

# РАДІОЕЛЕКТРОНІКА ТА РАДІОЕЛЕКТРОННЕ АПАРАТОБУДУВАННЯ

УДК 621.317

В. М. Кичак<sup>1</sup>  
 М. Д. Гузь<sup>1</sup>  
 В. В. Кичак<sup>1</sup>  
 В. В. Олійник<sup>1</sup>

## МЕТОД АВТОМАТИЗОВАНОГО СИНТЕЗУ ЦИФРОВИХ РАДІОТЕХНІЧНИХ ПРИСТРОЇВ З ЧАСТОТНО-ІМПУЛЬСНИМ КОДУВАННЯМ ІНФОРМАЦІЇ

<sup>1</sup>Вінницький національний технічний університет

*Розглянуто метод синтезу цифрових радіотехнічних пристроїв з частотно-імпульсним кодуванням інформації. Відмінність методу полягає в тому, що базовими елементами для синтезу є фізичні схеми, які реалізують елементарні функції.*

*Для синтезу таких пристроїв запропоновано підхід, який ґрунтується на виконанні фізичними схемами перетворень елементарних операцій в елементарні оператори, що реалізують частотно-імпульсні логічні функції.*

*Розроблено методику та алгоритм побудови таблиці функціонування цифрових радіотехнічних пристроїв з частотно-імпульсним кодуванням інформації для двійкового структурного алфавіту. Запропоновані аналітичні вирази розрахунку повних проміжних результатів для однозначно та неоднозначно залежних частотно-імпульсних логічних функцій.*

*З метою розрахунку кількості та значень допоміжних сигналів, використовуваних для побудови цифрових пристроїв, запропоновано розраховувати функції відхилення, які є різницею повних проміжних результатів і відповідного значення частотно-імпульсних логічних функцій. Кількість різних значень функцій відхилення відповідає кількості допоміжних сигналів, необхідних для реалізації пристрою, а різні значення функції відхилення є значеннями допоміжних сигналів.*

*Запропоновано алгоритм побудови функції належності, який опрацьовує значення функції відхилення та забезпечує фільтрацію.*

*Розроблена методика складання операторного опису, на базі якого будується структурна схема цифрових радіотехнічних пристроїв з частотно-імпульсним кодуванням інформації.*

*З використанням запропонованих алгоритмів розроблено програму структурного синтезу цифрових радіотехнічних пристроїв з частотно-імпульсним кодуванням інформації. Програма написана на мові програмування Java, 8 версія JRE SE. Для розробки програми використано платформу JavaFX 2.2. Фреймворк для автоматичного компонування jar пакету — Apache Maven. Додаткові бібліотеки: GSON, JFXtras.*

**Ключові слова:** частотно-імпульсне кодування, повний проміжний результат, функція відхилення, функція належності, частотно-імпульсна логічна функція.

### Вступ і постановка задач дослідження

Застосування частотно-імпульсного методу кодування інформації (ЧІКІ) для побудови цифрових радіотехнічних пристроїв (ЦРТП) забезпечує підвищення швидкодії за рахунок переходу в діапазон надвисоких частот та завадостійкості, оскільки частоти, які відповідають різним логічним станам, можуть бути значно рознесені по діапазону [1], [2].

В працях [3], [4] показано, що для підвищення швидкодії та зменшення кількості з'єднань між інтегральними елементами, які в сучасних великих інтегральних схемах можуть займати до 80 % об'єму, доцільно застосовувати недвійкові структурні алфавіти.

Водночас застосування недвійкових структурних алфавітів та імпульсно-потенціального кодування інформації приводить до зниження завадостійкості. Цей недолік певною мірою можна усунути шляхом застосування частотно-імпульсного методу кодування інформації [1].

Відомі методи синтезу ЦРТП ґрунтуються на застосуванні функціонально повного набору логічних елементів, які дають можливість синтезувати будь-які цифрові пристрої.

В роботі [1] показано, що для розробки таких пристроїв з ЧКІ доцільно використовувати нетрадиційний метод синтезу, який ґрунтується на застосуванні частотно-імпульсних логічних функцій (ЧЛФ) та базових фізичних схем для їх реалізації, що сприяє спрощенню структурних схем та зниженню енергоспоживання.

Метою роботи є розробка алгоритму та програми автоматизованого синтезу ЦРТП з ЧКІ.

### Метод синтезу ЦРТП з ЧКІ

Для заданих вхідної множини  $X$  і вихідної множини  $Y$  задача синтезу ЦРТП з ЧКІ полягає в знаходженні операторної послідовності  $P$ , яка задовольняє умовам  $Y = PX$ , і побудові на базі цієї послідовності структурну схему ЦРТП з ЧКІ. Використовуючи базові схеми, можна розробити принципову електричну схему.

Для пристроїв, які використовують довільний структурний алфавіт, пропонуються такі етапи синтезу ЦРТП з ЧКІ.

На першому етапі синтезу потрібно побудувати таблицю функціонування ЦРТП з ЧКІ. Кількість стовпчиків в таблиці функціонування визначається кількістю змінних  $n$ . Кількість рядків в таблиці функціонування залежить від кількості змінних функцій, які виконує цифровий пристрій, і дорівнює для довільного структурного алфавіту  $m^n$  частотних наборів,  $m = \{2, \dots, 10\}$  — визначає структурний алфавіт. Зміст кожного стовпчика визначається відповідним частотним набором.

У випадку двійкового структурного алфавіту, перший стовпчик містить  $2^{n-1}$  частотних наборів, які відповідають логічному «0», і  $2^{n-1}$  частотних наборів, що відповідають логічній «1» ( $\omega_0$  і  $\omega_1$ , відповідно). Другий стовпчик містить  $2^{n-2}$  частотних наборів, які відповідають логічному «0», і  $2^{n-2}$  частотних наборів, що відповідають логічній «1», і ці частотні набори повторюються двічі.

Третій стовпчик містить  $2^{n-3}$  частотних наборів, які відповідають логічному «0», і  $2^{n-3}$  частотних наборів, що відповідають логічній «1», і ці частотні набори повторюються тричі.

Четвертий стовпчик містить  $2^{n-4}$  частотних наборів, які відповідають логічному «0», і  $2^{n-4}$  частотних наборів, що відповідають логічній «1», і ці частотні набори повторюються чотири рази і т. д.

Тобто, кожний стовпчик містить  $2^{n-k}$  частотних наборів. Коли  $n = k$  процес формування таблиці функціонування ЦРТП з ЧКІ завершується. Структурна схема алгоритму побудови таблиці функціонування ЦРТП з ЧКІ показана на рис. 1.

На наступному етапі синтезу таблиця функціонування ЦРТП з ЧКІ доповнюється двома стовпчиками, один з яких відповідає значенню функції, за відповідного частотного набору та значенню повного проміжного результату (ППР), для відповідного частотного набору, який визначається за виразом

$$z_i = \omega_{i1} + \omega_{i2} + \omega_{in} = \sum_{j=1}^n \omega_{ij}. \quad (1)$$

Якщо однаковим значенням  $z_i$  відповідають різні значення  $y_i$ , то маємо справу з неоднозначно залежною функцією, і тоді для розрахунку ППР  $z_i$  будемо використовувати такий вираз

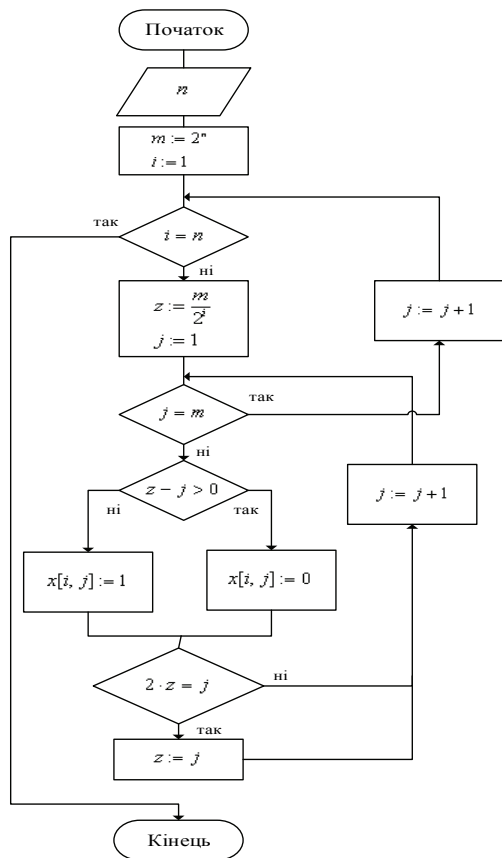


Рис. 1. Структурна схема алгоритму побудови таблиці функціонування ЦРТП з ЧКІ

$$z_i = \sum_{j=1}^n j\omega_{ij} \quad (2)$$

Тобто, для того, щоб різним значенням частотних наборів відповідали різні значення ППР, пропонується суму значень змінних для кожного рядка, формувати з різними коефіцієнтами, залежно від позиції змінної в частотному наборі.

На наступному етапі здійснюється визначення кількості та значень допоміжних сигналів. Для цього проводиться порівняння ППР  $z_i$  і відповідного значення функції  $y_i$  для кожного рядка, в результаті чого визначаються функції відхилення:

$$\Delta_i = z_i - y_i \quad (3)$$

Кількість різних значень функцій відхилення відповідає кількості допоміжних сигналів, необхідних для реалізації відповідного пристрою, а значення допоміжних сигналів відповідають різним значенням функції відхилення.

Розрахунок значень функції відхилення проводиться за виразом (3).

На наступному етапі визначаються функції належності.

Розрахунок значень функції належності реалізується алгоритмом, структурна схема якого зображена на рис. 2, який опрацьовує значення функції відхилення та повертає значення функції належності, що визначаються за виразом:

$$C_{ij} = \begin{cases} 1, & \text{якщо } \Delta_j \neq \Delta_i \text{ і } y_i = \omega_n; \\ 0, & \text{якщо } \Delta_j \neq \Delta_i. \end{cases} \quad (4)$$

Дії, які виконуються на цих етапах, для зручності, можна зобразити у вигляді суміщеної таблиці (табл. 1).

Таблиця 1

$x_1$	$x_2$	...	$x_n$	$y$	$z$	$\Delta$	$C_1$	$C_2$	...	$C_g$

На підставі цього, значення  $y_i$  ЧЛФ обчислюється за виразом

$$y_i = \delta_{i1}C_{i1} + \delta_{i2}C_{i2} + \dots + \delta_{ig}C_{ig},$$

де  $C_{ij}$  – забезпечують фільтрацію, тобто вказують на належність або неналежність значенням функції  $y_i$  значень  $\delta_{ij}$ .

Тому функції  $C_{ij}$  будемо називати функціями належності.

На наступному етапі визначаються функції фільтрації.

Таблиця 2

$C_j$	$f_i(\omega_0)$	$f_j(\omega_1)$	...	$F_j(\omega_{k-1})$

Оскільки значення функції  $C_{ij}$  забезпечують фільтрацію  $f(\omega_0), f(\omega_1), f(\omega_2), \dots, f(\omega_{i-1})$  (табл. 2), то для кожної з цих функцій будуються таблиці відповідності функціям фільтрації  $f_j(\omega_0), f_j(\omega_1), f_j(\omega_2), \dots, f_j(\omega_{k-1})$ , значення яких визначаються у такий спосіб:

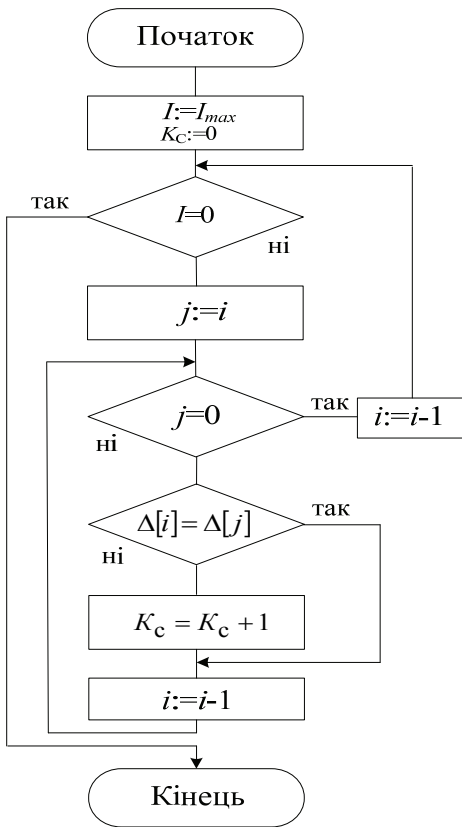


Рис. 2. Структурна схема алгоритму розрахунку значень функції належності

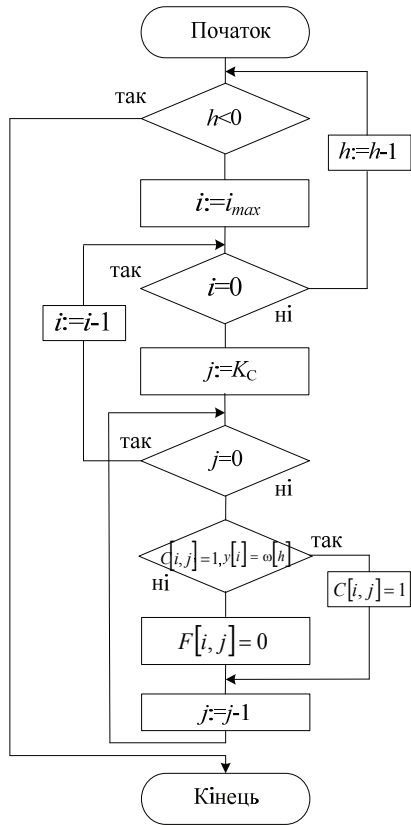


Рис. 3. Структурна схема алгоритму розрахунку значень функції фільтрації

$$F_{ij}(\omega) = \begin{cases} 1, C_{ij} = 1; \\ 0, C_{ij} \neq 1. \end{cases} \quad (5)$$

Розрахунок значень функції фільтрації реалізується алгоритмом, структурна схема якого зображена на рис. 3.

### Складання операторного опису

Операторний опис складається, на підставі структури узагальненого операторного опису (6). В полі вхідних сигналів вказуються всі інформаційні та допоміжні сигнали, що визначені на етапі 2. В полі операторів, до дужок, наводиться група операторів  $F\Phi_B$  у кількості  $(n - 1)$ , а також оператор  $T$ , після якого розташовано  $g$  символів  $\uparrow$ .

$$x_1 \uparrow x_2 \uparrow x_3 \uparrow \dots x_n \uparrow x_{n+1} \uparrow \dots x_{n+g} \uparrow : \\ \downarrow \downarrow F\Phi_B \downarrow F\Phi_B \dots \downarrow F\Phi_B T \uparrow \dots \uparrow \left( \downarrow \downarrow FT \uparrow \uparrow \dots \uparrow \downarrow \downarrow FT \uparrow \uparrow \dots \uparrow \right) (6) \\ \left( \downarrow \dots \downarrow A\Phi_c^0 \uparrow \downarrow \dots \downarrow A\Phi_c^1 \uparrow \dots \downarrow \dots A\Phi_c^{k-1} \right) \downarrow \dots \downarrow F\Phi_B \downarrow F\Phi_B \dots \downarrow F\Phi_B T \uparrow \\ \left( \downarrow \dots \downarrow A\Phi_c^0 \uparrow \downarrow \dots \downarrow A\Phi_c^1 \uparrow \dots \downarrow \dots A\Phi_c^{k-1} \right) \downarrow \dots \downarrow AT \downarrow \dots \downarrow A \uparrow \dots \downarrow \dots A \uparrow : \\ \downarrow y_1 \downarrow y_2 \dots \downarrow y_n$$

Кількість символів перед оператором  $A$  дорівнює кількості таблиць відповідності для  $C_{ij}$ , в яких стовпчик з назвою  $f_j(\omega_k)$  містить одиниці.

Кожний з цих символів пов'язаний однаковою номером з символом  $\uparrow$ , розташованим після оператора  $T$  в  $j$ -й групі в перших дужках.

Після других дужок записується  $m$  операторів  $A$ , перед кожним з яких записується  $k$  символів  $\uparrow \uparrow$ .

### Побудова структурної схеми

Структурна схема ЦРТП з ЧІКІ будується на підставі узагальненої структурної схеми, наведеної на рис. 4.

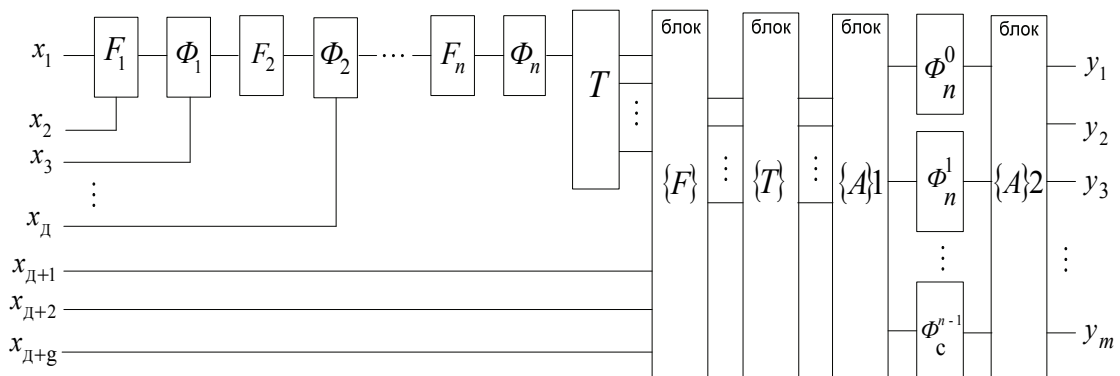


Рис. 4. Узагальнена структурна схема ЦРТП з ЧІКІ

На рис. 4 елементи  $F_B, F_H$  здійснюють операцію перетворення частоти двох сигналів, а елементи  $\Phi_B, \Phi_H$  — фільтри верхніх і нижніх частот, відповідно;  $\Phi_C$  — смугові фільтри,  $A$  — суматори,  $T$  — розгалужувачі.

Елементи  $T$  забезпечують розгалуження, а елементи  $A$  — об'єднання сигналів,  $\Phi_c^0 \dots \Phi_c^1$  — еле-

менти, що здійснюють виділення сигналів, частоти яких відповідають логічному нулю та одиниці, у випадку застосування двійкового структурного алфавіту.

Згідно із запропонованими алгоритмами, розроблено програму синтезу ЦРТП з ЧІКІ. Загальна структурна схема алгоритму синтезу ЦРТП з ЧІКІ показана на рис. 5.

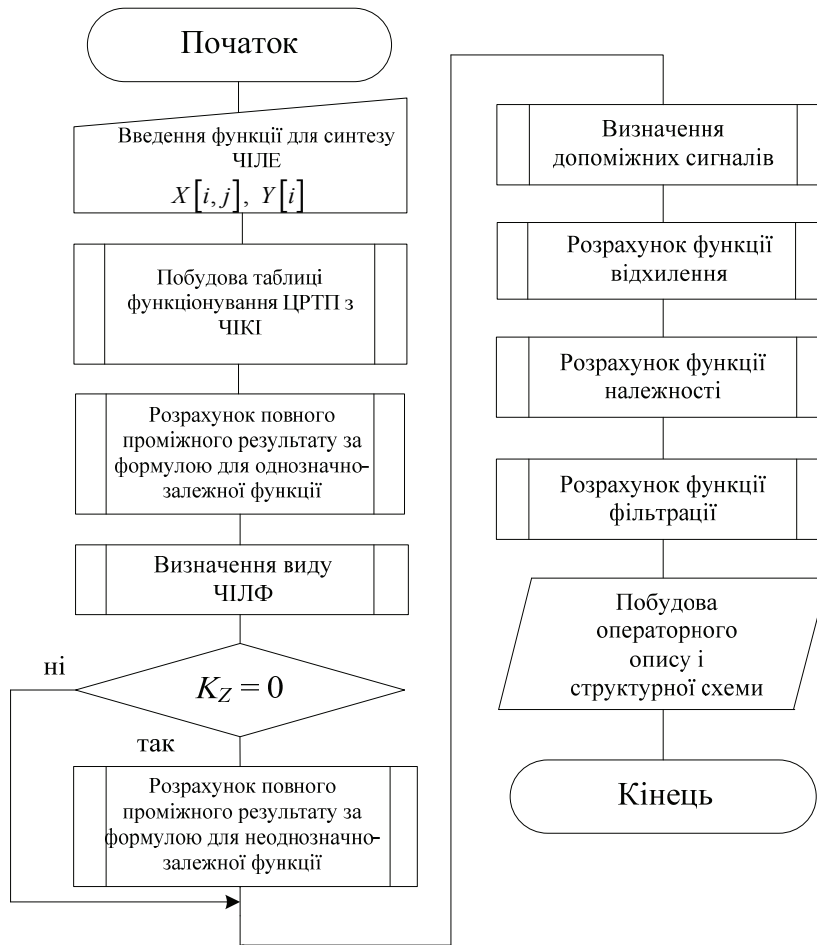


Рис. 5. Структурна схема програми синтезу ЦРТП з ЧІКІ

### Розробка програми синтезу цифрових радіотехнічних пристроїв з частотно-імпульсним кодуванням інформації

Програма написана на мові програмування Java, 8 версія JRE SE. Для розробки програми використано платформу JavaFX 2.2. Фреймворк для автоматичного компонування jar пакету — Apache Maven. Додаткові бібліотеки: GSON, JFXtras.

Програму можна розділити на 3 структурних компоненти відповідно до моделі MVC.

Точка входу в програму знаходиться в класі SchemeGenerator який розширює абстрактний клас JavaFX Application. В цьому класі завантажуються графічні компоненти з rootView.fxml файлу, який необхідний для створення та ініціалізації головної сцени аплікації та ініціалізується singleton об'єкт ApplicationContext, який відіграє роль базового контексту цієї аплікації.

Клас ApplicationContext розширяє абстрактний клас BaseModel. В ApplicationContext зберігається посилання на кореневу панель сцени, ініціалізується MainScreenController та зберігається посилання на цей контролер, приватне поле класу контексту Map <String, IBaseModel> models служить контейнером моделей даних цієї аплікації.

За ініціалізації MainScreenController з FXML ресурсів завантажуються графічний інтерфейс першого екрану рис. 6. InputController реєструє всі модулі, що спостерігають за подіями графічних компонентів, та зв'язує їх з моделлю даних InputModel.

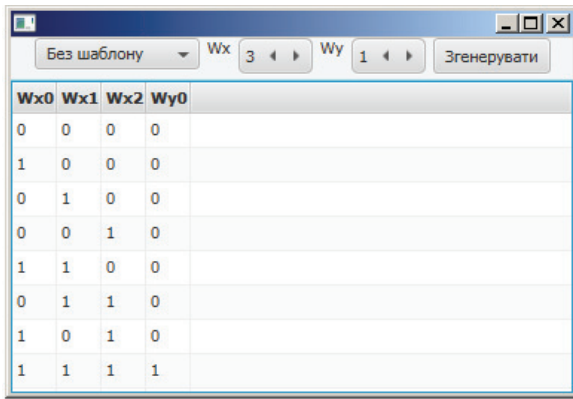


Рис. 6. Вікно програми для збору вхідних даних програми

В цьому вікні користувач заповнює таблицю функціонування пристрою. Значення вхідних сигналів в таблиці функціонування автоматично заповнюються відповідно алгоритму, що наведений на рис. 5. Значення вхідних сигналів заповнюються користувачем. Джерелами значень, якими заповнюється таблиця функціонування, є списки об'єктів типів `InputTableRow` та `TableColumn`. Зміст кожного об'єкту `TableColumn` визначається відповідним частотним набором. Діаграма класів, які реалізують збір та первинну обробку вхідних даних, показана на рис. 6.

`InputTableRow` доповнюється значенням  $z_i$  ППР,

Для визначення того, чи є функція однозначно залежною або неоднозначно залежною, об'єкт

яке обчислюється за виразом (1).

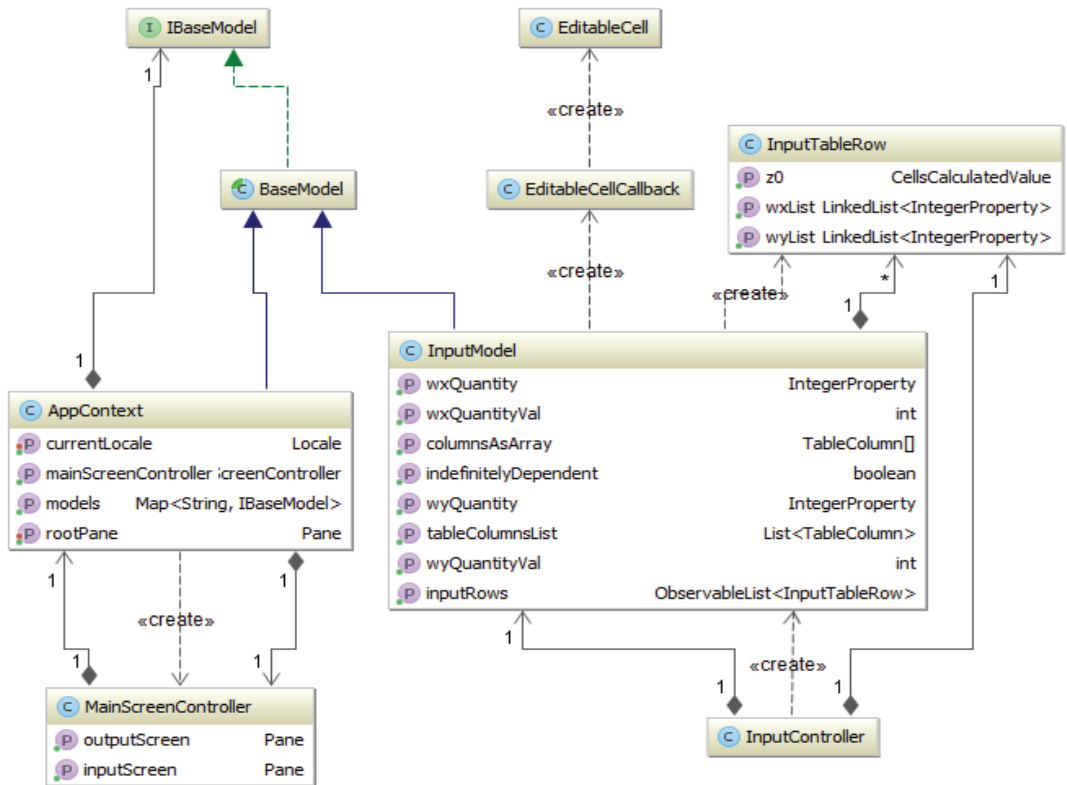


Рис. 7. Діаграма класів, що реалізують збір та первинну обробку вхідних даних

Визначення виду логічної функції реалізоване у методі `checkForDependency()` класу `InputModel`. Функція `checkForDependency()` створює таблицю, де ключовим полем є значення ППР, і значення функції  $y$ . Якщо під час заповнення таблиці значення ключового поля будуть однакові, а значення  $y$  різні, то логічна функція неоднозначно залежна. В результаті її виконання властивості об'єкта `isIndefinitelyDependent` встановлюється значення, що відповідає виду логічної функції. Тип частотно-логічної  $K_z$  функції визначається у відповідності до виразу (3).

Допоміжні сигнали у програмі зображуються об'єктами типу `CellsCalculatedValue`. Необхідно виконати порівняння ППР  $z_i$  та відповідного значення  $u_i$  для кожного рядка, в результаті чого можна визначити значення функції відхилення відповідно до виразу (4).

Для всіх частотних наборів визначаються значення  $\Delta_i$ , із загального переліку значень  $\Delta$  вибираються всі різні значення. Кількість різних значень  $\Delta_i$  визначає кількість допоміжних сигналів. Для забезпечення можливості виділення всіх унікальних значень  $\Delta_i$  клас `CellsCalculatedValue` реалізує інтерфейс `Comparable`, для цього необхідно реалізувати метод `compareTo()` та перевизначити метод `equals()`. Визначений набір об'єктів буде переліком значень допоміжних сигналів. Діаграма класу `CellsCalculatedValue` подана на рис. 8.



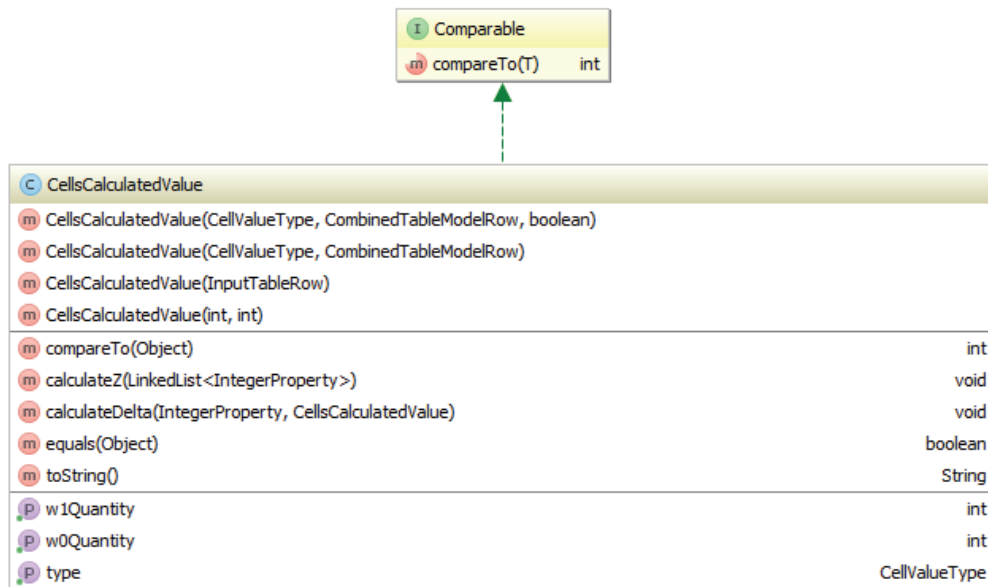


Рис. 8. Діаграма класу CellsCalculatedValue

### Визначення функції належності

Для кожного допоміжного сигналу за виразом (5) визначаються функції належності.

Значення ППР,  $\Delta_i$  допоміжних сигналів та функцій належності у програмі подано за допомогою моделі даних CombinedTableModel, структурна діаграма якої показана на рис. 9.

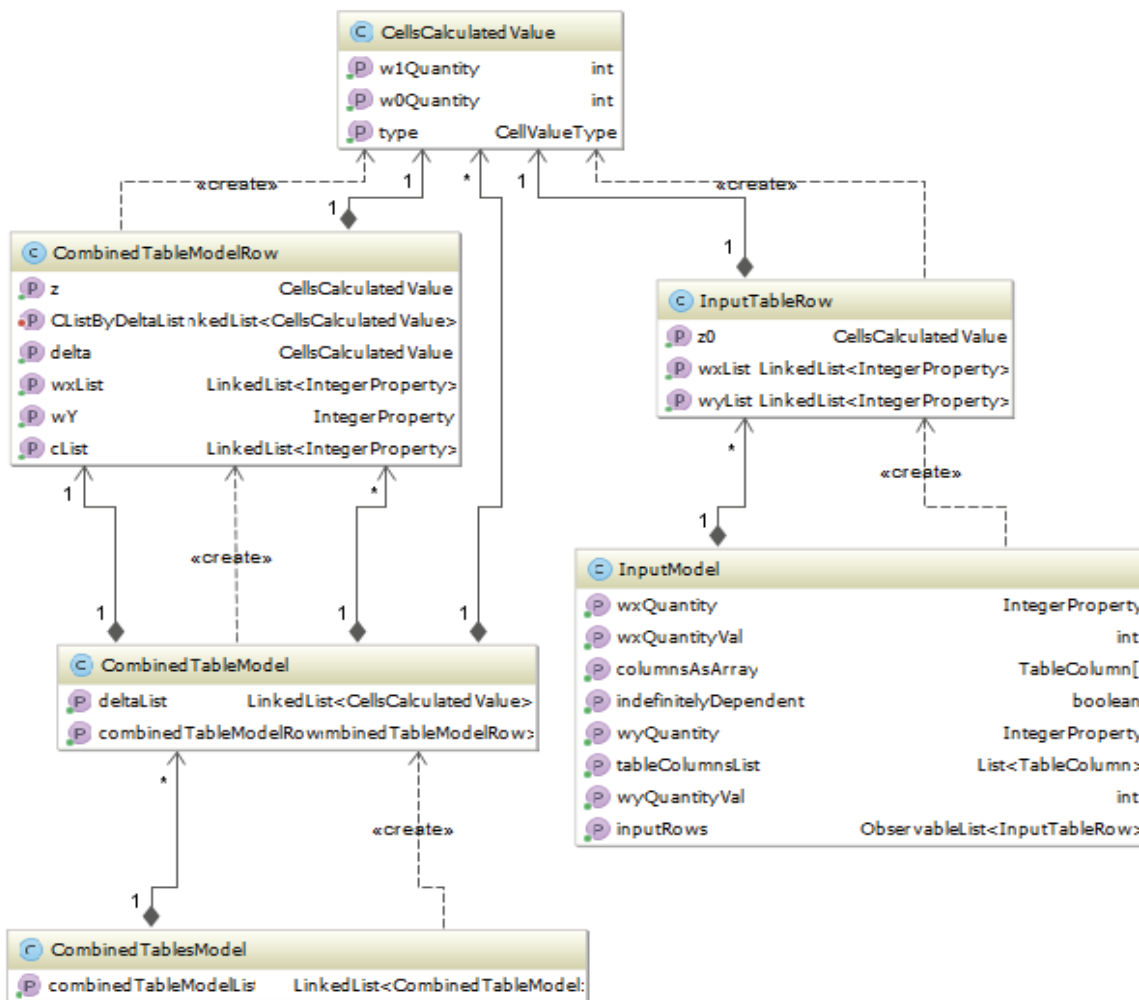


Рис. 9. Діаграма зв'язку класів, необхідних для створення комбінованої таблиці

Конструктор класу CombinedTableModel в якості аргументу приймає об'єкт типу InputModel, в якому зберігається заповнена таблиця функціонування синтезованого елемента.

Для визначення функції фільтрації  $C_j$  формуємо таблиці відповідності, які у програмі подані класом CorrespondenceTable. Логіка, що міститься в цьому класі, в циклі проходить всі рядки комбінованої таблиці та на базі даних, що знаходиться в цих об'єктах, створює екземпляр класу CorrespondenceTableRow, передаючи в конструктор об'єкт класу рядка суміщеної таблиці. Після створення об'єкта необхідно визначити значення функції фільтрації, для чого викликається метод fillSignals в об'єкта рядка. Таким чином, для цього рядка є всі необхідні параметри і можна визначити значення функції фільтрації відповідно до виразу (5).

В результаті виконання цього алгоритму створюється перелік таблиць відповідності для кожного стовпчика  $C_j$ .

### Складання операторного опису

На базі узагальненої структури операторного опису розроблено структуру OperatorEquationModel. Для подання складових частин операторного опису її розділено на три структурних блоки InputEquation, OperatorEquation, OutputEquation. Діаграма зв'язку класів необхідних для складання операторного опису подана на рис. 10.

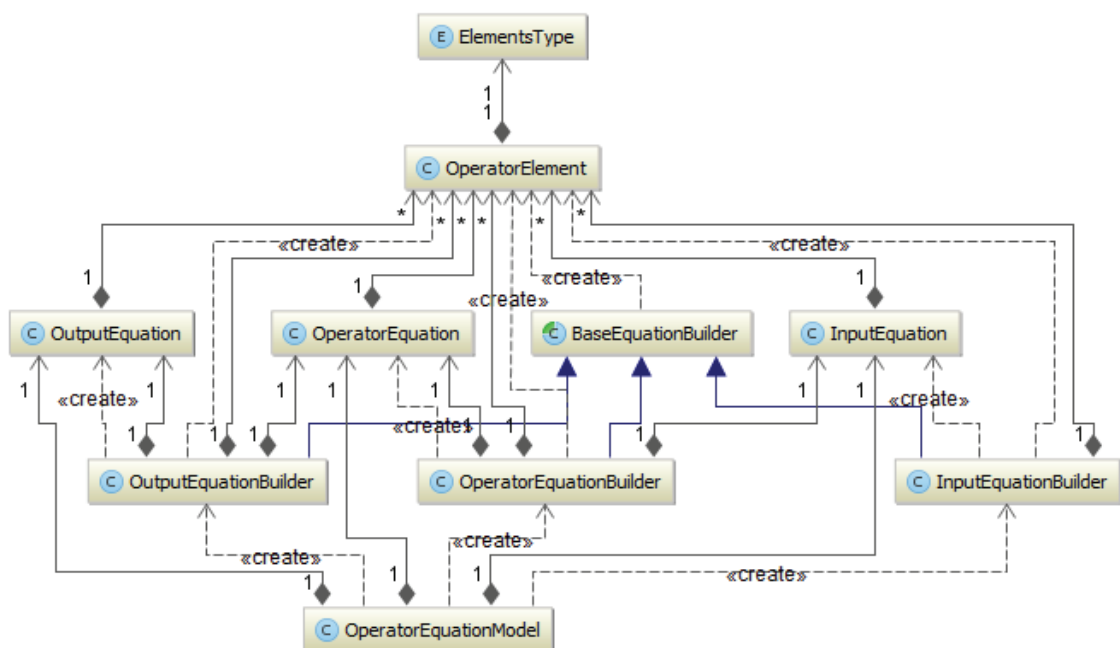


Рис. 10. Діаграма зв'язку класів, необхідних для складання операторного опису

Операторне рівняння відіграє роль об'єкта, який може бути інтерпретований як літеральне подання графу схеми ЦРТП з ЧІКІ. Для розв'язання задачі утворення зв'язків між елементами та подання сутності базового логічного елемента розроблено клас OperatorElement, у якого є поля для подання вхідних та вихідних зв'язків елемента графу, його типу та значення. Алгоритми створення об'єктів типу OperatorElement містяться в класах InputEquationBuilder, OperatorEquationBuilder, OutputEquationBuilder.

Операторне рівняння відіграє роль об'єкта, який може бути інтерпретований як літеральне подання графу схеми ЦРТП з ЧІКІ. Для розв'язання задачі утворення зв'язків між елементами та подання сутності базового логічного елемента розроблено клас OperatorElement, у якого є поля для подання вхідних та вихідних зв'язків елемента графу, його типу та значення. Алгоритми створення об'єктів типу OperatorElement містяться в класах InputEquationBuilder, OperatorEquationBuilder, OutputEquationBuilder.

Структурна схема ЦРТП з ЧІКІ будується на базі узагальненої структурної схеми рис. 4. Для графічного відображення підсумкової структурної схеми елемента розроблено низку компонентів: підсумкова модель даних OutputModel, її контролер OutputController та ряд класів для креслення графіки схеми. На рис. 11 подана діаграма зв'язків між класами, що необхідні для побудови екрана, який відображає синтезовану схему пристрою.



OperatorElementWrapper містить інформацію про елемент: координати елемента на Canvas, його тип, значення та зв'язки. Клас SchemeImageBuilder призначений для керування процесом створення графічного відображення структурної схеми, який послідовно проходить списки елементів графу з OperatorEquationModel, на їх базі створює об'єкти класу OperatorElementWrapper.

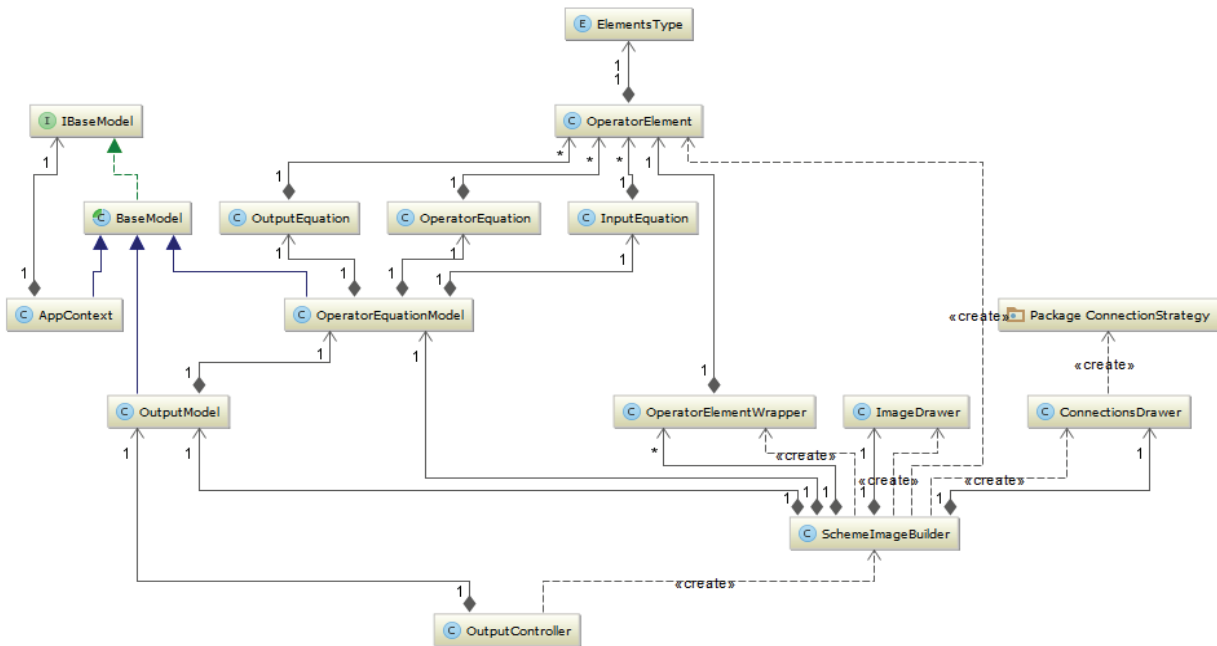


Рис. 11. Діаграма класів, необхідних для побудови схеми

Класи ImageDrawer та ConnectionDrawer безпосередньо працюють з Canvas, ImageDrawer призначений для нанесення на Canvas графічного зображення елементів схеми, залежно від їх типу та значення. ConnectionDrawer призначений для нанесення на Canvas графічного зображення між елементами. Для коректного відображення ліній зв'язків реалізовано ряд класів стратегій, що знаходяться в пакеті connectionStrategy, які на базі точок, що необхідно зв'язати, та попередньо нанесених елементів, приймають рішення щодо прокладання оптимального маршруту. Для виведення схеми передбачено дві опції: збереження схеми у графічний файл та друк на принтері.

На рис. 12 показано вікно програми з результатом роботи програми. На об'єкті Canvas зображено структурну схему пристрою, яку можна експортувати до графічного файлу або роздрукувати.

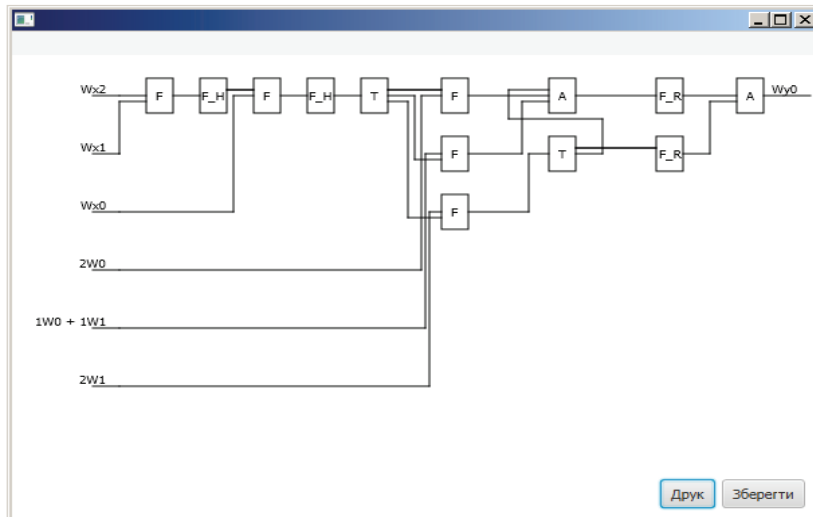


Рис. 12. Вікно з результатами роботи програми

## Висновки

1. Розроблено алгоритм побудови таблиці функціонування цифрових радіотехнічних пристроїв з частотно-імпульсним методом кодування інформації, алгоритми розрахунку повного проміжного

результату та визначення виду частотно-імпульсної логічної функції.

2. Розроблено структурні схеми алгоритмів побудови функції відхилення, функції належності та функції фільтрації.

3. Розроблено алгоритм та програму автоматизованого синтезу структурних схем цифрових радіотехнічних пристроїв з частотно-імпульсним методом кодування інформації.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

- [1] В. М. Кичак, *Синтез частотно-імпульсних елементів цифрової техніки*. Вінниця, Україна: УНІВЕРСУМ-Вінниця, 2005, 291 с.
- [2] В. М. Кичак, та О. О. Семенова, *Радіочастотні та широтно-імпульсні елементи цифрової техніки*. Вінниця, Україна: УНІВЕРСУМ-Вінниця, 2008, 162 с.
- [3] А. С. Винницький, «Частотная модуляция и цифровая обработка информации,» *Радиотехника*, № 6, с. 83-88, 1987.
- [4] V. M. Kychak., M. D. Huz, and D. S. Hromovyi, “Influence of Critical Current and Parameters of Josephson Junction on Frequency Stability of Oscillator,” *Information and Telecommunication Sciences*, vol. 7, no. 2, pp. 54-61, 2016.

Рекомендована кафедрою телекомунікаційних систем і телебачення ВНТУ

Стаття надійшла до редакції 10.01.2018

**Кичак Василь Мартинович** — д-р техн. наук, професор, професор кафедри телекомунікаційних систем і телебачення, e-mail: vmkychak@gmail.com ;

**Гузь Максим Дмитрович** — аспірант кафедри телекомунікаційних систем і телебачення, e-mail: huz.maksim@gmail.com ;

**Кичак Володимир Васильович** — канд. техн. наук, доцент кафедри телекомунікаційних систем і телебачення;

**Олійник Василь Валерійович** — студент факультету інфокомунікацій, радіотехніки та наносистем.

Вінницький національний технічний університет, Вінниця

**V. M. Kychak<sup>1</sup>**  
**M. D. Huz<sup>1</sup>**  
**V. V. Kychak<sup>1</sup>**  
**V. V. Oliinyk<sup>1</sup>**

## Method of Automatic Synthesis for Digital Radio Engineering Devices with Pulse-Frequency Encoding

<sup>1</sup>Vinnytsia National Technical University

*The method of synthesis of digital radio engineering devices with pulse-frequency encoding of information is considered in the work. The distinction of the method is the basic elements for synthesis being physical circuits that implement elementary functions.*

*An approach based on the transformation of elementary operations performed by physical schemes into elementary operators that implement pulse-frequency logic functions is proposed for the synthesis of such devices.*

*A methodology and algorithm for constructing a table for the functioning of digital radio devices with pulse-frequency encoding of information for a binary structural alphabet is developed. Analytical expressions for calculating of complete intermediate results for uniquely and ambiguously dependent pulse-frequency logic functions are proposed.*

*To calculate the number and values of auxiliary signals used for constructing digital devices it is proposed to calculate the deviation functions which represent the difference between the total intermediate results and the corresponding value of pulse-frequency logic functions. The number of different values of the deviation functions corresponds to the number of auxiliary signals needed to implement the device, and the various values of the deviation function are the values of auxiliary signals.*

*An algorithm for constructing a membership function is developed which processes the value of the deviation function and provides filtering.*

*The method of drawing up the operator description is developed, on the basis which the structure diagram of digital radio engineering devices with frequency-pulse encoding of information is built.*

*A program for the structural synthesis of digital radio devices with frequency pulse encoding of information is developed using the proposed algorithms. The program is written in Java programming language, version 8 JRE SE. The JavaFX 2.2 platform was used to develop the program. The framework for automatic jar package layout is Apache Maven. Additional libraries are GSON, JFXtras.*

**Keywords:** pulse-frequency encoding, full intermediate result, deviation function, membership function, pulse-frequency logic function.

**Kychak Vasyl M.** — Dr. Sc. (Eng.), Professor of the Chair of Telecommunication Systems and Television, e-mail: vmkychak@gmail.com ;

**Huz Maksym D.** — Post-Graduate Student of the Chair of Telecommunication Systems and Television, e-mail: huz.maksim@gmail.com ;

**Kychak Volodymyr V.** — Cand. Sc. (Eng.), Assistant Professor of the Chair of Telecommunication Systems and Television;

**Oliinyk Vasyl V.** — Student of the Department of Informatics, Radiotechnics and Nanosystems

**В. М. Кичак<sup>1</sup>**  
**М. Д. Гузь<sup>1</sup>**  
**В. В. Кичак<sup>1</sup>**  
**В. В. Олейник<sup>1</sup>**

## Метод автоматизированного синтеза цифровых радиотехнических устройств с частотно-импульсным кодированием информации

<sup>1</sup>Вінницький національний технічний університет

*Рассмотрен метод синтеза цифровых радиотехнических устройств с частотно-импульсным кодированием информации. Отличие метода состоит в том, что базовыми элементами для синтеза являются физические схемы, реализующие элементарные функции.*

*Для синтеза таких устройств предложен подход, базирующийся на исполняемых физическими схемами преобразовании элементарных операций в элементарные операторы, реализующие частотно-импульсные логические функции.*

*Разработана методика и алгоритм построения таблицы функционирования цифровых радиотехнических устройств с частотно-импульсным кодированием информации. Предложены аналитические выражения расчета полных промежуточных результатов для однозначно и неоднозначно зависимых частотно-импульсных логических функций.*

*С целью определения количества и значений вспомогательных сигналов, предложено рассчитывать функции отклонения, которые представляют собой разницу полных промежуточных результатов и соответствующего значения частотно-импульсных логических функций. Количество разных значений функций отклонения соответствует количеству вспомогательных сигналов, необходимых для реализации устройства, а разные значения функции отклонения являются значениями вспомогательных сигналов.*

*Предложен алгоритм построения функции принадлежности, который обрабатывает значение функции отклонения и обеспечивает фильтрацию.*

*Разработана методика составления операторного представления, на базе которого строится структурная схема.*

*С использованием предложенных алгоритмов разработана программа структурного синтеза цифровых радиотехнических устройств с частотно-импульсным кодированием информации. Программа написана на языке программирования Java, 8 версия JRE SE. Для разработки программы использовано платформу JavaFX 2.2. Фреймворк для автоматической компоновки jar пакета — Apache Maven. Дополнительные библиотеки: GSON, JFXtras.*

**Ключевые слова:** частотно-импульсное кодирование, полный промежуточный результат, функция отклонения, функция принадлежности, частотно-импульсная логическая функция.

**Кичак Василий Мартынович** — д-р техн. наук, профессор кафедры телекоммуникационных систем и телевидения, e-mail: vmkychak@gmail.com ;

**Гузь Максим Дмитриевич** — аспирант кафедры телекоммуникационных систем и телевидения e-mail: huz.maksim@gmail.com ;

**Кичак Владимир Васильевич** — канд. техн. наук, доцент кафедры телекоммуникационных систем и телевидения ;

**Олейник Василий Валерьевич** — студент факультета инфокоммуникаций, радиотехники и наносистем