу для застосунку	18.05.24	Виконав
9	Розробка алгоритмів розрахунку стабілізаторів	22.05.24	Виконав
10	Реалізація алгоритмів розрахунку мовою C++	24.05.24	Виконав
11	Випробування програмного застосунку	28.05.24	Виконав
12	Аналіз результатів, підготовка слайдів презентації	1.06.24	Виконав
13	Економічні розрахунки та питання з охорони праці	4.06.24	Виконав
14	Підготовка графічної частини проекту	7.06.24	Виконав
15	Підготовка проекту до захисту та тестування ПЗ	9.06.24	Виконав

Дипломник

  
(підпис)

Керівник

  
(підпис)



# ЗМІСТ

Вступ.....	7
1 Основний розділ.....	8
1.1 Огляд та аналіз принципів роботи джерел живлення.....	8
1.2 Аналіз методів розрахунку параметрів стабілізаторів напруги.....	13
1.2.1 Аналітичний огляд схем стабілізаторів напруги.....	13
1.2.2 Розрахунок ключових стабілізаторів напруги.....	17
1.3 Аналітичний огляд програмного забезпечення для математичного моделювання електронних схем.....	20
1.4 Вибір операційної системи для створюваного програмного забезпечення.....	22
1.5 Короткий аналіз і вибір програмних засобів для розробки.....	22
1.6 Аналіз можливостей інтегрованого середовища розробки Embarcadero RAD Studio C++Builder.....	31
1.6.1 Застосування компонентів введення і відображення текстової інформації.....	33
1.6.2 Застосування панелей та компонентів зовнішнього оформлення.....	37
1.7 Реалізація алгоритму розрахунку стабілізаторів напруги.....	39
1.8 Реалізація візуального інтерфейсу та тестування розробленого застосунку.....	45
1.8.1 Тестування у застосунку у режимі розрахунку стабілізатора знижувального типу.....	46
1.8.2 Тестування у застосунку у режимі розрахунку стабілізатора підвищувального типу.....	48
1.8.3 Тестування у застосунку у режимі розрахунку стабілізатора полярно-інвертуючого типу.....	50
2 Економічний розділ.....	53
2.1 Резюме .....	53
2.2 Визначення трудомісткості розробки програмного забезпечення.....	53

2.3 Розрахунок ціни програмного продукту .....	56
3 Розділ охорони праці та техніки безпеки.....	58
3.1 Вступ.....	58
3.2 Аналіз небезпечних та шкідливих чинників, що впливають на працівника.....	58
3.3 Розробка заходів з охорони праці.....	59
3.3.1 Виробничі приміщення.....	59
3.3.2 Мікrokлімат робочої зони працівників, вентиляція.....	59
3.3.3 Освітлення робочого місця, шум, вібрація.....	60
3.3.4 Організація робочого місця користувача ПК.....	60
3.3.5 Електробезпека.....	61
3.4 Пожежна безпека.....	62
Висновки.....	63
Перелік використаних інформаційних джерел.....	64
ДОДАТОК А. Код функції btnStabilizerClick мовою С++.....	65
ДОДАТОК Б. Слайди мультимедійної презентації .....	69

## ВСТУП

Стабілізуючі пристрої електроживлення в складі блоків живлення (джерел вторичного електроживлення) є невід'ємною частиною сучасних електронних пристроїв, зокрема комп'ютерної техніки, та у значній мірі визначають їх техніко-економічні показники.

Зважаючи на те, що проблема швидкої розроблення засобів сучасної електроніки посилюється, як правило, малим часовим інтервалом, що відводиться інженерові-розробникові на розроблення схеми, необхідно застосовувати сучасні комп'ютеризовані методики стабілізуючого пристрою, аналізу та оптимізації. Метою застосування цих методів є підвищення продуктивності розроблення та якості розроблюваної апаратури.

В сучасних умовах необхідність автоматизації стабілізуючого пристрою блоків живлення, тобто реалізації алгоритмічного і програмного забезпечення, обумовлена досить складним та тривалим процесом ручного стабілізуючого пристрою. При створенні відповідного програмного забезпечення з'явиться можливість істотно скоротити даний процес. Окрім цього, виключаються можливі помилки, котрі можуть виникнути при ручному стабілізуючого пристрою. Аналітичний розрахунок таких ланцюгів представляє значну трудність навіть при використанні програмних засобів. Проте, існують методи наближеного стабілізуючого пристрою схем електронних стабілізуючих пристроїв, що дозволяють із достатньою задля інженерної практики точністю визначити основні параметри проектованої схеми.

Мета дипломного проекту полягає в реалізації методики стабілізуючого пристрою схем електронних стабілізуючих пристроїв, що входять до складу блоків живлення сучасних електронних пристроїв, зокрема комп'ютерної техніки. На основі цієї методики треба розробити алгоритми стабілізуючого пристрою та виконати їх програмну реалізацію мовою C++. Розроблюваний програмний застосунок дозволить розраховувати параметри деталей, що входять до складу блоків живлення поза початковими даними і виводити їх на друк чи на екран.

					<b>КС 57. 14 000. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		7

# 1 ОСНОВНИЙ РОЗДІЛ

## 1.1 Огляд і аналіз принципів роботи джерел живлення

Джерела вторичного електро-живлення є невід'ємною частиною будь-якого сучасного електронного пристрою та у значній мірі визначають техніко-економічні показники апаратури.

Джерела вторичного електро-живлення, що входять до складу радіоелектронної апаратури (РЕА), важко піддаються мініатюризації зважаючи на наявність таких великогабаритних деталей, як трансформатори живлення, згладжувальні дроселі, конденсатори фільтрів, радіатори охолодження. Через це у даний період поступове вдосконалення РЕА, направлене на поліпшення її технічних характеристик та розширення круга виконуваних завдань, накладає усе більш жорсткі вимоги на параметри джерела вторичного електро-живлення у частині забезпечення споживачів електроенергією високої якості: із низьким рівнем пульсацій та шумів, більш високостабільних, із малим внутрішнім опором та певними параметрами перехідних процесів. Ці підвищені вимоги значною мірою ускладнюють процес розроблення джерел вторичного електро-живлення, роблячи його тривалим та дорогим.

Традиційні методи розроблення джерел вторичного електро-живлення, що ґрунтуються на наближених інженерних розрахунках та відробітку властивостей на макеті, вступили у протиріччя із вимогами підвищення техніко-економічних властивостей, скорочення термінів розроблення та зменшення вартості проектних робіт.

Сучасна РЕА у своєму складі містить розгалужені системи вторичного електро-живлення (СВЕЖ), котрі діляться на підсистеми по функціональному призначенню споживачів. Підсистеми електро-живлення поза способом здійснення перетворювально-стабілізуючих функцій можуть виконуватися централізованого, децентралізованого та комбінованого типів та реалізуються, як правило, із окремих функціональних вузлів чи їх поєднань. До функціональних вузлів, широко вживаних у СВЕЖ, відносяться: випрямні пристрої (ВП),

					КС 57. 14 000. 00 ДП ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		8

згладжуючі фільтри (ЗФ), лінійні компенсаційні стабілізуючі пристрої напруження та струма, ключові стабілізуючі пристрої напруження, статичні перетворювачі та ін.

Велике місце у РЕА займають джерела вторичного електро-живлення, котрі виконуються по схемі рис.1.1 на вихідну напругу з 4 до 250В та струми у навантаженні до 3А.

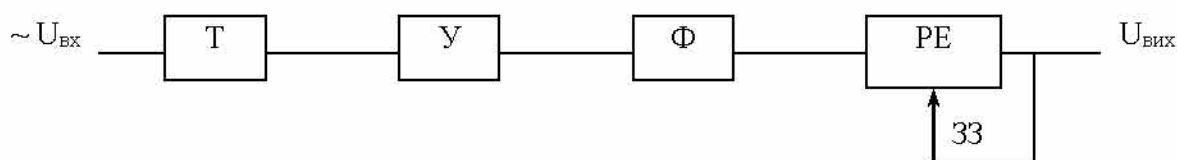


Рисунок 1.1. Структурна схема джерела вторичного електро-живлення із безперервною стабілізацією напруження: Т – трансформатор, У – випрямляч, Ф – фільтр, РЕ – регулювальний елемент, ЗЗ – зворотний зв'язок

Такі джерела вторичного електро-живлення використовуються задля широкого класу аналогових пристроїв там, де потрібні добра стабілізація та низькі рівні пульсації вихідної напруження. Враховуючи, що цей клас джерел живлення складає чималу групу розроблюваних джерел вторичного електро-живлення та визначає узагальнений рівень техніко-економічних показників СВЕЖ, даний напрямок схемотехніки вимагає подальшого розвитку у частині підвищення значення мініатюризації та уніфікації, поліпшення характеристик по динамічній стабільності та підвищення надійності.

У даний період із'явилася необхідність реалізації джерел вторичного електро-живлення із зниженим рівнем вихідної напруження до 5В та струмами споживання з 50 до 600А та вище. Прагнення зменшити масу та габарити джерела вторичного електро-живлення апаратури привело до необхідності застосовувати на практиці пристрої із безтрансформаторним входом та імпульсним регулюванням напруження у первинній мережі змінного струма.

В джерелі вторичного електро-живлення із безтрансформаторним входом (рис. 1.2) змінна напруження системи електропостачання перетворюється безтрансформаторним ВП у порівняно високу напругу постійного струма.

На виході 3Ф є ключовий стабілізатор, який знижує випрямлену напругу та здійснює стабілізацію вихідної напруги джерела вторичного електроживлення. До виходу стабілізуючого пристрою підключений статичний перетворювач (СП), що працює на підвищеній частоті перетворення. Дуже часто процедури ключового стабілізуючого пристрою та перетворювача поєднуються у СП із внутрішньою стабілізацією. При створенні таких пристроїв важливим є забезпечення їх надійної роботи через можливі великі "крізни" та пускові струми та наявність генерованих радіоперешкод високого значення при перемиканнях транзисторів та діодів у силових ланцюгах джерела.

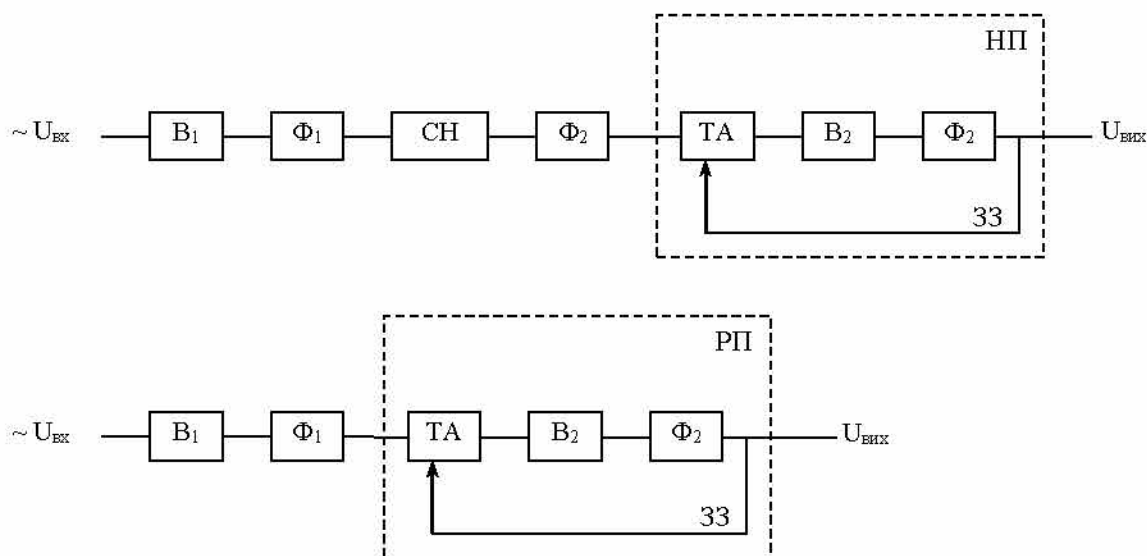


Рисунок 1.2. Структурні схеми безтрансформаторних джерел вторичного електро-живлення із нерегульованим перетворювачем та стабілізатором напруги (а) та із регульованим перетворювачем (б):  $B_1, B_2$  – випрямлячі;  $\Phi_1, \Phi_2$  – фільтри; І – інвертор; СН – стабілізатор напруги; НП – нерегульований перетворювач; РП – регульований перетворювач; 33 – зворотний зв'язок

Джерела вторичного електро-живлення із безтрансформаторним входом можуть виконуватися із питомими показниками з 70 до 300 Вт/дм<sup>3</sup>. При живленні РЕА з систем електропостачання постійного струму та з автономних джерел електричної енергії у СВЕЖ застосовуються функціональні схеми, змальовані на рис.1.3. Основним функціональним вузлом джерел вторичного електро-живлення, що використовує електроенергію автономного джерела чи

системи електропостачання постійного струма, є СП, що перетворює напругу постійного струма джерела у змінну напругу прямокутної чи ступінчастої форми. В багатоканальних джерелах вторичного електро-живлення використовуються способи централізованої стабілізації кожного каналу живлення (рис.1.3, а). В першому випадку сумарна стабільність забезпечується до  $\pm 5\%$ , а у другому – до 1,5...2%.

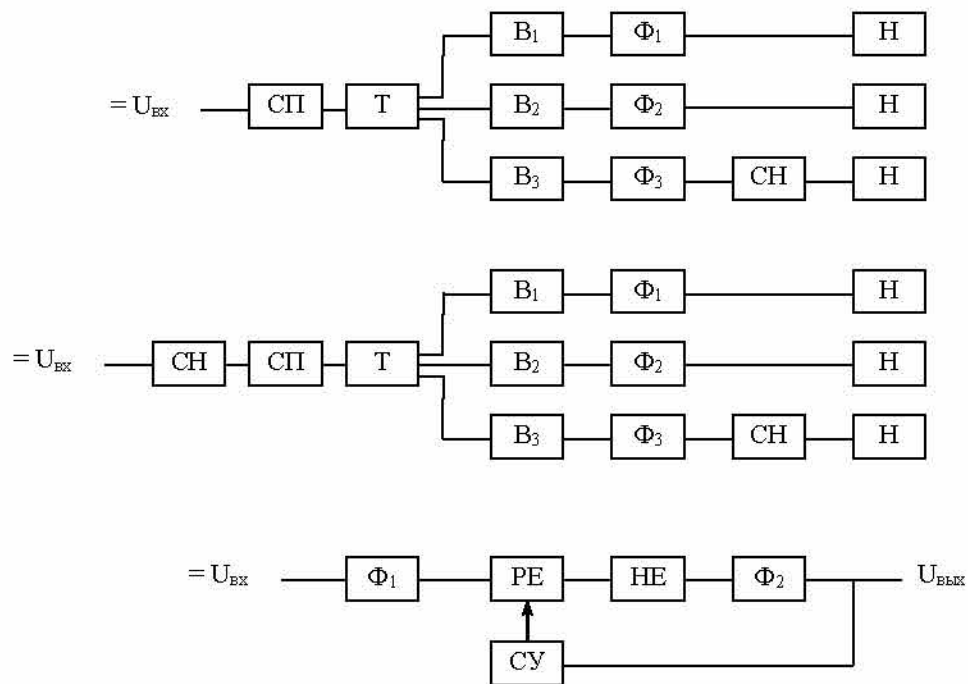


Рисунок 1.3. Структурні схеми джерел вторичного електро-живлення із живленням з системи електропостачання постійного струма на основі статичного перетворювача із внутрішньою стабілізацією (а), статичного перетворювача та стабілізуючого пристрою напруги (б), ключового стабілізуючого пристрою (у): СП – статичний перетворювач; СН – стабілізатор напруги; Т – трансформатор високочастотний; В1, В2, В3 – випрямлячі; Ф1, Ф2, Ф3 – фільтри; Н – навантаження; НЕ – накопичувальний елемент; СУ – схема керування; РЕ – регулювальний елемент

Часто використовується функціональна схема (рис.1.3, б), у якій у ланцюг живлення СП включається стабілізатор напруги лінійного чи ключового типу. Лінійний стабілізатор включається задля отримання на виході покращеної якості перехідного процесу при різких змінах струма у навантаженні та

напруження живлення. Ключовий стабілізатор застосовується в випадках коли у автономному джерелі чи системі електропостачання постійного струма межі зміни напруження живлення істотно перевершують сталий нормальний режим їх роботи та до якості вихідної електроенергії не пред'являються високі вимоги.

Інколи у РЕА не потрібно забезпечувати гальванічну розв'язку джерел вторичного електро-живлення з системи електропостачання постійного струма. В таких випадках у СВЕЖ застосовуються ключові стабілізуючі пристрої напруження (КСН), котрі виконуються по схемі рис. 1.3, у та забезпечують процедури трансформації та стабілізації напруження, а разом з цим при необхідності інвертування вихідної напруження. Схеми керування КСН, що представляють собою широтно-імпульсні модулятори, виконуються на основі спеціалізованих мікросхем. Інколи із метою побудови уніфікованих схем стабілізуючих пристроїв знижувального, підвищувального та полярно-інвертувального типів вони можуть існувати універсальними.

Використання високочастотних СП, що працюють з вхідного джерела постійного струма по схемах рис. 1.3, а, б, дає створювати джерела вторичного електро-живлення із питомими показниками до  $50 \text{ Вт/дм}^3$ , а у разі вживання КСН по схемі рис. 1.3, у питомі характеристики джерел вторичного електро-живлення досягають  $90 \text{ Вт/дм}^3$ . Подальше підвищення питомих показників цього класу джерел вторичного електро-живлення спроможне існувати досягнуте при використанні методів комплексної мініатюризації.

Тут були показані тільки основні шляхи реалізації джерел вторичного електро-живлення, обумовлені вимогами із боку РЕА, яка потребує економічно вигідних технічних рішень перетворення та регулювання енергії. Враховуючи, що проблема розроблення джерел вторичного електро-живлення посилюється, як правило, малим часовим інтервалом, що відводиться інженерів-розробників на розроблення схеми, подальший розвиток напрямків у області реалізації надійних та економічних джерел вторичного електро-живлення не спроможне існувати успішним без використання спеціалізованого програмного забезпечення.

					<b>КС 57. 14 000. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		12

## 1.2 Аналіз методів стабілізуючого пристрою властивостей стабілізуючих пристроїв напруження

### 1.2.1 Аналітичний огляд схем стабілізуючих пристроїв напруження

Стабілізуючі пристрої напруження поза принципом дії можливо розділити на два принципово різноманітних види:

- стабілізуючі пристрої безперервної дії, регулюючим елементом котрих є плавно регульований опір;
- ключові, чи імпульсні, стабілізуючі пристрої, регулюючим елементом котрих є ключ, що періодично замикається та розмикається; стабілізація у цьому випадку досягається зміною шпаруватості роботи ключа.

До переваг стабілізуючих пристроїв безперервної дії відносяться: простота схеми, високий ступінь згладжування пульсацій, відсутність перешкод, зазвичай супроводжуючих роботу ключевих стабілізуючих пристроїв.

Недоліком стабілізуючих пристроїв безперервної дії є неможливість отримання ККД вище 40-60% через втрати у регулюючому елементі. Крім того, задля розсіяння цієї потужності регульовальний елемент доводиться забезпечувати радіатором із великою поверхнею охолодження, що приводить до значних габаритів джерел живлення. Вказані недоліки обмежують можливість використання таких стабілізуючих пристроїв, наприклад, у апаратурі, призначеній задля установки на літальних апаратах всіх видів, зокрема, на штучних супутниках Землі, де енергоресурси вельми обмежені та об'єм і вага апаратури містять вирішальне значення. В цих випадках застосовуються ключові стабілізуючі пристрої.

Основними перевагами ключевих стабілізуючих пристроїв напруження (КСН) є: високий ККД, малі габарити та маса, висока питома сила. Всіх перерахованих властивостей ці джерела живлення набули завдяки вживанню ключового режиму при роботі силових деталей. До недоліків КСН зазвичай відносять: складність схеми, наявність високочастотних шумів та перешкод, збільшення пульсації вихідної напруження, великий період виходу на робочий режим.

					КС 57. 14 000. 00 ДП ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		13

Безперервні та ключові стабілізуючі пристрої, в свою чергу, можливо розділити на компенсаційні та параметричні.

Компенсаційні стабілізуючі пристрої характеризуються наявністю деталей, контролюючих величину вихідної напруги та виробляючих сигнал помилки. Останній через систему зворотного зв'язку впливає на регулювальний елемент, підтримуючи вихідну напругу постійною.

Параметричні стабілізуючі пристрої характеризуються відсутністю деталей, контролюючих вихідну напругу. Робота таких стабілізуючих пристроїв ґрунтується на використанні характеристик нелінійних деталей.

В даному дипломному проекті досліджуються компенсаційні стабілізуючі пристрої напруги ключового типу.

Існують три основні схеми КСН: знижувальних, підвищувальних, інвертуючих. Основні схеми КСН описуються системою математичних виразів, що визначає параметри електричних процесів у силових ланцюгах пристрою та енергетичних співвідношень.

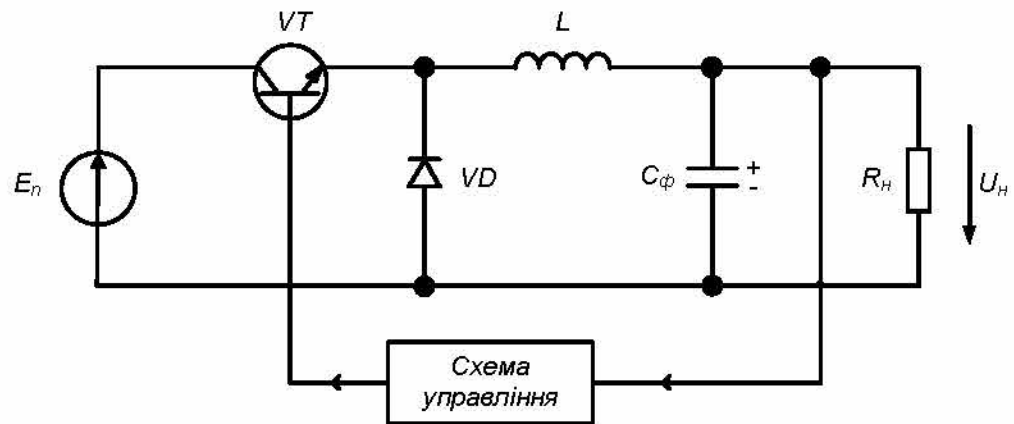
В КСН знижуючого типу (рисунок 1.4 а) регулювальний елемент РЕ та  $L$  включені послідовно із навантаженням  $R_H$ . У якості РЕ використовується транзистор, що працює у режимі перемикачів, при якому він по черзі знаходиться у режимі насичення (коли він повністю відкритий) чи у режимі відсічення (коли він повністю закритий).

При відкритому транзисторі протягом періоду  $t_1$  енергія з вхідного джерела постійного струму (чи випрямляча із вихідною напругою  $U_0$ ) передається у навантаження через дросель  $L$ , у якому накопичується надлишкова енергія. При закритому транзисторі протягом періоду  $t_2$  накопичена у дроселі енергія через діод  $VD$  передається у навантаження. Період комутації (перетворення) рівний  $T_H = t_1 + t_2$ .

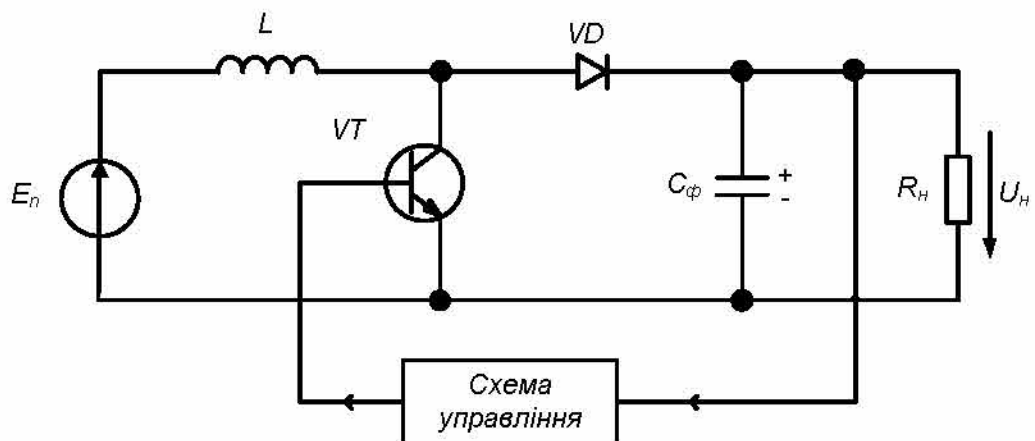
РЕ перетворює (моделює) вхідну постійну напругу  $U_H$  у серію послідовних імпульсів певної тривалості та частоти, а згладжуючий фільтр, що складається із діоду  $VD$ , дроселю  $L$  та конденсатора  $C$  демодлює їх знову у постійну напругу  $U_H$ . При зміні вхідної напруги  $U_H$  чи струму у навантаженні  $R_H$  у імпульсному

					<b>КС 57. 14 000. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		14

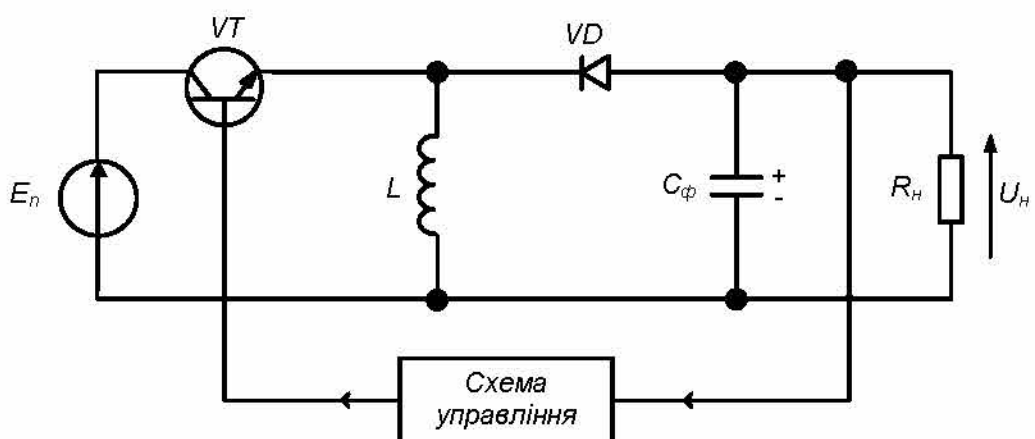
стабілізаторі поза помічною ланцюга зворотного зв'язку тривалість імпульсів змінюється таким чином, що вихідна напруга  $U_H$  залишається стабільною із певним ступенем точності.



а)



б)



у)

Рисунок 1.4. Схеми ключевих стабілізуючих пристроїв напруги: знижуючого (а), підвищувального (б) та інвертуючого (у)

Імпульсний режим роботи дає значно зменшити сила втрат у регулюючому елементі та тим самим підвищити ККД джерела живлення, зменшити його масу та габарити. В цьому полягає вирішальна перевага імпульсних стабілізуючих пристроїв перед безперервними стабілізаторами.

КСН залежно з способу керування регулюючим транзистором можуть виконуватися із широтно-імпульсною модуляцією (ШІМ), частотно-імпульсною модуляцією (ЧІМ) чи релейного типу. В ШІМ-стабілізаторах у процесі роботи змінюється тривалість імпульсу  $t_i$ , а частота комутації залишається незмінною, у ЧІМ – стабілізаторах змінюється частота комутації, а тривалість імпульсу  $t_i$  залишається постійною, у релейних стабілізаторах у процесі регулювання напруження змінюється та тривалість імпульсу та частота; це є їх основним недоліком, що обмежує вживання.

В підвищувальному стабілізаторі (рисунок 1.4 б) регулювальний елемент РЕ (транзистор) підключений паралельно навантаженню  $R_n$ . Діод  $VD$  блокує навантаження  $R_n$  та конденсатор фільтра  $C$  з регулюючого елементу РЕ. Коли регулювальний транзистор відкритий, потік з джерела живлення  $U_n$  протікає через дросель  $L$ , запасаючи у ньому енергію. Діод  $VD$  при цьому відсікає (блокує) навантаження та не дає конденсатору  $C$  розрядитися через відкритий регулювальний транзистор.

Потік у навантаження у цей проміжок періоду поступає тільки з конденсатора  $C$ . В наступний момент, коли регулювальний транзистор закритий, ЕРС самоіндукції дроселю  $L$  підсумовується із вхідною напругою та енергія дроселю віддається у навантаження; при цьому вихідна напруження виявляється більш вхідної напруження живлення  $U_n$ . На відміну з КСН знижуючого типа тут дросель не є елементом фільтра, а вихідна напруження стає більш вхідної на величину, визначувану індуктивністю дроселю  $L$  та шпаруватістю роботи регулюючого транзистора.

Схема керування стабілізатором на рисунку 1.4 б побудована таким чином, що при підвищенні, наприклад, вхідної напруження живлення  $U_n$  зменшується

тривалість відкритого стану  $t_1$  регулюючого транзистора на таку величину, що вихідна напруга  $U_H$  залишається незмінною із певним ступенем точності.

Схема інвертуючого КСН приведена на рисунку 1.4 в. На відміну з попередньої схеми тут паралельно навантаженню  $R_H$  включений дросель  $L$ , а регулювальний елемент РЕ включений послідовно із навантаженням. Блокуючий діод відокремлює конденсатор фільтра  $C$  та навантаження  $R_H$  з регулюючого елемента. Стабілізатор володіє властивістю зміни (інвертування) полярності вихідної стабільної напруги  $U_H$  щодо полярності вхідної напруги живлення. З розглянутих схем найбільше вживання знаходить послідовний імпульсний знижувальний стабілізатор, у якому згладжування пульсацій здійснюється VDLC-фільтром. В стабілізаторах підвищувального типу (рисунки 1.4 б та 1.4 у) дросель  $L$  не бере участь у згладжуванні пульсацій вихідної постійної напруги. В цих схемах згладжування пульсації досягається тільки поза рахунок збільшення ємкості конденсатора  $C$ . Це приводить до збільшення маси та габаритів фільтра та пристрою у цілому.

В проектуванні ключевих стабілізуючих пристроїв напруги важливе значення містить правильний вибір деталей силового ланцюга, що визначає техніко-економічні показники стабілізуючого пристрою у цілому. Характеристики регулювання визначаються у основному схемами керування, у якості котрих використовуються мікросхеми серії К142 чи аналогічні їм.

### 1.2.2 Розрахунок ключевих стабілізуючих пристроїв напруги

Початковими даними задля стабілізуючого пристрою ключевих стабілізуючих пристроїв напруги є:

- вхідна напруга  $E_{вх}$ ;
- вихідна напруга  $E_0$ ;
- робоча частота  $f$ ;
- напруга пульсацій  $U_p$ ;
- коефіцієнт пульсації  $a_{п}$ ;
- мінімальний  $I_{мін}$  та номінальний  $I_0$  струми навантаження.

					<b>КС 57. 14 000. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		17

Таблиця 1.1. Методика стабілізуючого пристрою силових ланцюгів ключевих стабілізуючих пристроїв напруження

Визначуваний параметр	Тип КСН		
	Знижувальний	Інвертуючий	Підвищувальний
Критична індуктивність дроселю $L$	$\frac{500 E_0 (E_{ex} - E_0)}{f I_0 E_{ex}}$	$\frac{500 E_0 E_{ex}^2}{f I_{min} (E_{ex} + E_0)^2}$	$\frac{500 E_{ex}^2 (E_0 - E_{ex})}{f I_{min} E_0^2}$
Середній потік дроселю $I_{др}$	$I_0$	$I_0 \frac{E_{ex} + E_0}{E_{ex}}$	$I_0 \frac{E_0}{E_{ex}}$
Змінна складова струма дроселю $I_{др}$	$\frac{E_0 (E_{ex} - E_0)}{2f E_{ex} L}$	$\frac{E_{ex} E_0}{2f (E_{ex} + E_0) L}$	$\frac{E_{ex} (E_0 - E_{ex})}{2f E_0 L}$
Амплітуда колекторного струма $I_{к макс}$	$I_{др} + I_{др} \sim$		
Ефективний потік колектору $I_k$	$I_0 \sqrt{\frac{E_0}{E_{ex}}}$	$\frac{I_0}{E_{ex}} \sqrt{E_0 (E_0 + E_{ex})}$	$\frac{I_0}{E_{ex}} \sqrt{E_0 (E_0 - E_{ex})}$
Напруження колектор-емітер $U_{кз}$	$E_{ex}$	$E_{ex} + E_0$	$E_0$
Сила, що розсіюється транзистором $P_{тр}$	$I_k \left( 0,07 \cdot E_{ex} + \frac{2E_0}{E_{ex}} \right)$	$I_k \left( 0,07 (E_{ex} + E_0) + \frac{2E_0}{E_{ex} + E_0} \right)$	$\frac{I_k E_0 \cdot (0,07 E_{ex} + 2)}{E_{ex}}$
Середній потік діоду $I_d$	$I_0 \frac{E_{ex} - E_0}{E_{ex}}$	$I_0$	$I_0$
Зворотна напруження діоду $U_{зв}$	$E_{ex}$	$E_{ex} + E_0$	$E_0$
Сила, що розсіюється діодом $P_d$	$0,8 \cdot I_d \frac{E_{ex} - E_0}{E_{ex}}$	$0,8 \cdot I_0 \frac{E_{ex}}{U_{обр}}$	
Вхідний потік стабілізуючого пристрою $I_{вх}$	$\frac{E_0 I_0 + P_{mp} + P_d}{E_{ex}}$		
ККД $\eta$	$\frac{E_0 I_0}{E_{ex} I_{ex}}$		

Методика стабілізуючого пристрою силових ланцюгів ключевих стабілізуючих пристроїв напруження представлена у таблиці 1.1. Окрім вказаних у ній властивостей ще необхідно визначити ємність фільтра. Задля кожного типу стабілізуючого пристрою вона обчислюється по-своєму. Її розрахунок можливо починати після визначення критичної індуктивності дроселю  $L$ .

Задля знижуючого ключевих стабілізуючих пристроїв напруження метод стабілізуючого пристрою наступний. Спочатку обчислюється коефіцієнт згладжування фільтра  $Q$

$$Q = \frac{E_0 \cdot \left[ 2,2 - 2,5 \cdot \left( \frac{E_0}{E_{ex}} - 0,1 \right) \right]}{U_p} \quad (1.1)$$

Добуток індуктивності на ємність фільтра дорівнює

$$X(N) = \frac{25 \cdot (Q^{\frac{1}{N}} + 2)}{f^2}, \quad (1.2)$$

де  $N$  – кількість ланок фільтра. Тоді ємність фільтра

$$C(N) = \frac{X(N)}{L}. \quad (1.3)$$

Амплітуду напруження пульсацій можливо обчислити поза формулою

$$U_{p_{max}}(N) = U_p \cdot Q^{\left(1 - \frac{1}{N}\right)}. \quad (1.4)$$

Задля підвищувального КСН ємність фільтра обчислюється по формулі

$$C = \frac{10^6 \cdot I_0 (E_0 - E_{ex})}{f E_0 U_p}. \quad (1.5)$$

Задля полярно-інвертуючого ключового стабілізатору напруження ємність фільтра дорівнює

$$C = \frac{10^6 \cdot I_0 E_0}{f (E_{ex} + E_0) U_p}, \quad (1.6)$$

а ємність вхідного конденсатора фільтра дорівнює

$$C_{ex} = \frac{10^6 \cdot I_0 E_0}{f a_{\Pi} E_{ex} (E_{ex} + E_0)}. \quad (1.7)$$

### **1.3 Аналітичний огляд програмного забезпечення задля математичного моделювання електронних схем**

У даний період рівень математичного моделювання електронних схем виключно високий. Упевнено моделюються складні фізичні процеси у найрізноманітніших схемах, число типів котрих дійсно невичерпно. Проте тільки останніми роками системи машинного моделювання стали доступними масам інженерів, науковців та студентів.

Зокрема, інтегровані системи моделювання схемотехніки Micro-CAP вигідно відрізняються з інших таких систем задля ПК (наприклад, PSPICE) своїми сервісами.

Найважчий етап розроблення (завдання схеми та її топологічний та математичний опис) у них реалізований простим та наочним графічним діалогом. Він нагадує збірку схем поза поміччю конструктора, що містить кубики із компонентами електронних схем, із котрих радіоаматор збирає потрібну схему. Не вимагається знання жодних вхідних мов задля завдання схем та керування системою у ході аналізу схем та їх моделювання.

Результати аналізу виходять як у числовій (табличній формі), так та в вигляді графіків, що нагадують осцилограми, отримувані при дослідженні схеми поза поміччю електронного осцилографа, характеріографа чи вимірювача частотних характеристик.

Після опису схеми система автоматично формує складні системи нелінійних та диференціальних рівнянь, що описують роботу схем, вирішує їх та виводить результати стабілізуючого пристрою у зручному задля користувача вигляді. Як вона це робить – секрет розробника.

Тільки по окремих нюансах досвідчений фахівець спроможне розпізнати деякі секрети системи. Наприклад, що вона реалізована на мові QuickBasic (на це вказують пускові модулі цієї мови фірми Microsoft Inc.), що при аналізі у тимчасовій області використовується змінний крок у часі (йде адаптація до швидкості модельованих процесів) та так далі.

					<b>КС 57. 14 000. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		20

Отже, головною відмітною особливістю системи автоматизованого аналізу електронних пристроїв (схем) Micro-CAP є завдання електронної схеми, що підлягає стабілізуємого пристрою, у графічному вигляді із контролем її зображення на екрані дисплея.

Система моделювання PSpice призначена задля аналізу аналогових електронних схем різного функціонального призначення: підсилювачів, активних фільтрів, компараторів, генераторів та через це подібне. До складу досліджуваних схем можуть входити елементи наступних видів:

- пасивні елементи R,L,C-елементи;
- незалежні та залежні з періоду джерела напруження та струма;
- напівпровідникові діоди;
- транзистори;
- лінійні багатообмотувальні трансформатори;
- залежні джерела струма, керовані струмом чи напругою;
- залежні джерела напруження, керовані струмом чи напругою;
- функціональні вузли.

Система PSpice дає проводити наступні види аналізу:

- аналіз статичного режиму нелінійних схем та розрахунок передавальної процедури по постійному струму;
- аналіз чутливості зміни статичного режиму та напруження на переходах транзисторів до варіації властивостей деталей схеми;
- аналіз частотних характеристик схеми;
- аналіз внутрішніх шумів схеми у частотній області із обчисленням значення теплового, дробу та фліккер-шуму на виході схеми і шуму, приведенного до входу;
- визначення вкладу кожного джерела шуму у загальний рівень шуму схеми;
- аналіз часових характеристик лінійних та нелінійних схем;
- Фур'є-аналіз у перехідному режимі;
- температурний аналіз.

					<b>КС 57. 14 000. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		21

## **1.4 Вибір операційної системи задля створюваного програмного забезпечення**

Операційна система забезпечує керування всіма апаратними компонентами та підтримку роботи програм користувача, надаючи йому можливість загального керування машиною. При виборі середовища здійснення програми необхідно враховувати декілька чинників, а саме:

- складність та трудомісткість процесу розроблення програмного забезпечення задля конкретного середовища;
- наявність інструментальних засобів розроблення програмного забезпечення;
- можливість внесення коректив у програму у процесі експлуатації;
- наявність засобів розроблення користувача інтерфейсу;
- швидкість здійснення програми;
- надійність роботи програми та захищеність з програмних збоїв.

Можливо виділити ряд переваг задля розроблення програмного забезпечення під управлінням ОС Windows:

- Графічний, призначений задля користувача інтерфейс;
- Незалежність з зовнішніх пристроїв;
- Доступність всього об'єму оперативної пам'яті;
- Підтримка масштабованих шрифтів.

Основна перевага при створенні програмного застосунку задля стабілізуючого пристрою схем напівпровідникових випрямлячів саме під ОС Windows 10 – зручний графічний інтерфейс користувача і широко розповсюджені засоби розроблення.

## **1.5 Короткий аналіз та вибір програмних засобів задля розроблення**

У даному підрозділі будуть розглянуті засоби, використання котрих спроможне існувати доцільним в розробці програмного застосунку задля стабілізуючого пристрою стабілізуючих пристроїв напруження живлення, і із

					<b>КС 57. 14 000. 00 ДП ПЗ</b>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		22

котрих будуть обрані найкращі інструменти. У першу чергу розглянемо доступний набір високорівневих мов програмування, на котрих буде написане ядро даного проекту. Задля поліпшення процесу розроблення мови програмування повинні відповідати наступним вимогам:

1. Зрозумілий і чіткий синтаксис;
2. Статична типізація;
3. Поглиблений інструментарій задля кращого доступу до пам'яті;
4. Наявність вичерпної документації;
5. Підтримка поза потреби інтеграції сторонніх засобів.

У даному випадку мова, яка найбільше підходить задля написання даного програмного продукту буде відібрана на основі відповідності вищезазначеним критеріям і судженню автора про доцільність її використання, через це будуть проаналізовані п'ять наступних мов програмування, обрані поза індексом ТЮВЕ, це: Java, C, C++, C# і Python.

Мова Java – це мова програмування, початково розроблювана компанією Sun Microsystems. Основною особливістю Java-програм є перетворення написаного користувачем тексту в спеціальний байт-код, завдяки чому такі програми можуть існувати запуснені в будь-якому середовищі, що підтримує машину Java. Станом на 2023 рік Java є однією із найпопулярніших мов програмування. Це в свою чергу є гарантією того, що задля рішення існує вичерпна документація. Перевагою здійснення програми на віртуальній машині є повна незалежність виконуваного коду з програмного та апаратного середовищ, що дає запускати програми на Java будь-де, де існує працююча віртуальна машина. Разом з цим одною із особливостей Java є стійка система безпеки, яка заснована на через це, що здійснення програми на Java контролює сама віртуальна машина. Через це дії, що перевищують встановлені права програми, можуть викликати негайне переривання. Програми Java переводяться у байт-код, який виконується відповідною Java-машиною, яка у свою чергу обробляє байт-код та надсилає інструкції до апаратної частини як інтерпретатор. Одним із основних недоліків даної віртуальної машини є суттєве зниження

					<b>КС 57. 14 000. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		23

продуктивності. Однак ряд деяких поліпшень дещо збільшив швидкість здійснення програм Java. Серед них можливо виділити наступні, такі як:

1. Використання технології перекладу байт-коду в машинний код безпосередньо під період здійснення програми (JIT-технологія) із можливістю збереження версій класу у машинному коді;

2. Використання специфічного задля платформи коду в стандартних бібліотеках;

3. Апаратне забезпечення, що значно прискорює оброблення байт-коду (наприклад, технологію Jazelle, підтримувану деякими процесорами архітектури ARM).

Згідно із інформацією, наданою сайтом wikipedia, задля тестового набору задач, період здійснення програми Java у середньому становить в 1.5-2 рази більш, ніж задля C/C++. Варто зазначити, що іноді Java працює швидше, а іноді в сім разів повільніше. Разом з цим одним із недоліків можливо зазначити той факт, що споживання фізичної пам'яті Java у 10–30 разів перевищувало аналогічне задля програми, написаної на C/C++. Разом з цим спроможне заслуговувати на увагу дослідження, що представила компанія Google, яке разом з цим вказує на те, що містить місце значно нижча продуктивність та більш високе використання пам'яті задля програм Java порівняно із аналогічними програмами C++.

У загалом, ідеї, втілені у концепції на різноманітних реалізаціях середовища віртуальних машин Java, надихнули багатьох ентузіастів розширити список мов, котрі можливо застосовувати задля реалізації програм, що працюють поза поміччю віртуальних машин. Ці ідеї разом з цим відображені у специфікації спільної мовної інфраструктури CLI, яка є основою платформи із загальною назвою Microsoft .NET.

Мова C – мова програмування загального призначення. Початковою ціллю розроблення була реалізація в операційній системі UNIX, проте згодом дана мова програмування була перенесена на багато інших платформ. Конструкції цієї мови тісно пов'язані із типовими машинними інструкціями, через це вона

знайшла застосування у проектах, задля котрих доцільно застосовувати мову асемблера, тобто як у операційних системах, так та у різноманітних прикладних програмах задля багатьох пристроїв – з суперкомп'ютерів до вбудованих систем. Мова програмування С суттєво вплинула на розвиток індустрії програмного забезпечення, та його синтаксис став основою задля таких мов програмування, як С++, С# і Java. Мова С була розроблена як мова системного програмування, задля якої можливо створити однопрохідний компілятор. Стандартна бібліотека мови разом з цим не дуже велика. ТА, як наслідок, компілятори можливо розробити відносно легко. Через це ця мова доступна на багатьох платформах. Крім того, незважаючи на свій низький рівень, мова орієнтована на багатоплатформність. Метою мови було полегшити написання великих програм із мінімізацією помилок в порівнянні із асемблером, дотримуючись принципів процедурного програмування, проте уникаючи всього, що спроможне призвести до додаткових накладних витрат, характерних задля мов високого значення. Можливо виділити наступні основні особливості С:

1. Простий синтаксис, із якого в стандартну бібліотеку було перенесено багато основних функцій, таких як математичні процедури чи процедури роботи із файлами;

2. Орієнтація на процедурне програмування;

3. Слаба статична типізація;

4. Використання препроцесора задля абстрагування подібних операцій;

5. Доступ до пам'яті поза поміччю вказівників;

6. Невелика кількість ключевих слів;

7. Наявність покажчиків на процедури та статичних змінних;

8. Області імен.

Однак в С відсутні:

1. Вкладені процедури;

2. Засоби автоматичного керування пам'яттю;

3. Вбудовані засоби об'єктно-орієнтованого програмування;

4. Засоби функціонального програмування.

					<b>КС 57. 14 000. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		25

Деякі із відсутніх можливостей можуть існувати змодельовані поза поміччю вбудованих інструментів (наприклад, імітацію властивостей можливо виконувати поза поміччю функцій `setjmp` та `longjmp`), деякі додаються поза поміччю сторонніх бібліотек (наприклад, задля підтримки багатозадачності і задля мережевих функцій можуть застосовувати `pthread`, сокети тощо, є бібліотеки задля підтримки автоматичного збору сміття), частина реалізована у деяких компіляторах як розширення мови (наприклад, вкладені процедури у компілятор `GCC`). Разом з цим існує дещо громіздкий, проте цілком працездатний метод, який дає реалізувати механізми ООП, засновані на фактичному поліморфізмі вказівників в мові `C` та підтримці вказівників на процедури в цій мові. `GLib` забезпечує базовий клас `GObject`, здатність спадкування з одного класу та реалізацію декількох інтерфейсів. У загалом, мова досить розповсюджена, що виступає гарантом того, що мова підтримується на різноманітних платформах і задля неї існує сформована і вичерпна документація, яку зручно застосовувати задля розроблення програмного забезпечення, а отже програмісти містять можливість досить точно зрозуміти, як саме виконуються їхні програми. Завдяки своїй близькості до мов низького значення програми на мові `C` працюють більш ефективно, аніж написані на багатьох інших мовах високого значення.

Мова програмування `C++` – це складена, статично типізована мова програмування загального призначення. Підтримує такі парадигми програмування, як процедурне програмування, об'єктно-орієнтоване програмування, загальне програмування, а разом з цим містить багату стандартну бібліотеку, яка включає загальні контейнери і алгоритми, регулярні вирази, підтримку багатопоточність і інші процедури. `C++` поєднує характеристики як мов високого значення, так та мов низького значення. В порівнянні зі своїм попередником, мовою `C`, найбільшу увагу приділено підтримці об'єктно-орієнтованого та узагальненого програмування. Мова `C++` широко використовується задля розроблення програмного забезпечення і є однією із найпопулярніших мов програмування. Її сфера застосування охоплює

					<b>КС 57. 14 000. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		26

реалізації операційних систем, різноманітні прикладні програми, драйвери пристроїв, додатки задля вбудованих систем, високопродуктивні сервери і ігри. У загалом існує багато реалізацій даної мови програмування, як вільних, так та комерційних, та задля різноманітних платформ. Разом з цим варто зазначити, що C++ мала величезний вплив на інші мови програмування, насамперед на Java та C#. Синтаксис C++ успадковується з C, адже одним із основних принципів розроблення була підтримка сумісності із C. Мова C++ містить засоби розроблення програмного забезпечення із контрольованою ефективністю задля широкого кола задач, з утиліт низького значення і драйверів до дуже складних програмних пакетів. Серед основних переваг цієї мови можливо виділити наступні:

1. Висока сумісність із мовою C. Тобто код C можливо компілювати із мінімальними модифікаціями компілятором C++. Зовнішній мовний інтерфейс прозорий, через це бібліотеки C можливо викликати із C++ без додаткових витрат;

2. Як наслідок попереднього пункту – обчислювальна продуктивність. Ця мова покликана дати програмісту максимальний контроль над усіма аспектами структури і порядку здійснення програми. Разом з цим існує можливість працювати із пам'яттю на низькому рівні;

3. Підтримка різноманітних стилів програмування: традиційне імперативне програмування (структуроване, об'єктно-орієнтоване), узагальнене програмування, функціональне програмування;

4. Автоматичний виклик деструкторів об'єктів у належному порядку (зворотний виклик задля конструкторів) спрощує та підвищує надійність керування пам'яттю і іншими ресурсами (відкриті файли, мережні підключення, із'єднання із базами даних тощо);

5. Перевантаження оператора дає коротко та просторо писати вирази задля користувацьких типів у природному алгебраїчному вигляді;

6. Можливо контролювати константність об'єктів (модифікатори `const`, `mutable`, `volatile`). Використання постійних об'єктів підвищує надійність та служить підказками задля оптимізації;

7. Шаблони C++ [6] дозволяють створювати загальні контейнери і алгоритми задля різноманітних типів даних. Разом з цим шаблони дозволяють виконувати розрахунки на етапі компіляції;

8. Можливість вбудовування предметно-орієнтованих мов програмування у основний код. Цей підхід використовує, наприклад, бібліотеку `Boost.Spirit`, яка дає вказувати граматику EBNF парсерів безпосередньо у C++ кодї;

9. Доступність. Задля C++ існує величезна кількість підручників, перекладених на різні мови. Та хоча мова містить високий поріг вступу, проте серед усіх мов такого роду він містить найширші можливості. Із попередньо перерахованих пунктів випливає те, що дана мова містить в собі весь необхідний інструментарій задля реалізації майже будь-якої програми, а її розповсюдженість є гарантом наявності вичерпної документації.

Мова C# – об'єктно-орієнтована мова програмування, розроблена групою інженерів Microsoft як мова розроблення додатків задля Microsoft .NET Framework. Мова C# відноситься до сімейства мов із C-подібним синтаксисом та його синтаксис найбільш близький до мов C++ чи Java. Мова статично типізована, підтримує поліморфізм, перевантаження оператора, делегати, атрибути, події, характеристики, загальні типи та методи, ітератори і анонімні процедури із закриттям. Мова C# багато перейняла з своїх попередників – C++, Pascal, Modula, Smalltalk та, зокрема, Java. Через це C#, виходячи із практики їх використання, виключає деякі моделі, що виявилися проблематичними при розробці програмного забезпечення. Наприклад, C#, на відміну з C++ та деяких інших мов, не підтримує наслідування з декількох класів. Мова C# була розроблена як мова на рівні додатків задля CLR та через це залежить насамперед з можливостей самого CLR. Це стосується, у першу чергу, системи типа C#, яка відображає BCL. Наявність чи відсутність певних виразних ознак мови диктується тим, чи спроможне конкретна мовна особливість існувати переведена

в відповідні конструкції CLR. Та хоча таку взаємодію та слід очікувати у майбутньому, вона частково була порушена із виходом C# 3.0, яка є розширенням мови, що не покладається на розширення .NET. Разом з цим C# підтримує інструменти задля безпосередньою роботи із пам'яттю, проте такі шматки коду містять позначатися ідентифікатором unsafe, і вимагають спеціального дозволу. Згідно із інформацією, наданою сайтом mattwarren.org, код, написаний на C# в порівнянні із C/C++ є приблизно на 30% повільніший поза рахунок більшої різниці синтаксису з низькорівневого машинного коду, який виконується комп'ютером. Та хоча існуюча документація стандартних бібліотек спроможне відрізнитися з версії до версії, у загалом вона цілком прийнятна і добре зрозуміла задля сприйняття.

Мова Python – мова програмування загального призначення, орієнтована на підвищення продуктивності розробника та читаності коду. Синтаксис, підтримуваний ядром Python, містить досить малий об'єм. Python підтримує структуроване, об'єктно-орієнтоване, функціональне, імперативне і аспектно-орієнтоване програмування. Основними архітектурними особливостями є динамічна типізація, автоматичне керування пам'яттю, повна інтроспекція, механізм обробки виключень, підтримка багатопотокових обчислень, високорівневі структури даних. Разом з цим підтримується поділ програм на модулі, котрі, в свою чергу, можуть існувати об'єднані у пакети. Python є мовою програмування, що активно розвивається, нові версії із додаванням/змінною властивостей мови виходять приблизно раз на два із половиною роки. Мова не підлягала офіційній стандартизації, роль стандарту фактично виконувала CPython, розроблена під контролем автора мови. Варто зазначити, що портування ядра Python виконано майже задля всіх платформ, однак в той же період, на відміну з багатьох інших портативних систем, Python підтримує технології, характерні задля цієї платформи. Прикладом спроможне слугувати Microsoft COM / DCOM. Більш того, існує версія Python розроблена задля Java – Jython, яка дає інтерпретатору працювати на усіх системах, котрі підтримують Java, тоді як Java-класи можливо застосовувати безпосередньо із Python та навіть

					<b>КС 57. 14 000. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		29

написані на Python. Разом з цим декілька розширень містять поза мету забезпечити сумісність Microsoft .NET, головною із котрих є IronPython та Python.Net. Із зазначених вище тез впливає те, що Python хоча та залежний з встановлення свого ядра, так само як та Java, однак спроможне виконуватися на великій кількості сучасних платформ. Разом з цим Python є популярною мовою програмування, насамперед задля рішень різноманітних математичних алгоритмів. З цього впливає величезна кількість вичерпної технічної документації, яка майже повністю покриває опис доступних модулів.

У даному підрозділі буде проведено порівняння мов програмування поза такими параметрами як: швидкодія, зручність синтаксису, гнучкість засобів задля роботи із компіляторами і доступність навчальних засобів і документації. Оцінка буде проводитись поза шкалою з 0 до 10 і є відображенням власної думки автора. Результати порівняння можливо переглянути в табл. 1.2.

Таблиця 1.2. Порівняльна характеристика властивостей мов програмування

	Java	C	C++	C#	Python
Швидкодія	7	10	10	8	8
Синтаксис	8	8	8	10	9
Засоби	9	5	10	8	9
Документація	10	9	10	10	10
Середня оцінка	8	8	9.5	9	9

У даному випадку вирішено віддати перевагу мові C++, через це що дана мова поєднує в собі характеристики високорівневих мов програмування із можливістю роботи із пам'яттю, а разом з цим містить велику кількість різноманітних матеріалів на дану тему. Отже, беручи до уваги власний досвід в програмуванні у інтегрованому середовищі розроблення RAD Studio C++Builder і вищезазначені аргументи можливо вважати доцільним обрати мовою реалізації програмного застосунку задля стабілізуючого пристрою стабілізуючих пристроїв напруження живлення саме мову програмування C++ і ICP Embarcadero RAD Studio C++Builder.



В інтегрованому середовищі розроблення Embarcadero RAD Studio C++Builder вдало поєднуються засоби візуального розроблення додатків та оптимізуючий компілятор. Проте сучасних розробників хвилює не тільки ефективність виконуваного коду. Їм потрібний засіб, який був би, із одного боку, досить потужним та гнучким, щоб виконати будь-яке завдання, що стоїть перед ними, та, із іншого боку, досить простим та зручним у роботі. Розробники хочуть мати систему, побудовану на принципах об'єктно-орієнтованого програмування, та у той же період, що дає застосовувати, в разі потреби, асемблер. Їм потрібна система, яка генерувала б не тільки виконувані файли в форматі EXE, проте та могла створювати файли DLL та драйвери пристроїв.

До складу C++Builder входить велика бібліотека деталей, поза поміччю якої можливо уникнути ручного написання програм, що широко поширене у інших засобах розроблення. Із іншого боку, програміст завжди спроможне звернутися до низькорівневих асемблерних процедур. Можливо створювати додатки в візуальному режимі, просто поміщаючи потрібні елементи на форму та, у той же період, зберігаючи доступ до всіх функцій програмного інтерфейсу Windows, системних повідомлень та процесів. Працюючи у C++Builder програміст спроможне поза поміччю натиснення однієї клавіші створити виконуваний файл в форматі EXE, проте, при необхідності, можливо компілювати та файли DLL, драйвери пристроїв, а разом з цим консольні додатки.

Використовуючи C++Builder можливо створювати додатки задля MS Windows із мінімальними витратами періоду оскільки у її основі лежить концепція швидкого реалізації додатків (RAD).

Інтегроване середовище розроблення додатків дає створювати, компілювати, тестувати та редагувати проект чи групу проектів у єдиному середовищі програмування.

Візуальна технологія розроблення програм дає швидко створювати додатки шляхом розміщення в формі стандартних деталей. При цьому відповідний код програми автоматично генерується C++Builder. Така технологія

визволяє розробника з рутинної роботи із реалізації призначеного задля користувача інтерфейсу та дає приділити більш уваги внутрішній організації даних та обробці даних.

Технологія Two Ways Tools робить ефективнішою роботу із компонентами. При зміні програмного коду в вікні редактора C++Builder відповідним чином змінює та самі елементи. Із іншого боку, при зміні властивостей деталей у інспекторві редактора об'єктів (Object Inspector) вони негайно відбиваються в вікні редактора коду.

Насамперед C++Builder призначений задля професіоналів-розробників корпоративних інформаційних систем. Проте C++Builder призначений не тільки задля програмістів-професіоналів. Будь-який програміст на C++ здатний практично відразу професійно освоїти C++Builder.

На відміну з більшістю компіляторів, що транслюють у р-код, у C++Builder програмний текст компілюється безпосередньо у машинний код, внаслідок чого C++Builder-додатки виконуються у 10-20 разів швидше (особливо додатки, що використовують математичні процедури).

Завдяки такій архітектурі додатки, виготовлені поза поміччю C++Builder, працюють надійно та стійко. Embarcadero RAD Studio C++Builder підтримує використання вже існуючих об'єктів, включаючи DLL, написані на C та C++, OLE сервера, VBX, об'єкти, створені поза поміччю C++Builder. Із готових компонент працюючі додатки збираються дуже швидко.

Бібліотека об'єктів VCL включає стандартні об'єкти побудови призначеного задля користувача інтерфейсу, об'єкти керування даними, графічні об'єкти, об'єкти мультимедіа, діалоги та об'єкти керування файлами, керування DDE та OLE.

### **1.6.1 Застосування деталей введення та відображення текстової інформації**

У бібліотеці візуальних деталей VCL є багато деталей, що дозволяють відображати, вводити та редагувати текстову інформацію.

Елементи Label, StaticText, Panel використовуються задля відображення

різноманітних написів на формі. Основне призначення панелі Panel – компоновка деталей в вікні форми. Однак панель можливо застосовувати та задля виведення текстів. Тексти, котрі відображаються у компонентах, є значеннями їх характеристики Caption. Його можливо встановлювати у процесі розроблення чи задавати та змінювати програмно під період здійснення програми. Характеристика Caption містить тип рядка AnsiString. При присвоєнні цьому типу числової інформації відбувається її автоматичне перетворення у рядок. Коли потрібно занести у мітку змішану інформацію, що складається із рядків символів та чисел, то необхідно скористатися функціями FloatToStr та IntToStr, що переводять відповідно числа із плаваючою комою та цілі у рядок, та операцією «+», яка задля рядків означає їх конкатенацію (склеювання).

В всіх компонентах колір фону обчислюється властивістю Color, а колір напису – підвластивістю Color характеристики Font. Задля мітки Label колір та шрифт – єдино доступні елементи оформлення напису. Елементи StaticText та Panel містять ще характеристика BorderStyle, що визначає рамку тексту – бордюр, та Panel – BevelInner, що дає різноманітно оформляти написи. Вертикальний та горизонтальний розмір Label та StaticText визначаються розміром напису, коли характеристика AutoSize встановити у true. У мітці Label є характеристика WordWrap – допустимість перенесення слів довгим написом, що перевищує довжину компонента, на нову сходинку. Задля здійснення переносу потрібно WordWrap встановити у true, AutoSize – у false, а горизонтальний та вертикальний розмір мітки зробити такими, щоб у ній можливо було помістити потрібне число рядків прийнятної довжини. У мітці StaticText перенос довгого тексту здійснюється автоматично, коли AutoSize встановити у false, а вертикальний розмір мітки – достатнім задля розміщення декількох рядків.

В однорядкових вікон редагування Edit, LabeledEdit та MaskEdit зовнішнє оформлення обчислюється властивістю BorderStyle. Введений та виведений текст міститься в характеристики Text типа AnsiString. Цю характеристика можливо встановлювати у процесі розроблення чи задавати програмно.

					<b>КС 57. 14 000. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		34

Перенесення рядків неможливе. Текст, що не міститься по довжині в вікно, просто зсувається, та можливо пересуватися по ньому поза поміччю курсору. Характеристика `AutoSize` дає встановлювати автоналаштування під розмір тексту тільки висоти, проте не ширини віконця.

Характеристика `AutoSelect` визначає, чи буде автоматично виділятися весь текст при передачі фокусу в вікно редагування. Воно задається рівним `true`, коли при перемиканні у це вікно користувач буде замінювати поточний текст, а не виправляти його.

Характеристики тільки періоду здійснення `SelLength`, `SelStart`, `SelText` визначають відповідно довжину виділеного тексту, позицію перед першим символом виділеного тексту та сам виділений текст.

Віконця редагування можливо застосовувати та просто як елементи відображення тексту. Задля цього треба встановити у `true` їх характеристика `ReadOnly` та доцільно встановити `AutoSelect` у `false`. В цьому випадку користувач не зможе змінювати відображений текст, та вікно редагування стає подібним міткам. Відмінності з останніх – віконця інакше оформлені та можуть вміщати текст, що перевищує їх довжину. В цьому випадку користувач спроможне прокручувати текст, переміщаючи курсор у вікні.

Характеристика `Text` містить тип рядка `AnsiString`. При присвоєнні цього типа числової інформації відбувається її автоматичне перетворення у рядок. Проте при введенні із віконця числової інформації треба застосовувати процедури `StrToFloat` – перетворення рядка у значення із плаваючою комою та `StrToInt` – перетворення рядка у ціле значення.

Характеристика `MaxLength` визначає максимальну довжину тексту, що вводиться. Коли `MaxLength=0`, то довжина тексту не обмежена. У іншому випадку `MaxLength` вказує максимальне число символів, яке спроможне ввести користувач.

Характеристика `Modified`, доступне тільки під період здійснення, показує, чи проводилося редагування тексту в вікні. В момент початку роботи користувача із текстом характеристика треба поставити у `false`. Тоді при

наступному зверненні до нього можливо по його значенню (true чи false) встановити, було чи не було редагування.

У компоненті LabeledEdit, крім самого віконця редагування, що не відрізняється поза виглядом з Edit, є мітка. Вона задається властивістю EditLabel із підвластивостями Caption, Color, Font, WordWrap тощо.

Характеристика LabelPosition компонента LabeledEdit вказує, із якого боку з віконця розміщується мітка: lpAbove – зверху, вирівняна по лівому краю, lpBelow – знизу, вирівняна по лівому краю, lpLeft – зліва, lpRight – праворуч.

У компоненті MaskEdit можливо задавати рядок маски в характеристиці EditMask. Це дає забезпечити синтаксично безпомилкове введення користувачем таких даних, як номери телефонів, паспортні дані, адреси, дати, період тощо.

Елементи Memo та RichEdit є вікнами редагування багаторядкового тексту. Розглянемо спочатку основні відмінності деталей. У Memo формат (шрифт, його атрибути, вирівнювання) однаковий задля всього тексту та обчислюється властивістю Font. RichEdit працює із текстом у збагаченому форматі RTF. Атрибути що знов вводяться фрагментом тексту можливо задати властивістю SelAttributes, яке містить підвластивості: Color (колір), Name (ім'я), Size (розмір), Style (стиль) тощо. Встановлені атрибути впливають на виділений текст чи, коли нічого не виділено, на атрибути нового тексту, що вводиться, починаючи із поточної позиції курсора (обчислюється властивістю SelStart). У RichEdit є разом з цим характеристика DefAttributes (доступно тільки під період здійснення), що містить атрибути поза стандартом, котрі діють, поки не зміняться атрибути в характеристиці SelAttributes. Проте значення атрибутів у DefAttributes зберігаються та у будь-який момент можуть існувати методом Assign присвоєні атрибутам характеристики SelAttributes, щоб повернутися до колишнього стилю. Значення підвластивостей характеристики Paragraph можливо задавати тільки у процесі здійснення програми, наприклад при натисканні кнопки.

Основна характеристика Memo та RichEdit – Lines, що містить текст віконця в вигляді списку рядків. Початкове значення тексту можливо встановити у процесі розроблення, натиснувши кнопку із трьома крапками біля

характеристики Lines в вікні Інспектора Об'єктів. В вікні Редактор рядка списку можливо редагувати чи вводити текст.

Під період здійснення програми можливо заносити текст в вікно поза поміччю методів характеристики Lines. Весь текст, поданий одним рядком типа String, всередині якої використовуються роздільники типа символів повернення каретки та переведення рядка, міститься в характеристиці Text. Характеристика тільки задля читання Count вказує число рядків у тексті. Задля очищення тексту у вікні використовується функція Clear(). Задля занесення нового рядка у кінець тексту віконця редагування користуються методами Add чи Append характеристики Lines. Задля роботи із файлом використовуються методи LoadFromFile та SaveToFile.

### **1.6.2 Застосування панелей і деталей зовнішнього оформлення**

Панелі є контейнерами, що слугують задля об'єднання інших керуючих деталей, пов'язаних один із одним поза призначенням, чи задля організації їх спільної роботи.

Панелі Panel використовуються найбільш широко. Із їх поміччю компонуються різні елементи інтерфейсу (кнопки, віконця редагування, списки), функціонально пов'язані один із одним. Панелі Panel можуть разом з цим використовуватися задля організації інструментальних панелей, смуг стану та т.п.

Одним із призначень панелей є разом з цим групування таких керуючих деталей, як RadioButton – радіокнопки. Аналогічно узгодженої групою працюють та розташовані на панелі швидкі кнопки SpeedButton, коли вони містять однакове значення характеристики GroupIndex.

Зовнішній вигляд панелі Panel обчислюється сукупністю властивостей BevelInner – стиль внутрішньої частини панелі, BevelOuter – стиль зовнішньої частини панелі, BevelWidth – ширина зовнішньої частини панелі, BorderStyle – стиль бордюру, BorderWidth – ширина бордюру.

Роздільник Splitter надає користувачеві можливість переміщати кордони, що розділяють різні панелі, змінюючи їх відносні розміри. Роздільник містить

					<b>КС 57. 14 000. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		37

характеристики `Beveled` та `ResizeStyle`, що визначають його вид, та характеристика `MinSize`, що обмежує мінімальний розмір панелей по обидва боки з роздільника.

Панель `GroupBox` містить вбудовану рамку із написом, яка зазвичай використовується задля виділення на формі групи функціонально об'єднаних деталей.

Компонент `Bevel` формально не є панеллю, він не спроможне служити контейнером задля деталей. Наприклад, поза поміччю `Bevel` можливо згрупувати радіокнопки.

Стиль відображення `Bevel` обчислюється властивістю `Style`, яке спроможне набувати значень `bsLowered` – втоплений, та `bsRaised` – піднятий. А контур компонента обчислюється властивістю `Shape`, яке спроможне набувати значень: `bsBox` – прямокутник, `bsFrame` – рамка, `bsSpacer` – пунктирна рамка, `bsTopLine`, `bsBottomLine`, `bsLeftLine`, `bsRightLine` – відповідно верхня, нижня, ліва та права лінії. Залежно з значення `Style` лінії можуть існувати утопленими чи виступаючими.

Панель із прокруткою `ScrollBox` призначена задля реалізації області, у якій можуть розміщуватися елементи, що займають площу більшу, ніж сам `ScrollBox`. Наприклад, `ScrollBox` можливо застосовувати задля розміщення довгих текстових рядків чи великих інструментальних панелей, котрі, виходячи із міркувань економії площі віконця, недоцільно відображати цілком. Коли розміри `ScrollBox` менше, ніж розміщені елементи, то із'являються смуги прокручування, котрі дозволяють користувачеві переміщатися по всіх розміщених у `ScrollBox` компонентах.

Розмістити в межах невеликої області `ScrollBox` великі елементи чи багато деталей, котрі займають у сумі велику площу, можливо у процесі розроблення наступним чином. Розміри `ScrollBox` збільшують тимчасово так, щоб у смузі прокрутки помістилося все, що потрібно розмістити, а потім скорочують розміри `ScrollBox` до необхідних.

Характеристика `BorderStyle` визначає стиль рамки компонента `ScrollBox`. Характеристика `AutoScroll` дає задати автоматичну появу необхідних смуг прокрутки, коли розмір розміщених деталей перевищує розмір області по горизонталі, вертикалі чи у обох вимірах. Проте можливо та самим керувати появою горизонтальною та вертикальною смуг поза поміччю властивостей `HorzScrollBar` та `VertScrollBar` відповідно. Значення переміщення при одноразовому натисканні користувачем кнопки прокрутки спроможне розраховуватися компонентом автоматично виходячи із розмірів області та віконця, коли характеристика `Smooth` встановлено у `true`. У іншому випадку потрібно задати величину одиничного переміщення в характеристики `Increment`.

Багатосторінкові панелі `TabControl`, `PageControl`, `TabSet`, `TabbedNotebook`, `Notebook` дозволяють економити простір віконця програми, розміщуючи на одному та через це ж місці сторінки різного змісту. Основні події компонента – `OnChanging` та `OnChange`. При цьому у обробник події передається по посиланню параметр `AllowChange` – дозвіл перемикачання. Коли у обробнику задати `AllowChange = false`, то перемикачання не відбудеться. Подія `OnChange` відбувається відразу після перемикачання.

Компонент `TabControl` зовні виглядає так само, як `PageControl`, та містить багато тих же властивостей: `Style`, `MultiLine`, `TabPosition`, `TabHeight`, `TabWidth`, `Images`, `ScrollOpposite`, ті ж події `OnChanging` та `OnChange`. Проте принципова відмінність його з `PageControl` полягає у через це, що `TabControl` не містить безлічі панелей (сторінок). Компонент являє собою одну сторінку із керуючим елементом типа кнопки із багатьма положеннями. Застосовувати компонент `TabControl` доречно у застосунках, у котрих потрібен багатопозиційний перемикач.

## **1.7 Реалізація алгоритму стабілізуючих пристроїв напруги**

Алгоритм стабілізуючого пристрою ключевих стабілізуючих пристроїв напруження із ємкісною реакцією навантаження складено по методиці стабілізуючого пристрою, розглянутій в підрозділі 1.2.

					<b>КС 57. 14 000. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		39

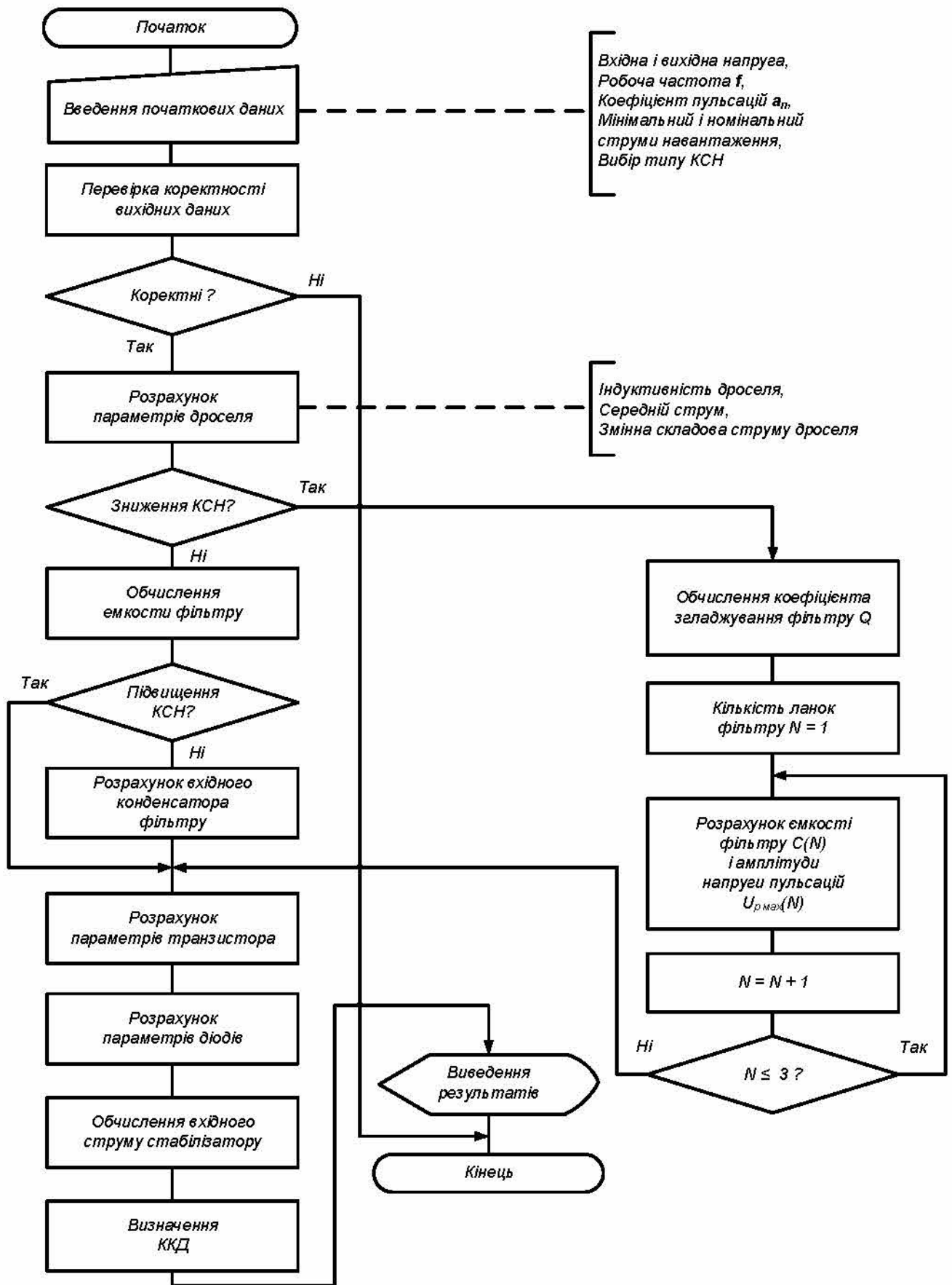


Рисунок 1.6. Алгоритм стабілізуючого пристрою ключового стабілізуючого пристрою напруги

Слід ще раз звернути увагу, що при ручному стабілізуючого пристрою деякі параметри та допоміжні коефіцієнти визначаються по графіках та таблицях. При програмній реалізації алгоритму використовуються відповідні апроксимуючі формули, по котрих отримують приблизне значення. При дотриманні обмежень на вхідні дані отримуваний результат містить достатній ступінь точності. Інакше, при порушенні певних умов, у алгоритмі передбачено виведення повідомлення про зниження точності стабілізуючого пристрою.

Після введення початкових даних здійснюється їх діагностика на наявність їх некоректності. На вхідні дані накладаються наступні обмеження:

- в початкових даних недопустимі негативні числа;
- задля знижуючого ключового стабілізуючого пристрою напруження вхідна напруження  $E_{вх}$  містить існувати більш випрямленої напруження  $E_0$ ;
- задля підвищувального ключового стабілізуючого пристрою напруження навпаки, вхідна напруження  $E_{вх}$  містить існувати менше випрямленої напруження  $E_0$ .

Поза наявності у початкових даних негативних чисел видається помилка та подальший розрахунок неможливий. Коли порушена якась інша умова, то буде тільки видане попередження про зниження точності результату та питання про необхідність продовження стабілізуючого пристрою.

На рис. 1.6 представлено розроблений алгоритм стабілізуючого пристрою ключового знижуючого стабілізуючого пристрою напруження, полярно-інвертуючого та підвищувального типів. Нижче представлено фрагмент програми, що реалізовує даний алгоритм мовою C++, і відповідні коментарі, що пояснюють роботу програми:

```
// Функція, що викликається при стабілізуючого пристрою стабілізуючого
пристрою напруження
void __fastcall TApp::btnStabilizerClick(TObject *Sender)
{
    // Вхідні дані
    double E2, E1, UP, YM, Y0, A;
    int F;
```

```

    E2 = editS_E2->Text.ToDouble(); // вхідна напруга, У
    E1 = editS_E1->Text.ToDouble(); // вихідна напруга, У
    F = editS_F->Text.ToInt(); // робоча частота, кГц
    UP = editS_UP->Text.ToDouble(); // напруга пульсацій, У
    YM = editS_YM->Text.ToDouble();
        // мінімальний потік навантаження, А
    Y0 = editS_Y0->Text.ToDouble();
        // номінальний потік навантаження, А
    A = editS_A->Text.ToDouble(); // коеф. пульсацій вхідного U
// Перевірка коректності
if( !(E2 > 0 && E1 > 0 && F > 0 && UP > 0
    && YM > 0 && Y0 > 0 && A > 0) ) {
    ErrMess(
        "Вхідні дані повинні існувати\позитивними числами!" );
        return;
    }
//----- Розрахунок вихідних властивостей -----
double D // індуктивність дроселю
    Y1 // середнє значення струма через дросель
    Y2 // амплітуда колекторного струма
    Y3 // значення діючого колекторного струма
    Y4 // середній потік діоду
    Y5 // змінна складова струма дроселю
    U2 // напруга колектор-емітер
    U0 // зворотна напруга діоду, У
    PT // сила розсіювання на транзисторі
    PD // сила розсіювання на діоді
    YB // вхідний потік стабілізуючого пристрою
    C1 // ємність вхідного конденсатора фільтра
    C2 // ємність фільтра
    R; // ККД
double Q // коеф. згладжування фільтра
    C[3] // ємність фільтра
    U[3]; // амплітуда напруги пульсацій
// id - тип стабілізуючого пристрою
int id = comboStabilizer->ItemIndex;
    if( id == 0 ) // знижуючий СН
    {
        if( E2 <= E1 ){
            ErrMess(
                "Вихідна напруга містить існувати менше вхідної" );
                return;
            }
        }
    D = E1*(E2-E1)/(2*F*YM*E2);
    Y1 = Y0;

```

```

// коеф. згладжування фільтра
Q = (2.2*E1 - 2.5*E1*(E1/E2-.1))/UP;
for( int i = 0; i < 3; i++ )
{
double N = i + 1;
// добуток L*C фільтра
int X = 1000*N*N*(pow(Q,1/N)+ 2)/(40*F*F);
// ємність фільтра, мкФ
C[i]= X/(D*N*N);
// напруження пульсацій, У
U[i]= E1*(2.2 - 2.5*(E1/E2-.1)) /pow(Q,1/N);
}
Y5 = E1*(E2-E1)/(2*F*E2*D);
U2 = U0 = E2;
Y3 = Y0*sqrt(E1/E2);
Y4 = Y0*(E2 - E1) /E2;
PT = Y3*(.07*E2 + 2*E1/E2);
PD = .8*Y4*(E2 - E1) /E2;
}
else if( id == 1 ) // полярно-інвертуючий СН
{
D = E1*E2*E2/(2*F*YM*pow((E1+E2),2));
Y3 = Y0*sqrt(E1*(E1+E2)) /E2;
Y4 = Y0;
Y5 = E1*E2/(2*F*(E1+E2)*D);
U2 = U0 = E1 + E2;
C1 = 1000*Y0*E1/(F*E2*(E1+E2)*A);
C2 = 1000*Y0*E1/(F*(E1+E2)*UP);
PT = Y3*(.07*U2 + 2*E1/U2);
PD = .8*Y0*E2/U2;
}
else // if( id == 2 ) // підвищувальний СН
{
if( E2 >= E1 ){
ErrMess(
"Вихідна напруження містить існувати більш вхідної" );
return;
}
D = E2*E2*(E1-E2)/(2*F*YM*E1*E1);
Y3 = Y0*sqrt(E1*(E1-E2)) /E2;
Y4 = Y0;
Y5 = E2*(E1-E2)/(2*F*E1*D);
U2 = U0 = E1;
C2 = 1E3*Y0*(E1-E2)/(F*E1*UP);
PT = Y3*E1*(.07*E2 + 2) /E2;
}

```



```

"Середнє значення струма через дросель, A, Y1); t += buf;
    sprintf(buf " %-45s%6.2fr\n"
"Амплітуда колекторного струма, A, Y2);           t += buf;
    sprintf(buf " %-45s%6.2fr\n"
"Значення колекторного струма, що діє, A, Y3);
t += buf;
    sprintf(buf " %-45s%6.2fr\n"
"Середній потік діоду, A, Y4); t += buf;
    sprintf(buf " %-45s%6.2fr\n"
"Змінна складова струма дроселю, A, Y5); t += buf;
    sprintf(buf " %-45s%6.2fr\n"
"Напруження колектор-емітер, У", U2);           t += buf;
    sprintf(buf " %-45s%6.2fr\n"
"Сила розсіяння на транзисторі, Вт", PT);t += buf;
    sprintf(buf " %-45s%6.2fr\n"
"Сила розсіяння на діоді, Вт", PD); t += buf;
    sprintf(buf " %-45s%6.2fr\n"
"Зворотна напруження діоду, У", U0);t += buf;
    sprintf(buf " %-45s%6.2fr\n"
"Вхідний потік стабілізуючого пристрою, A, YB); t += buf;
    sprintf(buf " %-45s%6.2fr\n", "ККД", R); t += buf;
// strStabilizerResult – рядок, що містить результати стабілізуючого
пристрою
strStabilizerResult = t;
// Виведення даних на екран
// Коли встановлено прапорець "+ вхідні дані",
// то вони теж будуть виведені
checkInputStabilizerClick(0);
}

```

## 1.8 Реалізація візуального інтерфейсу і тестування розробленого застосунку

В даному підрозділі буде розглянуто зовнішній вигляд спроектованого візуального інтерфейсу і можливості створеного програмного застосунку. На рис. 1.7 наведено робоче вікно застосунку задля стабілізуючого пристрою властивостей схем ключевих стабілізуючих пристроїв напруження різноманітних типів. В верхній частині віконця поза помічкою компонента ComboBox в контейнері GroupBox обирається тип стабілізуючого пристрою напруження: знижувальний, полярно-інвертуючий чи підвищувальний і вводяться вхідні дані поза помічкою деталей Edit. При цьому в звіт можливо додати разом з цим вхідні

					<b>КС 57. 14 000. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		45

вхідні дані поза поміттю активації компонента CheckBox. Поза бажанням результати стабілізуючого пристрою можливо роздрукувати поза поміттю кнопки Button, що спирається на дію компонента PrintDialog.

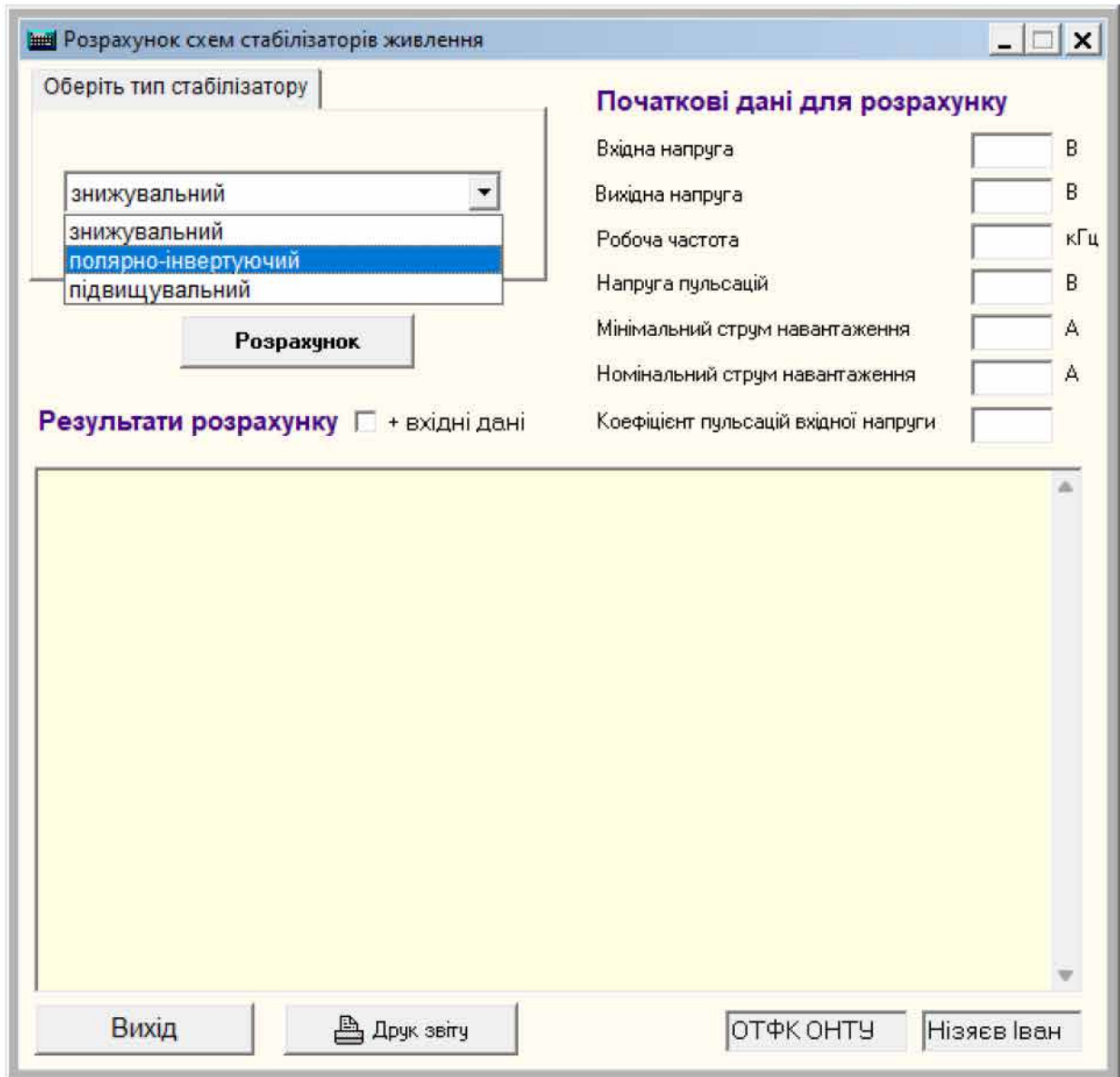


Рисунок 1.7. Візуальний інтерфейс застосунку задля стабілізуючого пристрою схем ключевих стабілізуючих пристроїв

### 1.8.1 Тестування в застосунку в режимі стабілізуючого пристрою стабілізуючого пристрою знижуючого типу

На рис. 1.8 наведено робоче вікно застосунку під період стабілізуючого пристрою властивостей схеми ключового стабілізуючого пристрою напруження знижуючого типу. Задля порівняння ручного стабілізуючого пристрою та автоматизованого (поза поміттю програмної реалізації розроблених алгоритмів)



Таблиця 1.3. Результати стабілізуючого пристрою задля стабілізатору знижуючого типу

Параметр	Ручний розрахунок	Програмний розрахунок	Відхилення, %
Коефіцієнт згладжування фільтра	997,8	993,52	-0,43
Індуктивність дроселю, мГн	0,185	0,2	7,5
Середнє значення струма через дросель, А	2,87	3,0	-4,3
Амплітуда колекторного струма, А	4,05	4,0	-1,25
Діаچه значення колекторного струма, А	1,28	1,29	0,78
Середній потік діоду, А	2,4	2,44	1,64
Змінна складова струма дроселю, А	0,97	1,0	3,0
Напруження колектор-емітер, У	26,1	27,0	3,3
Сила розсіяння на транзисторі, Вт	2,81	2,92	3,77
Сила розсіяння на діоді, Вт	1,63	1,59	-2,5
Зворотна напруження діоду, У	27,3	27,0	-1,1
Вхідний потік стабілізуючого пристрою, А	0,73	0,72	-1,4
ККД	0,75	0,77	2,6

В таблиці 1.3 представлені результати порівняння ручного та автоматизованого розрахунків. Слід зазначити, що при ручному стабілізуючого пристрою потрібно деякі допоміжні коефіцієнти визначати по графіках та разом з цим необхідно будувати характеристику навантаження. Ручний розрахунок займає близько години, тоді як при стабілізуючого пристрою поза поміччю програми потрібно менше хвилини задля того, щоб ввести вхідні дані та вибрати схему, отримати і роздрукувати результат. Як видно із таблиці 1.3, максимальне відхилення склало 7,5% задля Індуктивність дроселю. Задля приблизних розрахунків це допустиме значення. По решті властивостей погрішність була не більш 5%.

### 1.8.2 Тестування в застосунку в режимі стабілізуючого пристрою стабілізуючого пристрою підвищувального типу

На рис. 1.9 наведено робоче вікно застосунку під період стабілізуючого пристрою властивостей схеми ключового стабілізуючого пристрою напруження підвищувального типу. Задля порівняння ручного стабілізуючого пристрою та автоматизованого (поза поміччю програмної реалізації розроблених алгоритмів) задля ключового стабілізуючого пристрою підвищувального типу був проведений описаний нижче тест.



Таблиця 1.4. Результати стабілізуючого пристрою  
задня стабілізатору підвищувального типу

Параметр	Ручний розрахунок	Програмний розрахунок	Відхилення, %
Ємність фільтра, мкФ	17524	17600	0,43
Індуктивність дроселю, мГн	0,057	0,06	5
Середнє значення струма через дросель, А	48,1	47,52	-1,22
Амплітуда колекторного струма, А	55,7	55,85	0,27
Діаچه значення колекторного струма, А	15,4	15,63	1,47
Середній потік діоду, А	2,17	2,0	-8,5
Змінна складова струма дроселю, А	8,5	8,33	-2,04
Напруження колектор-емітер, У	100,76	100,0	-0,76
Сила розсіяння на транзисторі, Вт	372,4	370,02	-0,64
Сила розсіяння на діоді, Вт	0,2	0,19	-5,26
Зворотна напруження діоду, У	103,5	100,0	3,5
Вхідний потік стабілізуючого пристрою, А	47,9	47,52	-0,8
ККД	0,34	0,35	2,86

В таблиці 1.4 представлені результати порівняння ручного та автоматизованого розрахунків. Слід зазначити, що при ручному стабілізуючого пристрою потрібно деякі допоміжні коефіцієнти визначати по графіках та разом з цим необхідно будувати характеристику навантаження. Ручний розрахунок займає близько години, тоді як при стабілізуючого пристрою поза поміччю програми потрібно менше хвилини задля того, щоб ввести вхідні дані та вибрати схему, отримати і роздрукувати результат. Як видно із таблиці 1.4, максимальне відхилення склало біля 8,5% задля середнього струма діоду. Задля приблизних розрахунків це допустиме значення. По решті властивостей погрішність була біля 5%.

### 1.8.3 Тестування в застосунку в режимі стабілізуючого пристрою стабілізуючого пристрою полярно-інвертуючого типу

На рис. 1.10 наведено робоче вікно застосунку під період стабілізуючого пристрою властивостей схеми ключового стабілізуючого пристрою напруження полярно-інвертуючого типу. Задля порівняння ручного стабілізуючого пристрою та автоматизованого (поза поміччю програмної реалізації розроблених алгоритмів) задля ключового стабілізуючого пристрою полярно-інвертуючого типу був проведений описаний нижче тест.

Використовувані вхідні дані:

- робоча частота 10 кГц;
- напруження пульсацій 0,01 В;
- мінімальний потік навантаження 1 А;
- номінальний потік навантаження 2 А;
- коефіцієнт пульсацій вхідної напруження 0,1.

Результати розрахунку

Емність вхідного конденсатору, мкФ	49
Емність фільтру, мкФ	5882
Індуктивність дроселя, мГн	0.12
Середнє значення струму через дросель, А	3.93
Амплітуда колекторного струму, А	5.34
Діюче значення колекторного струму, А	1.54
Середній струм діоду, А	2.00
Змінні складова струму дроселя, А	1.42
Напруга колектор-емітер, В	17.00
Потужність розсіювання на транзисторі, Вт	2.73
Потужність розсіювання на діоді, Вт	1.13
Зворотня напруга діоду, В	17.00
Вхідний струм стабілізатору, А	1.16
ККД	0.72

Заклад освіти: ОТФК ОНТУ  
Розробник: Нізяєв Іван  
Дата: 23.03.2024

Рисунок 1.10. Візуальний інтерфейс застосунку задля стабілізуючого пристрою схем ключевих стабілізуючих пристроїв полярно-інвертуючого типу

В таблиці 1.5 представлені результати порівняння ручного та автоматизованого розрахунків.

Таблиця 1.5. Результати стабілізуючого пристрою задля стабілізатору полярно-інвертуючого типу

Параметр	Ручний розрахунок	Програмний розрахунок	Відхилення, %
Ємність вхідного конденсатору, мкФ	49,6	49	-1,22
Ємність фільтра, мкФ	5856	5852	0,07
Індуктивність дроселю, мГн	0,11	0,12	-8,3
Середнє значення струма через дросель, А	3,97	3,93	-1,02
Амплітуда колекторного струма, А	5,21	5,34	0,38
Діяче значення колекторного струма, А	1,57	1,54	-2,43
Середній потік діоду, А	2,1	2,0	-5
Змінна складова струма дроселю, А	1,44	1,42	-1,4
Напруження колектор-емітер, У	16,7	17,0	1,76
Сила розсіювання на транзисторі, Вт	2,71	2,73	-0,73
Сила розсіювання на діоді, Вт	1,1	1,13	2,65
Зворотна напруження діоду, У	17,7	17,0	-4,12
Вхідний потік стабілізуючого пристрою, А	1,13	1,16	2,59
ККД	0,7	0,72	2,78

Слід зазначити, що при ручному стабілізуючого пристрою потрібно деякі допоміжні коефіцієнти визначати по графіках та разом з цим необхідно будувати характеристику навантаження. Ручний розрахунок займає близько години, тоді як при стабілізуючого пристрою поза поміччю програми потрібно менше хвилини задля того, щоб ввести вхідні дані та вибрати схему, отримати і роздрукувати результат. Як видно із таблиці 1.5, максимальне відхилення склало біля 8% задля індуктивності дроселю. Задля приблизних розрахунків це допустиме значення. По решті властивостей погрішність була не більш 5%.

Очевидно, що особливо при стабілізуючого пристрою декількох схем та виборі оптимальної, розроблений програмний застосунок буде значно економити робочий період розробника електронної техніки та виключати появу помилок, можливих при ручному стабілізуючого пристрою.

## 2 ЕКОНОМІЧНИЙ РОЗДІЛ

### 2.1 Резюме

У даному дипломному проекті розроблено додаток задля стабілізуючого пристрою стабілізуючих пристроїв напруження живлення. Створено прикладне програмне забезпечення задля ПК із встановленою ОС сімейства MS Windows, яке дає поза початковими даними отримувати необхідні параметри деталей типових схем джерел вторичного електро-живлення. Програма створена на мові програмування C++ в інтегрованому середовищі візуальної розроблення Embarcadero RAD Studio. Ефективність кожного програмного продукту обчислюється його якістю і ефективністю процесу розроблення. Якість ПП обчислюється наступними складовими: із точки зору користувача; із позиції використання ресурсів; здійснення вимог до програмного забезпечення.

Оцінка якості програмного продукту із точки зору користувача обчислюється необхідним на стадії функціонування розміром оперативної пам'яті ЕОТ, витратами машинного періоду, пропускнуою спроможністю каналів передачі даних. Оцінка якості програмного продукту включає визначення трудомісткості та вартості його реалізації.

### 2.2 Визначення трудомісткості розроблення програмного забезпечення

Тривалість розроблення програмного продукту залежить з його обсягу, трудомісткості розроблення, кваліфікації виконавців, а разом з цим планових термінів, визначених умовами ринку. Методом структурної аналогії по відповідних каталогах аналогів програмного забезпечення обчислюється обсяг програмних засобів, в тисячах умовних машинних команд програми аналога

В таблиці 2.1 представлені аналоги програмного забезпечення, процедури котрих, в більшому чи меншому ступені, виконує розроблений програмний продукт. Задля нашого варіанта виділено сірим кольором.

					<b>КС 57. 14 000. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		53

Таблиця 2.1. Каталог аналогів

Найменування ПЗ	Обсяг процедури ПП = $V_0$ ум. машинних команд
1. ПП СУБД	1300 – 8600
2. ПП введення інформації	1800 – 8800
3. ПП оптимізаційних розрахунків	13000 – 10200

Вибравши аналог ПП, що містить  $V_0$  у умовних машинних командах, трудомісткості визначати на основі табл.2.2

Таблиця.2.2

Обсяг ПП, тис.умов.машинних команд	Норма періоду, люд/год
1.00	229
2.00	244
3.00	262

На підставі отриманого значення, по довіднику, обчислюється укрупнена норма періоду на розробку аналога програмного забезпечення (коректується поправочним коефіцієнтом враховуючої умови розроблення ПП, тобто у умовах комп'ютера,  $K_k=0,7\div 0,8$ ):  $T_{ар} = 244 \times 0,8 = 195.20$  (люд/годин).

Трудомісткість програмного продукту обчислюється по кожному етапу розроблення окремо на підставі трудомісткості аналога із урахуванням складності

розроблення, ступеня новизни та ступеня використання у розробці стандартних модулів на підставі формул:

$$T_{T_1} = T^a p \times L_1 \times K_H \quad (2.1)$$

$$T_{T_2} = T^a p \times L_2 \times K_H \quad (2.2)$$

$$T_{T_3} = T^a p \times L_3 \times K_H \times K_T \quad (2.3)$$

Задля стабілізуючого пристрою необхідні наступні коефіцієнти:

$L_i$  – питома вага  $i$ -го етапу розроблення (див. табл. 2.2.);

$K_H$  – поправочний коефіцієнт, що враховує ступінь новизни (див. табл. 2.3.);

$K_T$  – поправочний коефіцієнт, що враховує ступінь використання у розробці типових програм (див. табл. 2.4.).

Таблиця 2.2 Значення питомих коефіцієнтів трудомісткості стадії у загальній трудомісткості розроблення ПП

Код стадії	Ступінь новизни		
	А	Б	У
ТЗ (L <sub>1</sub> )	0,15	0,12	0,12
ТП (L <sub>2</sub> )	0,16	0,15	0,11
РП (L <sub>3</sub> )	0,55	0,58	0,61

Задля нашого варіанта виділено сірим кольором.

Таблиця 2.3. Значення поправочного коефіцієнта, що враховує ступінь новизни

Код ступеня новизни	Ступінь новизни	Значення K <sub>н</sub>
А	Принципово нові ПЗ	1,75 – 1,2
Б	ПЗ – розвиток визначеного параметричного ряду	1,0 – 0,8
У	ПЗ, що має аналог	0,7

Задля нашого варіанта виділено сірим кольором.

Таблиця 2.4 Значення коефіцієнта ступеня використання у розробці типових програм

Ступінь охоплення реалізованих функцій розроблювального ПО типовими програмами, %	Значення K <sub>т</sub>
60 та вище	0,6
40-60	0,7
20-40	0,8
До 20	0,9

Задля нашого варіанта виділено сірим кольором.

Тепер розраховуємо трудомісткість по кожному етапу окремо:

Трудомісткість технічного завдання

$$T_{ТЗ} = I * L_1 * K_n = 195,2 * 0,12 * 0,7 = 16,40 \text{ (люд/годин)}$$

Трудомісткість розроблення технічного проекту

$$T_{ТП} = I * L_2 * K_n = 195,2 * 0,11 * 0,7 = 15,04 \text{ (люд/годин)}$$

Трудомісткість розроблення робочого проекту

$$T_{РП} = I * L_3 * K_n * K_t = 195,2 * 0,61 * 0,7 * 0,7 = 58,35 \text{ (люд/годин)}$$

Задля подальших розрахунків визначили кількість папера, витраченого на кожен етап: технічне завдання N<sub>ТЗ</sub>=2 (стр), розробка ТП N<sub>ТП</sub>=15(стр), розробка

робочого проекту  $N_{\text{рп}}=25$  (стр), пояснювальна записка відповідно  $N_{\text{пз}}=50$  (стр)

Розрахунок зведений в таблицю 2.5

Таблиця 2.5 Розрахунок трудомісткості ПП

Найменування етапів	Розрахунок, годин.		
	2	3	4
1. ТЗ	$T_{\text{ТЗ}}=16,40$	$T_{\text{кк}}=0,7 \cdot N_{\text{ТЗ}}=0,7 \cdot 2=1,4$	$T_{\text{нк}}=0,15 \cdot N_{\text{ТЗ}}=0,15 \cdot 2=0,30$
2. Розробка ТП	$T_{\text{ТП}}=15,04$	$T_{\text{кк}}=0,7 \cdot N_{\text{ТП}}=0,7 \cdot 15=10,50$	$T_{\text{нк}}=0,15 \cdot N_{\text{ТП}}=0,15 \cdot 15=2,25$
3. Розробка РП	$T_{\text{РП}}=58,35$	$T_{\text{кк}}=0,7 \cdot N_{\text{РП}}=0,7 \cdot 25=17,5$	$T_{\text{нк}}=0,15 \cdot N_{\text{РП}}=0,15 \cdot 25=3,75$
4. Розробка ПЗ	$T_{\text{ПЗ}}=1,5 \cdot N_{\text{ПЗ}}=1,5 \cdot 50=75$	$T_{\text{кк}}=0,7 \cdot N_{\text{ПЗ}}=0,7 \cdot 50=35$	$T_{\text{нк}}=0,15 \cdot N_{\text{ПЗ}}=0,15 \cdot 50=7,5$
Усього, у т.ч.:	230,2		
- на розробку	$\Sigma T_{\text{р}}=152$		
- контроль керівника		$\Sigma T_{\text{кк}}=64,4$	
- нормоконтроль			$\Sigma T_{\text{нк}}=13,8$

### 2.3 Розрахунок ціни програмного продукту

В цьому розділі задля визначення ціни розраховуємо основну заробітну плату виконавців, матеріальні витрати, вартість машино – години та витрати на розробку ПЗ. Розрахунок основної заробітної плати виконавців приведений в таблиці 2.7. Відповідно до статті 8 «Закону про Державний бюджет України на 2024» встановлено мінімальну заробітну плату в місячному розмірі із 1 січня 2024 року - 6500 гривень; мінімальну погодинну тарифну ставку – 39.26 грн.

Таблиця 2.6 Розрахунок основної заробітної плати виконавців

Найменування робіт	Трудомісткість робіт, години	Погодинна тарифна ставка, грн.	Розрахунок, грн.
1. Розробка ПП	152	39.26	5958,15
2. Контроль керівника	65	38,50	2502,50
3. Нормоконт-роль	14	38,50	539,00
Усього	-	-	$\Sigma Z_0=8999,15$

Таблиця 2.7 Розрахунок матеріальних витрат на розробку ПЗ

Найменування матеріальних витрат	Тип, модель	Кількість	Ціна одиниці, грн.	Вартість, грн.
Папір	Лист А4	60	2.60	160,0
Разом	-	-	-	$V_{мга}=160,0$
Транспортно-заготівельні витрати (10%)				$V_{трз} = 0,1 \times V_{м1} = 0,1 \times 160,0 = 16,0$
Усього				$V_{м} = V_{м1} + V_{трз} = 176,0$

На підставі отриманих даних по окремих статтях витрат складена калькуляція планової собівартості у цілому ПП поза формою, приведеною у таблиці 2.8.

Таблиця 2.8 Розрахунок статей витрат планової собівартості

Стаття витрат	Значення, грн.	Формула стабілізуючого пристрою
1. Матеріали	176,0	$V_{м}$ (див. табл. 2.7)
2. Основна заробітна плата	8999,15	$Z_o$ (див. табл. 2.6)
3. Додаткова заробітна плата	1349,87	$Z_d = 0,15 \times Z_o = 8999,15 \times 0,15$
4. Відрахування до єдиного фонду соціального внеску	2276,78	$V_{е.с.в.} = 0,22 \times (Z_o + Z_d) = 0,22 \times (8999,15 + 1349,87)$
5. Накладні витрати	2699,75	$V_{нак.} = 0,3 \times Z_o = 0,3 \times 8999,15$
6. Повна собівартість	15501,55	$C_{пов} = V_{м} + Z_o + Z_d + V_{е.с.в.} + V_{нак.} = 176,0 + 8999,15 + 1349,87 + 2276,78 + 2699,75$

Розмір прибутку, що включається у ціну, визначаємо по наступній формулі:

$$\Pi = (C_{п} * P) / 100 = (15501,55 * 10) / 100 = 1550,15 \text{ грн} \quad (2.4)$$

Де  $p$  – плановий рівень рентабельності (10-15%).

Оптова ціна (кошторисна вартість) обчислюється по формулі:

$$C_o = C_{п} + \Pi = 15501,55 + 1550,15 = 17051,70 \text{ грн} \quad (2.5)$$

Податок на додану вартість визначаємо по наступній формулі:

$$\text{ПДВ} = 0,2 * C_o = 17051,70 * 0,2 = 3410,34 \text{ грн}; \quad (2.6)$$

Виходячи із отриманих даних, ціна реалізації розробленого програмного продукту на основі наступної формули, становитиме:

$$C_p = C_o + \text{ПДВ} = 17051,70 + 3410,34 = 20462,04 \text{ грн} \quad (2.7)$$

					<b>КС 57. 14 000. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		57

## **3 РОЗДІЛ ОХОРОНИ ПРАЦІ І ТЕХНІКИ БЕЗПЕКИ**

### **3.1 Вступ**

Охорона праці, як соціальний чинник, відіграє на підприємстві важливу, роль оскільки, якими б важливими не були трудові здобутки, вони не можуть компенсувати людині втраченого здоров'я, а тим більш життя. Те та інше дається тільки один раз. Необхідно пам'ятати, що внаслідок нещасних випадків і аварій гинуть на виробництві не просто робітники і службовці, на підготовку котрих держава вкладає значні кошти, а перш поза все люди – годувальники сімей, батьки і матері дітей. Незадовільний стан охорони праці відображається на економіці держави. Служба охорони праці створюється на підприємствах незалежно з форми власності і видів діяльності задля здійснення правових, організаційно-технічних, санітарно-гігієнічних, соціально-економічних та лікувально-профілактичних заходів, спрямованих на запобігання нещасних випадків, професійних захворювань та аваріям у процесі праці. Основна мета всіх цих заходів – створити на підприємстві безпечні і здорові умови праці.

В розділі охорона праці дипломного проекту наведені характеристики приміщень, де експлуатуються ВДТ. До розгляду взято робоче місце програміста (оператора ЕОМ).

### **3.2 Аналіз небезпечних і шкідливих чинників, що впливають на працівника**

Оператори ПК та програмісти зіштовхуються із впливом таких фізично небезпечних та шкідливих виробничих факторів, як підвищений рівень шуму, підвищена температура зовнішнього середовища, недостатня освітленість робочої зони, електричний потік і інші. Через це на робочому місці програміста повинні існувати створені умови задля високопродуктивної праці.

Перетворення та обробка інформації проводиться поза поміччю ПК. Робота спроможне кваліфікуватися як робота оператором ЕОМ.

					<b>КС 57. 14 000. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		58

### 3.3 Розробка заходів із охорони праці

#### 3.3.1 Виробничі приміщення

При плануванні виробничого приміщення врахована санітарна характеристика виробничих процесів, дотримуються норми корисної площі задля працюючих, а разом з цим нормативи площ задля розташування устаткування, що забезпечують безпечну роботу і зручне обслуговування устаткування.

Об'ємно-планувальні рішення будівель і приміщень задля роботи із ВДТ містять відповідати вимогам ДСанПіН 3.3.2.007-98. Розміщення робочих місць із ВДТ ЕОМ та ПЕОМ в підвальних приміщеннях, на цокольних поверхах заборонено. Площа на одне робоче місце становить не менше ніж 6,0 м<sup>2</sup>, а об'єм – не менше ніж 20,0м<sup>3</sup>.

Виробничі приміщення повинні обладнуватися шафами задля зберігання документів, полицями, стелажми, тумбами тощо, із урахуванням вимог до площі приміщення.

В приміщеннях із ВДТ слід щоденно робити вологе прибирання. Приміщення повинні існувати оснащені аптечками першої медичної допомоги.

#### 3.3.2 Мікроклімат робочої зони працівників, вентиляція

В виробничих приміщеннях на робочих місцях із ВДТ містять забезпечуватись оптимальні значення властивостей мікроклімату: температури, відносної вологості та рухливості повітря ( ДСанПіН 3.3.2.007-98).

Таблиця 3.1. Норми мікроклімату задля приміщень із ВДТ ЕОМ і ПЕМ

Пора року	Категорія робіт	Температура повітря, С, не більш	Відносна вологість повітря %	Швидкість руху повітря, м/с
Холодна	Легка-1а	22-24	40-60	0,1
	Легка-1б	21-23	40-60	0,1
Тепла	Легка-1а	23-25	40-60	0,1
	Легка-1б	22-24	40-60	0,1

Рівні позитивних та негативних іонів в повітрі приміщень із ВДТ містять відповідати санітарно-гігієнічним нормам № 2152-80.

Таблиця 3.2. Рівні позитивних та негативних іонів в повітрі

Рівні	Число іонів у 1 см <sup>3</sup> повітря	Число іонів у 1 см <sup>3</sup> повітря
	n+	n-
Мінімально необхідні	400	600
Оптимальні	1500-3000	3000-5000
Максимально допустимі	50000	50000

### 3.3.3 Освітлення робочого місця, шум, вібрація

Штучне освітлення у приміщеннях із робочими місцями, обладнаними ВДТ містить здійснюватись системою загального рівномірного освітлення. В виробничих і адміністративних приміщеннях, в разі переважної роботи із документами, допускається застосування системи комбінованого освітлення – крім системи загального освітлення додатково встановлюються світильники місцевого освітлення.

Значення освітленості на поверхні робочого столу у зоні розміщення документів містить становити 300-500лк.

Як джерела світла задля штучного освітлення містять застосовуватись переважно люмінесцентні лампи типу ЛД. Допускається застосування ламп розжарювання в світильниках місцевого освітлення.

### 3.3.4 Організація робочого місця користувача ПК

Робочі місця слід так розташовувати відносно світових прорізів, щоб природне світло падало збоку, переважно зліва. При розміщенні робочих столів із ВДТ слід дотримуватися таких відстаней: між бічними поверхнями ВДТ -1,2м; з тильної поверхні одного ВДТ до екрану іншого – 2,5м.

Екран ВДТ містить розташовуватися на оптимальній відстані з очей користувача, що становить 600...700 мм, проте не ближче ніж поза 600 мм із урахуванням розміру літерно-цифрових знаків та символів.

Клавіатуру розташовують на поверхні столу на відстані 100...300 мм з краю, зверненого до працюючого. В конструкції клавіатури містить

передбачатися опорний пристрій, який дає змогу змінювати кут нахилу поверхні клавіатури в межах 5... 15°.

При оснащенні робочого місця лазерним принтером параметри лазерного випромінювання повинні відповідати вимогам СанПіН № 5804-91.

ЕОМ ВДТ та ПК , інше устаткування , електропроводи і кабелі поза виконанням та ступенем захисту містять відповідати класу зони поза НПАОП 40.1-1.01-97, мати апаратуру захисту з струма короткого замикання і інших аварійних режимів. В приміщеннях, де одночасно експлуатується понад п'ять ЕОМ встановлюється аварійний резервний вимикач, який спроможне повністю вимкнути електричне живлення приміщення, крім освітлення. Не допускається підключати ЕОМ із ВДТ та ПК до звичайної двопровідної електромережі, у через це числі – із використанням перехідних пристроїв

### **3.3.5 Електробезпека**

Це система організаційних та технічних заходів і засобів, що забезпечують захист людей з шкідливої та небезпечної дії електричного струма, електричної дуги, електричного поля та статичної електрики.

Основні технічні засоби та заходи забезпечення електробезпеки при нормальному режимі роботи електроустановок включають:

- ізоляцію струмовідних частин;
- недоступність струмовідних частин;
- здійснення електроустановок, ізольованих з землі;
- захисне розділення електричних мереж;
- компенсацію ємнісних струмів замикання на землю;
- вирівнювання потенціалів.

Із метою підвищення значення безпеки, залежно з призначення, умов експлуатації та конструкції, у електроустановках застосовується одночасно більшість із перерахованих технічних засобів та заходів.

Особа відповідальна поза електрогосподарство призначається із числа працівників, котрі містять не нижче IV групи із електробезпеки і відповідний стаж роботи задля обслуговування електроустановок несе персональну

					<b>КС 57. 14 000. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		61

відповідальність поза допущення працівника застосовувати у роботі електричну енергію

### 3.4 Пожежна безпека

Пожежна небезпека – це можливість виникнення і розвитку пожежі у будь-якій речовині, процесі, стані.. Коли людина перебуває у зоні впливу пожежі, то вона спроможне потрапити під дію наступних небезпечних і шкідливих факторів: токсичні продукти згорання, вогонь, підвищена температура середовища, дим, недостатність кисню, руйнування будівельних конструкцій, вибухи, паніка.

Усі працівники повинні вміти користуватись наявними вогнегасниками, іншими первинними засобами пожежогасіння, знати місце їх знаходження.

До первинних засобів пожежогасіння відносяться:

- вогнегасники;
- пожежний інвентар (покривала із негорючого теплоізоляційного полотна, грубововняної тканини чи повсті;
- ящики із піском;
- бочки із водою, пожежні відра, совкові лопати) і пожежний інструмент (гаки, ломи, сокири тощо).

Пожежні щити (стенди) встановлюються на території об'єкта із стабілізуючого пристрою один щит (стенд) на площу 5000 м<sup>2</sup>.

Вибір типу і визначення необхідної кількості вогнегасників здійснюється відповідно до Типових норм належності вогнегасників, затверджених наказом Міністерства України із питань надзвичайних ситуацій і в справах захисту населення з наслідків Чорнобильської катастрофи з 02.04.2004 № 151 і зареєстрованих в Міністерстві юстиції України 29.04.2004 поза № 554/9153. Евакуаційні шляхи та виходи повинні втримуватися відповідно до НАПБ А.01.001 2004, існувати вільними, нічим не захаращуватися та в разі виникнення пожежі забезпечувати безпеку під період евакуації всіх людей, котрі перебувають в приміщеннях будівель і споруд.

					<b>КС 57. 14 000. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		62

## ВИСНОВКИ

Протягом дипломного розроблення виконано аналіз методики стабілізуючого пристрою схем електронних стабілізуючих пристроїв, що входять до складу блоків живлення сучасних електронних пристроїв, зокрема комп'ютерної техніки, а разом з цим розроблено алгоритми та здійснено їх програмну реалізацію. На основі побудованих блок-схем алгоритмів створено прикладний програмний застосунок під ОС сімейства MS Windows, який дає поза початковими даними розраховувати необхідні параметри деталей типових схем джерел вторичного електро-живлення, а саме – стабілізуючих пристроїв напруження. Програмний застосунок створено в інтегрованому середовищі візуальної розроблення Embarcadero RAD Studio мовою програмування C++.

В роботі докладно розглянуто найбільш популярні в розробників електронної апаратури види стабілізуючих пристроїв напруження та представлені спрощені формули їх стабілізуючого пристрою. Визначено, що задля ключевих стабілізуючих пристроїв напруження основними розраховуваними параметрами є параметри і режими роботи транзисторів, діодів, стабілітронів, конденсаторів, дроселів. Створені методики стабілізуючого пристрою дозволили здійснити їх подальшу програмну реалізацію. У окремих випадках використовувалися відповідні наближені формули.

Наведені в роботі фрагменти лістингу програми мовою C++ супроводжені докладними коментарями, що дозволить в майбутньому модифікувати та удосконалити застосунок. Виконано тестові випробування, котрі показали, що розрахунки, проведені поза поміччю розробленого програмного забезпечення містять достатню задля інженерної практики ступінь точності. Максимальне відхилення склало 10%. Задля приблизних розрахунків це припустиме значення. По решті властивостей погрішність у середньому не перевищила 5%. З урахуванням того, що аналогічний розрахунок класичним методом (уручну) спроможне віднімати дуже багато періоду, доцільність реалізації такого програмного застосунку є очевидною, особливо, коли необхідно виконати пошук найкращого варіанту схеми побудови блоку живлення електронної апаратури.

					<b>КС 57. 14 000. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		63

## ПЕРЕЛІК ВИКОРИСТАНИХ ІНФОРМАЦІЙНИХ ДЖЕРЕЛ

1. Гусев Б.С. Комп'ютерна схемотехніка [навчальний посібник] // – К.: НУБіП України, 2022. – 264с.
2. Матвієнко М.П., Розен В. П. Комп'ютерна схемотехніка: навчальний посібник. – К.: Видавництво Ліра-К, 2020. – 192 с.
3. Дейбук В.Г. Віртуальний електронний практикум: Навчальний посібник – Чернівці: Чернівецький нац. ун-т, 2021. – 188 с.
4. Трофименко О.Г. С++. Алгоритмізація та програмування: підручник / О.Г. Трофименко, Ю.В. Прокоп, Н.І. Логінова, О.В. Задерейко. 2-ге вид. перероб. і доповн. – Одеса : Фенікс, 2019. – 477 с.
5. Азаров О. Д., Гарнага В. А., Клятченко Я. М., Тарасенко В. П. Комп'ютерна схемотехніка: підручник. – Вінниця: ВНТУ, 2018. – 230 с.
6. Архангельский А.Я. Программирование в С++Builder, 7-е изд. – М.: ООО Бином-Пресс, 2010 г. – 1230с., ил.
7. Нікулін В.С. Перетворювальні пристрої, ведені мережею: Конспект лекцій. –Харків: УкрДАЗТ, 2008. – Ч.4. – 85 с.
8. Мосіюк О.О., Федорчук А.Л. Операційні системи та системне програмування: навчально-методичний посібник. Житомир: Вид-во ЖДУ ім. Івана Франка, 2022. – 76 с.
9. Stroustrup B. A Tour of C++ (Second Edition). – Addison-Wesley, 2018. – 240 p. – ISBN 978-0-13-499783-4.
10. Каганюк О.К. Комп'ютерна схемотехніка: Навчальний посібник. – Луцьк: РРВ Луцького НТУ, 2016. – 236 с.
11. Бібліотека MSDN [Електронний ресурс]. – Режим доступу: URL <http://msdn.microsoft.com/ru-ru/library/default.aspx>.

					КС 57. 14 000. 00 ДП ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		64

## ДОДАТОК А. Код функції btnStabilizerClick мовою C++

```
void __fastcall TApp::btnStabilizerClick(TObject *Sender)
{
    // Початкові дані
    double E2, E1, UP, YM, Y0, A;
    int F;

    E2 = editS_E2->Text.ToDouble(); // вхідна напруга, В
    E1 = editS_E1->Text.ToDouble(); // вихідна напруга, В
    F = editS_F->Text.ToInt();      // робоча частота, кГц
    UP = editS_UP->Text.ToDouble(); // напруга пульсацій, В
    YM = editS_YM->Text.ToDouble();

    // мінімальний струм навантаження, А
    Y0 = editS_Y0->Text.ToDouble();
    // номінальний струм навантаження, А
    A = editS_A->Text.ToDouble(); // коеф. пульсацій вхідного U

    // Перевірка коректності
    if( !(E2 > 0 && E1 > 0 && F > 0 && UP > 0
        && YM > 0 && Y0 > 0 && A > 0) ) {
        ErrMess(
            "Вхідні дані повинні бути \тпозитивними числами!");
        return;
    }

    //----- Розрахунок вихідних параметрів -----
    double D // індуктивність дроселя
        Y1 // середнє значення струму через дросель
        Y2 // амплітуда колекторного струму
        Y3 // значення діючого колекторного струму
        Y4 // середній струм діода
        Y5 // змінна складова струму дроселя
        U2 // напруга колектор-емітер
        U0 // зворотна напруга діода, В
        PT // потужність розсіювання на транзисторі
        PD // потужність розсіювання на діоді
        YB // вхідний струм стабілізатора
        C1 // ємкість вхідного конденсатора фільтру
        C2 // ємкість фільтру
        R; // ККД
    double Q // коеф. згладжування фільтру
        C[3] // ємкість фільтру
        U[3]; // амплітуда напруги пульсацій
    // id - тип стабілізатора
    int id = comboStabilizer->ItemIndex;

    if( id == 0) // знижуючий СН
    {
        if( E2 <= E1 ){
```

```

    ErrMess(
        "Вихідна напруга має бути менше вхідної");
    return;
}
D = E1*(E2-E1)/(2*F*YM*E2);
Y1 = Y0;

// коеф. згладжування фільтру
Q = (2.2*E1 - 2.5*E1*(E1/E2-.1))/UP;

for( int i = 0; i < 3; i++)
{
    double N = i + 1;
    // добуток L*C фільтру
    int X = 1000*N*N*(pow(Q,1/N) + 2)/(40*F*F);

    // ємність фільтру, мкФ
    C[i] = X/(D*N*N);

    // напруга пульсації, В
    U[i] = E1*(2.2 - 2.5*(E1/E2-.1)) / pow(Q,1/N);
}
Y5 = E1*(E2-E1)/(2*F*E2*D);
U2 = U0 = E2;
Y3 = Y0*sqrt(E1/E2);
Y4 = Y0*(E2 - E1) / E2;
PT = Y3*(.07*E2 + 2*E1/E2);
PD = .8*Y4*(E2 - E1) / E2;
}
else if( id == 1 ) // полярно-інвертуючий СН
{
    D = E1*E2*E2/(2*F*YM*pow((E1+E2),2));
    Y3 = Y0*sqrt(E1*(E1+E2)) / E2;
    Y4 = Y0;
    Y5 = E1*E2/(2*F*(E1+E2)*D);
    U2 = U0 = E1 + E2;
    C1 = 1000*Y0*E1/(F*E2*(E1+E2)*A);
    C2 = 1000*Y0*E1/(F*(E1+E2)*UP);
    PT = Y3*(.07*U2 + 2*E1/U2);
    PD = .8*Y0*E2/U2;
}
else // if( id == 2 ) // підвищувальний СН
{
    if( E2 >= E1 ){
        ErrMess(
            "Вихідна напруга має бути більше вхідної");
        return;
    }

    D = E2*E2*(E1-E2)/(2*F*YM*E1*E1);

```

```

Y3 = Y0*sqrt(E1*(E1-E2)) /E2;
Y4 = Y0;
Y5 = E2*(E1-E2)/(2*F*E1*D);
U2 = U0 = E1;
C2 = IE3*Y0*(E1-E2)/(F*E1*UP);
PT = Y3*E1*(.07*E2 + 2) /E2;
PD = .8*Y0*E2/E1;
}
YB = (E1*Y0 + PT + PD)/E2;
R = E1*Y0/E2/YB;

if( id == 1 ) // полярно-інвертуючий СН
Y1 = Y0*(E1+E2) /E2/R;
if( id == 2 ) // підвищувальний СН
Y1 = Y0*E1/E2/R;
Y2 = Y1 + Y5;

//----- Готуємо результати для виводу -----
// виводимо (якщо треба) інформацію про початкові дані
FillStabilizerInputText();

AnsiString t;
char buff[128];

if( id == 0 ) // Знижувальний КСН
{
t += "
t += " | К-ть ланок | Ємкість фільтру | Нанруга | | \r\n";
t += " | фільтру | мкФ | пульсацій, В | | \r\n";
t += " |-----+-----+-----| \r\n";
printf( buff " | 1 | %6.0f | %4.2f | | \r\n", C[0], U[0]);
t += buff;
printf( buff " | 2 | %6.0f | %4.2f | | \r\n", C[1], U[1]);
t += buff;
printf( buff " | 3 | %6.0f | %4.2f | | \r\n", C[2], U[2]);
t += buff;
t += " | | | | | \r\n\r\n";
printf(buff " %-45s%6.2f"
"Коефіцієнт згладжування фільтру", Q); t += buff;
}

if( id == 1 ) // полярно-інвертуючий
{
printf(buff, "\r\n %-45s%6.0f"
"Ємкість вхідного конденсатора, мкФ", C1); t += buff;
}

if( id == 1 || id == 2 )
{
printf(buff, "\r\n %-45s%6.0f"

```

```

    "Ємкість фільтру, мкФ", C2); t += buf;
}
    sprintf(buf, "\r\n %-45s%6.2f\r\n"
"Індуктивність дроселя, мГн", D); t += buf;
    sprintf(buf " %-45s%6.2f\r\n"
"Середнє значення струму через дросель, А, Y1); t += buf;
    sprintf(buf " %-45s%6.2f\r\n"
"Амплітуда колекторного струму, А, Y2); t += buf;
    sprintf(buf " %-45s%6.2f\r\n"
"Значення колекторного струму, що діє, А, Y3);
t += buf;
    sprintf(buf " %-45s%6.2f\r\n"
"Середній струм діода, А, Y4); t += buf;
    sprintf(buf " %-45s%6.2f\r\n"
"Змінна складова струму дроселя, А, Y5); t += buf;
    sprintf(buf " %-45s%6.2f\r\n"
"Напруга колектор-емітер, В", U2); t += buf;
    sprintf(buf " %-45s%6.2f\r\n"
"Потужність розсіювання на транзисторі, Вт", PT); t += buf;
    sprintf(buf " %-45s%6.2f\r\n"
"Потужність розсіювання на діоді, Вт", PD); t += buf;
    sprintf(buf " %-45s%6.2f\r\n"
"Зворотна напруга діода, В", U0); t += buf;
    sprintf(buf " %-45s%6.2f\r\n"
"Вхідний струм стабілізатора, А, YB); t += buf;
    sprintf(buf " %-45s%6.2f\r\n", "ККД", R); t += buf;

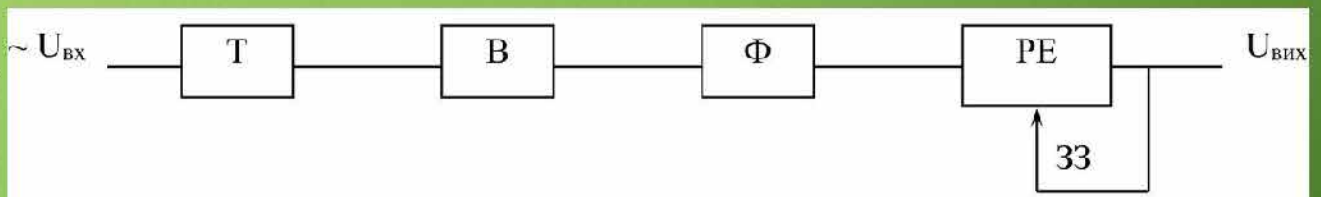
// strStabilizerResult – рядок, що містить рез-ти розрахунку
strStabilizerResult = t;
// Виведення даних на екран
// Якщо відмічена галочка "+ вхідні дані"
// тоді вони теж будуть виведені
checkInputStabilizerClick(0);
}

```

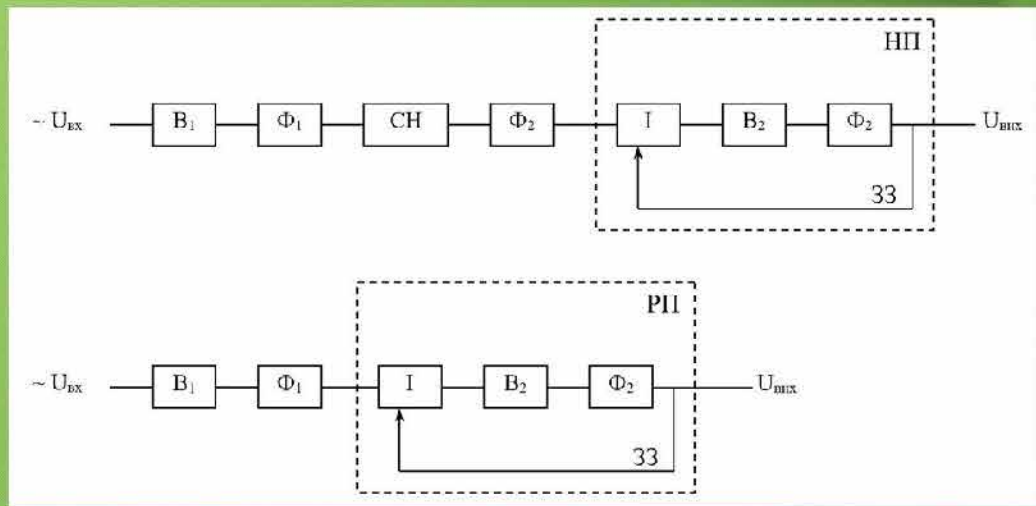
Розробка додатку для розрахунку  
стабілізаторів напруги живлення

Нізяєв Іван,  
гр. 4КС-57,  
ОТФК ОНТУ

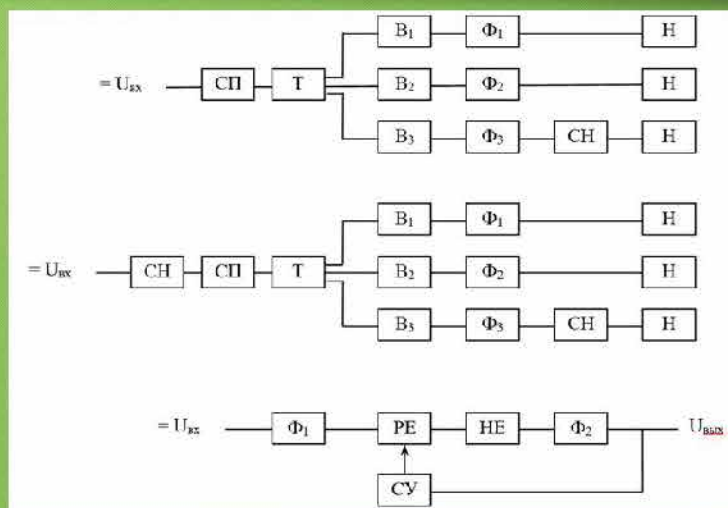
*Структурна схема джерела вторинного електроживлення  
з безперервною стабілізацією напруги*



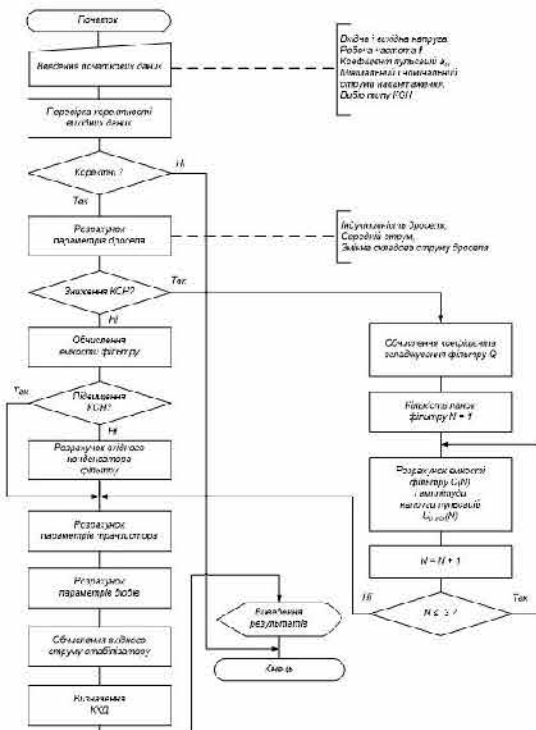
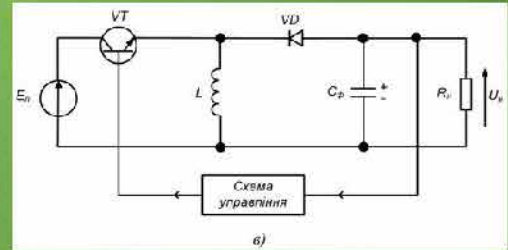
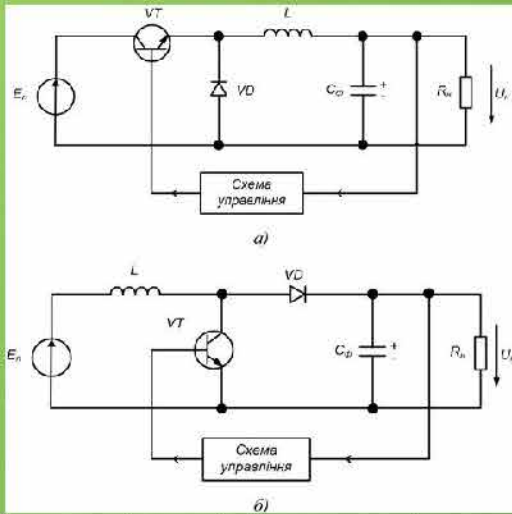
*Структурні схеми безтрансформаторних джерел вторинного електроживлення з нерегульованим перетворювачем і стабілізатором напруги і з регульованим перетворювачем*



*Структурні схеми джерел електроживлення на основі статичного перетворювача з внутрішньою стабілізацією, статичного перетворювача і стабілізатора напруги, ключового стабілізатора*

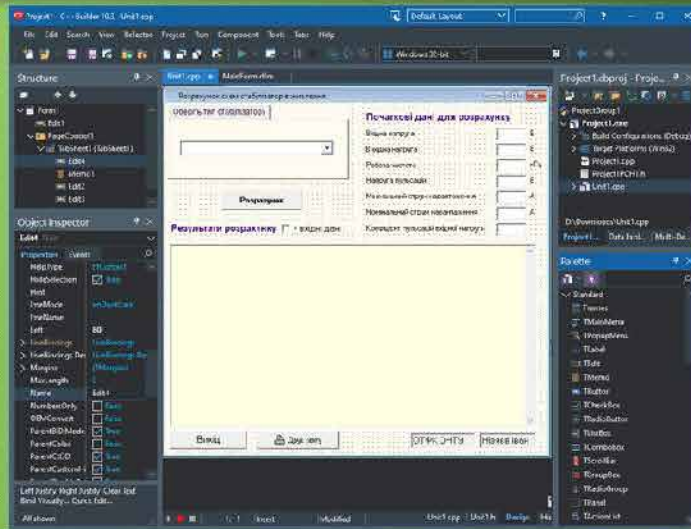


## Схеми ключових стабілізаторів напруги: знижувального (а), підвищувального (б) і інвертуючого (в)

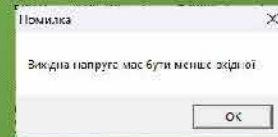
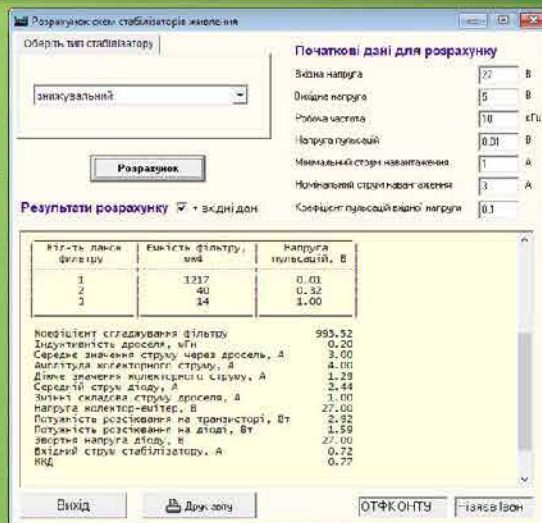


Алгоритм розрахунку  
ключових стабілізаторів  
напруги знижувального,  
підвищувального і полярно-  
інвертуючого типів

## Середовище розробки Embarcadero RAD Studio C++Builder



## Візуальний інтерфейс застосунку для розрахунку схем ключових стабілізаторів знижувального типу



## Порівняння результатів ручного та програмного розрахунків для стабілізатору знижувального типу

Параметр	Ручний розрахунок	Програмний розрахунок	Відхилення, %
Коефіцієнт згладжування фільтру	997,8	993,52	-0,43
Індуктивність дроселя, мГн	0,185	0,2	7,5
Середнє значення струму через дросель, А	2,87	3,0	4,3
Амплітуда колекторного струму, А	4,05	4,0	-1,25
Дічне значення колекторного струму, А	1,28	1,29	0,78
Середній струм діода, А	2,4	2,44	1,64
Змінна складова струму дроселя, А	0,97	1,0	3,0
Напруга колектор-емітер, В	26,1	27,0	3,3
Потужність розсіювання на транзисторі, Вт	2,81	2,92	3,77
Потужність розсіювання на діоді, Вт	1,63	1,59	-2,5
Зворотна напруга діода, В	27,3	27,0	-1,1
Вхідний струм стабілізатора, А	0,73	0,72	-1,4
ККД	0,75	0,77	2,6

## Візуальний інтерфейс застосунку для розрахунку схем ключових стабілізаторів підвищувального типу

Розрахунок схем стабілізаторів знизлення

Оберть тип стабілізатору:

Підвищувальний

Розрахунок

Результати розрахунку  вхідні дані

РЕЗУЛЬТАТИ РОЗРАХУНКУ

Ємність фільтру, мкФ	17600
Індуктивність дроселя, мГн	0,06
Середнє значення струму через дросель, А	47,52
Амплітуда колекторного струму, А	33,85
Дічне значення колекторного струму, А	13,63
Середній струм діода, А	2,00
Змінна складова струму дроселя, А	8,33
Напруга колектор-емітер, В	100,00
Потужність розсіювання на транзисторі, Вт	370,02
Потужність розсіювання на діоді, Вт	0,29
Зворотна напруга діода, В	100,00
Вхідний струм стабілізатора, А	47,52
ККД	0,35

Завод: світлі: ОТАК ОНТУ  
Розробник: Мізєв Іван  
Дата: 23.03.2024

Вихід Друк звіту ОТФК ОНТУ Мізєв Іван

Помилка

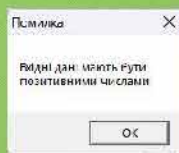
Вихідна напруга має бути більшою вхідної

OK

## Порівняння результатів ручного та програмного розрахунків для стабілізатору підвищувального типу

Параметр	Ручний розрахунок	Програмний розрахунок	Відхилення, %
Ємність фільтру, мкФ	17524	17600	0,43
Індуктивність дроселя, мГн	0,057	0,06	5
Середнє значення струму через дросель, А	48,1	47,52	1,22
Амплітуда колекторного струму, А	55,7	55,85	0,27
Дічне значення колекторного струму, А	15,4	15,63	1,47
Середній струм діода, А	2,17	2,0	8,5
Змінна складова струму дроселя, А	8,5	8,33	-2,04
Напруга колектор емітер, В	100,76	100,0	0,76
Потужність розсіювання на транзисторі, Вт	372,4	370,02	-0,64
Потужність розсіювання на діоді, Вт	0,2	0,19	-5,26
Зворотно напруга діода, В	103,5	100,0	3,5
Вхідний струм стабілізатора, А	47,9	47,52	-0,8
ККД	0,34	0,35	2,86

## Візуальний інтерфейс застосунку для розрахунку схем ключових стабілізаторів полярно-інвертуючого типу



Розрахунок схем стабілізаторів змінного

Оберіть тип стабілізатору

полярно-інвертуючий

**Розрахунок**

**Початкові дані для розрахунку**

Вхідна напруга: 12 В

Вихідна напруга: 5 В

Робоча частота: 0 кГц

Напруга пульсацій: 0,21 В

Мінімальний струм навантаження: 1 А

Мінімальний струм холостого ходу: 2 А

Коефіцієнт пульсацій вихідної напруги: 0,1

**Результати розрахунку**  вхідні дані

**РЕЗУЛЬТАТИ РОЗРАХУНКУ**

Ємність вхідного конденсатора, мкФ: 49

Ємність фільтру, мкФ: 3442

Індуктивність дроселя, мГн: 0,12

Середнє значення струму через дросель, А: 3,93

Амплітуда колекторного струму, А: 3,24

Дічне значення колекторного струму, А: 1,54

Середній струм діода, А: 2,00

Змінна складова струму дроселя, А: 1,42

Напруга колектор-емітер, В: 17,00

Потужність розсіювання на транзисторі, Вт: 2,75

Потужність розсіювання на діоді, Вт: 1,13

Зворотно напруга діода, В: 17,00

Вхідний струм стабілізатора, А: 1,16

ККД: 0,72

Заклад освіти: ІТФК ІНТУ

Розробник: Ніжеса Іван

Дата: 23.03.2024

Вихід Прикладі

ІТФК ІНТУ Ніжеса Іван

## Порівняння результатів ручного та програмного розрахунків для стабілізатору полярно-інвертуючого типу

Параметр	Ручний розрахунок	Програмний розрахунок	Відхилення, %
Смність вхідного конденсатору, мкФ	49,6	49	-1,22
Смність фільтру, мкФ	5856	5852	0,07
Індуктивність дроселя, мГн	0,11	0,12	-8,3
Середнє значення струму через дросель, А	3,97	3,93	-1,02
Амплітуда колекторного струму, А	5,21	5,34	0,38
Дієче значення колекторного струму, А	1,57	1,54	-2,43
Середній струм діода, А	2,1	2,0	-5
Змінна складова струму дроселя, А	1,44	1,42	-1,4
Напруга колектор емітер, В	16,7	17,0	1,76
Потужність розсіювання на транзисторі, Вт	2,71	2,73	0,73
Потужність розсіювання на діоді, Вт	1,1	1,13	2,65
Зворотна напруга діода, В	17,7	17,0	-4,12
Вхідний струм стабілізатора, А	1,13	1,16	2,59
ККД	0,7	0,72	2,78

## Висновки

Тестові випробування показали, що розрахунки, проведені за допомогою розробленого програмного застосунку мають достатню для інженерної практики ступінь точності. Максимальне відхилення склало 10%. Для приблизних розрахунків це допустиме значення. По решті параметрів середня погрішність не перевищила 5%. З урахуванням того, що аналогічний розрахунок класичним методом (уручну) може віднімати дуже багато часу, доцільність створення такого програмного застосунку є очевидною, особливо, якщо необхідно виконати пошук найкращого варіанту схеми побудови блоку живлення електронної апаратури.

**ВІДГУК**

керівника на дипломний проект здобувача (здобувачки) освіти  
відділення комп'ютерних систем

*Нізяєва Івана Романовича*

(прізвище, ім'я та по батькові)

Спеціальність: 123 «Комп'ютерна інженерія»

Освітня програма: «Обслуговування комп'ютерних систем і мереж»

Тема дипломного проекту: Розробка додатку для розрахунку стабілізаторів  
напруги живлення

**ХАРАКТЕРИСТИКА ДИПЛОМНОГО ПРОЕКТУ**

а) обсяг і якість виконання проекту (графічного матеріалу і розрахунково-пояснювальної записки) Дипломний проект виконано відповідно технічному завданню. Пояснювальна записка до дипломного проекту містить 75 сторінок. У пояснювальній записці описано етапи розробки програмного забезпечення для розрахунку схем ключових стабілізаторів напруги блоків живлення у ІСР Embarcadero RAD Studio C++Builder. Графічна частина складається з окремих слайдів, оформлених у вигляді презентації, передбачених технічним завданням. Якість виконання пояснювальної записки та слайдів добра.

б) самостійність роботи над проектом: Протягом виконання дипломного проекту здобувач освіти Нізяєв Іван поступово та послідовно виконував всі етапи, проявив ініціативу в створенні загальної концепції та реалізації роботи. Всі роботи здобувач освіти виконував самостійно, з оглядом на рекомендації керівника.

в) теоретична підготовка випускника (випускниці): Здобувач освіти Нізяєв Іван під час роботи над дипломним проектом вивчив достатньо багато літературних та інтернет-джерел за даною тематикою.

Вважаю, що теоретична підготовка дипломника достатня і він готовий до захисту проекту.

г) вміння розв'язувати виробничі та конструкторські питання Під час виконання дипломного проекту здобувач освіти Нізяєв Іван показав вміння організовано працювати над поставленим завданням, застосовувати знання у галузі програмування, електроніки, математики, розробляти, встановлювати та налаштовувати спеціалізоване програмне забезпечення, оформлювати слайди та складати презентації, користуючись сучасними комп'ютерними програмними засобами, такими як Embarcadero RAD Studio, Microsoft PowerPoint, Microsoft Visio та ін.

Оцінка розрахункової частини Добре  
Оцінка графічної частини Добре  
Загальна оцінка Добре

Прізвище, ім'я, по батькові керівника дипломного проекту Кривченко Юрій Вікторович

Місце роботи і посада керівника дипломного проекту ВСП "Одеський технічний фаховий коледж ОНТУ", викладач спецдисциплін комісії комп'ютерних технологій та програмної інженерії, голова циклової комісії комп'ютерних технологій та програмної інженерії

Підпис 

« 18 » серпень 2023 р.

**РЕЦЕНЗІЯ**

на дипломний проект здобувача (здобувачки) освіти  
відділення комп'ютерних систем

*Нізяєва Івана Романовича*

(прізвище, ім'я та по батькові)

Спеціальність 123 «Комп'ютерна інженерія»  
Освітня програма «Обслуговування комп'ютерних систем і мереж»  
Керівник дипломного проекту (роботи) Кривченко Юрій Вікторович

(прізвище, ім'я та по батькові)

Тема дипломного проекту (роботи) Розробка додатку для розрахунку стабілізаторів  
напруги живлення

Обсяг розрахунково-пояснювальної записки 75 сторінок  
Обсяг графічної (презентаційної) частини 14 аркушів (слайдів)

**ХАРАКТЕРИСТИКА ДИПЛОМНОГО ПРОЕКТУ (РОБОТИ)**

а) заключення про ступінь відповідності виконаного дипломного проекту завданню

Представлений на рецензію дипломний проект відповідає затвердженій темі та виконаний відповідно технічному завданню. Дипломний проект присвячений проблемі розрахунку схем стабілізаторів напруги живлення та складається з пояснювальної записки, додатку з програмним кодом та мультимедійної презентації, що містить приклади роботи програми.

б) характеристика виконання кожного розділу дипломного проекту

Пояснювальна записка складається з основного розділу (аналізу предметної області, проектування застосунку, реалізації застосунку, тестування застосунку), економічного розділу, розділу охорони праці та додатків. Перелічені розділи поетапно охоплюють розробку, виконані докладно та обґрунтовано. Розділ охорони праці містить загальну інформацію та вимоги до техніки безпеки оператора КТ. Економічний розділ проекту містить розрахунок витрат на НДР та реалізацію проекту.

в) оцінка якості виконання пояснювальної записки та графічної частини дипломного проекту

Графічна частина складається з 14 слайдів мультимедійної презентації, виконаної у програмному продукті MS PowerPoint, які містять ілюстративні схеми, скріншоти роботи програмного застосунку, передбачені технічним завданням. Пояснювальна записка виконана акуратно та у відповідності до норм. Якість виконання графічної частини проекту та пояснювальної записки добра, розробку виконано у повному обсязі.

г) перелік позитивних якостей дипломного проекту Реалізований алгоритм розрахунку схем стабілізаторів напруги живлення дозволяє достатньо точно виконувати розрахунки багатьох важливих параметрів за спрощеними методиками, передбачає перевірку некоректних даних. У роботі наведені докладні блок-схеми алгоритмів розрахунку стабілізаторів напруги живлення, виконано порівняння ручного та автоматичного розрахунку.

д) основні недоліки дипломного проекту \_\_\_\_\_

1. Існують більш точні методи розрахунку параметрів стабілізаторів напруги живлення.

2. У застосунку бажано було б також передбачити англomовний інтерфейс.

Оцінка розрахункової частини Відмінно

Оцінка графічної частини Добре

Загальна оцінка Відмінно

Прізвище, ім'я, по батькові рецензента к.т.н. Селіванова Алла Віталіївна

Місце роботи і посада рецензента Одеський національний технологічний університет,  
декан факультету комп'ютерної інженерії, програмування та кіберзахисту



Підпис: \_\_\_\_\_

« 17 » червня 2024 р.

Ім'я користувача:  
Катерина Григоріївна Краснокутська

ID перевірки:  
1016236339

Дата перевірки:  
08.05.2024 11:20:41 EEST

Тип перевірки:  
Doc vs Internet + Library

Дата звіту:  
08.05.2024 12:00:47 EEST

ID користувача:  
100011688

Назва документа: 4КС-57\_Іван\_Нізяєв

Кількість сторінок: 57 Кількість слів: 12972 Кількість символів: 95145 Розмір файлу: 932.34 KB ID файлу: 1016017859

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

## 22.4% Схожість

Найбільша схожість: 5.07% з Інтернет-джерелом ([https://stud.wiki/physics/3c0b65635b3bc78b4d53a89521206d37\\_1.html](https://stud.wiki/physics/3c0b65635b3bc78b4d53a89521206d37_1.html))

22.4% Джерела з Інтернету

881

Сторінка 59

Не знайдено джерел з Бібліотеки

## 0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

## 0% Вилучень

Немає вилучених джерел

## Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи

94

Підозріле форматування

9  
сторінок

**ДОЗВІЛ  
НА РОЗМІЩЕННЯ  
ВИПУСКНОЇ КВАЛІФІКАЦІЙНОЇ РОБОТИ  
(ДИПЛОМНОГО ПРОЕКТУ)  
В ЕЛЕКТРОННОМУ РЕПОЗИТАРІЇ ВСП «ОТФК ОНТУ»**

Ми, що нижче підписалися,

***Нізяєв Іван Романович,***  
здобувач освіти гр. 4КС-57, та  
***Кривченко Юрій Вікторович,***  
керівник дипломного проекту,

не заперечуємо щодо розміщення електронного варіанту пояснювальної записки до дипломного проекту фахового молодшого бакалавра на тему:

***«Розробка додатку для розрахунку стабілізаторів напруги живлення»  
(автор роботи – Нізяєв І.Р., керівник роботи – Кривченко Ю.В.)***

виконаного у ВСП «Одеський технічний фаховий коледж Одеського національного технологічного університету» в 2024 році, у повному обсязі в електронному репозитарії ВСП «ОТФК ОНТУ» для вільного доступу через мережу Інтернет.

Несемо відповідальність за ідентичність електронного та друкованого варіантів випускної кваліфікаційної роботи і даємо згоду на обробку персональних даних.

Виконавець



/ Нізяєв І.Р. /

Керівник



/ Кривченко Ю.В. /

«10» червня 2024 р.