

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Кваліфікаційна наукова праця
на правах рукопису

ЛЮБАРСЬКИЙ МИХАЙЛО МИХАЙЛОВИЧ

УДК 658: 512.011: 681.326: 519.713

ДИСЕРТАЦІЯ

КВАНТОВІ МОДЕЛІ ТА МЕТОДИ АНАЛІЗУ ЛОГІЧНИХ Х-ФУНКЦІЙ

05.13.05 – комп'ютерні системи та компоненти

Технічні науки

Подається на здобуття наукового ступеня кандидата технічних наук

Дисертація містить результати власних досліджень.

Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело _____ М.М. Любарський

Науковий керівник:
Хаханов Володимир Іванович,
доктор технічних наук, професор

Цей примірник дисертації ідентичний за змістом
з іншими примірниками, що подані до спеціалізованої
вченої ради Д 64.052.01

Учений секретар спеціалізованої вченої ради Д 64.052.01

Є.І. Литвинова

Харків – 2018

АНОТАЦІЯ

Любарський Михайло Михайлович. Квантові моделі та методи аналізу логічних X-функцій. – Кваліфікаційна наукова робота на правах рукопису.

Дисертація на здобуття наукового ступеня кандидата технічних наук (доктора філософії) за спеціальністю 05.13.05 «Комп'ютерні системи та компоненти». – Харківський національний університет радіоелектроніки, Міністерство освіти и науки України, Харків, 2018.

Мета дослідження – зменшення часу верифікації цифрових систем на кристалах шляхом використання memory-driven архітектур і кубітних структур даних для компактного опису логічних X-функцій і істотного підвищення продуктивності методів тестування і дедуктивного моделювання несправностей за рахунок паралельного комп'ютингу алгоритмів.

Задачі дослідження: 1) Розробити memory-driven архітектури та алгоритми для паралельного виконання логічних операцій над кубітними структурами даних при реалізації методів тестування та верифікації цифрових систем на кристалах. 2) Удосконалити квантові методи генерації тестів і дедуктивного моделювання несправностей логічних функцій шляхом синтезу матриць булевих похідних за їх кубітним покриттям. 3) Розробити структурну модель метричних властивостей X-функцій, орієнтовану на виконання паралельних операцій, з метою отримання лінійного часу генерації тестів і моделювання цифрових систем. 4) Розробити аналітичну модель для отримання кубітних покриттів X-функцій від кінцевого числа змінних з метою створення логічних схем, які не вимагають експоненційних витрат на синтез і аналіз тестів. 5) Розробити паралельний метод генерації тестів для несправностей X-функцій від кінцевого числа змінних на основі взяття похідних за кубітним покриттям, які формують тести мінімальної довжини. 6) Розробити паралельний метод синтезу дедуктивних кубітних покриттів для моделювання X-функцій на основі отримання одиначної матриці похідних з метою створення секвенсора моделювання дефектів.

Об'єкт дослідження – архітектури і технології паралельного комп'ютингу логічних операцій для вирішення завдань проектування, верифікації та тестування цифрових систем на кристалах.

Предмет дослідження – моделі, методи, алгоритми та процедури паралельного синтезу та аналізу спеціалізованих логічних схем для створення засобів тестування і верифікації на основі кубітних структур даних.

Науково-практична задача – верифікація спеціалізованих цифрових систем на кристалах шляхом використання memory-driven архітектур і кубітних структур даних для компактного опису логічних X-функцій та істотного підвищення продуктивності методів синтезу тестів і дедуктивного моделювання несправностей за рахунок паралельного комп'ютингу алгоритмів.

Сутність дослідження – квантові моделі і методи синтезу тестів і аналізу несправностей логічних X-функцій на основі використання кубітних структур даних і memory-driven архітектур для паралельного комп'ютингу алгоритмів з метою істотного підвищення продуктивності тестування і дедуктивного моделювання цифрових систем на кристалах.

Наукова новизна результатів дослідження:

1) *Вперше запропоновано* структурну модель метричних властивостей X-функцій, що орієнтована на виконання паралельних операцій на кубітних структурах даних в цілях отримання лінійного часу генерації тестів і моделювання цифрових систем.

2) *Вперше запропоновано* аналітичну модель синтезу кубітних покриттів X-функцій від кінцевого числа змінних, яка *характеризується* можливістю створення логічних схем, що не потребують експоненційних витрат на генерування і аналіз тестів перевірки несправностей.

3) *Вперше запропоновано* паралельний метод синтезу тестів для несправностей X-функцій від кінцевого числа змінних, який *характеризується* взяттям булевих похідних по кубітним покриттям, що *дає можливість* отримувати перевіряючі тести мінімальної довжини.

4) *Вперше запропоновано* паралельний метод синтезу дедуктивних кубітних покриттів для моделювання X-функцій, який *характеризується* отриманням одиначної матриці похідних, що *дає можливість* створювати сенсор моделювання дефектів, інваріантний до вхідних тестових наборів.

5) *Удосконалено* memory-driven архітектури та алгоритми для реалізації методів тестування та верифікації цифрових систем на кристалах, які *відрізняються* паралельним виконанням логічних операцій над кубітними структурами даних.

6) *Удосконалено* квантові методи генерації тестів і дедуктивного моделювання несправностей логічних функцій, які *відрізняються* від аналогів синтезом матриць булевих похідних за їх кубітним покриттям.

Практична значимість отриманих результатів. Розроблено моделі, структури даних, методи синтезу та аналізу логічних схем, включаючи X-функції, які дають можливість суттєво зменшити час синтезу тестів і моделювання несправностей, завдяки кубітному опису цифрових схем, що дає можливість паралельно виконувати обчислювальні процедури, які властиві квантовому комп'ютингу. Окремі сервіси синтезу та аналізу кубітних моделей цифрових пристроїв і компонентів реалізовані у вигляді програмних додатків і пройшли вичерпну апробацію у навчальному процесі при вивченні курсів «Квантові обчислення», «Основи комп'ютерної діагностики», «System on Chip». Середовище проектування: SWIFT, C++, Verilog, Java і платформи: Microsoft Windows, X Window і Macintosh OS X.

Публікації. Результати наукових досліджень опубліковані у 22 друкованих працях: 4 розділи у монографії (з них 4 входять до наукометричної бази Scopus), 9 статей (з них 6 – у наукових журналах, включених до «Переліку наукових фахових видань України»; 2 статті в міжнародних наукових журналах за кордоном; 9 статей входять до міжнародних наукометричних баз), а також 9 міжнародних наукових конференціях (з них 6 за кордоном, 3 входять до наукометричної бази Scopus). Пошукач має 7 публікацій у наукометричній базі Scopus, індекс Хірша $h=2$.

СПИСОК ОПУБЛІКОВАНИХ РОБІТ ЗА ТЕМОЮ ДИСЕРТАЦІЇ

Список публікацій здобувача, в яких опубліковані основні наукові результати дисертації:

1. Hahanov V. at all. Cyber Physical Computing for IoT-driven Services [Text] / V. Hahanov at all. – New York. USA. – Springer, 2018. – 279 p. [Part 4. Qubit Description of the Functions and Structures for Service Computing Synthesis [Text] / I. Hahanov, I. Iemelianov, *M. Liubarskyi*, V. Hahanov. – P. 71-93] (Indexed by Springer, Scopus).

2. Hahanov V. at all. Cyber Physical Computing for IoT-driven Services [Text] / V. Hahanov at all. – New York. USA. – Springer, 2018. – 279 p. [Part 5. Quantum Computing for Test Synthesis [Text] / V. Hahanov, Bani Amer T., I. Iemelianov, *M. Liubarskyi*. – P. 95-134] (Indexed by Springer, Scopus).

3. Hahanov V. at all. Cyber Physical Computing for IoT-driven Services [Text] / V. Hahanov at all. – New York. USA. – Springer, 2018. – 279 p. [Part 6. QuaSim – Cloud Service for Quantum Circuits Simulation [Text] / I. Hahanov, Bani Amer T., I. Iemelianov, *M. Liubarskyi*, V. Hahanov – P. 135-147] (Indexed by Springer, Scopus).

4. Hahanov V. at all. Cyber Physical Computing for IoT-driven Services [Text] / V. Hahanov at all. – New York. USA. – Springer, 2018. – 279 p. [Part 14. Cyber-Physical Technologies: Hype Cycle 2017 [Text] / V. Hahanov, Wajeb Ghari-bi, Ka Lok Man, I. Iemelianov, *M. Liubarskyi*, V. Abdullayev, E. Litvinova, S. Chumachenko – P. 259-272] (Indexed by Springer, Scopus).

5. Хаханов В.И. Синтез Q-тестов по кубитному описанию функциональностей [Text] / В.И. Хаханов, Т. Bani Amer, И.В. Емельянов, *М.М. Любарский* // Радиоэлектроника и информатика. – 2016. – № 2(72). – С. 38-48. (Входит до міжнародних наукометричних баз Index Copernicus, Google Scholar, OECSP, OAJI, Scholar Steer, SIS, Cyberleninka, CiteFactor, TIU Hannover, I2OR).

6. Хаханов В.И. Кубитный метод дедуктивного анализа неисправностей для логических схем [Text] / В.И. Хаханов, И.В. Емельянов, *М.М. Любарский*,

С.В. Чумаченко, Е.И. Литвинова, Т. Бани Амер // Электронное моделирование. – 2017. – Том 39, № 6. – С. 59-92 [Входит до міжнародних наукометричних баз Cambridge Scientific Abstracts, Computer and Information Systems Abstracts, INIS Collection, Inspec, ВИНТИ РАН].

7. Емельянов И.В. Квантовые модели и облачные сервисы для анализа и диагностирования логических схем [Text] / И.В. Емельянов, М.М. Любарский, В.И. Хаханов // Радиоэлектроника и информатика. – 2017. – № 4. – С.30-47. (Входит до міжнародних наукометричних баз Index Copernicus, Google Scholar, OECSP, OAJI, Scholar Steer, SIS, Cyberleninka, CiteFactor, TIU Hannover, I2OR)

8. Хаханов В.И. Квантовый метод синтеза тестов на основе кубитных структур данных [Text] / В.И. Хаханов, И.В. Емельянов, М.М. Любарский, С.В. Чумаченко, Е.И. Литвинова // Электронное моделирование. – 2018. – Том 40, № 1. – С. 63-80 (Входит до міжнародних наукометричних баз Cambridge Scientific Abstracts, Computer and Information Systems Abstracts, INIS Collection, Inspec, ВИНТИ РАН).

9. Хаханов В.И. Дедуктивный анализ и диагностирование логических Х-функций [Text] / Хаханов В.И., Емельянов И.В., Любарский М.М., Чумаченко С.В., Литвинова Е.И., Ларченко Л.В. // Радиоэлектроника и информатика. – 2018. – № 1. – С. 55-64. (Входит до міжнародних наукометричних баз Index Copernicus, Google Scholar, OECSP, OAJI, Scholar Steer, SIS, Cyberleninka, CiteFactor, TIU Hannover, I2OR)

10. Любарский М.М. Синтез и анализ логических Х-функций [Text] / М.М. Любарский, В.Г. Абдуллаев, В.И. Хаханов, С.В. Чумаченко, Е.И. Литвинова, И.В. Хаханов // Радиоэлектроника и информатика. – 2018. – №2. – С. 18-28. (Входит до міжнародних наукометричних баз Index Copernicus, Google Scholar, OECSP, OAJI, Scholar Steer, SIS, Cyberleninka, CiteFactor, TIU Hannover, I2OR)

11. Hahanov V. Quantum Data Structures for SoC Component Testing [Text] / V. Hahanov, W. Gharibi, S. Chumachenko, E. Litvinova, I. Iemelianov, M. Liubarskyi // International Journal of Design, Analysis & Tools for Integrated Circuits & Systems.– Oct. 2017.– Vol. 6, iss. 1. – P. 69-76. (Входить до міжнародної наукометричної бази EBSCO Information Services).

12. Хаханов В.И. Gartner 2017 топ-технологии: их анализ и применение [Text] / В.И. Хаханов, А.С. Мищенко, И.В. Емельянов, М.М. Любарский, Т.И. Соклакова, В.Г. Абдулаев // Paradigmata poznání. Vědecko vydavatelské centrum «Sociosféra-CZ», s.r.o., Praha, Česká republika. – 2017. – №4. – P. 33-62. (The journal is indexed by Electronic Research Library, Russia; Research Bible, China; Scientific Indexing Services, USA; Cite Factor, Canada; General Impact Factor, India; Scientific Journal Impact Factor, India; CrossRef, USA; ORCID, USA).

13. Bani Amer T. Компьютинговые модели облачных сервисов [Text] / T. Bani Amer, В.И. Хаханов, И.В. Емельянов, М. Любарский // АСУ и приборы автоматики. – 2015. – Вып. 173. – С.48-57. (Входить до міжнародних наукометричних баз Google Scholar, Cyberleninka).

Результати, які засвідчують апробацію матеріалів дисертації:

14. Hahanov V. Qubit description of the functions and structures for computing [Text] / V. Hahanov, E. Litvinova, S. Chumachenko, M. Liubarskyi // Proc. of the IEEE East-West Design & Test Symposium. – 2016. – Yerevan, Armenia. – 6 p. (Входить до міжнародних наукометричних баз Scopus, IEEE Xplore).

15. Hahanov V. Qubit test synthesis for the black box functionalities [Text] / V. Hahanov, E. Litvinova, S. Chumachenko, I. Iemelianov, M. Liubarskyi // Proc. of 5th Prague Embedded Systems Workshop.– June 29-30, 2017.– Roztoky u Prahy, Czech Republic.– P.45-51.

16. Hahanov I. Qubit deductive fault simulation of black box functionalities [Text] / I. Hahanov, A. Hahanova, V. Hahanov, E. Litvinova, S. Chumachenko, M. Liubarskyi // Proc. of 5th Prague Embedded Systems Workshop. June 29-30, 2017. Roztoky u Prahy, Czech Republic. P.83-91.

17. Hahanov V. Quantum Data Structures for SoC Component Testing [Електронний ресурс] / V. Hahanov, W. Gharibi, K. L. Man, S. Chumachenko, E. Litvinova, I. Iemelianov, *M. Liubarskyi* // Proc. of the International Conference on Recent Advancements in Computing, IoT and Computer Engineering Technology. – The Tamkang University. – Taipei, Taiwan. – October 23-25, 2017. – 1 p.

18. Hahanov V. Quantum memory-driven computing for test synthesis [Text] / V. Hahanov, W. Gharibi, E. Litvinova, *M. Liubarskyi*, A. Hahanova // Proc. of the IEEE East-West Design and Test Symposium. – Novi Sad. – October, 2017. – P. 6. (Входить до міжнародних наукометричних баз Scopus, IEEE Xplore).

19. Vladimir Hahanov. Qubit test synthesis of the functionality [Text] / V. Hahanov, Tamer Bani Amer, E. Litvinova, T. Soklakova, *M. Liubarskyi*, N. Shavlak, K. Dziuba // Proc. of the 14th International Conference “The Experience of Designing and Application of CAD Systems in Microelectronics” (CADSM). – Lviv-Polyana, Ukraine. – 21-25 February, 2017. – P. 251 – 255. (Входить до міжнародних наукометричних баз Scopus, IEEE Xplore).

20. Hahanov V. Qubit Minimization of Boolean Functions [Text] / Vladimir Hahanov, Ka Lok Man, *Mykhailo Liubarskyi*, Ivan Hahanov, Svetlana Chumachenko // Proc. of the International MultiConference of Engineers and Computer Scientists. – Hong Kong. – March 14-16, 2018. – Vol II IMECS 2018. – Pp. 690-695.

21. Hahanov V. Quantum Deductive Fault Simulation [Text] / V. Hahanov, W. Gharibi, *M. Liubarskyi*, E. Litvinova, M. Gharibi, S. Chumachenko, I. Hahanov, and A. Hahanova // Proc. of Int'l Conf. Modeling, Sim. and Vis. Methods (MSV'18) in The 2018 World Congress in Computer Science, Computer Engineering, & Applied Computing (CSCE2018) . – Jul 30-Aug 02, 2018. – Las Vegas, USA. – P. 10-16.

Публікації, які додатково відображають наукові результати дисертації:

22. *Mykhailo Liubarskyi*. IoT Computing Evolution [Text] / *M. Liubarskyi*, I. Hahanov // Матеріали XXI Міжнародного молодіжного форуму «Радіоелектроніка та молодь у XXI ст.». – 25-27 квітня 2017. – Ч. 5. – С.46-47.

Ключові слова: квантові моделі, квантові методи, логічні X-функції, цифрові системи на кристалах, memory-driven архітектура, кубітні структури даних.

АННОТАЦИЯ

Любарский Михаил Михайлович. Квантовые модели и методы анализа логических X-функций. – Квалификационная научная работа на правах рукописи.

Диссертация на соискание ученой степени кандидата технических наук (доктора философии) по специальности 05.13.05 «Компьютерные системы и компоненты». – Харьковский национальный университет радиоэлектроники, Министерство образования и науки Украины, Харьков, 2018.

Цель исследования – уменьшение времени верификации цифровых систем на кристаллах путем использования memory-driven архитектур и кубитных структур данных для компактного описания логических X-функций и существенного повышения производительности методов тестирования и дедуктивного моделирования неисправностей за счет параллельного компьютинга алгоритмов.

Задачи исследования: 1) Разработать memory-driven архитектуры и алгоритмы для параллельного выполнения логических операций над кубитными структурами данных при реализации методов тестирования и верификации цифровых систем на кристаллах. 2) Усовершенствовать квантовые методы генерации тестов и дедуктивного моделирования неисправностей логических функций путем синтеза матриц булевых производных по их кубитным покрытиям. 3) Разработать структурную модель метрических свойств X-функций, ориентированную на выполнение параллельных операций, в целях получения линейного времени генерации тестов и моделирования цифровых систем. 4) Разработать аналитическую модель для получения кубитных покрытий X-функций от конечного числа переменных в целях создания логических схем, не требующих экспоненциальных затрат на синтез и анализ тестов. 5) Разработать параллельный метод генерации тестов для неисправностей X-функций от конечного числа переменных на основе взятия производных по кубитным покрытиям, формирующих тесты минимальной длины. 6)

Разработать параллельный метод синтеза дедуктивных кубитных покрытий для моделирования X-функций на основе получения единичной матрицы производных в целях создания секвенсора моделирования дефектов.

Объект исследования – архитектуры и технологии параллельного компьютинга логических операций для решения задач проектирования, верификации и тестирования цифровых систем на кристаллах.

Предмет исследования – модели, методы, алгоритмы и процедуры параллельного синтеза и анализа специализированных логических схем для создания средств тестирования и верификации на основе кубитных структур данных.

Науково-практична задача – верификация специализированных цифровых систем на кристаллах путем использования memory-driven архитектур и кубитных структур данных для компактного описания логических X-функций и существенного повышения производительности методов синтеза тестов и дедуктивного моделирования неисправностей за счет параллельного компьютинга алгоритмов.

Сущность исследования – квантовые модели и методы синтеза тестов и анализа неисправностей логических X-функций на основе использования кубитных структур данных и memory-driven архитектур для параллельного компьютинга алгоритмов в целях существенного повышения производительности тестирования и дедуктивного моделирования цифровых систем на кристаллах.

Научная новизна результатов исследования:

1) *Впервые предложена* структурная модель метрических свойств X-функций, ориентированная на выполнение параллельных операций на кубитных структурах данных в целях получения линейного времени генерации тестов и моделирования цифровых систем.

2) *Впервые предложена* аналитическая модель синтеза кубитных покрытий X-функций от конечного числа переменных, которая *характеризует*

ся возможностью создания логических схем, не требующих экспоненциальных затрат на генерирование и анализ тестов проверки неисправностей.

3) *Впервые предложен* параллельный метод синтеза тестов для неисправностей X-функций от конечного числа переменных, который *характеризуется* взятием булевых производных по кубитным покрытиям, что *дает возможность* получать проверяющие тесты минимальной длины.

4) *Впервые предложен* параллельный метод синтеза дедуктивных кубитных покрытий для моделирования X-функций, который характеризуется получением единичной матрицы производных, что *дает возможность* создавать секвенсор моделирования дефектов, инвариантный к входным тестовым наборам.

5) *Усовершенствованы* memory-driven архитектуры и алгоритмы для реализации методов тестирования и верификации цифровых систем на кристаллах, которые *отличаются* параллельным выполнением логических операций над кубитными структурами данных.

6) *Усовершенствованы* квантовые методы генерации тестов и дедуктивного моделирования неисправностей логических функций, которые *отличаются* от аналогов синтезом матриц булевых производных по их кубитным покрытиям.

Практическая значимость полученных результатов. Разработаны модели, структуры данных, методы синтеза и анализа логических схем, включая X-функции, которые дают возможность существенно уменьшить время синтеза тестов и моделирования неисправностей, благодаря кубитному описанию цифровых схем, что дает возможность параллельно выполнять вычислительные процедуры, характерные квантовому компьютерингу. Отдельные сервисы синтеза и анализа кубитных моделей цифровых устройств и компонентов реализованы в виде программных приложений и прошли исчерпывающую апробацию в учебном процессе при изучении курсов «Квантові обчислення», «Основи комп'ютерної діагностики», «System on Chip». Среда

проектирования: SWIFT, C++, Verilog, Java и платформы: Microsoft Windows, X Window и Macintosh OS X.

Публикации. Результаты научных исследований опубликованы в 22 работах: 4 раздела в монографии (из них 4 входят в наукометрическую базу Scopus), 9 статей (из них 6 – в научных журналах, включенных в «Перечень научных специализированных изданий Украины»; 2 статьи в международных научных журналах за рубежом; 9 статей входят в международные наукометрические базы), а также 9 международных научных конференций (из них 6 за рубежом, 3 входят в наукометрическую базу Scopus). Соискатель имеет 7 публикаций в наукометрической базе Scopus и индекс Хирша $h=2$: <https://www.scopus.com/authid/detail.uri?authorId=57193611478>

СПИСОК ОПУБЛИКОВАННЫХ РАБОТ ПО ТЕМЕ ДИССЕРТАЦИИ

Список публикаций соискателя, в которых отражены основные научные результаты диссертации:

1. Hahanov V. at all. Cyber Physical Computing for IoT-driven Services [Text] / V. Hahanov at all. – New York. USA. – Springer, 2018. – 279 p. [Part 4. Qubit Description of the Functions and Structures for Service Computing Synthesis [Text] / I. Hahanov, I. Iemelianov, M. Liubarskyi, V. Hahanov. – P. 71-93] (Indexed by Springer, Scopus).

2. Hahanov V. at all. Cyber Physical Computing for IoT-driven Services [Text] / V. Hahanov at all. – New York. USA. – Springer, 2018. – 279 p. [Part 5. Quantum Computing for Test Synthesis [Text] / V. Hahanov, Bani Amer T., I. Iemelianov, M. Liubarskyi. – P. 95-134] (Indexed by Springer, Scopus).

3. Hahanov V. at all. Cyber Physical Computing for IoT-driven Services [Text] / V. Hahanov at all. – New York. USA. – Springer, 2018. – 279 p. [Part 6. QuaSim – Cloud Service for Quantum Circuits Simulation [Text] / I. Hahanov, Bani Amer T., I. Iemelianov, M. Liubarskyi, V. Hahanov – P. 135-147] (Indexed by Springer, Scopus).

4. Hahanov V. at all. Cyber Physical Computing for IoT-driven Services [Text] / V. Hahanov at all. – New York. USA. – Springer, 2018. – 279 p. [Part 14. Cyber-Physical Technologies: Hype Cycle 2017 [Text] / V. Hahanov, Wajeb Ghari-bi, Ka Lok Man, I. Iemelianov, M. Liubarskyi, V. Abdullayev, E. Litvinova, S. Chumachenko – P. 259-272] (Indexed by Springer, Scopus).

5. Хаханов В.И. Синтез Q-тестов по кубитному описанию функциональностей [Text] / В.И. Хаханов, Т. Bani Amer, И.В. Емельянов, М.М. Любарский // Радиоэлектроника и информатика. – 2016. – № 2(72). – С. 38-48. (Входит до міжнародних наукометричних баз Index Copernicus, Google Scholar, OECSP, OAJI, Scholar Steer, SIS, Cyberleninka, CiteFactor, TIU Hannover, I2OR).

6. Хаханов В.И. Кубитный метод дедуктивного анализа неисправностей для логических схем [Text] / В.И. Хаханов, И.В. Емельянов, М.М. Любарский, С.В. Чумаченко, Е.И. Литвинова, Т. Бани Амер // Электронное моделирование. – 2017. – Том 39, № 6. – С. 59-92 [Входит до міжнародних наукометричних баз Cambridge Scientific Abstracts, Computer and Information Systems Abstracts, INIS Collection, Inspec, ВИНТИ РАН].

7. Емельянов И.В. Квантовые модели и облачные сервисы для анализа и диагностирования логических схем [Text] / И.В. Емельянов, М.М. Любарский, В.И. Хаханов // Радиоэлектроника и информатика. – 2017. – № 4. – С.30-47. (Входит до міжнародних наукометричних баз Index Copernicus, Google Scholar, OECSP, OAJI, Scholar Steer, SIS, Cyberleninka, CiteFactor, TIU Hannover, I2OR)

8. Хаханов В.И. Квантовый метод синтеза тестов на основе кубитных структур данных [Text] / В.И. Хаханов, И.В. Емельянов, М.М. Любарский, С.В. Чумаченко, Е.И. Литвинова // Электронное моделирование. – 2018. – Том 40, № 1. – С. 63-80 (Входит до міжнародних наукометричних баз Cambridge Scientific Abstracts, Computer and Information Systems Abstracts, INIS Collection, Inspec, ВИНТИ РАН).

9. Хаханов В.И. Дедуктивный анализ и диагностирование логических Х-функций [Text] / Хаханов В.И., Емельянов И.В., *Любарский М.М.*, Чумаченко С.В., Литвинова Е.И., Ларченко Л.В. // Радиоэлектроника и информатика. – 2018. – № 1. – С. 55-64. (Входит до міжнародних наукометричних баз Index Copernicus, Google Scholar, OECSP, OAJI, Scholar Steer, SIS, Cyberleninka, CiteFactor, TIU Hannover, I2OR)

10. *Любарский М.М.* Синтез и анализ логических Х-функций [Text] / *М.М. Любарский*, В.Г. Абдуллаев, В.И. Хаханов, С.В. Чумаченко, Е.И. Литвинова, И.В. Хаханов // Радиоэлектроника и информатика. – 2018. – №2. – С. 18-28. (Входит до міжнародних наукометричних баз Index Copernicus, Google Scholar, OECSP, OAJI, Scholar Steer, SIS, Cyberleninka, CiteFactor, TIU Hannover, I2OR)

11. Hahanov V. Quantum Data Structures for SoC Component Testing [Text] / V. Hahanov, W. Gharibi, S. Chumachenko, E. Litvinova, I. Iemelianov, *M. Liubarskyi* // International Journal of Design, Analysis & Tools for Integrated Circuits & Systems.– Oct. 2017.– Vol. 6, iss. 1. – P. 69-76. (Входит до міжнародної наукометричної бази EBSCO Information Services).

12. Хаханов В.И. Gartner 2017 топ-технологии: их анализ и применение [Text] / В.И. Хаханов, А.С. Мищенко, И.В. Емельянов, *М.М. Любарский*, Т.И. Соклакова, В.Г. Абдулаев // Paradigmata poznání. Vědecko vydavatelské centrum «Sociosféra-CZ», s.r.o., Praha, Česká republika. – 2017. – №4. – P. 33-62. (The journal is indexed by Electronic Research Library, Russia; Research Bible, China; Scientific Indexing Services, USA; Cite Factor, Canada; General Impact Factor, India; Scientific Journal Impact Factor, India; CrossRef, USA; ORCID, USA).

13. Bani Amer T. Компьютерные модели облачных сервисов [Text] / T. Bani Amer, В.И. Хаханов, И.В. Емельянов, *М. Любарский* // АСУ и приборы автоматики. – 2015. – Вып. 173. – С.48-57. (Входит до міжнародних наукометричних баз Google Scholar, Cyberleninka).

Результаты, которые подтверждают апробацию материалов диссертации:

14. Hahanov V. Qubit description of the functions and structures for computing [Text] / V. Hahanov, E. Litvinova, S. Chumachenko, M. Liubarskyi // Proc. of the IEEE East-West Design & Test Symposium. – 2016. – Yerevan, Armenia. – 6 p. (Входить до міжнародних наукометричних баз Scopus, IEEE Xplore).

15. Hahanov V. Qubit test synthesis for the black box functionalities [Text] / V. Hahanov, E. Litvinova, S. Chumachenko, I. Iemelianov, M. Liubarskyi // Proc. of 5th Prague Embedded Systems Workshop.– June 29-30, 2017.– Roztoky u Prahy, Czech Republic.– P.45-51.

16. Hahanov I. Qubit deductive fault simulation of black box functionalities [Text] / I. Hahanov, A. Hahanova, V. Hahanov, E. Litvinova, S. Chumachenko, M. Liubarskyi // Proc. of 5th Prague Embedded Systems Workshop. June 29-30, 2017. Roztoky u Prahy, Czech Republic. P.83-91.

17. Hahanov V. Quantum Data Structures for SoC Component Testing [Електронний ресурс] / V. Hahanov, W. Gharibi, K. L. Man, S. Chumachenko, E. Litvinova, I. Iemelianov, M. Liubarskyi // Proc. of the International Conference on Recent Advancements in Computing, IoT and Computer Engineering Technology. – The Tamkang University. – Taipei, Taiwan. – October 23-25, 2017. – 1 p.

18. Hahanov V. Quantum memory-driven computing for test synthesis [Text] / V. Hahanov, W. Gharibi, E. Litvinova, M. Liubarskyi, A. Hahanova // Proc. of the IEEE East-West Design and Test Symposium. – Novi Sad. – October, 2017. – P. 6. (Входить до міжнародних наукометричних баз Scopus, IEEE Xplore).

19. Vladimir Hahanov. Qubit test synthesis of the functionality [Text] / V. Hahanov, Tamer Bani Amer, E. Litvinova, T. Soklakova, M. Liubarskyi, N. Shavlak, K. Dziuba // Proc. of the 14th International Conference “The Experience of Designing and Application of CAD Systems in Microelectronics” (CADSM). – Lviv-Polyana, Ukraine. – 21-25 February, 2017. – P. 251 – 255. (Входить до міжнародних наукометричних баз Scopus, IEEE Xplore).

20. Hahanov V. Qubit Minimization of Boolean Functions [Text] / Vladimir Hahanov, K.L. Man, *Mykhailo Liubarskyi*, Ivan Hahanov, Svetlana Chumachenko // Proc. of the International MultiConference of Engineers and Computer Scientists. – Hong Kong. – March 14-16, 2018. – Vol II IMECS 2018. – Pp. 690-695.

21. Hahanov V. Quantum Deductive Fault Simulation [Text] / V. Hahanov, W. Gharibi, *M. Liubarskyi*, E. Litvinova, M. Gharibi, S. Chumachenko, I. Hahanov, and A. Hahanova // Proc. of Int'l Conf. Modeling, Sim. and Vis. Methods (MSV'18) in The 2018 World Congress in Computer Science, Computer Engineering, & Applied Computing (CSCE2018) . – Jul 30-Aug 02, 2018. – Las Vegas, USA. – P. 10-16.

Публикации, которые дополнительно отражают научные результаты диссертации:

22. *Mykhailo Liubarskyi*. IoT Computing Evolution [Text] / *M. Liubarskyi*, I. Hahanov // Матеріали XXI Міжнародного молодіжного форуму «Радіоелектроніка та молодь у XXI ст.». – 25-27 квітня 2017. – Ч. 5. – С.46-47.

Ключевые слова: квантовые модели, квантовые методы, логические X-функции, цифровые системы на кристаллах, memory-driven архитектура, кубитные структуры данных.

ABSTRACT

Liubarskyi Mykhailo Mykhailovych. Quantum models and methods for logic X-function analysis – Qualification scientific work as a manuscript.

Thesis for the degree of candidate of technical sciences (Ph.D.) in specialty 05.13.05 "Computer systems and components". – Kharkov National University of Radio Electronics, Ministry of Education and Science of Ukraine, Kharkov, 2018.

The purpose of the research is reducing the verification time of digital systems-on-chips by using memory-driven architectures and qubit data structures to compactly describe logical X-functions and significantly improve the performance of methods for testing and deductive fault simulation through parallel computing of algorithms.

The research tasks: 1) Develop memory-driven architectures and algorithms for parallel execution of logical operations on qubit data structures when implementing methods for testing and verification of digital systems-on-chips. 2) Improve the quantum methods of test generation and deductive fault simulation of logical functions by synthesizing matrices of Boolean derivatives with respect to their qubit coverage. 3) Develop a structural model of the metric properties of X-functions, focused on the implementation of parallel operations, in order to provide the linear time of test generation and simulation of digital systems. 4) Develop an analytical model for defining qubit coverages of X-functions of a finite number of variables in order to create logic circuits, which do not require exponential costs for the synthesis and analysis of tests. 5) Develop a parallel test generation method for the faults of X-functions of a finite number of variables based on taking the derivatives with respect to qubit coverages, which define tests of minimal length. 6) Develop a parallel method for the synthesis of deductive qubit coverages for the simulation of X-functions based on obtaining a unit matrix of derivatives in order to create a sequencer for fault simulation.

The object of the research is architectures and technologies of parallel computing of logic operations for solving problems of designing, verifying and testing digital systems-on-chips.

The subject of the research is models, methods, algorithms and procedures for parallel synthesis and analysis of specialized logic circuits to create test and verification tools based on qubit data structures.

The scientific and practical task is verification of specialized digital systems-on-chips by using memory-driven architectures and qubit data structures to compactly describe logical X-functions and significantly improve the performance of methods for testing and deductive fault simulation through parallel computing of algorithms.

The essence of the research is quantum models and methods for test synthesis and fault analysis of logical X-functions based on the use of qubit data structures and memory-driven architectures for parallel computing of algorithms in order to significantly improve the performance of testing and deductive simulation of digital systems-on-chips.

Scientific novelty of research results:

1) A structural model of the metrical properties of X-functions is proposed for the first time, focused on executing parallel operations on qubit data structures in order to provide linear time of test generation and simulation of digital systems.

2) An analytical model for the synthesis of qubit coverages of X-functions of a finite number of variables is proposed for the first time, characterizing by the possibility of creating logic circuits, which do not require exponential costs of generating and analyzing tests.

3) A parallel method of test synthesis for the faults of X-functions of a finite number of variables is proposed for the first time, characterizing by taking Boolean derivatives with respect to qubit coverages, which makes it possible to generate tests of minimal length.

4) A parallel method for the synthesis of deductive qubit coverages for simulation of X-functions is proposed for the first time, characterizing by obtaining a unit matrix of derivatives, which makes it possible to create a sequencer for fault simulation that is invariant to the input test patterns.

5) Memory-driven architectures and algorithms for implementing methods for testing and verifying digital systems-on-chips are improved, which are distinguished by the parallel execution of logical operations on qubit data structures.

6) The quantum methods of test generation and deductive fault simulation of logic functions are improved, which differ from analogs in the synthesis of matrices of Boolean derivatives with respect to their qubit coverages.

The practical significance of the research results. Models, data structures, methods for synthesizing and analyzing logic circuits, including X functions, have been developed, which make it possible to significantly reduce the time of test synthesis and fault simulation through the qubit description of digital circuits, that makes it possible to simultaneously perform computational procedures related to quantum computing. Some services for the synthesis and analysis of qubit models of digital devices and components are implemented in the form of software applications and in the educational process in training courses “Quantum Computing”, “Fundamentals of computer diagnostics”, «System-on-Chip». Design environment includes SWIFT, C++, Verilog, Java and platforms: Microsoft Windows, X Window and Macintosh OS X.

The results of the dissertation work are reflected in 22 printed works, among which is 4 chapters in foreign monograph of the publishing house Springer (Scopus science-base); 9 articles (6 – in scientific journals included in the "Perelik Naukovych Fakhovikh Vidany of Ukraine"; 2 articles in international scientific journals abroad; 9 articles in the international scientific and metric database); as well as 9 materials of international scientific conferences (6 of them – in abroad, 3 are included in Scopus scientific and metric database). The applicant has 7 publications in the science-based Scopus database and the Hirsch index $h = 2$.

LIST OF PUBLICATIONS

The list of publications, which reflect the main scientific results of the thesis:

1. Hahanov V. at all. Cyber Physical Computing for IoT-driven Services [Text] / V. Hahanov at all. – New York. USA. – Springer, 2018. – 279 p. [Part 4.

Qubit Description of the Functions and Structures for Service Computing Synthesis [Text] / I. Hahanov, I. Iemelianov, *M. Liubarskyi*, V. Hahanov. – P. 71-93] (Indexed by Springer, Scopus).

2. Hahanov V. at all. Cyber Physical Computing for IoT-driven Services [Text] / V. Hahanov at all. – New York. USA. – Springer, 2018. – 279 p. [Part 5. Quantum Computing for Test Synthesis [Text] / V. Hahanov, Bani Amer T., I. Iemelianov, *M. Liubarskyi*. – P. 95-134] (Indexed by Springer, Scopus).

3. Hahanov V. at all. Cyber Physical Computing for IoT-driven Services [Text] / V. Hahanov at all. – New York. USA. – Springer, 2018. – 279 p. [Part 6. QuaSim – Cloud Service for Quantum Circuits Simulation [Text] / I. Hahanov, Bani Amer T., I. Iemelianov, *M. Liubarskyi*, V. Hahanov – P. 135-147] (Indexed by Springer, Scopus).

4. Hahanov V. at all. Cyber Physical Computing for IoT-driven Services [Text] / V. Hahanov at all. – New York. USA. – Springer, 2018. – 279 p. [Part 14. Cyber-Physical Technologies: Hype Cycle 2017 [Text] / V. Hahanov, Wajeb Ghari-bi, Ka Lok Man, I. Iemelianov, *M. Liubarskyi*, V. Abdullayev, E. Litvinova, S. Chumachenko – P. 259-272] (Indexed by Springer, Scopus).

5. Хаханов В.И. Синтез Q-тестов по кубитному описанию функциональностей [Text] / В.И. Хаханов, Т. Bani Amer, И.В. Емельянов, *М.М. Любарский* // Радиоэлектроника и информатика. – 2016. – № 2(72). – С. 38-48. (Входит до міжнародних наукометричних баз Index Copernicus, Google Scholar, OECSP, OAJI, Scholar Steer, SIS, Cyberleninka, CiteFactor, TIU Hannover, I2OR).

6. Хаханов В.И. Кубитный метод дедуктивного анализа неисправностей для логических схем [Text] / В.И. Хаханов, И.В. Емельянов, *М.М. Любарский*, С.В. Чумаченко, Е.И. Литвинова, Т. Бани Амер // Электронное моделирование. – 2017. – Том 39, № 6. – С. 59-92 [Входит до міжнародних наукометричних баз Cambridge Scientific Abstracts, Computer and Information Systems Abstracts, INIS Collection, Inspec, ВИНТИ РАН].

7. Емельянов И.В. Квантовые модели и облачные сервисы для анализа и диагностирования логических схем [Text] / И.В. Емельянов, *М.М. Любарский*, В.И. Хаханов // Радиоэлектроника и информатика. – 2017. – № 4. – С.30-47. (Входит до міжнародних наукометричних баз Index Copernicus, Google Scholar, OECSP, OAJI, Scholar Steer, SIS, Cyberleninka, CiteFactor, TIU Hannover, I2OR)

8. Хаханов В.И. Квантовый метод синтеза тестов на основе кубитных структур данных [Text] / В.И. Хаханов, И.В. Емельянов, *М.М. Любарский*, С.В. Чумаченко, Е.И. Литвинова // Электронное моделирование. – 2018. – Том 40, № 1. – С. 63-80 (Входит до міжнародних наукометричних баз Cambridge Scientific Abstracts, Computer and Information Systems Abstracts, INIS Collection, Inspec, ВИНТИ РАН).

9. Хаханов В.И. Дедуктивный анализ и диагностирование логических Х-функций [Text] / Хаханов В.И., Емельянов И.В., *Любарский М.М.*, Чумаченко С.В., Литвинова Е.И., Ларченко Л.В. // Радиоэлектроника и информатика. – 2018. – № 1. – С. 55-64. (Входит до міжнародних наукометричних баз Index Copernicus, Google Scholar, OECSP, OAJI, Scholar Steer, SIS, Cyberleninka, CiteFactor, TIU Hannover, I2OR)

10. *Любарский М.М.* Синтез и анализ логических Х-функций [Text] / *М.М. Любарский*, В.Г. Абдуллаев, В.И. Хаханов, С.В. Чумаченко, Е.И. Литвинова, И.В. Хаханов // Радиоэлектроника и информатика. – 2018. – №2. – С. 18-28. (Входит до міжнародних наукометричних баз Index Copernicus, Google Scholar, OECSP, OAJI, Scholar Steer, SIS, Cyberleninka, CiteFactor, TIU Hannover, I2OR)

11. Hahanov V. Quantum Data Structures for SoC Component Testing [Text] / V. Hahanov, W. Gharibi, S. Chumachenko, E. Litvinova, I. Iemelianov, *M. Liubarskyi* // International Journal of Design, Analysis & Tools for Integrated Circuits & Systems.– Oct. 2017.– Vol. 6, iss. 1. – P. 69-76. (Входит до міжнародної наукометричної бази EBSCO Information Services).

12. Хаханов В.И. Gartner 2017 топ-технологии: их анализ и применение [Text] / В.И. Хаханов, А.С. Мищенко, И.В. Емельянов, М.М. Любарский, Т.И. Соклакова, В.Г. Абдулаев // Paradigmata poznání. Vědecko vydavatelské centrum «Sociosféra-CZ», s.r.o., Praha, Česká republika. – 2017. – №4. – P. 33-62. (The journal is indexed by Electronic Research Library, Russia; Research Bible, China; Scientific Indexing Services, USA; Cite Factor, Canada; General Impact Factor, India; Scientific Journal Impact Factor, India; CrossRef, USA; ORCID, USA).

13. Bani Amer T. Компьютинговые модели облачных сервисов [Text] / T. Bani Amer, В.И. Хаханов, И.В. Емельянов, М. Любарский // АСУ и приборы автоматики. – 2015. – Вып. 173. – С.48-57. (Входит до міжнародних наукометричних баз Google Scholar, Cyberleninka).

Results that confirm the approbation of the thesis:

14. Hahanov V. Qubit description of the functions and structures for computing [Text] / V. Hahanov, E. Litvinova, S. Chumachenko, M. Liubarskyi // Proc. of the IEEE East-West Design & Test Symposium. – 2016. – Yerevan, Armenia. – 6 p. (Входит до міжнародних наукометричних баз Scopus, IEEE Xplore).

15. Hahanov V. Qubit test synthesis for the black box functionalities [Text] / V. Hahanov, E. Litvinova, S. Chumachenko, I. Iemelianov, M. Liubarskyi // Proc. of 5th Prague Embedded Systems Workshop.– June 29-30, 2017.– Roztoky u Prahy, Czech Republic.– P.45-51.

16. Hahanov I. Qubit deductive fault simulation of black box functionalities [Text] / I. Hahanov, A. Hahanova, V. Hahanov, E. Litvinova, S. Chumachenko, M. Liubarskyi // Proc. of 5th Prague Embedded Systems Workshop. June 29-30, 2017. Roztoky u Prahy, Czech Republic. P.83-91.

17. Hahanov V. Quantum Data Structures for SoC Component Testing [Электронный ресурс] / V. Hahanov, W. Gharibi, K. L. Man, S. Chumachenko, E. Litvinova, I. Iemelianov, M. Liubarskyi // Proc. of the International Conference on Recent Advancements in Computing, IoT and Computer Engineering Technology. – The Tamkang University. – Taipei, Taiwan. – October 23-25, 2017. – 1 p.

18. Hahanov V. Quantum memory-driven computing for test synthesis [Text] / V. Hahanov, W. Gharibi, E. Litvinova, M. Liubarskyi, A. Hahanova // Proc. of the IEEE East-West Design and Test Symposium. – Novi Sad. – October, 2017. – P. 6. (Входить до міжнародних наукометричних баз Scopus, IEEE Xplore).

19. Vladimir Hahanov. Qubit test synthesis of the functionality [Text] / V. Hahanov, Tamer Bani Amer, E. Litvinova, T. Soklakova, M. Liubarskyi, N. Shavlak, K. Dziuba // Proc. of the 14th International Conference “The Experience of Designing and Application of CAD Systems in Microelectronics” (CADSM). – Lviv-Polyana, Ukraine. – 21-25 February, 2017. - P. 251 – 255. (Входить до міжнародних наукометричних баз Scopus, IEEE Xplore).

20. Hahanov V. Qubit Minimization of Boolean Functions [Text] / Vladimir Hahanov, K.L. Man, Mykhailo Liubarskyi, Ivan Hahanov, Svetlana Chumachenko // Proc. of the International MultiConference of Engineers and Computer Scientists. – Hong Kong. – March 14-16, 2018. – Vol II IMECS 2018. – Pp. 690-695.

21. Hahanov V. Quantum Deductive Fault Simulation [Text] / V. Hahanov, W. Gharibi, M. Liubarskyi, E. Litvinova, M. Gharibi, S. Chumachenko, I. Hahanov, and A. Hahanova // Proc. of Int'l Conf. Modeling, Sim. and Vis. Methods (MSV'18) in The 2018 World Congress in Computer Science, Computer Engineering, & Applied Computing (CSCE2018) . – Jul 30-Aug 02, 2018. – Las Vegas, USA. – P. 10-16.

Publications that additionally reflect the scientific results of the thesis:

22. Mykhailo Liubarskyi. IoT Computing Evolution [Text] / M. Liubarskyi, I. Hahanov // Матеріали XXI Міжнародного молодіжного форуму «Радіоелектроніка та молодь у XXI ст.». – 25-27 квітня 2017. – Ч. 5. – С.46-47.

Key words: quantum models, quantum methods, logical X-functions, digital system-on-chip, memory-driven architecture, qubit data structures.

ЗМІСТ

ВСТУП	25
РОЗДІЛ 1	35
QUANTUM-DRIVEN АРХІТЕКТУРИ ДЛЯ ПАРАЛЕЛЬНОГО ПРОЕКТУВАННЯ ЦИФРОВИХ СИСТЕМ.....	35
1.1 КІБЕРКУЛЬТУРИ КВАНТОВОГО КОМП'ЮТИНГУ	35
1.2 QUANTUM MEMORY-DRIVEN COMPUTING FOR DESIGN AND TEST	36
1.3 STATE OF THE ART. КВАНТОВИЙ КОМП'ЮТИНГ	41
1.4. ПОСТАНОВКА МЕТИ І ЗАВДАНЬ ДОСЛІДЖЕННЯ	56
РОЗДІЛ 2	59
АРХІТЕКТУРИ КВАНТОВОЇ І КЛАСИЧНОЇ КОМП'ЮТИНГУ	59
2.1 MEMORY-DRIVEN АРХІТЕКТУРИ	59
2.2 КЛАСИФІКАЦІЯ КВАНТОВОГО КОМП'ЮТИНГУ	65
2.3 МЕТРИКА КЛАСИЧНОГО І КВАНТОВОГО КОМП'ЮТИНГУ	73
2.4 АРХІТЕКТУРИ КВАНТОВОГО КОМП'ЮТИНГУ	75
2.5. КОМП'ЮТИНГ В МЕТРИЦІ: ПРОСТІР-ЧАС, МАТЕРІЯ-ЕНЕРГІЯ	79
2.6. ВИСНОВКИ.....	86
РОЗДІЛ 3	87
ДЕДУКТИВНИЙ АНАЛІЗ ЛОГІЧНИХ ФУНКЦІЙ	87
3.1 STATE OF THE ART	87
3.2 СИНТЕЗ ДЕДУКТИВНИХ ФОРМУЛ ДЛЯ ПРИМІТИВІВ.....	90
3.3 СИНТЕЗ ДЕДУКТИВНИХ ФОРМУЛ ДЛЯ ХОR (NOT-ХОR) ФУНКЦІЙ	93
3.4 СИНТЕЗ БУЛЕВИХ ПОХІДНИХ ПО КУБІТНИМ ПОКРИТТЯМ	98
3.5 СИНТЕЗ ТЕСТІВ ДЛЯ ЛОГІЧНИХ Х-ФУНКЦІЙ.....	104
3.6. ВИСНОВКИ.....	110
РОЗДІЛ 4	111
СИНТЕЗ І АНАЛІЗ ЛОГІЧНИХ Х-ФУНКЦІЙ	111
4.1. ВЛАСТИВОСТІ ЛОГІЧНИХ Х-ФУНКЦІЙ	112
4.2 МЕТОД СИНТЕЗУ ДЕДУКТИВНОГО КУБІТНОГО ПОКРИТТЯ І ДНФ ПО ТАБЛИЦІ ІСТИННОСТІ І 15	
4.3. МЕТОД СИНТЕЗУ ДЕДУКТИВНОГО КУБІТНОГО ПОКРИТТЯ І ДНФ ПО КУБІТНИМ ПОКРИТТЯМ	
ФУНКЦІОНАЛЬНОСТЕЙ.....	118
4.4. МЕТОД СИНТЕЗУ МАТРИЦІ ПЕРЕСТАНОВКИ БІТІВ ДЛЯ ФОРМУВАННЯ ДЕДУКТИВНИХ	
ФУНКЦІЙ	122
4.5 ІМПЛЕМЕНТАЦІЯ КУБІТНИХ МОДЕЛЕЙ І МЕТОДІВ ЇХ АНАЛІЗУ	127
4.6. ВИСНОВКИ.....	133
ВИСНОВОК.....	135
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	138
ДОДАТОК А СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА ЗА ТЕМОЮ ДИСЕРТАЦІЇ.....	151
ДОДАТОК Б АПРОБАЦІЯ РЕЗУЛЬТАТІВ.....	155
ДОДАТОК В ДОКУМЕНТИ, ЩО ПІДТВЕРДЖУЮТЬ ВПРОВАДЖЕННЯ	
РЕЗУЛЬТАТІВ ДИСЕРТАЦІЇ.....	158
ДОДАТОК Г ПРОГРАМНИЙ ДОДАТОК (ФРАГМЕНТ).....	159

ВСТУП

Актуальність дослідження. Аналітичні прогнози останніх десяти років від компанії Gartner виділяють квантовий комп'ютинг як триггерну інноваційну технологію, в яку вкладаються мультиміліардні інвестиції від топ-компаній планети. Квантовий комп'ютинг – галузь знань, що займається теорією і практикою паралельного розв'язання комбінаторних задач на обчислювачах, що використовують субатомні частинки при створенні структур даних і їх фізичної взаємодії для реалізації логічних операцій. Квантовий комп'ютинг, що емулює, – структури даних, моделі, методи і алгоритми для створення software додатків з метою паралельного розв'язання комбінаторних задач на класичних комп'ютерах шляхом використання додаткової пам'яті. Квантовий комп'ютер без квантових паралельних алгоритмів – лише дорога іграшка. Тому стратегія створення квантового комп'ютингу полягає в одночасній розробці квантової апаратури і квантових програмних додатків на основі паралельних алгоритмів. Паралельний і роздільний розвиток двох гілок квантового комп'ютингу надає можливість науковцям з країн, що розвиваються, активно брати участь в проектуванні, моделюванні та верифікації квантових алгоритмів і програмних додатків на класичних комп'ютерах з метою їх подальшої імплементації в ринково доступні квантові комп'ютери в недалекому майбутньому. Моделювання на класичних комп'ютерах паралельних квантових алгоритмів дає істотний приріст продуктивності за рахунок використання надлишкової пам'яті для унітарного кодування кубітних структур даних. Більш того, моделювання квантових алгоритмів стратегічно покликане вирішувати проблему створення квантового інтелекту планети шляхом розробки банку паралельних програмних додатків для майбутнього парку ринково доступних квантових комп'ютерів. Design and Test є найбільш передовим напрямом діяльності вчених і компаній, спрямованої на

створення нових технологій комп'ютингу за метрикою: швидкодія, енергозбереження, компактність і надійність. Параметр *time-to-market* при створенні комп'ютингової продукції поряд з якістю є домінуючим. Тому квантові паралельні методи та алгоритми проектування, тестування, верифікації, моделювання і діагностування цифрових виробів на кубітних структурах даних є *актуальними* при їх імплементації в сучасні класичні та у майбутні квантові комп'ютери з метою істотного зменшення *time-to-market*. Особливий інтерес представляє використання кубітних покриттів примітивних функціональностей для мінімізації, синтезу, аналізу і діагностування цифрових систем на кристалах. В роботі розглядаються логічні (xor, notxor) X-функції, що задаються симетричними кубітними покриттями, для технологічного паралельного вирішення завдань побудови тестів і дедуктивного моделювання несправностей. Синтез дедуктивних формул для X-функцій формує компактну структуру для транспортування вхідних списків несправностей, яка інваріантна до вхідних тестових наборів. Це робить X-функцію привабливою для її використання при проектуванні тестопридатності цифрових систем. Унікальність X-функцій проявляється також у технологічній простоті процедури взяття булевих похідних на основі зустрічного зсуву двійкових розрядів, що є основою для генератора тестів, який використовує кубітні покриття логічних схем. Квантовий метод синтезу тестів для X-функцій від n змінних генерує 5 вхідних наборів, які покривають всі поодинокі константні несправності, перевірені за допомогою дедуктивного моделювання за кубітними покриттями функцій. Тест для X-функції від n змінних, довжиною $L(T) = \frac{1}{2} * (2^{**n}) + 1$, будується на основі використання кубітних покриттів шляхом об'єднання всіх одиничних координат кубітного вектора і однієї нульової координати вектора. Таблиця інтегральних властивостей X-функцій є корисною для вирішення завдань Design and Test на основі використання кубітних структур даних. Структура секвенсора моделювання та аналітичні вирази для швидкого синтезу X-функції будь-якої розмірності від n -змінних використо-

вують паралельні операції конкатенації і інверсії. Завдяки кубітній формі опису дедуктивних функцій, вдалося синтезувати дедуктивну матрицю для транспортування несправностей, яка характеризується високим рівнем компактності структур даних і технологічністю її отримання на основі пот-операції і процедури перестановки бітів в кубітному вектор-стовпці. Нова теорія дедуктивного аналізу цифрових систем на основі кубітних покриттів використовує найпростіші матричні операції, пов'язані з перестановкою і конкатенацією, для рекурсивного отримання як завгодно складної структури даних з метою моделювання логічних схем, які використовують X-функції. Отримувані матриці мають властивості вертикальної, горизонтальної та діагональної симетрії, завдяки внутрішнім властивостям X-функцій. Конструктивною властивістю кубітного покриття є паралелізм інтерпретативного моделювання цифрових систем, що використовує характеристичне рівняння, яке оперує при транзакціях даних тільки адресами кубітного покриття і вектора моделювання, який створює схемну структуру. Секвенсор синтезу тестів, що використовує логічні операції: інверсії, зсуву, порівняння і об'єднання, є прототипом апаратної реалізації модуля, вбудованого в інфраструктуру SoC, для online-тестування функціональностей. Такий апаратний модуль може бути доступний, як хмарний сервіс, що має IP-адресу, який створює прямий зв'язок з будь-якою функціональністю у масштабах кіберпростору. **Основним результатом** запропонованого дослідження є створення основ теорії і методів квантового memory-driven проектування і тестування цифрових систем, орієнтованих на паралельні алгоритми синтезу та аналізу кубітних структур даних, що імплементовані у emerging memory-driven quantum computing.

Розроблено квантові моделі і методи синтезу та аналізу логічних (X) - функцій на основі використання векторних кубітних структур даних і memory-driven архітектур для паралельного комп'ютингу алгоритмів з метою істотного підвищення продуктивності тестування і дедуктивного моделю-

вання цифрових систем на кристалах. Визначено структуру метричних властивостей X-функцій від кінцевого числа змінних, яка дає можливість істотно зменшити час синтезу тестів і моделювання несправностей за кубітним покриттям. Запропоновано новий квантовий метод генерації перевіряючих тестів для одиночних константних несправностей логічних функцій шляхом взяття булевих похідних за їх кубітним покриттям. Розроблено квантовий метод дедуктивного моделювання константних несправностей логічних функцій шляхом синтезу матриць булевих похідних за їх кубітним покриттям. Швидкодія методів синтезу тестів і їх аналізу не має аналогів за алгоритмічною продуктивністю.

Рішення проблем, пов'язаних з design and test, проектуванням, верифікацією, синтезом тестів і моделюванням цифрових систем, представлено в публікаціях наступних вчених: Y. Zorian, J. Bergeron, Z. Navabi, A. Jerraya, DB Armstrong, P. Prinetto, J. Abraham, H. Fujiwara, T. Nishida, X. Wang, P. Mueller, А. Петренко, Р. Убар, А. Ivanov, А. Романкевич, В. Романкевич, Д. Сперанський, А. Матросова, П. Пархоменко, J.P. Roth, В. Мелікян, С. Шукурян, В. Тарасенко, М. Коровай, А. Палагін, В. Опанасенко, В. Харченко, Л. Дербунович, В. Ярмолик, Р. Шейнаускас, Н. Євтушенко, Р. Базилевич, Г. Кривуля, М. Мірошник.

Зв'язок дисертаційної роботи з науковими проектами, держбюджетними темами. Розробка теми дисертації здійснювалася відповідно до планів держбюджетних НДР, виконуваних на кафедрі Автоматизації проектування обчислювальної техніки Харківського національного університету радіоелектроніки в період з 2015 року, в тому числі: 1) Договір про дружбу та співробітництво між ХНУРЕ та компанією «Aldec Inc.» (USA) № 04 від 01.11.2011. 2) Curricula Development for New Specialization: Master of Engineering in Microsystems Design 530785-TEMPUS-1-2012-1-PL-TEMPUS-JPCR MastMST (2012-2016). 3) Фундаментальна держбюджетна НДР №297 "Cyber Physical System – Smart Cloud Traffic Control" (2015-2017), № ДР

0115U-000712; 4) Фундаментальна держбюджетна НДР № 316 "Cyber Physical System – Smart Cyber University" (2017-2019), № ДР 0117U-0002524.

Автор дисертаційної роботи брав участь при виконанні зазначених договорів і проектів як розробник-архітектор і програміст-тестувальник засобів синтезу тестів і моделювання несправностей на основі використання IEEE стандартів.

Дисертаційна робота спрямована на вирішення науково-практичної задачі верифікації спеціалізованих цифрових систем на кристалах шляхом використання memory-driven архітектур і кубітних структур даних для компактного опису логічних X-функцій та істотного підвищення продуктивності методів синтезу тестів і дедуктивного моделювання несправностей за рахунок паралельного комп'ютингу алгоритмів.

Сутність дослідження – квантові моделі і методи синтезу тестів та аналізу несправностей логічних X-функцій на основі використання кубітних структур даних і memory-driven архітектур для паралельного комп'ютингу алгоритмів з метою істотного підвищення продуктивності тестування і дедуктивного моделювання цифрових систем на кристалах.

Об'єкт дослідження – архітектури і технології паралельного комп'ютингу логічних операцій для вирішення завдань проектування, верифікації та тестування цифрових систем на кристалах.

Предмет дослідження – моделі, методи, алгоритми та процедури паралельного синтезу й аналізу спеціалізованих логічних схем для створення засобів тестування і верифікації на основі кубітних структур даних.

Мета дослідження – зменшення часу верифікації цифрових систем на кристалах шляхом використання memory-driven архітектур і кубітних структур даних для компактного опису логічних X-функцій і істотного підвищення продуктивності методів тестування і дедуктивного моделювання несправностей за рахунок паралельного комп'ютингу алгоритмів.

Завдання дослідження:

1) Розробити memory-driven архітектури та алгоритми для паралельного виконання логічних операцій над кубітними структурами даних при реалізації методів тестування та верифікації цифрових систем на кристалах.

2) Удосконалити квантові методи генерації тестів і дедуктивного моделювання несправностей логічних функцій шляхом синтезу матриць булевих похідних за їх кубітним покриттям.

3) Розробити структурну модель метричних властивостей X-функцій, орієнтовану на виконання паралельних операцій, з метою отримання лінійного часу генерації тестів і моделювання цифрових систем.

4) Розробити аналітичну модель для отримання кубітних покриттів X-функцій від кінцевого числа змінних з метою створення логічних схем, які не вимагають експоненційних витрат на синтез і аналіз тестів.

5) Розробити паралельний метод генерації тестів для несправностей X-функцій від кінцевого числа змінних на основі взяття похідних за кубітним покриттям, які формують тести мінімальної довжини.

6) Розробити паралельний метод синтезу дедуктивних кубітних покриттів для моделювання X-функцій на основі отримання одиначної матриці похідних з метою створення секвенсора моделювання дефектів.

Методи дослідження – паралельний квантовий комп'ютинг, теорія цифрових автоматів, архітектура комп'ютерів, булева алгебра, теорія множин, теорія графів, кубітні структури даних – для синтезу векторних моделей цифрових пристроїв; логічний аналіз, теорія алгоритмів, мови проектування – для синтезу та аналізу цифрових систем; методи, метрики і критерії якості – для оцінювання тестопригодності цифрових виробів; IEEE стандарти – для тестування і верифікації програмних додатків.

Наукова новизна результатів дослідження:

1) *Вперше запропоновано* структурну модель метричних властивостей X-функцій, що орієнтована на виконання паралельних операцій на кубітних

структурах даних з метою отримання лінійного часу генерації тестів і моделювання цифрових систем.

2) *Вперше запропоновано* аналітичну модель синтезу кубітних покриттів X -функцій від кінцевого числа змінних, яка *характеризується* можливістю створення логічних схем, що не потребують експоненційних витрат на генерування і аналіз тестів перевірки несправностей.

3) *Вперше запропоновано* паралельний метод синтезу тестів для несправностей X -функцій від кінцевого числа змінних, який *характеризується* взяттям булевих похідних за кубітним покриттям, що *дає можливість* отримувати перевіряючі тести мінімальної довжини.

4) *Вперше запропоновано* паралельний метод синтезу дедуктивних кубітних покриттів для моделювання X -функцій, який *характеризується* отриманням одиничної матриці похідних, що *дає можливість* створювати сенсор моделювання дефектів, інваріантний до вхідних тестових наборів.

5) *Удосконалено* memory-driven архітектури та алгоритми для реалізації методів тестування та верифікації цифрових систем на кристалах, які *відрізняються* паралельним виконанням логічних операцій над кубітними структурами даних.

6) *Удосконалено* квантові методи генерації тестів і дедуктивного моделювання несправностей логічних функцій, які *відрізняються* від аналогів синтезом матриць булевих похідних за їх кубітним покриттям.

Практична значимість отриманих результатів. Розроблено моделі, структури даних, методи синтезу та аналізу логічних схем, включаючи X -функції, які дають можливість суттєво зменшити час синтезу тестів і моделювання несправностей, завдяки кубітному опису цифрових схем, що дає можливість паралельно виконувати обчислювальні процедури, характерні квантовому комп'ютингу. Окремі сервіси синтезу та аналізу кубітних моделей цифрових пристроїв і компонентів реалізовані у вигляді програмних додатків і пройшли вичерпну апробацію у навчальному процесі при вивченні

курсів «Квантові обчислення», «Основи комп'ютерної діагностики», «System on Chip». Середовище проектування: SWIFT, C ++, Verilog, Java і платформи: Microsoft Windows, X Window і Macintosh OS X.

Отримані при виконанні досліджень наукові висновки і практичні результати є достовірними. Вони підтверджуються достатньою кількістю виконаних експериментів, тестуванням і верифікацією моделей логічних схем і синтезованих тестів. Результати дослідження в складі моделей, методів і фрагментів додатків впроваджені у навчальний процес Харківського національного університету радіоелектроніки (акт про впровадження від 23.10.2018); в методологічне та технологічне забезпечення компанії «DIT Delight Limited» (довідка про впровадження від 25.10.2018).

Особистий внесок здобувача. Всі наукові і практичні результати отримані автором особисто. У роботах, опублікованих зі співавторами, здобувачеві належати (див. анотації, Додаток А, Б): [1] – кубітні моделі для квантового методу синтезу тестів; [2] – кубітні структури для опису функцій і тестів; [3] – квантові структури опису логічних елементів і архітектура системи моделювання; [4] – аналітичний огляд технологій квантового комп'ютингу стосовно Design and Test; [5] – аналітична модель для синтезу Q-тестів на основі кубітних покриттів; [6] – синтез аналітичних моделей для дедуктивного моделювання несправностей; [7] – синтез аналітичних моделей для діагностування одиночних і кратних константних несправностей; [8] – метод синтезу тестів для класу логічних функцій на основі кубітних векторів; [9] – метод діагностування несправностей логічних схем, дедуктивний метод моделювання несправностей за кубітним покриттям логічних елементів, паралельний метод синтезу дедуктивних кубітних покриттів для моделювання X-функцій, memory-driven архітектури та алгоритми для реалізації методів тестування та верифікації цифрових систем на кристалах; [10] – структурна модель метричних властивостей X-функцій, аналітична модель синтезу кубітних покриттів X-функцій від кінцевого числа змінних, метод синтезу дедуктивних кубітних

покриттів для моделювання X-функцій, memory-driven архітектури тестування і верифікації цифрових систем на кристалах, квантові методи генерації тестів і дедуктивного моделювання несправностей логічних функцій; [11] – квантові структури даних для генерації тестів і моделювання справної поведінки цифрових схем; [12] – аналітичний огляд кубітних методів діагностування та тестування цифрових систем; [13] – структури даних і архітектури хмарних сервісів для вирішення завдань тестування цифрових систем на кристалах; [14] – кубітний опис логічних функцій для синтезу та аналізу memory-driven комп'ютингу; [15] – аналітичні моделі для синтезу тестів на основі використання кубітних похідних; [16] – алгоритм кубітного дедуктивного моделювання логічних схем і його програмна реалізація; [17] – квантові структури даних для синтезу тестів і аналізу несправностей; [18] – архітектура квантового memory-driven комп'ютингу для синтезу тестів; [19] – моделі кубітного опису логічних схем для генерування тестів; [20] – алгоритм мінімізації логічних функцій на основі унітарного опису таблиць невизначених коефіцієнтів; [21] – метричні властивості X-функцій; [22] – аналітичний огляд хмарних технологій, орієнтованих на вирішення завдань технічної діагностики.

Апробація результатів дисертації. Результати роботи були представлені та обговорені на таких конференціях: 1-2) IEEE East-West Design and Test Symposium 2016 (Armenia); 2017 (Serbia); 3) XX Міжнародний молодіжний форум «Радіоелектроніка та молодь у XXI сторіччі» 2017 (Україна); 4) 13th International Conference: The Experience of Designing and Application of CAD Systems in Microelectronics, CADSM 2017 (Lvov, Ukraine); 5) 5th Prague Embedded Systems Workshop, 2017 Roztoky u Prahy (Czech Republic); 6) the International Conference on Recent Advancements in Computing, IoT and Computer Engineering Technology, 2017 (Taiwan); 7) the International MultiConference of Engineers and Computer Scientists, 2018 (Hong Kong); 8) World Congress in Computer Science, Computer Engineering, & Applied Computing, 2018 (Las Vegas, USA).

Публікації. Результати наукових досліджень опубліковані у 22 друкованих працях: 4 розділи у монографії (з них 4 входять до наукометричної бази Scopus), 9 статей (з них 6 – у наукових журналах, включених до «Переліку наукових фахових видань України»; 2 статті в міжнародних наукових журналах за кордоном; 9 статей входять до міжнародних наукометричних баз), а також 9 міжнародних наукових конференціях (з них 6 за кордоном, 3 входять до наукометричної бази Scopus). Пошукач має 7 публікацій у наукометричній базі Scopus та індекс Хірша $h=2$:

<https://www.scopus.com/authid/detail.uri?authorId=57193611478>

Структура дисертації представлена 159 сторінками (з них 124 представляють основний текст) і містить: 4 розділи, 32 рисунка, список джерел з 102 назв (на 13 с.), 4 додатки (на 9 с.), анотації на 22 с.

РОЗДІЛ 1

QUANTUM-DRIVEN АРХІТЕКТУРИ ДЛЯ ПАРАЛЕЛЬНОГО ПРОЕКТУВАННЯ ЦИФРОВИХ СИСТЕМ

Пропонується аналіз існуючих публікацій в області створення моделей, методів, алгоритмів і архітектур квантового комп'ютингу для тестування і моделювання цифрових систем з метою створення теорії і практики квантового проектування цифрових систем. Ставляться науково-практичні завдання дослідження в рамках технологій синтезу тестів і моделювання несправностей для спеціалізованих логічних схем, включаючи X-функції.

Мета дослідження – обґрунтування актуальності розв'язання задачі тестування і верифікації спеціалізованих логічних схем шляхом розробки паралельних квантових методів аналізу і синтезу кубітних структур даних цифрових систем для їх подальшої реалізації як серверні програмних додатків.

Завдання дослідження: 1) Технології квантових обчислювачів на основі субатомних структур даних і логічних відносин між ними. 2) Імплементация квантових структур даних, алгоритмів і технологій в класичний комп'ютинг. 3) Порівняльний аналіз квантових і класичних обчислювачів. 4) Архітектури квантового memory-driven комп'ютингу. 5) Постановка цілей і завдань дослідження, спрямованих на створення моделей, методів і програмних додатків паралельного проектування цифрових систем на основі кубітних структур даних.

1.1 Кіберкультура квантового комп'ютингу

Кіберкультура квантового memory-driven комп'ютингу інтегрує технології паралельного рішення времязатратних комбінаторних задач. Масштабна карта досліджень, відповідних квантової кіберкультурі, націлена на створен-

ня паралельних алгоритмів для SoC Design and Test. Дається аналітичний огляд кіберфізичних технологій, задекларованих в Gartner's Hype Cycle [1], а також деякі роз'яснення, пов'язані з їх застосуванням в науці, освіті, транспорті, промисловості та державних структурах. Показуються окремі напрямки, які не ввійшли в цикл ринково привабливих технологій, які стосуються кібер-соціального моніторингу та цифрового управління суспільством [2, 3]. Пропонується розширити значення технологій, пов'язаних з розумним цифровим світом, зеленими містами і 5G-телекомунікаціями [4]. Дано рекомендації до використання топ 10 компонентів Gartner супер-циклу в бізнесі і науково-освітніх процесах університетів. Формулюється D-метрика вимірювання відстаней між процесами і явищами в кіберфізическом і соціальному просторі у вигляді універсальної формули для точного визначення дуги зворотного транзитивного замикання в трикутнику по двом відомим відстаням, що доповнює дугу до конволюційного циклу. Видається memory-driven інноваційна архітектура квантового комп'ютингу, яка характеризується використанням елетромагнітних фотонних транзакцій записування на структурі електронів при відсутності логіки, пов'язаної з суперпозицією і змішування станів. Пропонуються паралельні методи мінімізації булевих функцій, діагностування несправностей цифрових схем і рішення задачі покриття на основі кубітних структур даних. Вводиться клас логічних X-функцій, технологічних для тестування і діагностування, визначаються кубітние моделі і методи для моделювання несправностей цифрових пристроїв і компонентів. Описується архітектура хмарних сервісів проектування, тестування і верифікації цифрових пристроїв на основі кубітних моделей логічних примітивів.

1.2 Quantum memory-driven computing for design and test

Пропонується технологічна культура quantum memory-driven computing, яка інтегрально має переваги структурної однорідності і паралелізму обробки великих даних за рахунок усунення транзакцій між пам'яттю і ALU-

процесором, і виключення квантових операцій суперпозиції і змішування. Пропонується memory-driven інноваційна архітектура квантового комп'ютингу, яка характеризується використанням елетромагнітних фотонних транзакцій запису-зчитування на структурі електронів при відсутності логіки, пов'язаної з суперпозицією і змішування станів. Пропонуються паралельні методи мінімізації булевих функцій, діагностування несправностей цифрових схем і розв'язання задачі покриття на основі унітарного кодування кубітних структур даних. Вводиться кубітна форма логічних X-функцій, технологічних для тестування і діагностування цифрових систем. Визначаються кубітні моделі і методи для моделювання несправностей цифрових пристроїв і компонентів. Описується архітектура хмарних сервісів проектування і моделювання цифрових пристроїв на основі кубітних моделей логічних і функціональних елементів. Gartner тенденції світової кіберкультури формують технологічну основу для створення глобального кіберфізичного комп'ютингу в межах технологічного укладу Internet of Things (рис. 1.1).

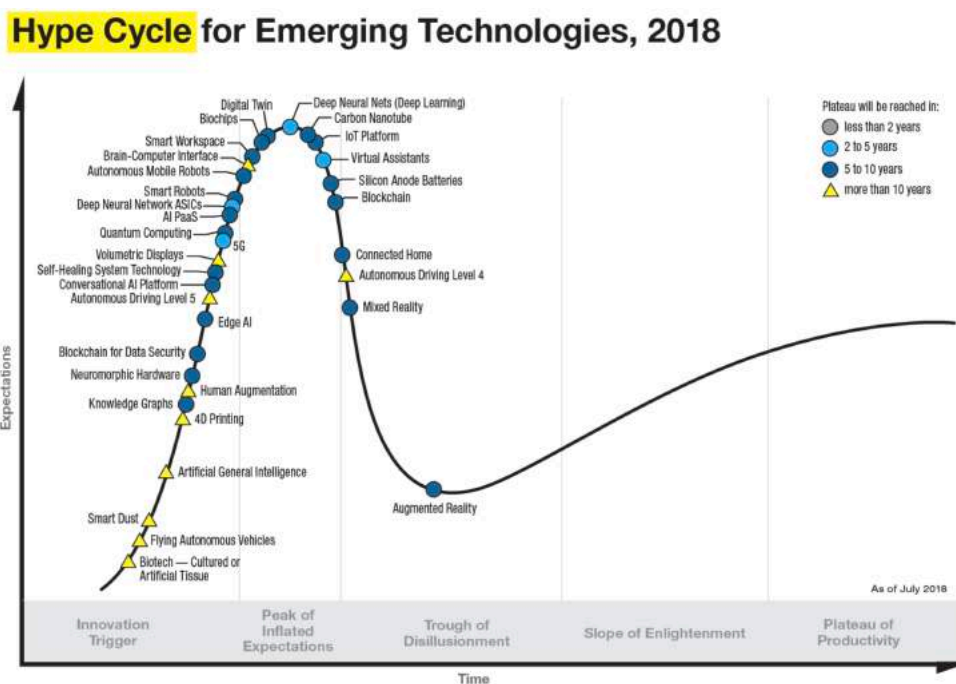


Рис. 1.1 – Цикл компанії Gartner для дзірапторних технологій 2018

При цьому квантовий комп'ютинг розглядається як енергозберігаюче майбутнє цифрового світу, створеного для підвищення якості життя і збереження екології планети [1, 5, 6]. Зокрема, квантовий паралельний комп'ютинг і кубітні структури даних дозволяють спростити алгоритми в області SoC Design and Test і підвищити швидкодію програмних продуктів на класичних комп'ютерах.

Стратегічні тенденції в області цифрових технологій приведуть в 2018 році до суттєвих дізрапціям, що надасть нові можливості розробникам корпоративної архітектури і конструктивних інновацій з метою створення конкурентних переваг при використанні нових трендів кіберкультури (рис. 1.2):

- 1) Автономні фізичні та віртуальні інтелектуальні і координовані речі;
- 2) Розширена (доповнена киберпростором) соціальна аналітика прав і можливостей громадянина для вироблення актюаторних впливів;
- 3) AI-кероване проектування, розширений (доповнений - augmented) і автоматичний розробник;
- 4) Цифрові близнюки; цифровий образ організації або компанії;
- 5) Спроможні, взаємодоповнюючі один одного Edge Computing and Cloud Computing. Роль 5G в комунікаціях між кінцевими пристроями;
- 6) Досвід занурення в цифрову дійсність. Сприйняття змін в цифровому світі. Virtual and Augmented Reality підвищують продуктивність праці. Майбутнє залежить від досвіду сьогодення, що охоплює.
- 7) Використання Blockchain в соціальній сфері [7, 8];
- 8) Smart Spaces. Розумні міські простори, що створюють шляхи. Інтелектуальний простір створює карти, а не напрям [9];
- 9) Цифрова етика і конфіденційність особистого життя;
- 10) Квантові обчислення; квантова безпека; розвиток і становлення квантового комп'ютингу. Застереження – слід поважати QC, працювати з обережністю.

Порівняння двох трендів (2017 [6] і 2018 [1]) від компанії Гартнер дає можливість визначити нові технології, а також ті, які не витримали випробування часом і зійшли з дистанції в результаті гонки. Істотною уявляється перманентність тренда розвитку квантових технологій протягом останніх 5 років. З нею зв'язуються оптимістичні прогнози у розв'язанні проблем комбінаторної оптимізації при вирішенні задач кіберзахисту і обробки великих даних.

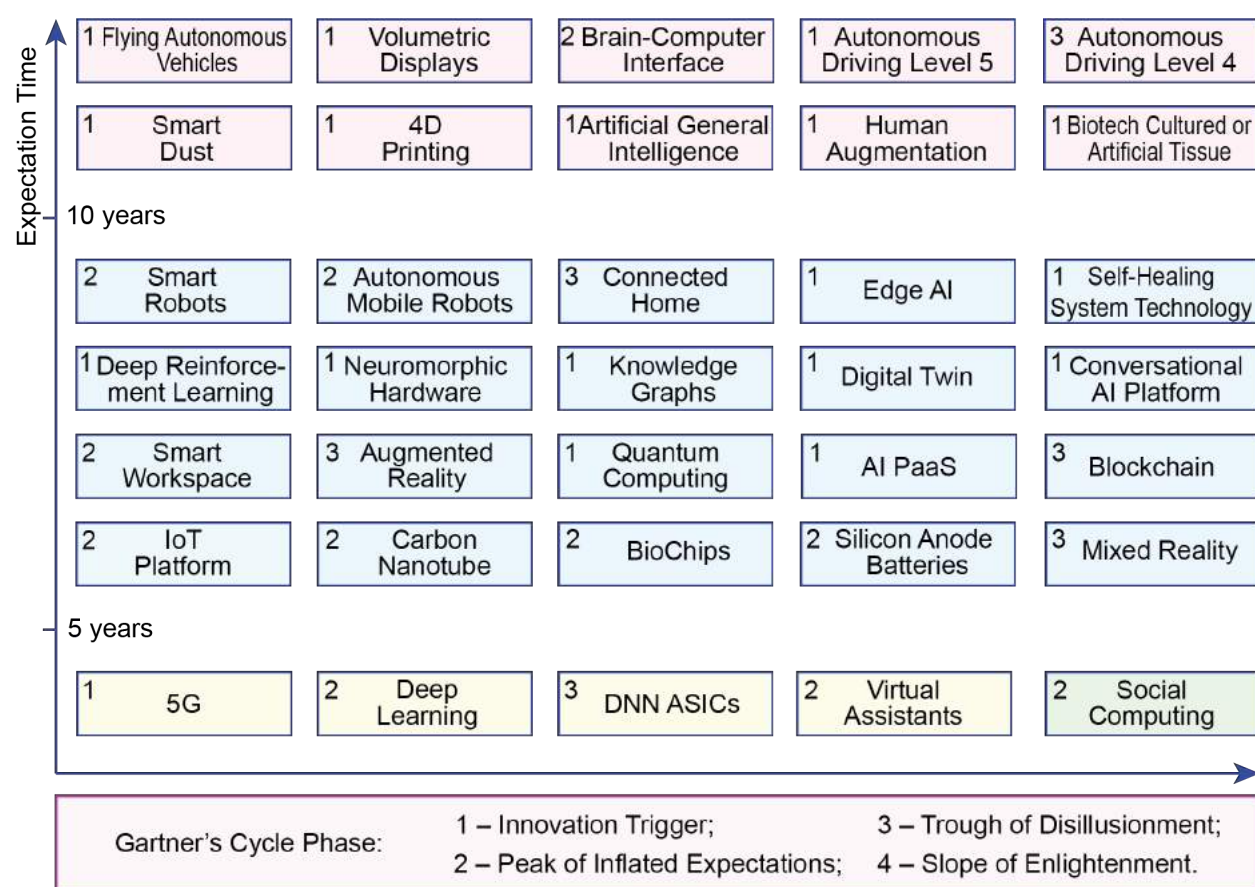


Рисунок 1.2 – 2018 Gartner's Table for Emerging Technologies

Сутність дослідження полягає в інноваційній емуляції Quantum Computing і Memory Computing на Classical Computing для синтезу та аналізу цифрових систем на кристалах. Мотивація і ринкова привабливість дослідження включає наступні пункти: 1) Згідно з журналом Форбс, близько 20 мільярдів доларів інвестується в квантовий комп'ютинг, з яких половина вит-

рачається на створення паралельних квантових алгоритмів, їх емуляцію і верифікацію на класичних комп'ютерах. 2) Перенесення існуючих додатків на квантовий комп'ютер, на думку експертів, не збільшить продуктивності існуючих послідовних алгоритмів, напрацьованих людством. Потрібно створювати паралельні методи. 3) Практично всі програмісти на планеті працюють в режимі створення послідовних обчислювальних процесів. Тому університети сьогодні повинні переучувати фахівців методам квантового паралельного програмування на основі кубітних структур даних. 4) В даний час актуальною видається розробка паралельних програмних алгоритмів, орієнтованих на ринково доступні квантові комп'ютери і використовують кубітні структури даних. 5) Продуктивність квантових memory-driven методів проектування і тестування цифрових систем на кристалах, імплементовані в класичні комп'ютери, являє собою інновацію, яка підвищує швидкість обчислень на основі використання додаткової пам'яті. 6) Реалізація memory-driven Quantum Computing дозволить істотно спростити парадигму комп'ютингу шляхом відмови від ALU, машини фон Неймана, в сторону паралельної обробки великих даних на обчислювальній пам'яті.

Мета дослідження – розробка квантового memory-driven computing для паралельного синтезу та аналізу цифрових систем на кристалах, що використовують кубітні структури даних, на класичних комп'ютерах.

Завдання дослідження: 1) Огляд існуючих квантових архітектур і практичних реалізацій комп'ютингу. 2) Метричне порівняння квантового та класичного комп'ютинга. 3) Розробка квантових memory-driven архітектур для емуляції паралельних алгоритмів на класичних комп'ютерах. 4) Створення паралельних кубітних методів аналізу цифрових систем на кристалах. 5) Реалізація методів, алгоритмів тестування і моделювання цифрових пристроїв.

1.3 State of the art. Квантовий комп'ютинг

Інноваційна пропозиція полягає у створенні квантового memory-driven комп'ютингу [10-12] без квантових операцій суперпозиції і змішування (АБО, ІІ) на основі використання характеристичного рівняння $M = Q [M (X)]$, що задає дві транзакції запису-зчитування на атомарній структурі електронів, що формує обчислювальну пам'ять M, Q, X . Виключити дві технологічно складні операції з квантового комп'ютингу – означає істотно спростити архітектуру і привести її до структури пам'яті на електронах для виконання транзакцій між ними за допомогою квантів або фотонів. Підтвердженням спроможності запропонованої інноваційної квантової архітектури можуть служити публікації, які підкреслюють стійку тенденцію до створення квантового комп'ютингу на атомарній структурі пам'яті з передачею інформації за допомогою фотонів або квантів.

Вчені з Каліфорнійського технологічного інституту створили оптичну квантову пам'ять [13], в яких інформація передається шляхом кодування даних з використанням квантового стану фотонів. Пам'ять реалізована на рідкоземельних елементах і здатна зберігати стану фотонів за допомогою резонаторів-посередників між атомом і світлом. Розмірність квантової пам'яті в 1000 разів менше, ніж традиційні класичні рішення. Вона реалізована в нано порожнині, яка дозволяє зберігати інформацію в дуже невеликому обсязі.

Практична реалізація ідеї заміни електронів фотонами призводить до створення комп'ютингу з швидкодією, близькою до швидкості світла [14]. Корейські дослідники зробили ще один крок до оптичних обчислень. Вони створили photon-triggered нано-дротяний транзистор на основі кристалічного і пористого кремнію, де перемикання і посилення величини токового сигналу здійснюється під впливом фотона. Використання фотонів в логічних вентилях AND, OR і NAND призведе до ультракомпактних нанопроцесорів і нанорозмірних фотоприймачів для отримання зображень з високою роздільною здатністю.

Вчені з Колумбійського університету провели успішні дослідження зі створення транзистора з одного атома в молекулярній електроніці [15]. Вони реалізували геометрично упорядкований кластер неорганічних атомів з центральним ядром, що складається з 14 атомів, яке пов'язали з золотими електродами, що дозволило керувати транзистором під впливом одного електрона при кімнатній температурі.

Вперше досягнута передача цифрових сигналів між молекулами, що є істотним досягненням на шляху розвитку молекулярного комп'ютингу [16]. Створення електронних компонентів з окремих молекул є багатообіцяючою стратегією для мініатюризації та інтеграції електронних пристроїв. Проте практична реалізація молекулярних пристроїв і схем для передачі і обробки сигналів при кімнатній температурі виявилася складним завданням, яке вирішене шляхом розміщення молекул SnCl_2Pc на поверхні міді (Cu). Площинна орієнтація молекул в міжмолекулярній взаємодії служить носієм інформації. В пов'язаних молекулярних масивах сигнал передається від однієї молекули до іншої за наперед заданими маршрутами, які реалізують логічні операції. Явища площинної орієнтації дозволяють використовувати молекули, які мають внутрішні бістабільні стани, для створення складних молекулярних пристроїв і схем.

Вчені з Інституту Нільса Бора (NBI, Копенгаген) винайшли нове джерело світла, здатне випромінювати поодинокі фотони з безпрецедентною точністю вимірювання 98,4%. Використання ефективних однофотонних джерел орієнтоване на квантові мережі, криптографію та симулятори, які представляють собою дізрапторні вирішення соціальних проблем в галузі забезпечення безпеки та створення суперкомп'ютерів. Однофотонні джерела в даний час поки не є комерційно доступними, проте квантові дослідження забезпечуються великими інвестиціями в практично всіх провідних компаніях планети. Наприклад, лабораторія Sparrow Quantum Technologies [17] (SQT), орієнтована на розробку і комерціалізацію фотонних квантових технологій,

отримала 2 млн євро для виробництва і продажу нового SQT-чіпа. Він складається з квантового джерела світла, вбудованого в фотонний чип, виготовлений за напівпровідниковою нанотехнологією. Наступний крок в бізнес-плані SQT - інноваційний проект: розробка набору ключових фотонних компонентів, які мають комерційну застосовність: Quantum Photonic Toolbox (QTOOL) як настроюваний інструментарій для досліджень в області фотонних квантових обчислень, криптографії, мереж.

Компанія IonQ завершила проект [18] з розробки квантових обчислень для комерційних додатків, вартістю 20 мільйонів доларів, за участю нових стратегічних інвесторів, включаючи: New Enterprise Associates (NEA) і GV (раніше Google Ventures). IonQ планує вивести на ринок квантові комп'ютери загального призначення до кінця 2018 року. Зусилля з використання квантової механіки для прискорення обчислень в останні роки отримали величезний імпульс, викликаний значним науковим прогресом, зростаючими ринковими інтересами та істотними інвестиціями з боку найбільших урядів, організацій і технологічних компаній. Необхідні інноваційні розробки конкретних програм для практичних квантових комп'ютерів. «Місія IonQ полягає в перетворенні квантових обчислень в надійний масштабований комп'ютинг» (David Moehring, CEO IonQ). Захоплення іонів є ключовою відмінністю підходу IonQ від інших досліджень в області квантових обчислень і створення платформи для реалізації комерційних додатків. Технологія захоплення іонів, що використовує лазери для охолодження і виділення окремих іонів, є ринково перспективною, оскільки спіймані іони ідентичні, стабільні, контрольовані, масштабовані і передбачувані.

Інтерес представляють «дивні» квантові частинки [19], що моделюються в краплі ультрахолодного газу, звані скірміон (skyrmion), вперше запропоновані у 1962 році фізиком і математиком США Тону Скурме. Завдяки точному контролю за допомогою електромагнітних котушок, що оточують скляну вакуумну камеру, заповнену надтекучим рубідієм, команда вчених з США

і Фінляндії вперше створила 3D скіріміон в полі спинів, що локалізовані у збудженому просторі, які важко розірвати. Вони сконцентровані у вигляді крапельки, що містить 200 000 переохолоджених атомів рубідію на рівні абсолютного нуля. Відкриття, опубліковане у Science Advances, дає уявлення про кульову блискавку, яка може проникати крізь стіни і раптово вибухати, подібно динаміту. Проте кульові блискавки, як і скіріміон, можуть утримуватися разом електромагнітним полем, які набувають дивовижну структурну незруйновність. Даний факт можна використовувати для організації квантового комп'ютингу, завдяки стабільності і передбачуваності атомної структури квантових частинок (протонів і нейтронів) в умовах електромагнітного утримання понад-охолодженого газу рубідію.

Публікація [20] спрямована на створення і використання сильного спин-фотонного зв'язку в кремнієвому кристалі для розробки квантових схем на основі затворів квантових точок напівпровідника. Потенційно, магнітний диполь або спин одного електрона для використання як кубіта має переваги в його часовій стійкості у порівнянні з зарядово-фотонним зчепленням (charge-photon coupling). Автори створили нано-гібрид шляхом об'єднання спіна електрона із зарядом електрона у вигляді подвійної кремнієвої квантової точки. Такий підхід призвів до створення сильного зв'язку між поодиноким електронним спіном і одним мікрохвильовим фотоном, що забезпечує шлях до масштабованих квантових схем зі спин-кубітами. Тривалий час когерентності (узгодженості) одиночних спинів в кремнієвих квантових точках роблять ці системи привабливими для квантових обчислень. Однак питання про масштабування спин-кубитових систем залишається відкритим. Для можливого вирішення даної проблеми автори продемонстрували сильний зв'язок між одним електронним спіном і одним мікрохвильовим фотоном. Електронний спин вловлюється подвійною квантовою точкою кремнію, а мікрохвильовий фотон зберігається у високоомному (high-impedance) надпроводовому резонаторі на кристалі. Компонент електричного поля фотонного резонатора

безпосередньо пов'язаний з зарядом диполя електрона в подвійній точці і опосередковано пов'язаний з спіном електрона через сильний локальний градієнт магнітного поля від сусіднього мікромагнетіка. Результати авторів забезпечують шлях до реалізації великих мереж квантових точково-спінових (dot-based spin) кубітних реєстрів.

Квантова акустика [21] може дати потужне прискорення до створення ембріональних квантових комп'ютерів. Такі машини функціонують на основі маніпуляцій з квантовими бітами або кубітами, які можуть бути встановлені в нуль, єдність і невизначений стан. Найбільш просунуті кубіти створюють схеми на основі надпровідного металу для запису або зчитування даних шляхом використання мікрохвильового резонатора (cavity quantum electrodynamics – CQED). Він представлений смужкою металу в чіпі, де мікрохвильові фотони резонансно активують кубіти. Існує можливість заміни мікрохвильового резонатора механічним, який коливається з квантовими коливаннями або фононами. Мікромініатюрний акустичний резонатор може функціонувати довше, ніж мікрохвильовий, що дає можливість виробляти більш компактні квантові вироби. Однак тут існує проблема управління квантовими вібраціями, яка може бути вирішена вченими в найближчі 3 роки.

Квантова культура сучасного комп'ютинга показана в публікації Paul Teich, TRIAS Research [22]. Тут представлені аналогії технологічного напівпровідникового поля інтегральних схем 50-річної давності, які на більш високому рівні, у вигляді поля квантових обчислень, повторюються у 2018 році.

Інтегральна схема на основі кремнію (IC) увійшла до фази «середньої» інтеграції у 1968 році. За кілька років IC піднялася від десяти транзисторів на чіпі до сотень. Подальше зростання ступеня інтеграції освоїло архітектурні рішення для тисяч транзисторів в чіпі, потім для користувачів стали доступними десятки тисяч, а сьогодні, через п'ятдесят років, технологія видає де-

сятки мільярдів транзисторів в одному кристалі. Технологія перемагає фантазію.

Квантовий комп'ютинг – обчислювальний процес на основі використання квантової механіки для управління елементами пам'яті – субатомними частинками, охолодженими до температури, близької до абсолютного нуля. Субатомні примітиви пам'яті називаються «кубітами». Кубіти можуть бути виготовлені за допомогою технологій CMOS у вигляді стандартних ІС. Але ланцюги з'єднання, управління і моніторингу, необхідні для створення квантових архітектур с великою кількістю кубітів у холодному робочому середовищі, вимагають розробки нових технологічних рішень.

Квантові обчислення в даний час використовують двозначну логіку кубіта. Рік 2017 відзначений появою універсального чіпа з 20 фізичними кубітами. Рік 2018 справить на світ комп'ютер з 50+ універсальними кубітами. Перші масові і ринково-орієнтовані квантові архітектури матимуть тисячу логічних кубітів, які забезпечать відмовостійкість на основі виявлення помилок в даних і їх корекції. Дані логічні властивості вимагають наявності на порядок більшого числа фізичних кубітів (10 000).

Еволюція структур даних від десятків до сотень фізичних кубітів може зайняти до 5 років. Для реалізації архітектур від сотень до тисяч кубітів потрібно більше десяти років. Квантова перевага неодмінно настане, але хотілося б раніше, ніж через 20 років.

Компанії IBM і Rigetti представили хмарні квантові (20 і 19-кубітні) комп'ютери загального призначення для публічного і приватного доступу з повним набором інструментів (devkit) з метою розробки програмного забезпечення. Компанія NTT (Nippon Telegraph and Telephone) представила хмарну архітектуру, засновану на квантовій крапці і photonics-based архітектурі, а також працюючу на її основі Quantum Neural Network для вирішення ринкових задач. Комп'ютер споживає всього 1 kW електроенергії в той час, як для роботи суперкомп'ютера необхідно 10 000 kW. Microsoft і Google показали

свої програми R & D з квантових обчислень загального призначення, а також повнофункціональні devkits та симулятори, але поки не відкрили для публічного доступу своє апаратне забезпечення. Компанія Intel також показала прототипи чіпів без демонстрації їх можливостей. Існують і багато інших лабораторій, такі як IonQ, Quantum Circuits і RIKEN, що вкладають кошти в розробку квантової апаратури без демонстрації своїх результатів. Тільки дві компанії вже продають користувачам спеціалізовані апаратні системи, які можна назвати квантовими комп'ютерами: архітектуру квантового відпалу (annealing) для вирішення завдання пошуку глобального екстремуму в дискретному просторі D-Wave і квантовий імітатор Atos. D-Wave і NTT реалізують архітектуру з 2048 фізичних кубітів, використовуючи абсолютно різні технології для вирішення деяких задач комбінаторної оптимізації, молекулярної динаміки та глибокого навчання (deep learning training).

Simulating Quantum Iron. Для моделювання квантових архітектур на основі десятків фізичних кубітів потрібні потужні «класичні» IC-driven комп'ютери. Такі симулятори працюють на один два порядки повільніше, ніж квантові комп'ютери. Проте дослідники мають намір створювати реальні потужні класичні системи для емуляції квантових архітектур і обчислень. Нещодавно (2017 рік) команда дослідників з двох університетів (Wuhan, Groningen) успішно змоделювала 46-кубітний універсальний квантовий комп'ютер і побила 45-кубітний рекорд, встановлений Energy's Lawrence Berkeley National Laboratory. Пізніше американська команда Harvard-MIT Center for Ultracold Atoms і California Institute of Technology змоделювала квантовий комп'ютер, об'ємом 51 кубіт для вирішення конкретного рівняння. Потім команда University of Maryland і US National Institute of Standards and Technology опублікувала статтю про 53-кубітний симулятор, який також призначений для вирішення конкретного завдання. Хмара IBM здатна моделювати на класичному суперкомп'ютері квантову систему із загальним обсягом 56 кубітів. Доступ користувачів до 16-ти кубітної обчислювальної структури є відкри-

тим в рамках нової парадигми – програми Q Network. Новий Microsoft Quantum Development Kit підтримує моделювання більш 40 кубітів в хмарі Azure. При цьому локальна комп'ютерна симуляція може масштабуватися до 30 кубітів на 16 ГБ пам'яті. Симулятор Forrest від компанії Rigetti може імітувати архітектуру до 36 кубітів, в той час як квантовий майданчик Google може симулювати 22 кубітний обчислювач.

Компанія IBM задекларувала 20-кубітний чіп наприкінці 2017 року, який став основою для IBM Q Network програми. Далі IBM стверджує, що вона вже побудувала і внутрішньо протестувала 50-кубітний чіп. Учасники мережі IBM Q Network сьогодні мають доступ до 20-кубітних систем, а завтра отримають ранній доступ до 50-кубітного чіпу. Компанія Intel випустила тестовий чіп на 17-кубітів для партнера QuTech і показала 49-кубітний чіп в 2018 році на виставці Consumer Electronics Show (CES). Компанія Rigetti оголосила про хмарну доступність для користувачів 19-кубітів в архітектурі з 20-кубітами, де один з них має дефект. Корпорація Google протестувала чіпи з шістьма, дев'ятьма і 20-кубітами і працює над архітектурою з 49 кубітів. Компанія Atos інформувала, що її 40-кубітний симулятор, заснований на процесорах Intel Xeon, в даний час додатково використовує спеціалізовані апаратні прискорювачі. IBM також використовує свої Power Systems для імітації квантових комп'ютерів під час розробки.

Для створення програмного забезпечення вже є розвинені devkits з метою залучення дослідників до квантових архітектур, орієнтованих на ефективне вирішення специфічних комбінаторних задач. IBM відкрила для загального користування QASM (Quantum ASseMbler), який є ключовим компонентом QISKit IBM (Quantum Information Software Kit) [23]. Засіб XACC (EXCEL ACCELERER) взаємодіє з симулятором і прототипом квантового комп'ютера Rigetti, а також з обчислювальними системами D-Wave. QuTiP (Quantum Toolbox in Python) [24] являє собою симулятор квантових обчислень з відкритим вихідним кодом, який використовується в сімействі кванто-

вих комп'ютерів (Alibaba, Amazon, Google, Honeywell, IBM, Intel, Microsoft, Northrup Grumman, Rigetti і RIKEN). QuTiP використовується для моделювання апаратних архітектур на стадії розробки. Google співпрацює з Rigetti на базі відкритого вихідного коду OpenFermion для компіляції з метою аналізу проблем квантової хімії. Microsoft представила користувачам мову квантового комп'ютингу Q # (Q-sharp).

Квантові комп'ютери використовують фізичні явища, що відбуваються в надзвичайно малих масштабах. Ринкова привабливість субатомної технології полягає в перевагах продуктивності і енергозбереження у порівнянні з класичними обчислювачами на базі напівпровідників. США щорічно витрачає близько 200 мільйонів доларів на дослідження і розробку квантових комп'ютерів, позиціонуючи їх як інфраструктуру для нової промислової революції. Японія, виділила близько 267 мільйонів доларів для квантових обчислень протягом десятиліття, починаючи з 2018 року [25]. Китай оголосив про створення Національної лабораторії квантових інформаційних наук вартістю 10 мільярдів доларів, яку планується відкрити у 2020 році. Компанії Alibaba, Baidu і Tencent інвестували свої фінансові можливості по 2017 року в AI і Deep Learning, тому слід очікувати їх інвестиційний інтерес до квантових обчислень у 2018 році.

Компанія Microsoft розпочала працювати над квантовим комп'ютингу майже два десятиліття тому, в 2000 році. На заході Ignite Microsoft оголосила про програму квантових обчислень на майораном-ферміонів (Majorana Fermions, 2012 рік), де масштабування логічної кількості кубітів є на 1-2 порядки більш економічним, де 10 фізичних кубітів доводиться на один логічний.

Microsoft фокусує свою активність на квантовому моделюванні нової мови Q #, інтегрованої у середовище розробки Visual Studio (IDE) і в інструменти моделювання квантових комп'ютерів, включаючи симулятор трасування для аналізу використання апаратних ресурсів. Microsoft також анонсу-

вав квантову оптимізацію Brainwave на базі FPGA-прискорювача для AI-рішень, що стосуються квантовізації глибокої нейронної мережі (DNN) для поліпшення точності і швидкості навчання. Інтеграція квантових обчислень з Visual Studio може залучити нове покоління академічних дослідників у бік від IDE з відкритим вихідним кодом, як Nvidia змогла зробити з програмуванням GPU через прикладний програмний інтерфейс CUDA (API).

IBM вже запустила свій QISKit API і devkit для розробників, щоб надати їм доступ до хмарних технологій Quantum Experience і локальним симуляторам. На додаток до цього IBM запустила проект з розвитку екосистеми Q Network. Учасниками проекту є регіональні центри освіти, дослідження, розробки і комерціалізації: Keio University, University of Melbourne, Oak Ridge National Laboratory (ORNL), University of Oxford, IBM Research, Daimler, JPMorgan Chase & Co, JSR, Samsung, Barclays, Honda, Materials Magic (Hitachi Metals Group), and Nagase.

Цільовими користувачами в Q-мережі IBM є аспіранти, академічні вчені та комерційні дослідники. Квантові обчислення все ще знаходяться на експериментальній стадії створення інфраструктури для програмування квантових комп'ютерів з метою вирішення практичних задач. Q-мережа використовується сьогодні у 1500 університетах, 300 приватних навчальних закладах і 300 вишах у межах їх навчальних програм з інженерії.

Попереду довгий шлях для комерціалізації квантових обчислень. У 2018 році ринок отримає системи з 50 або більше універсальними кубітами, а також спеціалізовані структури з 2000+ фізичними кубітами, що показують квантову перевагу у вузьких проблемних областях (International Conference on Quantum Communication, Measurement and Computing - QCMC).

Структура сучасного квантового комп'ютингу може бути представлена у вигляді пам'яті – хмар (множини) електронів, опромінюваних фотонами для формування двох примітивних станів $(0,1)$, ідентифікованих орбітою або спіном частинки, і одного похідного від них $X=\{0,1\}$. При цьому хмари

взаємодіють між собою, використовуючи логічні (теоретико-множинні) операції. Інша технологія використовує управління обертанням електрона у магнітному полі.

Спін електрона визначається, як незалежні від руху частинки у просторі, момент імпульсу, який має два напрямки (рис. 1.3) [26]: обертання частинки проти годинникової стрілки, яке формує імпульс, що співпадає з магнітним полем матеріалу (spin-up), що розглядається у квантовому комп'ютингі як логічна 1, і друге – обертання електрона за годинниковою стрілкою, яке формує імпульс, який протидіє вектору намагніченості (spin-down), що формує логічний 0. Якщо помістити тонку пластину немагнітного матеріалу між двома магнітами, то струм, що протікає, створює поле, яке здатне орієнтувати спіни електронів середньої пластини з арсеніду галію в одному з двох напрямків, які використовуються для зберігання 0 і 1 (Stuart Parkin). Формується нова галузь квантової електроніки – спінтоніка, яка, на відміну від переміщення зарядів в електричному полі, заснована на використанні спінів електронів, інваріантних до поступального руху частинок у просторі. Основна перевага спінтроніки пов'язана з незмінюваністю кінетичної енергії частинки, що є причиною відсутності тепловиділення при зміні спіна, на яке практично не потрібно витрат енергії. Майбутнє комп'ютингу полягає у використанні сонячного світла в денній фазі доби, достатнього для підтримки працездатності. Недоліком слід вважати малий період стійкості спіна, рівний 100 пікосекундам, чого поки недостатньо для фіксації стану у квантовому комп'ютингі. Однак вчені з IBM лабораторії вже розробили метод синхронізації електронів, збільшивши час стійкого існування спіна у 30 разів – до 1 наносекунди, що порівнянно з частотою мікропроцесора, яка дорівнює 1 GHz. Інший недолік пов'язаний з низькою температурою (мінус 233), при якій спіни електронів мінімально взаємодіють з навколишнім середовищем. Спінтроніка дає можливість управляти рухом елементарних магнітів у напівпровідникових пристроях і відкриває нові можливості для ство-

рення малогабаритної і енергозберігаючої електроніки для зберігання великих масивів даних у малому обсязі і паралельних квантових обчислень.

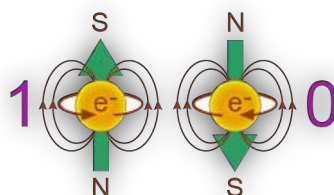


Рисунок 1.3 – Логічні стани спинів електрона у магнітному полі

Архітектура сучасного квантового комп'ютера використовує спін або орбіту електрона для зберігання логічних станів, перемикання яких управляється електромагнітними полями або фотонами у середовищі, близькому до абсолютного нуля. Така архітектура має два результати для проектування обчислювальних алгоритмів: 1) Реалізація квантової логіки на основі операцій суперпозиції і змішування для синтезу паралельних обчислювачів. 2) Використання операцій транзакції (read-write), вільних від квантової логіки, для програмування паралельних алгоритмів.

Суперкомп'ютер «Саміт», побудований ІВМ в 2018 році, займає площу, рівну двом тенісним кортам, і споживає 4 000 галонів води у хвилину для охолодження 37 000 процесорів. «Саміт» може забезпечити максимальну продуктивність 200 квадрильйонів обчислень у секунду (200 і 15 нулів) або 200 петафлоп. Це приблизно в мільйон разів швидше, ніж ноутбук, і майже в два рази перевищує максимальну продуктивність самого популярного в Китаї Sunway TaihuLight. Суперкомп'ютер містить 28 000 графічних процесорів виробництва Nvidia і більш, ніж 9000 процесорів від ІВМ. Інтенсивне використання графічних чіпів, незвичайне для суперкомп'ютера, дозволяє забезпечити прориви в розгортанні машинного навчання з крутих наукових проблем, – каже Томас Захарія, директор Oak Ridge National Lab. «Ми побудували найпотужніший і найрозумніший в світі суперкомп'ютер» [27].

Що стосується енергоспоживання, то майбутнє комп'ютингу полягає в використанні сонячного світла в денний фазі доби, достатнього для підтримки працездатності. Аргументом на користь цієї тези може слугувати публікація Euisik Yoon, Sung-Yun Park в журналі IEEE Spectrum 2018Apr "Self-Powered Image Sensor Could Watch You Forever" [28].

Основна ідея полягає в тому, що сонячні елементи (solar cells) і датчики зображення (image sensors) перетворюють світло в електрику. Якщо обидва компонента розмістити в одному чіпі, то можна отримати автономну камеру (self-powered camera), що працює тільки від денного світла, яка здатна захопити 15 зображень в секунду. Мікросистема для відеозйомки і збору енергії, інтегрована з мікромініатюрним процесором і бездротовим приймачем, дає можливість поміщати маленьку, майже невидиму, камеру в будь-якому місці.

Інтегровані кремнієво-оптичні чіпи можуть усунути проблеми, пов'язані з низькою швидкістю металевих з'єднань між схемними компонентами. Optical CMOS process could break communications bottleneck.

Мілош Попович [29], професор електротехніки та обчислювальної техніки в Бостонському університеті разом з колегами з Массачусетського технологічного інституту, Каліфорнійського університету Берклі, опублікували в Nature новий спосіб передачі оптичних сигналів на звичайних мікросхемах. Метод дозволяє прискорити на порядок зв'язок між мікропроцесорами, істотно зменшуючи тепловіддачу і збільшуючи обчислювальну продуктивність ноутбуків і смартфонів. Технологічна ідея полягає у додаванні матеріалу – тонкого шару діелектричного полікристалічного кремнію зверху існуючих компонентів кристала, виконаного за об'ємною комплементарною технологією. Щоб зробити матеріал більш відповідним для фотоніки, дослідники модифікували кристалічну структуру для запобігання витоку світла з їх полікристалічного кремнію на підкладку. Отримані кристали володіють усіма необхідними фотонними компонентами: хвилеводами, мікрорезонаторами, вертикальними ґратчастими сполуками, високошвидкісними модуляторами,

лавинними фотоприймачами і транзисторами, виконаними поки ще за технологією 65 нм. Джерело лазерного випромінювання знаходиться поза чіпа. Фотодетектори поглинають фотони. Мотивація дослідження полягає в тому, що виробники комп'ютерів все частіше використовують чіпи і графічні процесори для створення ігор і штучного інтелекту, які можуть містити сотні ядер. При цьому мідні дроти, що сполучають ядра, є основним вузьким місцем, стримуючим високу швидкодію, які до того ж виробляють велику кількість тепла, що вимагає відводу у зовнішнє середовище. Металевий провідник може переносити дані, від 10 до 100 гігабіт на секунду, а оптичне волокно - 10 до 20 терабіт в секунду. Слід також мати на увазі, що на мікровідстанях між мікропроцесорами теплові оптичні втрати практично дорівнюють нулю, тому опто-кремнієва система вимагає менших витрат енергії, ніж мідно-кремнієва. Новий метод може привести до створення чіпів з підвищеною обчислювальною потужністю для апаратної реалізації методів штучного інтелекту при розпізнаванні образів в iPhone і в недорогих датчиках LIDAR для driver-free автомобілів.

Пропонується однофотонний вентиль, як варіант створення твердотільного оптичного транзистора, де пропускання світла може управлятися одним фотоном, який діє як затвор або перемикач [30]. Використовується твердотільна система, що містить квантову точку, вбудовану в резонатор фотонного кристала, що дає можливість управляти світловим потоком в порожнині резонатора за допомогою одного фотона. Останній використовується для управління електронними рівнями енергії в квантовій точці, що змінює її оптичні властивості. Коли вентиль відкривається, в середньому через порожнину резонатора може проходити близько 28 фотонів, що демонструє однофотонне перемикання і посилення світлового потоку для оптичного транзистора.

Однофотонні перемикачі та транзистори генерують сильні фотон-фотон інтеракції, які необхідні для створення квантових схем і мереж. Однак для

детермінованого управління оптичним сигналом за допомогою одного фотона необхідно взаємодіяти з квантовою пам'яттю, що важко досягти на твердотільній платформі. Проте автори демонструють однофотонний перемикач і транзистор, що взаємодіють з твердотільною квантовою пам'яттю. Пристрій складається з напівпровідникового спіно-кубіта, сильно пов'язаного з нанофотонною порожниною. Спіновий кубіт дозволяє одному 63-пікосекундному вентиляльному фотону переключити множину сигналів, що містить в середньому 27,7 фотонів, до того, як внутрішній стан пристрою буде змінено. Результати показують, що напівпровідникові нанофотонні пристрої можуть створювати керовані фотон-фотон інтеракції, як фізичної основи фотонного квантового комп'ютингу для обробки інформації з високою пропускну здатністю.

«У цифровому світі екзабайти даних генеруються компаніями з виробництва, енергетики, телекомунікацій. Обсяг даних за останні 8 років виріс у 15 разів, а обчислювальна потужність центрів обробки даних збільшилася всього у 4 рази. Згідно з прогнозами IBM, кількість пристроїв, підключених до Інтернету, збільшиться у 7 разів за 3 роки і досягне 50 млрд. У 2020 році [31, 32]. Ця реальність збільшить розрив між обчислювальними потужностями центрів обробки даних і генеруються даними. Для вирішення проблем зберігання і аналізу даних IoT компанії реплікують сервери баз даних, що призводить до складної інфраструктури, збільшення витрат, простору і енергоспоживання. Середня європейська компанія, яка виробляє продукцію, з річним доходом 50 млн євро, витратить 158 тис. євро на створення дата центрів і 85 тис. євро на рік для її підтримки. Фірма Grovf пропонує інший архітектурний підхід для вирішення проблеми, де використовується FPGA структура як процесор. GData зростає швидше, ніж будь-коли раніше, і до 2020 року близько 1,7 мегабайт нової інформації буде створюватися кожен секунду для кожної людини на планеті. Відповідно до нового звіту IBM Marketing Cloud «10 ключових тенденцій в маркетингу на 2017 рік», 90% да-

них в світі сьогодні створено за останні два роки. Сервер баз даних Alogovf використовує FPGAAcceler для обробки даних NoSQL, що надходять від датчиків IoT. Прототип вже досяг 10-кратного прискорення транзакцій і 31-кратної економії енергії у порівнянні з найсучаснішими існуючими апаратними рішеннями. Основними цілями запропонованої архітектури є: 1) Впровадження нової, дізрапторної технології для управління даними в індустрії IoT. 2) Різке зниження енергоспоживання датацентров. 3) Забезпечення швидкої, недорогої і надійної локально-хмарної технології для індустрії IoT. 4) Оптимізація поточної завантаженості існуючих центрів обробки даних. 5) Допомога середнім виробникам-компаніям в енергетиці, логістиці та телекомунікаціях у скороченні витрат на IT-інфраструктуру при створенні конкурентоспроможних підприємств.

1.4. Постановка мети і завдань дослідження

Аналітичний огляд публікацій в області тестування і верифікації спеціалізованих логічних схем шляхом розробки паралельних квантових методів аналізу і синтезу кубітних структур даних цифрових систем дає підстави зробити наступні висновки: 1) Тенденція квантового комп'ютингу підтримується в актуальному стані протягом останніх п'яти років, про що свідчить Gartner Hype Cycle 2018. 2) Практично всі провідні компанії IT-галузі вкладають ресурси в розробку ринково затребуваного квантового комп'ютера на основі використання субатомних структур даних і логічних відносин між ними. 3) Як мінімум, 30 відсотків коштів припадає на розробку паралельних алгоритмів, орієнтованих на імплементацію в квантові обчислювачі, які з'являться у найближчі 5 років. 4) Особливий ринковий інтерес викликають алгоритми паралельного вирішення завдань комбінаторики за напрямками: Design and Test, Cyber Security, Bio-engineering, Ecology, Transportation. 5) Ринок також цікавиться імплементацією паралельних квантових алгоритмів на кубітних структурах даних, емульованого в класичні комп'ютери, які мають більш високу швидкодію за рахунок використання надлишкової пам'яті.

б) Актуальними також залишаються дослідження в області *memory-driven* комп'ютерів, без використання окремого процесора з шиною обміну даними, яка істотно зменшує швидкодію. Виходячи з представленого вище, далі формуються мета і завдання дослідження, спрямовані на вирішення науково-практичного завдання синтезу і моделювання квантових паралельних алгоритмів технічної діагностики на класичних комп'ютерах:

Мета дослідження – зменшення часу верифікації цифрових систем на кристалах шляхом використання *memory-driven* архітектур і кубітних структур даних для компактного опису логічних *X*-функцій і істотного підвищення продуктивності методів тестування і дедуктивного моделювання несправностей за рахунок паралельного комп'ютерного алгоритмів.

Завдання дослідження:

1) Розробити *memory-driven* архітектури та алгоритми для паралельного виконання логічних операцій над кубітними структурами даних при реалізації методів тестування та верифікації цифрових систем на кристалах.

2) Удосконалити квантові методи генерації тестів і дедуктивного моделювання несправностей логічних функцій шляхом синтезу матриць булевих похідних за їх кубітним покриттям.

3) Розробити структурну модель метричних властивостей *X*-функцій, орієнтовану на виконання паралельних операцій, з метою отримання лінійного часу генерації тестів і моделювання цифрових систем.

5) Розробити аналітичну модель для отримання кубітних покриттів *X*-функцій від кінцевого числа змінних з метою створення логічних схем, які не потребують експоненційних витрат на синтез і аналіз тестів.

4) Розробити паралельний метод генерації тестів для несправностей *X*-функцій від кінцевого числа змінних на основі взяття похідних за кубітним покриттям, які формують тести мінімальної довжини.

б) Розробити паралельний метод синтезу дедуктивних кубітних покриттів для моделювання Х-функцій на основі отримання одиничної матриці похідних з метою створення секвенсора моделювання дефектів.

Виконання досліджень і проведення експериментів ґрунтується на використанні метрики параметрів, інтегрованої в функцію мети:

$$L = \min \left(\sum_{i=1}^n T_i \times P_i^T + \sum_{i=1}^n M_i \times P_i^M + \sum_{i=1}^n C_i \times P_i^C \right),$$

$$T = (T_{\text{mod}}, T_{\text{tpg}}, T_{\text{tver}}, T_{\text{sim}}); n = 4;$$

$$M = (M_{\text{mod}}, M_{\text{tpg}}, M_{\text{tver}}, M_{\text{sim}});$$

$$C = (C_{\text{mod}}, C_{\text{tpg}}, C_{\text{tver}}, C_{\text{sim}}).$$

Формула показує зменшення часу T проектування цифрових систем на кристалах і забезпечення якості (обчислювальної складності) алгоритмів C за рахунок використання надлишкової пам'яті M для зберігання кубітних структур даних, які орієнтовані на паралельне виконання логічних операцій при синтезі і аналізі логічних схем, пов'язаних з тестуванням і верифікацією. Представлені вартісні характеристики P чотирьох основних процесів, пов'язаних з modeling (mod), test pattern generation (tpg), test verification (tver), test simulation (sim).

Результати розділу відображено у роботах [4, 12, 22].

РОЗДІЛ 2

АРХІТЕКТУРИ КВАНТОВОГО І КЛАСИЧНОГО КОМП'ЮТИНГУ

Мета – розробка архітектур і методів квантового паралельного аналізу кубітних моделей цифрових систем і компонентів для їх подальшої реалізації в формі сервісів-додатків.

Завдання: 1) Синтез квантових memory-driven архітектур паралельного комп'ютингу для аналізу кубітних моделей цифрових систем і компонентів. 2) Розробка метрики і моделі класифікації квантового комп'ютингу. 3) Створення карти або інфраструктури quantum design and test of memory-driven computing. 4) Розробка метричних моделей для спільного використання квантового та класичного комп'ютингу. 5) Створення кубітних квантових архітектур для паралельних обчислень при вирішенні задач design and test.

2.1 Memory-driven архітектури

Класичні комп'ютери, засновані на архітектурі фон Неймана (рис. 2.1, а), реалізують обчислення в процесорі ($CPU = ALU + CU + Cache$) на основі інтерактивного обміну даними по шині між пам'яттю і CPU [33]. Вузким місцем тут є шина передачі даних, яка істотно зменшує швидкодію при обробці великих обсягів інформації на основі характеристичного рівняння $A = f(A)$, де A - дані, які зберігаються в пам'яті і записуються в неї після арифметико-логічних обчислень f , виконуваних в ALU.

Створити альтернативну big data архітектуру комп'ютингу означає поєднати в просторі функції обчислення і зберігання шляхом побудови обчислювальної пам'яті (рис. 2.1, б). Такий комп'ютинг дає можливість в кілька порядків підвищити швидкодію обробки даних за місцем їх зберігання за рахунок усунення bottleneck (обмін даними між пам'яттю і ALU), а також в

кілька разів зменшити енергоспоживання і розміри обчислювальних пристроїв.

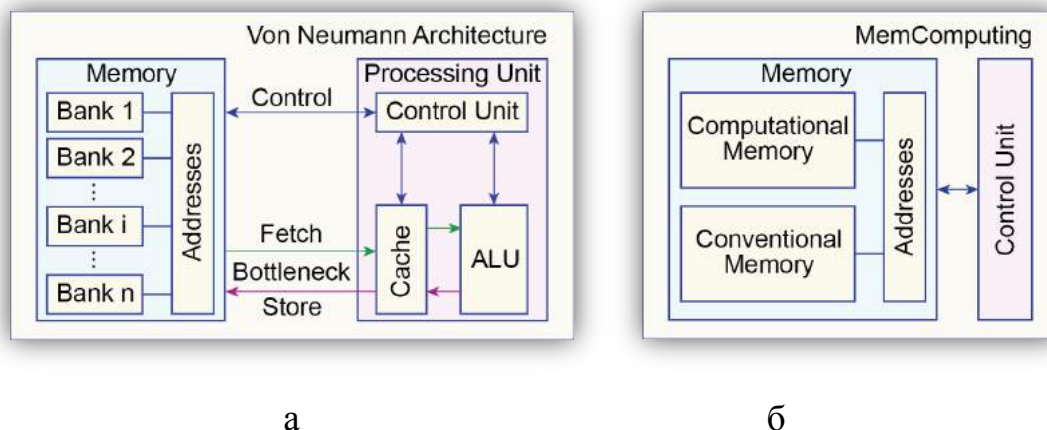


Рисунок 2.1 – Архітектура фон Неймана і MemComputing

Для зберігання даних зазвичай використовується динамічна пам'ять з довільним доступом (DRAM) на основі ефекту зарядженого стану конденсатора. Реалізація арифметичної або логічної операції пов'язана з виконанням наступної послідовності пунктів алгоритму: зчитування даних з пам'яті, передача їх у процесор, обробка даних на основі арифметико-логічної операції, запис даних назад в пам'ять. Даний алгоритм має недолік, пов'язаний з переміщенням (прийомом-передачею) значних обсягів даних між двома фізично і просторово розділеними компонентами архітектури (пам'ять і процесор), що вимагає істотних витрат часу і енергії. Усунути недолік слід шляхом переходу до гібридної архітектури – обчислювальної пам'яті, де операції можуть виконуватися за місцем зберігання даних. Одним із прикладів реалізації такої пам'яті є нанотехнологічні мемрісторні пристрої, що використовують резистивну незалежну пам'ять, де інформація зберігається в станах опору або провідності. Тут пристрої пам'яті використовуються для зберігання вхідних даних, виконання логічних операцій і збереження отриманих результатів. Згодом складніші логічні блоки були використані для виконання векторно-матричних операцій в аналоговий спосіб. Далі з'являється обчислювальна

пам'ять для реалізації високорівневих обчислювальних примітивів, що використовують динамічну поведінку складових компонентів для виконання обчислень на місці зберігання даних. Обчислення ініціюються шляхом подачі електричних сигналів управління від Control Unit до пам'яті пристрою. Автори пропонують алгоритм для визначення часової кореляції між event-based потоками даних з використанням Phase Change Memory (PCM) обчислювальної резистивної пам'яті (resistive memory). Проведено великомасштабні експерименти з використання масиву з мільйона PCM-пристроїв для обробки великих даних по погоді в масштабах регіонів планети. Пристрій або мемристор PCM складається з об'ємного матеріалу нанометрового масштабу зі зміною фазового стану між двома електродами. У спочатку виготовленому пристрої матеріал знаходиться у кристалічній фазі високої провідності. При впливі на мемристор імпульсом струму досить великої амплітуди значна частина матеріалу змінюється на розплавлену аморфну фазу, завдяки джоулевому нагріванню. Після закінчення імпульсу матеріал переходить у фазу застигання або низької провідності.

Приклад швидкодіючої обчислювальної системи для аналізу великих даних, що включає комп'ютер, FPGA-плату і PCM-кристал пам'яті, представлений на рис. 2.2.

Модуль PCM пов'язаний з апаратною платформою, що включає FPGA і блок аналогового інтерфейсу (AFE – analog-front-end board). Плата AFE забезпечує необхідні параметри по напрузі і струму для електричного живлення кристала PCM. Плати FPGA використовуються для реалізації системного моніторингу та управління даними, а також створюють інтерфейс зв'язку з блоком обробки даних. Експериментальна платформа працює під управлінням комп'ютера, де Matlab-середовище використовується для координації обчислювальних процесів при виконанні експериментів.

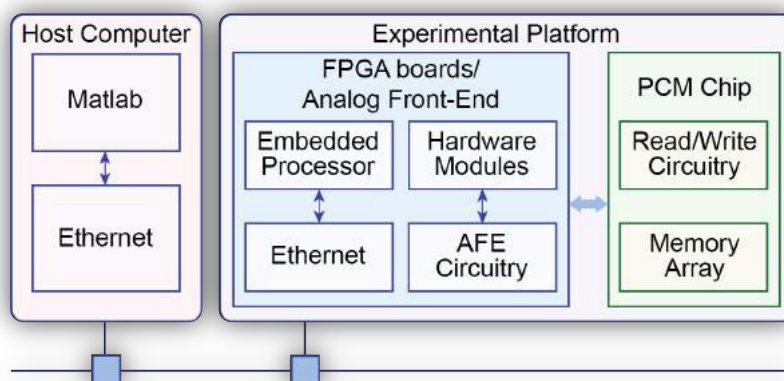


Рисунок 2.2 – MemComputing система на основі PCM-пам'яті

Автори показують, як динаміка кристалізації PCM пристрою може бути використана для виявлення статистичної кореляції між подійними потоками даних. Технологія може застосовуватися в різних областях: IoT, life sciences, networking, social networks, big data analysis, business processing, and large scientific experiments. Наприклад, можна створити потік даних на основі подій, що визначаються наявністю або відсутністю ключового слова в колекції твітів, електронної пошти, скріншотів. Обробка в реальному масштабі часу подієвих потоків даних, що надходять від динамічних відео-сенсорів або моніторів комп'ютерів, є ще однією перспективною сферою застосування обчислювальної пам'яті. Використовуючи модуль обчислювальної пам'яті, можна прискорити задачу виявлення кореляції на потоках даних у 200 разів у порівнянні з реалізацією алгоритму, який використовує 4 сучасних GPU-пристрої. Показано, що одношарова пікова нейронна мережа (single-layer spiking neural network) може також використовувати обчислювальну пам'ять memcomputing для визначення часових кореляцій потоків даних, заснований на подіях. PCM-мемристори можуть служити синаптичними елементами. Вірним є і зворотне, що синаптичні елементи є деякою формою обчислювальної пам'яті. Пристрої PCM були інтегровані в чіп 90 нм технології CMOS³². Матеріал з фазовою зміною є легованим кристалом Ge₂Sb₂Te₂. На

чипі доступні два блоки: перший містить 2 мільйони РСМ пристроїв, другий - 1 мільйон. Загальний час зчитування становить 1 мкс.

Модифікація memory-driven комп'ютингу полягає у внесенні блоку управління до пам'яті (рис. 2.3, а), що дає можливість виключити неоднорідність архітектури і привести її до чистої пам'яті, на якій реалізуються модулі управління, комп'ютингу і зберігання даних до і після обробки.

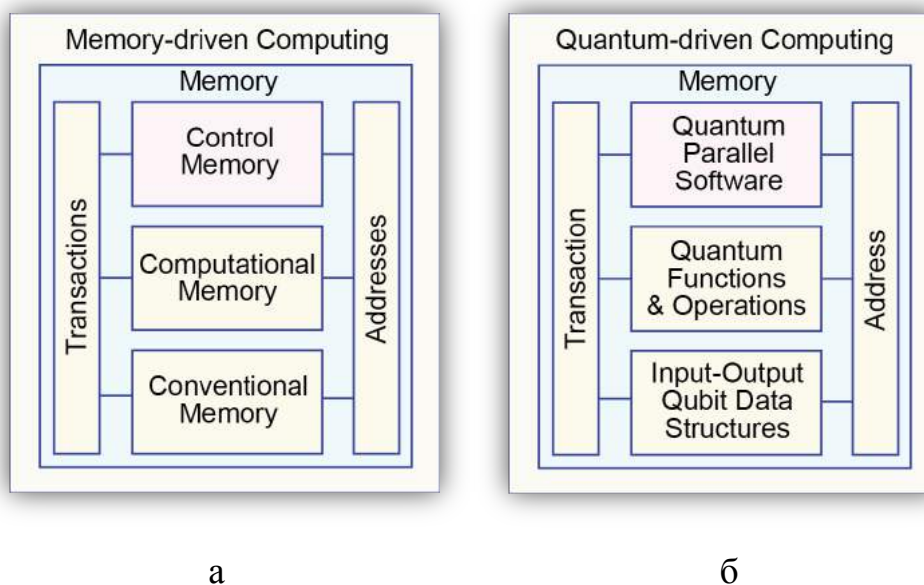


Рисунок 2.3 – Architectures: memory-driven and quantum computing

Структура, зображена на рис. 2.3, б, спрямована на підвищення швидкодії при вирішенні комбінаторних проблем за рахунок високого паралелізму квантових алгоритмів на основі використання кубітних структур даних.

Memory-driven computing містить Computational Memory, яка представлена покриттями функціональних примітивів, що реалізують логічні (арифметичні) операції. Control Memory являє собою алгоритм управління обчислювальним процесом, який зчитує дані з Conventional Memory, обробляє їх за допомогою Computational Memory і записує результати в регулярну пам'ять для зберігання. Така структура комп'ютингу на пам'яті є типовою, яку можна поширити на рішення практичної задачі, пов'язаної з моделюванням цифрових схем на основі кубітних структур даних, рис. 2.4, а.

Представлена архітектура пам'яті містить компоненти: структура-алгоритм, функції-операції, дані вхід-вихідні. Зокрема, memory-driven SIMD (Single Instruction Multiple Data) архітектура комп'ютингу орієнтована на паралельну обробку даних, характерну для quantum-driven аналізу цифрових схем шляхом використання характеристичного рівняння $M = Q [M (X)]$. Тут блок control memory представлений згаданим вище рівнянням; computational memory – кубітними покриттями, що формують функціональні примітиви; conventional memory – вхідними та вихідними даними до і після обробки. Іншими словами, створити структури даних, орієнтовані на Quantum Computing, означає закодувати стани і процеси в унітарний код, що дає можливість застосовувати паралельні логічні операції над множинами однотипних даних. Таким чином, архітектура для ALU-free обчислювального процесу, що використовує тільки пам'ять, на якій реалізуються адресні транзакції, формує MemComputing, який в структурному виконанні звучить як MAT-Computing (Memory-Address-Transaction). Інший приклад комп'ютингу на пам'яті представлений на рис. 2.4, б, який орієнтований на паралельну обробку даних з метою істотного підвищення швидкодії при вирішенні задач розпізнавання образів, текстів, зображень і фігур.

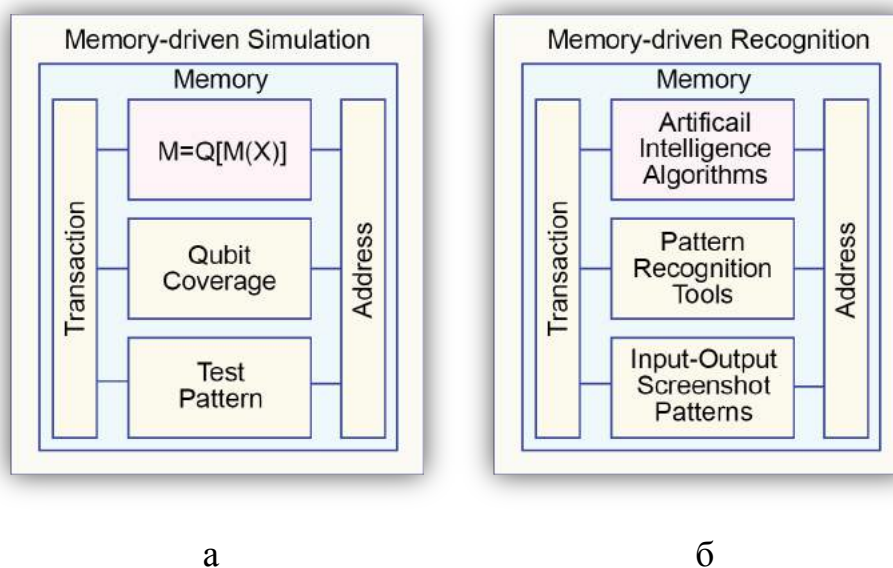


Рисунок 2.4 – Memory-driven quantum computing: Simulation and Recognition

Таким чином, поєднання квантового комп'ютингу з архітектурою обчислювальної пам'яті надає науці і практиці нову технологічну культуру *quantum memory-driven computing*, яка інтегрально отримує переваги структурної однорідності і паралелізму обробки великих даних за рахунок усунення транзакцій між пам'яттю і ALU-процесором, і виключення квантових операцій суперпозиції і змішування.

2.2 Класифікація квантового комп'ютингу

Інтегрально класифікація квантового комп'ютингу за такими чинниками апаратної, структурної і програмної реалізацій представлена на рис. 2.5. Вона узагальнює світові розробки вчених університетів і провідних компаній планети в частині фізичних основ квантового комп'ютингу, логіки його роботи, а також вже створені програмні продукти і компілятори-симулятори, орієнтовані на рішення комбінаторних, оптимізаційних, криптографічних та інтелектуальних проблем [34].

Практично, всі провідні компанії планети з NASDAQ-списку вважають за необхідне вкладати інвестиції в розробку і створення власних квантових комп'ютерів, а також програмних продуктів для їх обслуговування. На стадії проведення експериментів переваги окремих компаній є практично незначними. На ринку квантових обчислень, що виникає, швидше за все переможе гравець, який інвестує максимальну кількість фінансів в *memory-driven quantum computing*. Наприклад, Apple. Звичайно, вона займається даною тематикою без видимої реклами своїх досягнень. Проте компанія не буде ризикувати сьогодні з виходом експериментальних квантових зразків. У неї все добре зараз і без Q-комп'ютерів. Тим не менш, вона естетично красиво оформить вже апробований на ринку квантовий комп'ютинг в масовий продукт і зробить його доступним для всієї планети, випередивши просунутих сьогодні в цій галузі конкурентів.

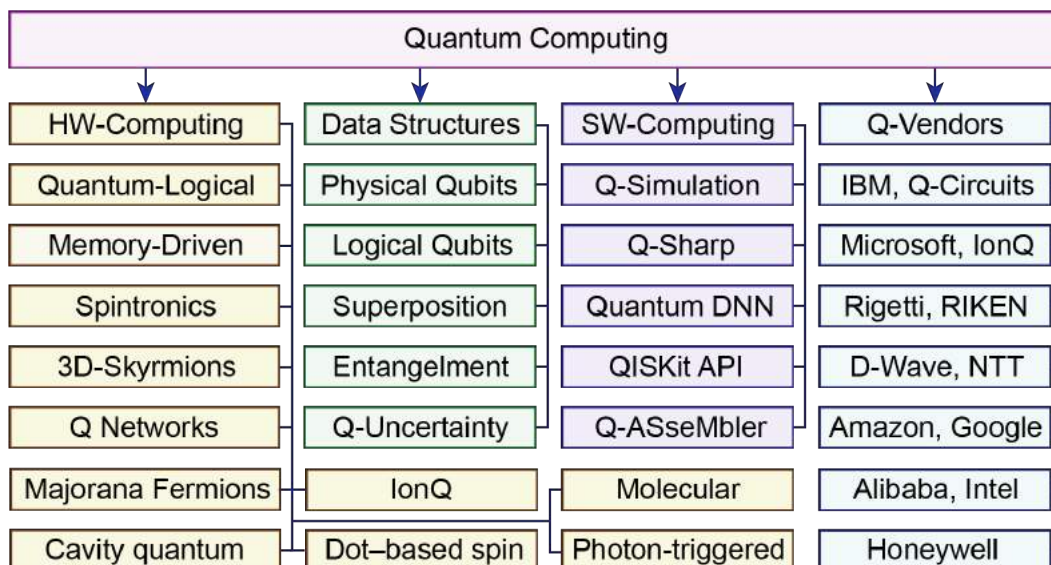


Рисунок 2.5 – Класифікація квантового комп'ютингу

Вже сьогодні можна виділити три комерційно життєздатних застосувань для ранніх квантових обчислювальних пристроїв – квантове моделювання, оптимізацію і комбінаторику для областей: охорони здоров'я, штучного інтелекту, кіберзахисту даних і хмарних сервісів, криптографії, транспорту, хімії, погоди [35]. Поки невідомо, чи зможуть існуючі (послідовні) алгоритми та програми збільшити свою продуктивність, використовуючи квантові процесори, які скоро будуть доступні споживачам. Тому сьогодні слід розробляти нові паралельні типи алгоритмів квантового моделювання, криптоаналіза, глибокого машинного навчання, оптимізації шляхом використання класичних, а також квантових процесорів Google, IBM, IonQ, доступних за допомогою хмарних сервісів [36, 37].

Кіберкультура – технологічна досконалість і соціальна значимість моральних відносин між розумним суспільством, фізичним світом і кіберпростором, яке визначається використанням інтернет-сервісів для цифрового моніторингу та надійного метричного управління процесами в усіх сферах людської діяльності, включаючи науку, освіту, охорону здоров'я, виробництво і транспорт, з метою поліпшення якості життя людей і збереження екосистеми планети (рис. 2.6, а).

Кіберкультура (коротко) – технологічна досконалість суспільства, спрямована на використання інтернет-ресурсів для морального управління фізичним світом і соціальними процесами без участі людини.

Квантова кіберкультура – технологічна та моральна досконалість у відносинах між парком ринково доступних квантових комп'ютерів, потужним банком квантових паралельних алгоритмів-додатків і освіченим, паралельно мислячим, суспільством, спрямована на ефективне вирішення комбінаторних завдань (рис. 2.6, б).

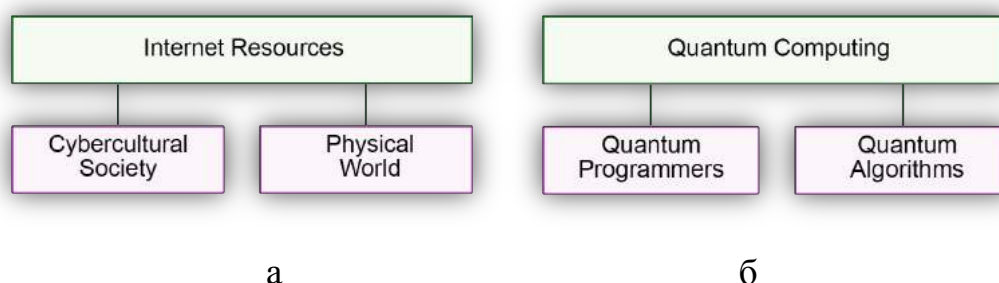


Рисунок 2.6 – Кіберкультура (а) і квантова культура (б) відносин

Кіберкультура квантового memoгу-driven комп'ютингу інтегрує в трикутну структуру технології паралельного рішення часовитратних комбінаторних задач, ринково доступні квантові обчислювачі і паралельно мислячих програмістів. Масштабована карта досліджень, відповідних квантовій кіберкультурі, що націлена на створення паралельних алгоритмів, зокрема, для SoC Design and Test, представлена на рис. 2.7.

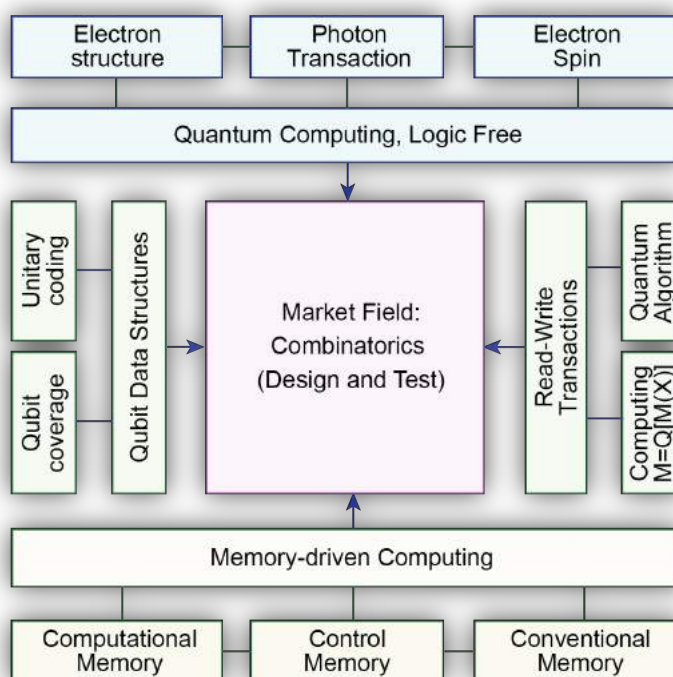


Рисунок 2.7 – Roadmap of quantum design and test memory-driven computing

Комп'ютинг – обчислювальний процес, представлений транзакціями запису-зчитування даних на адресованій пам'яті [10]. Таке визначення на найнижчому рівні прибирає теоретичні бар'єри, створювані теоремою Поста і повнотою функціонального базису, усуває логіку (ALU) і шини передачі даних між процесором і пам'яттю в класичному комп'ютингу. Більш того, транзакційний (MAT – Memory-Address-Transaction) комп'ютинг настійно пропонує відмовитися від квантової логіки, реалізація якої викликає технологічні проблеми при створенні квантового комп'ютера. Цілком достатньо реалізувати фотонні транзакції запису-зчитування даних на стійкій структурі атомарних електронів, що виконуються зі швидкістю світла.

Memory-driven комп'ютинг – обчислювальний процес, який реалізується в пам'яті, що містить три компонента архітектури: control-, computational- and conventional memory. Memory-driven класичний комп'ютинг сьогодні має істотний недолік, який полягає в тривалому циклі (read-write) звернення до пам'яті, який в десятки разів більше, ніж час виконання логічної операції.

Проте реалізація *memory-driven* комп'ютингу на квантових (фотонних) транзакціях в атомарній структурі електронів усуває низьку швидкість звернення до пам'яті, завдяки світловій швидкості фотонного обміну даними, що дає можливість прибрати з комп'ютингу квантову логіку, що технологічно складно реалізовується.

Квантовий комп'ютинг – галузь знань, що займається теорією і практикою паралельного рішення комбінаторних задач на обчислювачах, які використовують субатомні частинки при створенні структур даних і їх фізичної взаємодії для реалізації логічних операцій.

Квантовий комп'ютинг, що емулює – структури даних, моделі, методи і алгоритми для створення *software* додатків з метою паралельного рішення комбінаторних задач на класичних комп'ютерах шляхом використання додаткової пам'яті.

Квантовий комп'ютинг – обчислювальний процес, який використовує фотонні транзакції в атомарній структурі електронів для реалізації паралельних алгоритмів і програмних додатків. Квантовий комп'ютер без квантових паралельних алгоритмів – всього лише дорога іграшка. Тому стратегія створення квантового комп'ютингу полягає в одночасній розробці квантової апаратури і квантових програмних додатків на основі паралельних алгоритмів. Паралельний і роздільний розвиток двох гілок квантового комп'ютингу надає можливість науковцям з країн, що розвиваються, активно брати участь в проектуванні, моделюванні та верифікації квантових алгоритмів і програмних додатків на класичних комп'ютерах з метою їх подальшої імплементації в ринково доступні квантові комп'ютери в недалекому майбутньому.

Метрика квантового та класичного комп'ютингу не має суттєвих структурних відмінностей. На жаль, але в наукових публікаціях відсутнє метричне порівняння квантової і класичної логіки. Природно, існує аналогія між квантовими операціями на кубітних структурах даних і теоретико-множинними операціями на символах алгебри Кантора. Ізоморфізм двох структур: кванто-

вої логіки і алгебри множин полягає в подобі носіїв і сигнатур. Взаємно-однозначна відповідність носіїв визначається відношеннями між булеаном і кубітними структурами даних [38]:

Boolean $A^k =$	0	1	$X = 0 \cup 1$	$\emptyset = 0 \cap 1$
Qubit $ \psi\rangle =$	$ 0\rangle$	$ 1\rangle$	$\alpha 0\rangle + \beta 1\rangle$	$\alpha 0\rangle \beta 1\rangle$

Ізоморфізм сигнатур визначається взаємно-однозначною відповідністю між двома операціями: об'єднання – суперпозиція, доповнення – переплутування. Для операції перетину відповідний аналог у квантовій логіці не визначений. При цьому унітарне кодування символів алфавіту Кантора дає можливість паралельно виконувати будь-які логічні операції навіть на класичному комп'ютері. Але дізрапторне інноваційне рішення полягає у відмові від операцій суперпозиції і змішування в сторону створення memory-driven квантового комп'ютингу, що використовує тільки операції запису-зчитування на пам'яті, де виконавчі стани реалізовані, наприклад, у спінових моментах електронів.

Моделювання на класичних комп'ютерах паралельних квантових алгоритмів дає істотний приріст продуктивності за рахунок використання надлишкової пам'яті для унітарного кодування кубітних структур даних. Більш того, моделювання квантових алгоритмів стратегічно покликане вирішувати проблему створення квантового інтелекту планети шляхом розробки банку паралельних програмних додатків для майбутнього парку ринково доступних квантових комп'ютерів.

Design and Test є найбільш передовою областю діяльності вчених і компаній, спрямованої на створення нових технологій комп'ютингу за метрикою: швидкодія, енергозбереження, компактність і надійність. Параметр time-to-market при створенні комп'ютингової продукції, поряд з якістю, є домінуючим. Тому квантові паралельні методи та алгоритми проектування, тестування, верифікації, моделювання і діагностування цифрових виробів на кубітних структурах даних є актуальними при їх імплементації в сучасні кла-

сичні і в майбутні квантові комп'ютери, з метою істотного зменшення time-to-market [39]. Особливий інтерес представляє використання кубітних покриттів примітивних функціональностей для мінімізації, синтезу, аналізу і діагностування цифрових систем на кристалах. Розглядаються логічні (xor, notxor) X-функції, що задаються симетричними кубітними покриттями, для технологічного паралельного вирішення завдань синтезу тестів і дедуктивного моделювання несправностей. Синтез дедуктивних формул для X-функцій формує компактну структуру для транспортування вхідних списків несправностей, яка інваріантна до вхідних тестових наборів. Це робить X-функцію привабливою для її використання при проектуванні тестопригодності цифрових систем. Унікальність X-функцій проявляється також у технологічній простоті процедури взяття булевих похідних на основі зустрічного зсуву двійкових розрядів, що є основою для генератора тестів, що використовує кубітні покриття логічних схем.

Запропоновано квантовий метод синтезу тестів для X-функцій від n змінних. Мінімальний тест для кожної X-функції містить 5 вхідних наборів, які покривають всі поодинокі константні несправності, перевірені за допомогою дедуктивного моделювання по кубітним покриттям функцій. Тест для X-функції від n змінних будується на основі використання кубітних покриттів шляхом об'єднання всіх одиничних координат кубітного вектора і однієї нульовою координати вектора. Довжина тесту для X-функції завжди визначається половиною довжини кубітного покриття $L(T) = \frac{1}{2} * (2^{**n}) + 1$.

Представлена таблиця інтегральних властивостей X-функцій для вирішення завдань Design and Test на основі використання кубітних структур даних. Приведена також структура секвенсора і аналітичні вирази для швидкого синтезу X-функції будь-якої розмірності від n -змінних шляхом виконання операцій конкатенації і інверсії.

Завдяки кубітній формі опису дедуктивних функцій, з'явилася можливість синтезувати дедуктивну матрицю для транспортування несправно-

стей, яка характеризується високим рівнем компактності структур даних і технологічністю її отримання на основі not-операції і процедури перестановки бітів в кубітному векторі-стовпці. Нова теорія дедуктивного аналізу цифрових систем на основі кубітних покриттів використовує найпростіші матричні операції, пов'язані з перестановкою і конкатенацією, для рекурсивного отримання як завгодно складної структури даних в цілях дедуктивного аналізу логічних схем, які використовують X-функції. Отримувані матриці мають властивості вертикальної, горизонтальної та діагональної симетрії, завдяки внутрішнім властивостям X-функцій.

Основним результатом представленого аналізу X-функцій слід вважати компактність кубітних структур даних, яка забезпечує високу швидкодію методів синтезу тестів для цифрових пристроїв та аналізу їх якості на основі дедуктивного моделювання несправностей.

Одним з основних досягнень кубітного покриття є паралелізм інтерпретативного моделювання справної поведінки цифрових систем. Метод кубітного або квантового моделювання використовує одне характеристичне рівняння, що оперує при транзакціях даних тільки адресами кубітного покриття і вектора моделювання, який створює схемну структуру. Секвенсор синтезу тестів, що використовує логічні операції: інверсії, зсуву, порівняння і об'єднання, є прототипом апаратної реалізації модуля, вбудованого в інфраструктуру SoC, для online-тестування функціональностей. Такий апаратний модуль може бути доступний, як хмарний сервіс з IP-адресою, який створює прямий зв'язок з будь-якою функціональністю в масштабах кіберпростору планети.

Головним досягненням пропонованого дослідження є створення основ теорії і методів квантового memory-driven проектування і тестування цифрових систем, орієнтованих на паралельні алгоритми синтезу та аналізу кубітних структур даних, імплементованих в emerging memory-driven quantum computing.

2.3 Метрика класичного і квантового комп'ютингу

Метою створення цифрового світу є розумний кіберфізичний і соціальний комп'ютинг вичерпного цифрового моніторингу та метричного інтелектуального управління всіма процесами та явищами для підвищення якості життя людей і збереження екології планети. Обчислювальні потужності планети не здатні вирішити проблему моніторингу і управління через низьку продуктивність глобального комп'ютингу і високу енерговитратність існуючих класичних і квантових комп'ютерів. Можливим альтернативним рішенням проблеми є квантовий memory-driven logic-free комп'ютинг, що виключає коштовні, енерговитратні *on-not* операції суперпозиції і змішування. Будь-які процеси і явища класичного (квантового) комп'ютингу можна моделювати на квантовому (класичному) обчислювачі. Memory-driven і logic-free архітектура комп'ютингу є більш технологічною і менш енерговитратною в класичному і квантовому форматах реалізації.

Метрика, представлена на рис. 2.8, визначає взаємодію між квантовим і класичним комп'ютингом: 1) пам'ять, 2) структури даних, 3) операції, 4) алгоритми, 5) технології, 6) енерговитрати, 7) швидкодія, 8) температурні умови, 9) розв'язувані завдання, 10) невизначеність.

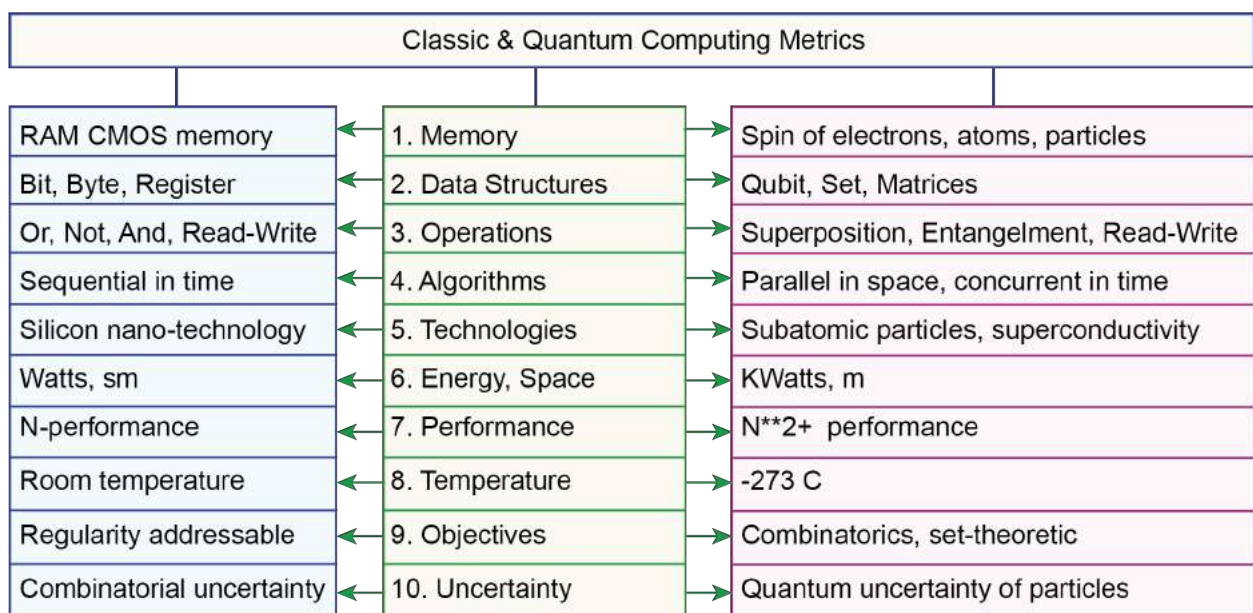


Рисунок 2.8 – Метрика порівняння класичного та квантового комп'ютингу

Квантова невизначеність кубіта, яка формується суперпозицією субатомних частинок (спинів або орбіт електронів), що одночасно формують значення нуля і одиниці, відповідає об'єднанню символів 0 і 1. Не існує перешкод для створення квантової невизначеності будь-якої кінцевої значності, у вигляді суперпозиції k елементарних частинок, що створює кінцеву множину кубітів, рівного $\log k$. В теорії множин багатозначній квантовій невизначеності відповідає універсум примітивів [10]. Кубіт або два примітиви $(0,1)$ породжують три реальних стани $\{0,1, X\}$. Чотири примітиви породжують 16 символів за формулою 2^{2^n} , серед яких – один пустий, чотири визначених, а решта 11 є квантово або теоретико-множинно невизначеними. Структурний порядок даних вимагає наявності адресних дешифраторів прямого доступу до даних. Адресація накладає заборону на паралельні операції, роблячи класичний комп'ютинг технологією послідовних обчислень. Усунення адресації є технологічний шлях до паралелізму. Прикладом виключення адресного простору в класичному комп'ютингу служить комбінаційна схема з'єднання логічних елементів, що створює паралелізм обробки даних. Комбінаційна схема являє собою апаратну реалізацію правильних двійкових відповідей на входні питання. Таку схему можна розглядати як швидкодіючу апаратну паралельну схему штучного інтелекту для прийняття рішень обчислювальним пристроєм. Пам'ять, як структура запам'ятовуючих елементів з адресними дешифраторами для виконання транзакцій (read-write), є головним компонентом сучасного послідовного комп'ютингу. Без комбінаційних схем можна створити комп'ютер, без пам'яті – неможливо. Пам'ять в комп'ютингі орієнтована на послідовне вирішення завдань, що знижує вартість апаратної реалізації алгоритму.

Паралельний квантовий комп'ютинг для неадресованих даних і класичний – для адресованих в майбутньому надасть людству глобальну комп'ютингову мережу з квантовими термінальними обчислювальними при-

строями. Цілком можливий і інший варіант, коли потужні квантові обчислювачі, зважаючи на їх високу вартість, будуть виконувати роль управління мережею класичних термінальних комп'ютерів. Квантовий комп'ютинг відрізняється від класичного паралельним вирішенням комбінаторних задач, що забезпечує високу швидкодію, за рахунок одночасного подання у квантовій точці субатомного масштабу кінцевої множини дискретних станів. Шляхи розвитку класичного і квантового комп'ютингу перетинаються в точці адитивного синтезу або вирощування субатомних, вільних від логіки, структур пам'яті, керованих фотонами, які матимуть добову циклічну сонячну енергозалежність, подібну рослинним організмам.

2.4 Архітектури квантового комп'ютингу

Структура класичної пам'яті комп'ютера призначена для послідовного зчитування-запису даних на основі використання дешифратора адрес і власне осередків пам'яті, що створює важко розв'язувані проблеми: 1) Неможливість здійснювати арифметико-логічні операції в пам'яті, що змушує розробників створювати спеціалізовані АЛП для обробки даних, пов'язані з використанням шини, яка негативно впливає на продуктивність. 2) Неможливість паралельно виконувати транзакції (read-write) даних шляхом одночасного звернення до всіх осередків адресної пам'яті. 3) Неможливість створювати адресовану пам'ять без апаратно-складних адресних дешифраторів, які роблять пам'ять дорогим виробом. 4) Неможливість створення комп'ютера без використання логічних операцій, що вимагають розробки спеціалізованих блоків АЛП поза основної пам'яті. Усунення згаданих недоліків призводить до створення нового типу memory-driven паралельних комп'ютерів [10] з практично необмеженою швидкістю обробки великих даних за рахунок: 1) Винятку блоку АЛП і шин передачі даних з пам'яті і назад. 2) Паралельної обробки даних шляхом одночасного звернення до всіх комірок пам'яті. 3) Винятку коштовних адресних дешифраторів, що обмежують адресний простір оброб-

люваних даних. 4) Реалізації всіх арифметичних і логічних операцій на пам'яті шляхом використання єдиної транзакції read-write.

Теоретично і практично всі згадані проблеми вирішуються шляхом створення квантового memory-driven комп'ютингу (рис. 2.9), що має наступні властивості: 1) відсутність блоку ALU для виконання обчислювальних операцій; 2) паралельна обробка теоретико-множинних даних; 3) відсутність адресних дешифраторів при зверненні до осередків пам'яті; 4) реалізація всіх операцій за допомогою транзакції read-write на пам'яті.

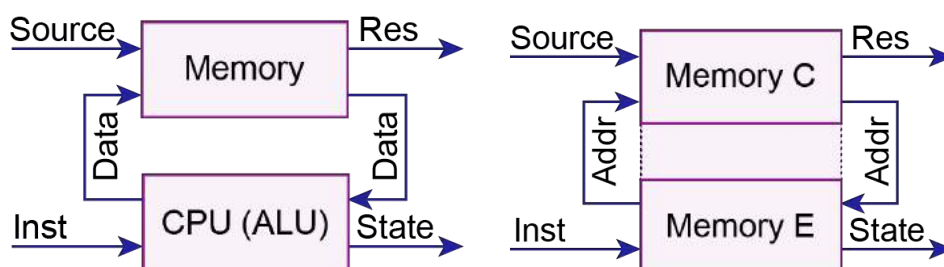


Рисунок 2.9 – Трансформація комп'ютингу від CPU- до memory-driven

Фізична основа класичного квантового обчислювача (рис. 2.10) полягає у використанні операцій суперпозиції і змішування над станами електронів (p), яких цілком достатньо для організації обчислювального процесу [10-12]. Електрон виконує функцію пам'яті для зберігання біта інформації. Транзакції між електронами здійснюються за допомогою квантів або фотонів (q) електромагнітного випромінювання. Низька і висока орбіти або спіни електронів відповідають значенням нуля і одиниці.

Базис логіки для квантового комп'ютингу представлений операціями суперпозиції та змішування, чому відповідають логічні функції або-ні (об'єднання-додаток). На основі пари або-ні будується технологічна система логічних вузлів і пристроїв для оптимізації обчислювальних процесів. Недоліки класичного квантового комп'ютингу: 1) Висока вартість забезпечення температурних умов для роботи субатомних структур на рівні, близь-

кому до нуля; 2) спостережуваність обчислювальних процесів, що призводять до руйнування даних після зчитування.

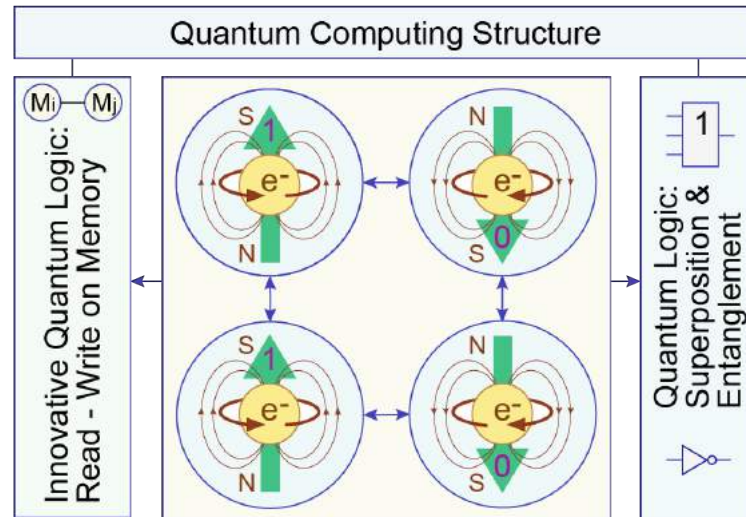


Рисунок 2.10 – Види квантового комп'ютингу

Усунення логіки, пов'язаної з суперпозицією і змішуванням, є інновацією в архітектурі квантового memory-driven комп'ютингу. Основа комп'ютера – адресована пам'ять, де реалізуються транзакції запису-зчитування даних на основі характеристичного рівняння обчислювального процесу [10,12]:

$$M_i = Q_i[M(X_i)].$$

Тут всі компоненти є пам'яттю для зберігання: вектора моделювання обчислювальної структури; вектор-кубітів і вектор-адрес логічних примітивів. Q-логіка реалізується на адресованій пам'яті, де також виконується асемблювання всіх примітивів за допомогою інтегруючого M-вектора обчислювального процесу, який формує виконавчі адреси $M(X)$ на основі використання вектора номерів вхідних змінних X .

Відмінності між класичним і квантовим комп'ютингом. Перший з них послідовно обробляє адресовані дані за n тактів або гомогенні дані паралельно за один автоматний такт. Обчислювальна складність обробки теоретично-множинних даних на класичному комп'ютері має квадратичну залежність.

Квантовий комп'ютинг вирішує завдання одночасної і паралельної обробки теоретико-множинних даних за один автоматний такт. Квантова суперпозиція (операція об'єднання) даних дає можливість зосередити в одній точці гільбертова простору кілька дискретних станів. Ілюстрацією сказаного є алгебра Кантора, яка оперує двома дискретними станами і створює 4 символи:

$$A^k = \{0,1, X = \{0,1\}, \emptyset\}.$$

Алфавіт являє собою теоретико-множинну інтерпретацію кубіта: суперпозиція двох станів одного кубіта утворює 4 символи. Два кубіта здатні згенерувати 16 станів, три кубіта – 64. Число станів Q має залежність від числа кубітів n , яка представлена формулою: $Q = 2^{2^n}$. Для паралельного виконання логічних операцій над кубітами виконується унітарне кодування примітивних символів алфавіту:

$a_i \in A^k$	0	1	X	\emptyset
$C(a_i)$	10	01	11	00

Паралелізм логічних операцій над множинами в класичному комп'ютері забезпечується збільшенням пам'яті для унарного кодування примітивних символів алфавіту.

Таблична модель логічного елемента представлена сукупністю кубів або дискретних відносин між вхідними та вихідними змінними. Натомість пропонується кубітний вектор упорядкованих за адресами вихідних станів, орієнтований на паралельне моделювання цифрових систем. Суперпозиція n рівнозначних елементів кінцевої множини в квантовому (Q) комп'ютингу взаємно-однозначно відповідає n -мірному вектору в класичному (C) адресованому комп'ютингу, рис. 2.11 [10], який отримано шляхом виконання операції над унарними кодами примітивних елементів множини.

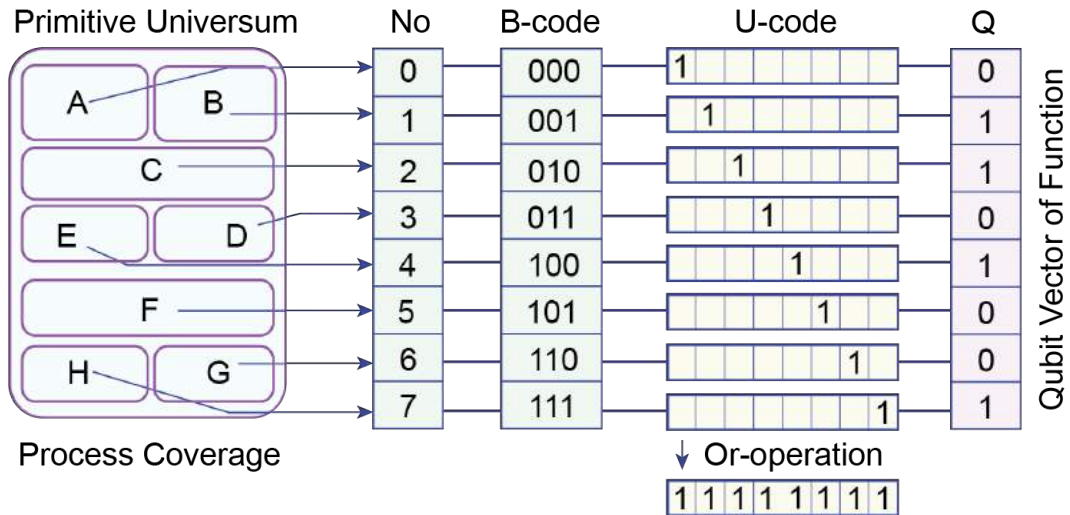


Рисунок 2.11 – Унітарне кодування символів універсуму

Будь-який перетин (and), об'єднання (or) або доповнення (not) в С-комп'ютері унарних кодів даних виконується паралельно, за один автоматний такт, як і в Q-комп'ютері. Плата за продуктивність обчислень – збільшення обсягу пам'яті, яке визначається формулою:

$$Q = n^2 / n \times \log n = n / \log n.$$

Таким чином, розширення потужності теоретико-множинного алфавіту можна поставити у відповідність нарощування кубітів в квантовому комп'ютері. Це дає можливість паралельно і в одному автоматному такті здійснювати обчислювальні процеси на основі використання логічних (теоретико-множинних) операцій. Суперпозиції n елементів кінцевої множини в квантовому (Q) комп'ютингу взаємно-однозначно відповідає n -мірний вектор в класичному (C) адресованому комп'ютингу, який виходить на основі використання логічної or-операції над унарними кодами символів вихідної множини, виконуваної паралельно за один автоматний такт.

2.5. Комп'ютинг в метриці: простір-час, матерія-енергія

Мета – використання космологічного комп'ютингу в практиці створення нових архітектур обчислювальних процесів, що враховують взаємодію

між простором і часом, матерією і енергією, яка визначається модельними аддитивними виразами: $s + t = 1$, $m + e = 1$ в будь-якій космологічній фазі [10].

Завдання: 1) Комп'ютерингові архітектури просторового паралелізму і часової послідовності обчислень. 2) Комп'ютерингові архітектури матеріально-енергетичної взаємодії обчислювальних процесів.

Використовувана доктрина – комп'ютеринг є транзакція (запис-зчитування) на компонентах, здатних зберігати дані. Компоненти для зберігання даних можуть бути представлені будь-якою формою існування матерії (рідка, тверда, газоподібна, плазмова). Транзакція є еквівалентом виконання роботи за допомогою застосування сили, яка змінює стан космологічних компонентів (m , e , s , t). Таким чином, комп'ютерингова транзакція є передача матеріально-енергетичної субстанції у просторі і часі. Носіями субстанції є примітивні квантові частинки: фотони, електрони, які представляють собою похідні від сил гравітації, електромагнетизму. Виникає пара: матерія-енергія для комп'ютерингової взаємодії: $m + e = 1$.

Простір пов'язаний з часом, вони не можуть існувати один без одного. При максимальному космологічному просторі модельний час дорівнює нулю, при мінімальному – час дорівнює одиниці. Для комп'ютерингу це означає, будь-який обчислювальний процес може бути реалізований паралельно в комбінаційній схемі за один автоматний такт, що можна вважати – процесорний час вирішення задачі дорівнює нулю. Альтернативний найтриваліший обчислювальний процес пов'язаний з максимальним часом рішення задачі при відсутності простору комбінаційної логіки для розпаралелювання процесів, коли використовується пам'ять і послідовне звернення до неї для запису-зчитування даних. Комп'ютеринг визначається просторово-часовою взаємодією матерії та енергії: чим більше комбінаційного простору, тим менше час вирішення завдання, вірно і зворотне: $s + t = 1$.

Космологічний комп'ютинг визначається гармонійним геномом розвитку Всесвіту з періодом в 60 мільярдів років [10], на якому в будь-якій точці фази виконуються дві рівності: $m + e = 1$ і $s + t = 1$. Що стосується комп'ютингу, то тут дві рівності також зберігаються у фазовому портреті, але відмінність полягає в неповторюваності фаз його розвитку, яка визначається двома ідентичними графіками, що формують взаємодію простору і часу, матерії та енергії в адитивно-модельній константі розвитку Земного Комп'ютингу, яка становить $\frac{1}{4}$ частини графіка-періоду розвитку Всесвіту, рис. 2.12.

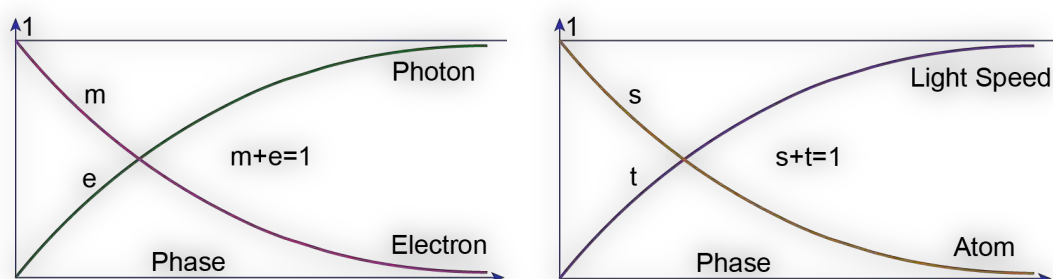


Рисунок 2.12. Космологія комп'ютингу

Космологія має єдину константу – швидкість світла, до якої прив'язані гармонійні зміни всіх чотирьох компонентів Всесвіту: простір-час, матерія-енергія, які між собою взаємодіють відповідно до рівняння Ейнштейна: $e=mc^{**2}$, $v=tc^{**2}$.

Перший графік – закон сталості матеріально-енергетичного комп'ютингу – зменшення площі кристала або кількості матерії, від кілограмів до електрона, для стійкого зберігання біта інформації суворо пов'язано зі зростанням ролі корисної енергії (ККД), необхідної для виконання транзакції. У межі така енергія дорівнює одному фотону на один електрон або біт, що приймається за одиницю.

Другий графік – закон сталості просторово-часового комп'ютингу – підвищення продуктивності обчислювального процесу, пов'язане з досягненням швидкості світла, при якому модельний час має максимальне значення,

строго взаємодіє зі зменшенням обчислювального простору, який перетворюється в орбіту або спіні електрона.

Квантовий комп'ютинг є не альтернативою, а земною межею класичного комп'ютингу, заснованого на використанні електромагнітних сил. Інші види комп'ютингу пов'язані із застосуванням гравітаційних і / або ядерних сил взаємодії матерії.

Квантовий memory-driven комп'ютинг, який використовує структуру електронів для зберігання даних, має швидкодію транзакцій, рівне швидкості світла. Це означає його незаперечні переваги, пов'язані з необмеженим паралелізмом обчислень на структурі електронів і максимальною продуктивністю фотонних транзакцій між електронами, що дорівнює швидкості світла при порівнянні з класичним квантовим комп'ютером, що використовують квантову логіку ALU-процесора і обмін даними з пам'яттю.

Збереженість простору комп'ютингу. Класичний комп'ютинг вимагає створення структур даних під існуючу реалізацію логічних операцій в кристалах кремнію: «дані для логіки». Альтернативний шлях може бути представлений реалізацією логічних операцій, які будуть впроваджуватися в хаос великих даних: «логіка для даних».

Представлення даних у вигляді компактних адрес-ідентифікаторів створює потужну технологію паралельних обчислювальних процесів, орієнтованих на високопродуктивну аналітику великих даних.

Разом з тим, адреса даних є головною перешкодою на шляху до створення паралельних обчислювальних процесів, оскільки він формує послідовність даних для їх завжди непаралельної обробки.

Позбавлення від адрес в структурах пам'яті призводить до комп'ютингу високої продуктивності за рахунок паралельної обробки даних.

Дивно, але факт, хаотичне невпорядковане теоретико-множинне уявлення даних в найближчому майбутньому можливо буде представляти осно-

ву для створення сучасного високопродуктивного паралельного комп'ютингу.

Існує два примітивних способи структурування даних: адресний і іменований. Перший завжди організовує дані в упорядковану послідовність, де вони ранжуються, створюючи ієрархію або строго певну чергу для їх обробки-обслуговування обчислювальним механізмом. Другий називає дані ідентифікаторами, які інваріантні до порядку, ранжирування, ієрархії або черги. Інакше дані, відмічені тільки іменами, є першими і рівними один перед одним для їх обслуговування обчислювачем.

Існуючі технології паралельної обробки безадресних даних включають: комбінаційні логічні схеми; регістрові логічні операції; SIMD, MIMD процесори; квантові обчислення на основі операцій суперпозиції і змішування; пам'ять без адресних дешифраторів.

Пропонується *address-free chaos-computing*, де одним з можливих варіантів апаратної реалізації є *quantum computing*. The Chaos Computer Club (CCC) is Europe's largest association of hackers. Дивно, але факт, що хаос-комп'ютингу займаються хакери, які прагнуть підвищити продуктивність своїх додатків за рахунок високого паралелізму обчислень, пов'язаного з апаратної надмірністю. Рівень або критерій паралелізму обчислень слід визначати, виходячи зі співвідношення паралельності структур даних до їх послідовності в часі: відношення висоти до ширини прямокутника обчислень. При відсутності помилок проектування деякого обчислювача твір сторін прямокутника (паралельність – послідовність) є величина постійна, що ілюструється геометричними фігурами, представленими на рис. 2.13.

$$\begin{array}{c}
 1 \times 9 = 9 \\
 \boxed{1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8 \ 9} =
 \end{array}
 =
 \begin{array}{c}
 3 \times 3 = 9 \\
 \boxed{\begin{array}{ccc} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{array}} =
 \end{array}
 =
 \begin{array}{c}
 9 \times 1 = 9 \\
 \boxed{\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \end{array}}
 \end{array}$$

Рисунок 2.13 – Аксиома збереження простору і часу в обчисленнях

Розвиток комп'ютингу також можна описати як мультиплікативну гру між простором (H) і часом (T), маючи на увазі, що простір являє собою апаратні витрати, а час – часові витрати, необхідні для реалізації алгоритму або обчислювача: $Q = H \times T = \text{const}$.

Отже, основою паралелізму є відсутність адрес або ранжирування даних, що призводить до їх однаковості для обчислювача, який повинен мати додаткові можливості для вже одночасної обробки даних. Метрика паралелізму – одночасне виконання логічних операцій (and, or, not, xor) над усіма даними за технологією SIMD. Таке можливо при унітарному кодуванні будь-яких даних: теоретико-множинних, ранжированих, упорядкованих, численних, адресних, символічних, графічних, табличних, аналітичних. Це дає підстави стверджувати, що послідовні обчислення завжди можна перетворити в паралельний комп'ютинг на класичному адресному комп'ютері шляхом трансформування структур даних з адресних в унітарно кодовані.

Далі розглядаються приклади розв'язання задач комп'ютингу в паралельному, послідовному і гібридному виконанні.

Перш за все, інтерес представляє оптимальне рішення проблеми покриття шляхом використання цифрової структури (Quantum Coverage Sequencer – QCS), що створює всі можливі поєднання вхідних векторів у формі кубітних структур даних, представлених булеаном або сукупністю всіх підмножин, рис. 2.14.

Схема має логічний аналізатор при кожному регістровому виході, який визначає повноту покриття шляхом формування одиничних значень в усіх розрядах регістрової змінної. Число функцій-перетворювачів "вектор-біт" відповідає кількості елементів і дорівнює $Q = 2^n - 1$. Апаратна складність цифрової схеми для пошуку оптимального покриття, де n – кількість рядків, m – довжина регістра, визначається: $Q = [2^{n+1} + 2^n] \times m + 2^n$. Обчислювальна складність процедур при пошуку оптимального покриття дорівнює n автоматним тактам. Запропоновані структури QC-секвенсора для паралельного, гібридного і послідовного вирішення комбінаторної задачі покриття підтверджують аксіому збереження прямокутника обчислювального процесу, де сторони змінюються в кінцевих межах, відмінних від нуля, формуючи просторово-часовий континуум комп'ютингу.

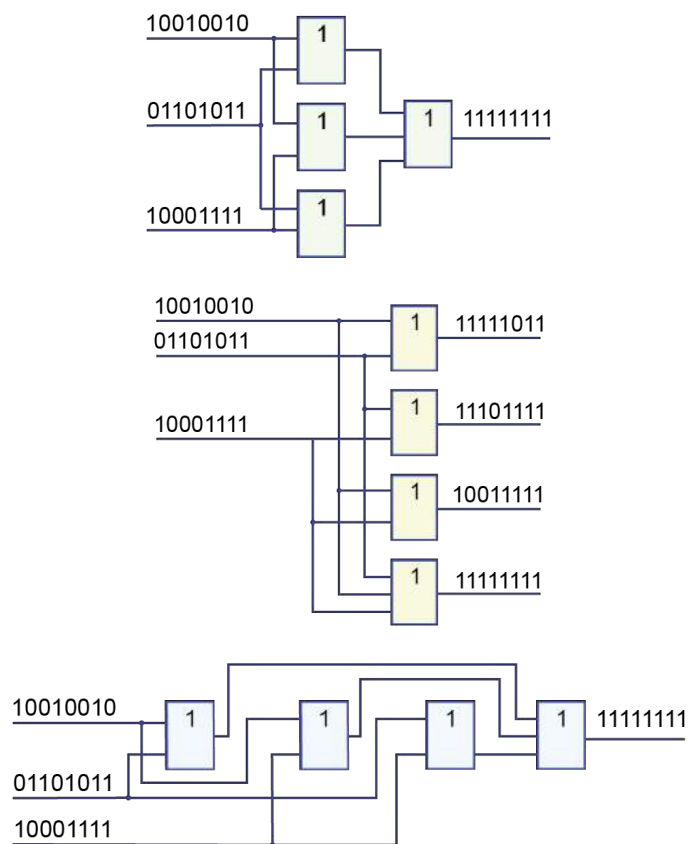


Рисунок 2.14 – QC-секвенсори для визначення оптимального покриття

2.6. Висновки до розділу 2

Дослідження архітектур для квантового паралельного комп'ютинга дозволило вирішити такі завдання:

1) Розроблено квантові архітектури memory-driven паралельного комп'ютингу для аналізу кубітних моделей цифрових систем і компонентів, які відрізняються невикористанням традиційної логіки у процесингу алгоритмів design and test.

2) Розроблено метрики і моделі для класифікації квантового комп'ютингу, які показують фізичні, математичні та архітектурні особливості обчислювальних процесів.

3) Створена інфраструктура quantum design and test of memory-driven computing, яка дозволяє дослідникам синтезувати нові моделі і методи на кубітних структурах даних.

4) Розроблено архітектури спільного використання квантового та класичного комп'ютингу, які враховують можливості паралелізму квантового комп'ютингу з перевагами класичного при вирішенні послідовних проблем.

5) Запропоновано модель просторово-часової та матеріально енергетичної взаємодії, яка служить методологією для теорії і практики всіх видів комп'ютингу, що пояснює технологічні та енергетичні межі обчислювальних процесів.

6) Запропоновано просторово-часові квантові архітектури для реалізації паралельних, послідовних і гібридних обчислювальних процедур і алгоритмів при вирішенні комбінаторно складних завдань покриття.

Результати розділу опубліковано у роботах [9, 10, 14, 18].

РОЗДІЛ 3

ДЕДУКТИВНИЙ АНАЛІЗ ЛОГІЧНИХ ФУНКЦІЙ

Пропонується структурна модель взаємодії X-функцій і похідних компонентів, орієнтованих на синтез і аналіз цифрових систем з метою зменшення часу проектування і тестування обчислювальних пристроїв. Вводиться поняття простих X-функцій від кінцевого числа змінних, які характеризуються відсутністю мінімізації та наявністю властивостей тестопригодності для вирішення завдань синтезу тестів, моделювання і діагностування. Формулюються метричні властивості X-функцій від кінцевого числа змінних для оцінки якості перевіряючих тестів шляхом дедуктивного моделювання перевірюваних константних несправностей на кубітних структурах даних. Пропонується аналітичний вираз для синтезу кубітних покриттів X-функцій від кінцевого числа змінних з метою подальшого синтезу і аналізу тестів перевірки та діагностування несправностей. Синтезуються дедуктивні формули транспортування вхідних списків несправностей на зовнішні виходи для X-функцій від кінцевого числа змінних для побудови секвенсора моделювання дефектів, інваріантного до вхідних тестових наборів. Пропонується метод синтезу дедуктивних кубітних покриттів для моделювання несправностей на основі використання Q-покриттів функціональностей.

3.1 State of the art

Технології дедуктивного аналізу застосовні до всіх систем, які мають дискретний опис поведінки структур і компонентів. Їх основна перевага – паралелізм виконання логічних операцій в обчислювальних процесах, спрямованих на визначення впливу несправних компонентів на поведінку цифрової системи. Природно, що апарат дедуктивного моделювання є найефективнішим засобом для синтезу тестів, оцінки їх якості та online діагностуван-

ня несправностей. Сутність дедуктивного аналізу полягає в використанні хог-операції для визначення відмінностей в поведінці цифрового пристрою при наявності і відсутності дефектів, модельно внесених у фрагменти логічної схеми. Технологічно ця операція використовується для синтезу дедуктивних підсхем-доповнень, до оригінальної функціональності, які здатні транспортувати в паралельному режимі вхідні списки несправностей від зовнішніх входів схеми до її зовнішніх виходів. Складність дедуктивних підсхем, реалізованих програмно або апаратно, на порядок вище, ніж корисна функціональність. Тому ефективність застосування дедуктивного аналізу слід розглядати з позиції підвищення швидкодії за рахунок апаратної надмірності, що дає зменшення інтегрального часу проектування цифрових систем на кристалах і виходу виробу на ринок електронних технологій.

Мета дослідження – зменшення часу проектування цифрової системи на кристалі за рахунок паралельного: 1) синтезу дедуктивних функцій логічних компонентів і схем, 2) паралельного аналізу списків несправностей при оцінці якості тестів, 3) використання кубітних форм опису поведінки цифрових схем, включаючи X-функції.

Завдання: 1) Розробити структурну модель дедуктивного аналізу X-функцій для синтезу тестів. 2) Визначити прості X-функції від кінцевого числа змінних, для синтезу тестів, моделювання і діагностування цифрових пристроїв. 3) Сформулювати метричні властивості X-функцій від кінцевого числа змінних для оцінки якості перевіряючих тестів на кубітних структурах даних. 4) Визначити аналітичну модель взяття похідних по кубітним покриттям X-функцій від кінцевого числа змінних. 5) Синтезувати дедуктивні формули транспортування списків несправностей для X-функцій від кінцевого числа змінних. 6) Розробити кубітний метод синтезу тестів на основі Q-покриттів X-функцій.

Запит по поєднанню слів "Fault Simulation" дає 35355 посилань на наукові публікації в IEEE Xplore Digital Library [40]. Такий інтерес ринку визна-

чається перманентними проблемами, пов'язаними з дефектами в усіх комп'ютерних виробках різного призначення: соціальні мережі, обчислення, транспорт, медицина, будівництво, космонавтика, озброєння, енергетика. Що стосується моделювання дефектів, тут загальна картина технічної діагностики може бути представлена взаємодією структур даних, моделей, методів і різних дефектів, зображених на рис. 3.1. Різноманіття теорій в області моделювання несправностей досліджується вченими протягом останніх 70 років, що відображено в десятках тисяч публікацій, включаючи [41-53]. Статті останніх років орієнтовані на створення нових моделей і методів моделювання, які є наслідком появи нових субатомних фізичних і апаратних рішень і технологій, забезпечених штучним інтелектом, хмарними сервісами і квантовими структурами даних [54-73].

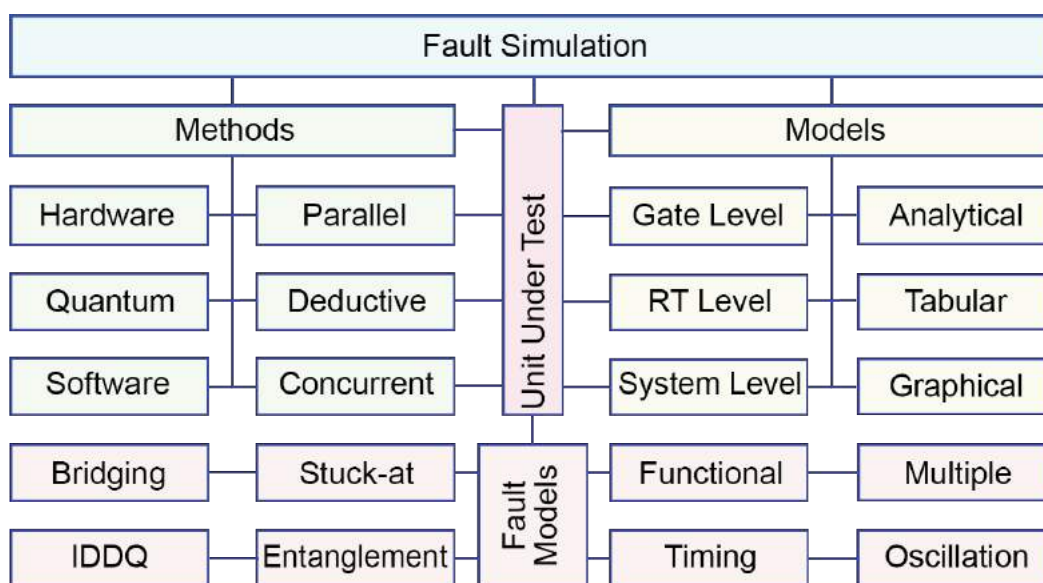


Рисунок 3.1 – Моделі і методи аналізу несправностей

Як найбільш паралельна технологія, близька до квантового комп'ютингу, дедуктивний аналіз використовується для визначення якості тесту щодо класу одиночних константних несправностей. Теорія дедуктивного аналізу [41] орієнтована на паралельну обробку списків несправностей спільно зі справним моделюванням вхідної двійкової послідовності. Базові поняття

теорії моделювання несправностей представлені моделями і методами транспортування списків дефектів через функціональні логічні елементи [45-53, 70]. Дедуктивний аналіз не має обмежень на рівень складності цифрових схем або компонентів. Метод використовується для вентильних, реєстрових, транзакційних та системних моделей цифрових виробів і їх компонентів. Сутність дедуктивного аналізу полягає в паралельному моделюванні хог-відношень між справним і несправним обчислювальним пристроєм. Дедуктивний аналіз заснований на використанні булевих похідних, коли зміни двійкових станів вхідних змінних призводять до зміни виходів логічної функції. Для транспортування активізації вхідних змінних на вихід схеми використовується хог-операція, яка дає можливість визначити наявність або відсутність активного логічного шляху на даному вхідному двійковому наборі. Якщо такий шлях існує, то всі поодинокі константні несправності, інверсні по відношенню до справної поведінки ліній на шляху активізації, є перевіряюваними. Для моделювання несправностей необхідно синтезувати дедуктивну схему від n змінних на кожній вхідній послідовності. Процедура синтезу має квадратичну обчислювальну складність від числа змінних, що можна віднести до недоліків методу моделювання. Для функцій, що використовують хог, not-хог операції, процедура синтезу дедуктивних формул практично відсутня, оскільки їх функціональність тотожно дорівнює дедуктивного логічного виразу, яке інваріантно до всіх вхідних послідовностей.

3.2 Синтез дедуктивних формул для примітивів

Базові, майже тотожності, дедуктивного аналізу цифрових систем на кристалах [10] дають технологічні можливості реалізувати моделювання як завгодно складних компонентів обчислювальних пристроїв. Дедуктивних функцій паралельного моделювання векторів несправностей на вичерпному тесті для функціональних елементів and, or, not досить для аналізу як завгодно складних схем. Інтегральний дедуктивний процесор для функції and має вигляд:

$$\begin{aligned}
& L[T = (00,01,10,11), F = (X_1 \wedge X_2)] = \\
& = L\{(\bar{x}_1\bar{x}_2 \vee \bar{x}_1x_2 \vee x_1\bar{x}_2 \vee x_1x_2) \wedge [(X_1 \oplus T_{t1} \wedge X_2 \oplus T_{t2}) \oplus T_{t3}]\} = \\
& = (\bar{x}_1\bar{x}_2)\{[(X_1 \oplus 0) \wedge (X_2 \oplus 0)] \oplus 0\} \vee (\bar{x}_1x_2)\{[(X_1 \oplus 0) \wedge (X_2 \oplus 1)] \oplus 0\} \vee \\
& \vee (x_1\bar{x}_2)\{[(X_1 \oplus 1) \wedge (X_2 \oplus 0)] \oplus 0\} \vee (x_1x_2)\{[(X_1 \oplus 1) \wedge (X_2 \oplus 1)] \oplus 1\} = \\
& = (\bar{x}_1\bar{x}_2)(X_1 \wedge X_2) \vee (\bar{x}_1x_2)(X_1 \wedge \bar{X}_2) \vee (x_1\bar{x}_2)(\bar{X}_1 \wedge X_2) \vee (x_1x_2)(X_1 \vee X_2).
\end{aligned}$$

Аналогічно визначається схема дедуктивного процесингу для логічної функції or:

$$\begin{aligned}
& L[T = (00,01,10,11), F = (X_1 \vee X_2)] = \\
& = L\{(\bar{x}_1\bar{x}_2 \vee \bar{x}_1x_2 \vee x_1\bar{x}_2 \vee x_1x_2) \wedge [(X_1 \oplus T_{t1} \vee X_2 \oplus T_{t2}) \oplus T_{t3}]\} = \\
& = (\bar{x}_1\bar{x}_2)\{[(X_1 \oplus 0) \vee (X_2 \oplus 0)] \oplus 0\} \vee (\bar{x}_1x_2)\{[(X_1 \oplus 0) \vee (X_2 \oplus 1)] \oplus 1\} \vee \\
& \vee (x_1\bar{x}_2)\{[(X_1 \oplus 1) \vee (X_2 \oplus 0)] \oplus 1\} \vee (x_1x_2)\{[(X_1 \oplus 1) \vee (X_2 \oplus 1)] \oplus 1\} = \\
& = (\bar{x}_1\bar{x}_2)(X_1 \vee X_2) \vee (\bar{x}_1x_2)(\bar{X}_1 \wedge X_2) \vee (x_1\bar{x}_2)(X_1 \wedge \bar{X}_2) \vee (x_1x_2)(X_1 \wedge X_2).
\end{aligned}$$

Тут $T_t = (T_{t1}, T_{t2}, T_{t3})$, $(t = \overline{1,4})$ – тест-вектор, який має 3 координати, де остання з них визначає стан виходу двухвходового елемента and (or), L – вихідний список несправностей, X – список несправностей на конкретному вході примітиву, $x = \{0,1\}$ – логічне значення на вході примітиву. Далі показана дедуктивна формула транспортування списків несправностей через інвертор. Тут $T_t = (T_{t1}, T_{t2})$, $(t = \overline{1,2})$ – тест-вектор, який має 2 координати, де друга – стан виходу примітиву:

$$\begin{aligned}
& L[T = (0,1), F = \bar{X}_1] = L\{(\bar{x}_1 \vee x_1)[(\bar{X}_1 \oplus T_{t1}) \oplus T_{t2}]\} = \\
& = \bar{x}_1[(\bar{X}_1 \oplus 0) \oplus 1] \vee x_1[(\bar{X}_1 \oplus 1) \oplus 0] = \bar{x}_1\bar{\bar{X}}_1 \vee x_1\bar{\bar{X}}_1 = \bar{x}_1X_1 \vee x_1X_1.
\end{aligned}$$

Результат ілюструє інваріантність інверсії до вхідного набору для транспортування дефектів. Вона трансформується у повторювач. Тому функції інверсії і повторення не фігурують в дедуктивних аналітичних виразах. Джерелом несправностей в memo-driven комп'ютингі є структура CLB, представлена на рис. 3.2, а. Для їх перевірки є апаратна реалізація дедуктивних функцій двухвходових елементів and, or на вичерпному тесті, що пред-

ставлено універсальною схемою (рис. 3.2, б) дедуктивно-паралельного аналізу несправностей [10].

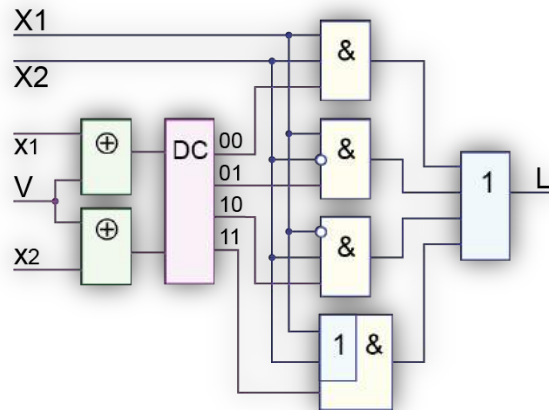
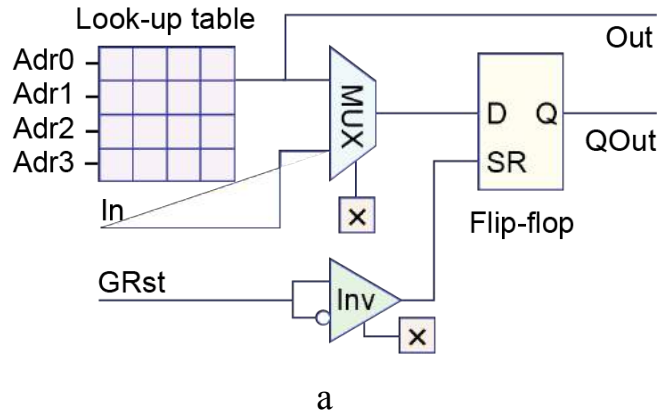


Рисунок 3.2 – Джерело і симулятор несправностей примітивів

У симуляторі представлені вхідні булеві (x_1 , x_2) і регістрові (X_1 , X_2) змінні для зберігання списків несправностей в формі векторів, змінна V вибору типу справної функції (and, or), вихідна регістрова змінна Y , в якій накопичуються транспортовані дефекти. Стани двійових входів x_1 , x_2 і змінна вибору елемента визначають одну з чотирьох дедуктивних функцій для отримання вектора Y перевірюваних несправностей цифрової схеми.

3.3 Синтез дедуктивних формул для хог (not-хог) функції

Логічна функція хог має унікальну властивість транспортування вхідних списків несправностей. Далі розглядається синтез дедуктивної моделі даної функції на всіх вхідних тестових послідовностях:

$$\begin{aligned} L[T = (00,01,10,11), F = X_1 \oplus X_2 = X_1\bar{X}_2 \vee \bar{X}_1X_2] &= \\ &= L\{(\bar{x}_1\bar{x}_2 \vee \bar{x}_1x_2 \vee x_1\bar{x}_2 \vee x_1x_2) \wedge \\ &\wedge [(X_1 \oplus T_{t1})(\bar{X}_2 \oplus T_{t2}) \vee (\bar{X}_1 \oplus T_{t1})(X_2 \oplus T_{t2})] \oplus T_{t3}\}: \end{aligned}$$

$$\begin{aligned} L(T_0 = 00) &= (\bar{x}_1\bar{x}_2) \wedge [(X_1 \oplus 0)(\bar{X}_2 \oplus 0) \vee (\bar{X}_1 \oplus 0)(X_2 \oplus 0)] \oplus 0\} = \\ &= (\bar{x}_1\bar{x}_2) \wedge [(X_1\bar{X}_2 \vee \bar{X}_1X_2)]; \end{aligned}$$

$$\begin{aligned} L(T_1 = 01) &= (\bar{x}_1x_2) \wedge [(X_1 \oplus 0)(\bar{X}_2 \oplus 1) \vee (\bar{X}_1 \oplus 0)(X_2 \oplus 1)] \oplus 1\} = \\ &= (\bar{x}_1x_2) \wedge [(X_1X_2) \vee (\bar{X}_1\bar{X}_2)] \oplus 1 = \\ &= (\bar{x}_1x_2) \wedge [(\bar{X}_1 \vee \bar{X}_2) \wedge (X_1 \vee X_2)] = (\bar{x}_1x_2) \wedge [(X_1\bar{X}_2 \vee \bar{X}_1X_2)]; \end{aligned}$$

$$\begin{aligned} L(T_2 = 10) &= (x_1\bar{x}_2) \wedge [(X_1 \oplus 1)(\bar{X}_2 \oplus 0) \vee (\bar{X}_1 \oplus 1)(X_2 \oplus 0)] \oplus 1\} = \\ &= (x_1\bar{x}_2) \wedge [(\bar{X}_1\bar{X}_2) \vee (X_1X_2)] \oplus 1 = \\ &= (x_1\bar{x}_2) \wedge [(X_1 \vee X_2) \wedge (\bar{X}_1 \vee \bar{X}_2)] = (x_1\bar{x}_2) \wedge [(X_1\bar{X}_2 \vee \bar{X}_1X_2)]; \end{aligned}$$

$$\begin{aligned} L(T_3 = 11) &= (x_1x_2) \wedge [(X_1 \oplus 1)(\bar{X}_2 \oplus 1) \vee (\bar{X}_1 \oplus 1)(X_2 \oplus 1)] \oplus 0\} = \\ &= (x_1x_2) \wedge [(\bar{X}_1X_2) \vee (X_1\bar{X}_2)] \oplus 0 = (x_1x_2) \wedge [(X_1\bar{X}_2 \vee \bar{X}_1X_2)]. \end{aligned}$$

В результаті синтезу виявилось, що всі дедуктивні формули на чотирьох вхідних тестових наборах мають однаковий вигляд, що означає інваріантність хог-функції до вхідних тестових наборів – будь-яке вхідне слово має одну єдину дедуктивну формулу:

$$\begin{aligned} L(\text{хог}) &= (\bar{x}_1\bar{x}_2 \vee \bar{x}_1x_2 \vee x_1\bar{x}_2 \vee x_1x_2) \wedge [(X_1\bar{X}_2 \vee \bar{X}_1X_2) = \\ &= (xx) \wedge [(X_1\bar{X}_2 \vee \bar{X}_1X_2)]. \end{aligned}$$

Це означає, що обчислювальна складність дедуктивних виразів для Х-функції від n змінних в n разів менше будь-якої іншої логічної оригінальної функції, а списки вхідних несправностей завжди будуть об'єднуватися на виході хог-елемента по функції «виключне або» за вирахуванням ситуації, коли дані списки будуть ідентичними. В даному випадку на виході логічного елемента хог не перевірятиметься жодна несправність.

Існують і інші функції, що мають єдину універсальну дедуктивну формулу для всіх вхідних тестових впливів. Нижче пропонується синтез дедуктивної формули для операції еквівалентності, яка є інверсною по відношенню до кубітного покриття хог-функції:

$$\begin{aligned} L[T = (00,01,10,11), F = X_1 \equiv X_2 = X_1 X_2 \vee \bar{X}_1 \bar{X}_2] &= \\ &= L\{(\bar{x}_1 \bar{x}_2 \vee \bar{x}_1 x_2 \vee x_1 \bar{x}_2 \vee x_1 x_2) \wedge \\ &\wedge [(X_1 \oplus T_{t1})(X_2 \oplus T_{t2}) \vee (\bar{X}_1 \oplus T_{t1})(\bar{X}_2 \oplus T_{t2})] \oplus T_{t3}\}: \end{aligned}$$

$$\begin{aligned} L(T_0 = 00) (\bar{x}_1 \bar{x}_2) \wedge [(X_1 \oplus 0)(X_2 \oplus 0) \vee (\bar{X}_1 \oplus 0)(\bar{X}_2 \oplus 0)] \oplus 1 &= \\ = (\bar{x}_1 \bar{x}_2) \wedge [(X_1 X_2) \vee (\bar{X}_1 \bar{X}_2)] \oplus 1 &= (\bar{x}_1 \bar{x}_2) \wedge [(X_1 \bar{X}_2 \vee \bar{X}_1 X_2)]; \end{aligned}$$

$$\begin{aligned} L(T_1 = 01) (\bar{x}_1 x_2) \wedge [(X_1 \oplus 0)(X_2 \oplus 1) \vee (\bar{X}_1 \oplus 0)(\bar{X}_2 \oplus 1)] \oplus 0 &= \\ = (\bar{x}_1 x_2) \wedge [(X_1 \bar{X}_2) \vee (\bar{X}_1 X_2)] \oplus 0 &= (\bar{x}_1 x_2) \wedge [(X_1 \bar{X}_2 \vee \bar{X}_1 X_2)]; \end{aligned}$$

$$\begin{aligned} L(T_2 = 10) (x_1 \bar{x}_2) \wedge [(X_1 \oplus 1)(X_2 \oplus 0) \vee (\bar{X}_1 \oplus 1)(\bar{X}_2 \oplus 0)] \oplus 0 &= \\ = (x_1 \bar{x}_2) \wedge [(\bar{X}_1 X_2) \vee (X_1 \bar{X}_2)] \oplus 0 &= (x_1 \bar{x}_2) \wedge [(X_1 \bar{X}_2 \vee \bar{X}_1 X_2)]; \end{aligned}$$

$$\begin{aligned} L(T_3 = 11) (x_1 x_2) \wedge [(X_1 \oplus 1)(X_2 \oplus 1) \vee (\bar{X}_1 \oplus 1)(\bar{X}_2 \oplus 1)] \oplus 1 &= \\ = (x_1 x_2) \wedge [(\bar{X}_1 \bar{X}_2) \vee (X_1 X_2)] \oplus 1 &= (x_1 x_2) \wedge [(X_1 \bar{X}_2 \vee \bar{X}_1 X_2)]. \end{aligned}$$

Таким чином, функція еквівалентності (not-хор) також має унікальні здібності із транспортування вхідних списків несправностей на вихід логічної функції, що не залежать від вхідного впливу:

$$\begin{aligned} L(\text{not} - \text{hor}) &= (\bar{x}_1 \bar{x}_2 \vee \bar{x}_1 x_2 \vee x_1 \bar{x}_2 \vee x_1 x_2) \wedge [(X_1 \bar{X}_2 \vee \bar{X}_1 X_2) = \\ &= (xx) \wedge [(X_1 \bar{X}_2 \vee \bar{X}_1 X_2)]. \end{aligned}$$

В результаті вийшло, що дедуктивні формули для двох різних логічних функцій хор і not-хор ідентичні один одному. Важливим також є факт, що будь-який тестовий вплив для цифрового пристрою, що складається зі згаданих логічних елементів, перевіряє 50 відсотків одиночних константних несправностей, інверсних по відношенню до справного стану вхідних ліній. Щоб перевірити всі поодинокі константні несправності вхідних ліній, необхідно подати на схему тільки два тестових взаємно інверсних набори.

Для викладу подальшого матеріалу інтерес представляє порівняння трьох форм опису логічних функцій: таблиця істинності (ТТ – Truth Table), диз'юнктивна нормальна форма (DF – Disjunctive Form), кубітний вектор або Q-покриття (QC – Qubit Coverage):

$$Y(\text{TT}) = \begin{array}{|c|c|c|c|} \hline X_1 & X_2 & X_3 & Y \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ \hline \end{array};$$

$$Y(\text{DF}) = \bar{X}_1\bar{X}_2X_3 \vee \bar{X}_1X_2\bar{X}_3 \vee X_1\bar{X}_2\bar{X}_3 \vee X_1X_2X_3;$$

$$Y(\text{QC}) = 01101001.$$

Кубітне покриття є вектор станів виходу логічної функції, упорядкованих за зростанням двійкових адрес або вхідних наборів таблиці істинності. Q-покриття є явною і найкомпактнішою формою опису поведінки логічної функції, яка має наступні переваги перед табличною і аналітичною формами:

- 1) Q-вектор в n раз вимагає меншого обсягу пам'яті для зберігання даних у порівнянні з таблицею істинності.
- 2) Q-вектор не вимагає $(n**2)$ -складні обчислювальних процедур, необхідних для визначення стану виходу логічної функції по диз'юнктивній нормальній формі.
- 3) Характеристичне рівняння для аналізу кубітного покриття містить тільки адресні операції запису-зчитування: $M_i = Q_i [M(X_i)]$, що мають високий паралелізм і лінійну обчислювальну складність.

З урахуванням викладеної інформації про переваги кубітного покриття далі пропонується синтез дедуктивної формули для більш складної логічної схеми, якою є хор-функція від трьох змінних, що має кубітне покриття: (01101001). Для аналітичного зображення згаданої функції

$$f(Q = 01101001) = \bar{X}_1\bar{X}_2X_3 \vee \bar{X}_1X_2\bar{X}_3 \vee X_1\bar{X}_2\bar{X}_3 \vee X_1X_2X_3$$

нижче виконується досить складна побудова дедуктивної формули транспортування списків вхідних несправностей на зовнішній вихід:

$$\begin{aligned}
 L(T_0 = 000) &= (\bar{X}_1\bar{X}_2X_3 \vee \bar{X}_1X_2\bar{X}_3 \vee X_1\bar{X}_2\bar{X}_3 \vee X_1X_2X_3); \\
 L(T_1 = 001) &= (\bar{X}_1\bar{X}_2\bar{X}_3 \vee \bar{X}_1X_2X_3 \vee X_1\bar{X}_2X_3 \vee X_1X_2\bar{X}_3) \oplus 1 = \\
 &= (\overline{\bar{X}_1\bar{X}_2\bar{X}_3 \vee \bar{X}_1X_2X_3 \vee X_1\bar{X}_2X_3 \vee X_1X_2\bar{X}_3}) = \\
 &= \bar{X}_1\bar{X}_2\bar{X}_3 \wedge \bar{X}_1X_2X_3 \wedge X_1\bar{X}_2X_3 \wedge X_1X_2\bar{X}_3 = \\
 &= (X_1 \vee X_2 \vee X_3) \wedge (X_1 \vee \bar{X}_2 \vee \bar{X}_3) \wedge (\bar{X}_1 \vee X_2 \vee \bar{X}_3) \wedge (\bar{X}_1 \vee \bar{X}_2 \vee X_3) = \\
 &= (X_1 \vee X_2\bar{X}_3 \vee \bar{X}_2X_3) \wedge (\bar{X}_1 \vee X_2X_3 \vee \bar{X}_2\bar{X}_3) = \\
 &= \bar{X}_1\bar{X}_2X_3 \vee \bar{X}_1X_2\bar{X}_3 \vee X_1\bar{X}_2\bar{X}_3 \vee X_1X_2X_3.
 \end{aligned}$$

На всіх інших тестових наборах (010, 011, 100, 101, 110, 111) логічна X-функція

$$f = \bar{X}_1\bar{X}_2X_3 \vee \bar{X}_1X_2\bar{X}_3 \vee X_1\bar{X}_2\bar{X}_3 \vee X_1X_2X_3$$

має аналогічний вираз дедуктивної формули, рівний вихідної X-функції.

Таким чином, X-функція, задана кубітним покриттям (01101001), на всіх тестових наборах має дедуктивну формулу для транспортування вхідних списків несправностей, яка дорівнює вихідної функції – результат вражаючий:

$$L(T = xxx) = f(X_1, X_2, X_3) = (\bar{X}_1\bar{X}_2X_3 \vee \bar{X}_1X_2\bar{X}_3 \vee X_1\bar{X}_2\bar{X}_3 \vee X_1X_2X_3).$$

З метою доказової повноти досліджень далі наводиться побудова дедуктивної формули транспортування списків вхідних несправностей для другої X-функції від трьох змінних:

$$f(Q = 10010110) = \bar{X}_1\bar{X}_2\bar{X}_3 \vee \bar{X}_1X_2X_3 \vee X_1\bar{X}_2X_3 \vee X_1X_2\bar{X}_3 :$$

$$\begin{aligned}
 L(T_0 = 000) &= (\bar{X}_1\bar{X}_2\bar{X}_3 \vee \bar{X}_1X_2X_3 \vee X_1\bar{X}_2X_3 \vee X_1X_2\bar{X}_3) \oplus 1 = \\
 &= (\overline{\bar{X}_1\bar{X}_2\bar{X}_3 \vee \bar{X}_1X_2X_3 \vee X_1\bar{X}_2X_3 \vee X_1X_2\bar{X}_3}) = \\
 &= \bar{X}_1\bar{X}_2\bar{X}_3 \wedge \bar{X}_1X_2X_3 \wedge X_1\bar{X}_2X_3 \wedge X_1X_2\bar{X}_3 = \\
 &= (X_1 \vee X_2 \vee X_3) \wedge (X_1 \vee \bar{X}_2 \vee \bar{X}_3) \wedge (\bar{X}_1 \vee X_2 \vee \bar{X}_3) \wedge (\bar{X}_1 \vee \bar{X}_2 \vee X_3) = \\
 &= (X_1 \vee X_2\bar{X}_3 \vee \bar{X}_2X_3) \wedge (\bar{X}_1 \vee X_2X_3 \vee \bar{X}_2\bar{X}_3) = \\
 &= \bar{X}_1\bar{X}_2X_3 \vee \bar{X}_1X_2\bar{X}_3 \vee X_1\bar{X}_2\bar{X}_3 \vee X_1X_2X_3.
 \end{aligned}$$

$$L(T_1 = 001) = (\bar{X}_1\bar{X}_2\bar{X}_3 \vee \bar{X}_1X_2X_3 \vee X_1\bar{X}_2X_3 \vee X_1X_2\bar{X}_3) \oplus 0 = \\ = (\bar{X}_1\bar{X}_2X_3 \vee \bar{X}_1X_2\bar{X}_3 \vee X_1\bar{X}_2\bar{X}_3 \vee X_1X_2X_3);$$

$$\dots \\ L(T = xxx) = \bar{X}_1\bar{X}_2X_3 \vee \bar{X}_1X_2\bar{X}_3 \vee X_1\bar{X}_2\bar{X}_3 \vee X_1X_2X_3.$$

Виявилося, що обидві X-функції (01101001, 10010110) від трьох змінних мають однаковий дедуктивний вираз для транспортування вхідних списків несправностей на вихід функції, інваріантний до вхідних тестових наборів і рівний ДДНФ функції, яка має кубітне покриття (01101001):

$$L(T = xxx, Q = 01101001 \vee 10010110) = \bar{X}_1\bar{X}_2X_3 \vee \bar{X}_1X_2\bar{X}_3 \vee X_1\bar{X}_2\bar{X}_3 \vee X_1X_2X_3.$$

Істотним є факт, що X-функція від n змінних, яка ототожнюється з хог-примітивом, є єдиною, де логічна і дедуктивна функції рівні між собою на будь-якому вхідному двійковому наборі. Зокрема, для трьох змінних рівність має вигляд:

$$f(01101001) = L(01101001) = \\ = \bar{X}_1\bar{X}_2X_3 \vee \bar{X}_1X_2\bar{X}_3 \vee X_1\bar{X}_2\bar{X}_3 \vee X_1X_2X_3 = \\ = X_1 \oplus X_2 \oplus X_3.$$

Структура для моделювання справної поведінки і дедуктивного моделювання несправностей X-функції на основі використання регістрових операцій представлена на рис. 3.3.

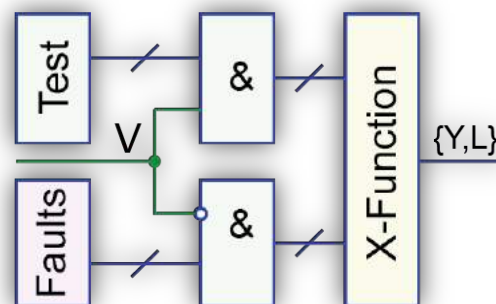


Рисунок 3.3 – Дедуктивний симулятор X-функцій

Переваги дедуктивного симулятора логічних X-функцій полягають в наступному: 1) Відсутня традиційна апаратна надмірність для дедуктивного моделювання несправностей X-функцій, в той час як всі інші логічні функції вимагають 10-кратного збільшення апаратури. 2) Для обробки вхідного впливу використовуються два автоматних тактів, які орієнтовані на справне моделювання і дедуктивний аналіз несправностей відповідно. 3) Дедуктивний симулятор використовує паралельні логічні регістрові операції для ефективного транспортування вхідних списків несправностей.

3.4 Синтез булевих похідних по кубітним покриттям

Формула для визначення похідних вхідних булевих змінних використовує хог-операцію між осередками сусідніх частин кубіта:

$$Q'(X_k) = \{Q_i^L, Q_i^R\} = Q_i^L \bigoplus_{i=1,2^{k-1}}^{k=1,n} Q_i^R.$$

Структура обчислювача-секвенсора кубітної похідної представлена на рис. 3.4, де результати виконання хог-операції заносяться в обидві частини кубіта.

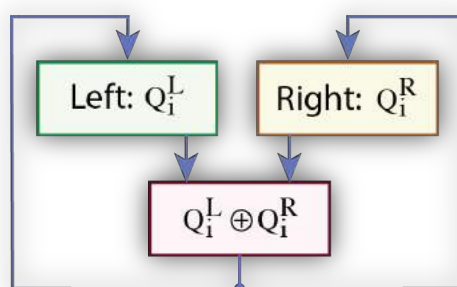


Рисунок 3.4 – Секвенсор для взяття кубітної похідної

Приклад використання секвенсора для отримання кубітних похідних для X-функції від трьох змінних представлений в наступному вигляді (верхні індекси є номером позиції кубітного вектора-покриття, куди записуються (w) результати хог-операції):

$$Q = (01101001) \leftarrow f = X_1 \oplus X_2 \oplus X_2;$$

$$Q'(X_1) = 0^0 \oplus 1^1 = 1 \xrightarrow{w} 1^0 1^1, 1^2 \oplus 0^3 = 1 \xrightarrow{w} 1^2 1^3, 1^4 \oplus 0^5 = 1 \xrightarrow{w} 1^4 1^5, 0^6 \oplus 1^7 = 1 \xrightarrow{w} 1^6 1^7;$$

$$Q'(X_2) = 0^0 1^1 \oplus 1^2 0^3 = 11 \xrightarrow{w} 1^0 1^1 2^1 3^1, 1^4 0^5 \oplus 0^6 1^7 = 11 \xrightarrow{w} 1^4 1^5 1^6 1^7;$$

$$Q'(X_3) = 0^0 1^1 1^2 0^3 \oplus 1^4 0^5 0^6 1^7 = 1111 \xrightarrow{w} 1^0 1^1 1^2 1^3 1^4 1^5 1^6 1^7.$$

У загальному випадку можна сформулювати аксіому-критерій існування дедуктивних функцій, інваріантних по відношенню до вхідних впливів: "Якщо кубітні похідні по всім змінним рівні одиничному вектору, то дедуктивна формула інваріантна до вхідних тестових наборів".

Для чотирьох логічних функцій від однієї змінної існують тільки два таких примітиви, які мають однакові одиничні вектори похідних, що отримуються шляхом виконання хог-операції на зустрічному зсуві частин кубіта:

$$Y = X; Q = 01, Q' = 11;$$

$$Y = \bar{X}; Q = 10, Q' = 11.$$

Для шістнадцяти логічних функцій від двох змінних існують також два булевих примітиви, які мають однакові одиничні вектори похідних, що отримуються шляхом виконання хог-операції на зустрічному зсуві частин кубіта:

$$Y = X_1 \oplus X_2; Q = 0110; Q' = 1111;$$

$$Y = X_1 \oplus X_2; Q = 1001; Q' = 1111.$$

Для 256 логічних функцій від трьох змінних існують також два примітиви, які мають однакові одиничні вектори похідних, що отримуються шляхом виконання хог-операції на зустрічному зсуві частин кубіта:

$$Y = X_1 \oplus X_2; Q = 01101001; Q' = 11111111;$$

$$Y = X_1 \oplus X_2; Q = 10010110; Q' = 11111111.$$

Тут два кубіта X-функцій від будь-якого кінцевого числа змінних є інверсії станів відповідних розрядів. Кожен з них може бути розділений на дві рівні за кількістю розрядів частини, де ліва є інверсією правої частини:

$$\begin{aligned}
Q(n=1) &= 01 \\
&\quad (10); \\
Q(n=2) &= 0110 \\
&\quad (1001); \\
Q(n=3) &= 01101001 \\
&\quad (10010110); \\
Q(n=4) &= 0110100110010110 \\
&\quad (1001011001101001); \\
Q(n=5) &= 0110100110010110 \ 1001011001101001 \\
&\quad (1001011001101001 \ 0110100110010110).
\end{aligned}$$

Твердження 1. Множина $|f(X)| = 2^{2^n}$ логічних функцій від кінцевого числа змінних n має тільки дві функції, для яких кубітні похідні визначені як одиничні вектори.

Твердження 2. Кількість логічних функцій від n змінних, де виконується умова

$$\bigvee_{i=1}^n \left[\frac{df}{dX_i} = 1 \right],$$

завжди дорівнює двом. Це означає, що не потрібно ніяких умов за іншими входами для активізації виходу при зміні входної змінної X_i .

Визначення. X -функція є логічна проста функція від кінцевого числа змінних ($n = 1, 2, 3, \dots$), яку неможливо мінімізувати.

Затвердження 3. Шахова форма подання нульових і одиничних станів в карті Карно задає X -функцію. Наступні два приклади логічних функцій, зображених в формі карти Карно, ілюструють дві взаємно інверсних булевих X -функції від чотирьох змінних:

$X_1X_2 \setminus X_2X_3$	00	01	11	10	$X_1X_2 \setminus X_2X_3$	00	01	11	10
00	0	1	0	1	00	1	0	1	0
01	1	0	1	0	01	0	1	0	1
11	0	1	0	1	11	1	0	1	0
10	1	0	1	0	10	0	1	0	1

$$\begin{aligned}
Y(0110100110010110) &= \bar{X}_1\bar{X}_2\bar{X}_3X_4 \vee \bar{X}_1\bar{X}_2X_3\bar{X}_4 \vee \bar{X}_1X_2\bar{X}_3\bar{X}_4 \vee \bar{X}_1X_2X_3X_4 \vee \\
&\vee X_1X_2\bar{X}_3X_4 \vee X_1X_2X_3\bar{X}_4 \vee X_1\bar{X}_2\bar{X}_3\bar{X}_4 \vee X_1\bar{X}_2X_3X_4; \\
Y(1001011001101001) &= \bar{X}_1\bar{X}_2\bar{X}_3\bar{X}_4 \vee \bar{X}_1\bar{X}_2X_3X_4 \vee \bar{X}_1X_2\bar{X}_3X_4 \vee \bar{X}_1X_2X_3\bar{X}_4 \vee \\
&\vee X_1X_2\bar{X}_3\bar{X}_4 \vee X_1X_2X_3X_4 \vee X_1\bar{X}_2\bar{X}_3X_4 \vee X_1\bar{X}_2X_3\bar{X}_4.
\end{aligned}$$

Аналогічно, за допомогою Карт Карно, можна уявити логічні Х-функції від трьох змінних, які мають 8 станів виходів:

$X_1 \setminus X_2X_3$	00	01	11	10	$X_1 \setminus X_2X_3$	00	01	11	10
0	0	1	0	1	0	1	0	1	0
1	1	0	1	0	1	0	1	0	1

Даним функціям можна поставити у відповідність їх аналітичні форми, які мають корисну властивість – кожен терм ДДНФ (Досконала Диз'юнктивна Нормальна Форма) відрізняється від інших за двома змінним:

$$\begin{aligned}
Y(01101001) &= \bar{X}_1\bar{X}_2X_3 \vee \bar{X}_1X_2\bar{X}_3 \vee X_1\bar{X}_2\bar{X}_3 \vee X_1X_2X_3; \\
Y(10010110) &= \bar{X}_1\bar{X}_2\bar{X}_3 \vee \bar{X}_1X_2X_3 \vee X_1\bar{X}_2X_3 \vee X_1X_2\bar{X}_3.
\end{aligned}$$

Це означає, що будь-яка одиночна константна несправність може бути ідентифікована і виправлена на вхідному наборі, що перевіряє. Структурна реалізація згаданих вище булевих логічних функцій ДДНФ представлена на рис. 3.5.

Програмна реалізація Хог-функції (ліва схема) виконана за допомогою програмного додатка QuaSim, який виконує синтез моделі та моделювання справної поведінки. В даному випадку виконана верифікація схемної структури на вичерпному тесті. Особливість верифікації полягає у використанні кубітних структур даних як функціонального опису елементів схеми. Сервіси додатку орієнтовані на верифікацію цифрових схем, які перевіряють тести. Візуальний інтерфейс технологічно зручний для швидкого введення схемної структури і функціональних описів в формі кубітних векторів-покриттів, що набагато ефективніше за часом, ніж введення таблиць істинності або використання складних мов опису апаратури. У середині елемента фігурує інформація про номер елемента, десятковий код його кубітного покриття та стан

виходу на заданому вхідному впливі. Обмежень на функціональну складність структури і примітивних елементів немає.

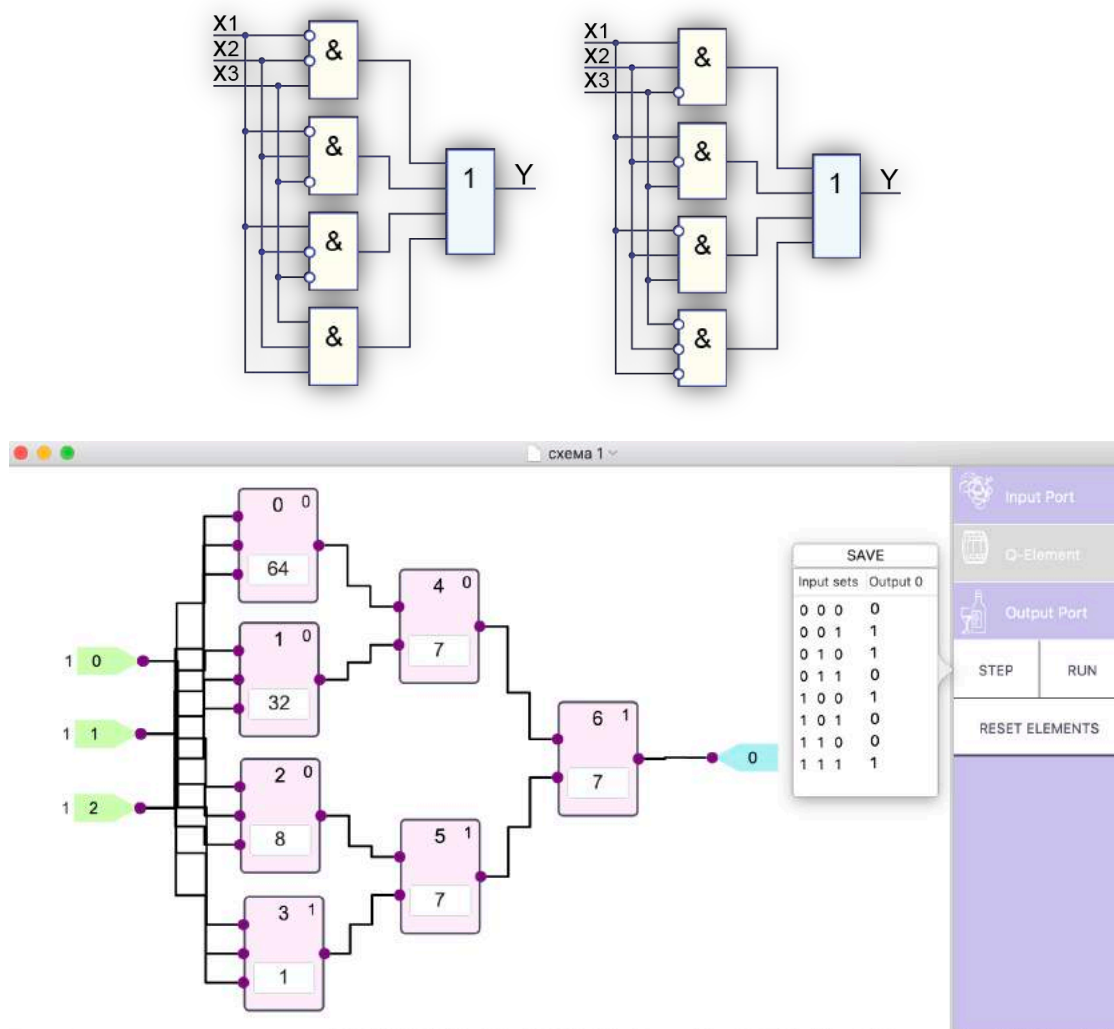


Рисунок 3.5 – Структурна і програмна реалізація булевих X-функцій

Моделювання справної поведінки всіх вхідних наборів (таблиця T) для першої X-схеми (розглядаються вхідні (3), внутрішні (4) і вихідні (1) змінні), аналіз несправностей (таблиця D), мінімізація тестових послідовностей (таблиця M) і кубітна форма мінімального тесту цифрової структури – таблиця T (Q), представлені нижче:

T	1	2	3	4	5	6	7	8	D	1	2	3	4	5	6	7	8	M	1	2	3	4	5	6	7	8	T(Q)
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1
1	0	0	1	1	0	0	0	1	1	1	1	0	0	.	.	.	0	1	1	1	0	0	.	.	.	0	1
2	0	1	0	0	1	0	0	1	2	1	0	1	.	0	.	.	0	2	1	0	1	.	0	.	.	0	0
3	0	1	1	0	0	0	0	0	3	1	0	0	1	1	1	1	1	4	0	1	1	.	.	0	.	0	1
4	1	0	0	0	0	1	0	1	4	0	1	1	.	.	0	.	0	5	0	1	0	1	1	1	1	1	0
5	1	0	1	0	0	0	0	0	5	0	1	0	1	1	1	1	1	7	0	0	0	.	.	.	0	0	0
6	1	1	0	0	0	0	0	0	6	0	0	1	1	1	1	1	1	C	x	x	x	x	x	x	x	x	0
7	1	1	1	0	0	0	1	1	7	0	0	0	.	.	.	0	0										1

Тут кожен вхідний набір перевіряє поодинокі константні несправності зовнішніх входів і виходів, що зафіксовано в таблиці D. Мінімальний тест, що покриває всі поодинокі константні несправності, представлені в таблиці M. В останньому рядку представлений результат покриття несправностей у вигляді вектора C, координати якого задані символами $x = \{0,1\}$. Стовець T(Q) = 11101001 своїми одиничними значеннями створює мінімальну кубітну форму тесту (виконавчі адреси одиничних координат), який необхідно подати на зовнішні входи, щоб перевірити всі поодинокі константні несправності зовнішніх і внутрішніх ліній цифрової логічної схеми.

Моделювання справної поведінки всіх вхідних наборів (таблиця T) для другої X-схеми, аналіз несправностей (таблиця D), мінімізація тестових послідовностей (таблиця M) і кубітна форма мінімального тесту – таблиця T(Q), представлені нижче:

T	1	2	3	4	5	6	7	8	D	1	2	3	4	5	6	7	8	M	1	2	3	4	5	6	7	8	T(Q)
0	0	0	0	0	0	0	1	1	0	1	1	1	.	.	.	0	0	0	1	1	1	.	.	.	0	0	1
1	0	0	1	0	0	0	0	0	1	1	1	0	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1
2	0	1	0	0	0	0	0	0	2	1	0	1	1	1	1	1	1	3	1	0	0	.	.	0	.	0	0
3	0	1	1	0	0	1	0	1	3	1	0	0	.	.	0	.	0	5	0	1	0	.	0	.	.	0	1
4	1	0	0	0	0	0	0	0	4	0	1	1	1	1	1	1	1	6	0	0	1	0	.	.	.	0	0
5	1	0	1	0	1	0	0	1	5	0	1	0	.	0	.	.	0	C	x	x	x	x	x	x	x	x	1
6	1	1	0	1	0	0	0	1	6	0	0	1	0	.	.	.	0										1
7	1	1	1	0	0	0	0	0	7	0	0	0	1	1	1	1	1										0

Тут кожен вхідний набір перевіряє поодинокі константні несправності зовнішніх входів і виходів, що зафіксовано в таблиці D. Мінімальний тест, що покриває всі поодинокі константні несправності, представлений в таблиці M. В останньому рядку представлений результат покриття несправностей у вигляді вектора C, координати якого задані символами $x = \{0,1\}$. Стовець T(Q) = 11101001 своїми одиничними значеннями створює мінімальну кубітну форму тесту (виконавчі адреси одиничних координат), який необхідно подати на зовнішні входи, щоб перевірити всі поодинокі константні несправності зовнішніх і внутрішніх ліній цифрової логічної схеми.

$(Q) = 110101101$ своїми одиничними значеннями створює мінімальну кубітну форму тесту (виконавчі адреси одиничних координат), який необхідно подати на зовнішні входи, щоб перевірити всі поодинокі константні несправності зовнішніх і внутрішніх ліній цифрової логічної схеми.

3.5 Синтез тестів для логічних X-функцій

Ідея синтезу кубітних тестів на основі використання операції зустрічного зсуву даних в кубітному покритті представлена формулою [40]:

$$T(S) = \bigvee_{j=1}^n [Q \oplus S_j(\bar{Q})].$$

Наступна таблиця ілюструє результати виконання чотирьох логічних операцій для отримання тесту кубітної форми:

Q – Test Synthesis	1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
\bar{Q}	0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0
$S_1(\bar{Q})$	1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1
$S_2(\bar{Q})$	1 1 0 1 1 1 1 1 1 1 1 1 1 1 0 1 1
$S_3(\bar{Q})$	1 1 1 1 0 1 1 1 1 1 1 1 0 1 1 1 1
$S_4(\bar{Q})$	1 1 1 1 1 1 1 0 0 1 1 1 1 1 1 1 1
$T_1 = Q \oplus S_1(\bar{Q})$	1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1
$T_2 = Q \oplus S_2(\bar{Q})$	1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1
$T_3 = Q \oplus S_3(\bar{Q})$	1 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 1
$T_4 = Q \oplus S_4(\bar{Q})$	1 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 1
$T(S) = T_1 \vee T_2 \vee T_3 \vee T_4$	1 1 1 0 1 0 0 1 1 0 0 1 0 1 1 1 1

Загальна довжина тесту визначається числом одиничних значень координат:

$$T(S) = T(A) = (1110100110010111).$$

З використанням булевої кубітної похідної процедура синтезу тесту спрощується до двох логічних операцій:

$$T(Q') = \bigvee_{i=1}^n Q'(X_i)$$

Результат в формі кубіта визначає тест перевірки несправностей, з числом вхідних наборів, рівним 10:

$Q'(X)$	1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
$Q'(X_1)$	1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1
$Q'(X_2)$	1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1
$Q'(X_3)$	1 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 1
$Q'(X_4)$	1 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 1
$T(Q') = \bigvee_{i=1}^n Q'(X_i)$	1 1 1 0 1 0 0 1 1 0 0 1 0 1 1 1

Обидва методи синтезу тестів дають ідентичний результат:

$$T(Q') = T(S) = T(A) = (1110100110010111)$$

Проте, процедури синтезу тестів для довільних логічних функцій досить непрості для їх реалізації навіть на кубітних структурах даних, де потрібні витратні процедури реєстрового зсуву, які в програмній реалізації є часовитратними.

Сутність або унікальна особливість тесту для будь-якої X -функції полягає в його непобудованості, оскільки існують дві T -аксіоми, що пояснюють відсутність синтезу: 1) Логічна X -функція, записана у вигляді поодиноких термів ДДНФ $T_i \in T, f(T_i) = 1$, являє собою тестові набори для перевірки одиначних константних 0-несправностей вхідних, внутрішніх змінних і виходу:

$$T^1(\equiv 0) = \bigvee_{\forall i[f(T_i)=1]} T_i,$$

$$T = \{T_0, T_1, \dots, T_i, \dots, T_k\},$$

$$k = 2^{2^n} - 1; f(T_i) = \{0, 1\}.$$

2) Тест в формі ДДНФ логічної X -функції завжди доповнюється єдиною вхідною послідовністю – будь-яким термом зворотньої ДДНФ

$T_i \in T^0, f(T_i) = 0$, який перевіряє всі константні 1-несправності внутрішніх ліній і виходу:

$$T^0 (\equiv 1) = T_i \leftarrow \forall i : f(T_i) = 0.$$

Зокрема, доповнення до 1-тесту X-функції ($Q = 01101001$) визначається нульовим тестовим набором (000) за всіма вхідними координатами; доповнення до 1-тесту для X-функції ($Q = 10010110$) задається першим тестовим набором (001), що містить в останньому розряді одиницю на тлі всіх інших нулів. Насправді доповненням до 1-тесту є будь-яка вхідна послідовність, яка відсутня в 1-тесті. Тому повним перевіряючим тестом для

$$Q = 1 + \frac{1}{2} \times 2^{2^n}.$$

X-функції завжди є тест розмірністю

Отже, тестом для логічної X-функції від n змінних є її ДДНФ, доповнена на будь-яким термом зворотньої ДДНФ даної функції:

$$\begin{aligned} T &= T^1 \vee T_i^0, \\ T^1 &= \forall T_i : f(T_i) = 1 \\ T_i^0 \in T^0 &= \forall T_i : f(T_i) = 0; \\ T(01101001) &= (001 \vee 010 \vee 100 \vee 111) \vee 000; \\ T(10010110) &= (000 \vee 011 \vee 101 \vee 110) \vee 001. \end{aligned}$$

Інша унікальна властивість логічної X-функції визначається як: покоординатна хог-сума всіх двійкових кодів, відповідних ДДНФ (зворотної ДДНФ), дорівнює нульовому за всіма двійковими розрядами, в форматі n змінних, вектору:

$$\begin{aligned} \bigoplus_{i=1}^{n_1} C(T_i^1) &= 0; \\ \bigoplus_{i=1}^{n_0} C(T_i^0) &= 0; \\ n_0 + n_1 &= 2^{2^n}. \end{aligned}$$

Ілюстрація факту конволюція простору ДДНФ або зворотної ДДНФ в нуль-вектор представлена двома Х-функціями від трьох змінних:

$$\begin{aligned}
 Y(01101001) &= \bar{X}_1\bar{X}_2X_3 \vee \bar{X}_1X_2\bar{X}_3 \vee X_1\bar{X}_2\bar{X}_3 \vee X_1X_2X_3 \rightarrow \\
 &\rightarrow 001 \oplus 010 \oplus 100 \oplus 111 = 000; \\
 Y(10010110) &= \bar{X}_1\bar{X}_2\bar{X}_3 \vee \bar{X}_1X_2X_3 \vee X_1\bar{X}_2X_3 \vee X_1X_2\bar{X}_3 \rightarrow \\
 &\rightarrow 000 \oplus 011 \oplus 101 \oplus 110 = 000.
 \end{aligned}$$

Властивість конволюції дає можливість: 1) верифікувати якість тестових послідовностей без застосування складних процедур моделювання несправностей; 2) обчислювати невідомі терми тесту або ДДНФ на основі відомих компонентів шляхом застосування, наприклад, наступної рівності для Х-функції від трьох змінних:

$$\begin{aligned}
 T_1 \oplus T_2 \oplus T_3 \oplus T_4 &= 0; \\
 001 \oplus 010 \oplus 100 \oplus 111 &= 0(000). \\
 T_2 \oplus T_3 \oplus T_4 &= T_1; \\
 010 \oplus 100 \oplus 111 &= 001.
 \end{aligned}$$

Для логічних Х-функцій від двох змінних (рис. 3.6), які відомі як xor, not-xor примітиви, нижче представлено моделювання справної поведінки всіх вхідних наборів (таблиця T), аналіз несправностей (таблиця D) і кубітна форма чотирьох варіантів мінімальних тестів – таблиця T (Q):

T(xor)	1 2 3 4 5		D	1 2 3 4 5		T(Q)	1 2 3 4
0	0 0 0 0 0	→	0	1 1 1 1 1	→	0	0 1 1 1
1	0 1 0 1 1		1	1 0 . 0 0		1	1 0 1 1
2	1 0 1 0 1		2	0 1 0 . 0		2	1 1 0 1
3	1 1 0 0 0		3	0 0 1 1 1		3	1 1 1 0

T(not-xor)	1 2 3 4 5		D	1 2 3 4 5		T(Q)	1 2 3 4
0	0 0 1 0 1	→	0	1 1 0 . 1	→	0	0 1 1 1
1	0 1 0 0 0		1	1 0 1 1 0		1	1 0 1 1
2	1 0 0 0 0		2	0 1 1 1 0		2	1 1 0 1
3	1 1 0 1 1		3	0 0 . 0 1		3	1 1 1 0

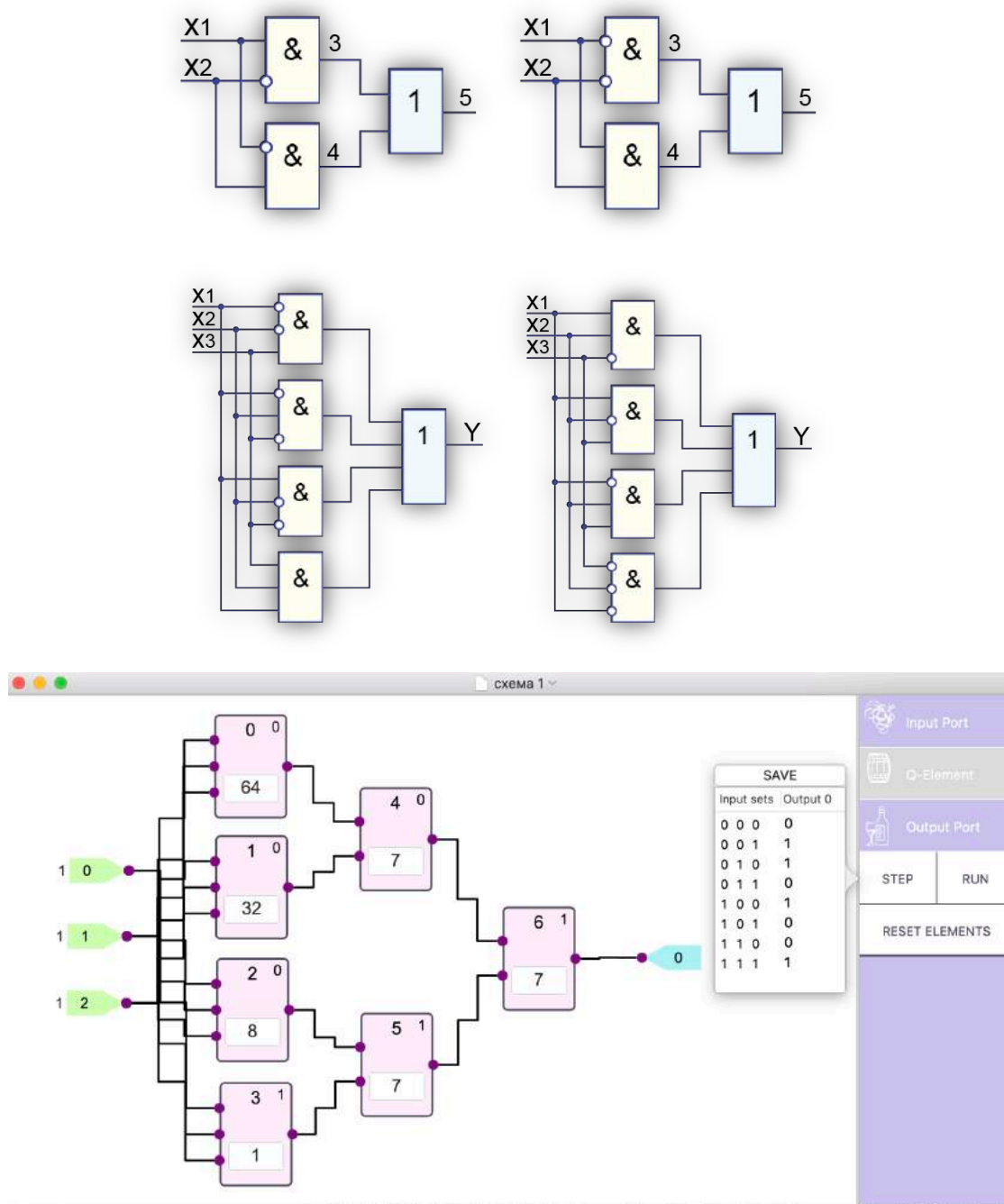


Рисунок 3.6 – Схеми X-функції від двох змінних

Для обох функцій тут отримані мінімальні тести, що складаються з трьох вхідних наборів. Це пов'язано з тим, що протилежні вхідні вектори мають однакові стани вихідної змінної.

Тест, який використовує Т-аксіоми для запису вхідних послідовностей, які перевіряють всі поодинокі константні несправності цифрових схем (xor, nxr), має наступний вигляд:

$$\begin{aligned} T(\text{xor}, \text{nxr}) &= T^1 \vee T_1^0, \\ T(Q = 0110)(\text{xor}) &= (01 \vee 10) \vee 00; \\ T(Q = 1001)(\text{nxr}) &= (00 \vee 11) \vee 01. \end{aligned}$$

Для двох логічних Х-функцій від однієї змінної (рис. 3.7)

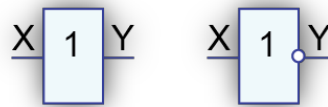


Рисунок 3.7 – Х-функції від однієї змінної

нижче представлено моделювання справної поведінки всіх вхідних наборів (таблиця Т), аналіз несправностей (таблиця D) і кубітна форма мінімальних тестів – таблиця Т (Q), розглядаються функціональні елементи (повторювач – rep, інвертор – not):

T(rep)	X	Y	D	X	Y	T(Q)
0	0	0	0	1	1	1
1	1	1	1	0	0	1

T(not)	X	Y	D	X	Y	T(Q)
0	0	1	0	1	0	1
1	1	0	1	0	1	1

Тест, який використовує Т-аксіоми для запису вхідних послідовностей, які перевіряють всі поодинокі константні несправності цифрових примітивів (rep, not), має наступний вигляд:

$$\begin{aligned} T(\text{rep}, \text{not}) &= T^1 \vee T_1^0, \\ T(Q = 01)(\text{rep}) &= 1 \vee 0; \\ T(Q = 10)(\text{not}) &= 0 \vee 1. \end{aligned}$$

Таким чином, використання двох згаданих вище T-аксіом дає можливість записувати без обчислень повний перевіряючий тест для одиночних константних несправностей вхідних, внутрішніх і вихідних ліній будь-якої як завгодно складної логічної X-функції.

3.6. Висновки до розділу 3

1) Розроблено структурну модель взаємодії X-функцій і похідних компонентів, орієнтованих на синтез і аналіз цифрових систем з метою отримання тестопригодності рішень, пов'язаних зі зменшенням часу проектування і тестування обчислювальних пристроїв.

2) Вперше введено поняття простих X-функцій від кінцевого числа змінних, які характеризуються відсутністю мінімізації та наявністю властивостей тестопригодності, що дає можливість синтезувати цифрові пристрої, технологічні для вирішення завдань тестування, моделювання та діагностування.

3) Сформульовано метричні властивості X-функцій від кінцевого числа змінних, які дають можливість використовувати їх в практиці розробки тестопригодності цифрових пристроїв, генерування перевіряючих тестів і оцінки їх якості шляхом дедуктивного моделювання перевірюваних константних несправностей на кубітних структурах даних.

4) Запропоновано кубітний метод синтезу тестів для X-функцій, який відрізняється лінійною обчислювальною складністю алгоритму для отримання перевіряючих послідовностей.

5) Подальші дослідження пов'язані зі створенням технологій перетворення фрагментів логічних схем до форми X-функцій, які технологічні для вирішення завдань тестування і верифікації цифрових систем.

Результати виображено у роботах [2, 3, 5, 7, 9, 10, 14].

РОЗДІЛ 4

СИНТЕЗ І АНАЛІЗ ЛОГІЧНИХ Х-ФУНКЦІЙ

Представлені моделі і методи кубітного синтезу та аналізу логічних Х-функцій (xor, not-xor) від n змінних [74-80], які є потужним математичним засобом для вирішення завдань генерації тестів, моделювання несправностей, створення тестопридатних схем. Їх основна перевага полягає в можливості перевірки несправностей, які є інверсними по відношенню до справної поведінки логічної схеми. Це означає, що будь-який вхідний набір перевіряє 50 відсотків всіх вхідних дефектів. Дана властивість використовується тестувальниками для синтезу тестопридатних схем, дедуктивних формул моделювання несправностей [74-75].

Мета дослідження – аналіз тестопридатності властивостей логічних Х-функцій від n змінних, а також їх синтез шляхом розробки рекурсивних моделей і формул, що використовують симетрію властивостей їх кубітних покриттів.

Завдання: 1) Формування моделі чудових властивостей Х-функцій, які використовуються для їх синтезу та аналізу. 2) Створення архітектури секвенсора для синтезу Q-покриттів. Х-функцій від кінцевого числа змінних. 3) Розробка методу синтезу дедуктивних ДНФ для моделювання несправностей на основі аналізу таблиць істинності. 4) Розробка методу паралельного синтезу дедуктивних кубітних покриттів і дедуктивних ДНФ для моделювання несправностей на основі аналізу кубітних векторів опису функціональностей. 5) Розробка методу паралельного синтезу матриць кубітних покриттів дедуктивного аналізу на основі властивостей симетрії евристично розроблених матриць перестановки бітів. 6) Розробка методу синтезу тестів для логічних Х-функцій шляхом використання перестановки бітів в кубітних покриттях.

4.1. Властивості логічних X-функцій

Практично корисними для синтезу та аналізу цифрових схем можуть бути наступні 12 властивостей X-функцій, інтегровані в модель відношень, представлені на рис. 4.1.

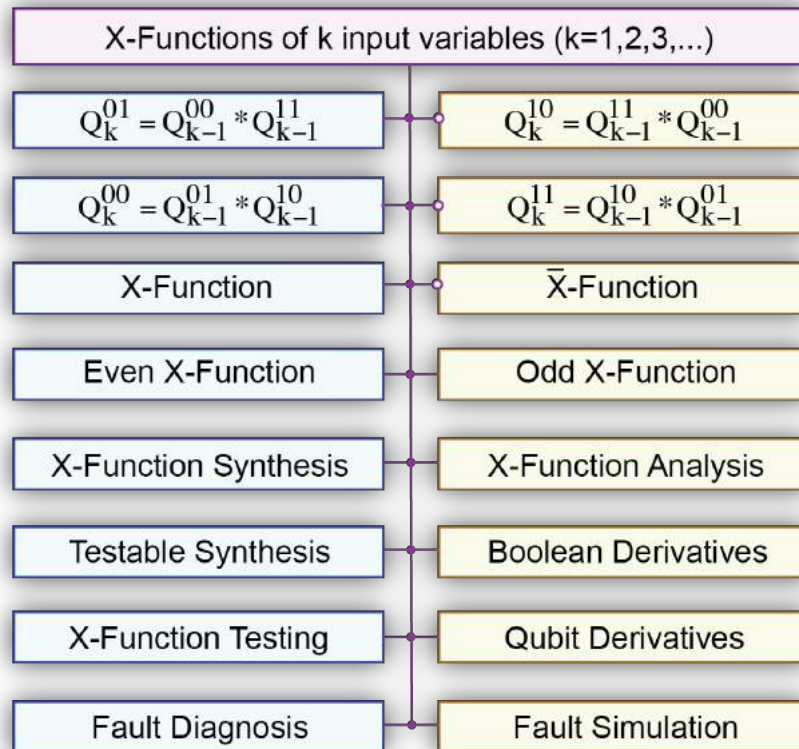


Рисунок 4.1 – Структурна модель взаємодії X-функцій

1) Кубітне покриття X-функції має рівну кількість нульових і одиничних координат.

2) Кількість X-функцій від n булевих змінних завжди дорівнює двом:

$$Q^{2^x}(n) = Q^x(n) \vee \bar{Q}^x(n).$$

Стан координат кубітних покриттів обох X-функцій від n змінних є взаємно інверсними.

3) X-функції від однієї логічної булевої змінної представлені повторувачем і інвертором: $Y = X$; $Y = \bar{X}$.

4) X-функції від двох булевих змінних представлені відомими логічними примітивами xor, not-xor: $Y = X_1\bar{X}_2 \vee \bar{X}_1X_2$; $Y = X_1X_2 \vee \bar{X}_1\bar{X}_2$.

5) Кубітна похідна за будь-якої змінної X-функції дорівнює одиничному вектору. Булева похідна по будь-якою змінною X-функції дорівнює одиниці.

6) Для активізації вхідної змінної X-функції з метою зміни виходу не потрібно ніяких умов за станами інших змінних.

7) Пара вхідних наборів, що має інверсні сигнали за всіма координатами, завжди змінює стан виходу X-функції від непарного числа змінних. Зміна стану входу X-функції завжди призводить до зміни стану виходу.

8) Синтез двох X-функцій від n змінних реалізується шляхом конкатенації (*) кубітних векторів X-функцій від n-1 змінної:

$$Q^{2^x(n)} = Q^x(n-1) * \bar{Q}^x(n-1) \vee \bar{Q}^x(n-1) * Q^x(n-1).$$

Структура секвенсора, призначеного для синтезу кубітних покриттів X-функцій від $n = 1, 2, 3, 4$ змінних за допомогою конкатенації і інверсії, позначеної кружечком, представлена на рис. 4.2.

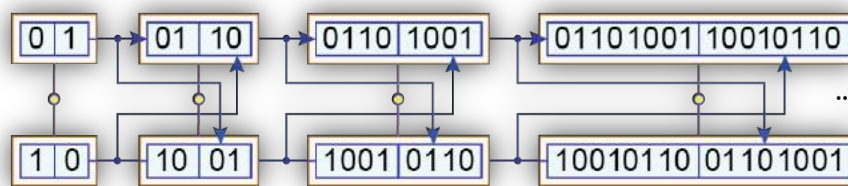


Рисунок 4.2 – Секвенсор синтезу Q-покриття для X-функції

Інша інтерпретація X-функцій пов'язана з їх ідентифікацією за першим і останнім бітом кубітного покриття: 01-10, 00-11. Тоді синтез кубітних векторів X-функцій від k змінних можна реалізувати шляхом застосування операції конкатенації до двох Q-покриттів X-функцій від k-1 змінних:

$$\begin{aligned}
Q_k^{01} &= Q_{k-1}^{00} * Q_{k-1}^{11}; \\
Q_k^{10} &= Q_{k-1}^{11} * Q_{k-1}^{00}; \\
Q_k^{00} &= Q_{k-1}^{01} * Q_{k-1}^{10}; \\
Q_k^{11} &= Q_{k-1}^{10} * Q_{k-1}^{01}.
\end{aligned}$$

Природно, що між двома кубітними покриттями Х-функцій від k змінних існує взаємно однозначне ставлення інверсії:

$$\begin{aligned}
Q_k^{01} &= \bar{Q}_k^{10}; \\
Q_k^{10} &= \bar{Q}_k^{01}; \\
Q_k^{00} &= \bar{Q}_k^{11}; \\
Q_k^{11} &= \bar{Q}_k^{00}.
\end{aligned}$$

Це означає, якщо відома одна Х-функція від k змінних, то легко можна отримати другу функцію, як двійкове доповнення до першої.

9) Будь-який вхідний набір для Х-функції перевіряє 50% несправностей за зовнішніми входами, які є інверсними по відношенню до станів справної поведінки згаданих ліній. Два взаємно-інверсних тестових набори перевіряють всі поодинокі константні несправності вхідних змінних і виходу в Х-функції від непарного числа змінних.

10) Дедуктивна формула Х-функції транспортує на вихід симетричеську різницю вхідних списків несправностей. Це означає, об'єднання вхідних списків перевірюваних дефектів за винятком випадку, коли списки несправностей на всіх входах ідентичні. Х-функція від n змінних, яка ототожнюється з хог-примітивом, є єдиною, де логічна і дедуктивна функції рівні між собою на будь-якому вхідному двійковому наборі.

11) Тестом для одиночних константних несправностей всіх ліній логічної Х-функції від n змінних є її ДДНФ, доповнена будь-яким термом зворотньої ДДНФ даної функції:

$T = T^1 \vee T_i^0$. Розмірність повного перевіряючого

тесту для Х-функції завжди дорівнює $Q = 1 + \frac{1}{2} \times 2^{2^n}$.

12) Покоординатно хог-сума всіх кодів Х-функції, що відповідають ДДНФ (зворотна ДДНФ), дорівнює нульовому (по всіх координатах) вектору.

Таким чином, логічні Х-функції, що володіють унікальними властивостями тестування, можуть бути використані для синтезу тестопридатності і самовідновлювання логічних цифрових пристроїв, а також для транспортування дефектів від зовнішніх входів до виходів булевої структури.

4.2 Метод синтезу дедуктивного кубітного покриття і ДНФ по таблиці істинності

Метод синтезу дедуктивної формули по аналітичній формі функціональності [77-79], представлена у вигляді ДНФ, є досить складною і нерегулярною обчислювальною процедурою, яка потребує спрощення. Далі пропонується проста і тривіальна формула-процедура отримання для вхідної послідовності (Т) дедуктивної таблиці (L), а по ній і записи дедуктивної формули для функціональності від n змінних, заданої таблицею істинності (С) або кубічним покриттям функціональності (рис. 4.3):

$$L_i = T_i \oplus C_j, \{i, j\} = 1, 2^{2^n}.$$

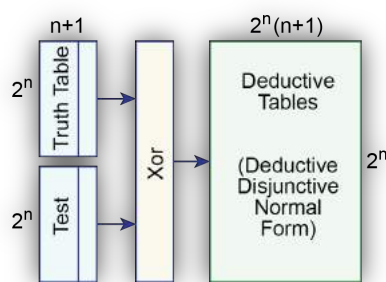


Рисунок 4.3 – Секвенсор синтезу дедуктивних таблиць для тесту

Для розуміння процедури синтезу дедуктивної таблиці нижче пропонуються логічні примітиви і результати, які можна використовувати для запису дедуктивних виразів в формі ДНФ за одиничними значеннями виходів логічних елементів (and, or, xor, not-xor, not, rep):

C			$T_0 = 000$	$T_1 = 010$	$T_2 = 100$	$T_3 = 111$
X_1	X_2	Y_{and}	$L_0 = T_0 \oplus C$	$L_1 = T_1 \oplus C$	$L_2 = T_2 \oplus C$	$L_3 = T_3 \oplus C$
0	0	0	000	010	100	111
0	1	0	010	000	110	101
1	0	0	100	110	000	011
1	1	1	111	101	011	000

C			$T_0 = 000$	$T_1 = 011$	$T_2 = 101$	$T_3 = 111$
X_1	X_2	Y_{or}	$L_0 = T_0 \oplus C$	$L_1 = T_1 \oplus C$	$L_2 = T_2 \oplus C$	$L_3 = T_3 \oplus C$
0	0	0	000	011	101	111
0	1	1	011	000	110	100
1	0	1	101	110	000	010
1	1	1	111	100	010	000

C			$T_0 = 000$	$T_1 = 011$	$T_2 = 101$	$T_3 = 110$
X_1	X_2	Y_{xor}	$L_0 = T_0 \oplus C$	$L_1 = T_1 \oplus C$	$L_2 = T_2 \oplus C$	$L_3 = T_3 \oplus C$
0	0	0	000	011	101	110
0	1	1	011	000	110	101
1	0	1	101	110	000	011
1	1	0	110	101	011	000

C			$T_0 = 001$	$T_1 = 010$	$T_2 = 100$	$T_3 = 111$
X_1	X_2	Y_{nxr}	$L_0 = T_0 \oplus C$	$L_1 = T_1 \oplus C$	$L_2 = T_2 \oplus C$	$L_3 = T_3 \oplus C$
0	0	1	000	011	101	110
0	1	0	011	000	110	101
1	0	0	101	110	000	011
1	1	1	110	101	011	000

C		$T_0 = 01$	$T_1 = 10$
X_1	Y_{not}	$L_0 = T_0 \oplus C$	$L_1 = T_1 \oplus C$
0	1	00	11
1	0	11	00

C		$T_0 = 00$	$T_1 = 11$
X_1	Y_{rep}	$L_0 = T_0 \oplus C$	$L_1 = T_1 \oplus C$
0	0	00	11
1	1	11	00

Представлені таблиці істинності не є впорядкованими за зростанням двійково-десяткових кодів вхідних наборів. Оскільки таблиця істинності є множина відносин, то такого упорядкування не потрібно, щоб записати аналітичну форму отриманої функціональності. Для кожної функції і кожного вхідного набору нижче записані дедуктивні функції примітивних елементів (and, or, xor, not-xor, not, rep) в формі ДДНФ, які можна використо-

увати як для апаратного синтезу логічних схем моделювання несправностей, так і для створення хмарного програмного сервісу, що виконує дедуктивний аналіз цифрових проектів:

$$\begin{aligned}L_{\text{and}}(000) &= X_1 X_2; \\L_{\text{and}}(010) &= X_1 \bar{X}_2; \\L_{\text{and}}(100) &= \bar{X}_1 X_2; \\L_{\text{and}}(111) &= \bar{X}_1 X_2 \vee X_1 \bar{X}_2 \vee X_1 X_2 = X_1 \vee X_2;\end{aligned}$$

$$\begin{aligned}L_{\text{or}}(000) &= \bar{X}_1 X_2 \vee X_1 \bar{X}_2 \vee X_1 X_2 = X_1 \vee X_2; \\L_{\text{or}}(011) &= \bar{X}_1 X_2; \\L_{\text{or}}(101) &= X_1 \bar{X}_2; \\L_{\text{or}}(111) &= X_1 X_2.\end{aligned}$$

$$\begin{aligned}L_{\text{xor}}(000) &= \bar{X}_1 X_2 \vee X_1 \bar{X}_2; \\L_{\text{xor}}(011) &= \bar{X}_1 X_2 \vee X_1 \bar{X}_2; \\L_{\text{xor}}(101) &= \bar{X}_1 X_2 \vee X_1 \bar{X}_2; \\L_{\text{xor}}(110) &= \bar{X}_1 X_2 \vee X_1 \bar{X}_2.\end{aligned}$$

$$\begin{aligned}L_{\text{nxr}}(001) &= \bar{X}_1 X_2 \vee X_1 \bar{X}_2; \\L_{\text{nxr}}(010) &= \bar{X}_1 X_2 \vee X_1 \bar{X}_2; \\L_{\text{nxr}}(100) &= \bar{X}_1 X_2 \vee X_1 \bar{X}_2; \\L_{\text{nxr}}(111) &= \bar{X}_1 X_2 \vee X_1 \bar{X}_2.\end{aligned}$$

$$\begin{aligned}L_{\text{not}}(01) &= X_1; \\L_{\text{not}}(10) &= X_1.\end{aligned}$$

$$\begin{aligned}L_{\text{rep}}(00) &= X_1; \\L_{\text{rep}}(11) &= X_1.\end{aligned}$$

Тут взаємодія логічних змінних визначає перетин, віднімання або об'єднання списків, або векторів несправностей $(X_1 X_2, X_1 \bar{X}_2, X_1 \vee X_2)$ відповідно, що належать зовнішнім входам примітивів.

Таким чином, представлений метод для синтезу дедуктивних покриттів і ДНФ, орієнтованих на дедуктивний аналіз цифрових систем, істотно відрізняється від існуючих аналогів швидкодією Q , яка визначається паралелізмом виконання регістрових операцій між черговим вхідним впливом і n кубами покриття: $Q = 2^{2^n}$.

4.3. Метод синтезу дедуктивного кубітного покриття і ДНФ по кубітним покриттям функціональностей

Ще більш високу швидкодію обчислювальних процедур для синтезу дедуктивних функцій можна отримати, якщо використовувати замість кубічних покриттів (таблиць) кубітні вектори. Для цього необхідно впорядкувати отримані раніше кубічні форми дедуктивних функції відповідно до зростання двійково-десяткових кодів вхідних наборів:

C			$T_0 = 000$	$T_1 = 010$	$T_2 = 100$	$T_3 = 111$
X_1	X_2	Y_{and}	$L_0 = T_0 \oplus C$	$L_1 = T_1 \oplus C$	$L_2 = T_2 \oplus C$	$L_3 = T_3 \oplus C$
0	0	0	000	000	000	000
0	1	0	010	010	011	011
1	0	0	100	101	100	101
1	1	1	111	110	110	111

C			$T_0 = 000$	$T_1 = 011$	$T_2 = 101$	$T_3 = 111$
X_1	X_2	Y_{or}	$L_0 = T_0 \oplus C$	$L_1 = T_1 \oplus C$	$L_2 = T_2 \oplus C$	$L_3 = T_3 \oplus C$
0	0	0	000	000	000	000
0	1	1	011	011	010	010
1	0	1	101	100	101	100
1	1	1	111	110	110	111

C			$T_0 = 000$	$T_1 = 011$	$T_2 = 101$	$T_3 = 110$
X_1	X_2	Y_{xor}	$L_0 = T_0 \oplus C$	$L_1 = T_1 \oplus C$	$L_2 = T_2 \oplus C$	$L_3 = T_3 \oplus C$
0	0	0	000	000	000	000
0	1	1	011	011	011	011
1	0	1	101	101	101	101
1	1	0	110	110	110	110

C			$T_0 = 001$	$T_1 = 010$	$T_2 = 100$	$T_3 = 111$
X_1	X_2	Y_{nxr}	$L_0 = T_0 \oplus C$	$L_1 = T_1 \oplus C$	$L_2 = T_2 \oplus C$	$L_3 = T_3 \oplus C$
0	0	1	000	000	000	000
0	1	0	011	011	011	011
1	0	0	101	101	101	101
1	1	1	110	110	110	110

C		$T_0 = 01$	$T_1 = 10$
X_1	Y_{not}	$L_0 = T_0 \oplus C$	$L_1 = T_1 \oplus C$
0	1	00	00
1	0	11	11

C		$T_0 = 00$	$T_1 = 11$
X_1	Y_{rep}	$L_0 = T_0 \oplus C$	$L_1 = T_1 \oplus C$
0	0	00	00
1	1	11	11

Замість упорядкованих за зростанням двійково-десяткових кодів вхідних наборів таблиць істинності можна записувати кубітні покриття, які представляють собою дедуктивну матрицю, розмірністю $m \times n$:

and	$Q_1 = 0$	$Q_2 = 0$	$Q_3 = 0$	$Q_4 = 1$
Q	$L_1 = Q_1 \oplus Q$	$L_2 = Q_2 \oplus Q$	$L_3 = Q_3 \oplus Q$	$L_4 = Q_4 \oplus Q$
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
1	1	0	0	1

or	$Q_1 = 0$	$Q_2 = 1$	$Q_3 = 1$	$Q_4 = 1$
Q	$L_1 = Q_1 \oplus Q$	$L_2 = Q_2 \oplus Q$	$L_3 = Q_3 \oplus Q$	$L_4 = Q_4 \oplus Q$
0	0	0	0	0
1	1	1	0	0
1	1	0	1	0
1	1	0	0	1

xor	$Q_1 = 0$	$Q_2 = 1$	$Q_3 = 1$	$Q_4 = 0$
Q	$L_1 = Q_1 \oplus Q$	$L_2 = Q_2 \oplus Q$	$L_3 = Q_3 \oplus Q$	$L_4 = Q_4 \oplus Q$
0	0	0	0	0
1	1	1	1	1
1	1	1	1	1
0	0	0	0	0

nxr	$Q_1 = 1$	$Q_2 = 0$	$Q_3 = 0$	$Q_4 = 1$
Q	$L_1 = Q_1 \oplus Q$	$L_2 = Q_2 \oplus Q$	$L_3 = Q_3 \oplus Q$	$L_4 = Q_4 \oplus Q$
1	1	1	1	1
0	0	0	0	0
0	0	0	0	0
1	1	1	1	1

not	$Q_1 = 1$	$Q_2 = 0$
Q	$L_1 = Q_1 \oplus Q$	$L_2 = Q_2 \oplus Q$
1	0	0
0	1	1

rep	$Q_1 = 0$	$Q_2 = 1$
Q	$L_1 = Q_1 \oplus Q$	$L_2 = Q_2 \oplus Q$
0	0	0
1	1	1

Далі пропонується більш простий шлях отримання матриці кубітних дедуктивних векторів без використання таблиць істинності на основі тільки кубітного покриття функціональності.

Метод синтезу дедуктивної матриці для паралельного аналізу одиночних константних несправностей на основі Q-покриття функціональності містить дві операції (рис. 4.4):

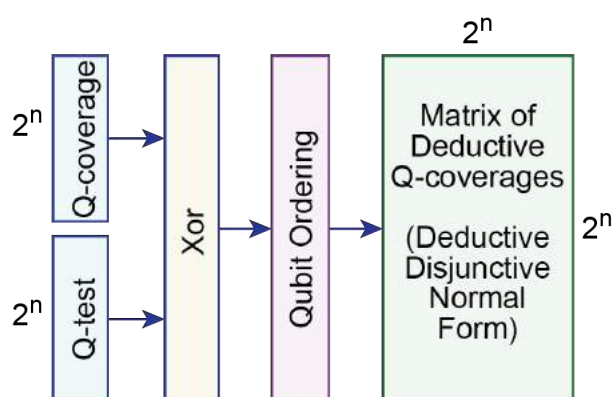


Рисунок 4.4 – Секвенсор синтезу дедуктивних покриттів для Q-тесту

1) Синтез матриці кубітних покриттів дедуктивних функцій, що залежить від двійково-десятькового номера вхідного набору і координат Q-вектора функціональності від n змінних:

$$L_{ij}^* = (Q_i \oplus Q_j)_{j=1}^m, m = 2^{2^n}.$$

Фактично береться перша координата кубітного покриття, яка хог-складається з усіма координатами Q-вектора для формування дедуктивного вектора на першій вхідній послідовності. Потім береться друга координата Q-вектора, яка також хог-складається з усіма координатами. Процедура закінчується після того, як всі координати Q-вектора були хог-складені з кубітним вектором. Обчислювальна складність даної процедури дорівнює $m \cdot 2$, яка

може бути зменшена до m шляхом апаратної паралельної реалізації хог-операції.

2) Після отримання дедуктивної матриці на всіх вхідних наборах по кубітному покриттю функціональності необхідно виконати процедуру перестановки бітів у стовпчиках за правилом

$$L_{ij} = [L^* (H_{ij})]_{\{i,j\}=1}^m$$

відповідно до номерів, представлених в матриці перестановки, яка за розмірністю дорівнює матриці кубітних покриттів дедуктивних функцій. Далі представлені приклади отримання дедуктивних функцій на основі використання тільки кубітних покриттів функціональностей:

Q _{and}	L ₁	L ₂	L ₃	L ₄
0	0	0	0	1
0	0	0	0	1
0	0	0	0	1
1	1	1	1	0

 \rightarrow

1	2	3	4
2	1	4	3
3	4	1	2
4	3	2	1

 \rightarrow

X ₁ X ₂	L ₁	L ₂	L ₃	L ₄
00	0	0	0	0
01	0	0	1	1
10	0	1	0	1
11	1	0	0	1

 \rightarrow

$$\rightarrow L = (00)(X_1 X_2) \vee (01)(X_1 \bar{X}_2) \vee (10)(\bar{X}_1 X_2) \vee (11)(X_1 \vee X_2).$$

Q _{or}	L ₁	L ₂	L ₃	L ₄
0	0	1	1	1
1	1	0	0	0
1	1	0	0	0
1	1	0	0	0

 \rightarrow

1	2	3	4
2	1	4	3
3	4	1	2
4	3	2	1

 \rightarrow

X ₁ X ₂	L ₁	L ₂	L ₃	L ₄
00	0	0	0	0
01	1	1	0	0
10	1	0	1	0
11	1	0	0	1

 \rightarrow

$$\rightarrow L = (00)(X_1 \vee X_2) \vee (01)(\bar{X}_1 X_2) \vee (10)(X_1 \bar{X}_2) \vee (11)(X_1 X_2).$$

Q _{xor}	L ₁	L ₂	L ₃	L ₄
0	0	1	1	0
1	1	0	0	1
1	1	0	0	1
0	0	1	1	0

 \rightarrow

1	2	3	4
2	1	4	3
3	4	1	2
4	3	2	1

 \rightarrow

X ₁ X ₂	L ₁	L ₂	L ₃	L ₄
00	0	0	0	0
01	1	1	1	1
10	1	1	1	1
11	0	0	0	0

 \rightarrow

$$\rightarrow L = (00 \vee 01 \vee 10 \vee 11)(\bar{X}_1 X_2 \vee X_1 \bar{X}_2) = (xx)(\bar{X}_1 X_2 \vee X_1 \bar{X}_2).$$

Q _{nrx}	L ₁	L ₂	L ₃	L ₄
1	0	1	1	0
0	1	0	0	1
0	1	0	0	1
1	0	1	1	0

 \rightarrow

1	2	3	4
2	1	4	3
3	4	1	2
4	3	2	1

 \rightarrow

X ₁ X ₂	L ₁	L ₂	L ₃	L ₄
00	0	0	0	0
01	1	1	1	1
10	1	1	1	1
11	0	0	0	0

 \rightarrow

$$\rightarrow L = (00 \vee 01 \vee 10 \vee 11)(\bar{X}_1 X_2 \vee X_1 \bar{X}_2) = (xx)(\bar{X}_1 X_2 \vee X_1 \bar{X}_2).$$

$$\begin{array}{c|cc} Q_{\text{not}} & L_1 & L_2 \\ \hline 1 & 0 & 1 \\ 0 & 1 & 0 \end{array} \rightarrow \begin{array}{c|c} 1 & 2 \\ \hline 2 & 1 \end{array} \rightarrow \begin{array}{c|cc} X & L_1 & L_2 \\ \hline 0 & 0 & 0 \\ 1 & 1 & 1 \end{array} \rightarrow \\ \rightarrow L = (0 \vee 1)(X \vee X) = (xx)(X).$$

$$\begin{array}{c|cc} Q_{\text{rep}} & L_1 & L_2 \\ \hline 0 & 0 & 1 \\ 1 & 1 & 0 \end{array} \rightarrow \begin{array}{c|c} 1 & 2 \\ \hline 2 & 1 \end{array} \rightarrow \begin{array}{c|cc} X & L_1 & L_2 \\ \hline 0 & 0 & 0 \\ 1 & 1 & 1 \end{array} \rightarrow \\ \rightarrow L = (0 \vee 1)(X \vee X) = (xx)(X).$$

4.4. Метод синтезу матриці перестановки бітів для формування дедуктивних функцій

У загальному випадку метод синтезу матриці перестановки бітів у стовпцях формування дедуктивних функцій від n змінних може бути представлений в наступному вигляді:

$$H_{ij}(n=1,2,3) = \begin{array}{c|c} 1 & 2 \\ \hline 2 & 1 \end{array} = \begin{array}{c|cccc} 1 & 2 & 3 & 4 \\ \hline 2 & 1 & 4 & 3 \\ 3 & 4 & 1 & 2 \\ 4 & 3 & 2 & 1 \end{array} = \begin{array}{c|cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ \hline 2 & 1 & 4 & 3 & 6 & 5 & 8 & 7 \\ 3 & 4 & 1 & 2 & 7 & 8 & 5 & 6 \\ 4 & 3 & 2 & 1 & 8 & 7 & 6 & 5 \\ 5 & 6 & 7 & 8 & 1 & 2 & 3 & 4 \\ 6 & 5 & 8 & 7 & 2 & 1 & 4 & 3 \\ 7 & 8 & 5 & 6 & 3 & 4 & 1 & 2 \\ 8 & 7 & 6 & 5 & 4 & 3 & 2 & 1 \end{array}$$

Кожна матриця має симетрію відносно діагоналей, які представлені мінімальними і максимальними числами. Існує також вертикальна і горизонтальна симетрія, де симетричні пари чисел перебувають у відношенні $N + \bar{N} = 2^n + 1$. Якщо нумерувати координати стовпців матриці, починаючи з нуля, то попереднє відношення матиме вигляд $N + \bar{N} = 2^n$. Квадратична за розміром матриця чисел в своєму синтезі має наступні закономірності: перший стовпець є вектором упорядкованих за зростанням чисел; другий стовпець представляє собою перестановку двох сусідніх чисел в кожній парі щодо попереднього стовпчика; третій створює перестановку сусідніх пар чисел; четвертий стовпець визначається перестановкою сусідніх тетрад чисел, що йдуть у зворотному порядку. Інші стовпці для функції від 3-х змінних гене-

руються як дзеркальні відображення синтезу чотирьох попередніх стовпців, починаючи з останнього стовпчика, який являє собою перший стовець, занумерований у зворотному порядку.

Синтез матриці перестановки бітів заснований на всіх видах симетрії (вертикальна, горизонтальна, діагональна), яка дає можливість, маючи один відомий квадрант отримати всі інші шляхом застосування формули:

$$N + \bar{N} = 2^n + 1 \rightarrow \bar{N} = (2^n + 1) - N.$$

Наступна структура матриць пояснює тривіальність побудови як завгодно складної матриці перестановки бітів від $n = 1, 2, 3, \dots$ змінних:

$$\begin{bmatrix} N & \bar{N} \\ \bar{N} & N \end{bmatrix} \rightarrow \begin{bmatrix} N & \bar{N} \\ \bar{N} & N \end{bmatrix} \begin{bmatrix} N & \bar{N} \\ \bar{N} & N \end{bmatrix} \rightarrow \begin{bmatrix} N & \bar{N} \\ \bar{N} & N \end{bmatrix} \begin{bmatrix} N & \bar{N} \\ \bar{N} & N \end{bmatrix} \begin{bmatrix} N & \bar{N} \\ \bar{N} & N \end{bmatrix} \begin{bmatrix} N & \bar{N} \\ \bar{N} & N \end{bmatrix}$$

Фактично, знаючи будь-який квадрант від $n-1$ змінної за 4 автоматних такти, можна отримати матрицю перестановки бітів для дедуктивної функціональності від n змінних: $N^1 \in N^2 \in N^3 \in \dots \in N^n$. Масштабованість квадрантів можна продемонструвати наступною структурою, яка інваріанта до кількості змінних аналізованої логічної функціональності:

$$\begin{bmatrix} N^1 \\ N & \bar{N} \\ \bar{N} & N \end{bmatrix} = N \in N^2 \rightarrow \begin{bmatrix} N^2 \\ N & \bar{N} \\ \bar{N} & N \end{bmatrix} = N \in N^3 \rightarrow \begin{bmatrix} N^3 \\ N & \bar{N} \\ \bar{N} & N \end{bmatrix} \dots = N \in N^n \rightarrow \begin{bmatrix} N^n \\ N & \bar{N} \\ \bar{N} & N \end{bmatrix}.$$

Прикладом рекурсивної генерації матриць перестановки бітів для змінних $n = 1, 2, 3$ може служити наступна структура:

$$H_{ij}(n=1,2,3) = \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 3 & 4 \\ 4 & 3 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 3 & 4 \\ 4 & 3 \end{bmatrix} \begin{bmatrix} 5 & 6 \\ 6 & 5 \end{bmatrix} \begin{bmatrix} 7 & 8 \\ 8 & 7 \end{bmatrix}$$

Рекурсивна взаємодія матриць перестановки бітів показує: для отримання як завгодно складної матриці від n змінних необхідно знати тільки один елемент (в даному випадку одиницю) в матриці для функції від однієї змінної.

Якщо першим елементом нумерації вхідних двійково-десяткових кодів (початкова адреса) прийняти нуль, то взаємодія матриць перестановки бітів матиме такий вигляд:

$$H_{ij}(n=1,2,3) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 2 & 3 \\ 3 & 2 \end{bmatrix} \rightarrow \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 2 & 3 \\ 3 & 2 \end{bmatrix} \begin{bmatrix} 4 & 5 \\ 5 & 4 \end{bmatrix} \begin{bmatrix} 6 & 7 \\ 7 & 6 \end{bmatrix}$$

З наведеної структури, що ілюструє синтез матриць перестановки бітів для отримання кубітних покриттів дедуктивного аналізу, впливають такі корисні властивості, що характеризують метод:

1) Матриця будь-якого рівня ієрархії містить 4 квадранта, які діагонально рівні один одному (рис. 4.5):

$$H = \begin{bmatrix} H_1 & H_2 \\ H_3 & H_4 \end{bmatrix} = \begin{bmatrix} H & \bar{H} \\ \bar{H} & H \end{bmatrix}, \quad H_1 = H_4; \quad H_1 = \bar{H}_2; \\ H_2 = H_3; \quad H_3 = \bar{H}_4.$$

Q_{and}	L_1	L_2	L_3	L_4
0	0	0	0	1
0	0	0	0	1
0	0	0	0	1
1	1	1	1	0

 \rightarrow

1	2	3	4
2	1	4	3
3	4	1	2
4	3	2	1

 \rightarrow

X_1X_2	L_1	L_2	L_3	L_4
00	0	0	0	0
01	0	0	1	1
10	0	1	0	1
11	1	0	0	1

 \rightarrow

$$\rightarrow L = (00)(X_1X_2) \vee (01)(X_1\bar{X}_2) \vee (10)(\bar{X}_1X_2) \vee (11)(X_1 \vee X_2).$$

Q_{or}	L_1	L_2	L_3	L_4
0	0	1	1	1
1	1	0	0	0
1	1	0	0	0
1	1	0	0	0

 \rightarrow

1	2	3	4
2	1	4	3
3	4	1	2
4	3	2	1

 \rightarrow

X_1X_2	L_1	L_2	L_3	L_4
00	0	0	0	0
01	1	1	0	0
10	1	0	1	0
11	1	0	0	1

 \rightarrow

$$\rightarrow L = (00)(X_1 \vee X_2) \vee (01)(\bar{X}_1X_2) \vee (10)(X_1\bar{X}_2) \vee (11)(X_1X_2).$$

Q_{xor}	L_1	L_2	L_3	L_4
0	0	1	1	0
1	1	0	0	1
1	1	0	0	1
0	0	1	1	0

 \rightarrow

1	2	3	4
2	1	4	3
3	4	1	2
4	3	2	1

 \rightarrow

X_1X_2	L_1	L_2	L_3	L_4
00	0	0	0	0
01	1	1	1	1
10	1	1	1	1
11	0	0	0	0

 \rightarrow

$$\rightarrow L = (00 \vee 01 \vee 10 \vee 11)(\bar{X}_1X_2 \vee X_1\bar{X}_2) = (xx)(\bar{X}_1X_2 \vee X_1\bar{X}_2).$$

Q_{nxr}	L_1	L_2	L_3	L_4
1	0	1	1	0
0	1	0	0	1
0	1	0	0	1
1	0	1	1	0

 \rightarrow

1	2	3	4
2	1	4	3
3	4	1	2
4	3	2	1

 \rightarrow

X_1X_2	L_1	L_2	L_3	L_4
00	0	0	0	0
01	1	1	1	1
10	1	1	1	1
11	0	0	0	0

 \rightarrow

$$\rightarrow L = (00 \vee 01 \vee 10 \vee 11)(\bar{X}_1X_2 \vee X_1\bar{X}_2) = (xx)(\bar{X}_1X_2 \vee X_1\bar{X}_2).$$

Q_{not}	L_1	L_2
1	0	1
0	1	0

 \rightarrow

1	2
2	1

 \rightarrow

X	L_1	L_2
0	0	0
1	1	1

 \rightarrow

$$\rightarrow L = (0 \vee 1)(X \vee X) = (xx)(X).$$

Q_{rep}	L_1	L_2
0	0	1
1	1	0

 \rightarrow

1	2
2	1

 \rightarrow

X	L_1	L_2
0	0	0
1	1	1

 \rightarrow

$$\rightarrow L = (0 \vee 1)(X \vee X) = (xx)(X).$$

Рисунок 4.5 – Структурна ієрархія матриць для отримання кубітних покриттів

2) Горизонтальні і вертикальні відносини між квадрантами зображених матриць визначаються доповненнями (при нульовому початковому елементі або адресі):

$$\begin{aligned}
H_1 + \bar{H}_2 &= 2^n - 1 \rightarrow \bar{H}_2 = (2^n - 1) - H_1; \\
\bar{H}_3 + H_4 &= 2^n - 1 \rightarrow \bar{H}_3 = (2^n - 1) - H_4; \\
H_1 + \bar{H}_3 &= 2^n - 1 \rightarrow \bar{H}_3 = (2^n - 1) - H_1; \\
\bar{H}_2 + H_4 &= 2^n - 1 \rightarrow \bar{H}_2 = (2^n - 1) - H_4.
\end{aligned}$$

Сума двох чисел, симетричних відносно горизонтальної або вертикальної осі, що ділить матрицю від n змінних на дві однакові частини, дорівнює $h + \bar{h} = 2^n - 1$. Четвірку компонентів примітивної матриці, яка не містить інших матриць як складових частин, завжди формують два сусідніх числа або цифри.

3) Рекурсивна формула отримання матриці перестановки бітів для синтезу дедуктивної функції від $i = 1, 2, 3, \dots$ змінних має такий вигляд:

$$\begin{aligned}
H^i &= \begin{bmatrix} H^i & \bar{H}^i \\ \bar{H}^i & H^i \end{bmatrix}, H^1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}; \\
H_1^i &= \begin{bmatrix} H^{i-1} & \bar{H}^{i-1} \\ \bar{H}^{i-1} & H^{i-1} \end{bmatrix}; \\
H_4^i = H_1^i &= \begin{bmatrix} H^{i-1} & \bar{H}^{i-1} \\ \bar{H}^{i-1} & H^{i-1} \end{bmatrix}; \\
H_2^i = \bar{H}_1^i &= (2^n - 1) - \begin{bmatrix} H^{i-1} & \bar{H}^{i-1} \\ \bar{H}^{i-1} & H^{i-1} \end{bmatrix}; \\
H_3^i = \bar{H}_2^i &= (2^n - 1) - \begin{bmatrix} H^{i-1} & \bar{H}^{i-1} \\ \bar{H}^{i-1} & H^{i-1} \end{bmatrix}.
\end{aligned}$$

На кожному кроці рекурсії формуються завжди чотири компонента, де два з них, розташованих по діагоналі, переносяться з матриці попереднього кроку, а два інших генеруються шляхом доповнення кожного нового елемента в квадраті до відповідних, щодо вертикальної або горизонтальної осі симетрії, бітів перенесених діагональних компонентів:

$$\bar{H}^i = (2^n - 1) - H^i.$$

Слід нагадати, що синтезовані стовпці матриці використовуються для остаточного формування кубітного покриття дедуктивної функції шляхом

перестановки бітів, отриманого для транспортування вхідних векторів (списків) несправностей на зовнішній вихід функціональності при заданому вхідному тестовому наборі.

Таким чином, запропонований метод синтезу дедуктивних кубітних покриттів для моделювання несправностей відрізняється від відомих в світі аналогів оригінальністю математичних рішень, високим рівнем паралелізму і компактністю структур даних, що дає можливість використовувати його програмну (апаратну) реалізацію для синтезу, аналізу, тестування, верифікації та діагностування цифрових систем на кристалах. Висока швидкодія синтезу дедуктивних кубітних покриттів на заданому вхідному наборі, яка визначається бітовими операціями $Q = 2 \times 2^n = 2^{n+1}$, є підставою для його використання з метою тестування, моделювання і діагностування цифрових систем в режимі online. Оскільки регістрова реалізація фактично двох операцій: хогпорівняння і перестановки бітів, дозволить звести згадану оцінку до двох (кількох) автоматних тактів. Крім того, дві згадані операції можна об'єднати в одну процедуру, виконувану в автоматному такті.

4.5 Імплементация кубітних моделей і методів їх аналізу

Software додаток QuaSim (Quantum Simulation) являє собою серверний додаток [10], що розвивається, з нарощування паралельних алгоритмів, орієнтований на моделювання дискретних процесів і явищ різної природи, включаючи цифрові логічні схеми і компоненти. Опис кубітного методу (алгоритму) моделювання супроводжується прикладом схеми і структур даних, представлених на рис. 4.6.

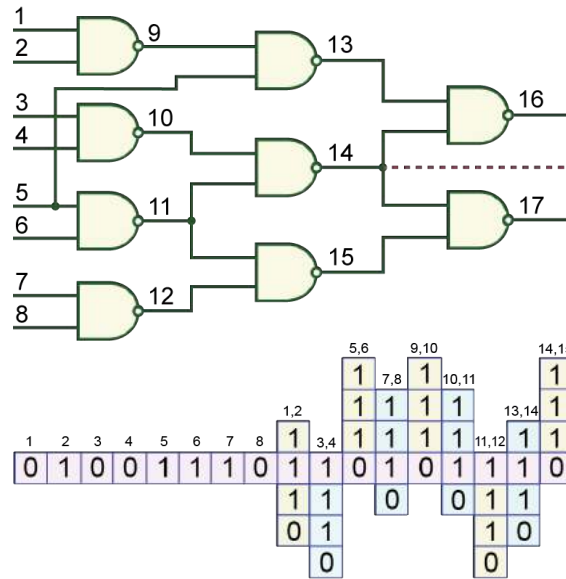


Рисунок 4.6 – Логічна схема і кубітна структура

Основна відмінність продукту QuaSim від існуючих симуляторів полягає в використанні інтерпретативних паралельних алгоритмів моделювання, що використовують транзакційне (read-write) характеристичне рівняння $M(i)=Q[I, M(X(i))]$, квантового комп'ютингу на кубітних структурах даних. Останні представляють собою матрицю $Q[I, M(X(i))]$, векторів-стовпців різної розмірності, актуальна довжина яких прив'язана до ступеневої функції від числа входів логічного елемента $q=2^{*n}$. При цьому впорядковані номери входних ліній $X(i) = (3,4), (5,6)$, створюють кортеж координат вектора моделювання $(5,6)$, зчитування і конкатенація яких формує двійковий вектор адрес $M(X(i)) = (11)$, але вже координати (3) кубітного покриття, зчитування значення $Q[I, M(X(i))] = (0)$ з якої супроводжується одночасним записом (0) в вектор моделювання по координаті (11), що дорівнює номеру вектор-стовпця (11) матриці кубітних покриттів $M(i) = Q[I, M(X(i))] = 0$. Паралелізм виконання алгоритму моделювання пов'язаний з ранжируванням логічних елементів по шарах структурної глибини, що дає можливість, в даному прикладі, обробляти одночасно наступні групи кубітних покриттів: 1) $\{9,10,11,12\}$; 2) $\{13,14,15\}$; $\{16,17\}$. Таким чином, замість 9 циклів моделю-

вання паралельний алгоритм виконає транзакційну обробку всієї схеми за 3 ітерації, що означає збільшення продуктивності обчислювального процесу в три рази.

Аналітична модель процесу моделювання на основі використання кубітних структур даних для синтезу та аналізу цифрових систем і компонентів має наступний вигляд:

$$\begin{aligned} W &= (M, Q, X), \\ M &= (M_1, M_2, \dots, M_i, \dots, M_n); \\ Q &= [Q_{ij}]; \\ Q &= (Q_1, Q_2, \dots, Q_i, \dots, Q_n); \\ Q_i &= (Q_{i1}, Q_{i2}, \dots, Q_{ij}, \dots, Q_{im}); \\ M(i) &= Q[i, M(X_i)]. \end{aligned}$$

Модель інтегрує наступні системоутворюючі компоненти: M – вектор моделювання логічної схеми, який пов'язує всі кубітні покриття примітивних елементів в структуру; X – вектор десяткових номерів вхідних змінних; Q – матриця кубітних покриттів, представлених у вигляді вектор-стовпців вихідних станів логічних елементів. Аналітичне рівняння адресних транзакцій у вигляді запису-зчитування даних між вектором моделювання і кубітними покриттями створює простий і ефективний алгоритм моделювання будь-яких дискретних обчислювальних процесів:

$$M(i) = Q [I, M(X(i))].$$

Аналітичне рівняння моделювання безпосередньо впливає на продуктивність квантового методу аналізу, яка залежить від операцій конкатенації k , зчитування r і запису w бітів, кількості q кубітних покриттів в цифровій схемі, а також довжини тестової послідовності вхідних наборів t :

$$Q = (k + r + w) \times q \times t.$$

Фактично необхідно виконати три транзакційних операції read-write для моделювання функціонального елемента будь-якої складності на основі тех-

нології memory-driven квантового паралельного комп'ютингу для вирішення завдань синтезу та аналізу.

Переваги memory-address-transaction [10] computing полягають в наступному: 1) Виключення з обчислювального процесу блоку АЛП, який знижує продуктивність обробки даних за рахунок прийому-передачі даних по шині обміну між процесором і пам'яттю. 2) Виключення традиційної логіки їх обчислювального процесу яка замінюється операціями зчитування-запису на елементах пам'яті. 3) Паралелізм обробки великих даних на SIMD-архітектурах пам'яті. 4) Істотне зниження енергоспоживання і підвищення продуктивності за рахунок відсутності значних відстаней між місцем зберігання даних і пристроєм для їх обробки – обчислювати за місцем наявності даних. 5) Становлення уніфікованої однорідності архітектур і технологічної простоти обчислювальних процесів, де немає нічого, крім взаємопов'язаних структур пам'яті, що реалізують зберігання, обчислення і управління. 6) Інтерпретативність методу моделювання, заснована на використанні кубітних структур даних, робить його привабливим для виконання наукових досліджень, де необхідно вносити зміни в гнучку структуру даних, які створюють модель цифрового виробу, чого не можна зробити в компілятивних програмах логічного синтезу та аналізу. 7) Оперативне створення в конкретному місці пам'яті (кіберпростору) memory-driven архітектури (управління і обчислення) комп'ютингу для локальної обробки великих даних.

Недоліки memory-address-transaction (quantum) computing представлені такими пунктами: 1) Небажання провідних компаній планети відмовитися від використання процесингової логіки, оскільки така акція супроводжується мільярдними витратами на побудову нових фабрик. Це може статися при виході на ринок недорогих квантових обчислювачів, вільних від традиційної логіки. 2) Порівняно невисока швидкодія транзакцій на пам'яті, яка в кілька разів нижче продуктивності виконання операцій на логічному процесорі. Проте, даний фактор може бути повністю виключений у разі появи квантово-

го комп'ютингу на субатомних частинках, де транзакції на пам'яті виконуються на максимально можливій швидкості світла фотонного обміну даними між електронами, як елементами для зберігання бітів. 3) Необхідність суттєвих інвестицій для розробки теоретичних основ і створення засобів автоматизованого проектування цифрових обчислювачів на парадигмі quantum memory-driven computing. 4) Існує певне незрозуміння необхідності створення та впровадження основ quantum memory-driven computing в освітню практику ВНЗ. 5) Непідготовленість існуючих фахівців в області проектування для створення і застосування паралельних алгоритмів, поки на класичних комп'ютерах, в практику обчислювальних процесів кіберпростору, орієнтованих на квантові цифрові пристрої.

Серверний додаток QuaSim пов'язаний з популяризацією нових технологій квантово-кубітного синтезу та аналізу цифрових пристроїв в середовищі студентів, вчених і фахівців, завдяки згаданим вище перевагам quantum memory-driven computing і має наступні функціональності (рис. 4.7):

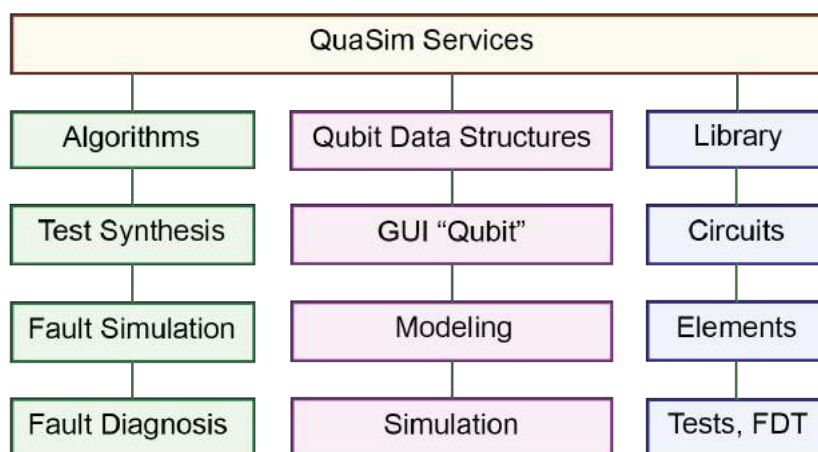


Рисунок 4.7 – Сервіси моделювання QuaSim

1) Modeling – створення кубітної моделі цифрового або іншого дискретно функціонуючого пристрою шляхом введення логічних елементів, вхідних і вихідних інтерфейсів, встановлення (зображення) зв'язків між ними, а також введення кубітних покриттів всередину кожного примітиву.

2) Simulation – моделювання цифрових схем на синтезованих тестових наборах, на таблиці істинності цифрової схеми або в покроковому режимі, коли на входи подається окремий тестовий двійковий набір.

3) Test Synthesis – виконується кубітний метод генерації тестів, який використовує булеві кубітні похідні.

4) Fault Simulation – здійснюється моделювання одиночних константних несправностей методом дедуктивного кубітного аналізу з метою визначення якості покриття перевіряючим тестом заданого класу дефектів.

5) Fault Diagnosis – пошук несправностей на основі використання кубітних структур даних для унітарного кодування станів таблиці перевірюваних дефектів, що дає можливість за три паралельних операції ідентифікувати можливі несправності.

6) Modeling and Simulation соціальних процесів і явищ шляхом синтезу кубітних покриттів, які відповідають даним, отриманим в результаті вичерпного моніторингу.

Програмні модулі QuaSim сервісів закодовані на мові Swift, операційна система OSX 10.9, компілятор XCode 7. Кількість вихідних файлів 55, загальне число рядків коду – 2900.

Верифікація QuaSim виконана на логічних схемах з бібліотеки ISCAS 89: 1) C5 sample. 2) C17 comb. 3) PatternB. 4) Hasse proc. 5) Sec. Circuit. 6) MUX8-1 Circuit. 7) Des Circuit. 8) Cov. processor. 9) RF4-16 Circuit. 10) C432 sample. Порівнювався час: введення схеми (Modeling Time), синтезу та аналізу якості тестів. Базовий засіб – продукт Active HDL, Aldec Inc., де моделі схеми вводиться на VHDL мові. Статистика порівняння Modeling, Synthesis, and Simulation Time для X-функцій представлена на рис. 4.8.

Аналіз показує певні переваги використання QuaSim для проектування логічних схем невеликої розмірності в порівнянні з синтезом HDL-опису. Природно, що методи квантового аналізу і синтезу логічних схем на кубітних покриттях X-функцій виграють перед класичними методами синтезу тестів і

моделювання несправностей в класі інтерпретативних моделей. Серверний додаток QuaSim застосовується в процесі навчання студентів технологіям проектування і верифікації цифрових проектів на основі кубітних структур даних і паралельних алгоритмів логічного синтезу та аналізу.

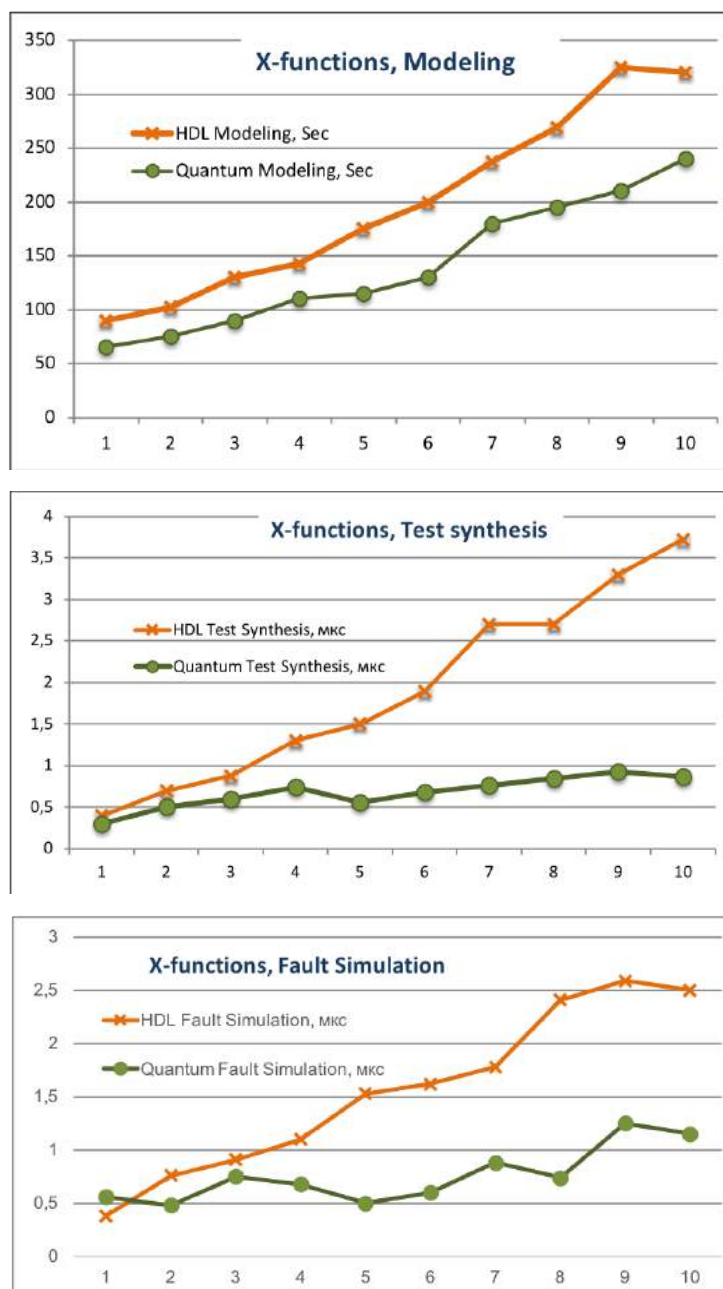


Рисунок 4.8 – Аналіз часу Modeling, Synthesis, and Simulation

4.6. Висновки до розділу 4

1) Запропоновано аналітичний вираз для синтезу кубітних покриттів Х-функцій від кінцевого числа змінних, що дає можливість створювати тестопридатні логічні схеми, які не потребують експоненційних витрат на синтез і аналіз тестів перевірки та діагностування несправностей.

2) Отримано дедуктивні формули транспортування вхідних списків несправностей на зовнішні виходи для Х-функцій від кінцевого числа змінних, які характеризуються поодинокими векторами похідних за всіма змінними, що дає можливість побудувати секвенсор моделювання дефектів, інваріантний до вхідних тестових наборів.

3) Запропоновано метод синтезу дедуктивних кубітних покриттів для моделювання несправностей на основі використання Q-покриттів функціональностей, який відрізняється від відомих аналогів оригінальністю математичних рішень, високим рівнем паралелізму і компактністю структур даних, що дає можливість використовувати його програмну (апаратну) реалізацію для синтезу, аналізу, тестування, верифікації та діагностування цифрових систем на кристалах.

4) Імплементувати кубітні моделі, методи і алгоритми в практику програмного серверного додатка QuaSim, що має на меті вирішення завдань синтезу кубітних моделей, генерування перевіряючих тестів і оцінки якості їх покриття для класу одиночних константних несправностей. Наведено експерименти над програмним продуктом в частині створення діагностичного забезпечення для десяти цифрових схем.

Результати розділу відображено у публікаціях [1, 4, 6, 8-13, 16, 17, 19, 20, 22].

ВИСНОВОК

Дисертаційна робота вирішує науково-практичну задачу верифікації спеціалізованих цифрових систем на кристалах шляхом використання memory-driven архітектур і кубітних структур даних для компактного опису логічних X -функцій і істотного підвищення продуктивності методів синтезу тестів і дедуктивного моделювання несправностей за рахунок паралельного комп'ютингу алгоритмів.

Сутність дослідження – квантові моделі і методи синтезу та аналізу логічних X -функцій на основі використання кубітних структур даних і memory-driven архітектур для паралельного комп'ютингу алгоритмів, з метою істотного підвищення продуктивності тестування і дедуктивного моделювання цифрових систем на кристалах.

Мета дослідження – зменшення часу верифікації цифрових систем на кристалах шляхом використання memory-driven архітектур і кубітних структур даних для компактного опису логічних X -функцій і істотного підвищення продуктивності методів тестування і дедуктивного моделювання несправностей за рахунок паралельного комп'ютингу алгоритмів.

Інтегрально розроблені квантові моделі і методи синтезу та аналізу логічних (X)-функцій на основі використання векторних кубітних структур даних і memory-driven архітектур для паралельного комп'ютингу алгоритмів, з метою істотного підвищення продуктивності тестування і дедуктивного моделювання цифрових систем на кристалах. Визначено структуру метричних властивостей X -функцій від кінцевого числа змінних, яка дає можливість істотно зменшити час синтезу тестів і моделювання несправностей по кубітним покриттям. Запропоновано новий квантовий метод генерації перевіряючих тестів для одиночних константних несправностей логічних функцій шляхом взяття булевих похідних по їх кубітним покриттям. Розроблено кванто-

вий метод дедуктивного моделювання константних несправностей логічних функцій шляхом синтезу матриць булевих похідних по їх кубітним покриттям. Швидкодія методів синтезу тестів і їх аналізу не має аналогів по алгоритмічній продуктивності.

Наукова новизна:

1) Вперше представлена структурна модель метричних властивостей Х-функцій, орієнтована на виконання паралельних операцій на кубітних структурах даних з метою отримання лінійного часу генерації тестів і моделювання цифрових систем.

2) Вперше створена аналітична модель синтезу кубітних покриттів Х-функцій від кінцевого числа змінних, яка характеризується можливістю створення логічних схем, які не потребують експоненційних витрат на генерування і аналіз тестів перевірки несправностей.

3) Вперше розроблено паралельний метод синтезу тестів для несправностей Х-функцій від кінцевого числа змінних, який характеризується взяттям булевих похідних по кубітним покриттям, що дає можливість отримувати перевіряючі тести мінімальної довжини.

4) Вперше запропоновано паралельний метод синтезу дедуктивних кубітних покриттів для моделювання Х-функцій, який характеризується отриманням одиначної матриці похідних, що дає можливість створювати сенсор моделювання дефектів, інваріантний до вхідних тестових наборів.

5) Удосконалено memory-driven архітектури та алгоритми для реалізації методів тестування та верифікації цифрових систем на кристалах, які відрізняються паралельним виконанням логічних операцій над кубітними структурами даних.

6) Удосконалено квантові методи генерації тестів і дедуктивного моделювання несправностей логічних функцій, які відрізняються від аналогів синтезом матриць булевих похідних по їх кубітним покриттям.

Практична значимість отриманих результатів. Розроблено моделі, структури даних, методи синтезу та аналізу логічних схем, включаючи Х-функції, які дають можливість суттєво зменшити час синтезу тестів і моделювання несправностей, завдяки кубітному опису цифрових схем, що дає можливість паралельно виконувати обчислювальні процедури, характерні квантовому комп'ютингу. Окремі сервіси синтезу та аналізу кубітних моделей цифрових пристроїв і компонентів реалізовані у вигляді програмних додатків і пройшли вичерпну апробацію в навчальному процесі при вивченні курсів «Квантові обчислення», «Основи комп'ютерної діагностики», «System on Chip». Середовище проектування: SWIFT, C ++, Verilog, Java і платформи: Microsoft Windows, X Window і Macintosh OS X.

Результати дослідження в складі моделей, методів і фрагментів додатків впроваджені в навчальний процес Харківського національного університету радіоелектроніки (акт про впровадження від 23.10.2018); в методологічне та технологічне забезпечення компанії «DIT Delight Limited» (довідка про впровадження від 25.10.2018).

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. 5 Trends Emerge in the Gartner Hype Cycle for Emerging Technologies, 2018. URL: <https://www.gartner.com/smarterwithgartner/5-trends-emerge-in-gartner-hype-cycle-for-emerging-technologies-2018/> (Дата звернення: 30.10.2018).
2. Kharchenko V. Green IT Engineering: Concepts, Models, Complex Systems Architectures: In the book series "Studies in Systems, Decision and Control" (SSDC) / V. Kharchenko, Y. Kondratenko, J. Касprzyk (Eds.). Vol. 1. Berlin, Heidelberg: Springer International Publishing, 2017.
3. Kharchenko V. Green IT Engineering: Components, Networks and Systems Implementation: In the book series "Studies in Systems, Decision and Control" (SSDC) / V. Kharchenko, Y. Kondratenko, J. Касprzyk (Eds.). Vol. 2. Berlin, Heidelberg: Springer International Publishing, 2017.
4. Gupta A. A Survey of 5G Network: Architecture and Emerging Technologies / A. Gupta, R. K. Jha. In IEEE Access. 2015. Vol. 3. P. 1206-1232.
5. Top 10 Strategic Technology Trends for 2019. URL: https://www.gartner.com/doc/3891569?src_Id=1-7251599992&cm_sp=swg_-_gi-_dynamic (Дата звернення: 30.10.2018).
6. Gartner Identifies Three Megatrends That Will Drive Digital Business Into the Next Decade. URL: <http://www.gartner.com/newsroom/id/3784363> (Дата звернення: 30.10.2018).
7. Christidis K. Blockchains and Smart Contracts for the Internet of Things / K. Christidis, M. Devetsikiotis. In IEEE Access. 2016. Vol. 4. P. 2292-2303.
8. Blockchains: How They Work and Why They'll Change the World IEEE Spectrum. October 2017. URL: <https://spectrum.ieee.org/computing/networks/blockchains-how-they-work-and-why-theyll-change-the-world> (Дата звернення: 30.10.2018).

9. Zanella A. A. Internet of Things for Smart Cities / A. A. Zanella, N. Bui, A. Castellani, L. Vangelista, M. Zorzi. In IEEE IoT Journal. Feb. 2014. Vol. 1, no. 1. P. 22-32.

10. Hahanov V. Cyber Physical Computing for IoT-driven Services. New York: Springer, 2018.

11. Hahanov V.I. Qubit technology for analysis and diagnosis of digital devices / V.I. Hahanov, T. Bani Amer, S.V. Chumachenko, E.I. Litvinova. In Electronic modeling. 2015. No 37 (3). P. 17-40.

12. Hahanov V. Quantum memory-driven computing for test synthesis / V. Hahanov, W. Gharibi, E. Litvinova, M. Liubarskyi, A. Hahanova. In IEEE East-West Design and Test Symposium. Novi Sad, Serbia. 2017. P. 123-128.

13. Zhong T. Nanophotonic rare-earth quantum memory with optically controlled retrieval / T. Zhong, J. M. Kindem, J. G. Bartholomew et al. In Science. 29 Sep 2017. Vol. 357, iss. 6358. P. 1392-1395.

14. Kim J. Photon-triggered nanowire transistors / J. Kim, H.-Ch. Lee, K.-H. Kim et al. In Nature Nanotechnology. 2017. No 12. P. 963–968.

15. Lovat G. Room-temperature current blockade in atomically defined single-cluster junctions / G. Lovat, B. Choi, D. W. Paley et al. In Nature Nanotechnology. Nov12(11). 2017. P. 1050-1054.

16. Li Ch. Conformation-based signal transfer and processing at the single-molecule level / Ch. Li, Zh. Wang, Y. Lu, X. Liu, L. Wang. In Nature Nanotechnology. 2017. Vol. 12. P. 1071–1076.

17. Quantum photonic researchers start new company, Sparrow Quantum. URL: <http://www.nbi.ku.dk/english/news/news16/quantum-photonic-researchers-start-new-company-sparrow-quantum/> (Дата звернення: 30.10.2018).

18. IonQ Raises \$20M Series B Round Led By NEA, GV To Advance Quantum Computing For Commercial Applications. URL: <https://www.prnewswire.com/news-releases/ionq-raises-20m-series-b-round-led->

by-nea-gv-to-advance-quantum-computing-for-commercial-applications-300494456.html (Дата звернення: 30.10.2018).

19. Hyman R. Weird quantum particles simulated in droplet of ultracold gas. URL: http://www.sciencemag.org/news/2018/03/weird-quantum-particles-simulated-droplet-ultracold-gas?utm_campaign=news_daily_2018-03-06&et rid=69734703&et cid=1891545 (Дата звернення: 30.10.2018).

20. Samkharadze N. Strong spin-photon coupling in silicon / N. Samkharadze, G. Zheng et al. In Science. 2018. Vol. 359, iss. 6380. P. 1123-1127.

21. Cho A. Vibrations used to talk to quantum circuits. In Science. 2018. Vol. 359, iss. 6381. P. 1202-1203.

22. P. Teich. Quantum Computing Enters 2018 Like it is 1968. URL: <https://www.nextplatform.com/2018/01/10/quantum-computing-enters-2018-like-1968/> (Дата звернення: 30.10.2018).

23. QISKit for quantum computation. 2018. URL: <https://www.ibm.com/blogs/research/2018/02/qiskit-index/> (Дата звернення: 30.10.2018).

24. QuTiP. Quantum Toolbox in Python. URL: <http://qutip.org> (Дата звернення: 30.10.2018).

25. Japan enters quantum computing race -- and offers free test drive. URL: <https://asia.nikkei.com/Tech-Science/Tech/Japan-enters-quantum-computing-race-and-offers-free-test-drive> (Дата звернення: 30.10.2018).

26. The Application of Spintronics. URL: <http://www-03.ibm.com/ibm/history/ibm100/us/en/icons/spintronics/> (Дата звернення: 30.10.2018).

27. IBM's world-class Summit supercomputer gooses speed with AI abilities. 2018. URL: <https://www.cnet.com/news/ibms-world-class-summit-supercomputer-gooses-speed-with-ai-abilities/> (Дата звернення: 30.10.2018).

28. Moore S. K. Self-Powered Image Sensor Could Watch You Forever. URL: <https://spectrum.ieee.org/tech->

talk/semiconductors/optoelectronics/selfpowered-image-sensor-could-watch-you-forever?utm_source=sensors&utm_campaign=sensors-04-17-18&utm_medium=email (Дата звернення: 30.10.2018).

29. Supercharging Chips by Integrating Optical Circuits. URL: https://spectrum.ieee.org/tech-talk/semiconductors/optoelectronics/optics-on-chips-could-speed-up-computing?utm_source=computingtechnology&utm_campaign=computingtechnology-05-01-18&utm_medium=email (Дата звернення: 30.10.2018).

30. Sun S. A single-photon switch and transistor enabled by a solid-state quantum memory / S. Sun, H. Kim, Z. Luo, G. S. Solomon, E. Waks. In Science. 06 Jul 2018. Vol. 361, issue 6397. P. 57-60. URL: <https://www.mram-info.com/stt-mram> (Дата звернення: 30.10.2018).

31. 10 Key Marketing Trends for 2017. URL: <https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=WRL12345USEN> (Дата звернення: 30.10.2018).

32. Zhu C. Green Internet of Things for Smart World / C. Zhu, V. C. M. Leung, L. Shu, E. C. H. Ngai. In IEEE Access. 2015. Vol. 3. P. 2151-2162.

33. Sebastian A. Temporal correlation detection using computational phase-change memory / A. Sebastian, T. Tuma, N. Papandreou, M. Le Gallo, L. Kull, T. Parnell, E. Eleftheriou. In Nature Communications. Vol. 8. Article number: 1115 (2017). URL: <https://www.nature.com/articles/s41467-017-01481-9> (Дата звернення: 30.10.2018).

34. Iftikhar Z. Tunable quantum criticality and super-ballistic transport in a "charge" Kondo circuit / Z. Iftikhar, A. Anthore, A. K. Mitchell, F. D. Parmentier, U. Gennser, A. Ouerghi, A. Cavanna, C. Mora, P. Simon and F. Pierre. In Science. 2018. No 360 (6395). P. 1315-1320.

35. M. Mohseni, P. Read, H. Neven, S. Boixo, V. Denchev, R. Babbush, A. Fowler, V. Smelyanskiy, J. Martinis. Commercialize quantum technologies in five

years. URL: <https://www.nature.com/news/commercialize-quantum-technologies-in-five-years-1.21583> (Дата звернення: 30.10.2018).

36. Квантовые вычисления: отжиг с выключателями и прочее веселье. URL: <https://habrahabr.ru/company/sberbank/blog/344830/> (Дата звернення: 30.10.2018).

37. Lipton R.J. Quantum Algorithms via Linear Algebra / R. J. Lipton, K. W. Regan. MIT Press eBook, 2014.

38. Hahanov V.I. Qubit data structure of computing devices / V.I. Hahanov, W. Gharibi, E.I. Litvinova, A.S. Shkil // Electronic modeling. 2015. Vol. 37, no 1. P. 76-99.

39. Hahanov V.I. Qubit technology analysis and diagnosis of digital devices / V.I. Hahanov, Tamer Bani Amer, S.V. Chumachenko, E.I. Litvinova // Electronic modeling. 2015. Vol. 37, no 3. P. 17-40.

40. Хаханов В.И. Квантовый метод синтеза тестов на основе кубитных структур данных / В.И. Хаханов, И.В. Емельянов, М.М. Любарский, С.В. Чумаченко, Е.И. Литвинова // Электронное моделирование. 2018. Том 40, № 1. С. 63-80.

41. Abramovici M. Digital System Testing and Testable Design. Comp. Sc. Press, 1998.

42. Hahanov V. Infrastructure intellectual property for SoC simulation and diagnosis service. In the book Design of Digital Systems and Devices / Editors: Adamski M., Barkalov A., Wegrzyn M. (Eds.). 2011. Springer. P. 289-330.

43. Drozd J. Models, Methods and Means as Resources for Solving Challenges in Co-Design and Testing of Computer Systems and their Components / J. Drozd, A. Drozd. In Proc. of the 9th International Conference on Digital Technologies 2013. Zhilina, Slovak Republic. 2013. P. 225-230.

44. J. Drozd, A. Drozd, S. Antoshchuk, "Green IT engineering in the view of resource-based approach: In the book series "Green IT Engineering: Concepts, Models, Complex Systems Architectures, Studies in Systems, Decision and Con-

troл” / V. Kharchenko, Y. Kondratenko, J. Kasprzyk (Eds.). Vol. 74. Berlin, Heidelberg: Springer International Publishing, 2017, P. 43-65.

45. Fujiwara H. Fault Simulation: In the book “Logic Testing and Design for Testability”. MIT Press, 1985. P. 84-108.

46. Карибский В.В. и др. Основы технической диагностики. Кн. 1, М.: «Энергия», 1976.

47. Чжен Г. Диагностика отказов цифровых вычислительных систем / Г. Чжен, Е. Мэннинг, Г. Метц. М.: Мир, 1972.

48. Пархоменко П.П. Основы технической диагностики / под ред. П.П. Пархоменко. М.: Энергия, 1981.

49. Marinissen E.J. Guest Editors' Introduction: The Status of IEEE Std 1500. / E.J. Marinissen, Yervant Zorian. In IEEE Design & Test of Computers, 2009. No 26(1). P. 6-7.

50. Малышенко Ю.В. Автоматизация диагностирования электронных устройств / Ю.В. Малышенко и др. М.: Энергоатомиздат, 1986.

51. Надежность технических систем / Под ред. И.А. Ушакова. М., 1985.

52. Хаханов В.И. Проектирование и тестирование цифровых систем на кристаллах / В. И. Хаханов, Е. И. Литвинова, И. В. Хаханова, О. А. Гузь. Новое Слово, 2009.

53. Pomeranz I. Aliasing Computation Using Fault Simulation with Fault Dropping / I. Pomeranz, M. Reddy Sudhakar. In IEEE Transactions on Computers. 1995. P. 139-144.

54. Lee S.M. Static fault analysis for resilient System-on-Chip design / S. M. Lee, S. E. Lee. In 2015 International SoC Design Conference (ISOCC). Gyungju, 2015. P. 5-6.

55. Admane N.C. Fault tolerant system for FPGA using simulation-based fault injection technique / N.C. Admane, D.R. Rotake. In 2015 International Conference on Communications and Signal Processing (ICCSP). Melmaruvathur, 2015. P. 0855-0859.

56. Yıldız A. Fault emulation on heterogeneous architectures / A. Yıldız, C.C. Gürsoy, S. Gören. In 2017 International Conference on Computer Science and Engineering (UBMK). Antalya, Turkey, 2017. P. 905-910.
57. Luo T. DaDianNao: A Neural Network Supercomputer / T. Luo et al. In IEEE Transactions on Computers. Vol. 66, no. 1. 2017. P. 73-88.
58. Gonzales D. Cloud-Trust—a Security Assessment Model for Infrastructure as a Service (IaaS) Clouds / D. Gonzales, J. M. Kaplan, E. Saltzman, Z. Winkelman, D. Woods. In IEEE Transactions on Cloud Computing. 2017. Vol. 5, no. 3. P. 523-536.
59. Hosokawa T. A Diagnostic Fault Simulation Method for a Single Universal Logical Fault Model / T. Hosokawa, H. Takano, H. Yamazaki, K. Yamazaki. In 2017 IEEE 22nd Pacific Rim International Symposium on Dependable Computing (PRDC). Christchurch, 2017. P. 217-218.
60. Dhiliban C. Fault simulation and analysis of VLSI circuit using n-detect test sets / C. Dhiliban, S. Govindaraju. In 2016 Online Int. Conference on Green Engineering and Technologies (IC-GET). Coimbatore, 2016. P. 1-5.
61. Jinling D. A fault simulation method based on mutated truth table of logic gates / D. Jinling, X. Aiqiang. In 2016 International Conference on Integrated Circuits and Microsystems (ICICM). Chengdu, 2016. P. 28-32.
62. Hadjitheophanous S. Scalable parallel fault simulation for shared-memory multiprocessor systems / S. Hadjitheophanous, S. N. Neophytou, M. K. Michael. In 2016 IEEE 34th VLSI Test Symposium (VTS). Las Vegas, NV, 2016. P. 1-6.
63. Chiang K.Y. Fault Simulation and Test Pattern Generation for Cross-gate Defects in FinFET Circuits / K.Y. Chiang, Y.H. Ho, Y.W. Chen, C. S. Pan, J. C. M. Li. In 2015 IEEE 24th Asian Test Symposium (ATS). Mumbai, 2015. P. 181-186.

64. Wu X. Substation Grounding Studies with More Accurate Fault Simulation Strategy and SI/MI Method / X. Wu, V. Simha, Y. Xue, R. Wellman. In IEEE Transactions on Power Delivery. 2016. Vol. PP, no. 99. P. 1-1.
65. Ubar R. Combinational fault simulation in sequential circuits / R. Ubar, J. Kõusaar, M. Gorev, S. Devadze. In 2015 IEEE International Symposium on Circuits and Systems (ISCAS). Lisbon, 2015. P. 2876-2879.
66. Rivière L. A novel simulation approach for fault injection resistance evaluation on smart cards / L. Rivière, J. Bringer, T. H. Le, H. Chabanne. In 2015 IEEE Eighth International Conference on Software Testing, Verification and Validation Workshops (ICSTW). Graz, 2015. P. 1-8.
67. Mirkhani S. EAGLE: A regression model for fault coverage estimation using a simulation-based metric / S. Mirkhani, J. A. Abraham. In 2014 International Test Conference. Seattle, WA, 2014. P. 1-10.
68. Xu J. The Research of Memory Fault Simulation and Fault Injection Method for BIT Software Test / J. Xu, P. Xu. In 2012 Second International Conference on Instrumentation, Measurement, Computer, Communication and Control. Harbin, 2012. P. 718-722.
69. Kochte M.A. Efficient BDD-based Fault Simulation in Presence of Unknown Values / M. A. Kochte, S. Kundu, K. Miyase, X. Wen, H. J. Wunderlich. In 2011 Asian Test Symposium. New Delhi, 2011. P. 383-388.
70. Reinsalu U. Fast RTL Fault Simulation Using Decision Diagrams and Bitwise Set Operations / U. Reinsalu, J. Raik, R. Ubar, P. Ellervee. In 2011 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems. Vancouver, BC, 2011. P. 164-170.
71. Janning A. A Cost-Effective FPGA-based Fault Simulation Environment / A. Janning, J. Heyszl, F. Stumpf, G. Sigl. In 2011 Workshop on Fault Diagnosis and Tolerance in Cryptography. Nara, 2011, P. 21-31.
72. Goudarzi H. Design of a universal logic block for fault-tolerant realization of any logic operation in trappedion quantum circuits / H. Goudarzi, M.J.

Dousti, A. Shafaei et al. In *Quantum Information Processing Journal*. Springer US, 2014. Vol. 13, iss. 5. P. 1267–1299.

73. Daley A. J. Quantum computing and quantum simulation with group-II atoms / A. J. Daley. In *Quantum Information Processing Journal*. Dec. 10: 865, Springer US, 2011.

74. Hahanov V.I. Qubit technology for analysis and diagnosis of digital devices / V.I. Hahanov, T. Bani Amer, S.V. Chumachenko, E.I. Litvinova. In *Electronic modeling*. 2015. No 37 (3). P. 17-40.

75. Hahanov V. Quantum memory-driven computing for test synthesis / V. Hahanov, W. Gharibi, E. Litvinova, M. Liubarskyi, A. Hahanova // In *IEEE East-West Design and Test Symposium*. Novi Sad, Serbia, 2017. P. 123-128.

76. Hahanov V. Infrastructure intellectual property for SoC simulation and diagnosis service / V. Hahanov. In the book “*Design of Digital Systems and Devices*.” Springer, 2011. P. 289-330.

77. Abramovici M. *Digital System Testing and Testable Design* / M. Abramovici. Comp. Sc. Press, 1998.

78. Fujiwara H. *Fault Simulation* / H. Fujiwara. In the book “*Logic Testing and Design for Testability*”. MIT Press, 1985. P. 84-108.

79. Pomeranz I. *Aliasing Computation Using Fault Simulation with Fault Dropping* / I. Pomeranz, M. Reddy Sudhakar. In *IEEE Transactions on Computers*, 1995. P. 139-144.

80. Hahanov V. Infrastructure intellectual property for SoC simulation and diagnosis service. In the book *Design of Digital Systems and Devices* / Editors: Adamski M., Barkalov A., Wegrzyn M. (Eds.). 2011. Springer. P. 289-330.

81. Hahanov V. at all. *Cyber Physical Computing for IoT-driven Services* [Text] / V. Hahanov at all. – New York. USA. – Springer, 2018. – 279 p. [Part 4. Qubit Description of the Functions and Structures for Service Computing Synthesis [Text] / I. Hahanov, I. Iemelianov, M. Liubarskyi, V. Hahanov. – P. 71-93] (Indexed by Springer, Scopus).

82. Hahanov V. at all. Cyber Physical Computing for IoT-driven Services [Text] / V. Hahanov at all. – New York. USA. – Springer, 2018. – 279 p. [Part 5. Quantum Computing for Test Synthesis [Text] / V. Hahanov, Bani Amer T., I. Iemelianov, M. Liubarskyi. – P. 95-134] (Indexed by Springer, Scopus).

83. Hahanov V. at all. Cyber Physical Computing for IoT-driven Services [Text] / V. Hahanov at all. – New York. USA. – Springer, 2018. – 279 p. [Part 6. QuaSim – Cloud Service for Quantum Circuits Simulation [Text] / I. Hahanov, Bani Amer T., I. Iemelianov, M. Liubarskyi, V. Hahanov – P. 135-147] (Indexed by Springer, Scopus).

84. Hahanov V. at all. Cyber Physical Computing for IoT-driven Services [Text] / V. Hahanov at all. – New York. USA. – Springer, 2018. – 279 p. [Part 14. Cyber-Physical Technologies: Hype Cycle 2017 [Text] / V. Hahanov, Wajeb Ghari-bi, Ka Lok Man, I. Iemelianov, M. Liubarskyi, V. Abdullayev, E. Litvinova, S. Chumachenko – P. 259-272] (Indexed by Springer, Scopus).

85. Хаханов В.И. Синтез Q-тестов по кубитному описанию функциональностей [Text] / В.И. Хаханов, Т. Bani Amer, И.В. Емельянов, М.М. Любарский // Радиоэлектроника и информатика. – 2016. – № 2(72). – С. 38-48. (Входить до міжнародних наукометричних баз Index Copernicus, Google Scholar, OECSP, OAJI, Scholar Steer, SIS, Cyberleninka, CiteFactor, TIU Hannover, I2OR).

86. Хаханов В.И. Кубитный метод дедуктивного анализа неисправностей для логических схем [Text] / В.И. Хаханов, И.В. Емельянов, М.М. Любарский, С.В. Чумаченко, Е.И. Литвинова, Т. Бани Амер // Электронное моделирование. – 2017. – Том 39, № 6. – С. 59-92 [Входить до міжнародних наукометричних баз Cambridge Scientific Abstracts, Computer and Information Systems Abstracts, INIS Collection, Inspec, ВИНТИ РАН].

87. Емельянов И.В. Квантовые модели и облачные сервисы для анализа и диагностирования логических схем [Text] / И.В. Емельянов, М.М. Любарский, В.И. Хаханов // Радиоэлектроника и информатика. – 2017. – № 4. – С.30-47. (Входить до міжнародних наукометричних баз Index Copernicus, Google

Scholar, OECSP, OAJ, Scholar Steer, SIS, Cyberleninka, CiteFactor, TIU Hannover, I2OR)

88. Хаханов В.И. Квантовый метод синтеза тестов на основе кубитных структур данных [Text] / В.И. Хаханов, И.В. Емельянов, *М.М. Любарский*, С.В. Чумаченко, Е.И. Литвинова // Электронное моделирование. – 2018. – Том 40, № 1. – С. 63-80 (Входит до міжнародних наукометричних баз Cambridge Scientific Abstracts, Computer and Information Systems Abstracts, INIS Collection, Inspec, ВИНТИ РАН).

89. Хаханов В.И. Дедуктивный анализ и диагностирование логических Х-функций [Text] / Хаханов В.И., Емельянов И.В., *Любарский М.М.*, Чумаченко С.В., Литвинова Е.И., Ларченко Л.В. // Радиоэлектроника и информатика. – 2018. – № 1. – С. 55-64. (Входит до міжнародних наукометричних баз Index Copernicus, Google Scholar, OECSP, OAJ, Scholar Steer, SIS, Cyberleninka, CiteFactor, TIU Hannover, I2OR)

90. *Любарский М.М.* Синтез и анализ логических Х-функций [Text] / *М.М. Любарский*, В.Г. Абдуллаев, В.И. Хаханов, С.В. Чумаченко, Е.И. Литвинова, И.В. Хаханов // Радиоэлектроника и информатика. – 2018. – №2. – С. 18-28.

91. Hahanov V. Quantum Data Structures for SoC Component Testing [Text] / V. Hahanov, W. Gharibi, S. Chumachenko, E. Litvinova, I. Iemelianov, *M. Liubarskyi* // International Journal of Design, Analysis & Tools for Integrated Circuits & Systems.– Oct. 2017.– Vol. 6, iss. 1. – P. 23. (Входит до міжнародної наукометричної бази EBSCO Information Services).

92. Хаханов В.И. Gartner 2017 топ-технологии: их анализ и применение [Text] / В.И. Хаханов, А.С. Мищенко, И.В. Емельянов, *М.М. Любарский*, Т.И. Соклакова, В.Г. Абдулаев // Paradigmata poznání. Vědecko vydavatelské centrum «Sociosféra-CZ», s.r.o., Praha, Česká republika. – 2017. – №4. – P. 33-62. (The journal is indexed by Electronic Research Library, Russia; Research Bible, China;

Scientific Indexing Services, USA; Cite Factor, Canada; General Impact Factor, India; Scientific Journal Impact Factor, India; CrossRef, USA; ORCID, USA).

93. Bani Amer T. КОМПЬЮТИНГОВЫЕ МОДЕЛИ ОБЛАЧНЫХ СЕРВИСОВ [Text] / T. Bani Amer, В.И. Хаханов, И.В. Емельянов, М. Любарский // АСУ и приборы автоматики. – 2015. – Вып. 173. – С.48-57. (Входит до міжнародних наукометричних баз Google Scholar, Cyberleninka).

94. Hahanov V. Qubit description of the functions and structures for computing [Text] / V. Hahanov, E. Litvinova, S. Chumachenko, M. Liubarskyi // Proc. of the IEEE East-West Design & Test Symposium. – 2016. – Yerevan, Armenia. – 6 p. (Входит до міжнародних наукометричних баз Scopus, IEEE Xplore).

95. Hahanov V. Qubit test synthesis for the black box functionalities [Text] / V. Hahanov, E. Litvinova, S. Chumachenko, I. Iemelianov, M. Liubarskyi // Proc. of 5th Prague Embedded Systems Workshop.– June 29-30, 2017.– Roztoky u Prahy, Czech Republic.– P.45-51.

96. Hahanov I. Qubit deductive fault simulation of black box functionalities [Text] / I. Hahanov, A. Hahanova, V. Hahanov, E. Litvinova, S. Chumachenko, M. Liubarskyi // Proc. of 5th Prague Embedded Systems Workshop. June 29-30, 2017. Roztoky u Prahy, Czech Republic. P.83-91.

97. Hahanov V. Quantum Data Structures for SoC Component Testing [Електронний ресурс] / V. Hahanov, W. Gharibi, K. L. Man, S. Chumachenko, E. Litvinova, I. Iemelianov, M. Liubarskyi // Proc. of the International Conference on Recent Advancements in Computing, IoT and Computer Engineering Technology. – The Tamkang University. – Taipei, Taiwan. – October 23-25, 2017. – 1 p.

98. Hahanov V. Quantum memory-driven computing for test synthesis [Text] / V. Hahanov, W. Gharibi, E. Litvinova, M. Liubarskyi, A. Hahanova // Proc. of the IEEE East-West Design and Test Symposium. – Novi Sad. – October, 2017. – P. 6. (Входит до міжнародних наукометричних баз Scopus, IEEE Xplore).

99. Vladimir Hahanov. Qubit test synthesis of the functionality [Text] / V. Hahanov, Tamer Bani Amer, E. Litvinova, T. Soklakova, M. Liubarskyi, N.

Shavlak, K. Dziuba // Proc. of the 14th International Conference “The Experience of Designing and Application of CAD Systems in Microelectronics” (CADSM). – Lviv-Polyana, Ukraine. – 21-25 February, 2017. - P. 251 – 255. (Входить до міжнародних наукометричних баз Scopus, IEEE Xplore).

100. Hahanov V. Qubit Minimization of Boolean Functions [Text] / Vladimir Hahanov, Ka Lok Man, *Mykhailo Liubarskyi*, Ivan Hahanov, Svetlana Chumachenko // Proc. of the International MultiConference of Engineers and Computer Scientists. – Hong Kong. – March 14-16, 2018. – Vol II IMECS 2018. – Pp. 690-695.

101. Hahanov V. Quantum Deductive Fault Simulation [Text] / V. Hahanov, W. Gharibi, *M. Liubarskyi*, E. Litvinova, M. Gharibi, S. Chumachenko, I. Hahanov, and A. Hahanova // Proc. of Int'l Conf. Modeling, Sim. and Vis. Methods (MSV'18) in The 2018 World Congress in Computer Science, Computer Engineering, & Applied Computing (CSCE2018) . – Jul 30-Aug 02, 2018. – Las Vegas, USA. – P. 10-16.

102. *Mykhailo Liubarskyi*. IoT Computing Evolution [Text] / M. Liubarskyi, I. Hahanov // Матеріали XXI Міжнародного молодіжного форуму «Радіоелектроніка та молодь у XXI ст.». – 25-27 квітня 2017. – Ч. 5. – С.46-47.

ДОДАТОК А

СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА ЗА ТЕМОЮ ДИСЕРТАЦІЇ

Список публікацій здобувача, в яких опубліковані основні наукові результати дисертації:

1. Hahanov V. at all. Cyber Physical Computing for IoT-driven Services [Text] / V. Hahanov at all. – New York. USA. – Springer, 2018. – 279 p. [Part 4. Qubit Description of the Functions and Structures for Service Computing Synthesis [Text] / I. Hahanov, I. Iemelianov, *M. Liubarskyi*, V. Hahanov. – P. 71-93] (Indexed by Springer, Scopus).

2. Hahanov V. at all. Cyber Physical Computing for IoT-driven Services [Text] / V. Hahanov at all. – New York. USA. – Springer, 2018. – 279 p. [Part 5. Quantum Computing for Test Synthesis [Text] / V. Hahanov, Bani Amer T., I. Iemelianov, *M. Liubarskyi*. – P. 95-134] (Indexed by Springer, Scopus).

3. Hahanov V. at all. Cyber Physical Computing for IoT-driven Services [Text] / V. Hahanov at all. – New York. USA. – Springer, 2018. – 279 p. [Part 6. QuaSim – Cloud Service for Quantum Circuits Simulation [Text] / I. Hahanov, Bani Amer T., I. Iemelianov, *M. Liubarskyi*, V. Hahanov – P. 135-147] (Indexed by Springer, Scopus).

4. Hahanov V. at all. Cyber Physical Computing for IoT-driven Services [Text] / V. Hahanov at all. – New York. USA. – Springer, 2018. – 279 p. [Part 14. Cyber-Physical Technologies: Hype Cycle 2017 [Text] / V. Hahanov, Wajeb Ghari-bi, Ka Lok Man, I. Iemelianov, *M. Liubarskyi*, V. Abdullayev, E. Litvinova, S. Chumachenko – P. 259-272] (Indexed by Springer, Scopus).

5. Хаханов В.И. Синтез Q-тестов по кубитному описанию функциональностей [Text] / В.И. Хаханов, Т. Вани Амер, И.В. Емельянов, *М.М. Любарский* // Радиоэлектроника и информатика. – 2016. – № 2(72). – С. 38-48. (Входить до міжнародних наукометричних баз Index Copernicus, Google Scholar, OECSP, OAJI, Scholar Steer, SIS, Cyberleninka, CiteFactor, TIU Hannover, I2OR).

6. Хаханов В.И. Кубитный метод дедуктивного анализа неисправностей для логических схем [Text] / В.И. Хаханов, И.В. Емельянов, *М.М. Любарский*, С.В. Чумаченко, Е.И. Литвинова, Т. Бани Амер // Электронное моделирование. – 2017. – Том 39, № 6. – С. 59-92 [Входит до міжнародних наукометричних баз Cambridge Scientific Abstracts, Computer and Information Systems Abstracts, INIS Collection, Inspec, ВИНТИ РАН].

7. Емельянов И.В. Квантовые модели и облачные сервисы для анализа и диагностирования логических схем [Text] / И.В. Емельянов, *М.М. Любарский*, В.И. Хаханов // Радиоэлектроника и информатика. – 2017. – № 4. – С.30-47. (Входит до міжнародних наукометричних баз Index Copernicus, Google Scholar, OECSP, OAJI, Scholar Steer, SIS, Cyberleninka, CiteFactor, TIU Hannover, I2OR)

8. Хаханов В.И. Квантовый метод синтеза тестов на основе кубитных структур данных [Text] / В.И. Хаханов, И.В. Емельянов, *М.М. Любарский*, С.В. Чумаченко, Е.И. Литвинова // Электронное моделирование. – 2018. – Том 40, № 1. – С. 63-80 (Входит до міжнародних наукометричних баз Cambridge Scientific Abstracts, Computer and Information Systems Abstracts, INIS Collection, Inspec, ВИНТИ РАН).

9. Хаханов В.И. Дедуктивный анализ и диагностирование логических Х-функций [Text] / Хаханов В.И., Емельянов И.В., *Любарский М.М.*, Чумаченко С.В., Литвинова Е.И., Ларченко Л.В. // Радиоэлектроника и информатика. – 2018. – № 1. – С. 55-64. (Входит до міжнародних наукометричних баз Index Copernicus, Google Scholar, OECSP, OAJI, Scholar Steer, SIS, Cyberleninka, CiteFactor, TIU Hannover, I2OR)

10. *Любарский М.М.* Синтез и анализ логических Х-функций [Text] / *М.М. Любарский*, В.Г. Абдуллаев, В.И. Хаханов, С.В. Чумаченко, Е.И. Литвинова, И.В. Хаханов // Радиоэлектроника и информатика. – 2018. – №2. – С. 18-28. (Входит до міжнародних наукометричних баз Index Copernicus,

Google Scholar, OECSP, OAJI, Scholar Steer, SIS, Cyberleninka, CiteFactor, TIU Hannover, I2OR)

11. Hahanov V. Quantum Data Structures for SoC Component Testing [Text] / V. Hahanov, W. Gharibi, S. Chumachenko, E. Litvinova, I. Iemelianov, *M. Liubarskyi* // International Journal of Design, Analysis & Tools for Integrated Circuits & Systems.– Oct. 2017.– Vol. 6, iss. 1. – P. 69-76. (Входить до міжнародної наукометричної бази EBSCO Information Services).

12. Хаханов В.И. Gartner 2017 топ-технологии: их анализ и применение [Text] / В.И. Хаханов, А.С. Мищенко, И.В. Емельянов, *М.М. Любарский*, Т.И. Соклакова, В.Г. Абдулаев // Paradigmata poznání. Vědecko vydavatelské centrum «Sociosféra-CZ», s.r.o., Praha, Česká republika. – 2017. – №4. – P. 33-62. (The journal is indexed by Electronic Research Library, Russia; Research Bible, China; Scientific Indexing Services, USA; Cite Factor, Canada; General Impact Factor, India; Scientific Journal Impact Factor, India; CrossRef, USA; ORCID, USA).

13. Bani Amer T. Компьютерные модели облачных сервисов [Text] / T. Bani Amer, В.И. Хаханов, И.В. Емельянов, *М. Любарский* // АСУ и приборы автоматики. – 2015. – Вып. 173. – С.48-57. (Входить до міжнародних наукометричних баз Google Scholar, Cyberleninka).

Результати, які засвідчують апробацію матеріалів дисертації:

14. Hahanov V. Qubit description of the functions and structures for computing [Text] / V. Hahanov, E. Litvinova, S. Chumachenko, *M. Liubarskyi* // Proc. of the IEEE East-West Design & Test Symposium. – 2016. – Yerevan, Armenia. – 6 p. (Входить до міжнародних наукометричних баз Scopus, IEEE Xplore).

15. Hahanov V. Qubit test synthesis for the black box functionalities [Text] / V. Hahanov, E. Litvinova, S. Chumachenko, I. Iemelianov, *M. Liubarskyi* // Proc. of 5th Prague Embedded Systems Workshop.– June 29-30, 2017.– Roztoky u Prahy, Czech Republic.– P.45-51.

16. Hahanov I. Qubit deductive fault simulation of black box functionalities [Text] / I. Hahanov, A. Hahanova, V. Hahanov, E. Litvinova, S. Chumachenko, *M.*

Liubarskyi // Proc. of 5th Prague Embedded Systems Workshop. June 29-30, 2017. Roztoky u Prahy, Czech Republic. P.83-91.

17. Hahanov V. Quantum Data Structures for SoC Component Testing [Електронний ресурс] / V. Hahanov, W. Gharibi, K. L. Man, S. Chumachenko, E. Litvinova, I. Iemelianov, *M. Liubarskyi* // Proc. of the International Conference on Recent Advancements in Computing, IoT and Computer Engineering Technology. – The Tamkang University. – Taipei, Taiwan. – October 23-25, 2017. – 1 p.

18. Hahanov V. Quantum memory-driven computing for test synthesis [Text] / V. Hahanov, W. Gharibi, E. Litvinova, *M. Liubarskyi*, A. Hahanova // Proc. of the IEEE East-West Design and Test Symposium. – Novi Sad. – October, 2017. – P. 6. (Входить до міжнародних наукометричних баз Scopus, IEEE Xplore).

19. Vladimir Hahanov. Qubit test synthesis of the functionality [Text] / V. Hahanov, Tamer Bani Amer, E. Litvinova, T. Soklakova, *M. Liubarskyi*, N. Shavlak, K. Dziuba // Proc. of the 14th International Conference “The Experience of Designing and Application of CAD Systems in Microelectronics” (CADSM). – Lviv-Polyana, Ukraine. – 21-25 February, 2017. – P. 251 – 255. (Входить до міжнародних наукометричних баз Scopus, IEEE Xplore).

20. Hahanov V. Qubit Minimization of Boolean Functions [Text] / Vladimir Hahanov, Ka Lok Man, *Mykhailo Liubarskyi*, Ivan Hahanov, Svetlana Chumachenko // Proc. of the International MultiConference of Engineers and Computer Scientists. – Hong Kong. – March 14-16, 2018. – Vol II IMECS 2018. – Pp. 690-695.

21. Hahanov V. Quantum Deductive Fault Simulation [Text] / V. Hahanov, W. Gharibi, *M. Liubarskyi*, E. Litvinova, M. Gharibi, S. Chumachenko, I. Hahanov, and A. Hahanova // Proc. of Int'l Conf. Modeling, Sim. and Vis. Methods (MSV'18) in The 2018 World Congress in Computer Science, Computer Engineering, & Applied Computing (CSCE2018) . – Jul 30-Aug 02, 2018. – Las Vegas, USA. – P. 10-16.

Публікації, які додатково відображають наукові результати дисертації:

22. *Mykhailo Liubarskyi*. IoT Computing Evolution [Text] / *M. Liubarskyi*, I. Hahanov // Матеріали XXI Міжнародного молодіжного форуму «Радіоелектроніка та молодь у XXI ст.». – 25-27 квітня 2017. – Ч. 5. – С.46-47.

ДОДАТОК Б

АПРОБАЦІЯ РЕЗУЛЬТАТІВ

Результати, які засвідчують апробацію матеріалів дисертації:

1. Hahanov V. Qubit description of the functions and structures for computing [Text] / V. Hahanov, E. Litvinova, S. Chumachenko, M. Liubarskyi // Proc. of the IEEE East-West Design & Test Symposium. – 2016. – Yerevan, Armenia. – 6 p. (Входить до міжнародних наукометричних баз Scopus, IEEE Xplore). – *очна участь з доповіддю*

2. Hahanov V. Qubit test synthesis for the black box functionalities [Text] / V. Hahanov, E. Litvinova, S. Chumachenko, I. Iemelianov, M. Liubarskyi // Proc. of 5th Prague Embedded Systems Workshop.– June 29-30, 2017.– Roztoky u Prahy, Czech Republic.– P.45-51. – *заочна участь з доповіддю*

3. Hahanov I. Qubit deductive fault simulation of black box functionalities [Text] / I. Hahanov, A. Hahanova, V. Hahanov, E. Litvinova, S. Chumachenko, M. Liubarskyi // Proc. of 5th Prague Embedded Systems Workshop. June 29-30, 2017. Roztoky u Prahy, Czech Republic. P.83-91. – *заочна участь з доповіддю*

4. Hahanov V. Quantum Data Structures for SoC Component Testing [Електронний ресурс] / V. Hahanov, W. Gharibi, K. L. Man, S. Chumachenko, E. Litvinova, I. Iemelianov, M. Liubarskyi // Proc. of the International Conference on Recent Advancements in Computing, IoT and Computer Engineering Technology. – The Tamkang University. – Taipei, Taiwan. – October 23-25, 2017. – 1 p. – *заочна участь з доповіддю*

5. Hahanov V. Quantum memory-driven computing for test synthesis [Text] / V. Hahanov, W. Gharibi, E. Litvinova, M. Liubarskyi, A. Hahanova // Proc. of the IEEE East-West Design and Test Symposium. – Novi Sad. – October, 2017. – P. 6. (Входить до міжнародних наукометричних баз Scopus, IEEE Xplore). – *очна участь з доповіддю*

6. Vladimir Hahanov. Qubit test synthesis of the functionality [Text] / V. Hahanov, Tamer Bani Amer, E. Litvinova, T. Soklakova, *M. Liubarskyi*, N. Shavlak, K. Dziuba // Proc. of the 14th International Conference “The Experience of Designing and Application of CAD Systems in Microelectronics” (CADSM). – Lviv-Polyana, Ukraine. – 21-25 February, 2017. – P. 251 – 255. (Входить до міжнародних наукометричних баз Scopus, IEEE Xplore). – *очна участь з доповіддю*

7. Hahanov V. Qubit Minimization of Boolean Functions [Text] / Vladimir Hahanov, Ka Lok Man, *Mykhailo Liubarskyi*, Ivan Hahanov, Svetlana Chumachenko // Proc. of the International MultiConference of Engineers and Computer Scientists. – Hong Kong. – March 14-16, 2018. – Vol II IMECS 2018. – Pp. 690-695. – *заочна участь з доповіддю*

8. Hahanov V. Quantum Deductive Fault Simulation [Text] / V. Hahanov, W. Gharibi, *M. Liubarskyi*, E. Litvinova, M. Gharibi, S. Chumachenko, I. Hahanov, and A. Hahanova // Proc. of Int'l Conf. Modeling, Sim. and Vis. Methods (MSV'18) in The 2018 World Congress in Computer Science, Computer Engineering, & Applied Computing (CSCE2018) . – Jul 30-Aug 02, 2018. – Las Vegas, USA. – P. 10-16. – *очна участь з доповіддю*

Публікації, які додатково відображають наукові результати дисертації:

9. *Mykhailo Liubarskyi*. IoT Computing Evolution [Text] / *M. Liubarskyi*, I. Hahanov // Матеріали XXI Міжнародного молодіжного форуму «Радіоелектроніка та молодь у XXI ст.». – 25-27 квітня 2017. – Ч. 5. – С.46-47. – *очна участь з доповіддю*

ДОДАТОК В
ДОКУМЕНТИ, ЩО ПІДТВЕРДЖУЮТЬ ВПРОВАДЖЕННЯ
РЕЗУЛЬТАТІВ ДИСЕРТАЦІЇ



АКТ

про впровадження у навчальний процес ХНУРЕ результатів дисертаційної роботи Любарського Михайла Михайловича «Квантові моделі та методи аналізу логічних Х-функцій» на здобуття наукового ступеня кандидата технічних наук за спеціальністю 05.13.05 – комп'ютерні системи та компоненти

Комісія у складі: зав. каф. АПОТ проф. Чумаченко С.В., доц. каф. АПОТ Шкиля О.С., проф. каф. АПОТ Литвинової Є.І. розглянула матеріали дисертаційної роботи пошукача Любарського М.М., які використовуються у навчальному процесі кафедри АПОТ ХНУРЕ у 2018-2019 навчальному році, і дійшла наступного висновку.

Розроблені у дисертаційній роботі моделі і методи аналізу логічних Х-функцій придатні до використання у навчальних цілях для курсів «Квантові обчислення», «Основи комп'ютерної діагностики», «System on Chip», а саме: 1) удосконалений квантовий метод генерації тестів, що перевіряють, для одиночних константних несправностей логічних функцій шляхом взяття булевих похідних за їх кубітним покриттям; 2) удосконалений квантовий метод дедуктивного моделювання константних несправностей логічних функцій шляхом синтезу матриць булевих похідних за їх кубітним покриттям; 3) паралельний метод генерації тестів для несправностей Х-функцій від кінцевого числа змінних на основі взяття похідних за кубітним покриттям, що характеризується мінімальністю тестових наборів; 4) дедуктивні формули аналізу несправностей для Х-функцій, які характеризуються поодинокими векторами похідних за всіма змінними, що дає можливість синтезувати секвенсор моделювання дефектів, інваріантний до вхідних тестових наборів; 5) метод синтезу дедуктивних кубітних покриттів для моделювання несправностей логічних функцій, який характеризується оригінальністю математичних рішень, високим рівнем паралелізму і компактністю структур даних, що дає можливість використовувати його для синтезу, аналізу, тестування, верифікації і діагностування цифрових систем на кристалах.

Зав. каф. АПОТ проф. Чумаченко С.В.,

Доц. каф. АПОТ Шкиль О.С.,

Проф. каф. АПОТ Литвинова Є.І.

IT DELIGHT

СПРАВКА

о возможности внедрения квантовых моделей и методов анализа логических X-функций в методологическое и технологическое обеспечение компании «DIT DELIGHT LIMITED»

Разработанные на кафедре АПВТ ХНУРЭ при 30% участии соискателя Любарского Михаила Михайловича модели, структуры данных, методы синтеза и анализа логических схем, включая X-функции, дают возможность существенно уменьшить время синтеза тестов и моделирования неисправностей благодаря кубитному описанию цифровых схем.

Отдельные сервисы синтеза и анализа кубитных моделей цифровых устройств и компонентов реализованы в виде программных приложений и пригодны для внедрения в научно-практическую компанию «DIT DELIGHT LIMITED» для параллельного выполнения вычислительных процедур, характерных квантовому компьютерингу. Среда проектирования: SWIFT, C++, Verilog, Java и платформы: Microsoft Windows, X Window и Macintosh OS X.

Основными результатами, которые составляют научную новизну и могут быть внедрены в производственный процесс, следует считать:

- 1) Параллельный метод синтеза дедуктивных кубитных покрытий для моделирования X-функций, который характеризуется получением единичной матрицы производных, что дает возможность создавать секвенсор моделирования дефектов, инвариантный к входным тестовым наборам.
- 2) Усовершенствованные memory-driven архитектуры и алгоритмы для реализации методов тестирования и верификации цифровых систем на кристаллах, которые отличаются параллельным выполнением логических операций над кубитными структурами данных.
- 6) Усовершенствованные квантовые методы генерации тестов и дедуктивного моделирования неисправностей логических функций, которые отличаются от аналогов синтезом матриц булевых производных по их кубитным покрытиям.

Директор компании «DIT Delight Limited»:

Емельянов Игорь Валерьевич,
кандидат технических наук
25 октября 2018 года



ДОДАТОК Г

ПРОГРАМНИЙ ДОДАТОК (ФРАГМЕНТ)

```

// CircuitDesignerDataSource.swift

// X-function Quantum Modeling and Simulation: logic module

protocol CircuitDesignerDataSource {
    var circuit: CircuitView? {get set }
    var updatedFrame: NSRect { get }
    var circuitDesignerView: CircuitDesignerView! {get set }
    var tracker: NSBezierPath? {get set }
    var dragInterType: DragInteractionType { get set }
}

//
// CircuitDesignerView.swift
// QuantumModeling
//
class CircuitDesignerView: NSView {
    var dataSource: CircuitDesignerDataSource?
    override func updateTrackingAreas () {
        super.updateTrackingAreas ()
        removeTrackingArea (trackingAreas [0])
        let trackingArea = NSTrackingArea (rect: bounds, options: [.activeInKeyWindow,
.mouseMoved], owner: self, UserInfo: nil)
        addTrackingArea (trackingArea)
    }
    override var isFlipped: Bool {
        return true
    }
    override func draw (_ dirtyRect: NSRect) {
        super.draw (dirtyRect)
        NSColor.black.setStroke ()
        for (_, Link) in dataSource! .circuit! .links {
            link.stroke ()
        }
        if dataSource? .dragInterType == .linking {
            NSColor.blue.setStroke ()
        } else if dataSource? .dragInterType == .selection {
            NSColor.black.setStroke ()
            NSColor.blue.withAlphaComponent (0.1) .setFill ()
            dataSource? .tracker? .fill ()
        }
        dataSource! .tracker? .stroke ()
    }
}

// CircuitEditorWindowController.swift
// X-function Modeling and Simulation
// MARK: Methods
override func windowDidLoad () {
    super.windowDidLoad ()
    let splitViewController = contentViewController as! SplitViewController
    let editorViewController = splitViewController.childViewControllers [1] as! EditorView-
Controller
    // NSNotificationCenter.defaultCenter (). addObserver (editorViewController, selector: Selec-
tor ("resizeDesignerViewToFit"), name: NSWindowDidResizeNotification, object: window)
    @IBAction func collapsingSideMenu (_ sender: AnyObject) {
        let splitVC = contentViewController as! SplitViewController
        let segControl = sender as! NSSegmentedControl
        if segControl.selectedSegment == 0 {
            splitVC.splitViewItems [0] .isCollapsed != SplitVC.splitViewItems [0] .isCollapsed
        } else {
            splitVC.splitViewItems [2] .isCollapsed != SplitVC.splitViewItems [2] .isCollapsed
        }
    }
    override func prepare (for segue: NSStoryboardSegue, sender: Any?) {
        if segue.identifier == "SimulationTableSegue" {
            let editorViewController = contentViewController as! EditorViewController
            let tableViewController = segue.destinationController as! ViewController
            tableViewController.circuit = editorViewController.circuit
        } else if segue.identifier == "QTableSegue" {
            let splitViewController = contentViewController as! SplitViewController
            let editorViewController = splitViewController.childViewControllers [1] as! Editor-
ViewController
            let targetViewController = segue.destinationController as! QVectorTableController
            targetViewController.circuit = editorViewController.circuit
            for outPort in editorViewController.circuit.outPorts {
                if outPort.element! != nil {
                    targetViewController.elements.append (outPort.element!)
                }
            }
        }
    }
}
}

```