

вычисления правил, которые не могут дать новых фактов. Указанный принцип является основой для разработки эффективных методов вычисления в СУДБД. Более того, без его применения невозможно решение целого класса задач, связанных с проблемами комбинаторного перечисления. Например, в ряде случаев необходимо учитывать вывод внешне одинаковых фактов, полученных на разных шагах вывода, т.е. имеющих различную внутреннюю структуру (если из вершины а в вершину b существуют несколько путей, стандартная процедура вывода определит только единственный путь). Таким образом, принцип минимальности итеративных вычислений позволяет работать с мультимножествами фактов.

6. Декларативность языковых расширений. Как отмечалось, проблема расширения языка новыми конструкциями не перестает быть актуальной для разработчиков СУДБД. Есть два способа расширения языка: декларативное и процедурное. Ко второму, например, можно отнести реализацию функции backtracking и предикатов явного изменения (удаления и добавления) фактов из базы знаний. Несмотря на относительную простоту реализации процедурных расширений в логическом языке, их внедрение, с одной стороны, может привести к неоднозначной интерпретации программ, а с другой стороны, противоречит идее декларативного решения задач. Рассматриваемый принцип предполагает использование декларативных расширений языка, либо таких процедурных расширений, которые имеют прозрачную естественно-языковую трактовку и однозначную интерпретацию.

Выводы.

С позиций системного подхода проанализирована проблема применения методов и технологий искусственного интеллекта в информационных системах типа СППР, сформулированы принципы организации взаимодействия традиционных СУБД с системами дедуктивного вывода. Изложенные принципы нашли отражение в разработанном авторами

действующем прототипе СУДБД «Datalog server», ориентированном на выполнение задач в реляционной среде. Опытно-промышленная эксплуатация сервера показала принципиальную возможность и практическую целесообразность его применения для решения как традиционных, так и слабоформализованных задач анализа данных в организационно-технических системах.

Литература

1. Гарсиа-Молина Г., Ульман Дж., Уидом Дж. Системы баз данных. Полный курс. : Пер. с англ. – М.: «Вильямс», 2003. – 1088 с.
2. Говоров Ф. П., Говоров В. Ф., Верещук Б. М. Оптимизация режимов систем электроснабжения городов // ЕЕJET, № 2, 2003. – С. 28-33.
3. Басакер Р., Саати Т. Конечные графы и сети. : Пер. с англ. – М.: Наука, 1973. – 368 с.
4. Анфилатов В. С. и др. Системный анализ в управлении / В.С. Анфилатов, А.А. Емельянов, А.А. Кукушкин; под ред. А.А. Емельянова. – М.: Финансы и статистика, 2003. – 368 с.
5. Ларичев О. И. Теория и методы принятия решений. – М.: Логос, 2002. – 392 с.
6. Фернандес Х., Минкер Дж. Теория и алгоритмы дизъюнктивных дедуктивных баз данных // Программирование, 1993, № 3. – С. 4-38.
7. Чери и др. Логическое программирование и базы данных. : Пер. с англ. – М.: Мир, 1992. – 352 с.
8. Nilsson U., Maluszynski J. Logic, Programming and Prolog. John Wiley & Sons Ltd., 1995. – 294 p.
9. Буслик Н.Н., Летучий Д.А. Отрицательные литералы и вычислительная парадигма Дейталога // Проблемы бионики. Харьков, 2001, № 55. – С. 3-8.
10. Буслик Н.Н., Летучий Д.А. Организация объектно-ориентированного предикативного синтаксического анализатора // Проблемы бионики. Харьков, 2002, № 56. – С. 21 – 25.

УДК 621.314

ГРАФІЧНИЙ РЕДАКТОР ЕЛЕКТРИЧНИХ СХЕМ ДЛЯ ОБ'ЄКТНО-ОРІЄНТОВАНОГО МОДЕЛЮВАННЯ ПЕРЕТВОРЮВАЧІВ ЕЛЕКТРИЧНОЇ ЕНЕРГІЇ

В.В. Коломієць

Аспірант кафедри Автоматики та радіоелектроніки Української інженерно-педагогічної академії, вул. Університетська 16, м. Харків, Україна, 61003.

Контактний телефон: +38(0572) 32-80-43.

E-mail: VladKolomiets@rambler.ru

В.Г. Ягуп

Доктор технічних наук. Професор кафедри Автоматики та радіоелектроніки Української інженерно-педагогічної академії, вул. Університетська 16, м. Харків, Україна, 61003.

Контактний телефон: +38(0572) 20-64-87.

Проектування та аналіз електромагнітних процесів у схемах електронних пристроїв за допомогою комп'ютера, як правило, супроводжується кропітким введенням

Розглянуто питання, пов'язані з розробкою графічного редактора обробки схем тиристорних перетворювачів. Розробка здійснюється для дослідження електромагнітних процесів в тиристорних перетворювачах за допомогою об'єктно-орієнтованої моделі тиристорного перетворювача, що заснована на методах діакоптики. Обґрунтовано застосування і представлена реалізація об'єктно-орієнтованої моделі графічного редактора тиристорного перетворювача. Працездатність редактора підтверджено прикладом.

Вступ

параметрів аналізованої схеми. Схема, яка підлягає аналізуванню, повинна бути перетворена у еквівалентну схему, зрозумілу для подальшого аналізування математичним апаратом комп'ютерної моделі. Введення параметрів елементів та складання еквівалентної схеми відіграє певне значення на результатах аналізу. Особливістю комп'ютерного моделювання слід зазначити, що правильно скласти еквівалентну схему та ввести її параметри в комп'ютер можливо лише за умови глибокого знання не лише предмету дослідження, а і самої математичної моделі.

Урахування великої кількості елементів та їх параметрів при проведенні аналізу електромагнітних процесів тиристорних перетворювачів електричної енергії вимагає від розробників засобів комп'ютерного моделювання розробку інструментів введення схеми. Завдання складання еквівалентної схеми та контролю правильності введення параметрів доцільно покласти на комп'ютерну програму для моделювання перетворювача електричної енергії.

Етапи розвитку моделювання схем перетворювачів

Розвиток комп'ютерного моделювання тиристорних перетворювачів електричної енергії безпосередньо пов'язаний з еволюцією цифрової обчислювальної техніки та програмного забезпечення (ПЗ).

Продовж значного часу ПЗ комп'ютерів базувалось на використанні програм консольного типу. Даний період розвитку характеризується величезними здобутками в методології та теорії прикладного ПЗ. Результатом даного періоду являється цілий ряд програм широкого застосування, у тому числі і програм для моделювання тиристорних перетворювачів електричної енергії. Особливостями застосування ПЗ цього періоду є поетапне введення елементів еквівалентної схеми в пам'ять комп'ютера та перегляд масивів введеної інформації на дисплеї. Можливості збереження в файлі і подальшого завантаження схеми з диску значно полегшують роботу.

Зростання технічних характеристик комп'ютерів у 90 роках забезпечило використання графічного інтерфейсу користувача, замінивши консольні програми. Використання пристроїв графічного введення інформації дає можливість максимально спростити процес введення, наблизивши його до малювання. Цей факт швидко знайшов застосування у програмах моделювання електротехнічних пристроїв. Значне зростання обчислювальних потужностей комп'ютерів дозволяє розробникам ПЗ оснастити програми можливостями автоматичного формування еквівалентних схем. Тому, першочерговим завданням для розробника ПЗ моделювання перетворювачів електричної енергії є розробка ПЗ, здатного представляти схему у виді повністю відповідного малюнка, з можливістю обробки вихідних даних у звичному для користувача виді: графіки, таблиці та ін.

Графічний редактор схеми – універсальна система обробки схем перетворювачів електричної енергії

Застосування графічного введення інформації про схему перетворювача електричної енергії дозволить звести до мінімуму помилки, пов'язані з введенням параметрів елементів схеми. Однак треба виключити помилки, пов'язані з топологією схеми та допоміжними перетвореннями. При появі помилок введення параметрів схеми їх можливо легко виправити замінюючи параметри новими,

використовуючи малюнок елемента на схемі, в якому допущено помилку. Відстеження топологічних з'єднань схеми та допоміжних перетворень повністю покладено на модель перетворювача та розробника ПЗ.

Сучасне ПЗ з аналізу електричних схем можливо поділити на дві групи. До першої групи можна віднести математичне ПЗ широкого застосування: MATLAB фірми MathWorks Inc та ін. До ПЗ другої групи можна віднести програми, спрямовані на розрахунок електричних схем: Accel EDA и P-CAD фірми ACCEL Technologies Inc., Serenade Design Environment фірми Ansoft Corporation, Electronics Workbench, MicroCAP та ін.

ПЗ першої групи засновано на універсальному математичному апараті та має у своєму розпорядженні математичні моделі елементів електронних систем.

Друга група ПЗ, хоча і орієнтується на розрахунок електромагнітних процесів в схемах та являються універсальними програмами, неспроможна ефективно виконати розрахунок перетворювачів електричної енергії з урахуванням запропонованого в [1] розділення на частини і подальшого ефективного відстеження усіх перемикачів, з можливостями розділення обчислювального процесу.

Найуживаніші елементи схем перетворювачів

Розглядаючи систему введення інформації в програмах для графічних ОС, як систему графічного вводу даних, слід вказати на доцільність використання графічних редакторів. Вцілому графічний редактор являє собою ПЗ з обробки графічної інформації. При розробці ПЗ з аналізу перетворювачів електричної енергії графічною інформацією виступає малюнок схеми. Завданням графічного редактора є забезпечення обробки схеми та передача даних про схему на програму аналізування. Для функціонування редактора схеми необхідна розробка графічних елементів керування редактором. Розвиток ПЗ для графічних ОС виправдав застосування панелей інструментів з обробки графічних об'єктів. При розробці графічного редактора схеми перетворювача слід виділити основний об'єкт редагування – лист схеми, та об'єкти малювання на листі, якими виступають елементи схеми перетворювача.

Функціонування графічного редактора схеми перетворювача забезпечується такими панелями інструментів:

- “Стандартна” – основна панель редактора забезпечує роботу інших панелей та роботу з малюнком схеми в цілому: зберегти, завантажити та ін;
- “Малювання” – містить інструменти малювання елементів схеми та інструменти управління;
- “Параметри” – містить параметри нового або виділеного елемента на схемі;
- “Орієнтація” – панель зміни орієнтації елемента на листі, містить додаткову інформацію про новий або виділений елемент на схемі.

Панель інструментів “Малювання” являється основною панеллю при редагуванні схеми перетворювача електричної енергії. Можливості панелі дозволяють виконувати обробку листа схеми такими інструментами:

- “Виділити” – інструмент дозволяє виділити один або групу елементів на листі та виконати до них дію переміщення або видалення;
- “З'єднання” – з'єднувальний провідник двох елементів на листі;

- “Джерело напруги”, “Джерело струму”, “Резистор”, “Ємність”, “Котушка індуктивності”, “Діод”, “Тиристор”, “Ключ” – інструменти відповідних елементів схем перетворювачів;

- “Заземлення” – інструмент заземлення;

- “Лінія розділення” – інструмент розділення схеми на частини, для діакоптичних перетворювачів.

Перераховані інструменти дозволяють виконувати основні дії з складання схеми перетворювача. Так, застосування інструментів “З’єднання”, “Лінія розділення” виконується методом протягування. Курсор миші встановлюється на листі, вказуючи початок дії інструменту, натиснувши ліву кнопку миші, виконують переміщення в потрібну зону, та відпускають клавішу, що вказує на кінець дії інструменту. При переміщенні розміри об’єкту змінюються, що дозволяє відстежити необхідний розмір об’єкта.

Інструмент “Виділити” може працювати як в режимі протягування, так і в режимі одиночного натискання клавіш миші. При протягуванні виділяються декілька об’єктів, при одиночному натисканні - тільки об’єкт, що знаходиться під курсором миші.

Інструменти групи елементів схеми працюють при одиночному натисканні лівої клавіші миші. Інструменти цієї групи відповідають за введення елементів схеми.

Зміна курсорів миші при роботі з редактором вказує на обраний інструмент і дозволяє забезпечити однозначність в процесі обробки схеми. Зовнішній вигляд курсорів миші тотожний обраним інструментам.

Принципи реалізації схеми на робочому листі

Основну частину вікна редактора займає робочий лист схеми перетворювача. При потребі, можна вимкнути всі панелі інструментів, і лист схеми буде займати все вікно редактора. Робота з редактором схеми в режимі введення елемента виконується у такій послідовності:

- Обирається інструмент, який відповідає необхідному елементу;
- За необхідності, для обраного інструменту встановлюються параметри;
- На листі встановлюється елемент (натисканням лівої клавіші миші).

Встановлення наступного елемента виконується таким самим чином. Поєднання елементів виконується інструментом “З’єднання”. При встановленні елементів на робочий лист редактор схеми автоматично визначає положення встановленого елемента по відношенню до інших. Тому точність введення елементів схеми не потрібна, редактор сам визначить необхідність з’єднання близько розміщених контактів елементів. На малюнку 1 зона дії сполучення контактів елементів позначена літерою «D», діаметр контактів «d», довжина та ширина елемента відповідно «L», «S».

Використання сполучення контактів при редагуванні схеми дозволяє автоматично підтримувати з’єднання елементів схеми, що дозволяє сформуванню однозначний образ схеми. Автоматичне з’єднання елементів схеми працює в області з’єднання «D» (рис. 1.), розмір якої заздалегідь встановлено рівним ширині елементів, але за необхідності його можна змінювати. Автоматичне з’єднання контактів розповсюджується на обидва контакти інструменту, що дозволяє відслідковувати введення елемента.

При автоматичному з’єднанні контактів елемент не змінює своїх розмірів, тому за головний контакт для з’єднання обирається той, що знаходиться під курсором миші.

На будь якому етапі введення схеми перетворювача можливе її збереження на диску. Файли даних схеми мають розширення імені *.sch, и являються файлами текстового типу. Кожен рядок файла містить інформацію про один елемент схеми. Інформація в рядок записується в наступному порядку: тип елемента; опис елемента; орієнтація; параметр; форма сигналу; період керування; час початку імпульсу керування; час закінчення імпульсу керування; номери вузлів; номер підсхеми; номера підсхем для зв’язку; графічні координати початкового та кінцевого вузлів.

На етапі введення схеми одночасно зі з’єднанням контактів елементів автоматично проводиться і нумерація вузлів схеми. В порівнянні з реалізацією нумерації для всієї схеми, нумерування вузлів в процесі введення елемента є очевидною перевагою. Робота з автоматичного нумерування вузлів схеми непомітна для користувача програми.

Заключним етапом по введенню схеми користувачем можна вважати етап розділення схеми на частини. Для виконання розділення схеми застосовується інструмент “Лінія розділення”. Редагування схеми, розділення схеми на частини супроводжується інформацією статусу: поточні координати, кількість підсхем та ін. Відправлення схеми на розрахунок супроводжується автоматичним занесенням даних до відповідних об’єктів розрахунку.

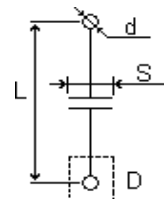


Рисунок 1.
Зображення елемента

Операції з елементами на листі:

Необхідність обробки елементів на листі вимагають визначення додаткових вимог до операцій з елементами на листі. Так для елементів на листі, визначено операції переміщення, видалення, зміни параметрів елементів, ці операції реалізує інструмент “Виділити”.

В редакторі схем перетворювача реалізовано зміну розмірів елементів схеми. Зміна розмірів виконується на панелі інструментів “Розміри”, через маркери розміру. Маркери розміру задають в точках розміри елементів, довжину стрілок, розмір контакту та зону сполучення контактів елементів.

Для листа схеми визначено операції копіювання схеми в буфер обміну з метою подальшого перенесення в інші редактори у виді графічної області. Надана можливість забезпечує вставку схеми в документи, звіти, в сукупності з можливістю копіювання графіків, це забезпечує формування звіту проведеного аналізу перетворювача.

Операції з інформаційними масивами, алгоритми обробки

Основним алгоритмом обробки графічного редактора є алгоритм додавання елемента на лист схеми, основними етапами якого є:

- вирахування координат другого вузла елемента;
- з’єднання графічних координат елементів та обробка номерів вузлів;
- малювання елемента;

- додавання елемента до контейнера елементів графічної схеми.

Вирахування координат другого вузла елемента виконується в залежності від орієнтації елемента на листі, та базується на координатах першого вузла елемента і довжині елемента.

З'єднання графічних координат елементів та обробка номерів вузлів заснована на перевірці координат елемента, що додається з усіма елементами на листі та обома їх контактами. З'єднання графічних координат елементів відбувається, якщо контакти попадають в площину «D» рис. 1. За умови відсутності з'єднання елемента з іншими елементами, він отримує унікальні графічні координати та номери вузлів.

Обробка елементів, що мають з'єднання з іншими, виконується за правилами:

- якщо елемент, що вводиться, являє собою “З'єднання”, то обидва вузли нумеруються єдиним номером. Якщо з'єднання виконується на обох вузлах “З'єднання”, то вузол групи елементів замінюється новим унікальним номером рис. 2;



Рисунок 2. З'єднання груп елементів інструментом “З'єднання”

- якщо елемент не являється “З'єднанням”, то його початковий вузол приймає значення вузла з'єднання, а інший вузол отримує унікальний номер;

- якщо елемент являє собою “Заземлення”, то група елементів, з якою він з'єднується, отримує номер базового вузла. Отримання номера елементами виконано в виді заміни всіх номерів, з якими виконано з'єднання на новий номер, в даному випадку номер базового вузла.

Алгоритм графічного з'єднання елементів та нумерування вузлів показано на рис. 3. Особливістю функціонування алгоритму є те, що в рамках одного алгоритму виконано функціонування графічного з'єднання елементів та присвоєння вузлам елементів нумерації.

Окремо слід зазначити алгоритм перевірки нумерації вузлів при видаленні елементів схеми. Функціонування даного алгоритму засновано на положенні перевірки нумерації в залежності від типу видаленого елемента. Так, при видаленні елемента “З'єднання”, утворюється дві рівноправні групи елементів, що веде до необхідності заміни вузла однієї з груп на новий унікальний номер. Вузол, який підлягає заміні, повинен бути не базовим вузлом. При видаленні елемента “Заземлення”, також треба виконати заміну вузла новим унікальним номером, за умови відсутності ще одного елемента “Заземлення”, підключеного до вузла.

Малювання елементів схеми покладено на спеціальний алгоритм. Образ, який підлягає малюванню, залежить від типу елемента та орієнтації елемента на листі.

Атрибути елементів



Рисунок 3. Структурна схема алгоритму автоматичного з'єднання елементів та нумерації вузлів

Обробка схеми редактором супроводжується зміною атрибутів елементів схеми перетворювача. З характером інформації атрибуту елементів можна виділити такі групи:

- група графічних атрибутів – до неї відносяться атрибути, що відповідають за положення елемента на листі, а також атрибути сукупності елементів: геометричні розміри елемента, розмір зони сполучення елементів;
- група електричних атрибутів елемента відповідає за зберігання електричних параметрів елемента;
- група топологічних атрибутів відповідає за структуру з'єднання елементів та включає такі атрибути: номери вузлів елемента, номер підсхеми до якої належить елемент, номера підсхем, якщо елемент являється гілкою сполучення підсхем. Слід окремо зазначити наявність додаткового атрибуту опису елемента, призначенням якого є нести додаткову інформацію, наприклад, номер елемента в групі собі подібних. Даний атрибут елемента може виступати як індекс, що відображає призначення елемента, наприклад, n – навантаження.

Об'єктно-орієнтована реалізація графічного редактора

Реалізація графічного редактора виконана у виді класів [2]. Центральне місце в структурі класів займає клас графічної схеми перетворювача. Відповідальність класу графічної схеми – це підтримання інформації про графічну схему перетворювача. В таблиці 1. наведено функції та атрибути класу графічної схеми перетворювача.

Таблиця 1.

Листинг класу графічної схеми перетворювача

Номер блоку	Зміст
1	#ifndef GSchemaH
2	#define GSchemaH
3	#include "GElements.h"
4	#include "ConForm.h"
5	class TGSchema{
6	private: AnsiString FileName; // ім'я файлу схеми
7	public: AnsiString GetFileName(){return FileName;}
8	void PutFileName(AnsiString fileName){FileName=fileName;}
9	void LoadFromFile(AnsiString fileName);
10	void SaveToFile(AnsiString fileName);
11	void GSchemaToDSchema(); // Перетворення графічної схеми
12	TGElements* GElements; // елементи схеми
13	TGSchema();~TGSchema();;
14	#endif

Реалізація класу графічної схеми перетворювача виконано в окремому модулі "GSchema". Основне завдання класу графічної схеми полягає в зберіганні (таблиця 1., блок 12) та передачі елементів на подальший аналіз (таблиця 1., блок 11). Перевагання різноманітними операціями по обслуговуванню елементів покладено на контейнерний клас елементів графічної схеми (таблиця 1., блок 12). Операції графічної обробки елементів схеми покладено на клас графічного інтерфейсу користувача – клас графічного редактора "TEditorSchemaForm". В рамках котрого виконуються операції графічної обробки схеми перетворювача. Взаємодія класів графічної схеми та редактора утворює єдиний механізм обробки схеми тиристорного перетворювача. Даний механізм дозволяє в зручному для користувача виді редагувати схему, виконувати аналіз електромагнітних процесів в схемі перетворювача електричної енергії.

Особливості реалізації графічного редактора

Інструментом реалізації графічного редактора обрано систему розробки програмних додатків C++Builder [3]. В якості основи для розробки редактора було обрано стандартну форму – екземпляр класу TForm. В якості листа схеми перетворювача виступає об'єкт зображення – екземпляр класу TImage, Класи TForm, TImage, а також допоміжні класи реалізації клавш TSpeedButton, панелей TPanel та TLabel є стандартними класами реалізації інтерфейсу користувача і автоматично підключаються з бібліотеки компонентів C++Builder. Використані бібліотечні компоненти C++Builder, мають реалізацію і в інших системах розробки програмних продуктів Visual Basic, Visual C++, Delphi, що дозволяє виконати реалізацію графічного редактора на інших платформах. Розроблений графічний редактор дозволяє поводити зрозумілі на інтуїтивному рівні операції зі схемою перетворювача та одержати програмний продукт легкий та зрозумілий у використанні.

Можливості використання UML та Rational Rose

Розробка графічного редактора схем перетворювачів – це реалізація варіанту використання [4] "Введення схеми перетворювача" на якісно новому рівні. Введення схе-

ми перетворювача за допомогою графічного редактора схем – це основна ланка роботи користувача ПЗ з аналізування електромагнітних процесів у перетворювачах електричної енергії (рис. 4).

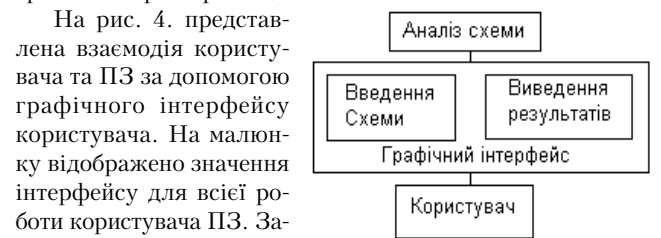


Рисунок 4.

Взаємодія користувача та ПЗ

чів виконати аналіз схеми перетворювача електричної енергії, не замислюючись над роботою по введенню схеми, підготовці необхідних налаштувань інформаційних масивів та інших підготовчих процесах аналізу схем перетворювачів.

Приклад використання

Виконаємо аналіз послідовно-паралельного інвертора, виконаного за мостовою схемою [5]. Нормовані параметри схеми: $E=100$ В, $L_d=47,5$ Гн., $C_{\text{посл.}}=C_{\text{пар.}}=0,0985$ Ф., $R_H=1,08$ Ом., $L_H=1$ Гн., B_1, B_2 – тиристори ($T_{\text{упр.}}=2$ с., $t_{\text{вкл.}}=0$ с.), B_3, B_4 – ($T_{\text{упр.}}=2$ с., $t_{\text{вкл.}}=1$ с.). Розрахунок виконувався з кроком 0,001 с., час закінчення 30 с. при розбивці схеми на дві підсхеми, з виділенням L_d зв'язком. Зовнішній вигляд інтерфейсу та робочого листа редактора схеми представлено на рис. 6.

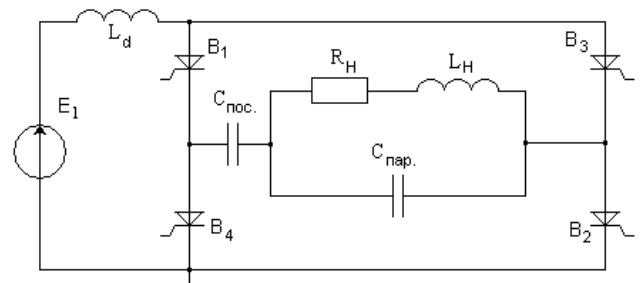


Рисунок 5. Послідовно-паралельний інвертор, мостова схема

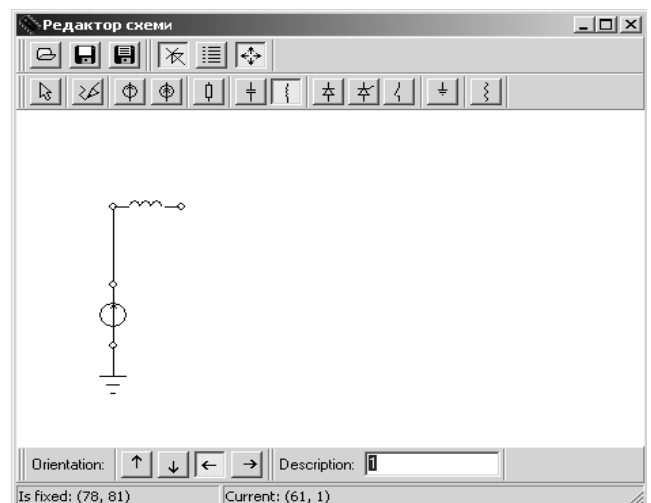


Рисунок 6. Вікно редактора схеми. Редагується схема інвертора

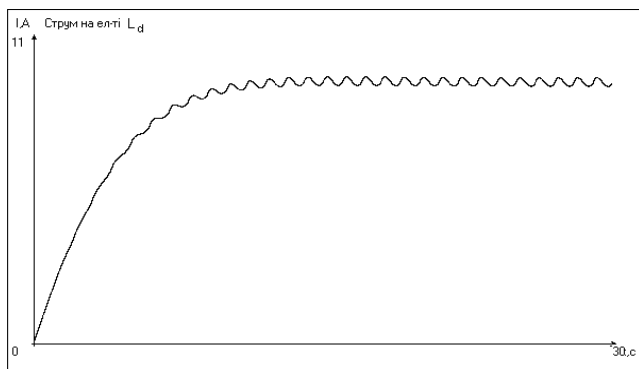


Рисунок 7. Вхідний струм інвертора

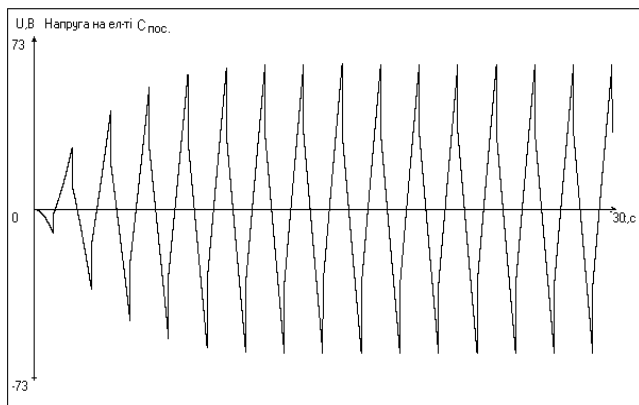


Рисунок 8. Напряга на конденсаторі послідовному навантаженню

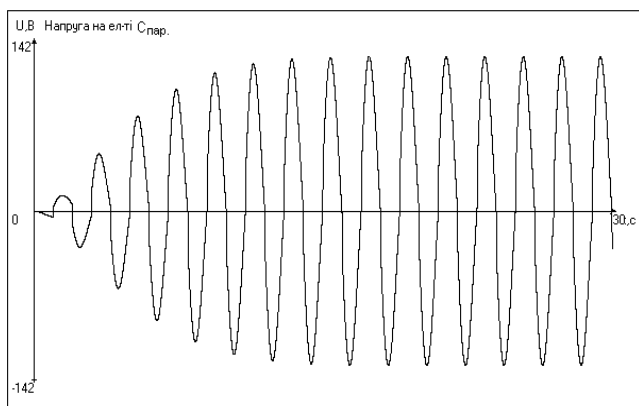


Рисунок 9. Напряга на конденсаторі навантаження

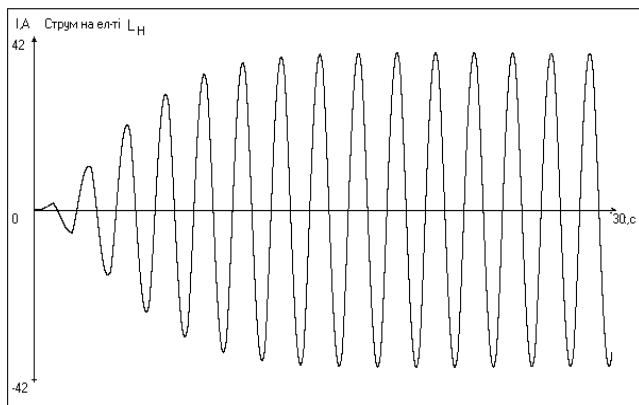


Рисунок 10. Струм навантаження

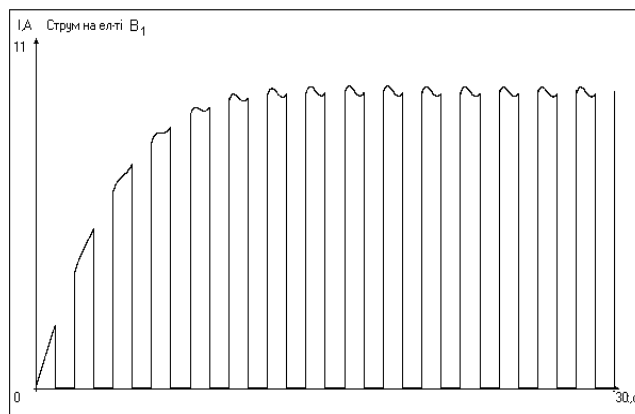


Рисунок 11. Струм тиристора V_1

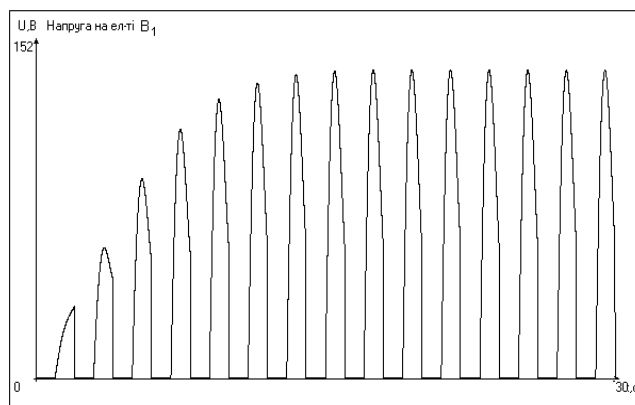


Рисунок 12. Напряга на тиристорі V_1

Висновки

Розробка графічного редактора обробки схем тиристорних перетворювачів дозволила значно полегшити процес редагування схем. Вона здійснювалась для дослідження електромагнітних процесів в тиристорних перетворювачах за допомогою об'єктно-орієнтованої моделі тиристорного перетворювача, що заснована на методах діакоптки. Застосування графічного інтерфейсу користувача в поєднанні з апаратом розділення схем на частини дозволяє підняти на якісно новий рівень процес аналізу схем перетворювачів електричної енергії.

Література

1. Коломієць В.В. Дослідження електромагнітних процесів в перетворювачах постійного струму на основі об'єктно-орієнтованого програмування // Східно-Європейський журнал передових технологій. №5, 2003. с. 58-61
2. Буч Гради. Объектно-ориентированный анализ и проектирование с примерами приложений на C++, 2-е изд. / Пер. с англ. – М.: Бинум, 2000. – 560 с.
3. Глушаков С. В., Зорянский В. Н., Хоменко С. Н. Программирование в среде Borland C++ Builder 6. Худож. оформитель А. С. Юхтман. – Х.: Фолио, 2003. – 508 с.
4. Кватрани Т. Rational Rose 2000 и UML. Визуальное моделирование. Пер. с англ. – М.: ДМК Пресс, 2001. – 176 с.: ил.
5. Беркович Е. И., Ивенский Г. В., Иоффе Ю. С., Матчак А. Т., Моргун В. В. Тиристорные преобразователи высокой частоты. Л.: «Энергия», Э.- 1973.- 200 с. с ил.