

<https://doi.org/10.31891/2307-5732-2025-357-73>

УДК 621.31

**БУНЯК ОЛЕГ**

Тернопільський національний технічний університет імені Івана Пулюя

<https://orcid.org/0000-0001-9304-8254>e-mail: [buniak0203@gmail.com](mailto:buniak0203@gmail.com)**СТАСІВ АНДРІЙ**

Тернопільський національний технічний університет імені Івана Пулюя

<https://orcid.org/0009-0005-2966-5529>e-mail: [astasisv@expres1.com](mailto:astasisv@expres1.com)**ОРОБЧУК БОГДАН**

Тернопільський національний технічний університет імені Івана Пулюя

<https://orcid.org/0000-0002-6375-2440>e-mail: [orobchuk@tu.edu.te.ua](mailto:orobchuk@tu.edu.te.ua)**СУДОМИР ВАЛЕНТИН**

Тернопільський національний технічний університет імені Івана Пулюя

<https://orcid.org/0009-0007-1173-9272>e-mail: [exopuff@gmail.com](mailto:exopuff@gmail.com)

## РОЗРОБКА ПРОГРАМНОГО МОДУЛЯ ДЛЯ РОЗРАХУНКУ ПОРІВНЯЛЬНИХ ХАРАКТЕРИСТИК ТРАНСФОРМАТОРІВ З ВРАХУВАННЯМ УМОВИ ЕКОНОМІЧНОСТІ

*В статті розглядається питання розробки програмного модуля розрахунку порівняльних характеристик трансформаторів з використанням Windows Forms на мові програмування C# для впровадження в навчальний процес кафедри електричної інженерії Тернопільського національного технічного університету імені Івана Пулюя.*

*Ключові слова: програмний модуль, трансформатор, програмний інтерфейс, програмний блок.*

**BUNIAK OLEH****STASIV ANDRII****OROBCHUK BOGDAN****SUDOMYR VALENTYN**

Ternopil National Ivan Puluj Technical University

## DEVELOPMENT OF SOFTWARE MODULE FOR TRANSFORMERS COMPARATIVE CHARACTERISTICS CALCULATION TAKING INTO ACCOUNT COST-EFFECTIVENESS CONDITION

*The development of the software module for calculating the transformers comparative characteristics by means of Windows Forms in C# programming language, intended for the implementation in the educational process of the Department of Electrical Engineering at Ternopil Ivan Puluj National Technical University is considered in this paper.*

*Physically and morally outdated equipment in the electrical energy distribution links results in the increased technological costs. While analysing the reliable operation of electrical links of different voltage classes, the choice of power transformers, where the optimal criteria are capital and operating costs, technical characteristics and cost of electricity, annual operating time and electrical load schedule is of great importance. This approach leads to the increased complexity of calculations, constant changes in input data or their uncertainty, taking into account operation in overload modes and frequent outages during martial law. Therefore, it is important to develop software for quick calculation of transformers comparative characteristics.*

*The proposed software module for calculating the transformers comparative characteristics has intuitive user interface, and high accuracy of calculations is ensured by the application of variables of the Double type for mathematical calculations, ensuring compatibility with most .NET mathematical libraries. One of the key functions of the program is the input data verification mechanism: the program module is highly resistant to user errors, and the program is built on the modular architecture and takes into account the possibility of future functionality updates minimizing the risk of incorrect results and increasing the reliability of calculations. The proposed software interface, which provides the choice of languages - the formation of input data for calculations in the software module in Ukrainian or English - is universal and can be integrated into any application with localized interface to facilitate adaptation for different language groups.*

*Key words: software module, transformer, program interface, program block.*

Стаття надійшла до редакції / Received 01.07.2025

Прийнята до друку / Accepted 15.08.2025

### Постановка проблеми

Воєнний стан в країні наочно підтвердив провідну роль електроенергетики для забезпечення основних цілей розвитку економіки. Натомість, в результаті здійснення постійних відключень споживачів від джерел електроенергії для забезпечення балансу потужності, загострилися проблеми ефективності роботи електричних мереж. Основні причини – експлуатація фізично та морально застарілого обладнання на ланках розподілення електричної енергії та, відповідно, зростання технологічних витрат. Велика кількість відмов електричного обладнання, зростання часу ліквідації аварійних режимів, підвищення експлуатаційних витрат є основними проблемами при функціонуванні електричних мереж [1,2]. Трансформаторні підстанції (ТП) є основним елементом забезпечення надійності роботи електричних мереж. Тому, забезпечення ефективної роботи ТП та розробка заходів підвищення їх експлуатаційної надійності є першочергове завдання.

### Аналіз останніх джерел

При проектуванні або модернізації трансформаторів різних класів напруги вибір силових трансформаторів проводять на основі розрахункової потужності та коефіцієнтів навантаження в номінальних та аварійних режимах роботи. При цьому не беруть до уваги ряд економічних складових: вартість приведених втрат при роботі силових трансформаторів, ціна на електричну енергію та режими споживання [3, 4].

Тому, для вибору потужності силових трансформаторів ТП при проведенні розрахунку їх порівняльних характеристик оптимальними критеріями є капітальні та експлуатаційні витрати, технічні характеристики й вартість електроенергії, річна тривалість роботи та графік електричних навантажень. [5].

Зрозуміло, що підвищення складності розрахунків, постійна зміна вхідних даних або їх невизначеність призводить до зростання похибок при обчисленні. Тому, актуальним є розробка програмного забезпечення для швидкого розрахунку порівняльних характеристик трансформаторів.

Запропонована методика вибору потужності трансформаторів використовувалася в навчальному процесі в ряді дисциплін кафедри електричної інженерії Тернопільського національного технічного університету імені Івана Пулюя. Прогресивні розробки кафедри в напрямку побудови програмних модулів для забезпечення ефективної роботи автоматизованих електроенергетичних комплексів [6-7] дозволили виконати поставлене завдання.

### Виклад основного матеріалу

Програма для розрахунку порівняльних характеристик трансформаторів розроблена з використанням Windows Forms на мові програмування C#. Вибір цієї платформи та мови був обумовлений простотою та швидкістю розробки програмного забезпечення з графічним користувацьким інтерфейсом. Windows Forms забезпечує легке створення візуально зрозумілих форм, кнопок та інших елементів інтерфейсу, що значно спрощує роботу користувачів з програмою. Крім того, ця технологія добре інтегрується з іншими компонентами .NET-екосистеми, що дозволяє швидко додавати нові функції або змінювати існуючі [8].

При розробці програми розрахунку було враховано необхідність високої точності обчислень, тому всі змінні, які використовуються для математичних розрахунків, мають тип Double. Використання 64-розрядних чисел подвійної точності для арифметики з плаваючою комою дозволяє уникнути помилок, які пов'язані із округленнями, та забезпечує більш точні результати, особливо при роботі з невеликими або дуже великими числовими значеннями. Використання типу Double має ще одну перевагу – забезпечення сумісності з більшістю математичних бібліотек .NET [9].

Однією з ключових функцій програми є механізм верифікації вхідних даних. Користувачі вводять числові значення у відповідні текстові поля, після чого програма автоматично перевіряє, чи можна введений текст перетворити на число типу Double. Якщо введене значення не відповідає числовому формату або є некоректним (наприклад, містить літери чи спеціальні символи), програма замінює його на одиницю, що дозволяє уникнути помилок під час розрахунків і забезпечує стабільну роботу програми навіть при некоректному введенні даних користувачем. Такий підхід мінімізує ризик некоректних результатів і підвищує надійність розрахунків [10]:

```
//Функція перевірки значення у TextBox на валідність та заміна його на 1 у разі не валідності
double parseTextBox(TextBox textbox)
{
    TypeConverter converter = TypeDescriptor.GetConverter(typeof(double));
    if (converter.IsValid(textbox.Text))
    {
        Return double.Parse(textbox.Text, System.Globalization.CultureInfo.InvariantCulture);
    }
    else
    {
        textbox.Text = "1";
    }
    return 1;
}
```

Необхідність створення означеної програми зумовлена потребою у спрощенні розрахунків для кінцевого користувача. Виконання всіх обчислень вручну за допомогою формул є складним, трудомістким і часто призводить до помилок. Програма автоматизує цей процес, забезпечуючи швидкість і точність розрахунків, що значно підвищує ефективність роботи інженерів і технічних фахівців. Завдяки цьому, інженери можуть зосередитися на аналізі отриманих результатів, замість витрачання часу на тривалі обчислення.

Програма розроблена з урахуванням потреб як новачків, так і досвідчених користувачів. Інтуїтивно зрозумілий інтерфейс та простота використання дозволяють швидко освоїти її функціонал без необхідності додаткового навчання. Це робить її ідеальним інструментом для освітніх цілей. Зручність використання також досягається завдяки чіткому розташуванню елементів інтерфейсу та

інтуїтивно зрозумілим кнопкам:

```
//Нажаття на кнопку розрахунку
private void button2_Click(object sender, EventArgs e)
{
    //Блок зчитування значень з полів, значень введених користувачем
    double full_capacity_of_power_consumers = parseTextBox(textBox1); //Sp
    double Power_on_time = parseTextBox(textBox2); //tvkl
    double Cost_change_factor = parseTextBox(textBox3); //Kzm.vtr
    double The_cost_of_one_kilowatt_of_energy = parseTextBox(textBox4); //m
    double coefficient_of_amortization_deductions = parseTextBox(textBox5); //φ
    double rated_power_of_transformers_tr1 = parseTextBox(textBox6); //Str1
    double load_factor_in_normal_mode_tr1 = parseTextBox(textBox8); //
    double load_factor_in_emergency_mode_tr1 = parseTextBox(textBox10); //
    double value_of_idle_losses_tr1 = parseTextBox(textBox12); //ДРxx.tr1
    double value_of_short_circuit_losses_tr1 = parseTextBox(textBox14); //ДРkz.tr1
    double value_of_no_load_current_tr1 = parseTextBox(textBox16); //Ixx.tr1
    double value_of_the_short_circuit_voltage_tr1 = parseTextBox(textBox18); //Ukz.tr1
    double The_cost_of_transformers_tr1 = parseTextBox(textBox20); //Cтр1
    double rated_power_of_transformers_tr2 = parseTextBox(textBox7); //Str2
    double load_factor_in_normal_mode_tr2 = parseTextBox(textBox9); //
    double load_factor_in_emergency_mode_tr2 = parseTextBox(textBox11); //
    double value_of_idle_losses_tr2 = parseTextBox(textBox13); //ДРxx.tr1
    double value_of_short_circuit_losses_tr2 = parseTextBox(textBox15); //ДРkz.tr1
    double value_of_no_load_current_tr2 = parseTextBox(textBox17); //Ixx.tr1
    double value_of_the_short_circuit_voltage_tr2 = parseTextBox(textBox19); //Ukz.tr1
    double The_cost_of_transformers_tr2 = parseTextBox(textBox21); //Cтр1
}
```

Крім того, розроблена програма враховує можливість майбутнього оновлення функціоналу. Вона побудована за модульною архітектурою, що дозволяє легко додавати нові функції або адаптувати існуючі під специфічні потреби користувачів. Це робить програму довгостроковим рішенням, яке може розвиватися разом із потребами користувачів:

```
//Блок розрахунку та заповнення полів обчисленими значеннями
double calculated_load_factor_in_normal_mode_tr1 = full_capacity_of_power_consumers /
(rated_power_of_transformers_tr1 * load_factor_in_normal_mode_tr1); //Kzav.tr1.nom
textBox22.Text = calculated_load_factor_in_normal_mode_tr1.ToString();
double calculated_load_factor_in_emergency_mode_tr1 = full_capacity_of_power_consumers /
(rated_power_of_transformers_tr1 * load_factor_in_emergency_mode_tr1); //Kzav.tr1.avar
textBox24.Text = calculated_load_factor_in_emergency_mode_tr1.ToString();
double Losses_of_electricity1_tr1 = rated_power_of_transformers_tr1 *
(value_of_no_load_current_tr1 / 100.0); //ДРxx.tr1
textBox26.Text = Losses_of_electricity1_tr1.ToString();
double Losses_of_electricity2_tr1 = rated_power_of_transformers_tr1 *
(value_of_the_short_circuit_voltage_tr1 / 100.0); //ДРkz.tr1
textBox28.Text = Losses_of_electricity2_tr1.ToString();
double Losses_of_electricity3_tr1 = value_of_idle_losses_tr1 + (Cost_change_factor *
Losses_of_electricity1_tr1); //ДРxx.sh.tr1
textBox30.Text = Losses_of_electricity3_tr1.ToString();
double Losses_of_electricity4_tr1 = value_of_short_circuit_losses_tr1 + (Cost_change_factor *
Losses_of_electricity2_tr1); //ДРkz.sh.tr1
textBox32.Text = Losses_of_electricity4_tr1.ToString();
double Losses_of_electricity5_tr1 = Losses_of_electricity3_tr1 +
(calculated_load_factor_in_normal_mode_tr1 * Losses_of_electricity4_tr1); //ДРsh.tr1
textBox34.Text = Losses_of_electricity5_tr1.ToString();
double Losses_of_electricity6_tr1 = load_factor_in_normal_mode_tr1 * Losses_of_electricity5_tr1;
//ДРPl.2.sh.tr1
textBox36.Text = Losses_of_electricity6_tr1.ToString();
double Losses_of_electricity7_tr1 = Losses_of_electricity6_tr1 * Power_on_time; //ДРetr1
textBox38.Text = Losses_of_electricity7_tr1.ToString();
double Losses_of_electricity8_tr1 = Losses_of_electricity7_tr1 *
The_cost_of_one_kilowatt_of_energy; //Ce.tr1
textBox40.Text = Losses_of_electricity8_tr1.ToString();
double Capital_costs_tr1 = load_factor_in_normal_mode_tr1 * The_cost_of_transformers_tr1; //
Kz.tr1
```

```

textBox42.Text = Capital_costs_tr1.ToString();
double Operating_costs_tr1 = coefficient_of_amortization_deductions * Capital_costs_tr1; // Ca1
textBox44.Text = Operating_costs_tr1.ToString();
double Total_costs_tr1 = Losses_of_electricity8_tr1 + Operating_costs_tr1; // C1
textBox46.Text = Total_costs_tr1.ToString();
double calculated_load_factor_in_normal_mode_tr2 = full_capacity_of_power_consumers /
(rated_power_of_transformers_tr2 * load_factor_in_normal_mode_tr2); // Kzav.tr2.nom
textBox23.Text = calculated_load_factor_in_normal_mode_tr2.ToString();
double calculated_load_factor_in_emergency_mode_tr2 = full_capacity_of_power_consumers /
(rated_power_of_transformers_tr2 * load_factor_in_emergency_mode_tr2); // Kzav.tr2.avar
textBox25.Text = calculated_load_factor_in_emergency_mode_tr2.ToString();
double Losses_of_electricity1_tr2 = rated_power_of_transformers_tr2 *
(value_of_no_load_current_tr2 / 100.0); // DQxx.tr2
textBox27.Text = Losses_of_electricity1_tr2.ToString();
double Losses_of_electricity2_tr2 = rated_power_of_transformers_tr2 *
(value_of_the_short_circuit_voltage_tr2 / 100.0); // DQkz.tr2
textBox29.Text = Losses_of_electricity2_tr2.ToString();
double Losses_of_electricity3_tr2 = value_of_idle_losses_tr2 + (Cost_change_factor *
Losses_of_electricity1_tr2); // DPxx.sh.tr2
textBox31.Text = Losses_of_electricity3_tr2.ToString();
double Losses_of_electricity4_tr2 = value_of_short_circuit_losses_tr2 + (Cost_change_factor *
Losses_of_electricity2_tr2); // DPkz.sh.tr2
textBox33.Text = Losses_of_electricity4_tr2.ToString();
double Losses_of_electricity5_tr2 = Losses_of_electricity3_tr2 +
(calculated_load_factor_in_normal_mode_tr2 * calculated_load_factor_in_normal_mode_tr2 *
Losses_of_electricity4_tr2); // DPsh.tr2
textBox35.Text = Losses_of_electricity5_tr2.ToString();
double Losses_of_electricity6_tr2 = load_factor_in_normal_mode_tr2 * Losses_of_electricity5_tr2;
// DP1.2.sh.tr2
textBox37.Text = Losses_of_electricity6_tr2.ToString();
double Losses_of_electricity7_tr2 = Losses_of_electricity6_tr2 * Power_on_time; // DEtr2
textBox39.Text = Losses_of_electricity7_tr2.ToString();
double Losses_of_electricity8_tr2 = Losses_of_electricity7_tr2 *
The_cost_of_one_kilowatt_of_energy; // Ce.tr2
textBox41.Text = Losses_of_electricity8_tr2.ToString();
double Capital_costs_tr2 = load_factor_in_normal_mode_tr2 * The_cost_of_transformers_tr2; //
Kz.tr2
textBox43.Text = Capital_costs_tr2.ToString();
double Operating_costs_tr2 = coefficient_of_amortization_deductions * Capital_costs_tr2; // Ca2
textBox45.Text = Operating_costs_tr2.ToString();
double Total_costs_tr2 = Losses_of_electricity8_tr2 + Operating_costs_tr2; // C2
textBox47.Text = Total_costs_tr2.ToString();

double Payback_period = Math.Abs((Capital_costs_tr1 - Capital_costs_tr2) / (Total_costs_tr1 -
Total_costs_tr2)); // Tok
textBox48.Text = Payback_period.ToString();
double Economic_efficiency = Total_costs_tr2 - Total_costs_tr1; // E
textBox49.Text = Economic_efficiency.ToString();

```

Запропонована програма не лише забезпечує високу точність розрахунків порівняльних характеристик трансформаторів, але й має інтуїтивно зрозумілий інтерфейс, підвищену стійкість до помилок користувачів.

Інтерфейс програми передбачає можливість вибору мови: користувачі мають можливість здійснювати формування вхідних даних для розрахунку в програмному модулі українською або англійською мовами. Це реалізовано для зручності використання розрахункового програмного модуля ширшим колом користувачів, зокрема, англомовними фахівцями або співробітниками міжнародних організацій. Усі текстові підписи, кнопки та повідомлення автоматично змінюють свою мову відповідно до вибору користувача, що робить програму інтуїтивно зрозумілою і комфортною для роботи. Блок-схему виконання програмного блоку зміни мови представлення інформації подано на рисунку 1.

Перемикання мов реалізоване за допомогою ресурсних файлів, що забезпечує легкість підтримки й масштабування цієї функції:

```

private void button1_Click (object sender, EventArgs e)
{
    string lng = CultureInfo.CurrentCulture.Name;

```

```

if (CultureInfo.CurrentCulture.Name == "en-US")
{
    System.Threading.Thread.CurrentThread.CurrentUICulture = new
System.Globalization.CultureInfo("uk");
    System.Threading.Thread.CurrentThread.CurrentCulture = new
System.Globalization.CultureInfo("uk");
}
else
{
    System.Threading.Thread.CurrentThread.CurrentUICulture = new
System.Globalization.CultureInfo("en-US");
    System.Threading.Thread.CurrentThread.CurrentCulture = new
System.Globalization.CultureInfo("en-US");
}
ComponentResourceManager resources = new ComponentResourceManager(typeof(Form1));
resources.ApplyResources(this, "$this");
applyResources(resources, this.Controls);
}

```

Представлений вище Код у програмному блоці для зміни мови користувацького інтерфейсу демонструє простий підхід до динамічної зміни мови у Windows Forms-додатках. Основна ідея полягає в тому, щоб змінити поточну культуру `CurrentUICulture` і `CultureInfo` відповідно до вибраної мови, а потім застосувати нові локалізовані ресурси до всіх елементів форми. Основними перевагами такого рішення є:

- простота реалізації: Код використовує стандартні .NET-засоби для управління локалізацією, що забезпечує зрозумілість та легкість у впровадженні;
- динамічність: зміна мови відбувається без необхідності перезапуску додатку, що підвищує зручність користування;
- масштабованість: метод `applyResources` дозволяє оновлювати мову для будь-якої кількості елементів форми, незалежно від їх складності.

Загальний вигляд запропонованого програмного модуля «Порівняльний розрахунок трансформаторів» (рис. 2) зрозумілий і наочний, умовно розділений на чотири блоки: два блоки – введення вхідних даних, а наступних два – отримання розрахункових значень для проведення аналізу щодо вибору оптимального варіанту побудови схеми ефективного використання трансформаторів.

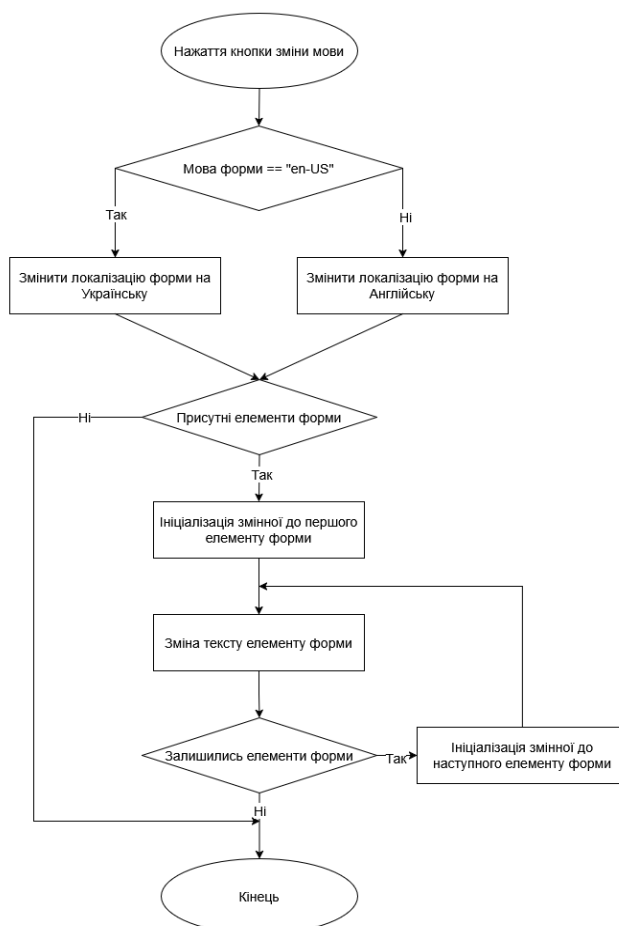


Рис.1. Блок – схема виконання блоку програмного модуля зміни мови представлення інформації

В основі розрахунків вибору одотрансформаторної чи двотрансформаторної організації трансформації електричної енергії покладено статистичні моделі, де, окрім врахування основних електричних параметрів й режимів роботи трансформаторів, а також потужності силових споживачів і вартості електричної енергії, до уваги беруться розраховані експлуатаційні затрати, як елемент економічної складової.

Рис.2. Загальний вигляд програмного модуля «Порівняльний розрахунок трансформаторів»

В першому блоці програмного модуля (рис. 3) в якості вхідних даних подається: значення потужності силових споживачів із врахуванням освітлення та компенсаційних установок; параметри, які враховують характер зміни електричних навантажень: час включення та коефіцієнт зміни втрат; вартість електричної енергії та коефіцієнт експлуатаційних відрахувань.

Рис.3. Вигляд першого блоку програмного модуля «Порівняльний розрахунок трансформаторів»

Другий блок програмного модуля (рис. 4) як елемент вхідних даних для проведення розрахунків означає характеристики силових трансформаторів за паспортними даними: повна потужність, втрати потужності холостого ходу та потужності короткого замикання, напруги короткого замикання та струму холостого ходу; параметри, які враховують характер впливу зміни електричних навантажень: перевантаження силових трансформаторів. В якості економічного чинника – вартість трансформаторів при проведенні реконструкції (заміни).

Рис.4. Вигляд другого блоку програмного модуля «Порівняльний розрахунок трансформаторів»

Третій блок програмного модуля (рис. 5), на основі врахування режимів завантаження трансформаторів (номінальний, аварійний), дозволяє розрахувати приведені втрати в трансформаторах як основні чинники експлуатаційних затрат: холостого ходу, короткого замикання, електричної енергії та сумарних річних затрат.

|                                                | Трансформатор 1      | Трансформатор 2      |
|------------------------------------------------|----------------------|----------------------|
| Коефіцієнт завантаження в нормальному режимі   | <input type="text"/> | <input type="text"/> |
| Коефіцієнт завантаження в аварійному режимі    | <input type="text"/> | <input type="text"/> |
| Реактивна потужність холостого ходу, кВАр      | <input type="text"/> | <input type="text"/> |
| Реактивна потужність короткого замикання, кВАр | <input type="text"/> | <input type="text"/> |
| Приведені втрати холостого ходу, кВт           | <input type="text"/> | <input type="text"/> |
| Приведені втрати короткого замикання, кВт      | <input type="text"/> | <input type="text"/> |
| Приведені втрати електроенергії, кВт           | <input type="text"/> | <input type="text"/> |
| Приведені втрати в трансформаторах, кВт        | <input type="text"/> | <input type="text"/> |
| Втрати електроенергії за рік, кВт*год          | <input type="text"/> | <input type="text"/> |
| Вартість втрат електроенергії за рік, грн      | <input type="text"/> | <input type="text"/> |
| Капітальні затрати, грн                        | <input type="text"/> | <input type="text"/> |
| Річні експлуатаційні затрати, грн              | <input type="text"/> | <input type="text"/> |
| Сумарні річні затрати, грн                     | <input type="text"/> | <input type="text"/> |

Рис.5. Вигляд третього блоку програмного модуля «Порівняльний розрахунок трансформаторів»

Кінцевий вибір схемної реалізації на основі проведених розрахунків з використанням програмного модуля здійснюється за розрахунковими значеннями четвертого програмного блоку (рис. 9): терміну окупності та економічної ефективності.

|                              |                      |
|------------------------------|----------------------|
| Термін окупності, років      | <input type="text"/> |
| Економічна ефективність, грн | <input type="text"/> |

Рис.6. Вигляд четвертого блоку програмного модуля «Порівняльний розрахунок трансформаторів»

Окрім означених вище особливостей статистичної моделі при проведенні вибору трансформаторів слід зазначити, що запропоновану методику з врахуванням умови економічності з використанням розробленого модуля можливо використовувати при заміні недовантажених трансформаторів на трансформатори меншої потужності.

### Висновки

Мета проекту – розробка програмного модуля розрахунку порівняльних характеристик трансформаторів з використанням Windows Forms на мові програмування C# для впровадження в навчальний процес кафедри електричної інженерії Тернопільського національного технічного університету імені Івана Пулюя, що дозволяє на відміну від використання стандартних пакетів прикладних програм, покращити наочність проведених досліджень. Запропонований підхід програмної реалізації зміни мови представлення інформації в програмному модулі є універсальним і може бути інтегрований у будь-який додаток з локалізованим інтерфейсом для полегшення адаптації до різних мовних аудиторій.

### Література

1. Трещов М. Модернізація енергетичного сектору як пріоритетний напрям повоєнного відновлення України. Науковий вісник: Державне управління. 2024. № 1 (15). С. 28–45. URL: [https://doi.org/10.33269/2618-0065-2024-1\(15\)-28-45](https://doi.org/10.33269/2618-0065-2024-1(15)-28-45) (дата звернення: 13.06.2025).
2. Methodology for selecting the rated parameters of power transformers in the town's electrical distribution networks in settlements / Yu. Papaika et al. Electrical Engineering and Power Engineering. 2021. No. 2. P. 33–43. URL: <https://doi.org/10.15588/1607-6761-2021-2-4> (date of access: 13.06.2025).
3. Романюк Ю., Соломчак О., Федорів М. Вибір оптимальної потужності силових трансформаторів знижувальних підстанцій підприємств нафтогазової галузі з урахуванням умови економічності. Нафтогазова енергетика. 2015. № 1. С. 45–51.
4. Харченко В., Якунін О., Воропай В. Електропостачання міст та промислових підприємств : конспект лекцій (для студентів усіх форм навчання галузі знань 14 – Електрична інженерія, спеціальності 141 – Електроенергетика, електротехніка та електромеханіка, професійне спрямування «Електротехнічні системи електроспоживання») : конспект лекцій. Харків : ХНУМГ ім. О. М. Бекетова, 2019. 238 с.
5. Лежнюк П., Добровольська Л., Кулик В. Електроощадні технології в електричних мережах енергосистем : навч. посіб. / ред. П. Лежнюк. Луцьк : IBV Луц. НТУ, 2018. 328 с.
6. Оробчук Б., Буняк О. Методика застосування системи керування режимами електропостачання у навчальному процесі. Матеріали VII міжнародної науково-технічної конференції «Підвищення рівня ефективності енергоспоживання в електротехнічних пристроях і системах». Луцьк, 2018. С. 67–71.
7. Development of Software for the Implementation of Automated Reserve Input Modes Operation / B. Orobchuk et al. 2nd International Workshop on Computer Information Technologies in Industry 4.0 (CITI 2023). Ternopil, Ukraine, 2024. P. 316–336.

8. Івохін Є., Махно М., Піскунов О. Розробка додатків засобами мови програмування C#: Навч.-метод. посібник для проведення лабораторних робіт для студентів вищих навчальних закладів спеціальності «Системний аналіз». К. : Видавничо-полігр. центр "Київ. ун-т", 2021. 100 с.

9. Базурін В., Чашечникова О. Об'єктно-орієнтоване програмування на мові C# Поглиблений рівень Лабораторний практикум. Суми, 2022. 81 с.

10. Коноваленко І., Марущак П. Платформа .NET та мова програмування C# 8.0 : навч. посіб. Тернопіль : ФОП Паляниця В. А., 2020. 320 с.

### References

1. Treshchov M. Modernizatsiia enerhetychnoho sektoru yak priorityetnyi napriam povoiennoho vidnovlennia Ukrainy. Naukovyi visnyk: Derzhavne upravlinnia. 2024. № 1 (15). S. 28–45. URL: [https://doi.org/10.33269/2618-0065-2024-1\(15\)-28-45](https://doi.org/10.33269/2618-0065-2024-1(15)-28-45) (data zvernennia: 13.06.2025).

2. Methodology for selecting the rated parameters of power transformers in the town's electrical distribution networks in settlements / Yu. Papaika et al. Electrical Engineering and Power Engineering. 2021. No. 2. P. 33–43. URL: <https://doi.org/10.15588/1607-6761-2021-2-4> (date of access: 13.06.2025).

3. Romaniuk Yu., Solomchak O., Fedoriv M. Vybir optymalnoi potuzhnosti sylovykh transformatoriv znyzhuvalnykh pidstantsii pidpriemstv naftohazovoi haluzi z urakhuvanniam umovy ekonomichnosti. Naftohazova enerhetyka. 2015. № 1. S. 45–51.

4. Kharchenko V., Yakunin O., Voropai V. Elektropostachannia mist ta promyslovykh pidpriemstv : konspekt lektsii (dlia studentiv usikh form navchannia haluzi znan 14 – Elektrychna inzheneriia, spetsialnosti 141 – Elektroenerhetyka, elektrotekhnika ta elektromekhanika, profesiine spriamuvannia «Elektrotekhnichni systemy elektrospozhyvannia») : konspekt lektsii. Kharkiv : KhNUMH im. O. M. Beketova, 2019. 238 s.

5. Lezhniuk P., Dobrovol'ska L., Kulyk V. Elektroshchadni tekhnolohii v elektrychnykh merezhakh enerhosystem : navch. posib. / red. P. Lezhniuk. Luts'k : IVV Luts'k. NTU, 2018. 328 s.

6. Orobchuk B., Buniak O. Metodyka zastosuvannia systemy keruvannia rezhymamy elektropostachannia u navchalnomu protsesi. Materialy VII mizhnarodnoi naukovo-tekhnicnoi konferentsii «Pidvyshchennia rivnia efektyvnosti enerhospozhyvannia v elektrotekhnichnykh prystroiah i systemakh». Luts'k, 2018. S. 67–71.

7. Development of Software for the Implementation of Automated Reserve Input Modes Operation / B. Orobchuk et al. 2nd International Workshop on Computer Information Technologies in Industry 4.0 (CITI 2023). Ternopil, Ukraine, 2024. P. 316–336.

8. Ivokhin Ye., Makhno M., Piskunov O. Rozrobka dodatkov zasobamy movy prohramuvannia C#: Navch.-metod. posibnyk dlia provedennia laboratornykh robot dlia studentiv vyshchykh navchalnykh zakladiv spetsialnosti «Systemnyi analiz». K. : Vydavnycho-polih. tsentr "Kyiv. un-t", 2021. 100 s.

9. Bazurin V., Chashechnykova O. Ob'iektno-orientovane prohramuvannia na movi C# Pohlyblenyi riven Laboratornyi praktykum. Sumy, 2022. 81 s.

10. Konovalenko I., Marushchak P. Platforma .NET ta mova prohramuvannia C# 8.0 : navch. posib. Ternopil : FOP Palianytsia V. A., 2020. 320 s.