

1. ITU Blue Book, Recommendation Q.24, Multi-Frequency Push-Button Signal Reception, Geneva, 1989.
2. *M. Felder, J. Mason, B. Evals*, "Efficient Dual-Tone Multi-Frequency Detection using the Non-Uniform Discrete Fourier Transform," IEEE Signal Processing Letters, vol. 5, pp.160-163, July 1998.
3. *Amey A. Deosthali, Shawn R. McCaslin, Brian L. Evans*, A Low-Complexity ITU-Compliant Dual Tone Multiple Frequency Detector, IEEE Trans. on Signal Processing, vol. 48, no. 3, pp. 911-916, Mar, 2000.
4. Various DTMF detections; Term project report for ME780 Digital signal analysis for mechanical systems, Ta-young Gabriel Choi.

*Поступила 28.02.2011 р.*

УДК 621.3

О.М. Колодчак, НУ "Львівська політехніка", ІКТА, каф. ЕОМ, м. Львів

## **ПРОЕКТУВАННЯ ОПТИМАЛЬНИХ СТРУКТУР КОНВЕЄРНИХ ПРИБОРІВ**

Запропоновано опис нового підходу до проектування оптимальних структур конвеєрних пристроїв, заснованого на теорії генетичних алгоритмів.

Предложено описание нового подхода к проектированию оптимальных структур конвейерных устройств, основанного на теории генетических алгоритмов.

This article proposes a new approach to create optimized structures of pipelined devices device using genetic algorithms.

### **Вступ**

Проектування комп'ютерних пристроїв є досить складним процесом. По-перше, існує багато підходів, алгоритмів для побудови структур комп'ютерних засобів та пошуку найоптимальнішої з множини можливих структур. По-друге, є багато варіантів відображення функціональних операторів в комбінаційних схемах нвіс, тобто треба робити вибір між апаратними затратами та швидкодією комп'ютерних засобів, або отримати оптимальну структуру, яка би була задовільною по цих критеріях в конкретних задачах. Звідси впливають такі особливості проектування комп'ютерних пристроїв як складність та трудоемність, необхідність пошуку оптимальних рішень на кожному етапі проектування, необхідність скорочення термінів проектування. Вказані особливості вимагають створення як нових методик проектування, так і систем автоматизованого проектування (САПР).

Існує багато алгоритмів пошуку оптимальної структури комп'ютерних пристроїв. Самий простий з них – перебір всіх можливих структур. Такий підхід є найліпшим при невеликій кількості оцінюваних структур. А коли множина оцінюваних структур буде вміщувати  $2^n$  структур, де  $n$  – кількість елементів структури, яка може сягати сотень або тисяч одиниць, то така задача буде складно реалізовуватись на практиці. Тому зараз проводяться дослідження інших методів та алгоритмів пошуку оптимального рішення поставленої задачі, зокрема симплекс-метод, жадібний метод, алгоритм з оберненою матрицею, алгоритм Дейкстри [2].

Одним з таких методів є еволюційний, який базується на еволюційній теорії та техніці підстановки необхідних елементів бібліотеки. Для вирішення даної задачі пропонується використати теорію генетичних алгоритмів, яка відображає біологічну концепцію “виживання найбільш пристосованого”[4]. Робота генетичного алгоритму починається з великої кількості потенційних рішень (в нашому випадку – це структури, сформовані з елементів бібліотеки), серед яких вибираються найкращі, які комбінуються і знову подаються на вхід алгоритму. Через певну кількість “поколінь” якість рішень підвищується. Таким чином, задача, що вимагає роки обчислень на суперкомп'ютері, може бути вирішена за декілька хвилин.

В статті описується новий підхід до проектування комп'ютерних пристроїв, який може бути використаний при створенні САПР, що максимально автоматизували би процес проектування.

### **Постановка задачі**

Розробити новий підхід до проектування конвеєрних пристроїв, використовуючи теорію генетичних алгоритмів. Оцінити ефективність розробленого підходу.

### **Традиційний підхід до проектування комп'ютерних пристроїв**

Проектування представляє собою багаторівневий, багатокроковий та ітераційний, з поверненням назад і переглядом раніше прийнятих рішень комплексу проектних робіт, як правило, включає в себе теоретичні та експериментальні дослідження, розрахунки та конструювання.

Функціональна декомпозиція проекту є стратегією проектування. Для системи в цілому та її блоків використовується концепція “чорного ящика”. Для “чорного ящика” розробляється функціональна специфікація, яка включає зовнішній опис блоку (входи і виходи) та внутрішній опис – функцію або алгоритм:

$$F = \Phi(X,t),$$

де  $X$  – вектор вхідних величин,  $F$  – вектор вихідних величин,  $t$  – час. При декомпозиції функції  $\Phi$  розбивається на більш прості функції  $\Phi_1 \dots \Phi_k$ , між якими повинні бути встановлені відповідні зв'язки, згідно прийнятого алгоритму реалізації функції  $\Phi$ . В результаті розбиття отримується структура.

Перехід від функції до структури є синтезом. Вибір найліпшого варіанту здійснюється згідно результатів аналізу, коли перевіряється правильність роботи та деякі показники, які характеризують пристрій. Процедури синтезу та аналізу постійно повторюються.

Традиційні етапи проектування :

1. системне проектування;
2. структурно-алгоритмічне проектування;
3. функціонально-логічне проектування;
4. конструкторсько-технологічне проектування.

На етапі системного проектування визначається архітектура, склад компонентів і основні характеристики системи, вибирається елементна база для її побудови. Це найбільш складний етап, оскільки його складно формалізувати та автоматизувати.

Під час структурно-алгоритмічного проектування визначаються алгоритми функціонування апаратних і програмних компонентів системи.

На етапі функціонально-логічного проектування розробляються функціональні та принципові електричні схеми, програми, готуються тестові та контрольні дані.

На конструкторському етапі виконується прив'язка елементів проекту до конструкторських елементів.

Вище описані загальні етапи проектування комп'ютерних систем. Кожний з етапів розбивається на підетапи.

На кожному етапі проектування може бути виявлена помилковість або неоптимальність вибраного раніше варіанту реалізації або прийнятого рішення. Така ситуація призводить до повернення і перегляду попереднього рішення. Розробка сучасних обчислювальних систем, які включають як мікропроцесорні рішення, так і НВІС з програмованою структурою, неможлива без застосування САПР. Враховуючи можливості сучасних САПР проектування можна рахувати закінченим після верифікації проекту в цілому, коли закінчене відлагодження готового виробу. Проект або його окремі фрагменти можуть включати аналогові, аналогово-цифрові та/або цифро аналогові елементи, будуватись на основі дискретних (цифрових) компонентів або спиратись на вбудовані мікропроцесорні засоби. Звідси випливає велика кількість варіантів проектування, які в сучасних технологіях часто називають потоком проектування (Design Flow). Наступним фактором є вибір технічної бази для реалізації фрагментів проекту, а також технічного способу реалізації самого проекту. Як правило, один і той же пристрій може бути реалізовано різними способами.

Від підходу до проектування, на основі якого створена САПР, залежить швидкість проектування і відповідно терміни випуску продукції, тобто час від специфікації проекту до випуску перших екземплярів (Time-to-Market). Для збільшення ефективності проектування комп'ютерних пристроїв треба знайти такий підхід до проектування, який максимально спрощував би роботу проектувальникам та прискорював сам процес проектування.

## **Опис підходу до проектування оптимальних структур конвеєрних пристроїв з використанням генетичних алгоритмів**

Як вже було сказано раніше, при великій кількості можливих реалізацій структури комп'ютерних пристроїв, процес проектування є досить громіздким та тривалим в часі. При цьому треба зробити наголошення, що потрібна ще й можливість оптимізації структури. В даному випадку задача оптимізації структури є індивідуальною задачею оптимізації, тобто для неї задаються конкретні вхідні дані, на яких може бути отримано певний конкретний розв'язок.

Даний підхід проектування передбачає використання генетичного алгоритму (ГА), який відіграє роль як механізму оптимізації так і механізму створення структур комп'ютерних пристроїв, та застосування бібліотеки елементів (рис.1).

Бібліотека елементів містить множину  $N$  базових елементів, які відповідно можуть мати певну кількість (від 1 до  $n$ ) різних реалізацій. Елементи описані на одній з мов опису апаратних засобів, наприклад, VHDL. Крім того, тут міститься інформація про кожний елемент бібліотеки (його параметри), яка обробляється програмою керування бібліотекою (слідкує, щоб не виникло нестикувань між елементами, за підбором елементів відносно параметрів та ін.), коли створюється множина структур пристрою на основі цих елементів за допомогою ГА.

Як видно з рис. 1, підхід передбачає наступні етапи:

- спочатку формуються вхідні дані, які описують структуру пристрою (кількість блоків, алгоритми їх функціонування, розрядності шин та ін.), вимоги щодо її проектування (наприклад, мінімізація апаратних затрат) та параметри ГА (розмір популяції, кількість батьківських пар, кількість поколінь – ітерацій та ін.);
- далі починає працювати ГА: випадковим чином формується деяка початкова множина структур пристрою (тобто автоматично генеруються відповідно до вхідних даних та за допомогою програми керування бібліотекою, яка оперує інформацією про кожний елемент, без втручання проектувальника) та оцінюється кожна з них (в бібліотеці міститься інформація про елементи, з яких вони формуються); наприклад, пристрій складається з 4 блоків, перший блок має 10 реалізацій в бібліотеці (1(1), 1(2),..., 1(10)), другий – 5, третій – 2, четвертий – 7; початкова множина (один з параметрів ГА) структур – 4;
- далі виконуються послідовно операції кросовера та мутації, до структур з гіршими параметрами відносно критеріїв, поставлених до структури; структура з ліпшими показниками зберігається в деякому буфері збереження оптимальних структур;
- наступним кроком є перевірка параметра ГА, який відповідає за кількість ітерацій – кількість поколінь: якщо він показує, що роботу ГА завершено, то проводиться остаточна оцінка структур в буфері збереження

оптимальних структур, та вибирається структура з ліпшими показниками, які відповідають критеріям, поставленим до структури з самого початку у вхідних даних, якщо ж ні, то робота ГА повертається до формування нової множини структур.



Рис.1 – Блок-схема основних етапів проектування згідно розробленого підходу

Отже, таким чином можна отримати оптимізовану структуру комп'ютерного пристрою, яка відповідатиме поставленим критеріям, але не завжди є ідеальною (тобто може бути одним з варіантів реалізації структури згідно вхідних умов).

### Оцінка ефективності розробленого підходу

Для оцінки ефективності застосування вищеописаного підходу до проектування, потрібно застосувати його до певних реальних проектів. Отже, була створена модель системи проектування (принцип роботи - див.рис.1), яка застосовується для проектування конвеєрних пристроїв. Система повинна оцінити декілька алгоритмів функціонування, щоб забезпечити побудову ефективнішої структури конвеєрного пристрою з усіх можливих варіантів, враховуючи умови, які задає проектувальник, крім того видати характеристики всіх варіантів пристрою. Для прикладу було розглянуто проектування спеціалізованого пристрою, який реалізує функцію сортування восьми 4-розрядних чисел .

Постановка задачі: побудувати ефективну структури спеціалізованого пристрою сортування восьми 4-розрядних чисел, оцінити такт його роботи ( $T$ ), визначити місця розділення ярусів регістрами (вектор розміщення регістрів,  $FopT$ ) для забезпечення необхідного часу виконання функції сортування при мінімальній кількості обладнання ( $W_k$ ). А також оцінити ефективність використання обладнання отриманої структури ( $E_k$ ).

Отже, отримавши таку задачу, спочатку модель системи оцінювала алгоритми, кожний з яких вже був описаний в бібліотеці. В даному випадку алгоритмів було два. Структура базового елемента порівняння для реалізації алгоритмів, яка була вибрана моделлю з бібліотеки елементів, показана на рис.3.

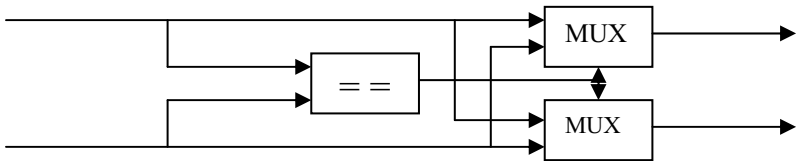


Рис.3 - Структура базового елемента порівняння, вибрана з бібліотеки елементів

Оцінюється спочатку перший алгоритм. Якщо даний алгоритм представити у вигляді потокового графу, то видно, що даний алгоритм сортування має 13 ярусів (рис.4). Отже, найбільша довжина конвеєра, який реалізує даний алгоритм, буде рівнятися тринадцятьом. Наприклад, для апаратної реалізації такого операційного пристрою береться програмована логічна інтегральна мікросхема Spartan XCS30XL фірми XILINX. За одиницю апаратних затрат рахується один вентиль. Таким чином можна оцінити апаратні затрати кожного ярусу конвеєра та відповідні затримки на них. Затримка на конвеєрному регістрі дорівнює 9 нс, а апаратні затрати на 136

нього – 256 вентилів.

Результати оцінки занесені в таблицю 1. З результатів видно, що модель системи, основою якої є генетичний алгоритм, проектує структури досить непогано, оскільки час обробки масиву чисел є близький до заданого в умові. Причому затрати на обладнання є меншими – це обумовлено обмеженням щодо кількості регістрів. Провівши зміну параметрів генетичного алгоритму, можна зробити висновок, що зі збільшенням популяції та кількості поколінь (ітерацій алгоритму або часу його виконання), результат наближається до ідеального, який в даному випадку видав би повний перебір всіх можливих структур.

Розглянемо тепер інший алгоритм сортування. Поточковий граф алгоритму зображений на рисунку 5. Цей алгоритм розрахований на сортування восьми 4-розрядних чисел. Результати оцінки показані в табл. 2.

Отже, як видно при застосуванні моделі системи проектування було отримано декілька варіантів реалізації алгоритмів сортування, результуючою стала структура другого алгоритму, приклад 1 (див. в таблиці 2 виділене шрифтом), оскільки при його реалізації будуть менші апаратні затрати, ефективність використання обладнання є більшою.

Таблиця 1

Параметри структури	Варіанти реалізації алгоритму	
	Приклади з різними вхідними даними	
	1	2
1. Вектор розміщення регістрів, $F_{opt}$	11010101011	111111111111
2. Такт конвеєра, $T_{нс}$	29	19
3. Апаратні затрати на конвеєр, $W_k$ вентилів	10820	12100
4. Ефективність використання обладнання конвеєра, $E_k$	0,1 699	0,174

Таблиця 2

Параметри структури	Варіанти реалізації алгоритму	
	Приклади з різними вхідними даними	
	1	2
1. Вектор розміщення регістрів, $F_{opt}$	<b>01010101</b>	10110111
2. Такт конвеєра, $T_{нс}$	<b>29</b>	29
3. Апаратні затрати на конвеєр, $W_k$ вентилів	<b>8580</b>	9092
4. Ефективність використання обладнання конвеєра, $E_k$	<b>0,2922</b>	0,2333

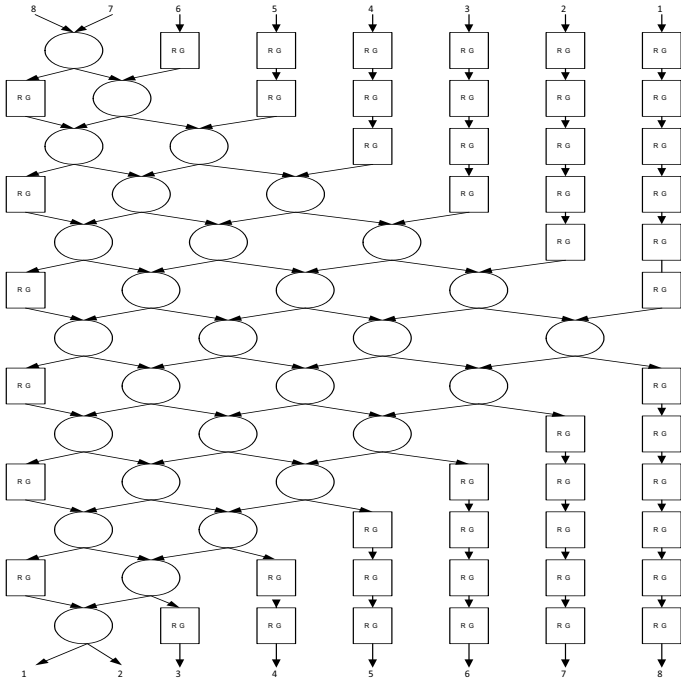


Рис.4 - Граф першого алгоритму сортування

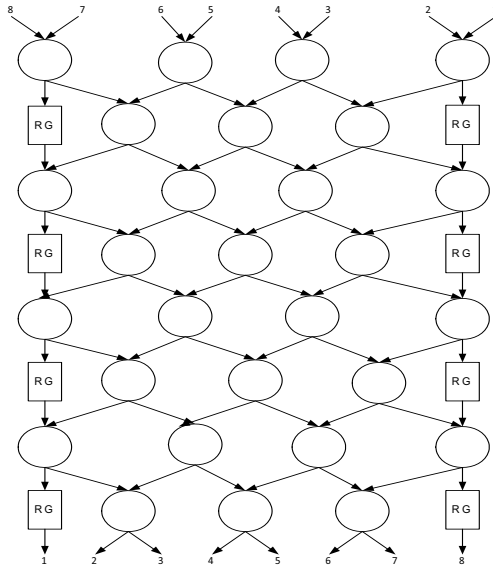


Рис.5 - Граф другого алгоритму сортування



Таким чином модель системи дає можливість досліджувати структури пристроїв та їх параметри, і вибирати ту структуру, яка найбільше відповідає поставленій задачі, виконуючи це автоматично без втручання проектувальника, що значно полегшує та суттєво прискорює процес проектування.

## **Висновки**

Запропонований підхід до проектування оптимальних структур конвеєрних пристроїв з використанням генетичних алгоритмів дозволяє значно спростити та прискорити процес проектування. Основними перевагами є можливість автоматичної генерації структур пристроїв, які проектуються, в залежності від вимог проектувальника, та отримання нових структур за допомогою генетичного алгоритму, що дозволяє розширити простір пошуку оптимальної структури. При цьому, робота проектувальника зводиться до формування початкових та кінцевих параметрів проектованої системи (апаратні та часові параметри, алгоритм функціонування блоків системи та інші) та задання режиму проектування (налаштування генетичного алгоритму, умови вибору елементів з бібліотеки та інші). Застосування даного підходу, при проектуванні пристроїв з конвеєрною архітектурою, показало його високу ефективність, що підтверджує доцільність його застосування для проектування комп'ютерних пристроїв інших архітектур.

1. *Глибовець М.М.* Основи комп'ютерних алгоритмів. - К.: Вид.дім "КМ Академія", 2003.- 452с.:іл. – Бібліогр.: с.428–432.
2. *Исаев Сергей.* Популярно о генетических алгоритмах. web: <http://www.chat.ru/~saisa/index.html>.
3. *Грушвицкий Р., Мурсаев А., Угрюмов Е.* Проектирование систем на микросхемах программируемой логики. – СПб. 6: БХВ-Петербург, 2002. – 608с.:ил.
4. *Норенков И.П., Маничев В.Б.* Основы теории и проектирования САПР: Учеб. Для втузов по спец."Вычислительные маш., компл., сист. и сети". – М.: Высш.шк., 1990. – 335 с.

*Поступила 04.04.2011 р.*