

УДК 658:004

Шадрін М.А.¹, Зеленьова І.Я.², Голуб Т.В.²

¹ студ. гр. КНТ-612м НУ «Запорізька політехніка»

² канд. техн. наук, доц. НУ «Запорізька політехніка»

ДОСЛІДЖЕННЯ СХЕМ ЗГОРТКОВОГО КОДУВАННЯ

Вбудовані системи являють собою важливі складові частини сучасних пристроїв та зазвичай призначені для виконання певної обмеженої кількості функцій. Обчислювальний процес здійснюється апаратними та програмним засобами. Різноманітність варіантів реалізації вбудованої системи буває дуже широкою. Відповідно, актуальним стає питання компактного розташування логічних схем функціональних блоків вбудованої системи в межах площі кристала ПЛІС. Чим компактніше розміщена вбудована система на кристалі, тим ліпше основні характеристики, зокрема, порібна площа кристала, а це веде до зменшення часу поширення сигналу та споживаної енергії.

Згортковий код використовується для коректування помилок. Якщо при передачі у каналах зв'язку згорткового коду з'являються помилки, то при декодуванні можна отримати вихідну коректну послідовність біт. Це досягається завдяки надлишковості при кодуванні згорткового коду [1].

Для декодування згорткового коду використовується, зокрема, алгоритм Вітербі. Вихідні біти згорткового коду формуються на основі попередніх біт. Завдяки залежності біт загорткового коду від попередніх біт, є змога коректувати декодовані біти згорткового коду за допомогою метрик.

Об'єктом даного дослідження є проектування схем декодеру Вітербі у базисі FPGA. Мета проектування – розробити декодер Вітербі різними методами, протестувати розроблені схеми, проаналізувати результати тестувань та порівняти їх.

Моделювання та тестування синтезованих схем декодерів здійснювалось у IDE Quartus II. Обраними засобами було реалізовано три різні структури декодеру, а саме: а) структура, яка реалізує метод із знаходженням декодованих послідовностей бітів в процесі декодування без зворотного трасування шляху найменшої метрики; б) структура, яка реалізує метод із збе-

реженням кодів гілок та зворотнім трасуванням шляху найменшої метрики на основі цих кодів без блоків внутрішньої пам'яті (EMB) кристалу FPGA; в) структура, яка реалізує метод із збереженням кодів гілок та зворотнім трасуванням шляху найменшої метрики на основі цих кодів з блоками внутрішньої пам'яті.

Перша структура була виконана в одному блоці, завдяки чому кількість використаних пінів повинна бути мінімальна, але використовуються проміжні комірки пам'яті для тимчасового зберігання метрик та декодованих біт, що підвищує кількість використовуваних регістрів. Друга структура була виконана на елементарних комбінаційних схемах (лічильниках, мультиплексах, компараторах і т. д.), що звичайно підвищує кількість пінів, але не має додаткових регістрів для проміжного зберігання метрик та декодованих біт, замість цього зберігаються коди гілок на основі метрик, що потребує менше регістрів. Хоч для зберігання метрик у другій структурі потрібно менше регістрів ніж у першій, але регістри використовуються і в інших частинах схеми, тому чи буде їх загалом більше чи менше ніж у першій структурі, покажуть результати тестувань.

Завдяки тому, що друга структура реалізована на відокремлених комбінаційних блоках, то блок зберігання кодів гілок, який реалізований на розподілених ресурсах, можна замінити на блок вбудованої пам'яті, у якому будуть зберігатися коди гілок, що зменшує кількість регістрів та розподілених ресурсів. Це і буде третя структура декодеру.

Виконано порівняння трьох структур декодерів за критерієм кількості використаних розподілених ресурсів кристалу, та за критерієм компактною топології схеми. Ці критерії обрані з метою звільнити місце на кристалі для розташування інших функціональних блоків складної вбудованої системи.

Результати експериментів наведені на рис. 1.

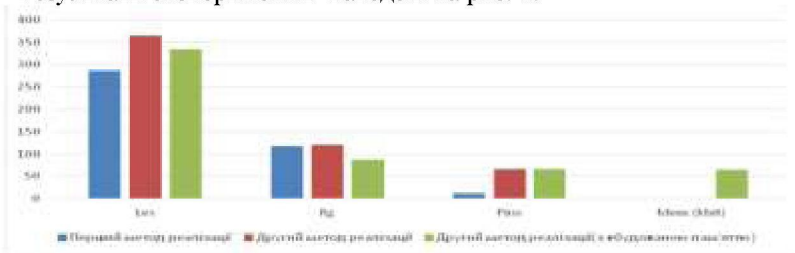


Рисунок 1 – Результати експериментів

Порівнюючи структури, можна зробити висновки щодо ефективності досліджених структур. Отже, друга структура потребує більше ресурсів, ніж перша структура. Третя структура, хоч і має найкращі показники у кількості використаних регістрів, але використовує вбудовані блоки пам'яті. Можна зробити висновок, що перша структура все ж буде кращім вибором порівняно

з іншими структурами, оскільки використовує значно менше розподілених ресурсів та пінів.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Signal processing & simulation newsletter [Електронний ресурс] – Режим доступу: <http://complextoreal.com/wp-content/uploads/2018/09/convolutional-codes.pdf>.