УДК 519.711; 004.421
УКПП
№ держреєстрації № 0115U000712 від 04.03.2015
Інв. №

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки
(ХНУРЕ)
61166, м. Харків, пр. Науки, 14, тел. (057) 70-21-425

ЗАТВЕРДЖУЮ
Ректор ХНУРЕ
д-р техн. наук, проф.

_____ В.В. Семенець

«___»_____2017 р.

# ЗВІТ

# ПРО НАУКОВО-ДОСЛІДНУ РОБОТУ

“КІБЕРФІЗИЧНА СИСТЕМА – «РОЗУМНЕ ХМАРНЕ УПРАВЛІННЯ ТРАНСПОРТОМ» (CYBER PHYSICAL SYSTEM – SMART CLOUD TRAFFIC CONTROL)”
(тема № 297)
(остаточний)

Керівник НДР

д-р техн. наук, проф. В.І. Хаханов

25.12.2017

2017

Рукопис закінчено 25 грудня 2017 р.

Результати роботи розглянуті науково-технічною радою ХНУРЕ,

протокол № від 26 грудня 2017

# СПИСОК АВТОРІВ

| | | |
|---|---|---|
| *Керівник НДР,<br>головний науковий співробітник кафедри АПОТ, д-р техн. наук, професор | | В.І. Хаханов (вступ; розділи 1-10, висновки) |
| *Відповідальний виконавець НДР,<br>Пров. наук. співроб. (за сум.), завідувач кафедри АПОТ, д-р техн. наук, професор | | С.В. Чумаченко<br>(реферат; розділи 3, 10; висновки) |
| *Виконавці: | | |
| Наук. співроб., канд. техн. наук | | Д.Є. Шуклін (розділ 10) |
| *Ст. наук. співроб., канд. техн. наук | | В.М. Таранов (розділ 10) |
| *Наук. співроб. кафедри АПОТ | | О.С. Міщенко (розділ 10) |
| *Наук. співроб. кафедри АПОТ | | І.В. Ємельянов (розділ 10) |
| *Мол. наук. співроб. кафедри АПОТ | | А.С. Ізотов (розділ 10) |
| *Наук. співроб. каф. АПОТ,<br>канд. техн. наук, доцент | | Г.В. Хаханова (розділ 10) |
| *Ст. наук. співроб. (за сум.)<br>доц. каф. АПОТ, канд. техн. наук | | О.С. Шкиль (розділ 10) |
| *Наук. співроб. каф. АПОТ (за сум.),<br>канд. техн. наук, доц. | | Г.П. Фастовець (розділ 10) |
| *Ст. наук. співроб. (за сум.), проф. каф. АПОТ, д-р техн. наук, професор | | І.В. Хаханова (розділ 7, 10) |
| *Наук. співроб. (за сум.), проф. каф. АПОТ, канд. техн. наук, доцент | | В.П. Немченко (розділ 10) |
| *Наук. співроб. (за сум.), доц. каф. АПОТ, доцент | | Е.М. Кулак (розділ 10) |
| *Інженер, аспірант каф. АПОТ | | А.С. Сірокурова (Гребенюк) (розділ 10) |
| *Інженер, аспірант каф. АПОТ | | О.Ю. Мова (розділ 10) |
| Проф. каф. АПОТ, д-р техн. наук, проф. | | Є.І. Литвинова (розділи 8, 10) |
| Проф. каф. АПОТ, д-р техн. наук | | Г.Ф. Кривуля (розділ 10) |
| Доцент каф. АПОТ, канд. техн. наук | | І.В. Філіпенко (розділ 9, 10) |
| Старш. викл. каф. АПОТ | | В.І. Обрізан (розділи 5, 10) |
| Старш. викл. каф. АПОТ | | О.С. Адамов (розділ 10) |
| Доц. каф. АПОТ, канд. техн. наук | | Д.Ю. Кучеренко (розділ 10) |
| Старш. викладач каф. АПОТ | | А.Н.Зіарманд (розділи 1-4, 10) |
| Аспірант каф. АПОТ | | В.О. Мізь (розділ 10) |
| Аспірант каф. АПОТ | | Т.Г. Соклакова (розділ 10) |
| Аспірант каф. АПОТ | | Тамер Бані Амер (розділи 6-8) |
| Аспірант кафедри АПОТ | | Багдаді Аммар Авні Аббас (розділ 10) |

* - з повною або частковою оплатою в межах фінансування

Нормоконтролер Т.Г. Силантьєва

# РЕФЕРАТ

Звіт про НДР: 418 с., 114 рис., 29 табл., 207 джерел, 2 додатки.

Публікації: монографії 4, навчальні посібники та підручники 3, статті 29, тези 140, патенти 3.

Ключові слова: КІБЕРФІЗИЧНА СИСТЕМА, ХМАРНИЙ СЕРВІС, МОНІТОРИНГ, УПРАВЛІННЯ, ІНТЕЛЕКТУАЛЬНА ІНФРАСТРУКТУРА ДОРОЖНЬОГО РУХУ, CLOUD TRAFFIC CONTROL, CLOUD STREETLIGHT, DIGITAL CAR IDENTIFICATION, КУБІТНІ СТРУКТУРИ ДАНИХ, КВАНТОВІ ОБЧИСЛЕННЯ.

Об’єкт дослідження – хмарні кіберсистемні технології моніторингу та управління транспортними засобами на основі використання віртуальних світлофорів, дорожніх знаків і мобільних гаджетів для ідентифікації, радіолокації і радіонавігації учасників дорожнього руху.

Предмет дослідження – транспортні потоки, віртуальна інфраструктура дорожніх повідомлень, програмно-апаратні мобільні системи ідентифікації, моніторингу та управління дорожнім рухом на основі застосування віртуальних світлофорів.

Мета роботи – підвищення якості та безпеки дорожнього руху за рахунок створення кібер-фізичної системи – інтелектуальної інфраструктури дорожнього руху (TCS), що включає моніторинг і управління у реальному масштабі часу на основі використання мобільних гаджетів транспортних засобів і хмарних (віртуальних) світлофорів, що дає можливість підвищити якість життя водія, мінімізувати часові і матеріальні витрати при організації дорожнього руху, створювати інноваційні рішення соціальних, гуманітарних, економічних та екологічних проблем.

Методи дослідження — булева алгебра, векторно-асоціативна логіка, теорія множин, теорія графів, теорія цифрових автоматів, теорії мереж, теорії аналізу кіберпростору та великих даних, захисту інформації, теорії кіберфізичних систем, теорії квантових обчислень, теорії віртуальних комп’ютерів, internet of things, big data analytics, кубітні структури даних, квантові технології обчислення, моделі кіберфізичних систем – для управління даними.

За завданням НДР проведено аналіз стану проблеми, висвітлені основні задачі та проблемні питання, означені шляхи їх реалізації, наведено характеристику основних наукових та практичних результатів роботи.

Науково-практичним результатом, на одержання якого була спрямована НДР, є розробка та впровадження методів і моделей, орієнтованих на створення хмарних сервісів human-free та streetlight-free керування транспортними засобами та дорожнім рухом на основі точного позиціонування (моніторингу) та управління автомобілями, спарених з мобільними пристроями.

ЗМІСТ

СПИСОК АВТОРІВ ..... 2
СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ ..... 7
ПЕРЕДМОВА ..... 8
ВСТУП ..... 9
1 МОДЕЛІ І МЕТОДИ МОНІТОРИНГУ ТА УПРАВЛІННЯ ТРАНСПОРТОМ ..... 15
1.1 Хмарний комп'ютинг і системи управління транспортом ..... 15
1.2 Моделі міських інфраструктур дорожнього руху ..... 22
1.3 Системи позиціювання та навігації транспортних засобів ..... 27
1.4 Електронне обладнання сучасного автомобіля ..... 34
1.5 Кіберфізичні системи хмарного управління транспортом ..... 37
2 АНАЛІЗ ЯКОСТІ ТОПОЛОГІЧНИХ СТРУКТУР ДОРОЖНЬОГО РУХУ ..... 48
2.1 Критерії якості інфраструктури дорожнього руху ..... 48
2.2 Функції прохідності дорожньої інфраструктури ..... 54
2.3 Метрика е-інфраструктури дорожнього руху ..... 59
2.4 Топології зв'язків компонентів системи ..... 64
2.5 Спрощений критерій якості топології ..... 68
2.6 Наведена метрика топології графа ..... 74
2.7 Висновки до розділу 2 ..... 81
3 МОДИФІКАЦІЯ АЛГОРИТМУ ДЕЙКСТРИ ДЛЯ ПОШУКУ МАРШРУТУ ..... 83
3.1 Вступ ..... 83
3.2 Метод пошуку найкоротших шляхів у дорожній інфраструктурі ..... 84
3.3 Метод пошуку найкоротших відстаней між парами вершин ..... 92
3.4 Пошук мінімальних шляхів для графа з одиничними дугами ..... 96
3.5 Визначення мінімальних шляхів у трикутній топології ..... 103
3.6 Пошук мінімальних шляхів у прямокутній топології ..... 110
3.7 Висновки до розділу 3 ..... 118
4 КІБЕРФІЗИЧНА СТРУКТУРА ХМАРНОГО ТРАНСПОРТНОГО КОМП'ЮТИНГУ ..... 120
4.1 Інноваційна привабливість "Cloud Traffic Control" ..... 121
4.2 Структура сервісів хмарного управління транспортом ..... 129
4.3 Розумний хмарний світлофор ..... 133
4.4. Структурно-аналітична модель хмарного управління ..... 140
4.5 Модельно-фізичні експерименти на road-інфраструктурі ..... 151
4.6 Висновки до розділу 4 ..... 157

5 МУЛЬТИВЕРСНИЙ ПАРАЛЕЛЬНИЙ СИНТЕЗ ЦИФРОВИХ СТРУКТУР МОНІТОРИНГА ТА УПРАВЛІННЯ ТРАФІКОМ НА ОСНОВІ SYSTEMC СПЕЦИФІКАЦІЇ ..... 161
5.1 Вступ ..... 161
5.2 Методи системного та архітектурного синтезу інтерфейсних структур ..... 163
5.3 Моделі опису логічних блоків системного рівня ..... 165
5.4 Програмно-апаратна реалізація моделей ..... 170
5.5 Висновки до розділу 5 ..... 177
6 КОМП'ЮТИНГОВІ МОДЕЛІ ХМАРНИХ СЕРВІСІВ ..... 180
6.1 Вступ ..... 180
6.2 MAT-комп'ютинг для фізичного, віртуального і космологічного простору ..... 190
6.3 Комп'ютинг квантової телепортації ..... 195
6.4 Висновки до розділу 6 ..... 197
7 СИНТЕЗ Q-ТЕСТІВ ЗА BLACK BOX КУБІТНИМ ОПИСОМ ..... 199
7.1 Реалізація логічних функціональностей на елементах пам'яті ..... 199
7.2 Кубітний метод синтезу тестів ..... 206
7.3 Обчислення булевих похідних для Q-синтезу тестів ..... 217
7.4 Дедуктивний аналіз несправностей цифрових структур ..... 226
7.5 Висновки до розділу 7 ..... 233
8 СИНТЕЗ І АНАЛІЗ КУБІТНИХ МОДЕЛЕЙ ЦИФРОВИХ СИСТЕМ ..... 236
8.1 Вступ ..... 236
8.2 Квантові або кубітні структури даних ..... 238
8.3 Синтез квант-вектора комбінаційної схеми ..... 240
8.4 Мінімізація квант-вектора схеми ..... 242
8.5 Імплементація куба функціональності в технологічну структуру PLD 244
8.6 Модель квантового процесора ..... 247
8.7 Алгоритм моделювання квантових покриттів цифрових компонентів 252
8.8 Моделювання синхронних цифрових схем ..... 256
8.9 Структура хмарного сервісу QuaSim для моделювання цифрових пристроїв ..... 261
8.10 Хмарний сервіс тестування ..... 266
8.11 Висновки до розділу 8 ..... 272
9 ПЕРЕВАГИ ХМАРНИХ СЕРВІСІВ КІБЕРСИСТЕМИ ..... 277
9.1 Технічні і функціональні можливості TCS ..... 278
9.2 Проблеми, які вирішуються за допомогою цифрової ідентифікації ..... 279
9.3 Аргументи проти впровадження хмари в масштабах країни ..... 280
9.4 Практичні приклади впровадження компонентів TCS-системи ..... 281
9.5 Технології високоточної ГНСС-навігації транспортних засобів ..... 282
9.6 Система управління корпоративними перевезеннями ..... 285
9.7 Організація зв'язків «хмара – автомобіль», «хмара – інфраструктура» 287

9.8 Структура автомобільного блоку CAR-ID ..... 291
9.9 Засоби моніторингу та управління дорожнім рухом ..... 293
9.10 Дистанційний модуль "SHERLOCK" ..... 295
9.11 Сьогодення і цифрове майбутнє планети ..... 299
9.12 Ринкова привабливість і техніко-економічна характеристика проекту TCS ..... 301
9.13 Завдання, що підлягають вирішенню для реалізації TCS-хмари ..... 307
10 ІНШІ РЕЗУЛЬТАТИ, ОТРИМАНІ ТА РЕАЛІЗОВАНІ В МЕЖАХ НДР 310
10.1 Результативність виконання НДР ..... 310
10.2 Монографії та навчальні посібники ..... 311
10.3 Патенти ..... 313
10.4 Наукові статті ..... 313
10.5 Тези доповідей на конференціях ..... 317
10.6 Договори з зацікавленими організаціями ..... 336
10.7 Організація і проведення конференцій та семінарів ..... 337
10.8 Виконані інноваційні проекти та проекти, що виконуються ..... 338
10.9 Отримані гранти ..... 338
10.10 Теми захищених та поданих дисертацій ..... 341
10.11 Виставки ..... 346
10.12 Теми захищених магістерських робіт за тематикою НДР ..... 349
10.13 Залучення позабюджетних коштів ..... 350
10.14 Розробки ..... 351
10.15 Використання результатів НДР у навчальному процесі ..... 359
10.16 Доробок авторів проекту до початку виконання НДР ..... 360
ВИСНОВОК ..... 369
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ ..... 374
ДОДАТОК А ..... 398
НАГОРОДИ, ДИПЛОМИ, СЕРТИФІКАТИ ..... 398
ДОДАТОК В ..... 417
ДОКУМЕНТИ, ЩО ПІДТВЕРДЖУЮТЬ ВПРОВАДЖЕННЯ РЕЗУЛЬТАТІВ ..... 417

# СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

ТЗ – транспортний засіб
TCS – Traffic Cyber-physical System (Інтелектуальна інфраструктура дорожнього руху
PtMS – Parallel transportation management systems
Adapts – Adaptive Platforms for Transportation Systems
VLSI – Very Large Scale Integrated
RTL – Register Transfer Level
LAMP – Logical Associative MultiProcessor
НВІС – надвелика інтегральна схема
SoC – System-on-Chip
MIH – Media Independent Handover
MIHF – функція MIH
SAP – Service Access Points
MIES – сервіс медіа-незалежних подій
MICS – сервіс медіа-незалежних команд
MIIS – сервіс медіа-незалежної інформації
FANSCI – fast handover scheme for car-to-car communication
HTB – планувальник пакетів Hierarchical Token Buckets
ПЛІС – Програмована логічна інтегральна схема
TAB – Tests – Assertions –Blocks
НС – нейронна мережа
MITCS – мобільна інтелектуальна система управління трафіком
DVNS/ SVNS – системи динамічної/статичної навігації
GIS-T – Geographic Information System for Transportation
ГА – генетичний алгоритм
ГТП – граф транспортних потоків
ДНФ – Диз'юнктивна нормальна форма
КНФ – Кон'юнктивна нормальна форма
ДТП – дорожньо-транспортна пригода
VANET – vehicular ad hoc networks
VTL – віртуальний світлофор
ГСА – граф-схема автомата
МОА – мова опису апартури
СУДР – система управління дорожнім рухом

# ПЕРЕДМОВА

Пропонується кіберфізична система для хмарного управління транспортом як інтелектуальна віртуальна інфраструктура моніторингу та управління дорожнім рухом в реальному масштабі часу на основі використання глобальних систем позиціонування і навігації, автомобільних комп'ютерів, мобільних гаджетів, розумних світлофорів та Інтернету з метою підвищення якості та безпеки пересування транспортних засобів, мінімізації часових і матеріальних витрат при русі автомобілів за заданими маршрутами.

Основна інноваційна ідея – поступове перенесення світлофорів з поверхні землі у віртуальний хмарний простір для управління транспортом, обладнаним мобільним гаджетом або автомобільним комп'ютером, на екран (лобове скло) якого виводяться: карта з дорожніми знаками, маршрут прямування, координати учасника дорожнього руху та реальні керуючі сигнали віртуального розумного світлофора.

Пропонується комплекс інноваційних технологій для вирішення соціальних, гуманітарних, економічних, паливно-енергетичних, страхових, кримінальних та екологічних проблем на основі створення та застосування хмарного сервісу цифрового моніторингу та управління транспортом. Технології та функціональні компоненти інтегруються в просту для реалізації автоматну модель кіберфізичної системи інтерактивної взаємодії у реальному часі.

# ВСТУП

**Актуальність дослідження.** Кіберфізична інфраструктура дорожнього руху і транспортний засіб створюють бінарну систему активної взаємодії для задоволення потреб людства. Вони неявно формулюють дві актуальних доктрини, здатні радикально змінити існуючий світ: 1) Світлофорів немає у водній стихії і у повітряному просторі, значить, їх не повинно бути на дорогах. Прибрати все знаки з дорожньої інфраструктури, означає створити зелену планету і чисті міста. 2) Зробити автомобіль розумним, з автономним управлінням, означає звести до нуля тисячі щоденних аварій на планеті. Щоб реалізувати дані доктрини необхідно економічно обгрунтовано вирішити такі технологічні ринкові проблеми: 1) Довести глибину позиціонування фізичного об'єкта в оцифрованному просторі до 10 сантиметрів. 2) На кілька порядків підвищити пропускну здатність бездротових телекомунікаційних каналів зв'язку для підтримки е-інфраструктури. 3) Створити надійні масштабовані компьютінгові сервіси цифрового моніторингу і хмарного управління трафіком. 4) Змусити населення планети повірити в інноваційну кіберкультуру управління транспортом. Наслідки від реалізації двох доктрин – трильйони доларів чистої економії, нова якість життя громадян і зелена планета. Ось кілька очевидних трансформацій у фізичному світі, що безпосередньо призводять до соціально-екологічних і економічних результатів: 1) Прибрати світлофори і дорожні знаки та автомобільні номери означає зберегти сотні тисяч тонн металу, сотні тисяч кіловат-годин електроенергії і мільярди доларів на їх установку і експлуатацію.

Отже, актуальність дослідження визначається наступними чинниками: 1) Збереження екології за рахунок зменшення забруднення довкілля, підвищення тривалості та якості життя людини, економія паливно-енергетичних ресурсів за рахунок впровадження в інфраструктуру інтелектуальних світлофорів. 2) Точна ідентифікація автомобілів, точний цифровий моніторинг позиціонування траснпортного засобу (ТЗ) у часі і

просторі, дистанційне керування автомобілем через хмару, істотне зниження аварійності. 3) Моніторинг позиціонування і пересування ТЗ, квазіоптимальне виконання замовлень з перевезення пасажирів і вантажів з точки зору мінімізації матеріальних та/або часових витрат. Інноваційна ідея проекту – хмарне online управління автомобілем за допомогою розумного хмарного світлофору на основі точного сантиметрового позиціонування, не мала аналогів на початку виконання проекту. За 4 останніх роки рівень капіталізації Automotive + Cloud Computing досяг трильйона доларів і, згідно з Top Gartner Emerging Technologies, 8 з 10 провідних лідерів планети переорієнтували свій бізнес на автономне пересування транспортних засобів, включаючи IBM, Google, Microsoft, NASA, Amazon, Synopsis. Остання разом з Google створила спеціальний чіп Automotive Cloud Control (2017).

**Проблема, на вирішення якої було спрямовано дослідження –** створення кіберфізичної системи цифрового моніторингу та оптимального управління транспортними засобами на основі використання віртуальних світлофорів, розробки надійних масштабованих комп'ютингових сервісів, яка інтегрує два найактуальніших і популярних наукових напрямки: транспорт і комп'ютинг для отримання якісно нових умов життя в розумному автомобілі.

**Науково-практична задача дослідження**: створення кіберфізичних моделей комп'ютингових сервісів для цифрового моніторингу транспортних потоків і хмарного керування автомобілями з метою істотного зменшення часу виконання замовлених маршрутів.

**Сутність дослідження** – створення компонентів кіберфізичної інфраструктури безпечного дорожнього руху за рахунок поступового перенесення дорожніх знаків, світлофорів у кіберпростір і точного позиціонування транспорту з метою хмарного online керування автомобілем на основі розумної суперпозиції замовленого маршруту з оперативним цифровим моніторингом дорожньої обстановки, яка відображається на сенсорному дисплеї автомобіля. На рис. 1 представлена тріада транспортного комп'ютингу (cloud-car-streetlight), яка еволюційно перетворюється в cloud-

car computing шляхом перенесення керуючих знаків з фізичної інфраструктури до кібер-хмари для їх відображення на моніторі автомобільного комп'ютера.

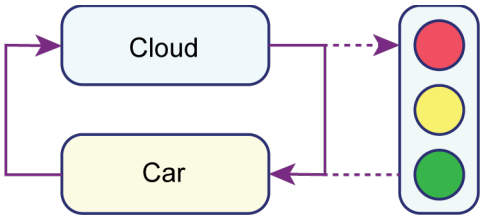



Рис. 1 – Транспортний комп'ютинг

**Ринкова привабливість дослідження.** Впровадження хмарного сервісу управління транспортом призведе до збереження екології планети і до зменшення: 1) часу проходження замовлених маршрутів; 2) споживання енергетичних ресурсів і матеріальних витрат на створення і експлуатацію світлофорів, дорожніх знаків, автомобільних номерів; 3) числа аварій і крадіжок автомобілів.

**Об’єкт дослідження –** хмарні кіберсистемні технології моніторингу та управління ТЗ на основі використання віртуальних світлофорів, дорожніх знаків і мобільних гаджетів для ідентифікації, радіолокації і радіонавігації учасників дорожнього руху.

**Предмет дослідження –** транспортні потоки, віртуальна інфраструктура дорожніх повідомлень, програмно-апаратні мобільні системи ідентифікації, моніторингу та управління дорожнім рухом на основі застосування віртуальних світлофорів.

**Методи дослідження** – топології інфраструктур дорожнього руху, булева алгебра, теорія множин, теорія графів, теорія цифрових автоматів і оптимізації шляхів на графах, паралельні методи хмарних обчислень і структури даних; кіберфізичний комп’ютинг, теорія алгоритмів, методи, засоби, мови проектування і моделювання; методи і критерії якості

оцінювання топологічних структур; засоби синтезу та аналізу програмно-апаратної інфраструктури хмарних сервісів.

**Мета дослідження** – підвищення якості та безпеки дорожнього руху за рахунок створення кібер-фізичної системи – інтелектуальної інфраструктури дорожнього руху (TCS), що включає моніторинг і управління у реальному масштабі часу на основі використання мобільних гаджетів ТЗ і хмарних (віртуальних) світлофорів, що дає можливість підвищити якість життя водія, мінімізувати часові і матеріальні витрати при організації дорожнього руху, створювати інноваційні рішення соціальних, гуманітарних, економічних та екологічних проблем.

**Задачі дослідження:**

1) Впровадження технології радіочастотного цифрового позиціонування транспортного засобу (гаджету) з точністю до 2-0,2 м.

2) Створення нової системи мнемонічних і супроводжуючих звукових сигналів на моніторі.

3) Розробка операційного та керуючого автоматів, які об'єднують компоненти TCS в хмарну систему.

4) Проектування масштабуємої системи TCS для світлофора.

5) Розробка сервісів (програмних додатків) на стороні клієнта.

6) Проектування сервісів на стороні хмари (сервера) для вирішення задач оптимізації.

7) Розробка кубітних структур даних і «квантових» матричних процесорів (обчислювачів) на основі технологій Big Data для одночасного і паралельного сервісного обслуговування користувачів TCS-хмари в реальному масштабі часу.

8) Створення інтелектуальних моделей, методів синтезу й аналізу віртуальної інфраструктури.

9) Розробка хмарних сервісів для автотранспортних підприємств.

10) Створення хмарних сервісів для водія.

11) Збір статистичної інформації (інтелектуалізація глобальної, корпоративної та персональної інфраструктури).

12) Створення засобів захисту інформації від санкціонованого доступу у хмарі.

**Стадії інновації:** 1) Створення квантових структур даних для паралельного зберігання даних і матричного аналізу Big Data форматів кіберпростору. 2) Розробка квантоподібних віртуальних паралельних процесорів для швидкого пошуку, розпізнавання та прийняття квазіоптимальних рішень з управління транспортом та інфраструктурою міст. 3) Створення комплексу програм для реалізації хмарних сервісів моніторингу та управління транспортом у реальному часі на основі використання віртуальних хмарних світлофорів. 4) Розробка і практична реалізація моделі експериментальної ділянки міста для управління дорожнім рухом за допомогою хмарних світлофорів, моніторингу та управління транспортними засобами з анімаційною візуалізацією всіх масштабованих процесів на екрані монітора. 5) Верифікація та впровадження програмних і технічних засобів хмарного сервісу моніторингу та управління дорожнім рухом шляхом використання віртуальних світлофорів.

**Основна інноваційна ідея** – поступове перенесення світлофорів з поверхні землі у віртуальний хмарний простір для управління транспортом, обладнаним мобільним гаджетом або автомобільним комп'ютером, на екран (лобове скло) якого виводяться: карта з дорожніми знаками, маршрут прямування, координати учасника дорожнього руху та реальні керуючі сигнали віртуального розумного світлофора.

Основні гіпотези. Пропонується комплекс інноваційних технологій для вирішення соціальних, гуманітарних, економічних, паливно-енергетичних, страхових, кримінальних та екологічних проблем на основі створення та застосування хмарного сервісу цифрового моніторингу та управління транспортом. Технології та функціональні компоненти інтегруються в просту для реалізації автоматну модель кіберфізичної системи інтерактивної взаємодії у реальному часі.

Отримані в процесі виконання дослідження наукові висновки і результати є достовірними, що підтверджується проведеними експериментами, тестуванням і верифікацією запропонованих моделей і методів моніторингу та управління дорожнім рухом.

Практична значимість наукових досліджень підтверджується істотним зменшенням виконання маршрутів руху при моделюванні реальних процесів на фрагментах моделей дорожньої інфраструктури.

Результати в складі моделей, методів і архітектури кіберфізичної інфраструктури впроваджені в навчальний процес Харківського національного університету радіоелектроніки при читанні курсів «Дискретна математика», «Cloud-Fog кіберфізичні системи». Розроблені моделі, методи, архітектура розумного хмарного світлофора, а також модель транспортного комп'ютингу можуть бути реалізовані як компонент проекту при синтезі хмарного сервісу (ДОДАТОК Б).

**Апробація результатів НДР.** Результати роботи були представлені та обговорені на наступних конференціях: IEEE East-West Design and Test Symposium 2015 (Georgia), 2016 (Armenia), 2017 (Сербія); XII International IEEE Conference "Modern Problems of Radio Engineering, Telecommunications, and Computer Science", 2016, Lviv-Slavske, Ukraine; 13th International Conference: The Experience of Designing and Application of CAD Systems in Microelectronics, CADSM 2015, 2017, Lvov, Ukraine; 2017 SAE Інтернешнл (Society of Automotive Engineers – організація професійних інженерів), Детройт, США; участь з експонатом «Qubit test synthesis for the black box functionality» у закордонній виставці при 22nd IEEE European Test Symposium, Limassol, CYPRUS, May 22-26, 2017; за участю студентів, магістрантів та молодих вчених на Міжнародний молодіжний форум «Радіоелектроніка та молодь у XXI столітті» 2015-2017 рр., Харків, Україна; DESSERT, Україна. Інноваційні проекти та розробки, презентації їх на виставках з отриманням призових місць.

# 1 МОДЕЛІ І МЕТОДИ МОНІТОРИНГУ ТА УПРАВЛІННЯ ТРАНСПОРТОМ

Пропонуються основні напрямки сталого розвитку кіберфізичного транспортного компютингу, пов'язані із засобами телекомунікацій, точного позиціонування, online е-картографії, диференціальної навігації, Internet of Car (е-інфраструктура автомобіля, доріг і хмарних сервісів). Представлені компоненти кіберфізической системы моніторингу та управління дорожнім рухом в реальному часу, яка грунтується на використанні глобальних систем позиціонування і навігації (GPS, GPRS), інтелектуальних дорожніх контролерів, засобів радіочастотної ідентифікації автомобілів та інфраструктури дорожнього руху.

## 1.1 Хмарний комп’ютинг і системи управління транспортом

У сучасному світі розвивається е-інфраструктура дорожнього руху в рамках порівняно нового поняття Internet of Car. В [1, 2, 3, 4, 5, 6, 7] представлена інформація про використання існуючих і майбутніх е-інфраструктур для поліпшення якості життя людей. Розробка е-інфраструктури для організації та проведення наукових досліджень є одним з ключових напрямків на ринку електронних технологій. Одним із прикладів застосування е-інфраструктури є науково-освітні заходи, які грунтуються на використанні хмарних або web-сервісів в режимі online. Крім того, е-інфраструктура є необхідною умовою в даний час для функціонування державних і соціальних служб, для створення розумних урядів, міст, будинків, а також моніторингу та управління транспортними потоками в режимі online. Крім того, електронна інфраструктура дає можливість створювати тимчасові творчі колективи, що включають фахівців з різних сфер діяльності, в тому числі, і з інших країн для вирішення складних науково-технічних проблем, що припускають наявність хмарних сервісів, а також розумних речей для збору, зберігання, аналізу великих даних,

реалізації сервісів моделювання на основі використання сучасних спеціалізованих логічних процесорів [8]. Одним із прикладів, що підтверджують значимість е-інфраструктури, є створення інституції, яка називається The European Strategy Forum on Research Infrastructures (ESFRI). Дана організація координує зусилля європейських університетів і компаній для створення стандартів і прототипів е-інфраструктур, спрямованих на забезпечення ефективного і квазі-оптимального функціонування кіберфізичних сервісів, пов’язаних з проблемами моніторингу та управління у всіх сферах людської діяльності. Сучасне уявлення е-інфраструктури в Європі включає мережу обчислювальних і інформаційних дата центрів, хмарні сервіси, гаджети аутентифікації, авторизації та інтерфейсні зв'язки з хмарами [9, 10]. Практика останніх 10 років свідчить про затребуваність е-інфраструктури і хмарних сервісів для більшої частини населення планети в режимі online 24/7 [11, 12, 13, 14, 15]. Хмарні сервіси [16] забезпечують легкий масштабований доступ до додатків, ресурсів і послуг, які повністю керуються провайдером хмарних сервісів. Властивість динамічного масштабування займає ключове значення для задоволення потреб користувачів і замовників [17], яке так само дозволяє істотно скоротити витрати на розгортання власних ресурсів і виділення персоналу для управління службою. У сучасному мобільному світі з високою пропускною здатністю і повсюдним доступом до мережі, користувачі можуть вільно співпрацювати через кордони, таким чином, у створюваному середовищі повинні бути чітко описані межі області доступу до даних і послуг для учасників різних приватних і державних організацій. Отже, довіру і ідентичність займає ключову позицію як окремий шар у е-інфраструктурі екосистеми. Федеративні авторизації і аутентифікації інфраструктури спрощують доступ до міжорганізаційних ресурсів, дозволяючи контролювати безпечний доступ [18] до різноманітно розташованих джерел, що дозволяє забезпечити користувачам надійне середовище зберігання цінних і цілісних даних без будь-якої шкоди. Даний шар з’єднує потужність обчислювальної

мережі та інфраструктури даних, створюючи відкритий, надійний і безпечний європейський простір.

В [19] представлено стратегічне бачення британської електронної інфраструктури, де е-інфраструктура – це об'єднання політичних, технологічних і адміністративних служб для легкого і економічно-ефективного, спільного використання розподілених електронних ресурсів. Е-інфраструктуру слід розглядати як живу екосистему, яка є відкритою і доступною для міжнародних співробітництв і постійно адаптується до мінливих вимог життя. Прориви у створенні окремих компонентів електронної інфраструктури повинні звільнити вчених від складних і відволікаючих бізнес-обчислень. Слід зазначити, що для благополучного розвитку досліджень в цій області [20] важлива відкрита конкуренція, співробітництво і технологічні інновації. Е-інфраструктура може бути встановлена тільки шляхом спільного і дійсно загального стратегічного плану зусиль між користувачами і організаціями, що беруть участь в розробці і експлуатації електронних інфраструктур. Досягнення значного прогресу на шляху до розвитку е-інфраструктури здійснюється за рахунок інвестування: у 2011 році британський уряд виділив 160 млн £ в області високопродуктивних обчислень і мереж; у 2012 році 189 млн £ для роботи з великими обсягами даних і енергоефективних обчислень. Таким чином, е-інфраструктура в Великобританії сьогодні є складною взаємодією сервісів, електронних пристроїв, мереж, послуг передачі даних, кібербезпеки і навичок людей, які створюють роботу електронної інфраструктури. Як показує практика, здатність генерувати, збирати і обробляти великі дані для експлуатації в промисловості, у наукових колах і державних установах дає реальну можливість підвищити конкурентоспроможність і зростання економіки кожної держави. Е-інфраструктура має вирішальне значення для інновацій в цілому ряді галузей. Наприклад, вона відіграє важливе значення повного використання можливостей і економії ресурсів [21], які приносять технологічні розробки. Так само займає важливе значення в екосистемі ресурсів, що дозволяє виконувати великомасштабне

моделювання, аналіз і візуалізацію великих даних. E-інфраструктура відноситься до екосистеми ресурсів, яка здійснює розподілене співробітництво, великомасштабне моделювання, обчислення та аналіз великих даних, містить розробку та використання передових інструментів для використання потенціалу наявних даних і ресурсів (рис. 1.1).

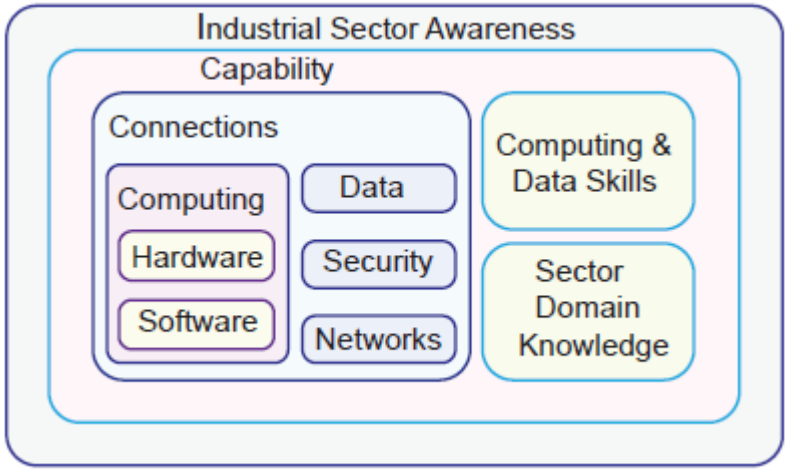



Рисунок 1.1 – Подання компонентів E-інфраструктури

В [22] розглядається історія розвитку систем управління міським транспортом і описується використання комп'ютерних систем для управління трафіком. Показано, що підвищення продуктивності комп'ютерів, поява LAN-мереж і хмарних обчислень (cloud computing) дозволили удосконалити технології управління, що пройшли шлях розвитку від централізованих моделей до розподілених систем на основі мобільних гаджетів і хмарних технологій. Запропоновано систему PtMS (parallel transportation management systems), ядром якої є «Artificial transportation system». Тут паралельно з діючою транспортною системою використовується віртуальне середовище, яке містить бази даних зі стандартними шаблонами управління трафіком, агентами і сценаріями руху. Як мобільні сенсори виступають будь-які детектори трафіку. Суть дослідження зводиться до інтелектуальної хмарної обробки дорожніх ситуацій, які відслідковуються дорожніми сенсорами. Актюаторні впливи з управління трафіком приймаються

на основі заздалегідь прописаних алгоритмів і стратегій. У разі відсутності шаблону рішення складне завдання декомпозується на більш прості. Переваги. Данний підхід "local simple, remote complex", запропонований в статті, орієнтований на інтелектуальну організацію управління транспортом в рамках Smart Cloud Traffic Control. Запропонована PtMS є складовою частиною системи «Adapts» (Adaptive Platforms for Transportation Systems), яка складається з 3 рівнів: організації, координації та виконання завдань по обробці дорожніх ситуацій. Організаційний рівень представлений 4 функціями: декомпозиція задачі, планування, інкапсуляція і управління. Це дає можливість істотно підвищити якість прийнятих регуляторних впливів за рахунок ускладнення алгоритмів. Недоліки. За певний проміжок часу можна обробити обмежену кількість перехресть. Для запуску і тестування системи на сервері накладаються істотні обмеження за обчислювальною потужністю і пам'яттю. Такі обмеження не відповідають реальним дорожнім умовам у великих містах, тому система повинна використовувати кілька серверів для online вирішення завдань управління транспортом на великій кількості перехресть. Для зберігання даних про трафік і дорожніх ситуаціях необхідні дата центри. У статті запропоновано два рішення даної проблеми: обладнати центри суперкомп'ютерами [23], або використовувати технології хмарних обчислень, такі як Google's Map-Reduce, IBM Blue Cloud і Amazon.

В [24] розглядається стандарт IEEE 802.21, який підтримує різні типи систем бездротового зв’язку, такі як сімейство IEEE 802.11, IEEE 802.16 (WiMAX) та технології стільникової передачі даних, розроблені групами 3GPP і 3GPP2. IEEE 802.21 надає незалежну передачу – Media Independent Handover (MIH) даних в стільникових мережах. Фреймворк IEEE 802.21 складається з компонентів: функція MIH (MIHF) і Service Access Points (SAP). Функція MIH є модулем, який розробляється для того, щоб надати теоретичні сервіси на більш високих рівнях за допомогою незалежного медіа-інтерфейсу. MIHF складається з трьох основних сервісів: сервіс медіа-незалежних подій (MIES), сервіс медіа-незалежних команд (MICS), сервіс

медіа-незалежної інформації (MIIS). Стандартні сервісні точки доступу та сервісні примітиви дозволяють додаткам MIHF мати загальний доступ до різнорідних медіа-шарів. Пропонується система FANSCI (fast handover scheme for car-to-car communication), призначена для адаптації стандарту IEEE 802.21 передачі даних, необхідних для безперервного і безперебійного зв'язку в гетерогенних мережах. FANSCI складається з двох компонентів: вишка (встановлюється в центрі управління мережею) та автомобільний компонент (встановлюється в автомобілі), реалізація FANSCI продемонстрована на рис. 1.2.

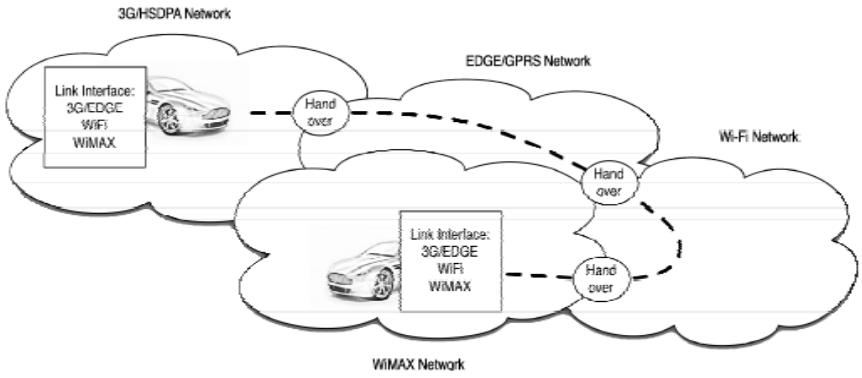



Рисунок 1.2 – Реалізація FANSCI

Автомобільний компонент містить: комунікаційний модуль Car-to-Car (що складається з менеджера з’єднання, колектора GPS-інформації, колектора даних з сенсорів автомобіля, колектора інформації про двигун) і модуль обробки даних на стороні клієнта (що складається з менеджера ініціації зв’язку, трекера сигналу, буферного комп’ютера і модуля передачі даних). Компонент вишки включає: комунікаційний модуль Car-to-Car (що складається з обробника трафіку, менеджера аутентифікації, менеджера маршрутизації) і пропонований модуль для передачі даних (що складається з менеджера запитів на передачу даних, функціональний MIH модуль, буфер трафіку і редиректор трафіку). З технологічної точки зору, технологія FANSCI може бути імплементована у будь-які бездротові мережі, оскільки вона заснована на стандарті IEEE 802.21, який базується на IEEE 802.11 (Wi-Fi) і IEEE 802.16 (WiMAX). Водій може постійно отримувати інформацію

про пробки, дорожню ситуацію, інші машини, що рухаються по перехрестю, несправні двигуни автомобілів, що їдуть попереду, та інших, що знаходяться поблизу, а також про доступні гетерогенні мережі. Впровадження технології FANSCI для здійснення комунікації між автомобілями сприяє створенню безпілотних автомобілів.

В [25] демонструється хмарний контроль для прийняття відповідних заходів щодо уникнення небезпечного водіння. Автономні транспортні засоби, що рухаються у міському середовищі, обробляють інформацію і приймають рішення, які стосуються руху в режимі «stop-and-go», відповідно до дорожніх знаків і правил, обгону інших ТЗ, зміни смуги руху, входу і виходу з потоку трафіку. Автономні ТЗ обладнані різними типами радарів, інерційною навігацією, системою GPS, компасом і системами управління, які як правило, коштують набагато дорожче самих ТЗ. Запропоновано новий control alogrithm-cloud, якому властива хороша продуктивність і гнучкість. Для створення системи автономного водіння вирішуються три завдання: 1) самостійне керування автомобілем, яке містить старт, прискорення, уповільнення руху, різики або плавні повороти; 2) рух автомобіля уздовж дороги або по центру; 3) безпечне водіння у трафіку, яке вимагає механізму визначення перешкод як для водія, так і для пішохода. Програмна архітектура складається з чотирьох частин: інтерфейс сенсорів для дороги, збір інформації про смугу руху і перешкоди, планувальник маршруту та інтерфейс ТЗ. Апаратна архітектура так само складається з чотирьох частин: Sensor Layer, Preprocess Layer, Integration Layer and Vehicle Layer. Для запобігання зіткнень автоматично керовані автомобілі повинні вміти розпізнавати ситуацію на смузі проїзду, а також на сусідніх смугах так, як це робить водій для безпечного водіння. Для цього в хмарному алгоритмі управління (control alogrithm-cloud) використовується метод MOR (My Own Range). У нормальних умовах, для отримання MOR, необхідно розрахувати кут і відстань між машиною і перешкодами, які виявляються за допомогою ультразвукових датчиків і камер. Однак, є особливі випадки, наприклад, коли машина, що їде по сусідній

смузі, має намір здійснити зміну смуги, включивши сигнал повороту. У таких ситуаціях необхідно зробити деяке прогнозування. Його можна здійснити, відштовхуючись від дистанції і кута повороту попереду автомобіля, що рухається. Наміри про подальший рух також можна визначити за рахунок режимів водіння, які в статті поділяють на 4 основних: 1) нормальне водіння, 2) уповільнення і гальмування, 3) обгін з лівої смуги, 4) обгін з правої смуги. Обчислюються коректні команди для прискорення, гальмування і рульового управління. Це завдання виконується за допомогою контролерів швидкості і рульового управління, грунтуючись на правилах.

## 1.2 Моделі міських інфраструктур дорожнього руху

В [26] пропонується інноваційний механізм паралельного мережевого управління трафіком для хмарних обчислень, заснований на планувальнику пакетів HTB (Hierarchical Token Buckets). Відмінністю між хмарою і традиційними центрами даних є масштаб трафіку даних і одночасний доступ користувачів. Хмара має гарантувати доступ до всіх ресурсів і послуг, за допомогою налаштованих SLA. Однак в хмарі можуть бути сотні тисяч користувачів, які одночасно звертаються до ресурсу, що створює надзвичайно сильний тиск на хмару. Для використання хмарних технологій потрібні великі смуги пропускання, однак на сьогоднішній день за допомогою HTB можна досягти лише швидкостей 0,5 Gbps, що унеможливлює застосування даного методу в хмарах. Для вирішення даної проблеми можна використовувати механізми програмного і апаратного розпаралелювання оброблюваних даних на багатоядерних процесорах, за умови використання багаторівневої мережевої структури і модифікації послідовного коду для паралельної обробки. Використовується технологія паралелізації багатоядерних процесорів FIFO без блокування. Результати моделювання підтверджують, що паралельна пропускна здатність HTB, що заснований на FIFO без блокування, може легко перевищити швидкість 1 Гбіт/с і досягти максимальної швидкості 2 Гбіт/с, збільшивши продуктивність до 300% у порівнянні з традиційним HTB. Дана

технологія розроблялася тільки під Linux, завдання роботи з іншими операційними системами не розглядалися.

В [27] представлено підхід управління світлофором за допомогою агента, який ефективно керує рухом відповідно до його поточного стану. У розглянутій дорожній інфраструктурі кожен агент контролює світлофор на перехресті по циклу «спостереження – рішення – дія»: агент спостерігає за поточним станом руху у вузлі, потім використовує отриману інформацію для порівняння з правилами руху та ефективного управління потоком. Значно скорочено середній час затримки кожного автомобіля на світлофорах в порівнянні з іншими підходами. Щоб спростити завдання управління рухом, світ реальних агентів замінюється віртуальним симулятором руху. Агент, запропонований в роботі для управління світлофором, складається з трьох компонентів: спостерігача руху, бази знань і механізму виведення.

Багато дослідників намагаються модернізувати поточну систему трафіку з фіксованими часовими затримками світлофора в інтелектуальну систему з часовою затримкою в залежності від частоти трафіку. Були створені математичні моделі для світлофорів, а потім класичні алгоритми і ідеї, які використовувалися для управління світловим сигналом. Винахід [28] включає мікропроцесор, пристрій ручного введення, примусовий комутаційний пристрій і інтелектуальний пристрій виявлення. Система може автоматично регулювати параметри управління світлофором відповідно до змін в потоці транспорту у різних напрямках, підвищуючи ефективність руху на перехрестях. У роботі [29] представлена нейронна мережа (НС) є основою для законів управління, при цьому оцінка ваги НС, яка відбувається в режимі реального часу у замкнутому режимі, здійснюється за допомогою алгоритму стохастичної апроксимації одночасного збудження станів. Процедура навчання може тривати місяці або навіть роки. Результати моделювання показали 10%-ве скорочення часу очікування транспортних засобів. Були розміщені 2 електромагнітних датчики на кожній смузі для підрахунку кількості транспортних засобів. Запропоновано контролер нечіткої логіки,

який відповідає за тривалість зеленого світла відповідно до умов трафіку, і кінцевий автомат для управління послідовністю станів, через який повинен циклічно проходити контролер нечіткого трафіку.

У [30, 31] запропоновано новий підхід до точного прогнозу трафіку на перехресті за допомогою технології бездротових сенсорних мереж, що включають взаємодію розумних вузлів: управління, індикатори та транспортних засобів.

В [32-34] запропоновано інтегральний контролер швидкості і алгоритм управління перевантаженням, призначений для інтернету, для вирішення проблеми перевантаженості на сюжеті ділянці дороги складної дорожньої мережі. Запропоновано два типи пристроїв, встановлених уздовж дороги, на в'їзді і виїзді сегмента дороги для контролю обсягу трафіку і світлофорів. При виборі стану світлофора пристрої виконують «опитування» попередньо станів трафіку і динамічно встановлюють періоди часу для сигналів світлофора в залежності від умов потоку, що прибуває. Враховується також зворотний зв'язок від умов убування потоку. Досвід останніх років свідчить про те, що конструювання традиційних дорожніх інфраструктур і використання виключно традиційних підходів до управління не тільки коштовно, але і неефективно вирішує питання транспортних перевантажень. Аналіз показав, що однією з важливих причин заторів на дорогах є неефективне управління транспортним потоком. Стан світлофора має динамічно підлаштовуватися під умови трафіку в режимі реального часу.

Інтелектуальна система управління рухом [35-36] використовує ревлізацію двопроцессорної інтелектуальної системи в FPGA для відстеження частоти трафіку. Існують різні способи контролю трафіку, наприклад, обробка відеозображення [37], підрахунок транспортних засобів через пересічні масиви датчиків наближення – торкання. Нова структура управління трафіком – мобільна інтелектуальна система управління трафіком (MITCS), розроблена для Тайваню, запропонована у [38]. Аналогічним чином, інтелектуальна система управління трафіком працює у Пекіні під назвою SCOOT

[39]. Автори прагнуть створити інтелектуальну систему, яка розуміє транспортний рух і сигнальну систему руху, яка сприймає трафік, використовуючи набір безконтактних датчиків і сенсорних датчиків, встановлених на дорозі, обчислює щільність, використовуючи код, вбудований в мікроконтролер, і застосовує зміну On-The-Go. Ця система змінює затримки, що застосовуються в поточному робочому циклі світлофора, для максимального комфорту учасників дорожнього руху. Візуалізація здійснюється за допомогою світлодіодів високої інтенсивності.

В [40] представлена система управління трафіком, де навантаження дорожнього потоку безперервно вимірюється датчиками, підключеними до системи на базі мікроконтролерів. Світлофори пов'язані між собою мережею, через яку відбувається обмін інформацією про навантаження і синхронізації. Тривалість кожного циклу світлофора змінюється динамічно. Структура системи складається з трьох модулів. У першому частота трафіку визначається датчиками наближення або сенсорними датчиками на основі магнітного захоплення. У другому модулі запрограмований вбудований процесор визначає затримки сигналів світлофора відповідно до частоти трафіку. У третьому модулі вихідні дані від різних портів мікроконтролера надходять на різні світлофори. Інтелектуальна система управління трафіком, показана на рис. 1.3, працює наступним чином: 1) Датчики наближення посилають імпульс кожен раз, коли автомобіль наближається до нього. 2) Імпульс посилюється і перетворюється в цифровий імпульс. 3) Вихід подається на мікроконтролер. 4) Відповідно до кількості активованих датчиків визначається час зупинки і переходу. 5) Зміна часу здійснюється в одному циклі.

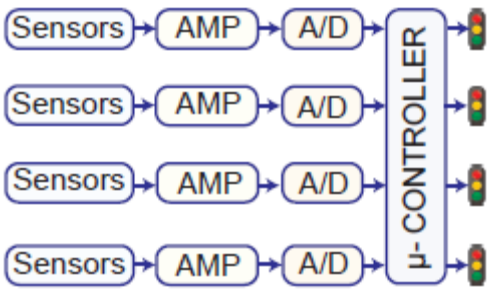



Рисунок 1.3 – Інтерфейс датчика з мікро контролером

У цьому недорогому прототипі були використані малогабаритні датчики наближення і світлодіоди для терміналів світлофора. Але в реальності порт мікроконтролера не може управляти високопотужними світловими лампами. Це можна зробити за допомогою реле, яке переключається через транзистор. Число пристроїв виявлення трафіку зі світлофорами може бути збільшено за рахунок використання чіпа інтерфейсу 8255 PPI з портом мікроконтролера. Перевагою даної системи є низька вартість – 400 доларів США і висока точність, в той час як існуючі системи стоять понад 30 000 доларів.

В [41-43] запропоновано управління сигналами світлофора за рахунок обчислення параметрів нечіткого управління трафіком нейромережі. В [44, 45] використовуються інтелектуальні агенти з можливістю Q-learning для управління одним перетином. Результати моделювання показали, що ці підходи краще, ніж метод управління з фіксованим часом. Показана самонавчальна модель управління світлофором, яка заснована на нечіткій кластеризації і генетичному алгоритмі. Використовуються теорія нечіткої логіки і машинне навчання. Шляхом нечіткої кластеризації схеми управління поміщають числа прибуваючих автомобілів в базу даниху вигляді набору правил. Вони визначають управління сигналом в певних дорожніх ситуаціях відповідно до досвіду і принципами управління трафіком. Генетичний алгоритм використовується для поліпшення набору змінних правил і є частиною процесу управління сигналами, виконуючи роль розрахунку оптимального управління. В [46] представлена скоординована система управління дорожнім рухом на основі мультіагента. Вона має ту ж структуру, що і модель в [47-49], яка являє собою систему управління стилем піраміди з трьома рівнями, які представляють собою глобальне, групове та індивідуальне планування зверху вниз.

В [50] реалізована розподілена і скоординована система управління сигналами руху, яка ґрунтується на мультиагентах (DTCMAS) без центру управління. Кожне перехрестя контролюється локальним агентом незалежно

один від одного, проте зв'язок між ними існує, і вона відбувається у суміжних перехрестях. Таким чином, якщо масштаб дорожньої мережі збільшується, складність алгоритму управління залишається незмінною. Внутрішня структура агента показана на рис. 1.4. Кожен агент управління сигналами складається з частин: інтерфейс зв’язку, процесори сусідніх з’єднань, локального з’єднання і виявлення автомобіля.

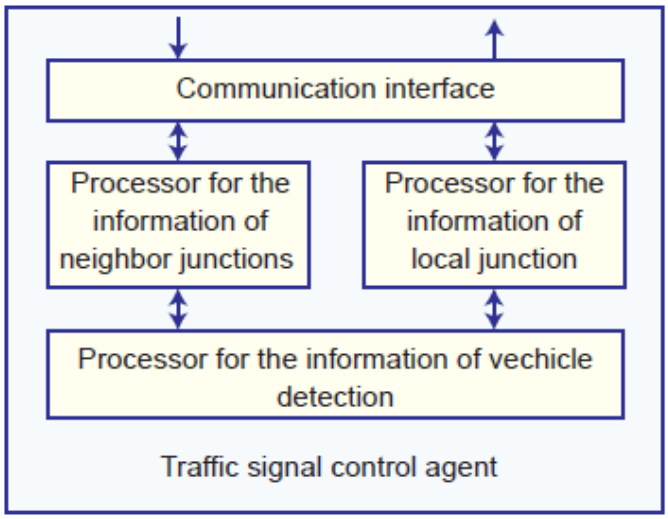



Рисунок 1.4 – Структура агента управління сигналами.

Оскільки кожен агент містить базу даних і механізм міркувань, DTCMAS не тільки повністю враховує інформацію про стан руху на перехрестях, але також виконує глобальну оптимізацію для управління рухом дорожньої мережі і зменшує складність алгоритму управління. Результат моделювання показує, що підхід управління сигналами DTCMAS зменшує загальну затримку зупинки автомобілів в середньому на 37,8% у порівнянні з методами фіксованого часу і знижує загальну затримку автомобілів у середньому на 17,8%, в порівнянні з іншими методами за тих же умов.

## 1.3 Системи позиціювання та навігації транспортних засобів

В [51] розглядається проблема зростання кількості автомобілів в місті, а також невідповідність пропускної здатності доріг такому інтенсивному росту. Пропонується використовувати системи динамічної (DVNS) і статичної (SVNS) навігації [52] для моніторингу та управління

транспортними засобами. Дорожня мережа і архітектура динамічного трафіку з використанням просторово-часової моделі даних може бути описана за допомогою технологій: 1) зіставлення і злиття простору мульти-модального дорожнього трафіку і атрибутів інформації; 2) оновлення інформації про дорожню мережу для різних проміжків часу; 3) створення бази даних масштабованої дорожньої мережі. Система DVNS складається з трьох рівнів: базових і топологічних мереж, а також інформації про дорожній рух. Уявити функціонування дорожньої мережі можна, грунтуючись на вузлах або на сторонах руху. У запропонованому підході позиціонування [53, 54] транспортних засобів здійснюється шляхом використання GPS-сигналу, а у випадках щільної міської забудови, тунелів застосовується фільтр Кальмана [55]. Показані методології та моделі моніторингу дорожніх мереж, принципи управління ними на основі просторово-часових критеріїв, а також короткострокове прогнозування дорожньої ситуації з метою створення високоефективної масштабованої мульти-модальної навігаційної системи. Запропонований підхід дозволяє вирішити завдання збору і обробки інформації про дорожній рух, що включає потоки трафіку в режимі реального часу, дорожні правила, а також інформацію про дорожні пригоди з метою подальшого управління об’єктами дорожньої інфраструктури. Використання тільки однієї технології бездротового зв’язку для визначення місця розташування транспортного засобу скорочує область використання даної системи до зони покриття GPS [56, 57]. Хмарні сервіси не розглядаються. В [58] запропоновано систему динамічної навігації, представлена її архітектура і методика організації та управління даними. Традиційні системи навігації [59] транспорту мають недоліки: 1) Відсутність актуальної інформації про дорожню ситуацію і про трафік в режимі реального часу; 2) Навігація без урахування змін у трафіку і інформації дорожньої ситуації; 3) Відсутність можливості для з'єднання з іншими мережами обслуговування. Система динамічної навігації складається з трьох

частин (рис. 1.5): Traffic information Provider, Service Center і Navigation Terminal.

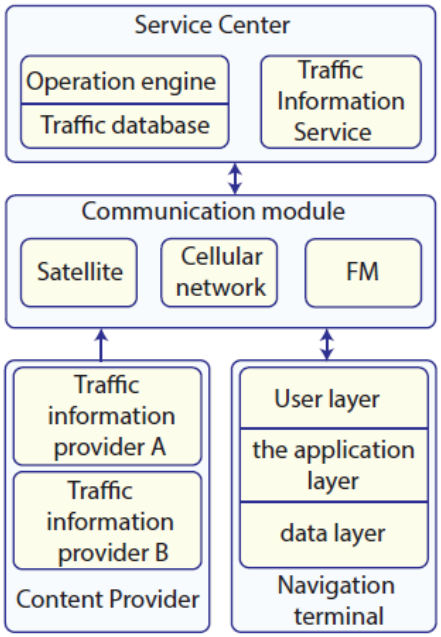



Рисунок 1.5 – Системна архітектура

Traffic information Provider – основна служба дорожнього руху в Китаї, яка отримує інформацію про дорожній трафік з відеодетектора, мікрохвильових детекторів, систем ідентифікації автомобільних прав і системи моніторингу проїзду на червоне світло. Так само використовуються транспортні засоби, які при русі в потоці збирають дані за допомогою бездротових мереж в режимі реального часу з метою подальшої обробки масивів даних в сервісному центрі. Service Center – розподіл і обробка всієї навігаційної інформації, яка складається з таких компонентів: operation engine (ПЗ проміжного шару, яке надає сукупність інтерфейсів для перетворення інформації в уніфікованому вигляді і зберігання її в базі даних), база даних трафіку (використовується на додаток до інформації про real-time трафіку для надання рекомендованого оптимального маршруту і очікуваного часу прибуття з більшим ступенем точності), служба дорожньої інформації (надає всі види сервісних модулів: сервіс обмеження трафіку, сервіс пробок і сервіс погоди, новинний і сервіс розваг) і комунікаційний модуль (забезпечує зв’язок між сервісним центром, контент-провайдером і

мобільними терміналами). Navigation terminal – здійснює функції відображення карти, рекомендації маршруту, планування маршруту і пошук. Термінали масштабуються, складаються з комунікаційного модуля, рівня даних, додатку і виконання. В системі GIS-T (Geographic Information System for Transportation) навігаційні дані складаються з статичних даних, не змінних на протязі тривалого часу (існуючі дороги, сервісні станції та заправні станції) і динамічних (потік руху, дорожні події). Просторові характеристики показують зміни розподілу завантажених ділянок дороги в різний час. Часові характеристики показують, що стан завантаженості на одній і тій же дорозі змінюється з плином часу. Запропонована система орієнтована на збір актуальної інформації про дорожню ситуацію і про трафік в режимі реального часу; на створення навігації з урахуванням змін трафіку і інформації про дорожню ситуацію; можливості забезпечення з'єднання з іншими мережами обслуговування. Експериментальна система, що базується на методиці запропонованої в даній статті, вже імплементована в урядовій організації збору та надання інформації про трафік в Пекіні. Багато розвинених країн займаються розробкою систем позиціонування (VNS) – "VICS" в Японії, "TravTek" в США і "Ali-Scout" в Європі. Недоліки. Через проблеми урбанізації важко реалізовувати системи VNS з real-time інформацією про трафік. У запропонованій системі не організовано управління інтеграцією real-time інформації з дорожніми мережами. В [60] пропонується система автомобільної навігації, заснована на гібридному генетичному алгоритмі. Обчислення маршруту – задача комбінаторної оптимізації, поряд із завданням "мандрівного торговця". Для її вирішення використовується генетичний алгоритм (ГА) [61, 62], як потужний засіб оптимізації. Логіка роботи полягає в пошуку маршруту з декількома точками проїзду до прибуття в кінцевий пункт призначення. Завдання вибору маршруту розподіляється на 2 підзадачі: вибір точок проїзду (вирішується за допомогою ГА), вибір між цими точками (вирішується за допомогою алгоритму Дейкстри). Надається можливість визначити оптимальний спосіб

побудови маршруту, виконуючи найбільш важливу функцію для системи автомобільної навігації. Алгоритми Дейкстри і Мазе визначають найкоротший маршрут між двома точками, проте не вирішують задачу з кількома точками.

В [63] пропонується модель руху зі змінною швидкістю, ядром якого є контролер, заснований на хмарній нейронній мережі для здійснення рульового управління. В рамках проекту «Інтелектуальна система транспортування» було розроблено сукупність підходів моделювання трафіку, які вирішують завдання, пов’язані з пробками і дорожньо-транспортними пригодами. Всі підходи засновані на системі Equation Based Modeling (EBM). Незважаючи на успіхи в моделюванні мережі реального трафіку, досі не були емульовані складні процеси поведінки водія на дорогах [64]. Моделювання дій водія –складне завдання, що включає теорію управління, робототехніку та психологію. Симуляція рульового керування водієм є ключовою технологією в автономних системах руху. При виконанні алгоритму навчання нечітка область виражається системою лінійних рівнянь для локальної апроксимації нелінійної функції. Це відображає накопичений досвід водія з такими ж умовами водіння. Отриманий в результаті моделювання сценарій зміни смуги демонструє ефективність алгоритму навчання, заснованого на хмарній моделі і нейронних мережах. Запропонована модель дозволяє частково здійснювати "driver-free car control", з огляду на рух попереду і ззаду транспортного засобу, що рухається.

В [65] описується проект AutoNomos – управління пробками на однополосній дорозі. Динамічно змінювана система, як дорожня пробка, складається з множини постійно змінюваних об'єктів, наприклад, машин, розташованих в різних місцях, що продовжують рухатися в кінці або на початку пробки. Для отримання корисної інформації, що співвідноситься з кінцем затору, потрібно враховувати зміщення одного автомобіля по відношенню до іншого, разом з усіма відповідними даними. Подальший

розвиток проекту направлено на інтелектуальну організацію управління трафіком на двополосному шосе (з перехрестями і різними виїздами), а також на вирішення проблеми «пляшкового горлечка» (вузьких місць) в трафіку, наприклад, викликані конвоєм, повільними вантажівками, аваріями або спеціалізованими транспортними засобами.

В [66] пропонується рішення ключової проблеми міського руху – інтелектуальна модель управління міською магістраллю. Показується вплив контролю швидкості на потік трафіку; розробляється стратегія попереднього контролю для забезпечення максимальної ефективності системи міських магістралей, особливо в умовах завантаженості; представлена система хмарного прогнозування [67] пробок на дорогах, яка дозволяє реалізувати стратегію випереджувального керування транспортним засобом. Заропонована інтегрована розумна динамічно керована архітектура (рис. 1.6) складається з 3 підсистем: 1) вимірювання даних про трафік на магістралі; 2) управління інформацією про трафік; 3) попереднього контролю швидкості для поліпшення рівня ефективності та безпеки дорожнього руху.

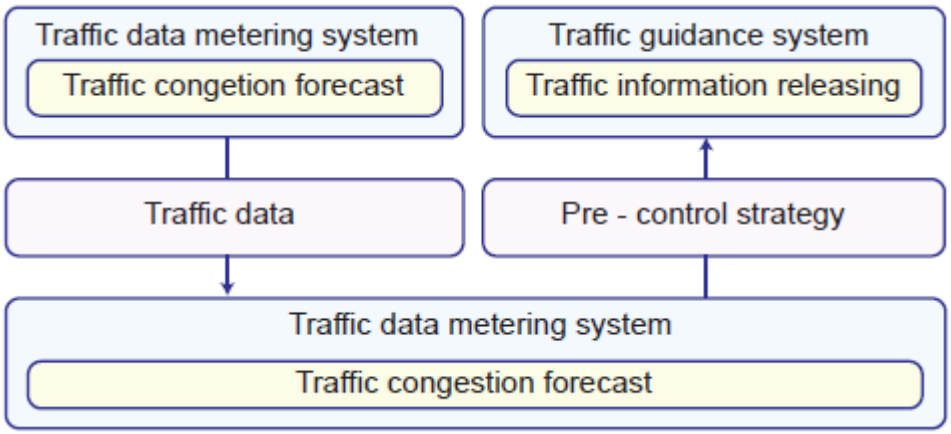



Рисунок 1.6 – Архітектура інтелектуальної системи управління

Зниження швидкості призводить до зменшення заторів у «вузьких місцях», що може поліпшити загальну продуктивність магістральної системи. «Вузьке місце» в даному контексті означає не тільки те місце, яке має більш низьку пропускну здатність трафіку [68], ніж інші частини магістралі, а й місце, в якому деякі затори будуть відбуватися на основі прогнозу заторів на

дорогах. У пропонованому рішенні не вказується застосування хмарних технологій.

В [69] розглядається система адаптивного управління дорожнім сигналом (TSC), як продовження розвитку інтелектуальної транспортної системи (ITC). Досліджується система спостереження на основі аналізу непередбачуваних і важко вимірюваних порушень, що впливають на дорожню ситуацію. Використано методології відео-спостережень, які включають сегментацію, класифікацію і відстеження об'єктів для визначення стану міської магістралі в режимі реального часу. Алгоритм управління адаптивним сигналом дорожнього руху погоджує перемикання світлофорів в ситуаціях «руху по прямій», «повороту направо» і «повороту наліво». Ефективне, безпечне і менш забруднюване транспортування людей і товарів вимагає оптимального використання інфраструктури за допомогою застосування різних заходів контролю дорожнього руху. Керування транспортними засобами безпосередньо залежить від ефективності та актуальності застосовуваних методологій управління [70, 71, 72, 73, 74, 75, 76, 77, 78, 79]. Для цього визначаються і аналізуються об'єкти, потім вимірюється кількість автомобілів, після чого інформація про транспорт екстраполюється. Існують три основні методики виявлення транспортних засобів на магістралі: «часова відмінність», «оптичний потік» і «віднімання фону». Методи часового диференціювання і оптичного потоку дозволяють обробляти зображення і знаходити рухомі об'єкти, але вони не підходять для нерухомих або повільно рухомих об'єктів. Результати експерименту з використання системи управління сигналами [70-80] дорожнього руху підтверджують ефективність адаптивного підходу TSC на основі відеоспостереження.

Метод S-TRAC (system-wide traffic-adaptive control) [81] був введений як засіб для створення оптимальних таймінгів для сигналів у режимі реального часу. S-TRAC має кілька особливостей, які роблять його практично здійсненним і теоретично обгрунтованим у вирішенні проблем управління транспортом. Серед цих особливостей: (1) не потребує ніякої моделі потоку

трафіку в масштабі всієї системи; (2) S-TRAC автоматично адаптується до довгострокових змін у системі (наприклад, сезонних змін), забезпечуючи при цьому команди реагування в режимі реального часу; (3) S-TRAC може працювати з існуючими апаратними засобами і конфігураціями датчиків в використаній мережі.

## 1.4 Електронне обладнання сучасного автомобіля

Тенденції розвитку конструкцій автомобілів обумовлені економічними, соціальними і екологічними причинами: підвищення паливної економічності і безпеки автомобілів, забруднення навколишнього середовища відпрацьованими. Автомобільні технології набирають темпи розвитку, завдяки комп'ютеризації та впровадження «smart» компонентів в бортові системи автомобілів. Фінансові та енергетичні кризи також спричинили за собою зростання інтересу і наукові дослідження в області створення електромобілів і гібридних автомобілів, що використовують енергію з відновлюваних джерел. Збільшення обсягів трафіку у великих мегаполісах призводить до ідеї використання оцифрованого трафіку і систем моніторингу та управління ним, а також постійні дорожні пробки та часткові випадки ДТП вимушені прийняти концепцію «free human control car» і максимально автоматизувати процес управління рухом транспортних засобів [82-84]. Розробкою і моделюванням автомобілів майбутнього цікавляться і ведуть дослідження такі компанії як: Apple, Google, Tesla, Synopsys, світові автогіганти BMW, VW, Mercedes, Toyota, Ford, Honda. Існує багато наукових центрів з розвитку інфраструктури автомобілів, в яких співпрацюють університети і компанії, наприклад, центр автомобільних досліджень (CAR) в Коледжі інженерії штату Огайо [85]. Hitachi automotive systems group пропонує різні рішення з питань розвитку сучасного автомобіля [86]: екологічно чисті високоефективні системи управління двигунами для скорочення викидів вуглекислого газу і запобігання забруднення повітря; поставка інтегрованого електромеханічного рішення прокладання дороги для електроустаткування з механічним і електронним

управлінням; послуги хмарної інформаційної мережі, багатофункціональні навігаційні системи та інше інформаційне обладнання. Компанія надає широкий вибір сенсорів [87]: Airflow Sensor (slot-in, robust and silicon types), Differential Pressure Sensor, Exhaust Pressure Sensor, Pressure Sensor, Hall Effect Type Revolution Sensor. Глобальна мережа заводів, магазинів розташована по всьому світу: Європа, Азія, Китай, Америка.

Автомобіль майбутнього оснащений бортовим комп'ютером, сенсорами [88], актюатором [89], радарами [90], лидара [91], системою GPS [92], супутникові антени [93], системою безпеки. Архітектура транспортного засобу (рис. 1.7) дозволяє взаємодіяти з хмарними технологіями і використовувати всілякі сервіси [94] для поліпшення якості пересування.

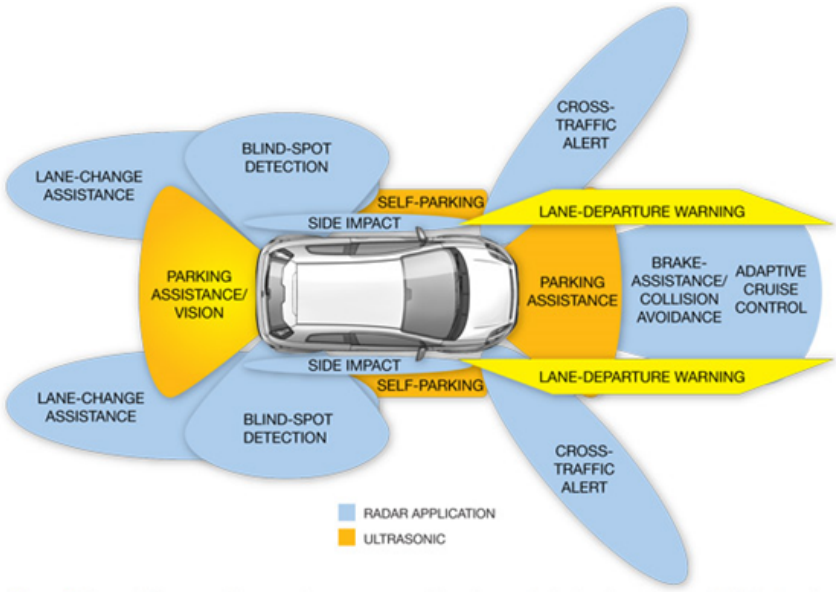



Рисунок 1.7 – Використання радар технологій

Вбудовані Satellite системи дозволяють здійснювати бездротовий зв'язок автомобіля з хмарними сервісами, об'єктами інтелектуальної дорожньої інфраструктури і зв'язку з мобільним пристроєм водія, а також іншими транспортними засобами в потоці в режимі реального часу. Використання сенсорів і актюаторів дозволяє здійснювати збір даних і моніторинг параметрів, а також здійснювати на підставі цього управління всіма аналоговими і цифровими системами автомобіля. Вбудований бортовий комп'ютер відповідає за обробку даних, одержуваних від датчиків і різних систем авто-

мобіля, призначених для бездротового зв'язку з хмарними сервісами і дорожньою інфраструктурою. Взаємодія машин один з одним відбувається за рахунок встановлення контакту шляхом зв'язку з хмарою, або шляхом використання каналів і протоколів зв'язку на ближній дистанції [95]. У разі викрадення машини власник отримує унікальну можливість повноцінного віддаленого спостереження за місцем розташування автомобіля, використовуючи GPS, вбудовані бортові камери і камери зовнішнього огляду автомобіля, а також шляхом отримання на мобільний телефон або інший пристрій SMS, e-mail або іншого виду повідомлень з даними про останньому зафіксованому в системі розташування його автомобіля. Спеціальним службам дорожнього контролю, поліції буде посланий сигнал про несанкціонований доступ до автомобіля, передані координати і параметри автомобіля. Як відповідна реакція може бути повне блокування ТЗ аж до дистанційного блокування двигуна, КПП і гальмівної системи. З розвитком обладнання автомобілів (рис. 1.8) змінюються і удосконалюються елементи дорожньої інфраструктури: розумні світлофори, віртуальні дорожні карти, знаки дорожнього руху. Сучасні світлофори стають інтелектуальним засобом взаємодії транспортних засобів з хмарним сервісом, будучи приймачем сигналів, які надходять від автомобілів через хмару [96].




Рисунок 1.8 – Сучасний автомобіль

Таким чином, компьютингове оснащення сучасного автомобіля робить його все більш високоінтелектуальним роботом, що розумно взаємодіє з навколишнім середовищем і хмарою за допомогою сенсорів і актюаторів при його точному позиціонуванні в просторі за рахунок каналів позиціонування і навігації.

## 1.5 Кіберфізичні системи хмарного управління транспортом

Процеси несинхронізованого розвитку двох компонентів: реального світу і кіберпростору в даний час набувають векторний формат, спрямований на створення структурованих і взаємно інтегрованих компонентів КіберГео-Системи (КГС). Остання еволюціонує шляхами створення в кіберпросторі хмарних сервісів точного моніторингу та оптимального управління недосконалим реальним світом на основі моделей взаємно однозначного відображення всіх земних процесів і явищ. Спрощено КГС можна представити двома модулями: «хмарного» кібер управління і технічного земного виконання, представленого ресурсами, засобами і людьми. Механізми управління і виконання не повинні перетинатися з матеріальних і людських ресурсів. Формально КГС представляється у вигляді двох механізмів: Cyberity і Humanity, які пов'язані між собою чотирма сигналами: Моніторингу, Управління та Ініціювання обох механізмів з метою реалізації бажань. Аналітична форма завдання СН-системи і її структурний еквівалент зображені на рис. 1.9.

$$S = (C, H, M, U, X, R, Y, P),$$
$$C \cap H = \varnothing;$$
$$\begin{cases} Y(t) = f[(X(t), (R(t), M(t), Y(t-1)]; \\ P(t) = g[X(t), (R(t), U(t), Y(t)]. \end{cases}$$

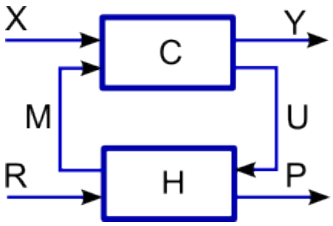



Рисунок 1.9 – Аналітична і структурна форми завдання СН-системи

Тут представлені $(C, H, M, U, X, R, Y, P)$ відповідно: блоки управління (менеджери і кібери – cybers) і виконання (інфраструктура, виконавці та

роботи); сигнали моніторингу та управління, входи керуючих ідей і виконавчих ресурсів (час – гроші – матеріали), виходи індикації стану алгоритму реалізації ідеї і випуску продукції (сервісу).

В рамках створення КГС актуальними і ринково привабливими є такі напрями досліджень: 1) Технології диференціювання мобільних систем на спеціалізовані та структурований кіберпростір хмарних сервісів. 2) Квантові паралельні обчислення і спеціалізовані мультипроцесори. 3) Сенсорні інтелектуальні пристрої для реального світу і швидкодіючі роботи для кіберпростору. 4) Тривимірні мультипроцесори, 3D-обчислювальні процеси і багатовимірний кіберпростір. Тут мова йде про неминучі переходи обчислювальних пристроїв, процесів і компонентів в 3D-вимірювання у зв'язку з появою тривимірного FinFETs транзистора. 5) Це приведе весь комп'ютерний світ до реалізації абсолютно нової парадигми – паралельного програмування обчислювальних структур в трьох вимірах. 6) Важливим є створення 3D-мультипроцесора на об'ємному кристалі, що стане можливим у найближчі 5 років. Тут слід впроваджувати трикутну, а також тетраедну структуру міжкомпонентних сусідніх зв'язків як базових елементів для формування обчислювальних обсягів мікро- і мультипроцесоров, а також кіберпростору сервісів. 7) Стає актуальною розробка 3D-принтерів для реалізації технологій друкування (виготовлення) тривимірних оптимальних спеціалізованих архітектур, функціонально орієнтованих під конкретні алгоритми вирішення.

Еволюція кіберпростору планети розподіляється на наступні періоди: 1) 1980-і роки – формування парку персональних комп'ютерів; 2) 1990-ті роки – впровадження Інтернет-технологій у виробничі процеси і побут людини; 3) 2000-і роки – підвищення якості життя за рахунок впровадження мобільних пристроїв і хмарних сервісів; 4) 2010-і роки – створення цифрової інфраструктури моніторингу та активного управління рухомими об'єктами (повітряний, морський, наземний транспорт і роботи). Тому в даний час привабливою проблемою є системна інтеграція хмарного сервісу

моніторингу та управління, блоків радіочастотної ідентифікації транспорту, а також цифрових засобів дорожньої інфраструктури для оптимального on-line керування транспортними засобами та дорожнім рухом з метою вирішення соціальних, гуманітарних, економічних та екологічних проблем.

Що є основою кіберпростору планети? Кристал кремнію і його аналоги. Сучасна мікроелектроніка дає можливість створювати вже не плоскі, а об'ємні транзисторні структури (3D – FinFETs) в 14-7, 5, 3.5 і 2.5 нанометровому діапазоні, порівнянними з розміром атома. Це означає появу об'ємних 3D-System-on-Chip. Переваги таких об'ємних кристалів істотно впливають на характеристики промислових виробів в частині: енергоспоживання, розмірності, швидкодії, вартості та якості за рахунок зменшення не тільки розмірності компонентів, але і зв'язків між ними. Таким чином, мікросвіт кіберпростору переходить в 3D-вимірювання. Макросвіт все ще залишається плоским при об'єднанні в систему націнок компонентів, комп'ютерів, мереж, хмарних сервісів кіберпростору. Які аргументи можна висунути за перехід макросвіту в 3D-простір? Компактність інформаційних обсягів, швидкодія переміщення в кіберпросторі і його розмірність. Трикутна плоска структура системи, де всі вершини є сусідніми, має істотний недолік в двох вимірах – кодування трьох вершин або ребер займає три коди, а це означає, що один код двухбітового вектора не використовується. Тому створити примітивну структуру, де всі вершини є сусідніми, а їх кількість дорівнює чотирьом для повного використання кодового простору двох бітів, значить знову відкрити дивовижну за своїми властивостями 3D-фігуру – тетраедр. Він має шість ребер або відстаней, xor-сума яких дорівнює нулю. При завданні фігури два ребра є надлишковими, що можна використовувати для зменшення інформаційних обсягів до 66% при зберіганні і передачі даних. Формування кібернетичного простору на основі використання примітивів-тетраедрів дає можливість оптимізувати ставлення структурної складності простору до середньої відстані між двома точками.

Куди сьогодні прагне реальний кібернетичний світ? Корпоративні мережі, персональні комп'ютери, а також окремі сервіси (програмні продукти) йдуть в хмари кіберпростору, які мають яскраво виражену тенденцію до розшарування Інтернету по спеціалізованим сервісів, рис. 1.10.

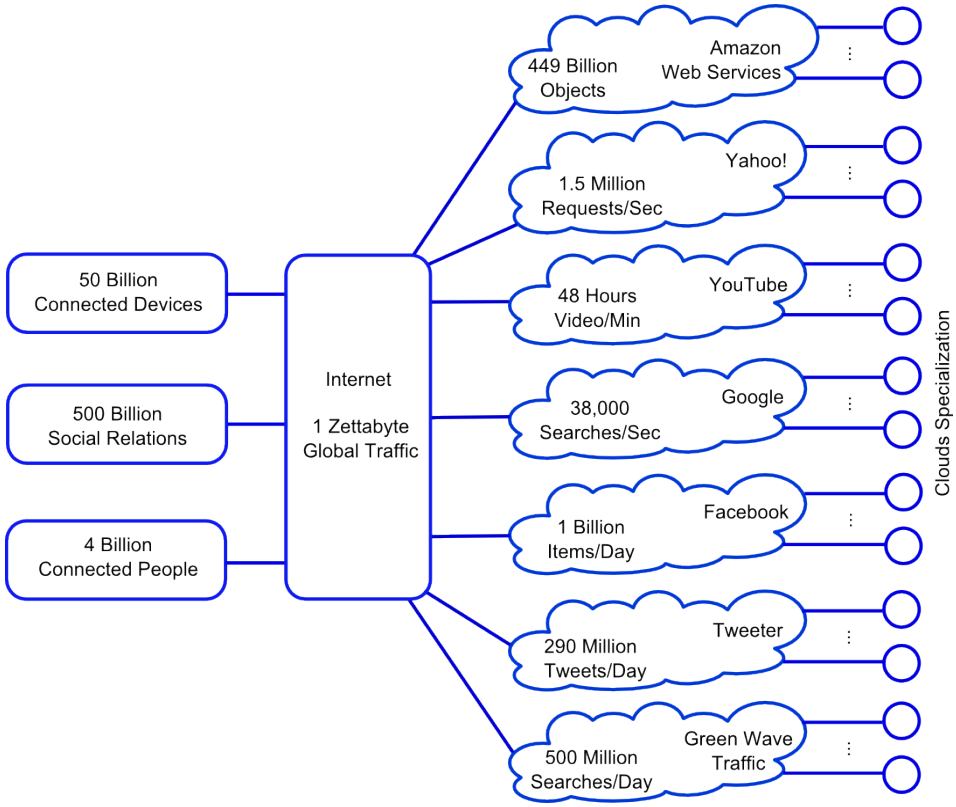



Рисунок 1.10 – Віртуалізація реального світу

Якщо сьогодні 4 мільярди користувачів з'єднуються в інтернеті (1 zettabytes $=10^{21}=2^{70}$ байт) за допомогою 50 мільярдів гаджетів, то через п'ять років кожен активний користувач буде мати не менше 10 пристроїв для зв'язку з кіберпростором. Стає неможливим використання персональних комп'ютерів без часткового або повного їх відображення або синхронізації на хмарах Інтернету. Це дає можливість вирішувати проблему віддаленого доступу до особистих даних і сервісів персонального комп'ютера при переміщенні користувачів в просторі. Економічний чинник неефективного використання придбаних додатків, розміщених в гаджетах і персональних комп'ютерах, змушує користувача відмовлятися від їх покупки на користь майже безкоштовної оренди сервісів на хмарах. Все згадане вище є істотним

аргументом і незаперечним доказом неминучого переходу всього людства в кіберпростір віртуальних мереж і комп'ютерів, розташованих в професійно надійних хмарах сервісів. Хмарні технології знімають практично всі згадані вище проблеми надійності, безпеки, сервісного обслуговування та практично не мають недоліків. Сказане безпосередньо стосується й сервісу дорожнього руху, який має цифрове відображення у кіберпросторі для подальшого моделювання всіх процесів на хмарі з метою запропонувати кожному водієві якісні умови пересування з економією часу і коштів.

Кіберфізична система online хмарного управління транспортом орієнтована на надання сервісів для 9 мільйонів водіїв України і 7 тисяч транспортних компаній. Аналогів таких систем у світі поки що не існує. Але є окремі компоненти для створення е-інфраструктури: електронні карти, супутникові системи локації і навігації, спеціалізовані бази даних в хмарах, засоби моніторингу, збору та захисту інформації, автомобільні комп'ютери і водійські гаджети з прийомопередавачами, централізовано керовані світлофори, стільниковий зв'язок, як частина необхідної інфраструктури для реалізації проекту. Існує розуміння з боку держави, громадян і поліції про необхідність створення і використання інтелектуальної інфраструктури і хмарного сервісу для якісного і безпечного дорожнього руху.

1. Теоретичні розробки інтелектуальних моделей, методів і програмно-апаратних засобів аналізу кіберпростору, пов'язані з дискретної оптимізацією, пошуку, розпізнавання та прийняття рішень [97-100].

2. Досвід розробки і застосування вбудованих і RFID цифрових систем для моніторингу дорожнього руху [101-105].

3. Досвід розробки і впровадження програмних продуктів і хмарних сервісів для оптимізації маршрутів транспортних засобів українських підприємств з метою мінімізації матеріальних і часових витрат, підвищення якості обслуговування пасажирів [106-111].

4. Розробки розподіленої системи управління дорожнім рухом в умовах великих міст і мегаполісів на основі високонадійної обчислювальної техніки [107-109].

5. Існуючі системи моніторингу дорожнього руху в США, Канаді та Японії – OnStar і NEXCO Central. Система OnStar орієнтована на моніторинг окремих машин, NEXCO Central здійснює глобальний моніторинг трафіку на основних і найбільш жвавих магістралях країни. OnStar сервіси доступні власникам автомобілів: Acura, Audi, Isuzu, Subaru, Volkswagen. На даний момент налічується близько 4 000 000 користувачів даного сервісу. Вартість одного пристрою моніторингу близько 200 $. Використовується CDMA канал зв'язку, що надається переважно Verizon Wireless в США і Bell Mobility в Канаді. Для визначення місця розташування використовується GPS. Є можливість голосового зв'язку з операторами. Інформація з сенсорів, в основному це датчики ударів і спрацьовування подушок безпеки, автоматично передається в call-центри. Це дозволяє негайно сповіщати про місцезнаходження аварії рятувальні та правоохоронні органи. Крім цього, всі машини, обладнані цією системою, мають GPS передавач, який дозволяє відстежити викрадений автомобіль. Існує також можливість отримання інформації про швидкість, витрату палива, напрямку руху і стиль водіння автомобіля. Дана інформація використовується страховими компаніями для розрахунку вартості індивідуальних страхових полісів. Нові моделі автомобілів обладнуються системою віддаленої зупинки двигуна. Автомобіль можна завести тільки після введення спеціального секретного коду. Вартість сервісів – Safe & Sound – 18.95 $ в місяць. Тариф включає автоматичне оповіщення про аварії, моніторинг викраденого автомобіля, аварійні сервіси. Directions & Connections – 28.90 $ в місяць. Додатково є можливість моніторингу напрямку руху і стилю водіння автомобіліста. Система NEXCO Central розроблена Japan Highway Public Corporation. Принцип роботи полягає в глобальному моніторингу дорожнього руху на головних автострадах країни. Система покриває близько 2000 км доріг. Дата-центр

обробляє дані, одержувані від датчиків з хвилинним інтервалом за допомогою глобальної IP мережі. На дорогах встановлено 744 точок зняття і передачі інформації по телефонних каналах про дорожню ситуацію.

6. Глобальні інфраструктурні проекти кілька століть поспіль об'єднують країни. Трансконтинентальні залізниці і автомагістралі, підводні тунелі і гігантські мости, космічні та авіаційні проекти служать найсильнішим каталізатором для розвитку економік багатьох держав. Розвиток мобільних і хмарних технологій зробили можливим безмежне масштабування обчислювальних потужностей і обсягів збережених даних. Наступний шар кіберпростору – Internet of Things (Car), що передбачає спілкування об'єктів між собою без людини, автоматизований збір, обробку та аналіз big data, що генеруються розумними сенсорами. Стає можливим побудова високоточних моделей поведінки навколишнього середовища і життєдіяльності міст, які стають розумними, що дозволить значно підвищити рівень життя, комфорту та безпеки водія за рахунок постійного моніторингу показів інтелектуальних датчиків у реальному часі.

7. Розвиток і впровадження концепції Internet of Things вимагає вирішення складних інженерних і наукових проблем в області комп'ютерної інженерії (http://eai.eu/, http://iot.ieee.org/), якими займаються світові науково-дослідні спільноти (IEEE, EAI) і компанії (Intel, NASA, IBM, Apple, Google, Samsung, Dell, AT & T, CISCO, THALES, WorldSensing, Aguila technologies, Connit, SigFox, Guglielmo, DQuid, BitCarrier). Ідея Internet of things заснована на використанні бездротових гетерогенних мереж, що працюють в різних частотних діапазонах за стандартами ZigBEE, WiFi, LTE (3G, 4G, 5G), Bluetooth, які забезпечують дальність дії, швидкість передачі даних і низьке енергоспоживання. Функціональність інфраструктури Internet of things – збір і аналіз даних використовує структури: big data, open data, cloud computing. Перші орієнтовані на аналіз великих обсягів неструктурованих даних. Другі – це концепція форматування інформації для використання комп'ютерами без втручання людини. Хмарні обчислення є масштабованою платформою для

обробки і зберігання big data, що надходять з розумних сенсорів в реальному часі. Невід'ємною частиною IoT є міжмашинна взаємодія M2M – набір технологій і підходів, що забезпечують обмін інформацією між механізмами для створення розумних міст (http://www.android.com/auto/, https://www.apple.com/ios/carplay/, http://java.dzone.com/articles/car-wars-con-cars. Тут фігурують компанії, що надають платформи і готові апаратні рішення: Connit, Aguila Technologies, WorldSensing. Проект WorldSensing FastPrk – це сервіс розумної парковки, що дозволяє водієві за допомогою мобільного пристрою знаходити вільне місце, а міській владі ефективніше використовувати паркувальні місця. Сенсор парковки працює в частотному діапазоні Sub-GHz з радіусом 500 м.

8. Вже сьогодні автомобіль немислимий без сервісів Інтернет, орієнтованих на суміщення приємної поїздки, комфортного відпочинку і реалізації бізнес-операцій всередині машини, яка виконує роль улюбленого і єдиного місця з повним набором функціональних сервісів ділової людини (голосові: зв'язок, повідомлення та пошта, банкінг та навігація, а також музика, ігри, відео) [112-128]. Все це стало можливим завдяки підключенню автомобіля до мережі Інтернет шляхом синхронізації мобільного телефону з монітором (комп'ютером) транспортного засобу, що дає можливість аутентифікувати водія у реальному світі і віртуальному кіберпросторі. Тут лідирують компанії Apple і Google. Вони анонсували платформи зв'язку, навігації та розваг, вбудовані до автомобілю (Apple iOS CarPlay, Google Android Auto), які використовують мікропроцесорну операційну систему Blackberry QNX. Для інсталяції системи необхідно підключити смартфон Android або iOS кабелем USB до комп'ютера автомобіля. Поява на ринку автомобільних ОС дозволяє зробити висновок, що через 3-4 роки всі машини будуть підключені до глобальних сервісів мережі Інтернет, спрямованих на підвищення якості життя людини в процесі дорожнього руху. Така домовленість є між 31 провідними автомобільними компаніями світу. Практично все, що було задекларовано в проекті iCTC [97-99] три роки тому, поступово реалізується в

сервісах Aplle CarPlay, крім одного – віртуального вуличного світлофора на лобовому склі-моніторі автомобіля [129-131]. Світлофорів немає на морі і у повітрі, не буде їх і на землі через 5-10 років.

9. Ринкова та науково-технічна привабливість проекту Smart Cloud Traffic Control підтверджується також оновленою структурою всесвітнього товариства інженерів IEEE, яке включає 38 спільнот, 9 комітетів і 7 рад, в тому числі, цікаві для проекту: IEEE Geoscience and Remote Sensing Society, IEEE Intelligent Transportation Systems Society, IEEE Professional Communication Society, IEEE Systems, Man, and Cybernetics Society, IEEE Vehicular Technology Society, IEEE Cloud Computing Community, IEEE Electric Vehicles Community, IEEE Biometrics Council, IEEE Sensors Council.

Функція мети, що формує якість інфраструктури, визначається інтегральним критерієм ефективності хмарної системи моніторингу та управління дорожнім рухом, що складається з трьох суперечливих параметрів: рівень помилкових рішень L, що призводять до колізій; Y – параметр якості обслуговування, що зменшує до нуля час вимушених простоїв транспортних засобів T. Критерій також враховує рівень додаткових витрат або структурну складність $H^a$ на створення і експлуатацію кіберфізичної інфраструктури H у порівнянні з базовою $H^s$, що реалізують механізми спостереження і керування транспортними потоками:

$$E = F(L, T, H) = \min[\frac{1}{3}(L + T + H)],$$
$$Y = (1 - P)^n;$$
$$L = 1 - Y^{(1-k)} = 1 - (1 - P)^{n(1-k)};$$
$$T = \frac{(1-k) \times H^s}{H^s + H^a};\ H = \frac{H^a}{H^s + H^a}.$$

Тут k і P є прохідність інфраструктури та ймовірність виникнення колізій відповідно, n – число прихованих помилок інфраструктури. Час пропуску заданого транспортного потоку через фрагмент інфраструктури визна-

чається її прохідністю k, помноженою на структурну складність дорожньої функціональності $H^s$, віднесеної до загальної складності інфраструктури $H^s + H^a$. Рівень технічної надмірності H знаходиться у функціональній залежності від складності механізмів прохідності, віднесеної до загальної складності заданої інфраструктури (дорожня функціональність + інфраструктурні механізми прохідності). При цьому компонент структурної надлишковості прохідності, а саме, спостережуваність інфраструктури забезпечує задану глибину діагностування пробок або транспортних колізій на дорожній структурі. У той час як керованість інфраструктури покликана усувати колізії інтелектом хмарних сервісів і світлофорів у межах заданого часового інтервалу.

Локальна функція мети хмарного сервісу для водія транспортного засобу визначається максимальним значенням функціоналу (Q=1), який складений з трьох суперечливих і безрозмірних параметрів метрики, наведеної до інтервалу (0,1): час T (X) – гроші M (X) – якість Y (X):

$$Q = \frac{1}{3}[T(X) + M(X) + Y(X)],$$

$$T(X) = \frac{1}{2}\left[\frac{V^r}{V} + \frac{T^m(X)}{T^r(X)}\right];$$

$$M(X) = \frac{M}{M + M^{\#}(X) + M^p + M^r};$$

$$Y(X) = [1 - P(X)]^n.$$

Тут кожен параметр істотно залежить від зеленого сигналу світлофора (X) на маршруті, який доставляє середнє арифметичне значення функціоналу за відсутністю інших колізій. Інші параметри позначають: $[V, V^r, T^r(X), T^m(X), M, M^{\#}(X), M^p, M^r, P(X), n]$ – модельну і реальну швидкість автомобіля на маршруті; реальний і модельний час очікування на перехресті; мінімальні грошові витрати на проходження маршруту без колізій; витрати, пов’язані з простоєм на перехресті; штрафи за порушення правил

дорожнього руху; непередбачені ремонти; кількість перехресть (X) на маршруті руху транспортного засобу.

Таким чином, за відсутністю світлофорів і транспортних колізій якість (життя водія) виконання маршруту автомобілем дорівнює одиниці. За їх наявністю використовується запропонована метрика з метою зменшення часу простою транспортного засобу на перехресті.

# 2 АНАЛІЗ ЯКОСТІ ТОПОЛОГІЧНИХ СТРУКТУР ДОРОЖНЬОГО РУХУ

Пропонуються критерії оцінювання якості дорожньої інфраструктури, наведені до аналізу топології на графах, які визначають середню довжину шляху між двома точками в обмеженому просторі. На прикладах графових структур оцінюються різні топології з метою їх можливого поліпшення. Представлені метрики і структури для вибору кращого рішення (маршрут руху, управління світлофором).

Мета розділу – розробка метрики, критеріїв оцінювання маршрутів руху і дорожніх інфраструктур для реалізації хмарних сервісів управління дорожнім рухом, що надаються водієві у режимі реального часу. Завдання дослідження: 1) Створення критеріїв оцінювання якості інфраструктури дорожнього руху. 2) Розробка аналітичних моделей прохідності (спостереження та управління) дорожньої інфраструктури. 3) Пропозиція метрики для вимірювання е-інфраструктури дорожнього руху. 4) Аналіз різних топологій і зв’язків компонентів інфраструктури. 5) Створення спрощених критеріїв оцінювання якості топології дорожньої інфраструктури. 6) Створення наведеної метрики вимірювання топології графа дорожньої інфраструктури.

## 2.1 Критерії якості інфраструктури дорожнього руху

Критерії якості топології доріг пов’язані зі статичним аналізом графової структури, що має E дуг і n вершин. Особливість першого з них полягає в обчисленні абсолютного і не наведеного до одиничного інтервалу значення, яке формується вартістю з'єднань, помноженої на якість транзакцій між усіма парами вершин:

$$Q_1 = \frac{E}{n} \times \sum_{i=1}^{n} \min_{j} (p_{ij}).$$

Застосування цієї формули до оцінювання трьох графових структур (рис. 2.1), що мають 6 вершин і різні топології з'єднання, представлені нижче:

$G_1$ =

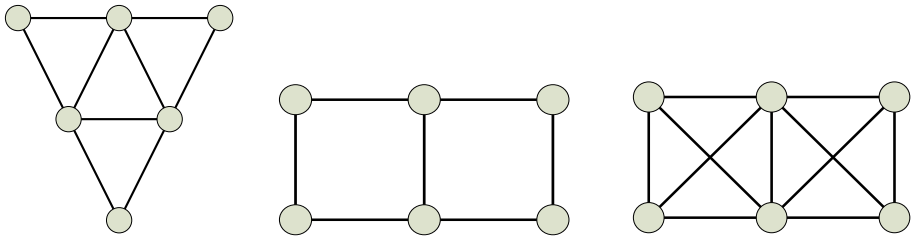
| | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| a | . | 1 | 1 | . | . | . |
| b | 1 | . | 1 | 1 | . | 1 |
| c | . | 1 | . | . | . | 1 |
| d | 1 | 1 | . | . | 1 | 1 |
| e | . | . | . | 1 | . | 1 |
| f | . | 1 | 1 | 1 | 1 | . |

$G_2$ =

| | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| a | . | 1 | . | 1 | . | . |
| b | 1 | . | 1 | . | 1 | . |
| c | . | 1 | . | . | . | 1 |
| d | 1 | . | . | . | 1 | . |
| e | . | 1 | . | 1 | . | 1 |
| f | . | . | 1 | . | 1 | . |

$G_3$ =

| | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| a | . | 1 | . | 1 | 1 | . |
| b | 1 | . | 1 | 1 | 1 | 1 |
| c | . | 1 | . | . | 1 | 1 |
| d | 1 | 1 | . | . | 1 | . |
| e | 1 | 1 | 1 | 1 | . | 1 |
| f | . | 1 | 1 | . | 1 | . |



а б в

Рисунок 2.1 – Структури сполук графових вершин

Тут три графа мають 9, 7 і 11 дуг відповідно. Підрахунок критерію відповідно до останньої формулою дає наступні результати:

$$Q_1(G_1) = \frac{E}{n} \times \sum_{i=1}^{n} \min_j (p_{ij}) = \frac{9}{6} \times (9 \times 1 + 6 \times 2) = 31{,}5;$$

$$Q_1(G_2) = \frac{7}{6} \times (7 \times 1 + 6 \times 2 + 2 \times 3) = 29{,}2;$$

$$Q_1(G_3) = \frac{11}{6} \times (11 \times 1 + 4 \times 2) = 34{,}8.$$

Модифікація оцінки ефективності топології пов'язана з формуванням інтервальної оцінки (0,0 – 1,0) шляхом приведення реальних витрат (число дуг E) до максимально можливої кількості $V = \frac{n^2 - n}{2}$ парних з'єднань в графі $\frac{E}{V}$, що забезпечує якість комунікаційних властивостей $\frac{V}{\sum_{i=1}^{V} \min_j (p_{ij})}$:

$$Q_2 = \frac{E}{\frac{n^2-n}{2}} \times \frac{\frac{n^2-n}{2}}{\sum_{i=1}^{\frac{n^2-n}{2}} \min_j (p_{ij})} = \frac{E}{\sum_{i=1}^{\frac{n^2-n}{2}} \min_j (p_{ij})} = \frac{E}{\sum_{i=1}^{V} \min_j (p_{ij})}.$$

Оцінка має одиничне значення, якщо чисельник і знаменник дорівнюють $V = \frac{n^2-n}{2}$. В цьому випадку графовая структура має всі можливі парні сполуки між вершинами графа, які визначаються половиною декартова квадрата потужності безлічі вершин мінус n вершин. Значення від'ємника формується відсутністю ідемпотентна замикань всіх вершин графа. При цьому кожна пара вершин має довжину шляху, що дорівнює одиниці. Якщо структура представлена неорієнтованим графом, то потужність всіх пар вершин подвоюється $V = n^2 - n$. Перерахунок критеріїв ефективності міжз'єднань всіх пар вершин, представлених на останньому рисунку, дає наступний результат:

$$Q_2(G_1) = \frac{E}{\sum_{i=1}^{V=12} \min_j (p_{ij})}. = \frac{9}{9 \times 1 + 6 \times 2} = 0{,}428;$$

$$Q_2(G_2) = \frac{7}{7 \times 1 + 6 \times 2 + 2 \times 3} = 0{,}28;$$

$$Q_2(G_3) = \frac{11}{11 \times 1 + 4 \times 2} = 0{,}578.$$

Збільшення числа зв'язків (чисельника) призводить до зменшення суми довжин шляхів між усіма парами вершин (знаменника). У межі для сильного графа значення дробу дорівнює одиниці. Найкраща якість інфраструктури. При цьому платою за якість комунікацій є прихована від експерта потужність з'єднань, приведена до максимально можливої кількості ребер:

$$H_2(G_1) = \frac{E}{\frac{n^2 - n}{2}} = \frac{9}{15} = 0,60;$$

$$H_2(G_2) = \frac{7}{15} = 0,46;$$

$$H_2(G_3) = \frac{11}{15} = 0,73.$$

Таким чином, мультиплікативность двох критеріїв: складності графа і якості парних комунікацій створюють структурну невизначеність, яка приховує вартість з'єднань і якість комунікацій. Дану ситуацію можна вирішити шляхом формування структурної оцінки, яка містить критерій складності комунікаційної інфраструктури, що формує інтервальні витрати часу – середню досяжність між кожною парою вершин графової структури, приведену до максимально можливої довжини з'єднань. Це демонструє вартість якості інфраструктури, що має цільову функцію, яка мінімізує середню досяжність (довжину шляху або часу) між парою вершин графової структури:

$$Q_3 = \frac{E}{V} \times \left[ \frac{1}{n-1} \times \frac{1}{V} \times \sum_{i=1}^{V} \min_{j} (p_{ij}) \right] = \frac{E}{V^2 (n-1)} \times \sum_{i=1}^{V} \min_{j} (p_{ij})$$

,

$$Q_3(G_1) = \frac{9}{15} \times (\frac{21}{15 \times 5}) = 0,168;$$

$$Q_3(G_2) = \frac{7}{15} \times (\frac{25}{15 \times 5}) = 0,155;$$

$$Q_3(G_3) = \frac{11}{15} \times (\frac{19}{15 \times 5}) = 0,185.$$

В останній оцінці існує проблема отримання мінімального значення добутку, якій менше меншого множника. Для того, щоб интервальна оцінка була більш виваженою і значущою в діапазоні від 0 до1, її необхідно зробити адитивною по відношенню до середнього значення двох компонентів, що складають критерій:

$$Q_4 = \frac{1}{2} \times \left[ \frac{E}{V} + V \times \left( \sum_{i=1}^{V} \min_{j} (p_{ij}) \right)^{-1} \right],$$

$$Q_4(G_1) = \frac{1}{2} \times \left[ \frac{9}{15} + \frac{15}{21} \right] = \frac{1}{2} \times [0{,}6 + 0{,}71] = 0{,}655;$$

$$Q_4(G_2) = \frac{1}{2} \times \left[ \frac{7}{15} + \frac{15}{25} \right] = \frac{1}{2} \times [0{,}467 + 0{,}60] = 0{,}504;$$

$$Q_4(G_3) = \frac{1}{2} \times \left[ \frac{11}{15} + \frac{15}{19} \right] = \frac{1}{2} \times [0{,}73 + 0{,}79] = 0{,}751;$$

Тут перший компонент – інфраструктурна складність (вартість проекту) – наближається до одиниці за мірою збільшення числа ребер в графі, максимальна кількість яких дорівнює $V = (n^2 - n)/2$. Другий компонент – інфраструктурна якість – наближається до одиничного значення лише в тому випадку, якщо нові зв’язки забезпечують зменшення довжини шляхів між парами вершин графа. Оцінка може мати одиничну складність проекту або максимальне число з'єднань, але при цьому не досягти одиничної якості, якщо зв'язку невірно розставлені. Цим пояснюється обов’язкова наявність двох компонентів структурного критерію, які одночасно здатні оцінити складність і якість інфраструктури. Оцінка буде максимальною і дорівнює 1 при побудові сильного графа, де кожна пара вершин стає суміжній:

$$Q_4(G_1) = \frac{1}{2} \times \left[ \frac{15}{15} + \frac{15}{15} \right] = 1{,}0;$$

$$Q_4(G_2) = \frac{1}{2} \times \left[ \frac{15}{15} + \frac{15}{15} \right] = 1{,}0;$$

$$Q_4(G_3) = \frac{1}{2} \times \left[ \frac{15}{15} + \frac{15}{15} \right] = 1{,}0;$$

Цікаво побудувати критерій відношення складності до якості, який буде характеризувати властивість – якою ціною досягається якість інфраструктури:

$$Q_5 = \frac{\frac{E}{V}}{V \times \left( \sum_{i=1}^{V} \min_{j}(p_{ij}) \right)^{-1}} = \frac{E}{V^2} \times \sum_{i=1}^{V} \min_{j}(p_{ij})$$

$$Q_5(G_1) = \left[ \frac{15}{225} \times 15 \right] = 1{,}0; \qquad Q_5(G_1) = \left[ \frac{9}{225} \times 21 \right] = 0{,}84;$$

$$Q_5(G_2) = \left[ \frac{15}{225} \times 15 \right] = 1{,}0; \qquad Q_5(G_2) = \left[ \frac{7}{225} \times 25 \right] = 0{,}778;$$

$$Q_5(G_3) = \left[ \frac{15}{225} \times 15 \right] = 1{,}0; \qquad Q_5(G_3) = \left[ \frac{11}{225} \times 19 \right] = 0{,}93;$$

Критерій знаходиться в інтервалі від 0 до 1 і показує ефективність (яка наближається до одиниці) введення нових зв’язків для зменшення середньої довжини шляху між парою вершин інфраструктури. В чисельник дробу виходить твір складності на якість, віднесене до максимального значення компонентів даного твору, який має місце бути для сильного графа. Як правило, підвищення числа зв'язків призводить до зменшення суми довжин шляхів між усіма парами вершин інфраструктури. Останній критерій

$$Q_5 = \frac{E}{V^2} \times \sum_{i=1}^{V} \min_{j}(p_{ij})$$

цікавий не тільки інтегральним значенням ефективності, але і структурними компонентами, які являють собою оцінки складності і якості інфраструктурного рішення для оптимального вибору в умовах обмежень. Для практичного використання будь-якого з п’яти критеріїв ефективності необхідно мати швидкодіючий паралельний алгоритм (Дейкстра) обчислення довжин шляхів між усіма парами вершин графа. Необхідно також шукати більш прості оцінки ефективності графових структур, де замість суми всіх пар мінімальних шляхів буде фігурувати обчислювально нескладний компонент, що корелюється зі згаданою сумою.

## 2.2 Функції прохідності дорожньої інфраструктури

Інший цікавий напрямок метрики вимірювання і оцінювання інфраструктури системи моніторингу і управління пов'язано з поняттям прохідності або тестопригодності (trafficability), що включає два критерії: керованість і спостережуваність, які дають можливість оцінювати як систему в цілому, так і якість руху автомобіля за заданим маршрутом в режимі on-line (в реальному масштабі часу враховувати транспортні потоки). Існуючі стандарти тестопригодного проектування комп'ютерних систем і мереж (IEEE 11.49, IEEE 1500) можна адаптувати для проектування, тестування і верифікації інфраструктури дорожнього руху з метою її подальшої ефективної експлуатації.

Досить істотна надмірність інфраструктури дорожнього руху передбачає її ефективне використання з метою підвищення її прохідності. Для цього використовується (орієнтований) граф транспортних потоків (ГТП), який надає користувачеві інформацію про взаємозв'язки доріг і перехресть. Для інтегрального оцінювання прохідності Q вводяться наступні критерії:

$$Q = \frac{Z(S)}{Z(S)+Z(F)+Z(B)+Z(A)} \times \frac{1}{n}\sum_{i=1}^{n}(U_i \times N_i);$$

$$U_i = \frac{1}{T}\sum_{j=1}^{x_i} T_j^i \times \frac{1}{d_i^x \vee t_i^x}; N_i = \frac{1}{T}\sum_{j=1}^{y_i} T_j^i \times \frac{1}{d_i^y \vee t_i^y}.$$

де n – кількість вершин графа. Керованість і спостережуваність є метрикою оцінювання прохідності інфраструктури дорожнього руху. Керованість вершини (перехрестя) має функціональну залежність від структурної глибини $d_i^x$ знаходження вершини щодо пункту відправлення або довжини кон'юнктивного терма – $t_i^x$. Спостережуваність вершини має аналогічну залежність $d_i^y \vee t_i^y$ щодо пункту прибуття. Для підрахунку прохідності можна використовувати один з параметрів $d_i^x, t_i^x (d_i^y, t_i^y)$. Оцінки управління та

спостереження залежать також від процентного відношення числа вхідних дуг $\frac{1}{T}\sum_{j=1}^{x_i} T_j^i$ (вихідних $\frac{1}{T}\sum_{j=1}^{y_i} T_j^i$) розглядуваної вершини до загальної кількості дуг T, де $x_i$ – кількість дуг, що формують доступ до вхідних вершині; ( $y_i$ ) – число дуг, що визначають вершину як джерело. Прохідність Q залежить від керованості U, спостережливості (N), а також від вартості реалізації (Z) компонентів інфраструктури, тут визначена метрика структурного покриття (F), транспортні потоки (B), механізм спостереження (A), функціональність (S). Керованість (спостережуваність) є функція від числа дуг, що входять до вершини (що виходять з вершини) графа, а також від структурної глибини вершини (перехрестя) – відстані від пункту відправлення (прибуття). Наведений критерій прохідності може бути також використаний і для оцінки якості граф-схеми або інфраструктури управління дорожнього руху. Тут розглядаються вершини, навантажені вхідними умовами, а також позиція вершини по відношенню до початку або закінчення запланованого маршруту. Кількість умов або вхідних дуг в кожній вершині, об'єднане операцією Or, впливає на прохідність графа в частині керованості. Аналогічно обчислюється спостережуваність, на яку впливає структурна глибина і потужність вихідних (дуг) умов, створювана операціями And, Or. Таким чином, прохідність даної вершини орієнтованого графа може бути представлена логічною функцією, заданою у вигляді кон'юнктівної нормальної форми (КНФ). При цьому керованість і спостережуваність буде визначатися оцінкою за Квайном обчислювальної складності КНФ. У загальному випадку логічні функції спостереження та управління вершини графа задаються кон’юнкцією діз’юнктивних умов (перший рядок):

$$1)U_r^f = \bigwedge_{i=1}^{n_r^x}\left(\bigvee_{j=1}^{x_i} T_{rij}^x\right); N_r^f = \bigwedge_{i=1}^{n_r^y}\left(\bigvee_{j=1}^{y_i} T_{rij}^y\right);$$

$$2)U_r^f = \bigvee_{i=1}^{x_r}\left(\bigwedge_{j=1}^{n_i^x} T_{ij}^x\right); N_r^f = \bigvee_{i=1}^{y_r}\left(\bigwedge_{j=1}^{n_i^y} T_{ij}^y\right).$$

Тут функція керованості $U_r^f$ (спостережуваності $N_r^f$) визначається кон'юнкцією всіх вершин-попередників $n_r^x$ (наступників $n_r^y$), Де кожна з них має $x_i$ дуг, що входять (виходять $y_i$), з'єднаних знаками диз'юнкції. Потужність діз'юнктівних термів відповідає кількості вхідних до вершини дуг, а число кон'юнкцій є структурна глибина розташування розглянутого компонента в графі. Далі кон'юнктивна форма перетвориться до виду ДНФ – другий рядок в попередньому виразі, де число термів для функції керованості (спостережуваності) $x_r(y_r)$ дорівнює всім можливим шляхам формування стану розглянутої вершини, а довжина терма керованості (спостережливості) $n_i^x(n_i^y)$ є умова досяжності вершини – структурна глибина від входів (виходів).

Цікавим видається нестандартне рішення, коли критерії спостережуваності та управління поточної вершини транзакційного графа обчислюються на підставі моделі у вигляді побудованих логічних функцій спостережуваності та управління $(U_i, N_i)$ і інтегральної оцінки прохідності (Q) при використанні апарату – алгебраїчної форми подання графа. Формули підрахунку, які передбачають облікові оцінки, мають такий вигляд:

$$U_i = \frac{1}{t_{max}^x \times n_t^x} \times \sum_{i=1}^{n_t^x} \sum_{j=1}^{k_i^x} (t_{max}^x - \left|t_{ij}^x\right| + 1);$$

$$N_i = \frac{1}{t_{max}^y \times n_t^y} \times \sum_{i=1}^{n_t^y} \sum_{j=1}^{k_i^y} (t_{max}^y - \left|t_{ij}^y\right| + 1);$$

$$Q = \frac{1}{n} \sum_{i=1}^{n} (U_i \times N_i),$$

де $t_{max}^x, n_t^x, k_i^x, \left|t_{ij}^x\right|$ – кон'юнктивний терм максимальної довжини для визначення критерію керованості; кількість термів в логічній функції керованості; кількість букв (змінних або дуг) в поточному термі функції;

потужність розглядуванового терма. Аналогічні позначення використовуються і при підрахунку критерію спостережуваності – $t^y_{max}, n^y_t, k^y_i, \left|t^y_{ij}\right|$ кожної вершини графа.

Для фрагмента графової структури, представленої на рис. 2.2,

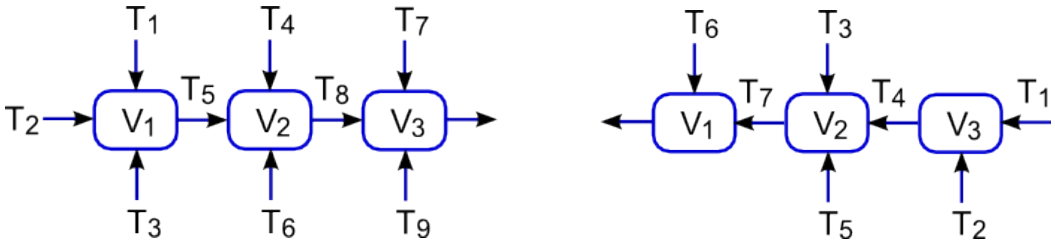



Рисунок 2.2 – Графи для підрахунку спостереження та управління

перетворення кон'юнктівной форми у діз'юнктивну модель щодо вершин $V_1 - V_3$ за заданими вище правилами формує такі логічні функції керованості:

$$U^f(V_1) = T_1 \vee T_2 \vee T_3;$$
$$U^f(V_2) = (T_1 \vee T_2 \vee T_3)T_5 \vee T_4 \vee T_6 = T_1T_5 \vee T_2T_5 \vee T_3T_5 \vee T_4 \vee T_6;$$
$$U^f(V_3) = (T_1 \vee T_2 \vee T_3)T_5T_8 \vee (T_4 \vee T_6)T_8 \vee (T_7 \vee T_9) =$$
$$= T_1T_5T_8 \vee T_2T_5T_8 \vee T_3T_5T_8 \vee T_4T_8 \vee T_6T_8 \vee T_7 \vee T_9.$$

Дотримуючись виразу, який визначає процедури підрахунку оцінок, можна визначити керованість компонента $V_3$:

$$U(V_3) = \frac{1}{t^x_{max} \times n^x_t} \times \sum_{i=1}^{n^x_t} \sum_{j=1}^{k^x_i} (t^x_{max} - \left|t^x_{ij}\right| + 1) = \frac{1}{3 \times 7} \times (1+1+1+2+2+3+3) = 0{,}61.$$

Для другого фрагмента графа, представленого на рис. 2.2, перетворення кон'юнктивної форми у діз'юнктивну структуру дозволяє визначити логічну функцію і чисельне значення спостережуваності для вершини $V_1$:

$$N^f(V_1) = T_6 \vee T_7(T_3 \vee T_5 \vee T_4(T_1 \vee T_2) = T_6 \vee T_7T_3 \vee T_7T_5 \vee T_7T_4T_1 \vee T_7T_4T_2;$$

$$N^f(V_2) = T_3 \vee T_5 \vee T_4(T_1 \vee T_2) = T_3 \vee T_5 \vee T_4T_1 \vee T_4T_2; V_3 = T_1 \vee T_2.$$

$$N(V_1) == \frac{1}{t^y_{max} \times n^y_t} \times \sum_{i=1}^{n^y_t} \sum_{j=1}^{k^y_i} (t^y_{max} - \left|t^y_{ij}\right| + 1) = \frac{1}{3 \times 5} \times (3 + 2 + 2 + 1 + 1) = \frac{9}{15} = 0{,}6.$$

Для випадку, коли дуги в графі мають вагові коефіцієнти $b^x_{ij}$ ($b^y_{ij}$), що показують чисельне значення транспортних потоків, задіяних у смугах руху між вершинами (мультидуги), формули підрахунку прохідності суттєво ускладнюються:

$$U_i = \frac{\sum_{i=1}^{n^x_t} \prod_{j=1}^{k^x_i} b^x_{ij} (t^x_{max} - \left|t^x_{ij}\right| + 1)}{t^x_{max} \times (\sum_{i=1}^{n^x_t} \prod_{j=1}^{k^x_i} b^x_{ij})}; \quad N_i = \frac{\sum_{i=1}^{n^y_t} \prod_{j=1}^{k^y_i} b^y_{ij} (t^y_{max} - \left|t^y_{ij}\right| + 1)}{t^y_{max} \times (\sum_{i=1}^{n^y_t} \prod_{j=1}^{k^y_i} b^y_{ij})}$$

Синтезовані логічні функції моделі дорожнього руху задають всі можливі шляхи управління, як в часі, так і у просторі, що можна вважати новою аналітичною формою опису прохідності інфраструктури. За ДНФ можна визначити критерії керованості (спостережуваності) для всіх компонентів граф-моделі. Тут розглядаються два варіанти (сценарії) обрахунку інфраструктури. 1) Чи враховується тільки графова модель, де вага кожної дуги дорівнює 1, незалежно від величини транспортного потоку. 2) Всі дуги графа відзначаються реальною кількістю транспортних транзакцій, що мають місце бути між двома вершинами графа. Оцінки прохідності описаних варіантів можуть застосувати перший сценарій, або мати більш складну і точну модель транспортних транзакцій, розподілених у часі, на множині графових компонентів.

Результат обчислення керованості (спостережуваності) для всіх вершин графа дає можливість отримати графік, який дозволяє визначити критичні

вершини для установки необхідних точок управління (світлофорів) та спостереження (моніторів). Після визначення спостереження та управління вершин транзакційного графа виконується підрахунок узагальненого критерію прохідності аналізованої інфраструктури. Вона характеризує якість проектного варіанту, що є досить суттєвим при порівнянні декількох альтернативних рішень.

## 2.3 Метрика е-інфраструктури дорожнього руху

Для ідентифікації або розпізнавання компонентів, об'єктів, процесів або явищ в кіберпросторі необхідно мати ефективну метрику вимірювання відстаней в дискретно булевом (векторно-логічному) просторі, на основі якої будуються швидкодіючі механізми, що визначають похідну, ступінь подібності або відмінності між ними.

Дискретний векторно-логічний простір (кіберпростір) – сукупність взаємодіючих за відповідною метрикою інформаційних процесів і явищ, що описуються векторами логічних змінних і використовують як носій пам’яті.

Метрика – спосіб вимірювання відстані у просторі між компонентами процесів або явищ, описаних векторами логічних змінних. Відстань у кіберпросторі – це хог-відношення між парою векторів, які позначають компоненти процесу або явища, що відрізняє його від кодової відстані за Хеммінгом. Відстань, похідна (булева), ступінь зміни, відмінності або близькості є ізоморфні поняття, пов'язані з визначенням відношення двох компонентів процесу або явища. Поняття відстані між компонентами у кіберпросторі є міра їх відмінності. Процедури порівняння, вимірювання, оцінювання, розпізнавання, тестування, діагностування, ідентифікування є спосіб визначення ставлення при наявності не менше ніж одного об’єкта.

Компонент простору представлений $k$-мірним (двійковим) вектором $a = (a_1,...,a_j,...,a_k), a_j \in \{0,1\}, k = 1,2,3,...$, де кожна його координата визначена у

двійковому алфавіті, 0 – «неправда», 1 – «істина». Нуль-вектор є k-мірний кортеж, всі координати якого дорівнюють нулю: $a_j = 0, j = \overline{1,k}$ .

Метрика $\beta$ кіберпростору визначається єдиною рівністю, яка формує нуль-вектор для xor-суми відстаней $d_i$ між ненульовим і кінцевим числом точок (об'єктів), замкнутих в цикл:

$$\beta = \bigoplus_{i=1}^{n} d_i = 0,$$

де n – кількість (ціле число) відстаней між компонентами (векторами) простору, складовими цикл $D = (d_1,...,d_i,...,d_n)$, $d_i$ – є вектор відстані, відповідний ребру циклу, що з'єднує два компонента (вектора) a, b простору, який далі позначається без індексу як $d(a,b)$. Відстань між двома об'єктами (векторами) a і b є похідний вектор: $d(a,b) = (a_j \oplus b_j)_1^k$. Векторному значенн. відстані відповідає норма – скалярнf відстань за Хеммінгом між двома векторами – як число одиниць вектора $d(a,b)$. Інакше: метрика $\beta$ векторного логічного двійкового простору є рівна нуль-вектору xor-сума відстаней між кінцевим числом точок (вершин) графа, що утворюють цикл. Сума n-мірних двійкових векторів, які задають координати точок циклічної фігури, дорівнює нуль-вектору. На основі введеної метрики можна дати більш формальне визначення кіберпростору, який є векторно-логічним, нормованим $\beta$-метрікою, де xor-сума відстаней між кінцевим числом точок циклу дорівнює нуль-вектору. Визначення метрики ставить на перше місце не елементи множини, але відношення, що дозволяє скоротити систему аксіом (тотожності, симетрії і транзитивного трикутного замикання) з трьох до одного і поширити її дію на як завгодно складні конструкції n-мірного логічного простору. Класичне завдання метрики для визначення взаємодії однієї, двох і трьох точок у векторному логічному просторі є окремим випадком $\beta$-метріки при $i = 1,2,3$ :

$$M = \begin{cases} d_1 = 0 \leftrightarrow a = b; \\ d_1 \oplus d_2 = 0 \leftrightarrow d(a,b) = d(b,a); \\ d_1 \oplus d_2 \oplus d_3 = 0 \leftrightarrow d(a,b) \oplus d(b,c) = d(a,c). \end{cases}$$

Векторно-логічний транзитивний трикутник має повну аналогію чисельному вимірі відстані в метричному М-просторі, яке задається системою аксіом, що визначає взаємодію однієї, двох і трьох точок в будь-якому просторі:

$$M = \begin{cases} d(a,b) = 0 \leftrightarrow a = b; \\ d(a,b) = d(b,a); \\ d(a,b) + d(b,c) \geq d(a,c). \end{cases}$$

Специфіка аксіоми трикутника метричного простору полягає у чисельному (скалярному) порівнянні відстаней трьох об’єктів, коли інтервальна невизначеність відповіді – дві сторони трикутника можуть бути більше або рівні третій – малопридатна для визначення точної довжини останньої сторони. Векторно-логічний простір усуває даний недолік, повністю виключає ступінь невизначеності у бінарному відношенні детермінованих станів процесів або явищ. В цьому випадку чисельна невизначеність третьої сторони трикутника у векторному логічному просторі набуває форму точного двійкового вектора, який характеризує відстань між двома об'єктами і обчислюється на основі знання відстаней двох інших сторін трикутника: $d(a,b) \oplus d(b,c) = d(a,c) \rightarrow d(a,b) \oplus d(b,c) \oplus d(a,c) = 0$.

Метрика $\beta$ багатозначного векторно-логічного кіберпростору є вектор, що дорівнює значенню $\Delta$ за всіма координатами, отриманий шляхом застосування симетричної різниці відстаней між кінцевим числом точок, що утворюють цикл: $\beta = \overset{n}{\underset{i=1}{\Delta}} d_i = \varnothing$, де кожна координата вектора, що відповідає об'єкту, визначена в алфавіті, який становить булеан на універсуме примітивів потужністю p: $a_j = \{\alpha_1,...,\alpha_r,...,\alpha_m\}, m = 2^p$ :

$$a\Delta b = \{0\Delta 0 = \varnothing,\ 0\Delta 1 = x,\ 0\Delta x = 1,\ 0\Delta\varnothing = 0,\ 1\Delta 1 = \varnothing,$$
$$1\Delta x = 0,\ 1\Delta\varnothing = 1,\ x\Delta x = \varnothing,\ x\Delta\varnothing = x,\ \varnothing\Delta\varnothing = \varnothing\}.$$

Рівність порожньому вектору симетричної різниці покоординатної теоретико-множинної взаємодії підкреслює рівнозначність компонентів (відстаней), що беруть участь у формуванні рівняння, де єдина координатна операція $d_{i,j}\Delta d_{i+1,j}$ визначена у чотиризначній моделі Кантора $A = \{0, 1, x, \varnothing\}, x = \{0,1\}$. Замкнутість будь-якого теоретико-множинного алфавіту можна перевіряти за допомогою $\Delta$-операції (багатозначний аналог xor-операції), коли $\Delta$-сума всіх символів повинна дорівнювати пустой множині: $0\Delta 1\Delta x\Delta\varnothing = \varnothing$. Якщо кодувати дану процедуру двійковими кодами-векторами, то xor-сума буде дорівнювати нуль вектору: $01\oplus 10\oplus 11\oplus 00 = 00$. Для практичного використання введеної метрики кіберпростору далі пропонується доказовий перехід від чисельної характеристики бінарного відношення об'єктів, що об'єднує три скалярні оцінки їх взаємодії, до чисто векторно-логічного критерію якості відношень двох об'єктів.

Нехай вхідний вектор $m = (m_1,..., m_j,..., m_k), m_j \in \{0,1, x\}$ і аналізований об'єкт $A = (A_1,..., A_j,..., A_k), A_j \in \{0,1, x\}$, який також представлений вектором, мають однакову розмірність k. Ступінь приналежності m-вектора до A позначається як $\mu(m \in A)$.

Визначення. Інтегральна теоретико-множинна метрика для оцінювання якості запиту (послуги) є функція взаємодії багатозначних за координатами векторів $m\Delta A$, яка визначається середньою сумою трьох параметрів: кодова відстань $d(m, A)$, функція приналежності $\mu(m \in A)$ і функція приналежності $\mu(A \in m)$:

$$Q = \frac{1}{3}[d(m, A) + \mu(m \in A) + \mu(A \in m)],$$
$$d(m, A) = \frac{1}{n}[n - card[(i : m_i \cap A_i = \varnothing, i = 1,..., k)]]; \mu(m \in A) = 2^{c-a}; \mu(A \in m) = 2^{c-b};$$
$$a = card[(i : A_i = x, i = 1,..., k)]; b = card[(i : m_i = x, i = 1,..., k)]; c = card[(i : m_i \cap A_i = x, i = 1,..., k)];$$

Перетин (об'єднання) векторів – є векторна операція, заснована на відповідних координатних теоретико-множинних операціях. Операції координатного перетину і об’єднання визначені в алфавіті Кантора $A = \{0,1,x = \{0,1\},\varnothing\}$. Нормування параметрів дозволяє оцінити рівень взаємодії векторів в чисельному інтервалі [0,1].

Процедура обчислення векторного критерію якості залежить від значности алфавіту:

$$Q' = \begin{cases} m \oplus A \leftarrow \{m_i, A_i\} \in \{0,1\}; \\ m \Delta A \leftarrow \{m_i, A_i\} \in \{0,1,x\}. \end{cases}$$

Критерій якості Q однозначно визначає три форми взаємодії двох будь-яких об'єктів у n-вимірному векторному логічному просторі: відстань і дві функції приналежності. Процес-модель пошуку рішення з мінімальним числом одиничних координат з більш, ніж двох альтернатив, представлена на рис. 2.3. Вона включає наступні операції: 1) Спочатку до вектору-результату $Q$, у якому буде збережено краще рішення, заносяться одиничні значення в усі координати (найгірше рішення) і одночасно здійснюється операція slc зсуву вліво з ущільненням одиниць поточного вектора $Q_i$. 2) Виконується порівняння двох векторів: Q і чергової оцінки $Q_i$ зі списку рішень. 3) Реалізується векторна операція and $(Q \wedge Q_i)$, а результат порівнюється з вектором $Q$, що дає можливість змінити його, якщо вектор $Q_i$ має менше число одиничних значень. 4) Процедура пошуку оцінки кращого рішення повторюється n раз.

$$Q = Q\overline{(\vee((Q \wedge Q_i) \oplus Q))} \vee Q_i(\vee((Q \wedge Q_i) \oplus Q));$$
$$Y = \vee((Q \wedge Q_i) \oplus Q);$$
$$Q = Q\overline{Y} \vee Q_i Y.$$

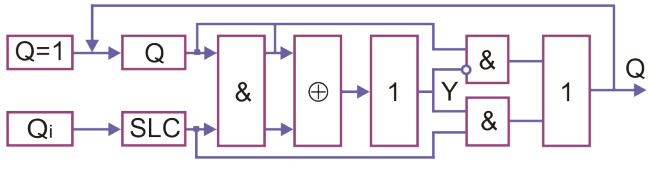



Рисунок 2.3 – Процес-модель вибору рішення

Тут or-оператор редукції (після xor елемента) формує бінарне однобітове рішення на основі логічної операції or над n розрядами критерію

якості. Таким чином, векторний логічний критерій якості взаємодії об'єктів у кіберпросторі дозволяє отримувати оцінку рішення з високою швидкодією логічних паралельних операцій, що особливо істотно для критичних хмарних систем управління трафіком у реальному часі.

## 2.4 Топології зв’язків компонентів системи

Топологія зв’язків або відношень між компонентами різної природи формує різноманіття системоутворюючих структур. При цьому оптимізація числа компонентів або кількості зв'язків стає критично важливою для функціонування системи, що має на меті і обмеження. Далі просувається ідея, що трикутні примітиви створюють ефективні транзакційні структури. Тріада компонентів домінує в фізичній та соціальній природі, як одна з найпоширеніших структур, що має властивості: 1) Транзитивність трикутного замикання є основою будь-якої метрики вимірювання процесів і явищ. 2) Трикутник – найпримітивніша геометрична фігура на поскості, а тетраедр, складений з трикутних площин, в просторі. 3) Всі геометричні фігури на площині можуть бути отримані за допомогою суперпозиції трикутних примітивів, а в просторі – шляхом використання тетраедрів. 4) Всі вершини в трикутнику суміжні, що означає – від кожної вершини існує мінімальний шлях до будь-якої іншої, чого немає ні в одному іншому геометричному примітиві. 5) Трикутник має мінімальну вартість міжз’єднань, що утворюють площину, яку можна прийняти за одиницю. 6) Комунікації, складені з трикутних структур, є оптимальними за критерієм: середня довжина шляху між будь-якими двома точками топології, віднесена до вартості її реалізації, чого немає ні в одному іншому геометричному примітиві. 7) Трикутник має мінімальну вартість міжз’єднань, що утворюють площину, яку можна прийняти за одиницю. 8) Комунікації, складені з трикутних структур, є оптимальними за критерієм: середня довжина шляху між будь-якими двома точками топології, віднесена до вартості її реалізації. чого немає ні в одному іншому геометричному примітиві. 9) Трикутник має

мінімальну вартість міжз’єднань, що утворюють площину, яку можна прийняти за одиницю. 7) Комунікації, складені з трикутних структур, є оптимальними за критерієм: середня довжина шляху між будь-якими двома точками топології, віднесена до вартості її реалізації.

На основі введеної раніше β-метріки можна формувати квазіоптимальний кіберпростір планети. Воно не прив’язане до геометричної топології поверхні і в цьому сенсі є віртуальним. Але в частині покриття поверхні і простору телекомунікаціями фізична основа Cyber Space повинна бути мажорірованою з метою забезпечення відмовостійкості та надійності функціонування каналів зв'язку при виникненні катаклізмів. Інформаційна структура Cyber Space повинна бути ієрархічною та замкненою як глобально, так і локально, на будь-якому рівні ієрархії. Елементарна комірка структури простору повинна бути трикутною. Це забезпечить суттєве зменшення інформаційних обсягів Cyber Space, у межі – на третину. Що означає підвищення продуктивності всіх приймально-передавачів і сховищ контенту планети на 33%? Дане твердження пов’язано з аксіомою транзитивного замикання кінцевого числа (1,2,3,4, ..., n) точок в інформаційному векторно-логічному просторі:

$$1)\ d_1 = 0;$$
$$2)\ d_1 \oplus d_2 = 0;$$
$$3)\ d_1 \oplus d_2 \oplus d_3 = 0;$$
$$4)\ d_1 \oplus d_2 \oplus d_3 \oplus d_4 = 0;$$
$$5)\ d_1 \oplus d_2 \oplus \ldots \oplus d_i \oplus \ldots \oplus d_n = 0.$$

Розглядаючи всі можливі варіанти транзитивного замикання з позиції мінімізації умов для відновлення інформації, очевидним є факт відновлення третьої сторони трикутника по двом відомим. В цьому випадку достатньо, наприклад, передати по каналу зв'язку дві сторони, щоб відновити третю. Скорочення переданих обсягів в даному випадку дорівнює 33%. У всіх інших

випадках зменшення обсягів інформації буде менш істотним. Наприклад, для замкнутого чотирикутника необхідно передати три відстані або сторони, щоб відновити четверту, виграш – 25%. Можлива площинна інтерпретація Cyber Space має вигляд, представлений на рис. 2.4.

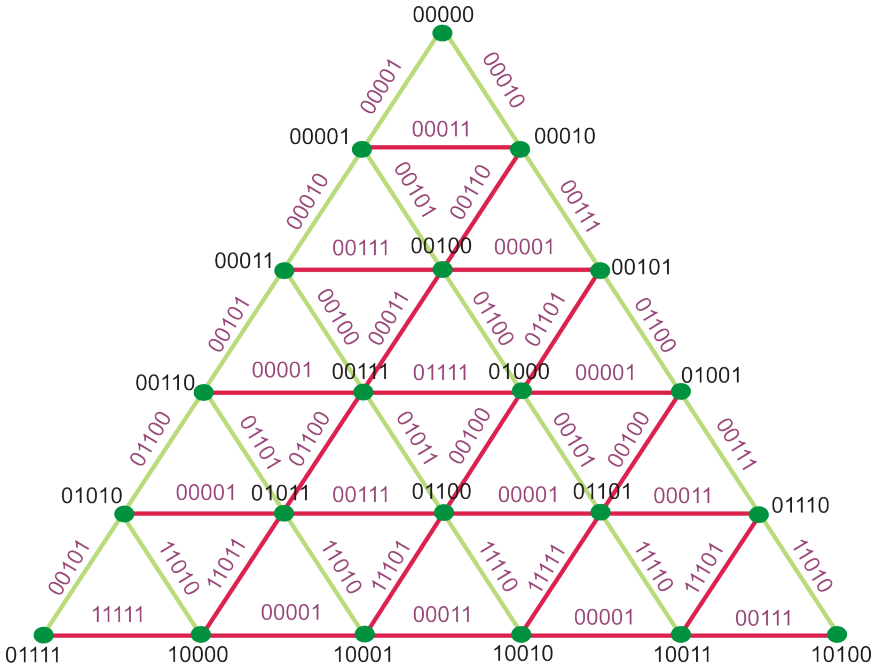



Рисунок 2.4 – Трикутний кіберпростір

Щоб відновити цю структуру, необхідно знати вже не 2/3 обсягу інформації, яка складена кожними двома сторонами трикутників. Досить вже мати сторони, задані зеленим кольором, що становить 44% від обсягу кодів-відстаней. Решта 56% компонентів трикутників можуть бути визначені за допомогою аксіоми:

$$d_1 \oplus d_2 \oplus d_3 = 0 \rightarrow d_3 = d_1 \oplus d_2.$$

У загальному випадку функціональна залежність відношення відновлюваних сторін замкнутого у трикутник простору до загальної кількості n шарів трикутної структури визначається як:

$$\eta = \frac{\sum(i+1)}{\sum(3\times i)}\Big|_{i=\overline{1,n}} = \frac{1}{3}\left(\frac{2}{n+1}+1\right) = \frac{1}{3}\cdot\frac{2+n+1}{n+1} = \frac{1}{3}\cdot\frac{n+3}{n+1} = \frac{n+3}{3(n+1)}.$$

Графік відношення кількості переданої інформації до її повного обсягу відстаней в трикутному кібер-просторі представлений на рис. 2.5. Граничне значення графіка наближається до 33%. Це означає – для великих обсягів трикутного простору досить передати третю частину всіх кодів-відстаней, щоб відновити повний простір кодів.

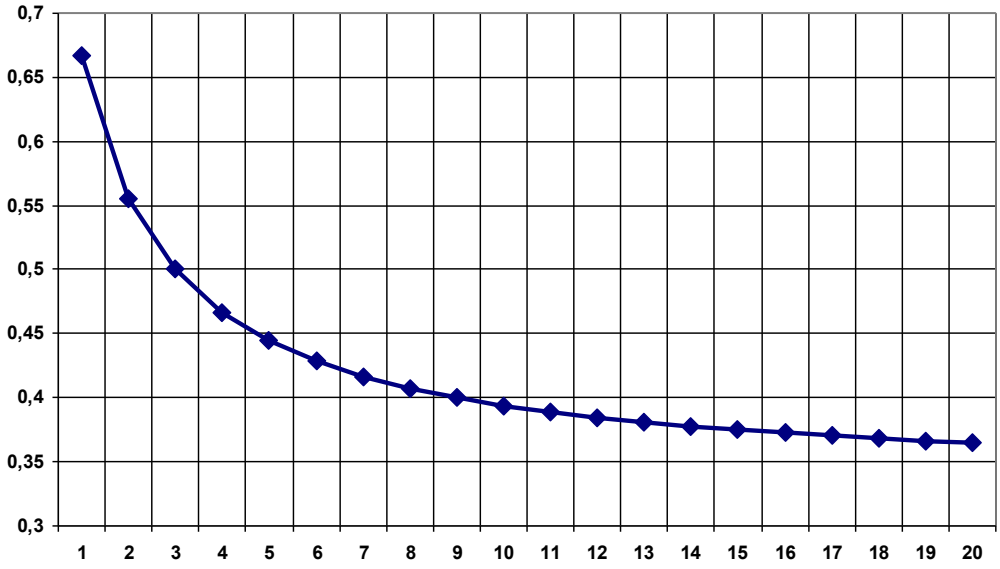



Рисунок 2.5 – Функція числа відстаней для відновлення простору

Трикутна метрика простору є самою економічною. Вона створює короткі відстані та шляхи між об’єктами завдяки наявності транзитивного замикання. Сусідство об’єкта з шістьма точками робить метрику більш кращою у порівнянні з топологією «Манхеттен» в частині прокладання шляху між двома точками. Оптимальна структура мультипроцесорної системи повинна бути складена з трикутників. Тут оптимальність визначається відношенням середньої довжини шляху між будь-якими двома точками до загальної кількості сторін, формує вартість топології. Довжина шляху у трикутній топології завжди буде не гірше, ніж в «Манхеттені», але загальне число зв’язків на одну діагональ більше в кожному чотирикутнику. Це дає можливість прямого доступу до 6 сусідів, що істотно для мультипроцесора. «Манхеттен» з діагональними зв’язками має ту ж довжину шляху, що і трикутний простір, маючи при цьому на один зв’язок більше. Тут

задіяна друга діагональ чотирикутника, що забезпечує сусідство з вісьмома вершинами простору. Оскільки трикутник – найпримітивніша фігура, яка формує площину, то вага однієї сторони, яку не слід зберігати для ідентифікації простору, буде максимальним у трикутнику. Система трикутників дає вже 66% структурних компонентів, які можна не описувати при формуванні простору для подальшого відновлення. Всі інші примітивні площинні фігури дають менший виграш при формуванні Cyber Space. Дуже важливо для підвищення надійності цифрових систем вміти відновлювати до 66 відсотків несправних компонентів без втрати інформації.

## 2.5 Спрощений критерій якості топології

Середня довжина шляху між компонентами топології (цифрової системи) є основний параметр, що впливає на швидкодію виконання транзакцій між вузлами структури. При розгляді варіантів реалізації топології комп'ютингової системи необхідно визначати інтегральну характеристику у вигляді суми всіх відстаней між кожною парою компонентів або вершин відповідного графа, наведеної до числа ребер або дуг. Тут інтерес представляють примітивні фігури: чотирикутник, трикутник і тетраедр. Останній має унікальну властивість – кожна вершина тетраедра має три сусідніх, в той час як трикутник має унікальні дві суміжні вершини на площині. Для оцінки вартості інформаційної транзакції між двома компонентами системи в графі, що містить n вершин, вводиться критерій якості структури, $\mathbf{minP_i}$ між усіма парами вершин $\mathbf{\frac{1}{2}(n^2 - n)}$ до кількості ребер топології k:

$$L = \frac{\sum_{i=1}^{\frac{1}{2}(n^2-n)} \min P_i}{\frac{1}{2}(n^2-n)} \times \frac{\frac{1}{2}(n^2-n)}{k} = \frac{\sum_{i=1}^{\frac{1}{2}(n^2-n)} \min P_i}{k} = \frac{1}{k}\sum_{i=1}^{\frac{1}{2}(n^2-n)} \min P_i.$$

Тут аргумент k – кількість ребер в графі є витратною частиною або вартісною характеристикою топологічної структури, тому вона фігурує в знаменнику. Але збільшення даного параметра може істотно зменшити

чисельник дробу, в якому записана інтегральна довжина мінімальних шляхів між усіма парами вершин графа, яка наближається до мінімуму. Таким чином, чим менше інтегральна вартість тразакції L, тим вище ефективність топологічної структури.

Приклад 1. Визначити критерії якості трьох структурних реалізацій, що мають 4 вершини-компонента (a, b, c, d): $\mathbf{G_1}$= {ab, bc, ad, cd}, $\mathbf{G_2}$ = {ab, ac, ad, cd, bc, bd}, $\mathbf{G_3}$= {ab, bc, ac, ad, cd}:

$G_1$ =

| | a | b | c | d |
|---|---|---|---|---|
| a | . | 1 | 1 | . |
| b | 1 | . | . | 1 |
| c | 1 | . | . | 1 |
| d | . | 1 | 1 | . |

$G_2$ =

| | a | b | c | d |
|---|---|---|---|---|
| a | . | 1 | 1 | 1 |
| b | 1 | . | 1 | 1 |
| c | 1 | 1 | . | 1 |
| d | 1 | 1 | 1 | . |

$G_3$ =

| | a | b | c | d |
|---|---|---|---|---|
| a | . | 1 | 1 | . |
| b | 1 | . | 1 | 1 |
| c | 1 | 1 | . | 1 |
| d | . | 1 | 1 | . |

Використовуючи наведену формулу, досить просто підрахувати відношення суми довжин мінімальних шляхів для кожної з 6 пар: {ab, ac, ad, cd, bc, bd}, з'єднаних k = (4,6,5) ребрами топології:

$$L_1 = \frac{1+2+1+1+2+1}{4} = \frac{8}{4} = 2{,}0;$$

$$L_2 = \frac{1+1+1+1+1+1}{6} = \frac{6}{6} = 1{,}0;$$

$$L_3 = \frac{1+1+1+1+2+1}{5} = \frac{7}{5} = 1{,}4.$$

Важливою характеристикою L-критерію є його чутливість, коли незначна модифікація графа – додавання одного ребра – може привести до істотної зміни оцінки в бік зменшення. Ринкова привабливість аналізу ефективності структур актуальна для транспортної інфраструктури міст при проектуванні і перебудові комунікацій.

Приклад 2. Показано застосування критерію якості до оцінювання трьох графових структур, що мають 6 вершин і різні топології з'єднання, представлені на рис. 2.1:

$G_1$ =

| | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| a | . | 1 | 1 | . | . | . |
| b | 1 | . | 1 | 1 | . | 1 |
| c | . | 1 | . | . | . | 1 |
| d | 1 | 1 | . | . | 1 | 1 |
| e | . | . | . | 1 | . | 1 |
| f | . | 1 | 1 | 1 | 1 | . |

$G_2$ =

| | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| a | . | 1 | . | 1 | . | . |
| b | 1 | . | 1 | . | 1 | . |
| c | . | 1 | . | . | . | 1 |
| d | 1 | . | . | . | 1 | . |
| e | . | 1 | . | 1 | . | 1 |
| f | . | . | 1 | . | 1 | . |

$G_3$ =

| | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| a | . | 1 | . | 1 | 1 | . |
| b | 1 | . | 1 | 1 | 1 | 1 |
| c | . | 1 | . | . | 1 | 1 |
| d | 1 | 1 | . | . | 1 | . |
| e | 1 | 1 | 1 | 1 | . | 1 |
| f | . | 1 | 1 | . | 1 | . |

Тут три графа мають 9, 7 і 11 дуг відповідно. Підрахунок критерію за L-формулою дає наступні результати:

$$L(G_1) = \frac{9\times1+6\times2}{9} = \frac{21}{9} = 2{,}33;$$

$$L(G_2) = \frac{7\times1+6\times2+2\times3}{7} = \frac{25}{7} = 3{,}57;$$

$$L(G_3) = \frac{11\times1+4\times2}{11} = \frac{19}{11} = 1{,}73.$$

Висновки: 1) Отриманий критерій якості топології досить просто реалізувати у хмарному сервісі, доступному для широкого кола користувачів, що бажають визначити ефективність старої або інноваційної структури комп’ютингової системи, університету, соціальної мережі, компанії або транспортної інфраструктури. 2) Критерій показує очевидні переваги трикутної топології взаємодії системних компонентів для обслуговування транзакційних процесів у кіберфізичному просторі. 3) Трикутна метрика простору є сама економічна. Вона створює найкоротші відстані та шляхи між координатами, завдяки наявності транзитивного замикання. Трикутна топологія міст виграє у метрики Манхеттена по скороченню маршрутів руху від 33 до 66 відсотків. 4) Впровадження даної топології в структурну організацію цифрових систем на 2D, 3D кристалах забезпечить скорочення

довжини з'єднувальних ліній і підвищення швидкодії не менше, ніж на 33 відсотки. 5) Дана топологія забезпечує підвищення надійності обчислювальних систем і мереж до 66 відсотків, завдяки трикутній організації зв'язків структурних компонентів, коли будь-який відмовивший компонент має не менше двох сусідів. 6) Об'ємна організація системних компонентів за трикутною тетраедрною топологією надає ще більші переваги щодо оптимізації транзакційних маршрутів для кіберфізичних об’єктів, включаючи авіатранспорт, глобальний комп’ютинг і 3D-SoC. 7) Трикутна інфраструктура забезпечує підвищення надійності цифрових систем завдяки можливості відновлення до 66 відсотків несправних зв’язків компонентів без втрати інформації. Інакше, 66% несправних зв'язків у трикутній топології системи залишає її в працездатному стані. 8) Недоліком трикутної топології можна вважати незвичайність для людини трикутної системи координат, що звик до декартовой метриці. Тут мова йде про ідентифікацію точки на площині трьома координатами. Декартова метрика передбачає використання двох координат для позиціонування об'єкта.

Приклад 3. Показано застосування критерію якості до оцінювання двох графових структур, що мають 21 вершину, які складають топології "Манхеттен" і "трикутник", рис. 2.6 і 2.7. Кожному рисунку ставиться у відповідність матриця смежностей (табл. 2.1, 2.3), яка визначає сусідство вершин в графі, а також матриця відстаней (табл. 2.2, 2.4), що задає довжину шляхів між кожною парою вершин. Щоб оцінити якість топології кожного графа з позиції довжини шляху, необхідно скласти всі мінімальні шляхи між парами вершин і розділити їх на кількість пар, яке визначається числом сполучень з 21 вершини по 2.

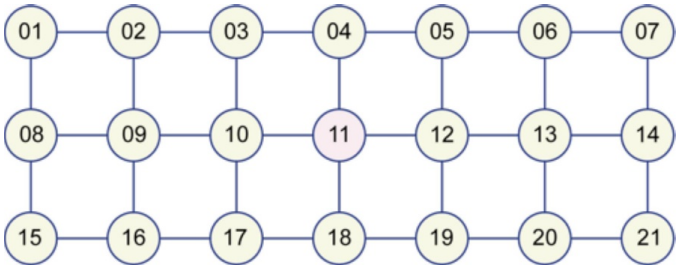



Рисунок 2.6 – Прямокутний граф топології "Манхеттен"

Таблиця 2.1 – Матриця суміжностей для прямокутного графа

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | | | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | | | | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 | | | | | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 | | | | | | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7 | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8 | | | | | | | | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 9 | | | | | | | | | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 10 | | | | | | | | | | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 11 | | | | | | | | | | | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 12 | | | | | | | | | | | | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 13 | | | | | | | | | | | | | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 14 | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 15 | | | | | | | | | | | | | | | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 16 | | | | | | | | | | | | | | | | 0 | 1 | 0 | 0 | 0 | 0 |
| 17 | | | | | | | | | | | | | | | | | 0 | 1 | 0 | 0 | 0 |
| 18 | | | | | | | | | | | | | | | | | | 0 | 1 | 0 | 0 |
| 19 | | | | | | | | | | | | | | | | | | | 0 | 1 | 0 |
| 20 | | | | | | | | | | | | | | | | | | | | 0 | 1 |
| 21 | | | | | | | | | | | | | | | | | | | | | 0 |

Таблиця 2.2 – Таблиця відстаней для прямокутного графа

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 2 | | 0 | 1 | 2 | 3 | 4 | 5 | 2 | 1 | 2 | 3 | 4 | 5 | 6 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 3 | | | 0 | 1 | 2 | 3 | 4 | 3 | 2 | 1 | 2 | 3 | 3 | 5 | 4 | 3 | 2 | 3 | 4 | 5 | 6 |
| 4 | | | | 0 | 1 | 2 | 3 | 4 | 3 | 2 | 1 | 2 | 3 | 4 | 5 | 4 | 3 | 2 | 3 | 4 | 5 |
| 5 | | | | | 0 | 1 | 2 | 5 | 4 | 3 | 2 | 1 | 2 | 3 | 6 | 5 | 4 | 3 | 2 | 3 | 4 |
| 6 | | | | | | 0 | 1 | 6 | 5 | 4 | 3 | 2 | 1 | 2 | 7 | 6 | 5 | 4 | 3 | 2 | 3 |
| 7 | | | | | | | 0 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 8 | 7 | 6 | 5 | 4 | 3 | 2 |
| 8 | | | | | | | | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 9 | | | | | | | | | 0 | 1 | 2 | 3 | 4 | 5 | 2 | 1 | 2 | 3 | 4 | 5 | 6 |
| 10 | | | | | | | | | | 0 | 1 | 2 | 3 | 4 | 3 | 2 | 1 | 2 | 5 | 4 | 5 |
| 11 | | | | | | | | | | | 0 | 1 | 2 | 3 | 4 | 3 | 2 | 1 | 2 | 3 | 4 |
| 12 | | | | | | | | | | | | 0 | 1 | 2 | 5 | 4 | 3 | 2 | 1 | 2 | 3 |
| 13 | | | | | | | | | | | | | 0 | 1 | 6 | 5 | 4 | 3 | 2 | 1 | 2 |
| 14 | | | | | | | | | | | | | | 0 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| 15 | | | | | | | | | | | | | | | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
| 16 | | | | | | | | | | | | | | | | 0 | 1 | 2 | 3 | 4 | 5 |
| 17 | | | | | | | | | | | | | | | | | 0 | 1 | 2 | 3 | 4 |
| 18 | | | | | | | | | | | | | | | | | | 0 | 1 | 2 | 3 |
| 19 | | | | | | | | | | | | | | | | | | | 0 | 1 | 2 |
| 20 | | | | | | | | | | | | | | | | | | | | 0 | 1 |
| 21 | | | | | | | | | | | | | | | | | | | | | 0 |

Для прямокутної топології графа критерій якості визначається наступною оцінкою: L = sum / k, k = 32, Sum = 714; L = 714/32 = 22,3125.

Таблиця 2.3 – Матриця суміжностей для трикутного графа

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | | | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | | | | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 | | | | | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 | | | | | | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7 | | | | | | | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8 | | | | | | | | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 9 | | | | | | | | | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | | | | | | | | | | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11 | | | | | | | | | | | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 12 | | | | | | | | | | | | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 13 | | | | | | | | | | | | | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 14 | | | | | | | | | | | | | | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 |
| 15 | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 16 | | | | | | | | | | | | | | | | 0 | 1 | 0 | 0 | 0 | 0 |
| 17 | | | | | | | | | | | | | | | | | 0 | 1 | 0 | 0 | 0 |
| 18 | | | | | | | | | | | | | | | | | | 0 | 1 | 0 | 0 |
| 19 | | | | | | | | | | | | | | | | | | | 0 | 1 | 0 |
| 20 | | | | | | | | | | | | | | | | | | | | 0 | 1 |
| 21 | | | | | | | | | | | | | | | | | | | | | 0 |

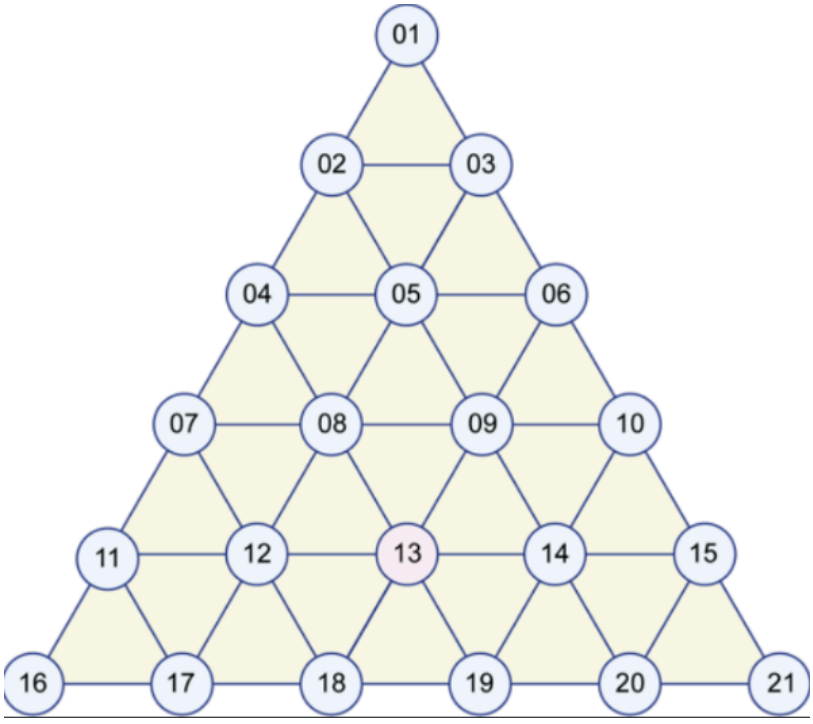



Рисунок 2.7 – Трикутний граф

Таблиця 2.4 – Таблиця відстані для трикутного графа

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 1 | 2 | 2 | 2 | 3 | 3 | 3 | 3 | 4 | 4 | 4 | 4 | 4 | 5 | 5 | 5 | 5 | 5 | 5 |
| 2 | | 0 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 3 | 3 | 3 | 3 | 3 | 4 | 4 | 4 | 4 | 4 | 4 | 5 |
| 3 | | | 0 | 0 | 1 | 1 | 3 | 2 | 2 | 2 | 4 | 3 | 3 | 3 | 4 | 5 | 4 | 4 | 4 | 4 | 4 |
| 4 | | | | 0 | 1 | 2 | 1 | 1 | 2 | 3 | 2 | 2 | 2 | 3 | 4 | 3 | 3 | 3 | 3 | 4 | 5 |
| 5 | | | | | 0 | 1 | 2 | 1 | 1 | 2 | 3 | 2 | 2 | 2 | 3 | 4 | 3 | 3 | 3 | 3 | 4 |
| 6 | | | | | | 0 | 3 | 2 | 1 | 1 | 4 | 3 | 2 | 2 | 2 | 5 | 4 | 3 | 3 | 3 | 3 |
| 7 | | | | | | | 0 | 1 | 2 | 3 | 1 | 1 | 2 | 3 | 4 | 2 | 2 | 2 | 3 | 4 | 5 |
| 8 | | | | | | | | 0 | 1 | 2 | 2 | 1 | 1 | 2 | 3 | 3 | 2 | 2 | 2 | 3 | 4 |
| 9 | | | | | | | | | 0 | 1 | 3 | 2 | 1 | 1 | 2 | 4 | 3 | 2 | 2 | 2 | 3 |
| 10 | | | | | | | | | | 0 | 4 | 3 | 2 | 1 | 1 | 5 | 4 | 3 | 2 | 2 | 2 |
| 11 | | | | | | | | | | | 0 | 1 | 2 | 3 | 4 | 1 | 1 | 2 | 3 | 4 | 5 |
| 12 | | | | | | | | | | | | 0 | 1 | 2 | 3 | 2 | 1 | 1 | 2 | 3 | 4 |
| 13 | | | | | | | | | | | | | 0 | 1 | 2 | 3 | 2 | 1 | 1 | 2 | 3 |
| 14 | | | | | | | | | | | | | | 0 | 1 | 4 | 3 | 2 | 1 | 1 | 2 |
| 15 | | | | | | | | | | | | | | | 0 | 5 | 4 | 3 | 2 | 1 | 1 |
| 16 | | | | | | | | | | | | | | | | 0 | 1 | 2 | 3 | 4 | 5 |
| 17 | | | | | | | | | | | | | | | | | 0 | 1 | 2 | 3 | 4 |
| 18 | | | | | | | | | | | | | | | | | | 0 | 1 | 2 | 3 |
| 19 | | | | | | | | | | | | | | | | | | | 0 | 1 | 2 |
| 20 | | | | | | | | | | | | | | | | | | | | 0 | 1 |
| 21 | | | | | | | | | | | | | | | | | | | | | 0 |

Для трикутної топології графа критерій якості визначається наступною оцінкою: L = sun / k, k = 45, Sum = 545 L = 545/45 = 12,1. Це означає, що дорожня інфраструктура, побудована на трикутних з’єднаннях пунктів відправлення і прибуття в два рази ефективніше, ніж загальноприйнята топологія "Манхеттен", що має оцінку L = 22,3125.

## 2.6 Наведена метрика топології графа

Раніше згадувалося, що середня довжина шляху між компонентами топології є основний параметр, що впливає на час виконання маршруту між вузлами інфраструктури. Для топології, яка має 4 компоненти (a, b, c, d), на площині досить просто підрахувати відношення кількості пар вершин до суми довжин мінімальних шляхів на кожній з n пар: ab, ac, ad, bc, bd, cd, з'єднаних k ребрами топології:

$$L = \frac{k}{\frac{1}{2}(n^2-n)} \times \frac{\frac{1}{2}(n^2-n)}{\sum_{i=1}^{n} \min P_i}.$$

Перший співмножник визначає приведену до повного графу (кожна вершина безпосередньо пов'язана з іншими) вартість міжз’єднань структури, а другий – формує зворотну оцінку середньої довжини шляху для кожної пари вершин графа. Обидва співмножника визначені в інтервалі (0,1), де краще рішення задається оцінкою L=1. Тут $\min P_i$ визначається шляхом використання модифікованого методу Дейкстри, наведеного в наступному розділі. Обчислення оцінок для трьох топологій системних зв'язків дають наступний результат: 1) чотирикутник: L = 4 / (1 + 2 + 1 + 1 + 2 + 1) = 0,50; 2) трикутна структура, яка містить два примітиви: L = 5 / (1 + 1 + 1 + 1 + 2 + 1) = 0,71; 3) чотирикутник з двома діагоналями: L = 6 / (1 + 1 + 1 + 1 + 1 + 1) = 1,0.

Далі можна визначити інтервальний [0,1] критерій структурної ефективності, як число n всіх можливих пар (сполучень по два) на фіксованій множині вершин графа, віднесене до суми мінімальних відстаней між ними:

$$Q = \sum_{i=1}^{n} \frac{1}{\min p_i} = \frac{n}{\sum_{i=1}^{n} \min p_i}.$$

В цьому випадку наведені вище оцінки матимуть вигляд:

$$Q_1 = \left[\frac{1}{1} + \frac{1}{1} + \frac{1}{1} + \frac{1}{1} + \frac{1}{2} + \frac{1}{2}\right] = \frac{6}{8} = 0{,}75$$
$$Q_2 = \frac{1}{7}\left[\frac{1}{1} + \frac{1}{1} + \frac{1}{1} + \frac{1}{1} + \frac{1}{1} + \frac{1}{1}\right] = \frac{6}{6} = 1{,}0$$
$$Q_3 = \left[\frac{1}{1} + \frac{1}{1} + \frac{1}{1} + \frac{1}{1} + \frac{1}{1} + \frac{1}{2}\right] = \frac{6}{7} = 0{,}86$$

Платою за підвищення ефективності топології служить структурна складність проекту, яка визначається кількістю дуг n.

Таким чином, ефективність дорожньої інфраструктури є відношення кількості (ребер) парних з’єднань n компонентів до суми мінімальних шляхів (досяжностей p) кожної пари:

$$Q_t = \frac{\frac{1}{2}(n^2 - n)}{\sum_{i=1}^{n} \min_j (p_{ij})}.$$

Дана оцінка буде дорівнювати одиниці, якщо відповідний інфраструктурі граф буде повним, коли дві будь-які вершини є суміжними. Однак реалізація такого графа має високу вартість, що призводить до необхідності ввести оцінку складності інфраструктури як відношення реальної кількості E ребер графа, що містить n вершин до повного графу:

$$Q_s = \frac{E}{\frac{1}{2}(n^2 - n)}.$$

Нехай є три структури, що містять 4 вершини, представлені у формі матриці суміжностей:

$G_1$

| | a | b | c | d |
|---|---|---|---|---|
| a | . | 1 | 1 | . |
| b | 1 | . | . | 1 |
| c | 1 | . | . | 1 |
| d | . | 1 | 1 | . |

$G_2$

| | a | b | c | d |
|---|---|---|---|---|
| a | . | 1 | 1 | 1 |
| b | 1 | . | 1 | 1 |
| c | 1 | 1 | . | 1 |
| d | 1 | 1 | 1 | . |

$G_3$

| | a | b | c | d |
|---|---|---|---|---|
| a | . | 1 | 1 | . |
| b | 1 | . | 1 | 1 |
| c | 1 | 1 | . | 1 |
| d | . | 1 | 1 | . |

Обчислення двох оцінок, введених вище, дає наступний результат:

$$Q_t(G_1) = \frac{6}{8} = 0{,}75 \quad Q_s(G_1) = \frac{4}{6} = 0{,}66$$
$$Q_t(G_2) = \frac{6}{6} = 1{,}0 \quad Q_s(G_2) = \frac{6}{6} = 1{,}0$$
$$Q_t(G_3) = \frac{6}{7} = 0{,}86 \quad Q_s(G_3) = \frac{5}{6} = 0{,}83$$

Якщо мультиплікувати обидві оцінки, то вийде узагальнений критерій ефективності інфраструктури для довільного числа компонентів або вершин, який залежить від кількості з’єднань E, віднесених до суми мінімальних шляхів всіх пар вершин:

$$Q = \frac{\frac{1}{2}(n^2 - n)}{\sum_{i=1}^{n} \min_{j}(p_{ij})} \times \frac{E}{\frac{1}{2}(n^2 - n)} = \frac{E}{\sum_{i=1}^{n} \min_{j}(p_{ij})}.$$

Перерахунок критерію ефективності для трьох варіантів структури представлений наступними оцінками:

$$Q(G_1) = \frac{4}{8} = 0{,}5;$$
$$Q(G_2) = \frac{6}{6} = 1{,}0;$$
$$Q(G_3) = \frac{5}{7} = 0{,}71.$$

Подальша модифікація критерію спрямована на отримання зваженої оцінки, яка враховує вартість шляху (середньої довжини маршруту між двома вершинами), наведеної до витрат – фактичного числа ребер графа апаратної реалізації обчислювальної структури:

$$Q = \frac{E}{\frac{1}{2}(n^2 - n)} \times \left[ \frac{1}{n \times E} \times \sum_{i=1}^{n} \min_{j}(p_{ij}) \right] = \frac{2 \times \sum_{i=1}^{n} \min_{j}(p_{ij})}{n \times (n^2 - n)} = \frac{2 \times \sum_{i=1}^{n} \min_{j}(p_{ij})}{n^2 (n - 1)}.$$

Тут перший співмножник $\frac{E}{\frac{1}{2}(n^2 - n)}$ – витратна частина або число дуг, що формують граф, віднесене до максимально можливого їх числа повного графа, а другий співмножник $\frac{1}{n \times E} \times \sum_{i=1}^{n} \min_{j}(p_{ij})$ є середня довжина шляху між двома вершинами, приведена до максимально можливого маршруту, складеного з усіх дуг повного графа. Таке приведення (поділ на число дуг повного графа) дає можливість нормувати другий співмножник, який без приведення завжди не менш одиниці, до одиничного інтервалу. Однак незручність використання даного критерію пов’язано з характером функцій,

що задаються першим і другим співмножники, які, по суті, працюють в протифазі. Якщо вартість графа збільшується, то середня довжина шляху зменшується, що призводить до зменшення критерію. Для усунення даного недоліку і з метою отримання оцінки, коли її значення, чим ближче до 1, тим краще, необхідна її модифікація щодо другого співмножника:

$$Q = \frac{E}{\frac{1}{2}(n^2 - n)} \times \left[1 - \frac{1}{n \times E} \times \sum_{i=1}^{n} \min_{j}(p_{ij})\right] = \frac{2 \times E}{(n^2 - n)} \times \left[1 - \frac{1}{n \times E} \times \sum_{i=1}^{n} \min_{j}(p_{ij})\right].$$

Несуттєве ускладнення оцінки призводить до характеру функціональної залежності, коли довжина з'єднання (другий співмножник) прямо пропорційно залежить не тільки від числа дуг у відповідному графі (перший співмножник), але і від яких саме дуг, що роблять шлях між двома вершинами коротше. Важливою характеристикою даної оцінки є її чутливість, коли незначна модифікація графа – додавання одного ребра – може привести до істотної зміни оцінки в сторону збільшення. Ринкова привабливість аналізу ефективності структур актуальна для інфраструктури міст в умовах існування транспортних заторів:

$$Q_1 = \frac{E}{\sum_{i=1}^{n} \min_{j}(p_{ij})};$$

$$Q_2 = \frac{2 \times \sum_{i=1}^{n} \min_{j}(p_{ij})}{n^2(n-1)};$$

$$Q_3 = \frac{2 \times E}{(n^2 - n)} \times \left[1 - \frac{1}{n \times E} \times \sum_{i=1}^{n} \min_{j}(p_{ij})\right].$$

Ще одна модифікація критерію пов'язана зі спрощенням обробки графової структури, що має E дуг і n вершин. Його особливість полягає в

обчисленні абсолютного і не наведеного до інтервалу значення, який формується вартістю з'єднань, помноженої на якість транзакцій між усіма парами вершин:

$$Q_4 = \frac{E}{n} \times \sum_{i=1}^{n} \min_j (p_{ij}).$$

Застосування цієї формули до оцінювання трьох графових структур, що мають 6 вершин і різні топології з'єднання представлені нижче (див. рис. 2.1):

$G_1$ =

| | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| a | . | 1 | 1 | . | . | . |
| b | 1 | . | 1 | 1 | . | 1 |
| c | . | 1 | . | . | . | 1 |
| d | 1 | 1 | . | . | 1 | 1 |
| e | . | . | . | 1 | . | 1 |
| f | . | 1 | 1 | 1 | 1 | . |

$G_2$ =

| | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| a | . | 1 | . | 1 | . | . |
| b | 1 | . | 1 | . | 1 | . |
| c | . | 1 | . | . | . | 1 |
| d | 1 | . | . | . | 1 | . |
| e | . | 1 | . | 1 | . | 1 |
| f | . | . | 1 | . | 1 | . |

$G_3$ =

| | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| a | . | 1 | . | 1 | 1 | . |
| b | 1 | . | 1 | 1 | 1 | 1 |
| c | . | 1 | . | . | 1 | 1 |
| d | 1 | 1 | . | . | 1 | . |
| e | 1 | 1 | 1 | 1 | . | 1 |
| f | . | 1 | 1 | . | 1 | . |

Тут три графа мають 9, 7 і 11 дуг відповідно. Підрахунок критерію відповідно до останньої формули дає наступні результати:

$$Q_4(G_1) = \frac{E}{n} \times \sum_{i=1}^{n} \min_j (p_{ij}) = \frac{9}{6} \times (9 \times 1 + 6 \times 2) = 31{,}5;$$

$$Q_4(G_2) = \frac{7}{6} \times (7 \times 1 + 6 \times 2 + 2 \times 3) = 29{,}2;$$

$$Q_4(G_3) = \frac{11}{6} \times (11 \times 1 + 4 \times 2) = 34{,}8.$$

Модифікація оцінки ефективності топології пов'язана з приведенням реальних витрат (число дуг E) до максимально можливої кількості $V = \frac{n^2 - n}{2}$

парних з’єднань в графі $\frac{E}{V}$, що забезпечує якість комунікаційних властивостей $\frac{V}{\sum_{i=1}^{V}\min_{j}(p_{ij})}$:

$$Q_4 = \frac{E}{\frac{n^2-n}{2}} \times \frac{\frac{n^2-n}{2}}{\sum_{i=1}^{\frac{n^2-n}{2}}\min_{j}(p_{ij})} = \frac{E}{\sum_{i=1}^{\frac{n^2-n}{2}}\min_{j}(p_{ij})} = \frac{E}{\sum_{i=1}^{V}\min_{j}(p_{ij})}.$$

Оцінка дорівнює одиниці, якщо чисельник і знаменник рівні $V = \frac{n^2-n}{2}$.

У даному випадку графова структура має всі можливі парні сполуки між вершинами графа, які визначаються половиною декартова квадрата потужності множини вершин за вирахуванням n вершин. Віднімання визначається вершинами графа, що не мають ідемпотентних замикань. При цьому кожна пара вершин має довжину шляху, що дорівнює одиниці. Перерахунок критеріїв ефективності, дає наступний результат:

$$Q_5(G_1) = \frac{E}{\sum_{i=1}^{V=12}\min_{j}(p_{ij})}. = \frac{9}{9\times 1 + 6\times 2} = 0{,}428;$$

$$Q_5(G_2) = \frac{7}{7\times 1 + 6\times 2 + 2\times 3} = 0{,}28;$$

$$Q_5(G_3) = \frac{11}{11\times 1 + 4\times 2} = 0{,}578.$$

При цьому платою за якість комунікацій є потужність з’єднань, приведена до максимально можливої кількості ребер:

$$H_5(G_1) = \frac{E}{\frac{n^2 - n}{2}} = \frac{9}{15} = 0{,}60;$$

$$H_5(G_2) = \frac{7}{15} = 0{,}46;$$

$$H_5(G_3) = \frac{11}{15} = 0{,}73.$$

Доцільно мати дві оцінки: інтегральний критерій якості комунікацій, який неявно визначає витрати часу на середню досяжність між кожною парою вершин графових структури, а також наведену до максимально можливого числа потужність з'єднань, яка демонструє вартість якості інфраструктури системи, що має цільову функцію, що мінімізувала середню досяжність (довжину шляху або часу) між парою вершин графової структури. Залежно від числа з'єднань перший критерій має тенденцію до зростання від 0 до 1, другий також збільшується за мірою збільшення числа ребер в графі. Тому мультиплікування двох критеріїв не дає нової властивості при оцінюванні інфраструктури досяжностей кожної пари вершин. Висновки: 1) Необхідно використовувати обидва критерії для оцінювання структурного проекту. 2) Слід модифікувати алгоритм Дейкстра для обчислення середнього значення досяжностей між парою вершин в графі, що і представлено нижче.

## 2.7 Висновки до розділу 2

1) Розроблено метрику і критерії практичного оцінювання маршрутів руху і дорожніх інфраструктур для реалізації хмарних сервісів управління дорожнім рухом, що надаються водієві у режимі реального часу.

2) Запропоновано критерії оцінювання якості інфраструктури дорожнього руху, які враховують середнє значення довжини шляху в інфраструктурі, а також вартість створення дорожньої структури.

3) Розроблено аналітичні моделі прохідності дорожньої інфраструктури, які враховують поняття спостереження та управління для формування алгоритмів управління на основі використання світлофорів.

4) Запропоновано метрики для вимірювання е-інфраструктури дорожнього руху, які дають можливість створювати ефективні маршрути руху на основі використання трикутних примітивів. Представлені приклади різних топологій і зв'язків компонентів інфраструктури.

5) Створено систему спрощених і наведених критеріїв оцінювання якості топології дорожньої інфраструктури, яка орієнтована на прокладання хмарних маршрутів в реальному масштабі часу.

# 3 МОДИФІКАЦІЯ АЛГОРИТМУ ДЕЙКСТРИ ДЛЯ ПОШУКУ МАРШРУТУ

Розглядається алгоритм побудови оптимального маршруту на дорожній інфраструктурі між двома координатами, які ототожнюються з пунктом відправлення та прибуття. Можливі обмеження пов’язані з дорожніми колізіями, аваріями і ремонтом доріг, що також має бути враховано при виборі оптимального маршруту. Передбачається, що реалізація алгоритму буде представлена як online хмарний сервіс для водіїв транспортних засобів в рамках створення кіберфізічної системи інтелектуального хмарного управління транспортом. Алгоритм використовується для обчислення всіх можливих шляхів або відстаней між кожною парою компонентів або вершин відповідної топології.

## 3.1 Вступ

Підстави для представлених нижче досліджень відображені у наступних публікаціях і розробках:

– теоретичні розробки інтелектуальних моделей, методів і програмно-апаратних засобів аналізу кіберпростору, пов’язані з дискретною оптимізацією, пошуком, розпізнаванням та прийняттям рішень [97-100];

– досвід розробки та застосування вбудованих і RFID цифрових систем для моніторингу дорожнього руху [101-105];

– досвід розробки та впровадження програмних продуктів і хмарних сервісів для оптимізації маршрутів транспортних засобів українських підприємств з метою мінімізації матеріальних та часових витрат і підвищення якості обслуговування пасажирів [106-111];

– розробки розподіленої системи управління дорожнім рухом в умовах великих міст і мегаполісів на основі високонадійної обчислювальної техніки [107-109], прототипування розумних кіберфізічних систем управління трафіком [132, 133];

– методи оцінювання обчислювальних структур і пошуку найкоротших шляхів між парою вершин [135-138], відомі модифікації алгоритму Дейкстра [134-139].

Мета дослідження – синтез оптимального маршруту руху транспортного засобу шляхом використання модифікованого алгоритму Дейкстри на топологічній інфраструктурі дорожнього руху, що приводиться до графу, з урахуванням існування дорожніх колізій. Завдання: 1) Визначення всіх найкоротших шляхів і відповідних відстаней з вершини a до всіх інших вершин інфраструктури. 2) Для графа з одиничними ваговими коефіцієнтами, заданого матрицею суміжності, знайти всі найкоротші відстані між усіма парами вершин. 3) Для різних типів графів, заданих матрицями суміжності, знайти найкоротші відстані між усіма парами вершин і оцінити інфраструктури за розробленими критеріями якості. 4) Виконати порівняння різних архітектур і графових структур з метою визначення ефективних рішень для скорочення часу руху за замовленим маршрутом.

Відоме удосконалення алгоритму Дейкстри відшукання ланцюгів найменшої довжини з обраної вершини графа (пункту відправлення) до всіх інших його вершин (можливих пунктів прибуття) пов’язане зі скороченням кількості операцій (додавань і порівнянь) і збереженням отриманої інформації на одному з етапів для наступних обчислень. Це досягається процедурою розміщення міток алгоритму Дейкстри, яка зменшує складність алгоритму до $O(n^2)$. Відомо, що при визначенні найкоротших шляхів між усіма парами вершин використовується алгоритм Флойда-Уоршелла, складність якого становить $O(n^3)$.

## 3.2 Метод пошуку найкоротших шляхів у дорожній інфраструктурі

Завдання 3.1. Для графа G1, представленого на рис. 3.10, визначити всі найкоротші шляхи і відповідні їм відстані з вершини a до всіх інших вершин.

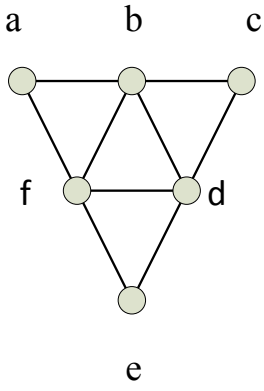



Рисунок 3.10 – Граф G1 – конфігурація мережі

Розв’язок. Розглянутий на рис. 3.10 граф є неорієнтованим. Він має дві рівноправні між собою групи вершин: 1) a, c, e; 2) b, d, f. «Рівноправність» розуміється в сенсі інваріантності уявлення найкоротших шляхів і їх дерев при пошуку з вершин a, c, e, що породжує першу групу шляхів, або з вершин b, d, f – другу групу шляхів. Отже, для даної задачі замість алгоритму Флойда можна двічі послідовно застосувати алгоритм Дейкстри, щоб знайти найкоротші шляхи між усіма парами вершин.

Отже, завдання розподіляється на дві підзадачі: 1) знайти найкоротші відстані і вказати всі найкоротші шляхи від вершини a до всіх інших вершин; 2) знайти найкоротші відстані і вказати всі найкоротші шляхи від вершини b до всіх інших вершин.

Матриця суміжності графа G1 має вигляд:

$$G_1 = \begin{array}{c|cccccc} & a & b & c & d & e & f \\ \hline a & . & 1 & . & . & . & 1 \\ b & 1 & . & 1 & 1 & . & 1 \\ c & . & 1 & . & 1 & . & . \\ d & . & 1 & 1 & . & 1 & 1 \\ e & . & . & . & 1 & . & 1 \\ f & 1 & 1 & . & 1 & 1 & . \end{array} \qquad (3.1)$$

*Підзадача* 3.1.1. Знайти найкоротші відстані і вказати всі відповідні їм найкоротші шляхи від вершини a до всіх інших вершин графа G1 (див. рис. 3.10).

В процесі реалізації алгоритму Дейкстри заповнюється таблиця, кількість рядків і стовпців якої визначається потужністю множини вершин графа, тобто 6x6. У заголовках рядків таблиці вказуються вершини, до яких належить знайти найкоротшу відстань (табл. 3.1).

Таблиця 3.1 – Обчислення найкоротших відстаней в графі G1 з вершини a

| | **1** | **2** | **3** | **4** | **5** |
|---|---|---|---|---|---|
| | **u = a**<br>**r = 0** | **u = b**<br>**r = 1** | **u = f**<br>**r = 1** | **u = c**<br>**r = 2** | **u = d**<br>**r = 2** |
| **b** | a, 1 | | | | |
| **c** | a, ∞ | b, 2 | b, 2 | | |
| **d** | a, ∞ | b, 2 | b, 2 | b, 2 | |
| **e** | a, ∞ | a, ∞ | f, 2 | f, 2 | f, 2 |
| **f** | a, 1 | a, 1 | | | |

Коментарі до заповнення табл. 3.1:

1) У заголовку першого стовпчика як пункт відправлення вказується початкова вершина u = a, пройдена відстань приймається рівною нулю: r = 0. Кожне ребро ідентифікується стовпчиковою (початок) і рядковою (кінцевою) вершинами і читається «стовпець – рядок». При заповненні першого стовпчика враховуються такі правила: початкова вершина a не змінюється і повторюється в кожному осередку; відстань збігається з довжиною ребра, яке з'єднує початкову (стовпчикову) вершину з вершиною з рядка, якщо таке ребро не існує, то відстані присвоюється тимчасова мітка $\infty$.

Після заповнення першого стовпчика з усіх кінцевих числових міток вибирається мінімальна – 1 (якщо таких декілька, то вибирається будь-яка з них). Їй відповідає постійно позначена вершина в рядку – b. Це означає, що найкоротша відстань до вершини b вже знайдена і дорівнює 1. Постійно

позначена вершина приймається як новий відправний пункт u=b і разом з мінімальною відстанню r=1 заноситься до заголовка чергового стовпчика.

2) При заповненні другого стовпчика таблиці числові мітки модифікуються згідно з правилом: до довжини ребра (b, j) додається пройдена відстань 1 і результат порівнюється з попередньою числовою міткою в рядку; якщо нова сумарна відстань менше попередньої, то вона приймається як поточна, при цьому вершина змінюється на вершину з поточного стовпчика. Так, ребро (b, c) має довжину 1, тоді пройдена відстань r = 1 + 1 = 2 має кінцеве значення, отже, вона приймається як поточна замість $\infty$. При цьому вершина a змінюється на вершину b з поточного стовпчика. Ребро (b, f) має довжину 1, тоді пройдена відстань r = 1 + 1 = 2 > 1, тому в рядку f зберігається попередня числова мітка, позначення вершини при цьому також не змінюється.

З усіх кінцевих числових міток другого стовпчика вибирається мінімальна – 1. Їй відповідає вершина f. Тепер вершина u = f є постійно поміченою, вона обирається як новий відправний пункт і запам’ятовується у заголовку третього стовпця одночасно з пройденою відстанню r = 1.

3) Третій стовпець заповнюється аналогічно попередньому. Тут модифікується мітка в рядку e. Всі числові мітки виявляються однаковими, тому обирається будь-яка. Нехай це буде u = c, яка тепер є постійно поміченою, розглядається як новий відправний пункт і записується до заголовку четвертого стовпчика разом з пройденою відстанню r = 2.

4) У четвертому стовпці мітки не змінюються, тому вибирається вершина u=d. Вона тепер постійно позначена, приймається як новий відправний пункт і зберігається в заголовку у наступній колонці поряд з пройденою відстанню r = 2.

5) В п’ятому стовпці числова мітка в рядку e не змінюється, оскільки ребро (d, e) має довжину 1, і сумарна пройдена відстань більше попередньої: 1+2=3>2. Таким чином, залишається єдина постійно позначена вершина. Отже, процес пошуку мінімальних відстаней завершується.

Примітки: 1) Якщо в деякому стовпці зустрічаються дві або більше вершини з однаковими мінімальними числовими позначками, то вибирається будь-яка з них. Це означає, що можливо існують дві різні ланцюги однакової довжини. 2) Якщо при обчисленні поточних числових міток нова сумарна відстань збігається з попередньою, то зберігається стара числова мітка. 3) Якщо граф орієнтований, то при обчисленні числових міток враховується напрямок ребра: ребро з вершини $x_i$ до вершини $x_j$ існує, якщо в графі є дуга $(x_i, x_j)$, Тоді можна врахувати її довжину. Але ребро $(x_j, x_i)$, при цьому може бути відсутнім, тоді відповідна позначка приймається рівною нескінченності. 4) Постійно помічені вершини в заголовках стовпців не повторюються. 5) Відстані в заголовках стовпців не зменшуються, тобто розташовуються в порядку $\leq$. 6) Поточні числові мітки в рядку не зростають ($\geq$).

Таким чином, з табл. 3.1 можна отримати інформацію про всі найкоротші ланцюги і відповідні їм довжини.

Наприклад, потрібно знайти найкоротший ланцюг з вершини a до вершини e. Послідовність вершин у ланцюзі виписується з кінця: останній заповнений осередок в рядку e містить інформацію про довжину найкоротшого ланцюга r = 2 і передостанній вершині в цьому ланцюзі – f. Інформація про попередню вершину знаходиться в останній комірці рядка f, в даному випадку – вершина a, яка є початком маршруту:

$$a \xrightarrow{1} f \xrightarrow{1} e \Rightarrow dist = 1 + 1 = 2.$$

Дані для всіх найкоротших ланцюгів представлені у табл. 3.2.

Таблиця 3.2 – Найкоротші ланцюга з вершини a для графа G1

| Ланцюг | Довжина |
|---|---|
| $a \xrightarrow{1} b$ | $r(a,b)=1$ |
| $a \xrightarrow{1} b \xrightarrow{1} c$ | $r(a,c)=1+1=2$ |
| $a \xrightarrow{1} b \xrightarrow{1} d$ | $r(a,d)=1+1=2$ |
| $a \xrightarrow{1} f \xrightarrow{1} e$ | $r(a,e)=1+1=2$ |
| $a \xrightarrow{1} f$ | $r(a,f)=1$ |

Граф, який ілюструє дерево найкоротших ланцюгів з вершини a, зображений на рис. 3.11.

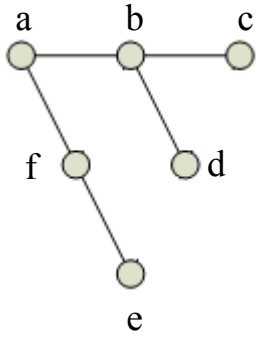



Рис. 3.11 – Дерево найкоротших ланцюгів з вершини a для графа G1

Оскільки всі ребра в графі (див. рис. 3.10) мають вагу 1, з табл. 3.1 видно, що відстані при обчисленні можуть збільшуватися тільки на 1. Тому фактично, як тільки нескінченна мітка змінилася на кінцеву числову мітку, вона згодом вже не модифікується, тобто не може зменшитися. Це означає, що відповідна найкоротша відстань між вершинами вже визначена. Тоді кількість складань і порівнянь в алгоритмі Дейкстри скорочується на 2 * 5 = 10 операцій (табл. 3.3).

Таблиця 3.3 – Модифікована (скорочена) таблиця 3.1 для обчислення найкоротших відстаней в графі G1 з вершини a

| | **1** | **2** | **3** |
|---|---|---|---|
| | **u = a**<br>**r = 0** | **u = b**<br>**r = 1** | **u = f**<br>**r = 1** |
| **b** | a, 1 | | |
| **c** | a, ∞ | b, 2 | |
| **d** | a, ∞ | b, 2 | |
| **e** | a, ∞ | a, ∞ | f, 2 |
| **f** | a, 1 | a, 1 | |

Слід зауважити, що найкоротші маршрути з вершин a, c, e і дерева найкоротших шляхів будуть ідентичними.

*Підзадача* 3.1.2. Для графа на рис. 3.10 відшукати найкоротші відстані і вказати відповідні їм шляхи від вершини b до всіх інших його вершин.

В ході виконання алгоритму Дейкстри складається таблиця, розмір якої 6x6 (число рядків і стовпців) визначається кількістю вершин в графі. У заголовках рядків таблиці вказуються вершини, до яких належить знайти найкоротшу відстань (табл. 3.4).

Таблиця 3.4 – Обчислення найкоротших відстаней з вершини b для графа G1

| | **1** | **2** | **3** | **4** | **5** |
|---|---|---|---|---|---|
| | **u = b**<br>**r = 0** | **u = a**<br>**r = 1** | **u = c**<br>**r = 1** | **u = d**<br>**r = 1** | **u = f**<br>**r = 1** |
| **a** | b, 1 | | | | |
| **c** | b, 1 | b, 1 | | | |
| **d** | b, 1 | b, 1 | b, 1 | | |
| **e** | b, ∞ | b, ∞ | b, ∞ | d, 2 | d (f), 2 |
| **f** | b, 1 | b, 1 | b, 1 | b, 1 | |

Заповнення табл. 3.4 аналогічно попередньому (див. коментарі до табл. 3.1).

У табл. 3.5 представлені обчислення з урахуванням модифікації алгоритму Дейкстри. Тут видно, що на початковому етапі після розміщення

міток в першому стовпці визначені найкоротші відстані від пункту відправлення – вершини b – до інших вершин графа (пунктів прибуття), за винятком вершини e. Всі вони збігаються з довжиною ребра 1, а самі найкоротші ланцюга ідентичні з ребер, які з'єднують початковий пункт (вершину b) з кінцевими пунктами прибуття – вершинами a, c, d, f. Залишається знайти найкоротша відстань і ланцюг від початкового пункту b до пункту прибуття e. Поетапне виконання алгоритму показує, що з усіх кінцевих числових міток в першому стовпці можна вибрати будь-яку, оскільки всі вони дорівнюють 1. Для даного прикладу можна бачити, що вибір вершин f (або d) дозволить завершити пошук ще за один прохід (див. табл. 3.5) з тим же результатом, що і в табл. 3.4.

Таблиця 3.5 – Модифікована (скорочена) таблиця 3.4 найкоротших відстаней з вершини b для графа G1

| | **1** | **2** |
|---|---|---|
| | **u = b**<br>**r = 0** | **u = f**<br>**r = 1** |
| **a** | b, 1 | b, 1 |
| **c** | b, 1 | b, 1 |
| **d** | b, 1 | b, 1 |
| **e** | b, ∞ | f, 2 |
| **f** | b, 1 | |

Дані для всіх найкоротших ланцюгів представлені в табл. 3.6.

Таблиця 3.6 – Найкоротші ланцюга з вершини b для графа G1

| Ланцюг | Довжина |
|---|---|
| $b \xrightarrow{1} a$ | $r(b,a)=1$ |
| $b \xrightarrow{1} c$ | $r(b,c)=1$ |
| $b \xrightarrow{1} d$ | $r(b,d)=1$ |
| $b \xrightarrow{1} d \xrightarrow{1} e$ | $r(b,e)=1+1=2$ |
| $b \xrightarrow{1} f$ | $r(b,f)=1$ |

Граф, який ілюструє дерево найкоротших ланцюгів з вершини b, наведено на рис. 3.12.

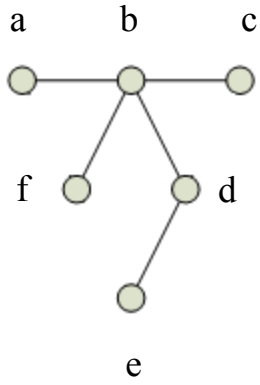



Рисунок 3.12 – Граф (дерево) найкоротших ланцюгів з вершини b для графа G1

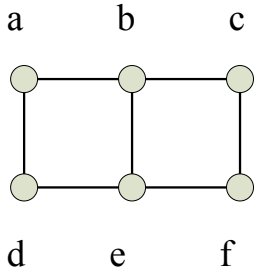
Слід зауважити, що найкоротші ланцюга з вершин b, d, f і дерева найкоротших шляхів будуть ідентичними, відповідно.

Матриця найкоротших відстаней між усіма парами вершин графа G1 представлена нижче:

$$\mathrm{Dist}_1 = \begin{array}{c|cccccc} & a & b & c & d & e & f \\ \hline a & . & 1 & 2 & 2 & 2 & 1 \\ b & 1 & . & 1 & 1 & 2 & 1 \\ c & 2 & 1 & . & 1 & 2 & 2 \\ d & 2 & 1 & 1 & . & 1 & 1 \\ e & 2 & 2 & 2 & 1 & . & 1 \\ f & 1 & 1 & 2 & 1 & 1 & . \end{array} \qquad (3.2)$$

## 3.3 Метод пошуку найкоротших відстаней між парами вершин

Задача 3.2. Для графа G2 з одиничними ваговими коефіцієнтами, представленого на діаграмі (рис. 3.13, а) з відповідною матрицею суміжності (рис. 3.13, б), знайти всі найкоротші відстані між усіма парами вершин.




а

$G_2$

| | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| a | . | 1 | . | 1 | . | . |
| b | 1 | . | 1 | . | 1 | . |
| c | . | 1 | . | . | . | 1 |
| d | 1 | . | . | . | 1 | . |
| e | . | 1 | . | 1 | . | 1 |
| f | . | . | 1 | . | 1 | . |

б

Рисунок 3.13 – Граф G2 (а) і його матриця суміжності (б)

*Підзадача* 3.2.1. Знайти відстані і вказати всі найкоротші шляхи від вершини a до решти вершин графа G2.

В процесі реалізації алгоритму Дейкстри заповнюється таблиця, кількість рядків і стовпців якої визначається потужністю безлічі вершин графа (6x6). У заголовках рядків таблиці вказуються вершини, до яких належить знайти найкоротша відстань (табл. 3.7).

Таблиця 3.7 – Обчислення найкоротших відстаней з вершини a для графа G2

| | **1** | **2** | **3** | **4** | **5** |
|---|---|---|---|---|---|
| | **u = a**<br>**r = 0** | **u = b**<br>**r = 1** | **u = d**<br>**r = 1** | **u = c**<br>**r = 2** | **u = e**<br>**r = 2** |
| **b** | a, 1 | | | | |
| **c** | a, ∞ | b, 2 | b, 2 | | |
| **d** | a, 1 | a, 1 | | | |
| **e** | a, ∞ | b, 2 | b, 2 | b, 2 | |
| **f** | a, ∞ | a, ∞ | a, ∞ | c, 3 | c, 3 |

У табл. 3.8 наведені обчислення з урахуванням модифікації алгоритму. Видно, що кількість операцій в даному випадку зменшується несуттєво: на одну ітерацію – порівняння.

Таблиця 3.8 – Обчислення найкоротших відстаней з вершини a для графа G2 з урахуванням модифікованого алгоритму

| | **1** | **2** | **3** | **4** |
|---|---|---|---|---|
| | **u = a r = 0** | **u = b r = 1** | **u = d r = 1** | **u = c r = 2** |
| **b** | a, 1 | | | |
| **c** | a, ∞ | b, 2 | b, 2 | |
| **d** | a, 1 | a, 1 | | |
| **e** | a, ∞ | b, 2 | b, 2 | b, 2 |
| **f** | a, ∞ | a, ∞ | a, ∞ | c, 3 |

Дані для всіх найкоротших ланцюгів представлені в табл. 3.9.

Таблиця 3.9 – Найкоротші ланцюга, відновлені по таблиці 3.8

| Ланцюг | Довжина |
|---|---|
| $a \xrightarrow{1} b$ | $r(a,b)=1$ |
| $a \xrightarrow{1} b \xrightarrow{1} c$ | $r(a,c)=1+1=2$ |
| $a \xrightarrow{1} d$ | $r(a,d)=1$ |
| $a \xrightarrow{1} b \xrightarrow{1} e$ | $r(a,e)=1+1=2$ |
| $a \xrightarrow{1} b \xrightarrow{1} c \xrightarrow{1} f$ | $r(a,f)=1+1+1=3$ |

Граф, який ілюструє дерево найкоротших ланцюгів з вершини a, представлений на рис. 3.14.

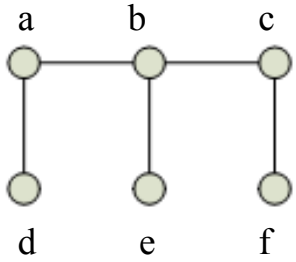



Рис. 3.14 – Дерево найкоротших ланцюгів з вершини a для графа G2

Слід зауважити, що найкоротші маршрути з вершин c, d, f і дерева найкоротших шляхів будуть ідентичними тим, що побудовані з вершини a.

*Підзадача* 3.2.2. Для графа G2, представленого на рис. 3.13, необхідно знайти найкоротші відстані і вказати всі найкоротші шляхи від вершини b до всіх інших вершин. У заголовках рядків таблиці вказуються вершини – пункти прибуття, до яких належить знайти найкоротші відстані і маршрути (табл. 3.10).

Таблиця 3.10 – Обчислення найкоротших відстаней з вершини b графа G2

| | **1** | **2** | **3** | **4** | **5** |
|---|---|---|---|---|---|
| | **u = b**<br>**r = 0** | **u = a**<br>**r = 1** | **u = c**<br>**r = 1** | **u = e**<br>**r = 1** | **u = d**<br>**r = 2** |
| **a** | b, 1 | | | | |
| **c** | b, 1 | b, 1 | | | |
| **d** | b, ∞ | a, 2 | a, 2 | a, 2 | |
| **e** | b, 1 | b, 1 | b, 1 | | |
| **f** | b, ∞ | b, ∞ | c, 2 | c, 2 | c, 2 |

Обчислення з урахуванням модифікації алгоритму наведені в табл. 3.11.

Таблиця 3.11 – Скорочена таблиця 3.10 з урахуванням модифікації алгоритму

| | **1** | **2** | **3** |
|---|---|---|---|
| | **u = b**<br>**r = 0** | **u = a**<br>**r = 1** | **u = c**<br>**r = 1** |
| **a** | b, 1 | | |
| **c** | b, 1 | b, 1 | |
| **d** | b, ∞ | a, 2 | a, 2 |
| **e** | b, 1 | b, 1 | b, 1 |
| **f** | b, ∞ | b, ∞ | c, 2 |

Дані для всіх найкоротших ланцюгів представлені в табл. 3.12.

Таблиця 3.12 – Найкоротші ланцюга з вершини b для графа G2

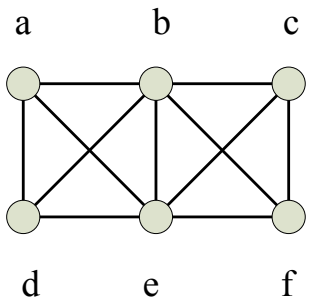
| Ланцюг | Довжина |
|---|---|
| $b \xrightarrow{1} a$ | $r(b,a)=1$ |
| $b \xrightarrow{1} c$ | $r(b,c)=1$ |
| $b \xrightarrow{1} a \xrightarrow{1} d$ | $r(b,d)=1+1=2$ |
| $b \xrightarrow{1} e$ | $r(b,e)=1$ |
| $b \xrightarrow{1} c \xrightarrow{1} f$ | $r(b,f)=1+1=2$ |

Дерево найкоротших ланцюгів з вершини b буде таким самим, як і на рис. 3.14. Матриця найкоротших відстаней між усіма парами вершин графа G2 має вигляд:

$$\mathrm{Dist}_2 = \begin{array}{c|cccccc} & a & b & c & d & e & f \\ \hline a & . & 1 & 2 & 1 & 2 & 3 \\ b & 1 & . & 1 & 2 & 1 & 2 \\ c & 2 & 1 & . & 3 & 2 & 1 \\ d & 1 & 2 & 3 & . & 1 & 2 \\ e & 2 & 1 & 2 & 1 & . & 1 \\ f & 3 & 2 & 1 & 2 & 1 & . \end{array} \qquad (3.3)$$

## 3.4 Пошук мінімальних шляхів для графа з одиничними дугами

Задача 3.3. Для графа з одиничними ваговими коефіцієнтами, представленого на діаграмі (рис. 3.15, а) з відповідною матрицею суміжності (рис. 3.15, б), необхідно знайти найкоротші відстані між усіма парами вершин.




G3

а

$G_3$ =

| | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| a | . | 1 | . | 1 | 1 | . |
| b | 1 | . | 1 | 1 | 1 | 1 |
| c | . | 1 | . | . | 1 | 1 |
| d | 1 | 1 | . | . | 1 | . |
| e | 1 | 1 | 1 | 1 | . | 1 |
| f | . | 1 | 1 | . | 1 | . |

б

Рисунок 3.15 – Граф G3 (а) і його матриця суміжності (б)

*Підзадача* 3.3.1. Знайти відстані і вказати всі найкоротші шляхи від вершини a до решти вершин графа G3. В процесі реалізації алгоритму Дейкстри заповнюється таблиця, кількість рядків і стовпців якої визначається потужністю множини вершин графа, тобто 6x6. У заголовках рядків таблиці вказуються вершини, до яких належить знайти найкоротшу відстань (таблиця 3.13).

Таблиця 3.13 – Обчислення найкоротших відстаней з вершини a для графа G3

| | **1** | **2** | **3** | **4** | **5** |
|---|---|---|---|---|---|
| | **u = a<br>r = 0** | **u = b<br>r = 1** | **u = d<br>r = 1** | **u = e<br>r = 1** | **u = f<br>r = 1** |
| **b** | a, 1 | | | | |
| **c** | a, ∞ | b, 2 | b, 2 | b, 2 | b, 2 |
| **d** | a, 1 | a, 1 | a, 1 | | |
| **e** | a, 1 | a, 1 | a, 1 | | |
| **f** | a, ∞ | b, 2 | b, 2 | b, 2 | |

Модифіковані обчислення наведені в табл. 3.14.

Таблиця 3.14 – Скорочена таблиця найкоротших відстаней
за модифікованим алгоритмом

| | **1** | **2** |
|---|---|---|
| | **u = a**<br>**r = 0** | **u = b**<br>**r = 1** |
| **b** | a, 1 | |
| **c** | a, ∞ | b, 2 |
| **d** | a, 1 | a, 1 |
| **e** | a, 1 | a, 1 |
| **f** | a, ∞ | b, 2 |

Дані для всіх найкоротших ланцюгів в графі з вершини a наведені в табл. 3.15.

Таблиця 3.15 – Найкоротші ланцюги з вершини a графа G3

| Ланцюг | Довжина |
|---|---|
| $a \xrightarrow{1} b$ | $r(a,b)=1$ |
| $a \xrightarrow{1} b \xrightarrow{1} c$ | $r(a,c)=1+1=2$ |
| $a \xrightarrow{1} d$ | $r(a,d)=1$ |
| $a \xrightarrow{1} e$ | $r(a,e)=1$ |
| $a \xrightarrow{1} b \xrightarrow{1} f$ | $r(a,f)=1+1=2$ |

Граф, який ілюструє дерево найкоротших ланцюгів з вершини a, що відповідають найкоротшим маршрутам із заданого пункту відправлення в усі можливі пункти прибуття, представлений на рис. 3.16.

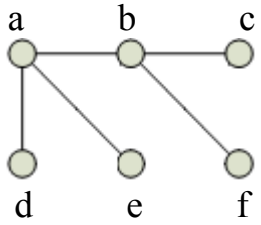



Рисунок 3.16 – Дерево найкоротших ланцюгів з вершини a

Слід зауважити, що найкоротші маршрути з вершин c, d, f і дерева найкоротших шляхів будуть ідентичними тим, що побудовані з вершини a.

*Підзадача* 3.3.2. Для графа на рис. 3.15, а знайти найкоротші відстані і всі відповідні шляхи від вершини b до всіх інших вершин. У заголовках рядків таблиці вказуються вершини (пункти прибуття), до яких належить знайти найкоротші відстані (табл. 3.16).

Таблиця 3.16 – Обчислення найкоротших відстаней з вершини b графа G3

| | **1** |
|---|---|
| | **u = b**<br>**r = 0** |
| **a** | b, 1 |
| **c** | b, 1 |
| **d** | b, 1 |
| **e** | b, 1 |
| **f** | b, 1 |

Stop

З табл. 3.16 видно, що алгоритм Дейкстри зупинений після першого проходу, оскільки всі найкоротші відстані визначені на початковому етапі розміщення міток (1-й стовпець). Дані для всіх найкоротших ланцюгів з вершини b представлені в табл. 3.17.

Таблиця 3.17 – Найкоротші ланцюга з вершини b графа G3

| Ланцюг | Довжина |
|---|---|
| $b \xrightarrow{1} a$ | $r(b,a)=1$ |
| $b \xrightarrow{1} c$ | $r(b,c)=1$ |
| $b \xrightarrow{1} d$ | $r(b,d)=1$ |
| $b \xrightarrow{1} e$ | $r(b,e)=1$ |
| $b \xrightarrow{1} f$ | $r(b,f)=1$ |

Дерево найкоротших ланцюгів з вершини b має топологію типу «зірка» і представлено на рис. 3.17.

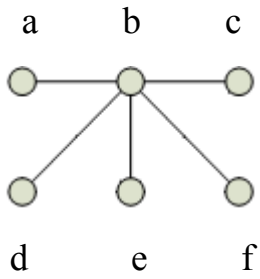



Рисунок 3.17 – Дерево найкоротших ланцюгів з вершини b

Матриця найкоротших відстаней між усіма парами вершин графа G3 має вигляд:

$$Dist_3 = \begin{array}{c|cccccc} & a & b & c & d & e & f \\ \hline a & . & 1 & 2 & 1 & 1 & 2 \\ b & 1 & . & 1 & 1 & 1 & 1 \\ c & 2 & 1 & . & 2 & 1 & 1 \\ d & 1 & 1 & 2 & . & 1 & 2 \\ e & 1 & 1 & 1 & 1 & . & 1 \\ f & 2 & 1 & 1 & 2 & 1 & . \end{array} \quad (3.4)$$

Дерево найкоротших ланцюгів для вершини b буде ілюструватися графом, аналогічним графу на рис. 3.17, все найкоротші відстані також будуть дорівнювати 1.

Опис модифікованого алгоритму Дейкстри.

Кожній вершині з множині V ставиться у відповідність мітка, яка визначає мінімальну відоме відстань від цієї вершини до початкової вершини a (пункту відправлення). Алгоритм виконується крок за кроком. На кожному кроці він «відвідує» одну вершину і намагається зменшити числові мітки. Реалізація алгоритму завершується, коли всі вершини відвідані.

Ініціалізація. Числова мітка самої вершини a покладається рівною 0, іншим вершинам присвоюється тимчасова мітка – нескінченність. Це

означає, що відстані від a до інших вершин поки невідомі. Всі вершини графа позначаються як невідвідані.

Крок алгоритму. Якщо всі вершини відвідані, алгоритм завершує роботу. В іншому випадку, з ще не відвіданих вершин вибирається вершина u з мінімальною міткою. При цьому розглядаються всі можливі маршрути, де u є передостаннім пунктом. Вершини, в які ведуть ребра з u, називаються сусідніми по відношенню до u. Для кожного сусіда вершини u, крім позначених як відвідані, розглядається нова довжина шляху, що дорівнює сумі значень поточної мітки вершини u і довжини ребра, що з'єднує u з цим сусідом.

У традиційному алгоритмі Дейкстри далі враховується такий крок: якщо отримане значення довжини менше значення мітки сусіда, то мітка сусіда замінюється отриманим значенням довжини.

Для графів з ребрами одиничної довжини (ваги) сума відстаней кожен раз може збільшуватися тільки на 1. Тому в згаданому пункті тільки нескінченні мітки сусіда можуть змінюватися на кінцеві числові мітки, які згодом не змінюються, тобто зменшуватися вже не можуть. Відповідні відстані є числами натурального ряду. З цієї причини порівняння доцільно проводити тільки з метою визначення кінцевих числових міток для тих вершин, які таких поки не мають, тобто їх тимчасові мітки рівні нескінченності. Якщо не існує ребра, яке з'єднує постійно позначену вершину з вершиною, що має нескінчену мітку, то як черговий пункт вибирається постійно позначена вершина з мінімальною міткою у поточному стовпці (як і раніше), що дозволяє реалізувати спробу знайти мінімальний маршрут через іншу вершину.

Після розгляду всіх сусідів вершина u позначається як відвідана і крок алгоритму повторюється.

LABEL – масив для зберігання поточних відміток.

PERM – масив для вказівки постійно помічених вершин (вершини стають постійно поміченими, коли вони виявляються рівними $u_i$ для будь-якого i).

Якщо PERM (v) = 1, то v – постійно позначена вершина і її мітка дорівнює d (s, v). Спочатку PERM(s)=1 і PERM(v)=0 при $v \neq s$. PRED – масив покажчиків на вершини, з яких здійснено перехід до вершин з постійною міткою. Якщо вершина v позначена постійною міткою, то послідовність v, PRED (v), PRED (PRED (v)), ..., s – вершини, які складають найкоротший орієнтований шлях з s до v.

1. Початок. Покласти LABEL (s) = 0, PERM (s) = 1, PRED (s) = s; $\forall v \neq s$ покласти LABEL (v) = $\infty$, PERM (v) = 0, PRED (v) = v.

2. Нехай i = 0, u = s (u – остання з вершин з незмінною міткою. Тепер це вершина s).

3. Обчислення LABEL і зміна елементів масиву PRED. Покласти i=i+1. Виконати для кожної вершини v з нескінченною міткою наступні дії (у традиційному алгоритмі Дейкстри цей пункт виконувався для всіх вершин v, крім вершин з незмінною міткою, а в модифікованому алгоритмі він застосовується тільки до вершин з тимчасовими мітками LABEL (v) = $\infty$, Оскільки інші мітки модифікуватися не будуть):

3.1. Покласти M = min {LABEL (v), LABEL (u) + w (u, v)}, де w(u, v)=1 – довжина ребра, що з'єднує вершини u і v, якщо таке існує, інакше (т.е. коли не існує ребра (u, v)) – як постійно помічена обирається вершина з мінімальною кінцевою числовою міткою, а якщо таких кілька, то вибирається одна з них);

3.2. Якщо M<LABEL (v), то покласти LABEL (v) = M, PRED (v) = u.

4. Виділення вершини $u_i$. Серед всіх вершин, що не позначені незмінною міткою, знайти вершину w з найменшою міткою (якщо таких вершин кілька, то вибір можна зробити довільно). Покласти PERM (w) = 1, $u_i$=W (є останньою вершиною з незмінною міткою).

5. Якщо i<n–1, то повернення до кроку 3, інакше – кінець (всі найкоротші шляхи знайдені).

Мітки вершин являють собою довжини найкоротших шляхів; v, PRED (v), PRED (PRED (v)), ..., s – є вершини найкоротшого орієнтованого st шляху.

Таким чином, наведена модифікація алгоритму Дейкстри для неорієнтованих зважених графів довжиною в один символ ребра, що дозволяє скоротити кількість складань і порівнянь за рахунок виключення з цього процесу вже знайдених на попередньому етапі кінцевих числових міток, які в подальшому не можуть зменшуватися, а залишаються константами, але можливості перетворення тільки нескінченних міток сусіда в кінцеві числові мітки.

## 3.5 Визначення мінімальних шляхів у трикутній топології

Задача 3.4. Для графа G4 з 21 вершини, пов’язаних трикутної топологією (рис. 3.18), визначити всі найкоротші шляхи і відповідні їм відстані з вершини v1 (на рис. 3.18 – коротко 01) до всіх інших вершин.

Рішення. Розглянутий на рис. 3.18 граф G4 є неорієнтованим. Застосування алгоритму Дейкстри дозволяє знайти рішення поставленого завдання.

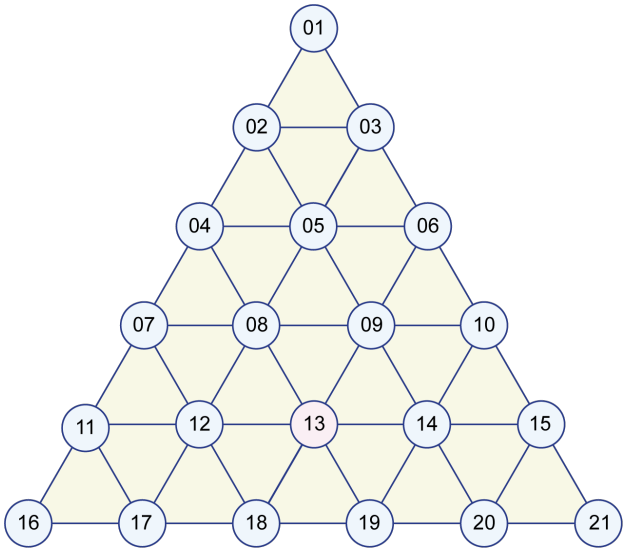



Рисунок 3.18 – Граф G4 – конфігурація мережі

Матриця суміжності графа G4 має вигляд:

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 9 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 12 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 |
| 15 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 16 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 17 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 18 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 19 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| 20 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 |
| 21 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |

(3.5)

В процесі реалізації алгоритму Дейкстри заповнюється таблиця, кількість рядків і стовпців якої визначається потужністю множини вершин графа, тобто 21x21. У заголовках рядків таблиці вказуються вершини, до яких належить знайти найкоротшу відстань (табл. 3.18).

Коментарі до заповнення табл. 3.18:

1) У заголовку першого стовпчика як пункт відправлення вказується початкова вершина u = v1, пройдена відстань приймається рівною нулю: r = 0. Кожне ребро ідентифікується стовпчиковою (початок) і рядковою (кінець) вершинами і читається «стовпець-рядок». При заповненні першого стовпчика враховуються такі правила: початкова вершина v1 не змінюється і повторюється в кожному осередку; відстань збігається з довжиною ребра (дорівнює 1), що з'єднує початкову (стовпчикову) вершину з вершиною з рядка. За відсутністю такого ребра відстані присвоюється тимчасова мітка $\infty$.

Таблиця 3.18 – Обчислення найкоротших відстаней з вершини v1 графа G4

| $G_4$ | u= $v_1$ r=0 | u= $v_2$ r=1 | u= $v_3$ r=1 | u=$v_4$ r=2 | u=$v_5$ r=2 | u=$v_6$ r=2 | u=$v_7$ r=3 | u=$v_8$ r=3 | u=$v_9$ r=3 | u=$v_{10}$ r=3 | u=$v_{11}$ r=4 | u=$v_{12}$ r=4 | u=$v_{13}$ r=4 | u=$v_{14}$ r=4 | u=$v_{15}$ r=5 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $v_2$ | $v_1$,1 | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| $v_3$ | $v_1$,1 | $v_1$,1 | – | – | – | – | – | – | – | – | – | – | – | – | – |
| $v_4$ | $v_1$,∞ | $v_2$,2 | $v_2$,2 | – | – | – | – | – | – | – | – | – | – | – | – |
| $v_5$ | $v_1$,∞ | $v_2$,2 | $v_2$,2 | $v_2$,2 | – | – | – | – | – | – | – | – | – | – | – |
| $v_6$ | $v_1$,∞ | $v_1$,∞ | $v_3$,2 | $v_3$,2 | $v_3$,2 | – | – | – | – | – | – | – | – | – | – |
| $v_7$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_4$,3 | $v_4$,3 | $v_4$,3 | – | – | – | – | – | – | – | – | – |
| $v_8$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_4$,3 | $v_4$,3 | $v_4$,3 | $v_4$,3 | – | – | – | – | – | – | – | – |
| $v_9$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_5$,3 | $v_5$,3 | $v_5$,3 | $v_5$,3 | – | – | – | – | – | – | – |
| $v_{10}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_6$,3 | $v_6$,3 | $v_6$,3 | $v_6$,3 | – | – | – | – | – | – |
| $v_{11}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_7$,4 | $v_7$,4 | $v_7$,4 | $v_7$,4 | – | – | – | – | – |
| $v_{12}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_7$,4 | $v_7$,4 | $v_7$,4 | $v_7$,4 | $v_7$,4 | – | – | – | – |
| $v_{13}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_8$,4 | $v_8$,4 | $v_8$,4 | $v_8$,4 | $v_8$,4 | – | – | – |
| $v_{14}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_9$,4 | $v_9$,4 | $v_9$,4 | $v_9$,4 | $v_9$,4 | – | – |
| $v_{15}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_{10}$,4 | $v_{10}$,4 | $v_{10}$,4 | $v_{10}$,4 | $v_{10}$,4 | – |
| $v_{16}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_{11}$,5 | $v_{11}$,5 | $v_{11}$,5 | $v_{11}$,5 | $v_{11}$,5 |
| $v_{17}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_{11}$,5 | $v_{11}$,5 | $v_{11}$,5 | $v_{11}$,5 | $v_{11}$,5 |
| $v_{18}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_{12}$,5 | $v_{12}$,5 | $v_{12}$,5 | $v_{12}$,5 |
| $v_{19}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_{13}$,5 | $v_{13}$,5 | $v_{13}$,5 |
| $v_{20}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_{14}$,5 | $v_{14}$,5 |
| $v_{21}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_{15}$,5 |

Після заповнення першого стовпчика з усіх кінцевих числових міток вибирається мінімальна – 1 (якщо таких декілька, то вибирається будь-яка з них). Їй відповідає постійно позначена вершина у рядку – v2. Це означає, що найкоротша відстань до вершини v2 вже знайдена і дорівнює 1. Постійно позначена вершина приймається як новий відправний пункт u= v2 і разом з мінімальним відстанню r = 1 заноситься до заголовку чергового стовпчика.

2) При заповненні другого стовпчика таблиці числові мітки модифікуються згідно з правилом: до довжини ребра (v2, j) додається пройдена відстань 1 і результат порівнюється з попередньою числовою міткою в рядку; якщо нова сумарна відстань менше попередньої, то вона приймається як поточна, при цьому вершина змінюється на вершину з

поточного стовпчика. Так, ребро (v2, v4) має довжину 1, тоді пройдена відстань r = 1 + 1 = 2 має кінцеве значення, отже, воно приймається як поточна замість ∞. При цьому вершина v1 змінюється на вершину v2 з поточного стовпчика.

Ребро (v2, v3) має довжину 1, тоді пройдена відстань r = 1 + 1 = 2 > 1, тому в рядку v3 зберігається попередня числова мітка, позначення вершини при цьому також не змінюється. *Слід зауважити, що алгоритм Дейкстри в цьому випадку зазнає такої модифікації: якщо кінцева числова мітка вже визначена на поточному етапі, далі вона вже не змінюється (оскільки може тільки збільшуватися), тобто мінімальна відстань (довжина ланцюга) вже виявляється визначеною цим значенням числової мітки. Таким чином, виконувати додавання і порівняння для вершин з кінцевими числовими позначками на наступних етапах не має сенсу, їх слід розглядати по черзі (в порядку зростання індексів) і вибирати на кожній наступній ітерації як постійно помічені, поки всі вони не будуть вичерпані*.

З усіх кінцевих числових міток другого стовпчика вибирається мінімальна – 1. Їй відповідає вершина v3. Тепер вершина u = v3 є постійно поміченою, вона вибирається як новий відправний пункт і запам'ятовується у заголовку третього стовпця одночасно з пройденою відстанню r = 1.

3) Третій стовпець заповнюється аналогічно попередньому. Тут модифікується мітка в рядку v6. Всі числові мітки виявляються однаковими, тому вибирається перша за порядком, а саме v4. Тепер u=v4 є постійно поміченою, розглядається як новий відправний пункт і записується до заголовку четвертого стовпчика разом з пройденою відстанню r = 2.

4) У четвертому стовпці не змінюються постійні мітки з рядків v5 і v6, рівні 2. При цьому модифікуються мітки в рядках v7 і v8. Тому вибирається вершина u=v5, яка тепер постійно позначена, приймається як новий відправний пункт і зберігається в заголовку у наступній колонці поряд з пройденою відстанню r = 2.

5) В п’ятому стовпці числова мітка 2 в рядку v6 не змінюється і є мінімальною. При цьому з’являється кінцева мітка 3 в рядку v9. Таким чином, як постійно помічена вершина тепер вибирається v6 і заноситься до заголовку шостого стовпчика разом з пройденою відстанню 2.

6) У шостому стовпці з’являється нова кінцева числова мітка 3 в рядку v10, інші мітки не змінюються. Як постійно помічена вершина тепер вибирається v7 і заноситься до заголовку сьомого стовпчика разом з пройденою відстанню 3.

7) У сьомому стовпці з’являються дві нові кінцеві числові мітки 4 в рядках v11 і v12, інші мітки не змінюються. Як постійно помічену слід вибрати вершину v8, яка разом з пройденою відстанню 3 заноситься до заголовку восьмого стовпця.

8) У восьмому стовпці з'являється одна нова кінцева числова мітка 4 в рядку v13, інші мітки не змінюються. Тепер як постійно помічену слід вибрати вершину v9, яка разом з пройденою відстанню 3 заноситься до заголовку дев’ятого стовпчика.

9) У дев'ятому стовпці додається ще одна нова кінцева числова мітка 4 в рядку v14, інші мітки не змінюються, тому як постійно помічена вибирається вершина v10, яка разом з пройденою відстанню 3 заноситься до заголовку десятого стовпчика.

10) У десятому стовпці додається ще одна нова кінцева числова мітка 4 в рядку v15, інші позначки не модифікуються, тому як постійно помічена вибирається вершина v11, яка разом з пройденою відстанню 4 заноситься до заголовку одинадцятого стовпця.

11) У одинадцятому стовпці з’являються відразу дві нові кінцеві числові мітки 5 в рядках v16 і v17, інші позначки не модифікуються. Як постійно помічена вибирається вершина v12, яка разом з пройденою відстанню 4 заноситься до заголовку дванадцятого стовпчика.

12) У дванадцятому стовпчику з’являється ще одна числова мітка 5 в рядку v18, інші мітки не змінюються. Як постійно помічена вибирається вершина v13, яка разом з пройденою відстанню 4 заноситься до заголовку тринадцятого стовпчика.

13) У тринадцятому стовпці з'являється ще одна числова мітка 5 в рядку v19, інші мітки не змінюються. Як постійно помічена вибирається вершина v14, яка разом з пройденою відстанню 4 заноситься до заголовку чотирнадцятого стовпчика.

14) У чотирнадцятому стовпці з’являється ще одна числова мітка 5 в рядку v20, інші мітки не змінюються. Як постійно помічена вибирається вершина v15, яка разом з пройденою відстанню 4 заноситься у заголовок п'ятнадцятого стовпчика.

15) У п’ятнадцятому стовпці з’являється кінцева числова мітка 5 в рядку v21, інші мітки не змінюються. Всі рядки мають кінцеві числові мітки. Отже, процес пошуку мінімальних відстаней завершено.

Таким чином, з табл. 3.18 можна отримати інформацію про всі найкоротші ланцюги і відповідні їм довжини.

Наприклад, потрібно знайти найкоротший ланцюг з вершини v1 до вершину v21. Послідовність вершин у ланцюзі виписується з кінця: остання заповнена осередок в рядку v21 містить інформацію про довжину найкоротшого ланцюга rmin = 5 і передостанній вершині в цьому ланцюзі – v15. Інформація про попередню вершину знаходиться в останній комірці рядка v15, в даному випадку – вершина v10, далі – v6, v3 і v1, яка є початком маршруту:

$$v1 \xrightarrow{1} v3 \xrightarrow{1} v6 \xrightarrow{1} v10 \xrightarrow{1} v15 \xrightarrow{1} v21,\ dist = 5.$$

Дані всіх найкоротших ланцюгів представлені в табл. 3.19. Граф, який ілюструє дерево найкоротших ланцюгів з вершини v1, представлений на рис. 3.19.

Таблиця 3.19 – Найкоротші ланцюги з вершини v1 графа G4

| Ланцюг | Довжина |
|---|---|
| $v1 \xrightarrow{1} v2,\ r_{min} = 1$ | $r(v1, v2) = 1$ |
| $v1 \xrightarrow{1} v3,\ r_{min} = 1$ | $r(v1, v3) = 1$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v4,\ r_{min} = 2$ | $r(v1, v4) = 2$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v5,\ r_{min} = 2$ | $r(v1, v5) = 2$ |
| $v1 \xrightarrow{1} v3 \xrightarrow{1} v6,\ r_{min} = 2$ | $r(v1, v6) = 2$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v4 \xrightarrow{1} v7,\ r_{min} = 3$ | $r(v1, v7) = 3$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v4 \xrightarrow{1} v8,\ r_{min} = 3$ | $r(v1, v8) = 3$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v5 \xrightarrow{1} v9,\ r_{min} = 3$ | $r(v1, v9) = 3$ |
| $v1 \xrightarrow{1} v3 \xrightarrow{1} v6 \xrightarrow{1} v10,\ r_{min} = 3$ | $r(v1, v10) = 3$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v4 \xrightarrow{1} v7 \xrightarrow{1} v11,\ r_{min} = 4$ | $r(v1, v11) = 4$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v4 \xrightarrow{1} v7 \xrightarrow{1} v12,\ r_{min} = 4$ | $r(v1, v12) = 4$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v4 \xrightarrow{1} v8 \xrightarrow{1} v13,\ r_{min} = 4$ | $r(v1, v13) = 4$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v5 \xrightarrow{1} v9 \xrightarrow{1} v14,\ r_{min} = 4$ | $r(v1, v14) = 4$ |
| $v1 \xrightarrow{1} v3 \xrightarrow{1} v6 \xrightarrow{1} v10 \xrightarrow{1} v15,\ r_{min} = 4$ | $r(v1, v15) = 4$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v4 \xrightarrow{1} v7 \xrightarrow{1} v11 \xrightarrow{1} v16,\ r_{min} = 5$ | $r(v1, v16) = 5$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v4 \xrightarrow{1} v7 \xrightarrow{1} v11 \xrightarrow{1} v17,\ r_{min} = 5$ | $r(v1, v17) = 5$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v4 \xrightarrow{1} v7 \xrightarrow{1} v12 \xrightarrow{1} v18,\ r_{min} = 5$ | $r(v1, v18) = 5$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v4 \xrightarrow{1} v8 \xrightarrow{1} v13 \xrightarrow{1} v19,\ r_{min} = 5$ | $r(v1, v19) = 5$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v5 \xrightarrow{1} v9 \xrightarrow{1} v14 \xrightarrow{1} v20,\ r_{min} = 5$ | $r(v1, v20) = 5$ |
| $v1 \xrightarrow{1} v3 \xrightarrow{1} v6 \xrightarrow{1} v10 \xrightarrow{1} v15 \xrightarrow{1} v21,\ r_{min} = 5$ | $r(v1, v21) = 5$ |

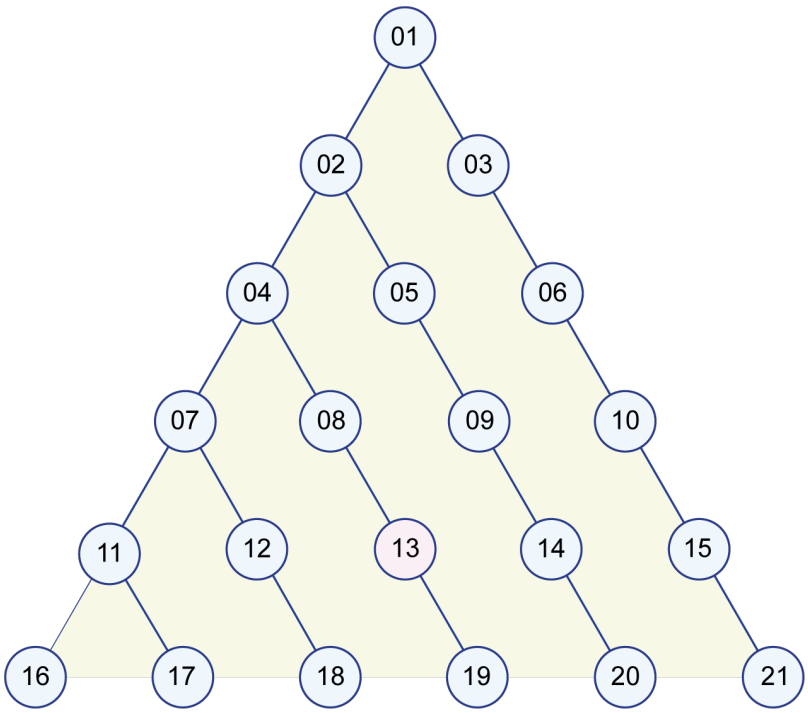



Рисунок 3.19 – Дерево найкоротших ланцюгів з вершини v1

Оскільки всі ребра в графі (див. рис. 3.18) мають вагу 1, з табл. 3.18 видно, що відстані при обчисленні можуть збільшуватися тільки на 1. Тому фактично, як тільки нескінченна мітка змінилася на кінцеву числову мітку, вона згодом вже не модифікується, тобто не може зменшитися. Це означає, що відповідна найкоротша відстань між вершинами вже визначена. Тоді кількість складань і порівнянь в алгоритмі Дейкстри можна скоротити за рахунок вилучення порівнянь вже з отриманими кінцевими числовими позначками та послідовного вибору постійно помічених вершин зі списку тих, що мають мінімальні числові мітки, до їх вичерпання з подальшим переходом до міток на одиницю більших. Час реалізації алгоритму скорочується на 30%.

## 3.6 Пошук мінімальних шляхів у прямокутній топології

Задача 3.5. Визначити всі найкоротші шляхи і відстані з вершини v1 (на рис. 3.20 – коротко 01) до всіх інших вершин прямокутної топології графа G5, що містить 21 вершину.

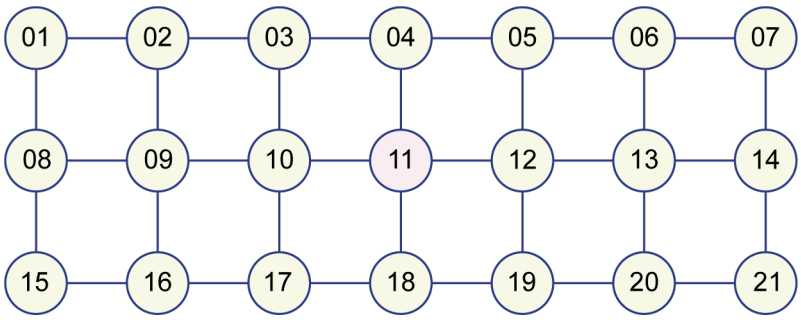



Рисунок 3.20 – Граф G5 – конфігурація мережі

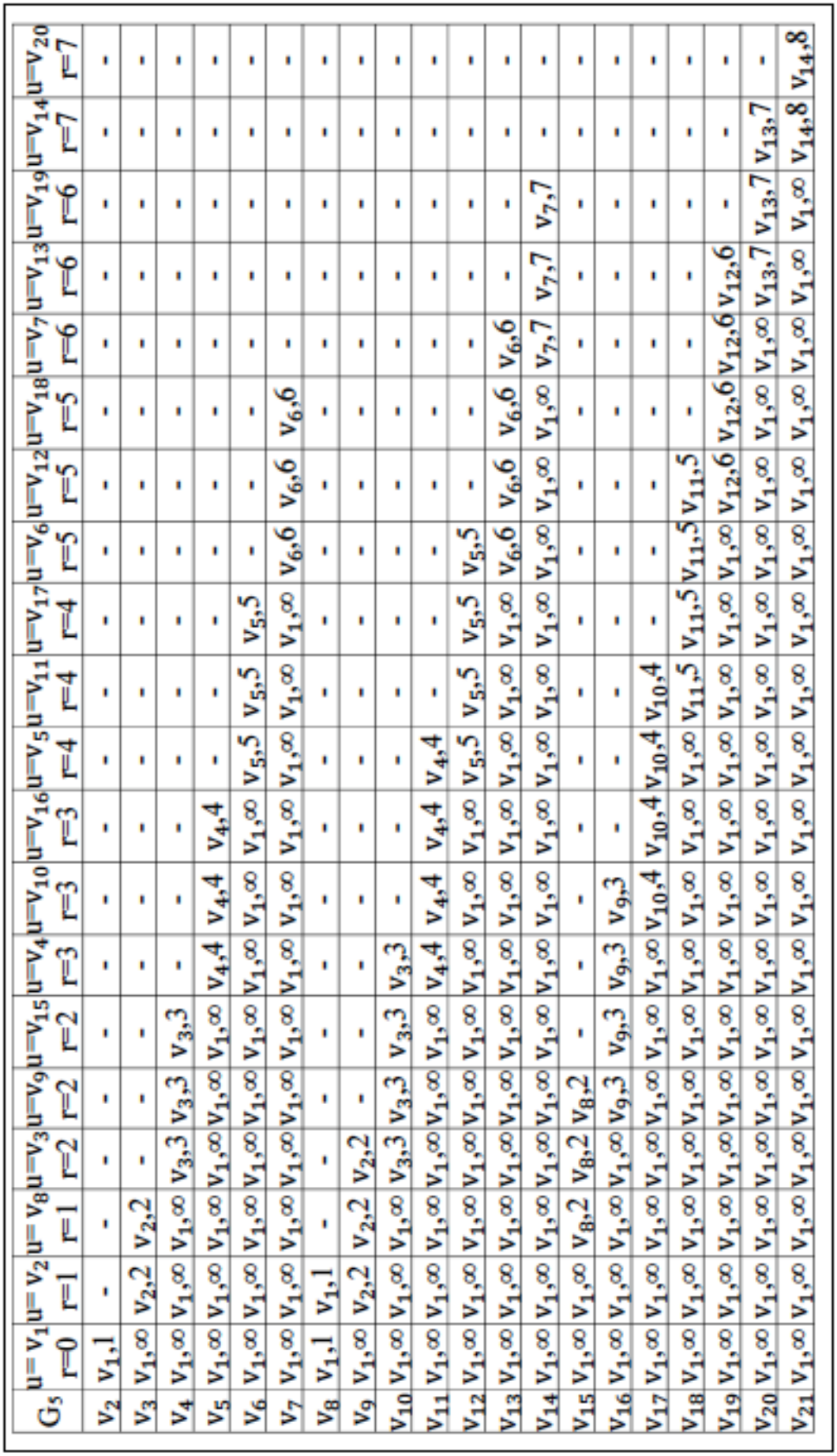
Розв’язок. Розглянутий на рис. 3.20 граф G5 є неорієнтованим. Застосування алгоритму Дейкстри дозволяє знайти рішення поставленого завдання. Матриця суміжності графа G5 має вигляд:

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 9 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 10 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 11 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 12 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 13 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 14 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 15 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 16 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| 17 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 18 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 19 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| 20 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| 21 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |

(3.6)

В процесі реалізації алгоритму Дейкстри заповнюється таблиця, кількість рядків і стовпців якої визначається потужністю множини вершин графа, тобто 21x21. У заголовках рядків таблиці вказуються вершини, до яких належить знайти найкоротші відстані (табл. 3.20).

Таблиця 3.20 – Обчислення найкоротших відстаней з вершини v1 графа G5



| $G_5$ | u= $v_1$ r=0 | u= $v_2$ r=1 | u= $v_8$ r=1 | u=$v_3$ r=2 | u=$v_9$ r=2 | u=$v_{15}$ r=2 | u=$v_4$ r=3 | u=$v_{10}$ r=3 | u=$v_{16}$ r=3 | u=$v_5$ r=4 | u=$v_{11}$ r=4 | u=$v_{17}$ r=4 | u=$v_6$ r=5 | u=$v_{12}$ r=5 | u=$v_{18}$ r=5 | u=$v_7$ r=6 | u=$v_{13}$ r=6 | u=$v_{19}$ r=6 | u=$v_{14}$ r=7 | u=$v_{20}$ r=7 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $v_2$ | $v_1$,1 | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - |
| $v_3$ | $v_1$,∞ | $v_2$,2 | $v_2$,2 | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - |
| $v_4$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_3$,3 | $v_3$,3 | $v_3$,3 | - | - | - | - | - | - | - | - | - | - | - | - | - | - |
| $v_5$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_4$,4 | $v_4$,4 | $v_4$,4 | - | - | - | - | - | - | - | - | - | - | - |
| $v_6$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_5$,5 | $v_5$,5 | $v_5$,5 | - | - | - | - | - | - | - | - |
| $v_7$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_6$,6 | $v_6$,6 | $v_6$,6 | - | - | - | - | - |
| $v_8$ | $v_1$,1 | $v_1$,1 | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - |
| $v_9$ | $v_1$,∞ | $v_2$,2 | $v_2$,2 | $v_2$,2 | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - |
| $v_{10}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_3$,3 | $v_3$,3 | $v_3$,3 | $v_3$,3 | - | - | - | - | - | - | - | - | - | - | - | - | - |
| $v_{11}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_4$,4 | $v_4$,4 | $v_4$,4 | $v_4$,4 | - | - | - | - | - | - | - | - | - | - |
| $v_{12}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_5$,5 | $v_5$,5 | $v_5$,5 | $v_5$,5 | - | - | - | - | - | - | - |
| $v_{13}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_6$,6 | $v_6$,6 | $v_6$,6 | $v_6$,6 | - | - | - | - |
| $v_{14}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_7$,7 | $v_7$,7 | $v_7$,7 | - | - |
| $v_{15}$ | $v_1$,∞ | $v_1$,∞ | $v_8$,2 | $v_8$,2 | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - |
| $v_{16}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_9$,3 | $v_9$,3 | $v_9$,3 | $v_9$,3 | - | - | - | - | - | - | - | - | - | - | - | - |
| $v_{17}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_{10}$,4 | $v_{10}$,4 | $v_{10}$,4 | $v_{10}$,4 | - | - | - | - | - | - | - | - | - |
| $v_{18}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_{11}$,5 | $v_{11}$,5 | $v_{11}$,5 | $v_{11}$,5 | - | - | - | - | - | - |
| $v_{19}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_{12}$,6 | $v_{12}$,6 | $v_{12}$,6 | $v_{12}$,6 | - | - | - |
| $v_{20}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_{13}$,7 | $v_{13}$,7 | $v_{13}$,7 | - |
| $v_{21}$ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_1$,∞ | $v_{14}$,8 | $v_{14}$,8 |

Коментарі до заповнення табл. 3.20:

1) У заголовку першого стовпчика як пункт відправлення вказується початкова вершина u = v1, пройдена відстань приймається рівною нулю: r = 0. Кожне ребро ідентифікується стовпчиковою (початок) і рядковою (кінець) вершинами і читається «стовпець-рядок». При заповненні першого стовпчика враховуються такі правила: початкова вершина v1 не змінюється і повторюється в кожному осередку; відстань збігається з довжиною ребра (дорівнює 1), що з'єднує початкову (стовпчикову) вершину з вершиною з рядка. За відсутністю такого ребра відстані присвоюється тимчасова мітка $\infty$.

Після заповнення першого стовпчика з усіх кінцевих числових міток вибирається мінімальна – 1 (якщо таких декілька, то вибирається будь-яка з них). Їй відповідає постійно позначена вершина в рядку – v2. Це означає, що найкоротшу відстань до вершини v2 вже знайдено і вона дорівнює 1. Постійно позначена вершина приймається як новий відправний пункт u = v2 і разом з мінімальною пройденою відстанню r = 1 заноситься до заголовку чергового стовпчика.

2) При заповненні другого стовпчика таблиці числові мітки модифікуються згідно з правилом: до довжини ребра (v2, j) додається пройдена відстань 1 і результат порівнюється з попередньою числовою міткою в рядку; якщо нова сумарна відстань менше попередньої, то вона приймається як поточна, при цьому вершина змінюється на вершину з поточного стовпчика. Так, ребро (v2, v3) має довжину 1, тоді пройдена відстань r = 1 + 1 = 2 має кінцеве значення, отже, воно приймається як поточне замість тимчасової мітки $\infty$. При цьому вершина v1 змінюється на вершину v2 з поточного стовпчика.

Ребро (v2, v8) має довжину 1, тоді пройдена відстань r = 1 + 1 = 2 > 1, тому в рядку v8 зберігається попередня числова мітка, позначення вершини при цьому також не змінюється. Слід зауважити, що алгоритм Дейкстри в цьому випадку зазнає таку ж модифікацію, що і в задачі 3.4, а саме: якщо кінцева числова мітка вже визначена на поточному етапі, далі вона вже не

змінюється (оскільки може тільки збільшуватися), тобто мінімальна відстань (довжина ланцюга) вже виявляється визначено цим значенням числової мітки. Таким чином, виконувати додавання і порівняння для вершин з кінцевими числовими позначками на наступних етапах не має сенсу, їх слід розглядати по черзі (в порядку зростання індексів) і вибирати на кожній наступній ітерації як постійно помічені, поки всі вони не будуть вичерпані.

З усіх кінцевих числових міток другого стовпчика (для вершин v3, v8) вибирається мінімальна – 1. Їй відповідає вершина v8. Тепер вершина u = v8 є постійно поміченою, вона вибирається як новий відправний пункт і запам’ятовується у заголовку третього стовпця одночасно з пройденою відстанню r = 1.

3) Третій стовпець заповнюється аналогічно попередньому. Тут модифікується мітка в рядку v15. Всі отримані тут кінцеві числові мітки виявляються однаковими (дорівнюють 2), тому вибирається перша за порядком, а саме v3. Тепер u = v3 є постійно поміченою, розглядається як новий відправний пункт і записується до заголовку четвертого стовпчика разом з пройденою відстанню r = 2.

4) У четвертому стовпці не змінюються постійні мітки з рядків v9 і v15, рівні 2. При цьому модифікуються мітки в рядках v4 і v10. Далі вибирається перша за порядком вершина u = v9 з мінімальною міткою у стовпці, яка тепер постійно позначена, приймається як новий відправний пункт і зберігається в заголовку у наступній колонці поряд з пройденою відстанню r = 2.

5) В п’ятому стовпці числова мітка 2 в рядку v15 не змінюється і є мінімальною. При цьому з’являється кінцева мітка 3 в рядку v16. Таким чином, як постійно помічена вершина тепер вибирається v15 і заноситься до заголовку шостого стовпчика разом з пройденою відстанню r = 2.

6) У шостому стовпці нові кінцеві числові позначки не з’являються, при цьому як постійно помічена вершина тепер вибирається v4 і заноситься до заголовку сьомого стовпчика разом з пройденою відстанню r = 3.

7) У сьомому стовпці з’являються дві нові кінцеві числові мітки 4 в рядках v5 і v11, інші мітки не змінюються. Як постійно помічену слід вибрати вершину v10, яка разом з пройденою відстанню r = 3 заноситься до заголовку восьмого стовпця.

8) У восьмому стовпці з’являється одна нова кінцева числова мітка 4 в рядку v17, інші мітки не змінюються. Тепер як постійно помічену слід вибрати вершину v16, яка разом з пройденою відстанню r = 3 заноситься до заголовку дев’ятого стовпчика.

9) У дев'ятому стовпці нова кінцева числова мітка не з’являється, а інші мітки не змінюються, тому як постійно помічена вибирається вершина v5, яка разом з пройденою відстанню r = 4 заноситься до заголовку десятого стовпчика.

10) У десятому стовпці додаються ще дві нові кінцеві числові мітки 5 в рядках v6 і v12, інші позначки не модифікуються, як постійно помічена вибирається вершина v11, яка разом з мінімальною числовою міткою – пройденою відстанню r = 4 заноситься до заголовку одинадцятого стовпця.

11) У одинадцятому стовпці з’являється одна нова кінцева числова мітка 5 в рядку v18, інші позначки не модифікуються. Як постійно помічена вибирається вершина v17, яка разом з мінімальною пройденою відстанню r=4 заноситься до заголовка дванадцятого стовпчика.

12) У дванадцятому стовпці нові кінцеві числові мітки не з'являються. Як постійно помічена вибирається вершина v6, яка разом з пройденою відстанню r = 5 заноситься до заголовку тринадцятого стовпчика.

13) У тринадцятому стовпці з’являються дві числові мітки 6 в рядках v7 і v13, інші мітки не змінюються. Як постійно помічена вибирається вершина v12, яка разом з пройденою відстанню r=5 заноситься до заголовку чотирнадцятого стовпчика.

14) У чотирнадцятому стовпці з’являється ще одна числова мітка 6 в рядку v19, інші мітки не змінюються. Як постійно помічена вибирається

вершина v18, яка разом з пройденою відстанню r = 5 заноситься до заголовку п’ятнадцятого стовпчика.

15) У п’ятнадцятому стовпці мітки не змінюються і нові не з’являються, тому як постійно помічена вибирається перша за списком вершина v7 з кінцевою числовою міткою r = 6, яка заноситься до заголовку шістнадцятого стовпчика.

16) У шістнадцятому стовпці з’являється нова кінцева числова мітка 7, якій відповідає вершина v14. При цьому як постійно помічена вибирається вершина v13 і разом з пройденою відстанню r = 6 заноситься до заголовку сімнадцятого стовпчика.

17) У сімнадцятому стовпці модифікується мітка з рядка v20, відповідна відстань дорівнює 7. Як постійно помічена вибирається вершина v19 і разом з мінімальною відстанню r = 6 запам’ятовується у заголовку вісімнадцятого стовпчика.

18) У вісімнадцятому стовпці нові кінцеві числові позначки не з’являються, тому як постійно помічена вибирається вершина v14 і разом з пройденою відстанню r=7 заноситься до заголовку дев'ятнадцятого стовпчика.

19) У дев'ятнадцятому стовпці модифікується мітка, яка відповідає рядку v21. Тепер рядки мають кінцеві числові мітки, які більш змінюватися не можуть. Отже, процес пошуку мінімальних відстаней завершено.

Таким чином, з табл. 3.20 можна отримати інформацію про всі найкоротші ланцюги і відповідні їм довжини.

Наприклад, потрібно знайти найкоротший ланцюг з вершини v1 до вершини v21. Послідовність вершин у ланцюзі виписується з кінця: останній заповнений осередок в рядку v21 містить інформацію про довжину найкоротшої ланцюга rmin = 8 і передостанній вершині в цьому ланцюзі – v14. Інформація про попередню вершину знаходиться в останній комірці рядка v14, в даному випадку – вершина v7, далі – v6, v5, v4, v3, v2 і v1, яка є початком маршруту:

$$v1 \xrightarrow{1} v2 \xrightarrow{1} v3 \xrightarrow{1} v4 \xrightarrow{1} v5 \xrightarrow{1} v6 \xrightarrow{1} v7 \xrightarrow{1} v14 \xrightarrow{1} v21,\ r_{min} = 8.$$

Дані всіх найкоротших ланцюгів представлені в табл. 3.21.

Таблиця 3.21 – Найкоротші ланцюга з вершини v1 графа G5

| ланцюг | довжина |
|---|---|
| $v1 \xrightarrow{1} v2,\ r_{min} = 1$ | $r(v1, v2) = 1$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v3,\ r_{min} = 2$ | $r(v1, v3) = 2$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v3 \xrightarrow{1} v4,\ r_{min} = 3$ | $r(v1, v4) = 3$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v3 \xrightarrow{1} v4 \xrightarrow{1} v5,\ r_{min} = 4$ | $r(v1, v5) = 4$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v3 \xrightarrow{1} v4 \xrightarrow{1} v5 \xrightarrow{1} v6,\ r_{min} = 5$ | $r(v1, v6) = 5$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v3 \xrightarrow{1} v4 \xrightarrow{1} v5 \xrightarrow{1} v6 \xrightarrow{1} v7,\ r_{min} = 6$ | $r(v1, v7) = 6$ |
| $v1 \xrightarrow{1} v8,\ r_{min} = 1$ | $r(v1, v8) = 1$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v9,\ r_{min} = 2$ | $r(v1, v9) = 2$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v3 \xrightarrow{1} v10,\ r_{min} = 3$ | $r(v1, v10) = 3$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v3 \xrightarrow{1} v4 \xrightarrow{1} v11,\ r_{min} = 4$ | $r(v1, v11) = 4$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v3 \xrightarrow{1} v4 \xrightarrow{1} v5 \xrightarrow{1} v12,\ r_{min} = 5$ | $r(v1, v12) = 5$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v3 \xrightarrow{1} v4 \xrightarrow{1} v5 \xrightarrow{1} v6 \xrightarrow{1} v13,\ r_{min} = 6$ | $r(v1, v13) = 6$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v3 \xrightarrow{1} v4 \xrightarrow{1} v5 \xrightarrow{1} v6 \xrightarrow{1} v7 \xrightarrow{1} v14,\ r_{min} = 7$ | $r(v1, v14) = 7$ |
| $v1 \xrightarrow{1} v8 \xrightarrow{1} v15,\ r_{min} = 2$ | $r(v1, v15) = 2$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v9 \xrightarrow{1} v16,\ r_{min} = 3$ | $r(v1, v16) = 3$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v3 \xrightarrow{1} v10 \xrightarrow{1} v17,\ r_{min} = 4$ | $r(v1, v17) = 4$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v3 \xrightarrow{1} v4 \xrightarrow{1} v11 \xrightarrow{1} v18,\ r_{min} = 5$ | $r(v1, v18) = 5$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v3 \xrightarrow{1} v4 \xrightarrow{1} v5 \xrightarrow{1} v12 \xrightarrow{1} v19,\ r_{min} = 6$ | $r(v1, v19) = 6$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v3 \xrightarrow{1} v4 \xrightarrow{1} v5 \xrightarrow{1} v6 \xrightarrow{1} v13 \xrightarrow{1} v20,\ r_{min} = 7$ | $r(v1, v20) = 7$ |
| $v1 \xrightarrow{1} v2 \xrightarrow{1} v3 \xrightarrow{1} v4 \xrightarrow{1} v5 \xrightarrow{1} v6 \xrightarrow{1} v7 \xrightarrow{1} v14 \xrightarrow{1} v21,\ r_{min} = 8$ | $r(v1, v21) = 8$ |

Граф, який ілюструє дерево найкоротших ланцюгів з вершини v1, зображено на рис. 3.21.

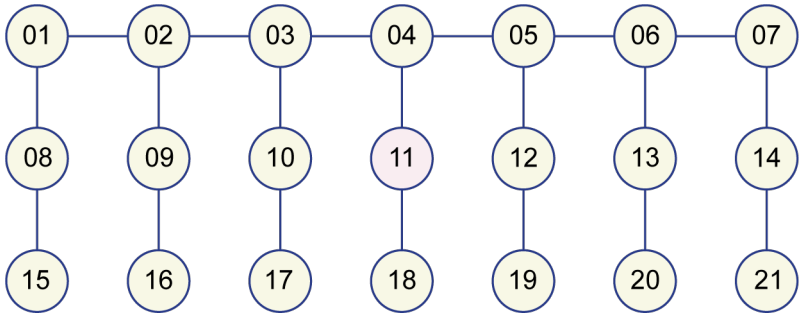



Рис. 3.21 – Дерево найкоротших ланцюгів з вершини v1 графа G5

## 3.7 Висновки до розділу 3

Розглянуто метод побудови оптимального маршруту на дорожній інфраструктурі між двома координатами, який має дорожні колізії, з метою його подальшої реалізації як online хмарного сервісу для водіїв транспортних засобів.

Виконано покроковий синтез алгоритму побудови оптимального маршруту руху транспортного засобу шляхом використання модифікованого алгоритму Дейкстри на топологічній інфраструктурі дорожнього руху, що приводиться до графу, з урахуванням існування дорожніх колізій.

Вирішені задачі:

1) Визначення всіх найкоротших шляхів і відповідних відстаней з вершини a до всіх інших вершин інфраструктури.

2) Пошуку всіх найкоротших відстаней між усіма парами вершин для графа з одиничними ваговими коефіцієнтами, заданого матрицею суміжності.

3) Оцінювання різних типів графів за розробленими критеріями якості, заданих матрицями суміжності, шляхом обчислення найкоротших відстаней між усіма парами вершин і складності графа.

4) Порівняння різних графових архітектур з метою визначення ефективних рішень при проектуванні нових дорожніх інфраструктур для скорочення часу руху за замовленим маршрутом.

# 4 КІБЕРФІЗИЧНА СТРУКТУРА ХМАРНОГО ТРАНСПОРТНОГО КОМП'ЮТИНГУ

Пропонується структура кіберфізичної системи, орієнтована на якісні інновації в моніторингу і управлінні транспортом, що усувають забруднення планети, численні аварії та колізії на дорогах за рахунок поступового переміщення дорожніх знаків в кіберпростір, делегування координат кожного автомобіля і хмарного online надання інформації про маршрут руху. Хмарний сервіс призначений для моніторингу та управління дорожнім рухом в реальному масштабі часу на основі використання глобальних систем позиціонування, навігації (GPS, GPRS), інтелектуальних дорожніх контролерів, засобів радіочастотної ідентифікації автомобілів з метою підвищення якості та безпеки пересування транспортних засобів, а також мінімізації часових і матеріальних витрат при русі автомобілів за заданими маршрутами. Розроблені моделі та методи інтегруються в кіберфізичну систему online взаємодії хмари моніторингу та управління з транспортними засобами та інфраструктурою дорожнього руху. Кожен автомобіль має особистий кабінет у кіберпросторі, інваріантний по відношенню до водіїв, які обслуговують транспортний засіб.

Задачі дослідження:

1) Визначення інноваційної привабливості для розробки і впровадження кіберфізичноъ системи хмарного комп'ютингового управління траспортом на основі цифрового моніторингу.

2) Створення хмарної кіберфізичної інфраструктури, що відображає і моделює процеси пересування транспортних засобів в часі і просторі, для сервісного обслуговування учасників дорожнього руху в реальному масштабі часу.

3) Створення архітектурних рішень для компонентів кіберфізичного комп'ютингу, пов'язаного з цифровим моніторингом і хмарним керуванням транспортом.

4) Верифікація моделей і методів, інтегрованих в структуру прототипа кіберфізичної системи хмарного управління транспортом.

## 4.1 Інноваційна привабливість "Cloud Traffic Control"

Запропонована кіберфізична система – дорожня інфраструктура і фізичний транспорт з цифровими сенсорами, а також хмарні сервіси моніторингу та управління дорожнім рухом відрізняється від існуючих структурною інтеграцією online взаємодіючих інтерактивних компонентів: 1) Суперпозиція сервісів електронної картографії, радіолокації, радіонавігації і online вибору маршруту руху. 2) Хмарний сервіс сенсорного моніторингу автомобіля і управління дорожнім рухом на основі поступового перетворення реальних знаків дорожнього руху у віртуальні. 3) Створення повного рефлекторного відображення фізичних транспортних процесів і явищ в хмарних сервісах, що включають кабінети автомобілів з їх дорожньою просторово-часовою історією (рис. 4.1).

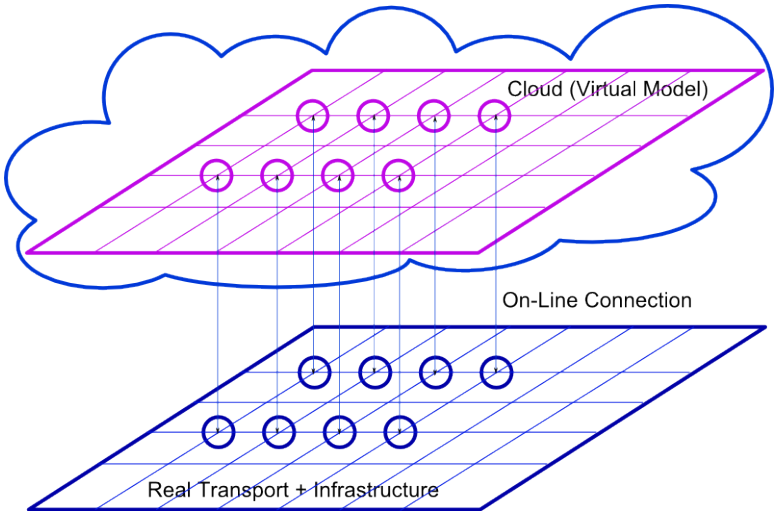



Рисунок 4.1 – Відображення інфраструктури і транспорту в хмарі

Структурна модель взаємодії хмар моніторингу та управління з транспортними засобами представлена на рис. 4.2, де автомобілі, перебуваючи в режимі on-line, делегують свої ідентифікатори (персональні дані), параметри руху і поточні координати хмарі, а натомість отримують в реальному масштабі часу сервіси оптимального (за часом, витратами і

якістю) маршруту і режиму руху для досягнення кінцевого пункту. Інтегральний аналіз дорожньої обстановки на основі оперативного збору даних від автомобілів і інфраструктурних моніторів дає можливість в режимі on-line оптимально управляти дорожніми контролерами перемикання світлофорів.

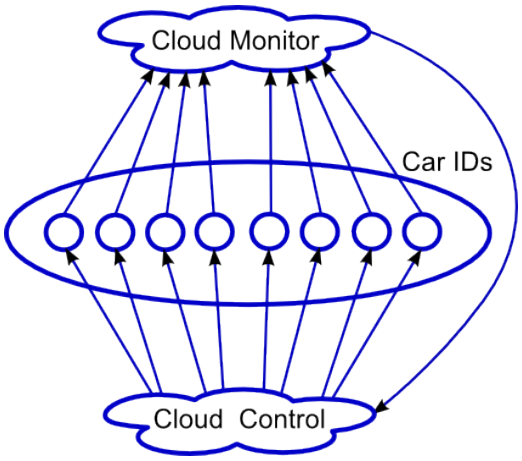



Рисунок 4.2 – Структура взаємодії транспорту та хмари

Взаємодія реального світу (автомобіль і інфраструктура) з хмарою формує два типи відношень, що задаються автоматними моделями, рис. 4.3: 1) транспортної інфраструктури з хмарою моніторингу та управління; 2) автомобіля з хмарою для оптимізації та ефективності пересування.

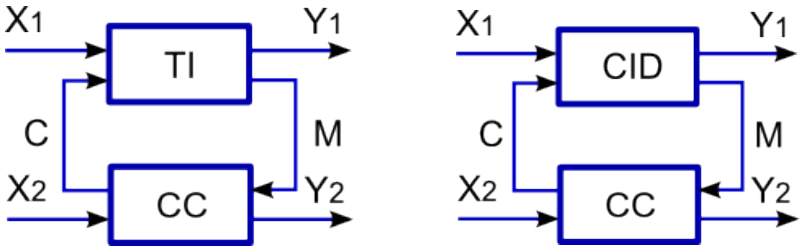



Рисунок 4.3 – Взаємодія інфраструктури і транспорту з хмарою моніторингу та управління

Тут представлені сигнали $X_1, Y_1, X_2, Y_2, C, M$, які визначаються як: 1) вхідні умови або операнди, необхідні для виконання замовлених сервісів; 2)

вихідні сповіщальні сигнали, що підтверджують виконання сервісних операцій; 3) вхідні управляючі дії, що формують запити на виконання сервісів; 4) вихідні змінні, що формують і ідентифікують стан системи управління; 5) сигнали розумного управління автомобілем або інфраструктурою дорожнього руху; 6) сповіщальні сигнали про виконання операційних сервісів. Автоматні моделі системи управління дорожнім рухом та автомобілем у формі взаємодії змінних, що представлені функціями переходів і виходів автомата першого роду:

$$CC = \{X, Y, C, M, f, g\},$$
$$\begin{cases} Y(t) = f[(X(t), M(t), Y(t-1)]; \\ C(t) = g[(X(t), M(t), Y(t-1)]. \end{cases}$$

Кожен з двох автоматів взаємодії інфраструктури і транспорту з хмарою має дві вхідні змінні (замовлення сервісів та стану об'єктів управління), а також два вихідних сигнали для моніторингу стану автомата (хмари) і управління хмарними сервісами.

В даному випадку автоматне моделювання процесів управління транспортом реалізує тезу «потоки автомобілів керують світлофорами, але не навпаки». Ідея була описана в інтернет-публікації [http://www.wired.com/wiredscience/2010/09/traffic-lights-adapt/] і реалізована в центральній частині Дрездена. Це забезпечило на практиці істотне зниження часу простою на перехрестях (56% для автобусів і трамваїв, 9% для автомобілів, вантажівок і 36% для пішоходів). В даному випадку інтелектуальні світлофори мають функції моніторингу вхідних і вихідних потоків автомобілів на перехрестях, що дає можливість реалізувати квазіоптимальне управління вуличними контролерами. Інший приклад. Населення США витрачає 500 000 років в пересуваннях по дорогах за один рік, що становить 100 мільярдів доларів сумарних витрат. Якщо зменшити час, проведений в автомобілях за рік на 10%, то економія складе кругленьку суму в 10 мільярдів доларів.

Більш детальна картина взаємодії реальних і віртуальних компонентів з хмарою моніторингу та управління транспортом представлена на рис. 4.4 (буферні комп’ютери дорожньої інфраструктури, інтелектуальні вуличні світлофори, Інтернет, розумна пил, Car-ID, супутники навігації, автомобіль, електронна карта, засоби захисту хмарних сервісів, спеціальні послуги державним службам, інтерфейси або гаджети для зв'язку з інтернетом).

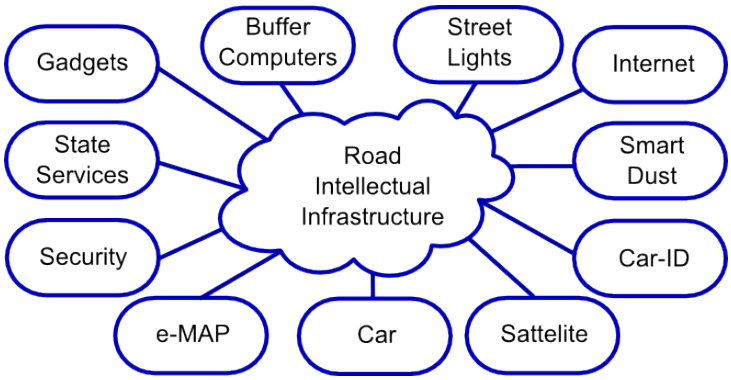



Рисунок 4.4 – Інтелектуальна інфраструктура дорожнього руху

Переваги від впровадження і ринкова привабливість хмарних сервісів дорожнього руху, аж ніяк не очевидні для громадян:

1) Для планети – збереження екології за рахунок зменшення забруднення навколишнього середовища, підвищення тривалості і якості життя людини, економія паливно-енергетичних ресурсів шляхом скорочення часу руху завдяки вибору оптимального маршруту, зменшення кількості і складності пробок за рахунок впровадження в інфраструктуру інтелектуальних світлофорів.

2) Для державних структур – міліція, дорожня інспекція – точна ідентифікація автомобілів, моніторинг позиціонування транспортних засобів в часі і просторі, включаючи викрадення, колізії, несанкціоновані маршрути. Істотне зниження аварійності за рахунок безпеки маневрів, зменшення наслідків дорожньо-транспортних пригод, підвищення безпеки і комфорту учасників дорожнього руху; моніторинг реальної швидкості руху всіх транспортних засобів та інформування про зони швидкісного режиму.

Цифровий моніторинг проїзду на заборонні знаки і сигнали світлофорів. Протидія угону транспортного засобу і самовільного покидання місця ДТП, завдяки моніторингу місця розташування кожного автомобіля. Інформування за допомогою тривожної кнопки спеціальних служб про події, що сталися на дорогах або з автомобілем. Автоматична реєстрація динаміки ДТП внутрішніми засобами ідентифікаційного модуля. Реєстрація стилю водіння для страхових компаній.

3) Для транспортних компаній – моніторинг позиціонування і пересування транспортних засобів, квазіоптимальне виконання замовлень з перевезення пасажирів і вантажів з точки зору мінімізації матеріальних і / або часових витрат;

4) Для водія – надання сервісів, пов’язаних з прокладанням квазіоптимальних маршрутів і графіка руху з урахуванням негативних чинників існуючої інфраструктури з метою мінімізації матеріальних і часових витрат в режимі реального часу; зниження аварійності завдяки моніторингу закритих для візуального перегляду ділянок дороги і прорахунку рівня безпеки маневрів. Попередження водія про потенційну небезпеку на замовленому маршруті, отриманої від хмари в процесі руху.

5) Для пасажира – надання сервісів з моніторингу позиціонування і руху пасажирських транспортних засобів на зупиночних або транспортних терміналах за допомогою використання стаціонарних моніторів або мобільних гаджетів для зв’язку з відповідними хмарними сервісами. Візуалізація на екрані в автомобілі критичних точок маршруту руху транспортного засобу в реальному масштабі часу шляхом використання камер відеоспостереження.

6) Для міста – економія палива, зменшення забруднення навколишнього середовища, скорочення часу руху, завдяки вибору оптимального маршруту, наданого хмарою. Зменшення кількості та складності пробок за рахунок планування руху транспортних засобів, що враховує майбутні маршрути інших учасників. Коригування маршруту руху

транспортного засобу в реальному часі при зміні дорожньої ситуації. Інтелектуальне управління циклами перемикання світлофорів залежно від дорожньої обстановки на перехрестях. Генерування аналітичних звітів і рекомендацій щодо поліпшення інфраструктури доріг, розстановки знаків, світлофорів та централізоване програмування циклів їх перемикання.

Аргументи проти впровадження хмари в масштабах країни: «Порушення права на недоторканність приватного життя, оскільки теоретично хмара здійснює тотальний моніторинг всіх транспортних засобів». Насправді сьогодні існує система законного перехоплення телекомунікацій, реалізована відповідно до міжнародних вимог. Але можливість перехоплення телефонних переговорів будь-якого абонента використовується тільки в ході слідства і з санкції суду. У тому числі є можливість відслідковувати місце знаходження абонента. Даний факт для законослухняних громадян ніякої проблеми не створює. «Додаткові витрати на придбання апаратно-програмних засобів ідентифікації та зв’язку транспорту з хмарою». Вартість цих коштів невисока і порівнянна із середньою сумою штрафу за порушення ПДР. Економічні переваги хмари, пов’язані з економією палива і зменшення часу, поїздки компенсують витрати протягом року.

В рамках системної інтеграції захищених від несанкціонованого доступу, хмарних компонентів (рис. 4.5): хмар моніторингу, управління і збереження історії, блоків радіочастотної ідентифікації транспорту, а також інтелектуальних сенсорів дорожньої інфраструктури, створюються мікросервіси квазіоптимального управління транспортом і дорожнім рухом в режимі реального часу для вирішення соціальних, гуманітарних, економічних та екологічних проблем:

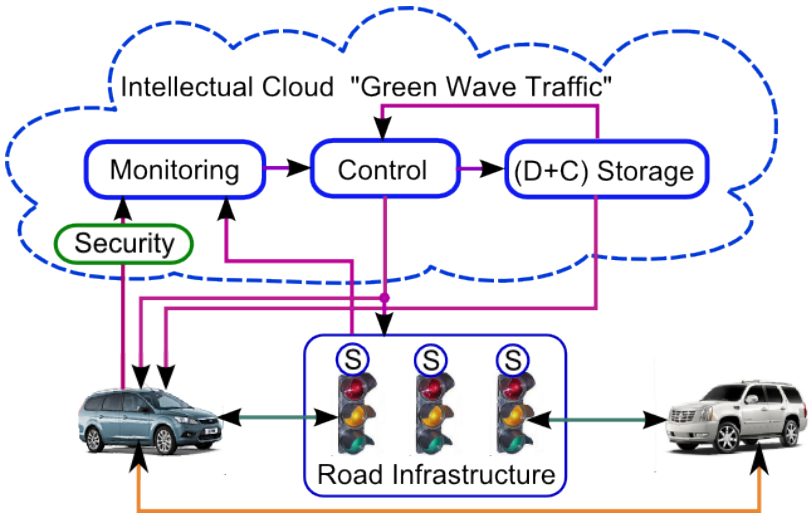



Рисунок 4.5 – Інтелектуальне хмара «Зелена хвиля»

1. Спеціального управління перемиканням дорожніх світлофорів в режимі on-line для автоматичного забезпечення безперешкодного руху по заданому маршруту спеціалізованих машин або кортежів (перевезення дітей, важливі державні чиновники, швидка допомога, пожежна служба, військові колони, небезпечні вантажі).

2. Оптимального управління світлофорами в режимі on-line на дорогах і перехрестях за допомогою точного цифрового моніторингу дорожнього руху шляхом використання RFID міток автомобіля, що дає можливість мінімізувати час проходження маршруту усіма учасниками дорожнього руху.

3. Планування оптимального маршруту для досягнення одного або кількох пунктів призначення автомобілем в часі і в просторі, що дає можливість зменшити часові та матеріальні витрати при заданій якості комфорту (час доби, року, дорожнє покриття, ліві повороти, погодні умови, затори, ремонтні роботи) пересування транспортного засобу.

4. Інтелектуальної історії пересування автомобіля, що має віртуальну модель в кіберпросторі – індивідуальну комірку в хмарі, інваріантну по відношенню до водіїв, які обслуговують транспортний засіб, що дає можливість відстежити будь-які пересування транспортного засобу в

минулому; а також прогнозувати бажані маршрути і поїздки в майбутньому без участі водія.

5. Сервіс хмарного моніторингу міток радіочастотної ідентифікації транспортних засобів в режимі on-line, який усуває автомобільні номери з системи обліку, що дає можливість: 1) Виключити безпосередню участь дорожньої поліції в фіксації порушень правил дорожнього руху (перевищення швидкості, проїзд на заборонні сигнали світлофорів, порушення правил маневрування); 2) Заощадити тисячі тонн металу на виготовлення номерів і спростити реєстрацію автомобілів при покупці з декількох днів до декількох хвилин; 3) Автоматизувати оформлення ДТП без участі співробітників дорожньої поліції шляхом цифрового моніторингу цифрової карти події, скопійованого з хмари; 4) Істотно (х2-х5) скоротити чисельний склад дорожньої поліції, оскільки історія переміщень автомобіля і його дорожніх порушень стає абсолютно прозорою для хмари, що дозволить автоматично списувати з рахунків водія вартість порушення відповідно до законодавства цієї країни; 5) Повністю виключити корупцію у відносинах водія з дорожньою поліцією, завдяки неможливості стерти інформацію про порушення в хмарі; 6) Практично ліквідувати криміналітет в області викрадення автомобілів, завдяки вбудованому в машину блоку радіочастотної ідентифікації, що забезпечує цілодобову спостережуваність транспорту в режимі on-line, якщо автомобіль фізично не знищений; 7) Спростити легалізацію водія шляхом ідентифікації водія ліцензії (driver's license) в списку дозволених осіб блоку радіочастотної ідентифікації автомобіля по протоколу "blue tooth", що дозволяє усунути виготовлення паперів та довіреностей на водіння автомобіля іншими особами; 8) У кілька разів зменшити кількість ДТП та істотно підвищити якість життя водіїв і пасажирів завдяки тотальному моніторингу порушень і невідворотності покарань за них; 9) Зменшити на 30% автомобільні викиди вуглекислого газу за рахунок зменшення часу простоїв на перехрестях і вибору оптимальних режимів і маршрутів пересування транспорту; 10) Забезпечити високу

ринкову привабливість хмари за рахунок продажу сервісів компаніям і приватним особам, що гарантує отримання високого прибутку – від сотень мільйонів до десятків мільярдів доларів, що масштабується залежно від площі покриття сервісами: міста, області, країни, всього світу.

Створення цифрової інфраструктури моніторингу та управління дорожнім рухом за допомогою детермінованих керуючих і спостерігаючих вузько випромінювань є майбутнє транспорту без водіїв, аварій, колізій на землі, воді і в повітрі. Для цього необхідні більш точні (до 10 сантиметрів) тривимірні стандарти ідентифікації простору планети з можливістю його моніторингу та управління.

## 4.2 Структура сервісів хмарного управління транспортом

Деталізується масштабована кіберфізична система цифрового моніторингу і хмарного управління дорожнім рухом в реальному масштабі часу на основі використання глобальних систем позиціонування і навігації, автомобільних комп’ютерів, навігаторів або мобільних гаджетів, розумних світлофорів, технологій Big Data і Internet of Things з метою підвищення якості і безпеки експлуатації транспортних засобів, а також мінімізації часових і матеріальних витрат при русі автомобілів за заданими маршрутами. Основна інноваційна ідея – поступове перенесення світлофорів з поверхні землі в віртуальний кібер простір для хмарного управління транспортом, обладнаним мобільним гаджетом (навігатором) або автомобільним комп’ютером, коли на екран (лобове скло) виводяться: карта з дорожніми знаками, маршрут прямування, координати учасника дорожнього руху та реальні сигнали віртуального розумного світлофора. Пропонується контейнер інноваційних технологій для комплексного вирішення соціальних, гуманітарних, економічних, паливно-енергетичних, страхових, кримінальних і екологічних проблем на основі створення і застосування хмарного сервісу цифрового моніторингу та управління транспортом. Згадані технології та функціональні компоненти інтегруються в просту для реалізації кібер

комп'ютингову модель інтерактивної взаємодії в реальному часі розумної інфраструктури цифрового моніторингу і хмарного управління з гаджетом водія або комп'ютером автомобіля.

Структура масштабованої розумної кіберфізичної компьютингової системи Cloud Traffic Control (рис. 4.6) включає наступні компоненти:

1) Оцифровані відношення: правила дорожнього руху, дорожнє законодавство, культура учасників дорожнього руху, традиції.

2) Транспортні засоби з цифровими ідентифікаторами, водії і пішоходи з розумними гаджетами.

3) Кіберфізична інфраструктура дорожнього руху, що включає дорожнє полотно і розв'язки, знаки, світлофори, парковки, заправки, пункти харчування та ремонту, електронні карти, засоби позиціонування і навігації, засоби телекомунікації, Edge Gadgets, цифрові ідентифікатори, Embedded MicroSystems and Smart Things.

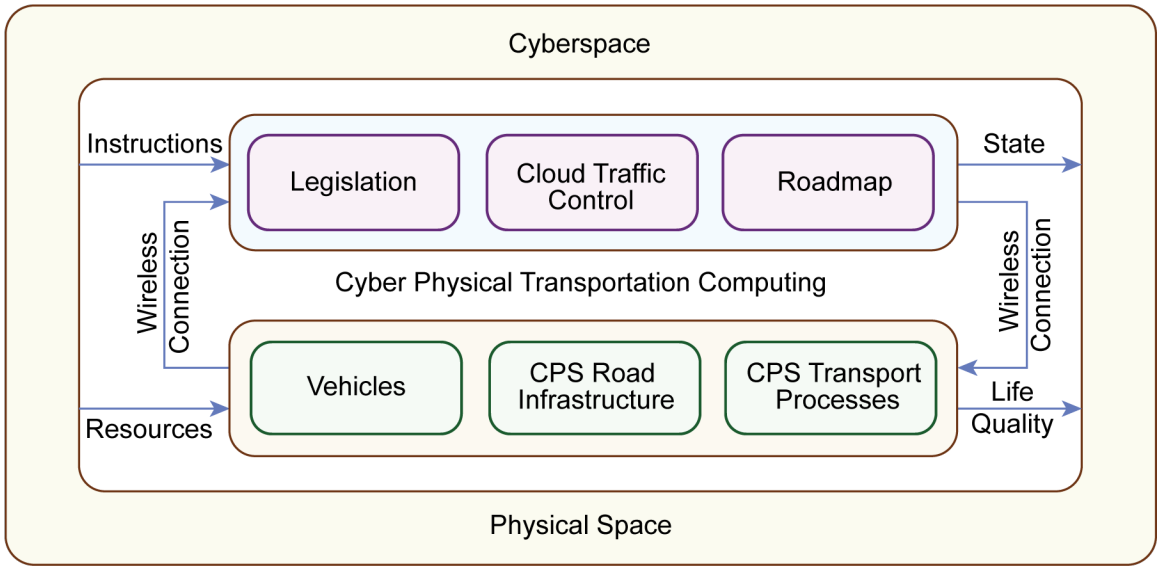



Рисунок 4.6 – Cyber Physical Transportation Computing

4) Хмарне планування і управління дорожнім рухом на основі цілодобового точного цифрового online моніторингу позиціонування і маршрутів руху транспортних засобів.

5) RoadMap сталого розвитку транспортної системи планети для підвищення якості життя людини і збереження екосистеми планети на основі: точного до 10 см позиціонування, створення достатньої потужності online 24/7 засобів телекомунікацій, переходу на використання електромобілів, очищення вулиць і доріг від знаків і світлофорів, driver-free car control, що виключає дорожні аварії, викрадення автомобілів, транспортні пробки і дорожню поліцію.

6) Комфортні, безаварійні, оцифровані транспортні процеси, що підлягають глобальній та локальній мінімізації за інтегральною метрикою: витрачання енергоресурсів, часу доставки вантажів і пасажирів, фінансових і матеріальних витрат на проектування і експлуатацію кіберфізичної інфраструктури.

Хмарні кіберфізичні компьютингові сервіси (рис. 4.7):

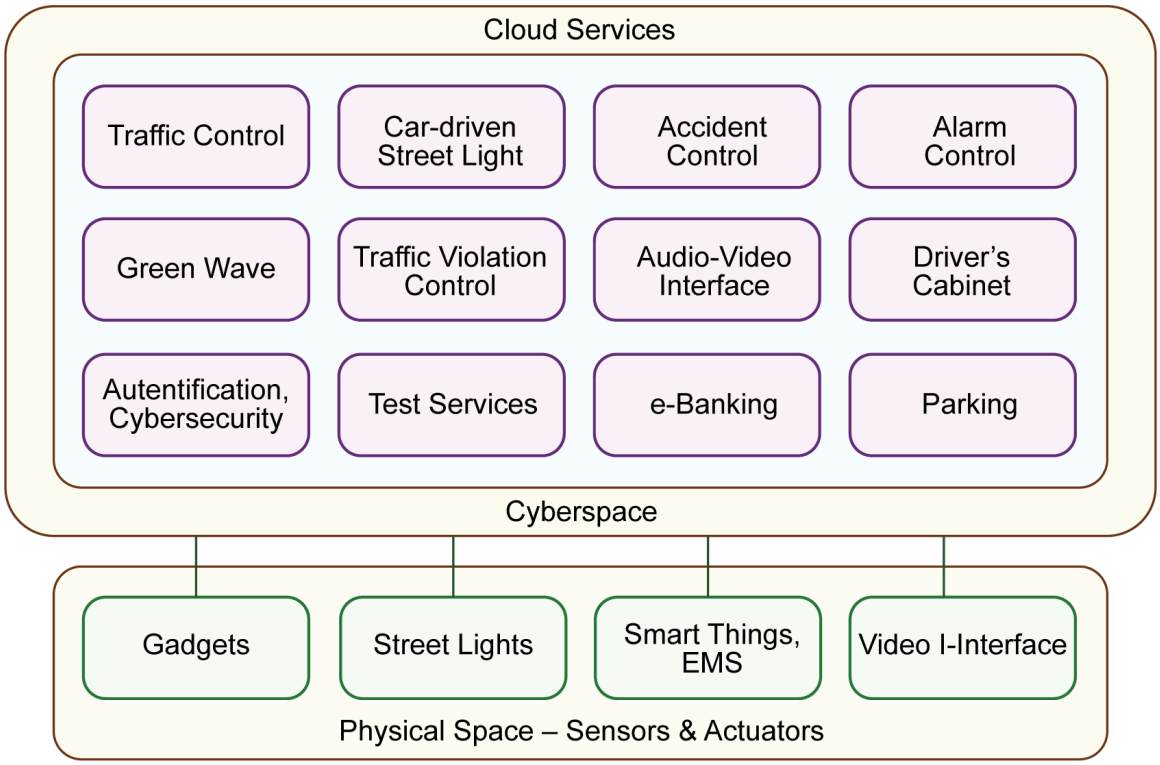



Рисунок 4.7 – Cloud Traffic Control Services

1) Хмарне online управління дорожнім рухом на основі цифрового моніторингу транспортних засобів і їх маршрутів, використання віртуальних знаків і світлофорів. Персональне планування і online квазіоптимальне управління маршрутом руху на основі делегування хмарі координат автомобіля і замовленого шляху.

2) Car-driven хмарне online управління віртуальним і/або реальним світлофором на перехресті шляхом цифрового моніторингу позиціонування транспортних засобів.

3) Точний цифровий моніторинг і організаційне управління дорожньо-транспортною пригодою на основі просторово-часового та причинно-наслідкового аналізу колізії, яка трапилася.

4) Цифровий моніторинг несанкціонованого використання транспортного засобу в режимі 24/7, візуалізація позиціонування і екстрене управління блокуванням автомобіля.

5) Забезпечення безперешкодного руху за заданим маршрутом транспорту, спеціалізованих машин або кортежів (перевезення дітей, державні чиновники, швидка допомога, пожежна служба, військові колони, небезпечні вантажі).

6) Online сервіс попередження: водія про потенційну небезпеку на замовленому маршруті, отриманої від хмари в процесі руху; спеціальних служб про події і порушення, що трапилися на дорогах або з автомобілем. Сервіс аналітичних звітів і рекомендацій щодо поліпшення інфраструктури доріг, аналітика популярних запитуваних маршрутів, планування збільшення пропускної здатності завантажених ділянок. Online сервіс моніторингу стану дорожнього покриття в інтеграції з системами прогнозування і відображення погодних умов.

7) Сервіс ергономічного online надання відео-аудіо інформації водієві на склі-моніторі зі звуковим дублюванням дорожньої обстановки.

8) Сервіс особистого віртуального кабінету водія-автомобіля, що зберігає переваги в маршрутах, режимах руху, колізіях і інші особливості подорожей.

9) Сервіс захисту персональних даних і аутентифікації водія-автомобіля на основі персональних гаджетів, RFID-міток транспортного засобу і первинних біометричних показників.

10) Online сервіс тестування обладнання автомобіля за запитом водія з особистого віртуального кабінету при виникненні несподіваних поломок. Дана послуга була запропонована компанією Synopsys, Virtual Hardware ECU для істотного підвищення якості хмарного тестового сервісу automotive electronic an software systems.

11) Online банкінг товарів і послуг, пов’язаних з обслуговуванням автомобіля на заправках, станціях технічного обслуговування і користування платними дорогами і парковками.

12) Цифровий моніторинг і хмарне управління парковками на основі Embedded MicroSystems and Smart Things.

### 4.3 Розумний хмарний світлофор

Ключовий інноваційний компонент інфраструктури TCS – віртуальний і/або реальний смарт-світлофор (Smart-Streetlight – SS) функціонально являє собою стаціонарний мікроконтролер (цифрову систему на кристалі) з приймачем, безпосередньо пов’язаний з хмарою управління трафіком. Він організовує бездротову мережу для взаємодії з автомобільними комп’ютерами (Car-Computer – CC) на дистанції до 100 метрів за протоколом Wi-Fi. Стан світлофора є доступним для моніторингу в хмарному сервісі Інтернету і виводиться на екрани всіх машин, які на маршруті транспортування перетинають перехрестя і тих, які знаходяться в зоні його компетенції. Сигналами світлофора можуть управляти дистанційно спеціальні служби, або запити від автомобілів, які перетинають перехрестя. Розумний світлофор сканує і обробляє всі заявки від транспортних засобів,

надходять через TCS-хмара в реальному масштабі часу. Він генерує відповідні сигнали управління, функціонально залежні від дорожньої обстановки на сторонах перехрестя. Всі світлофори міста об'єднуються в мережевий технологічний шар хмарного сервісу, який обслуговує інфраструктуру вулиць і доріг в інтелектуальному і/або автономному режимах з можливістю ручного і/або дистанційного керування. Метод управління світлофором використовує взяття булевої похідної за лініями руху. Останні мають лінійні графіки накопичення $N^{\bullet} = k*t + b$ і пропуску автомобілів $N^{\circ} = -k*t + b$ через перехрестя – вузьке горлечко дорожньої інфраструктури – з періодом сканування автомобілів, рівним 1 секунді. Тут k – пропускна здатність або кількість проїжджаючих автомобілів через перехрестя в секунду, t – реальний час, b – вихідна кількість машин на перехресті. Обчислення або оновлення похідної за лініями руху перехрестя для формування сигналів світлофора здійснюється синхронно з вказаним періодом, але зміна його стану, за наявністю черг на всіх сторонах перехрестя, здійснюється з затримкою, що мінімізує колізії (сумарний час простою автомобілів) на всіх лініях руху транспортних потоків. Цикл перемикання світлофора має мінімальний період, що дорівнює часу проїзду перехрестя автомобілем.

Світлофор, як спочатку примітивний засіб регулювання рухом на перехресті, поступово трансформується в спеціалізовану кіберфізичну систему моніторингу та управління, що включає компоненти: LED-монітор, відеокамера, Wi-Fi прийомо-передавач для зв'язку з автомобільними комп'ютерами та Інтернетом, програмні засоби інтелектуального управління транспортом. Останні вже сьогодні використовують високу математичну культуру: теорії автоматів, черг та знань, мережі Петрі і нейронні мережі, генетичні алгоритми і прискореного навчання (reinforcement learning). Система світлофорних агентів або сенсорів для моніторингу транспорту включає: досить дорогі відеодетектори і індуктивні петлі, вбудовані в дорожнє покриття, значно рідше використовуються звукові індикатори шуму,

мікрохвильові радари. Позбутися від усіх агентів світлофора можна шляхом цифрової ідентифікації автомобілів на основі точного позиціонування автомобільного гаджета або комп'ютера. Сьогодні таким агентом може бути мобільний телефон водія авто, який за допомогою глобальної системи позиціонування делегує своє місце розташування в хмару управління трафіком. У відповідь мобільний телефон отримує всі сервіси з управління дорожнім рухом: сигнали хмарного віртуального світлофора, синхронізовані з реальними світлофорами, оптимальні маршрути пересування, «зелену хвилю» при неінтенсивному трафіку, економію часу і палива. Взаємодія світлофора з системою управління рухом по перехрестю показана на рис. 4.8, де представлені: Streetlight – LED-монітор, Gadget – гаджет автомобіля або мобільний телефон, автомобіль, ідентифікований на хмарній мапі гаджетом водія, GPS Satellite – для точного позиціонування транспорту або гаджета, Camera – відеокамера спостереження рухом на перехресті, Computer – для реалізації розумного управління світлофором, TCS-хмара-сервіс управління трафіком в Інтернеті.

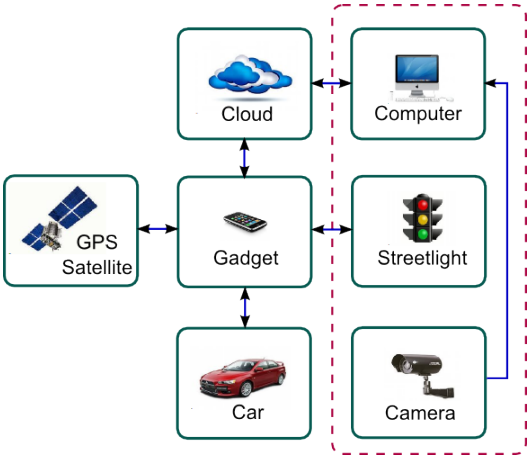



Рисунок 4.8 – Взаємодія світлофора з інфраструктурою і TCS-хмарою

Червоним пунктиром обведені фізичні компоненти, які згодом перемістяться у віртуальний простір, що дозволить зробити інфраструктуру доріг набагато більше зеленої. Суттєвим доповненням до функціональності управління світлофором може служити технологічне рішення, запропоноване

в [140-141]. Аналіз показує, що до 2020 року більшість ДТП відбуватиметься на перехрестях. Найбільш частими причинами аварій тут є неадекватні спостереження (44,1%), помилкові припущення про маневрах інших автомобілів (8,4%), повороти з утрудненим оглядом (7,8%), незаконні маневри (6,8%), внутрішні відволікання уваги (5,7%), недооцінка інтервалу або швидкості інших транспортних засобів (5,5%). Підвищення безаварійності переїзду перехрестя автори пов'язують з розподіленою взаємодією (vehicular ad hoc networks – VANET) автомобілів між собою за протоколом V2V (IEEE 802.11p) за допомогою використання бездротових гаджетів, що передбачає наявність віртуального світлофора VTL для формування керуючих сигналів за принципом Car2X. Збільшення пропускної здатності перехрестя без додаткових витрат на розширення інфраструктури автори пов'язують з двома взаємодіями Car2Car і Car2 (X) VTL, які формують інтелектуальну децентралізовану мережу, на 60% підвищує пропускну здатність перехрестя. Подальше вдосконалення вуличного світлофора в інфраструктурі міст пов'язане з його перетворенням в кіберфізичную систему управління рухом на перехресті, структура якої представлена на рис. 4.9.

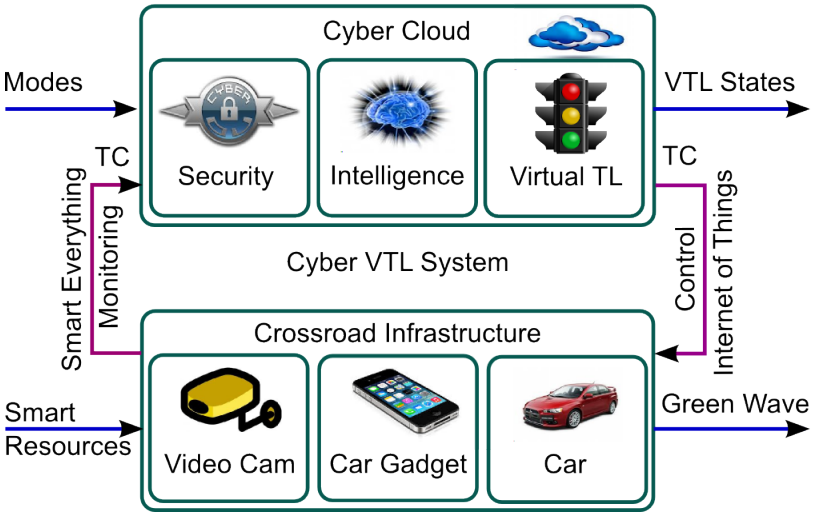



Рисунок 4.9 – Кіберфізична система VTL-управління перехрестям

Кібер VTL-система призначена для організації «зеленої хвилі» при

проїзді перехрестя автомобілем і містить два основних компоненти: віртуальний – Cyber Cloud і фізичний – Crossroad Infrastructure. Перший з них призначений для формування керуючих впливів, що визначають сигнали світлофора, які залежать від передісторії, реального трафіку на перехресті. Другий компонент виконує моніторинг дорожньої обстановки за допомогою відеокамер, розумних датчиків і автомобільних гаджетів. Взаємний зв'язок хмарного світлофора з фізичною інфраструктурою перехрестя здійснюється за допомогою телекомунікаційних засобів: GPS Satellite, мобільного Інтернету, GSM / GPRS. Разом з тим, кіберсистема має можливість ручного управління світлофором шляхом використання входу Mode, а також спостереження за його станами в процесі функціонування за виходом VTL States. Два блоки Intelligence і Security забезпечують реалізацію розумних алгоритмів управління в залежності від реального трафіку, а також захист світлофора від несанкціонованого доступу. Інноваційна цінність кіберфізичної системи VTL-управління транспортом на перехресті полягає в використанні віртуального світлофора, його інтелектуальному управлінні залежно від дорожньої обстановки, а також у моніторингу останньої шляхом збору сигналів позиціонування і маршрутизації транспорту від гаджетів автомобілів.

Car-driven управління світлофором. Структура розумного світлофора характеризується оптимальним керуванням сигналами, які залежать від кількості транспортних засобів на смугах руху перехрестя. Кожна смуга руху складається з паралельних непересічних потоків транспортних засобів, спрямованих один проти одного. Ці потоки формують стан регістра (пам`яті), де кожен біт впливає на формування сигналу світлофора. Перевага віддається регістру, який має максимальну кількість одиничних значень. Беручи до уваги, що регістр може мати нульові і одиничні значення, в кожному циклі управління виконується регістрова операція зсуву вліво з ущільненням одиниць. Вона виключає арифметичні інструкції множення, додавання, ділення, вирахування з процесу управління інтелектуальним світлофором, що

значно спрощує віртуальну (фізичну) реалізацію світлофора і збільшує швидкодію його роботи.

Для ущільнення даних за один такт використовується оригінальне запатентоване рішення у вигляді регістра зсуву одиничних значень вліво з ущільненням [7]. Схема для обчислення керуючого сигналу, як гри двох транспортних потоків, представлена на рис. 4.10. Існують два потоки A і B, які за своєю суттю конфліктують (суперечать один одному). Вхід m визначає штучно створені переваги, які повинні мати місце в міській інфраструктурі. За замовчуванням ці переваги визначаються одиничними сигналами регістру m. Процедура управління світлофором складається з трьох етапів:

1) Зрушення вліво з ущільненням

$A = SLC(A),\ B = SLC(B).$

2) Визначення смуги переможця

$a = (A\&B) \oplus A,\ \ b = (A\&B) \oplus B$

2) Формування світлофорних сигналів перехрестя

$$Q^a = \bigvee_{i=1}^{n} a_i \; Q^b = \bigvee_{i=1}^{n} b_i \; i = \overline{1,n}$$

де n – число бітів регістрів (кількість автомобілів для смуг, що не перетинаються).

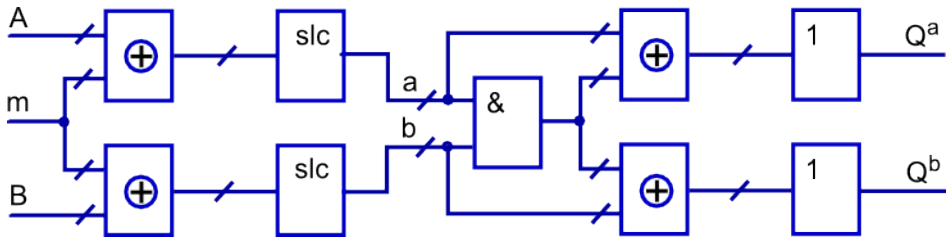



Рисунок 4.10 – Cloud Sequenser світлофора на простому перехресті

Регістрові змінні a і b, що представляють вектори стислих вліво одиничних значень, об’єднані і інвертовані для одночасної реалізації xor-операції. Результати у вигляді станів регістрів є двома входами or-елемента, які утворюють стан двох булевих вихідних змінних для створення трьох комбінацій: 00, 01 і 10. Нульове значення (0) однієї з двох вихідних змінних є

кращим рішенням, яке необхідно вибрати. Два нульових стани означають, що обидва рішення еквівалентні рівню переваги. В цьому випадку за замовчуванням вибирається потік А. Єдина комбінація двох булевих змінних неможлива.

Таким чином, точне управління світлофором відповідно до потоків трафіку на перехресті може і повинно виконуватися на основі тільки векторних логічних операцій and, or, not, xor, slc, без використання арифметичних функцій, що дає можливість проектувати віртуальні / фізичні and-or мультипроцесори, які значно скорочують часову затримку обчислень в циклі виконання хмарного управління перехрестям. У разі керування складним перетином доріг, де існує n конфліктуючих потоків трафіку, використовується модифікована схема управління, яка реалізується у вигляді векторно-логічного секвенсора, показаного на рис. 4.11. Тут вхідні регістри або пам'яті формують суперпозицію одиничних значень, що відповідають кількості автомобілів на непересічних напрямках руху. На одному з виходів секвенсора управління світлофором сигнал має нульове значення, яке ідентифікує потік трафіку з максимальною кількістю автомобілів. Установка сигналу трафіку в нульове значення являє собою компромісне рішення, яке залежить не тільки від кількості транспортних засобів на смузі руху, а й від числа автомобілів на інших потоках, які перетинають перехрестя.

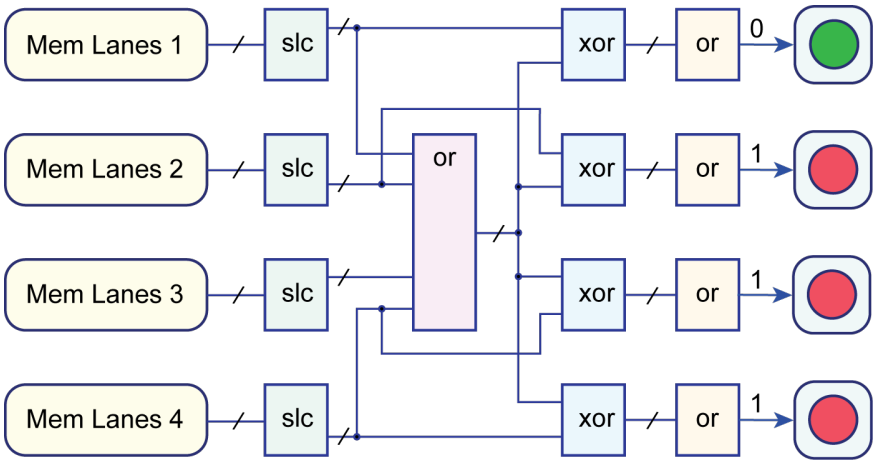



Рисунок 4.11 – Cloud Sequenser світлофора на складному перехресті

Представлені схеми двох логічних секвенсоров інтегруються в структуру хмарного керування транспортними засобами на перехресті, рис. 4.12. При цьому світлофор поступово повинен трансформуватися в чисто хмарний сервіс, виведений на дисплей автомобіля.

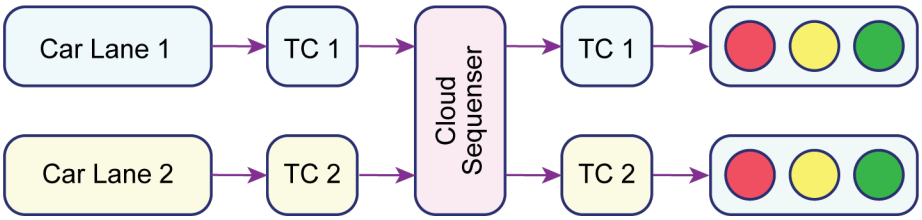



Рисунок 4.12 – Структура хмарного управління транспортом

Тут Cloud Sequenser представлений логічною схемою формування керуючого сигналу для включення зеленого світлофора, зображеної на рис. 4.10, 4.11. Модуль Car Lane позначає множину автомобілів, що створюють лінію перетину перехрестя, яка трансформується у відповідний регістр з кількістю одиниць, що дорівнює кількості машин. Модуль ТС є телекомунікаційний канал, що передає координати кожного автомобіля в хмару, а також із хмари до світлофора (автомобілю). Якщо число ліній перетину перехрестя більше двох, то структура управління збільшується пропорційно до їх кількості.

## 4.4. Структурно-аналітична модель хмарного управління

Являє собою кібер-фізичну систему у вигляді інтелектуальної хмари управління дорожнім рухом (Cyber Physical System – Smart Cloud Traffic Control) в реальному масштабі часу на основі створення хмарної інфраструктури дорожнього руху, інтегрованої з віртуальними вуличними світлофорами і дорожніми знаками, мобільними засобами ідентифікації автомобілів з метою підвищення якості та безпеки пересування транспортних засобів, мінімізації часових і матеріальних витрат при виконанні заданих маршрутів.

Інтелектуальне хмарне управління дорожнім рухом (Smart Cloud Traffic Control) має на меті включення параметра часу в цифрову карту планети, а також поступове перенесення дорожніх знаків і світлофорів в хмари, що радикально озеленює інфраструктуру дорожнього руху на землі і створює потенційні можливості для економії: тисяч тонн металу на виготовлення світлофорів і непотрібних автомобільних номерів, мільйонів кіловат годин-електроенергії на підтримку працездатності, мільйонів доларів на установку світлофорів, дорожніх знаків і експлуатаційні витрати, а також зменшення часу установки і актуалізації світлофорів у віртуальній інфраструктурі міст з декількох днів до декількох хвилин.

Формальна модель кіберсистеми представлена у вигляді двох хмарних компонентів або механізмів [1]: 1) f – моніторинг і управління; 2) g – виконавчі інфраструктурні механізми, які пов'язані між собою сигналами моніторингу, управління і ініціювання обох компонентів для реалізації сервісів. Аналітична форма [4] завдання TCS-системи і її структурний еквівалент зображені на рис. 4.13.

$$A = (f, g, \mu, \upsilon, X, R, Y, P),$$
$$\begin{cases} Y_t = f[(X, R, \mu)_t, Y_{t-1}]; \\ P_t = g[(X, R, \upsilon, Y)_t]. \end{cases}$$
$$X = (v, p, s); Y = [G(k), L, M, \vec{P}];$$
$$R = (G, \vec{P})_R; P = (G(k), L, M, \vec{P})_P;$$
$$\mu = [G(k), L, M, \vec{P}];$$
$$\upsilon = \{L_i, L_t, L_h, L_x\} = L;$$
$$G\{R, P\} = \{G(k), L, M, \vec{P}\}.$$

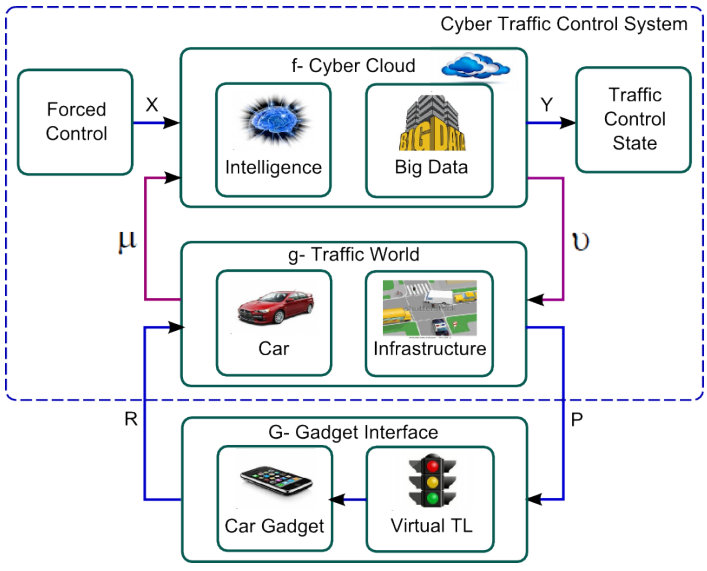



Рисунок 4.13 – Аналітична і структурна форми TCS-системи

У моделі фігурують компоненти $A = (f, g, \mu, \upsilon, X, R, Y, P)$ відповідно: блоки управління і виконання, сигнали моніторингу та управління, входи

керуючих завдань і виконавчих ресурсів, виходи індикації стану алгоритму реалізації завдання і надання сервісу. Тут також присутні сигнали зовнішнього управління дорожнім рухом $X=(v,p,s)$ для регулювання проїзду урядових персон, поліцейських машин і автомобілів спеціального призначення відповідно. Сигнали управління транспортом $\upsilon=\{L_i,L_t,L_h,L_x\}=L$ використовують віртуальний світлофор, який працює в режимах: 1) інтелектуальний, функціонально залежний від дорожньої обстановки; 2) автоматичний, з фіксованими періодами перемикання; 3) ручний віртуальний режим на основі цифрового моніторингу перехрестя на екрані поліцейського комп'ютера, аналог – управління повітряним транспортом за допомогою монітора диспетчера в аеропорту; 4) екстренна зупинка $L_x\in\upsilon$ транспортного засобу за цифровим запитом поліції, який візуалізується на екрані монітора гаджета автомобіля. Хмарному моніторингу підлягають відповідно: $\mu=[G(k),L,M,\vec{P}]$ всі мобільні гаджети автомобілів з їх координатами, стан світлофорів, прив’язані до карти місцевості M, а також виконання замовлених маршрутів руху транспорту.

Перенесення світлофора $L\in\{\upsilon,P\}$ з реального перехрестя на хмарний завершує створення віртуальної інфраструктури планети, утворюючи замкнутий цикл системи моніторингу та управління з єдиним реальним компонентом у вигляді мобільного гаджета $G\{R,P\}=\{G(k),L,M,\vec{P}\}$ учасника дорожнього руху (УДД). Гаджет реалізує функцію інтерфейсу для зв’язку з хмарою: R-вхід до хмари – замовлення сервісу $R=(G,\vec{P})_R$ (делегування в хмару ID-гаджета і шляхи пересування) і P-вихід з нього – отримання сервісу $P=[M,G(k),\vec{P},L]$ (карта, координати гаджета, оптимальний маршрут, світлофор).

Учасник отримує сервіс віконного скролінгу карти і сигнали світлофорів в реальному масштабі часу за шляхом руху $\vec{P}$, якщо він делегує свій гаджет в хмару. При замовленому шляху він додатково отримує квазіоптимальний маршрут руху і пріоритетний проїзд світлофорів.

Насправді систему з позиції користувача створюють два компоненти: хмара і гаджет. При цьому новизна та оригінальність системи полягає в наданні хмарного сервісу – сигналів світлофора на екрані гаджета учасника дорожнього руху. Все інше: карти, маршрути руху – вже існують і працюють. Впровадження запропонованої TCS-системи буде відбуватися шляхом створення віртуальних світлофорів, які дублюють реальні в синхронному режимі, а потім поступове усунення всіх фізичних пристроїв і знаків земної інфраструктури дорожнього руху за мірою набуття водіями нової технологічної культури еволюційним шляхом. Всі великі міста вже мають фактично централізоване компёютерне (хмарне) управління світлофорами. Тому перенесення світлофорів і дорожніх знаків до хмари не буде пов'язане з істотними додатковими витратами, а скоріше навпаки великі експлуатаційні витрати на підтримку світлофорної і знакової інфраструктури міст в працездатному стані трансформуються в нуль.

Мобільний гаджет транспортного засобу G є головним керуючим впливом для TCS-хмари, так само як і воно є основним споживачем світлофорних сигналів L управління рухом автомобіля, що виводяться на лобове скло, $L = \{L_i, L_t, L_h, L_x\} = F(L, G, V, T, D, \vec{P})$ де V – сигнали спецуправління, T – програмований цикл автономного управління світлофором, D – накопичені інтелектуальні статистичні дані по світлофору (проспекту, району), в тому числі, враховують пори року і доби, $\vec{P}$ – актуальні замовлення на маршрути руху транспорту. Створення віртуальної системи світлофорів дає можливість практично без фінансових, часових, матеріальних і енергетичних витрат розміщувати шляхом програмування нові світлофори у віртуальному просторі, так само як і видаляти їх з хмари в процесі модернізації інфраструктури. Візуалізація на лобовому склі (мобільному моніторі) сигналів світлофора і голосове дублювання підвищить якість і безпеку дорожнього руху, знизить аварійну обстановку як для водія, так і для інфраструктури міст в цілому. Хмарний світлофор, як цифровий сигнал, на відміну від аналогового сприйняття водієм реального світлофора, є більш

надійним засобом управління транспортом, в тому числі, і для подальшого впровадження в дорожній рух автопілота, що сприймає тільки детерміновані сигнали управління.

Учасники дорожнього руху ідентифікуються в хмарі гаджетом або iPhone, який паруется при посадці в автомобіль. Статус учасника підвищується при проходженні світлофорів, якщо маршрут руху замовляється заздалегідь. Інші учасники дорожнього руху (пішоходи, мотоцикліст і велосипедисти) також мають право замовляти маршрут, підвищуючи свій статус для використання світлофорів. Пішоходи мають можливість отримати сервіс для замовлення комбінованого маршруту, що включає всі види наземного і підземного транспорту (автобуси, метро). Для автомобіля система управління формує функціонал, що оптимізує критерій якості обслуговування, який залежить від наступних змінних (час, довжина і якість маршруту): $Q = \min f(T, \vec{P}, K)$.

Для світлофора система управління формує функціонал, що мінімізує сумарний час простою автомобілів протягом доби (Z – цикл перемикання світлофора):

$$Q = \min \frac{1}{n} \sum_{i=1}^{n} \left[ \frac{T_i(\vec{P}_i, V_i, L_i, J_i)}{Z(\vec{P}_i, V_i, J_i)} \right]^{-1},$$

де в чисельнику і знаменнику представлені функціональні залежності часу простою і циклу від зазначених в дужках параметрів. Фактично підсумовуються частини світлофорних циклів, необхідні для проїзду кожному автомобілю через перехрестя. Якщо в результаті вийде оцінка якості, близька до одиниці Q = 1, то перехрестя функціонує нормально. В іншому випадку необхідно модифікувати цикл перемикання або реконструювати перехрестя. Для інфраструктури міста або району система управління формує функціонал, що оптимізує критерій якості обслуговування автомобілів за проміжок часу (годину, добу), що залежить від сумарного часу проїзду автомобілів за заданими маршрутами з

рекомендованою швидкістю, часу простою автомобілів на світлофорах і в пробках:

$$Q = \min \frac{1}{n} \sum_{i=1}^{n} \left[ \frac{T_i(\vec{P}_i, V_i, L_i, J_i)}{T_i(\vec{P}_i, V_i)} \right]^{-1}.$$

Для проспекту міста система управління формує функціонал, що оптимізує критерій сумарного часу проїзду автомобілів від початку і до кінця вулиці за проміжок часу (годину, добу):

$$Q = \min \frac{1}{n} \sum_{i=1}^{n} \left[ \frac{T_i(V_i, L_i, J_i)}{T_i(V_i)} \right]^{-1}.$$

Деталізована структура кібер-фізичної системи управління і моніторингу представлена на рис. 4.14, де основними блоками є: гаджет автомобіля, який одночасно є сенсором і активатором TCS, а також хмара, яка ділиться на дві частини. Перша з них g містить існуючу в часі інфраструктуру з компонентами: карта місцевості, координати гаджета – автомобіля, світлофори і дорожні знаки, а також пам’ять для зберігання замовлених маршрутів і статистики пересування транспорту. Друга частина хмари f представлена блоками пам’яті, моніторингу та управління, а також модулем захисту від несанкціонованого доступу.

Таким чином, запропонована інноваційна привабливість TCS-хмари характеризується наявністю взаємопов’язаних компонентів: фізичних – автомобіль з гаджетом і віртуальних – підсистем моніторингу та управління дорожнім рухом на основі хмарних світлофорів, що дає можливість: 1) квазіоптимально управляти кожним транспортним засобом в режимі реального часу на основі використання існуючих каналів зв’язку і мобільних гаджетів, спарених з автомобільними комп’ютерами; 2) оптимізувати процеси оптимального за часом, витратами і якістю управління дорожнім рухом для вирішення соціальних, гуманітарних, економічних, кримінальних, страхових і екологічних проблем; 3) радикально зменшити вартість реальної

інфраструктури дорожнього руху і заощадити: матеріали для виготовлення дорожніх знаків і світлофорів, електроенергію на забезпечення їх працездатності, фінанси на установку світлофорів і експлуатаційні витрати за рахунок перенесення світлофорів у віртуальну інфраструктуру дорожнього руху планети.

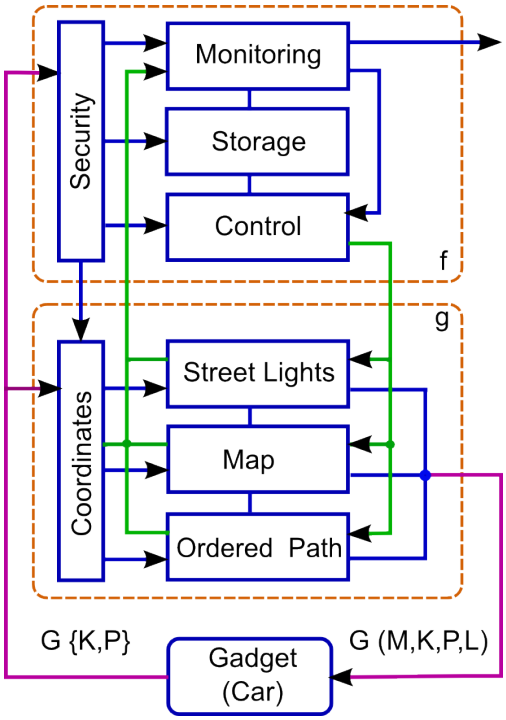



Рисунок 4.14 – Структура TCS-системи

Алгоритм роботи хмарної системи управління містить наступні кроки: при посадці в автомобіль водій за допомогою гаджета замовляє маршрут пересування шляхом звернення до кібер-системі, яка верифікує валідність користувача за допомогою блоку «Security», який дозволить вхід до хмари, що визначає координату гаджета або автомобіля у вікні карти місцевості, обчислює оптимальний шлях у часі і просторі, а також здійснює супровід транспорту в реальному часі шляхом надання сервісів по управлінню за допомогою хмарних світлофорів, прив’язаних до координат або перехресть на маршруті руху. За відсутністю замовленого шляху користувач отримує тільки вікно карти місцевості зі світлофорами на поточному шляху

пересування транспортного засобу, але при цьому сама кіберсистема може запропонувати найбільш ймовірний маршрут руху, виходячи зі статистики поїздок, що зберігається в блоці історії «Storage». Гаджет може бути використаний і пішоходом для переходу через перехрестя і проспекти, замовлення маршруту пересування на основі використання існуючих транспортних засобів: автобус, метрополітен, поїзд, літак. Режими роботи кіберсистеми для користувача: 1) супровід учасника дорожнього руху шляхом надання інфраструктурної карти, координат і сигналів світлофора; 2) замовлення маршруту пересування, коли до функціональності першого режиму додається квазіоптимальний шлях, що мінімізує час, дистанцію, якість пересування, включаючи статистичну світлофорну зелену хвилю по шляху проходження; 3) особливі замовлення маршруту учасниками $X=(v,p,s)$, коли попереду та за шляхом проходження спеціального транспорту забезпечується запланована і гарантована зелена хвиля, що пропонує іншим учасникам дорожнього руху не перешкоджати спеціальним машинам, аж до зупинки, шляхом цифрової та голосової індентифікації відповідних сигналів управління на екрані гаджетів автомобілів в зоні взаємного інфраструктурного впливу транспорту.

На рис. 4.15 основним компонентом є пам'ять як будь-яка субстанція для зберігання інформації, на якій визначаються структурні компоненти обчислень (Cloud Control і Car & Infrastructure Execution). Експлуатація системи здійснюється за допомогою моніторингу (сигналів датчиків) і управління (сигналів актюації) компонентами комп’ютингової системи. Входи і виходи системи з'єднуються із зовнішнім кібер-фізичним простором.

На рис. 4.16 показана кібер-фізична система для цифрового моніторингу і хмарного управління трафіком, яка реалізована за технологією Internet-of-Things. Інноваційна особливість визначається відсутністю входів і виходів, які ідентифікуються з Big Data.

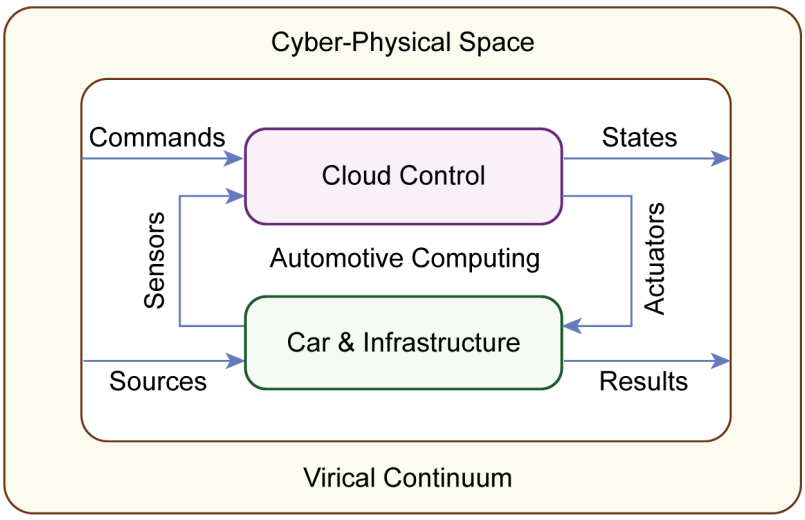



Рисунок 4.15 – Automotive Computing for Cloud Traffic Control

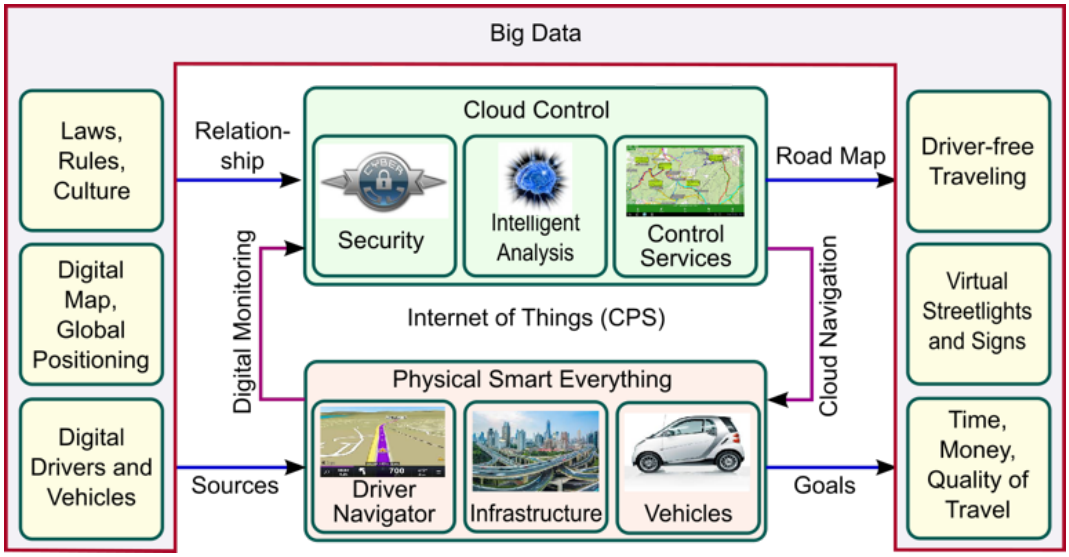



Рисунок 4.16 – Global CPS for digital monitoring and cloud traffic control

Це означає, що вхідні дані пов’язані з інформаційними масивами кібер-фізичного простору. Результат аналізу даних, що надається хмарними службами CPS, записується в спеціалізовані масиви великих даних. Природно, кіберфізична система має монітори і інтелектуальні датчики, розташовані в фізичному просторі планети. Ці монітори: сучасні автомобільні комп’ютери і гаджети водіїв або мандрівників. Гаджети є виконавчими механізмами, які контролюють маршрутизацію транспортних засобів. Обов’язковою умовою автономного пересування транспорту є

глобальне позиціонування автомобіля в оцифрованному фізичному просторі на основі цифрової карти Google з точністю до 10 см.

На рис. 4.17 представлена доменно і IP-ідентифікована структура взаємодії кіберфізичних компонентів, що реалізують хмарний комп'ютинг-сервіс Cloud Traffic Control в контейнерно-орієнтованому середовищі компанії Docker, що працює в Google Computing Engine. Мета – створення відкритого і доступного для користувача сервісів Cloud Traffic Control, призначених для зменшення часових і матеріальних витрат на виконання заданого маршруту руху.

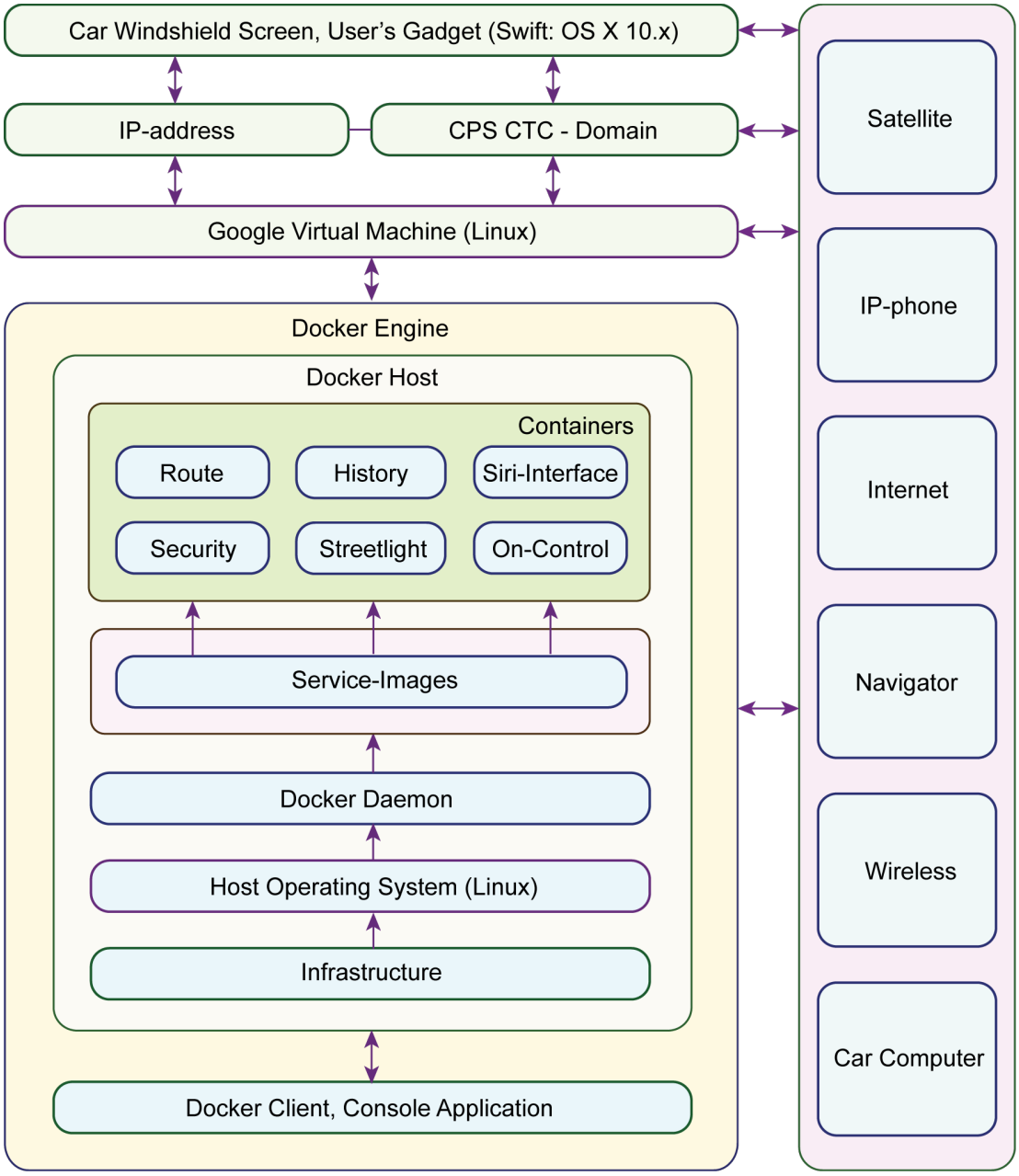



Рисунок 4.17 – Хмарні сервіси Cloud Traffic Control

Задачі: 1) Створення хмарного середовища визначення квазіоптимального маршруту на основі контейнетов. 2) Розробка структури прототипу хмарного управління вуличним світлофором. 3) Інтеграція сервісів мовного введення інформації, зберігання історії виконання маршрутів, захисту даних і моніторингу дорожньої обстановки. 4) Верифікація структури і функціональностей хмарного сервісу.

Огляд ринку проектування сучасних програмних продуктів показав, що в останні три роки набирають популярність гнучкі і масштабовані технології проектування програмних продуктів, що використовують контейнеризацію, орієнтовані на мінімізацію циклу time-to-maret. Основна ідея контейнера полягає в створенні захищеної і доступного віртуального середовища проектування і експлуатації у кіберпросторі на основі використання локальної або глобальної апаратури. На інфраструктуру встановлюється практично будь-яка операційна система Host operating system, зручна для розробника, наприклад, Ubuntu, Debian. У локально виконаній hardware інфраструктурі контейнерів можна інсталювати також OSX і Windows.

Розробник інсталюється Docker Engine, якщо він не встановлений спочатку, який за допомогою Docker Daemon (реалізація на Golang, C), управляє контейнерами, запущеними на основі попередньо побудованих образів (Images). Контейнери Containers уявляють собою легковажну віртуальну машину, під якою працюють функціональні блоки: Route, History, Siri, Security, Streetlight, Online Control, реалізовані на Swift (або Jawa), і відповідні бібліотеки. Docker Engine представляє собою комп’ютингову систему контейнеризації, яка складається з двох взаємодіючих між собою компонентів: Docker Host, Docker Client (Console Application), які можна інтерпретувати як виконавчий Host і керуючого (Client) механізму хмарної системи Cloud Traffic Control. Далі Docker Engine є компонентом, який входить до складу Google Virtual Machine, якій ставиться у відповідність IP-address and CyUni domain, що робить розробку, а далі хмарний сервіс,

доступним для фахівців і студентів. Останньою ланкою у вертикальній зв’язці (Back-End, Front-End) є User's Gadget (Swift OS X.10x). На кінцевому пристрої користувача установлюються додатки, здатні привести результати роботи хмарних сервісів Cloud Traffic Control до мультіоконному формату візуалізації даних, зручною для сприйняття людиною. Слід зазначити, що набагато більш технологічно створити Front-End у вигляді додатку на JavaScript, HTML, CSS, який буде працювати через Internet browser. Це гарантує відсутність блокування клієнта з боку постачальника конкретної ОС (vendor lock). Для мобільних пристроїв доцільніше використовувати додаток індивідуального клієнта під кожну ОС. Переваги структурної організації хмарного сервісу Cloud Traffic Control: 1) Захищеність і доступність контейнерів для командної розробки і верифікації моделей і методів. 2) Глобальна доступність сервісів в кіберпросторі планети. 3) Контейнерна розширюваність хмарного комп’ютингу при реалізації нових мікросервісів, що відповідають додатковим методам і алгоритмам Cloud Traffic Control. 4) Просторово-часова інваріантність проектування і експлуатації кібер-фізичної системи, спрямована на доступність хмарних сервісів з будь-якої точки планети в будь-який час доби. 5) Інфрструктурна незалежність хмарного віртуального комп’ютингу забезпечує безвідмовну роботу невидимою апаратури, її своєчасну замінюваність на стороні Google (Amazon, IBM) без додаткових фінансових витрат для власника сервісів.

## 4.5 Модельно-фізичні експеріменти на road-інфраструктурі

Заміна традиційного світлофора з фіксованим часом перемикання сигналів на розумне online управління дорожнім рухом (підрозділ 4.3), залежне від фактичних транспортних потоків, істотно впливає на час проходження перехрестя автомобілем. Ефективність роботи перехрестя може бути підвищена за рахунок скорочення часу простою зеленого сигналу, коли на дозволеній «зеленій» a-смузі немає автомобілів, а на альтернативній b-смузі вони є. В цьому випадку розумний світлофор перемикається для

пропуску автомобілів по b-смузі. Практично розумний світлофор виконує роль постового-регулювальника дорожнього руху, який здатний розрулити складні дорожні колізії на перехрестях, якщо є простої зеленого сигналу світлофора. Час простою зеленого сигналу світлофора на непересічних лініях перехрестя за наявністю транспортних засобів на червоному світлофорі є резерв часу для підвищення пропускної здатності за рахунок розумного регулювання. Тому функція зміни стану світлофора ставиться у залежність не прямо від транспортного потоку, а від резерву часу, що визначається інтервалом простою зеленого кольору світлофора (T0) на обох лініях (a, b) руху автомобілів: $F = f([T0]_i^a, [T0]_i^b)$.

Діаграми взаємодії транспортних потоків на двох пересічних лініях руху автомобілів при зеленому сигналі світлофора представлені на рис. 4.18.

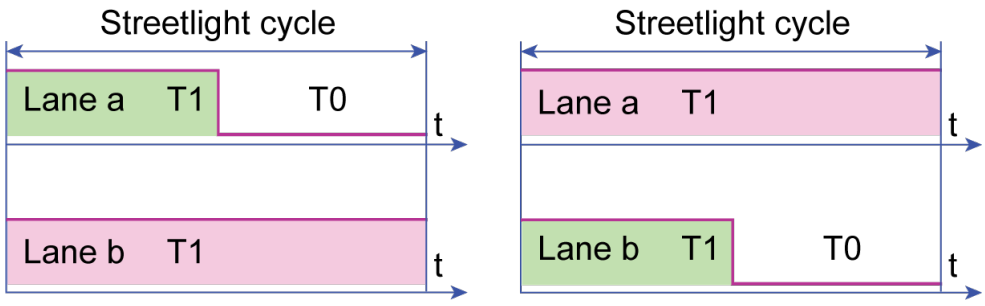



Рисунок 4.18 – Взаємодія транспортних потоків на зеленому світлофорі

Тут представлені два зелених сигнали на напрямках перетину, де інтервал T0 – час простою зеленого світлофора або резерв, який можна використовувати для перпендикулярного напрямку руху. Інтегральний час простою зеленого світлофора D на обох лініях руху рівнозначного перехрестя може бути метрикою оцінювання пропускної здатності (n – кількість циклів в інтервалі спостереження):

$$T_i^a = [T0]_i^a + [T1]_i^a;\ T_i^b = [T0]_i^b + [T1]_i^b;\ D = \sum_{1=1}^{n} ([T0]_i^a + [T0]_i^b).$$

Природно, що параметр D можна і потрібно зменшувати до нуля шляхом введення розумного світлофора, що працює під управлінням реальних транспортних потоків в режимі online. На рис. 4.19 представлені графіки проведених натурних і модельних експериментів на 10 "рівнозначних" перехрестях проспекту Науки в Харкові, обладнаних світлофорами. В залежності від часу доби (інтервал спостереження – 4 години) результати інтегрального оцінювання простою зеленого світлофора істотно відрізняються: в нічний час зелений світлофор на другорядних вулицях, що перетинають проспект, максимально впливає на затримку транспорту, що рухається прямо по проспекту.

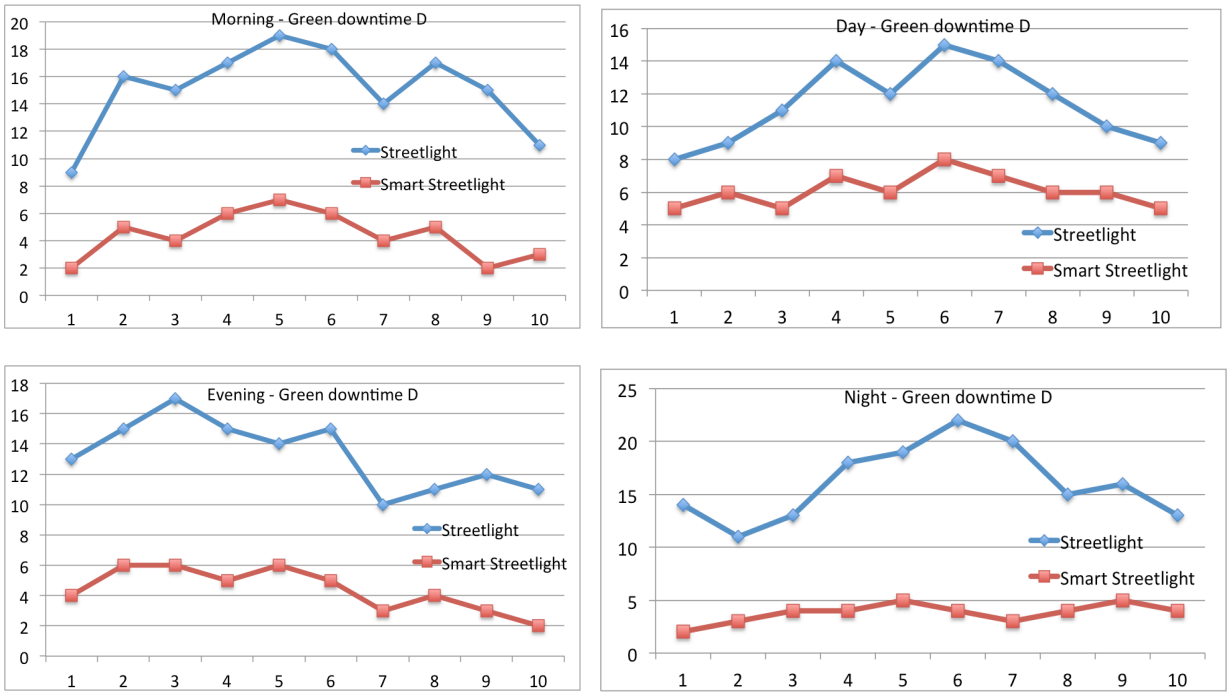



Рисунок 4.19 – Графіки простоїв на звичайному і розумному світлофорах

Нульове значення часу простою при використанні розумного світлофора повёязано з інертністю перемикання сигналу через жовтий колір і часом переїзду автомобіля через перехрестя, що становить перехідний процес транспортних потоків на перетині доріг при зміні сигналів світлофора.

Таким чином, впровадження розумних світлофорів дозволить на 10-30 відсотків снизити час простою зеленого сигналу, що в масштабах міста ско-

ротить інтегральний час знаходження транспорту в шляху і витрати на енергоносії не менше, ніж на 10 відсотків.

Далі представлені експерименти на двох типах топологій перехресть. Середньої складності міське перехрестя зображене на рис. 4.20. Воно містить по дві смуги руху в кожному з чотирьох напрямків, які забезпечують маневр автомобіля за трьома напрямками.

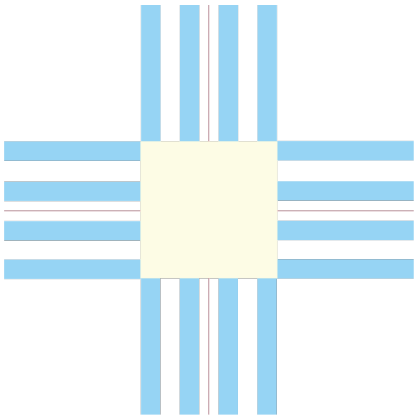


Рисунок 4.20 – Приклад топології перехрестя

Сигнали світлофора формують чотири стани перехрестя, які забезпечують два типи топологій (діагональна і прямокутна) для формування транспортних потоків при проїзді перехресть, представлених на рис. 4.21, 4.22. Перша і друга топології забезпечують два типи проїзду, які створюють метрику (n – число фрагментів смуг $P_i$ до і після перетину перехрестя):

$$Q = \frac{1}{n} \times \sum_{i=1}^{n} P_i ,$$

інтегрально визначаються наступними оцінками, що формують прохідність перехрестя:

$$Q_1 = \frac{10}{16} + \frac{10}{16} + \frac{6}{16} + \frac{6}{16} = \frac{32}{16} = 2{,}0; \quad Q_2 = \frac{8}{16} + \frac{8}{16} + \frac{8}{16} + \frac{8}{16} = \frac{8}{16} = 2{,}0.$$

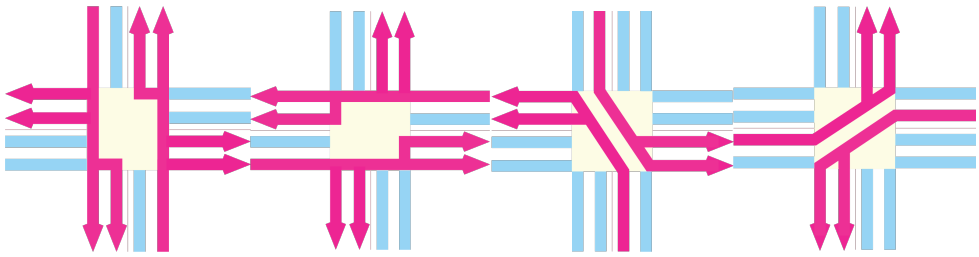


Рисунок 4.21 – Діагональна топологія транспортних потоків

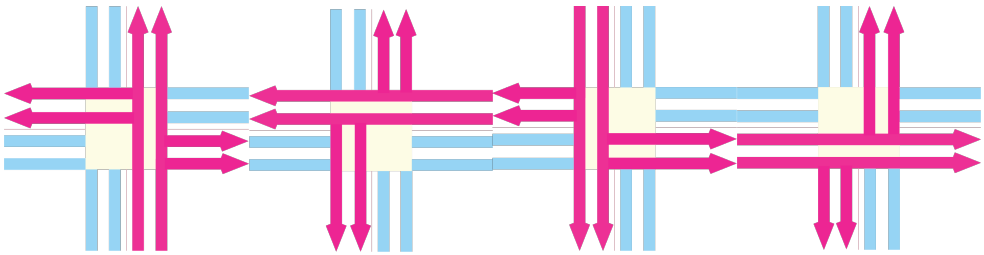


Рисунок 4.22 – Прямокутна топологія транспортних потоків на світлофорі

Таким чином, обидві топології мають однакові оцінки. Однак друга топологія транспортних потоків, що формуються світлофорами, є більш ефективною для підвищення пропускної здатності, якщо необхідно швидко розвантажити транспортний потік з однієї вхідної черги. Важливі не тільки потенційні смуги перехрестя, але управління зеленими сигналами світлофора, які фактично формують транспортні потоки або ефективність використання смуг.

Кількість автомобілів N, що проїжджають через рівнозначне перехрестя, можна вирахувати, задавши наступні параметри відповідно: $V_{car}, L_{car}, \Delta T, \tau^x, \tau^1, S$ – швидкість автомобіля на перехресті; довжина автомобіля з дистанцією до транспорту, що йде попереду; проміжок часу, за який визначається кількість транспорту, що проїхав через перехрестя; час утримання жовтого сигналу світлофора; час утримання зеленого сигналу світлофора; кількість станів світлофора на перехресті. Формула для обчислення пропускної здатності однополосного перехрестя матиме такий вигляд:

$$N = \frac{V_{car}}{L_{car}}\left[\Delta T - \frac{S \times \tau^{x} \times \Delta T}{(\tau^{l} + \tau^{x}) \times S}\right] = \frac{V_{car}}{L_{car}}\left[\Delta T - \frac{\tau^{x} \times \Delta T}{(\tau^{l} + \tau^{x})}\right].$$

Підставивши дані за всіма параметрами, заданими у формулі, можна отримати модельне число транспортних засобів, що проїжджають через перехрестя без колізій, в даному випадку за одну годину або 3600 секунд (діагональна топологія транспортних потоків на перехресті) в одному напрямку:

$$N = \frac{2{,}77\text{m}/\text{c}}{7\text{m}}\left[3600\text{c} - \frac{4\text{c} \times 3600\text{c}}{(30\text{c} + 4\text{c})}\right] = 1257.$$

З урахуванням існування P паралельних транспортних смуг, що проходять через перехрестя, пропускна здатність буде пропорційно збільшуватися:

$$N = P \times \frac{V}{L}\left[\Delta T - \frac{\tau^{x} \times \Delta T}{(\tau^{l} + \tau^{x})}\right].$$

У прямокутної (діагональної) топології транспортних потоків на перехресті на кожному сигналі світлофора число транспортних паралельних потоків дорівнює двом (P = 2). Тому модельне значення пропускної здатності буде визначатися таким виразом:

$$N = 2 \times \frac{2{,}77\text{m}/\text{c}}{7\text{m}}\left[3600\text{c} - \frac{4\text{c} \times 3600\text{c}}{(30\text{c} + 4\text{c})}\right] = 2504.$$

У реальності пропускна здатність перехрестя може виявитися істотно нижче. Це пов'язано з тим, що існують простої зеленого сигналу світлофора, затягування жовтого сигналу, наявність інертності в реакції водіїв на дорожню обстановку, аварійні ситуації, проїзди через перехрестя спеціальних

транспортних засобів, транспортів великої довжини. Підтвердженням цього факту є вимір реальних транспортних потоків на діагональному перехресті в місті Харкові. Виявилося, що за 4 години він обслужив 8,4 тисячі автомобілів в період з 9 до 13 годин. Модельний експеримент на перехресті показав можливу пропускну здатність транспорту за 4 години, рівну 10016 автомобілів. Це на 19 відсотків краще, ніж в реальності. Виключити істотну різницю між модельними і реальними потоками транспорту, які проїжджають через перехрестя, слід шляхом впровадження розумних світлофорів, які враховують вхідні транспортні потоки на перехрестях, попереджувальний час простою зеленого сигналу.

## 4.6 Висновки до розділу 4

Важко передбачити і перерахувати всі позитивні соціальні, технологічні та технічні наслідки радикального перетворення існуючого світу після впровадження цифрових сервісів інтелектуальної хмари дорожнього руху. Через 5 років слід очікувати масової появи безпілотного транспорту при виконанні маршрутів [SAE World Congress eXperience 2017, Detroit, Forbes]. Для найближчого майбутнього нижче представлені інноваційні науково-технічні рішення соціальних, гуманітарних, економічних та екологічних проблем, пов'язані з хмарним моніторингом і управлінням дорожнього руху.

*Наукова новизна* Cyber Physical Systems "Smart Cloud Traffic Control" характеризується введенням в цифрову карту планети параметра часу і перенесенням світлофорів у віртуальний кіберпростір, що завершує створення віртуальної інфраструктури дорожнього руху для підвищення якості життя водіїв, зменшення часу перебування в дорозі і витрат на паливо, економії тисяч тонн металу для виготовлення світлофорів і дорожніх знаків, мільйонів кіловат-годин електроенергії на підтримку працездатності, мільйонів доларів на установку світлофорів і експлуатаційні витрати, а також

зменшення часу установки і актуалізації світлофорів і дорожніх знаків у віртуальній інфраструктурі міст до декількох хвилин, що у сукупності дає можливість автоматизувати процеси квазіоптимального управління транспортом і дорожнім рухом в режимі реального часу і вирішувати соціальні, гуманітарні, економічні та екологічні проблеми.

*Практична значимість* визначається наданням нових послуг державним структурам, учасникам дорожнього руху, дорожній поліції, спеціальним службам і організаціям:

1) Сервіс безперешкодного руху за заданим маршрутом спеціалізованих машин або кортежів (перевезення дітей, важливі державні чиновники, швидка допомога, пожежна служба, військові колони, небезпечні вантажі).

2) Сервіс оптимального управління віртуальними світлофорами в режимі on-line на дорогах і перехрестях за допомогою точного цифрового моніторингу дорожнього руху шляхом використання гаджета водія або комп'ютера автомобіля, що дає можливість мінімізувати час проходження маршруту усіма учасниками дорожнього руху.

4) Сервіс інтелектуальної історії пересування автомобіля, що має віртуальну модель у кіберпросторі – індивідуальну комірку в хмарі, інваріантну по відношенню до водіїв, які обслуговують транспортний засіб, що дає можливість відстежити будь-які пересування транспортного засобу в минулому, а також прогнозувати бажані маршрути і поїздки в майбутньому вже без участі водія.

5) Сервіс хмарного моніторингу (мобільних) цифрових автометрічних паспортів транспортних засобів в режимі on-line дозволить прибрати автомобільні номери з системи обліку, і як наслідок: виключити безпосередню участь дорожньої поліції в фіксації порушень правил

дорожнього руху (перевищення швидкості, проїзд на заборонні сигнали світлофорів, порушення правил маневрування, легкі зіткнення); заощадити тисячі тонн металу на виготовлення металевих номерів і спростити реєстрацію автомобілів при покупці з декількох днів до декількох хвилин; автоматизувати оформлення ДТП без участі співробітників дорожньої поліції шляхом цифрового моніторингу цифрової карти події, скопійованої з хмари; скоротити чисельний склад дорожньої поліції, оскільки історія переміщень автомобіля і його дорожніх порушень стає абсолютно прозорою для хмари, що дозволить автоматично списувати з рахунків водія вартість порушень відповідно до законодавства конкретної країни; виключити корупцію у відносинах водія з дорожньою поліцією, завдяки неможливості стерти інформацію про порушення в хмарі; ліквідувати криміналітет в області викрадення автомобілів, завдяки вбудованому в машину цифровому автометричному паспорту, що забезпечує цілодобову спостережуваність транспорту в режимі on-line, якщо автомобіль фізично не знищений; спростити легалізацію водія шляхом ідентифікації водія ліцензії (driver's license) в списку дозволених осіб цифрового паспорта хмарного осередку автомобіля за протоколом «blue tooth», що дозволяє усунути виготовлення паперів та довіреностей на водіння автомобіля іншими особами і виключити посередницькі послуги нотаріуса; зменшити кількість ДТП, істотно підвищити якість життя водіїв і пасажирів, завдяки тотальному моніторингу порушень і невідворотності покарань; зменшити автомобільні викиди вуглекислого газу за рахунок зниження часу простоїв на перехрестях і вибору оптимальних режимів і маршрутів пересування транспорту; забезпечити високу ринкову привабливість хмари за рахунок продажу сервісів компаніям і приватним особам, що гарантує отримання високого прибутку – від сотень мільйонів до мільярдів доларів, що масштабується залежно від площі покриття сервісами: міста, країни, всього світу.

Запропонована архітектура розумного хмарного світлофора, яка характеризується використанням логічних операцій і часом простою зеленого сигналу, що дає можливість на 10-30 відсотків зменшити час простою зеленого світлофора і істотно збільшити пропускну здатність транспортних потоків на перехресті доріг. Впровадження розумних світлофорів в масштабах міста дозволить скоротити інтегральний час знаходження транспорту в шляху і витрати на енергоносії не менше, ніж на 10 відсотків.

# 5 МУЛЬТИВЕРСНИЙ ПАРАЛЕЛЬНИЙ СИНТЕЗ ЦИФРОВИХ СТРУКТУР МОНІТОРИНГА ТА УПРАВЛІННЯ ТРАФІКОМ НА ОСНОВІ SystemC СПЕЦИФІКАЦІЇ

Пропонується мультиверсний паралельний синтез цифрових структур моніторинга та управління трафиком на основі SystemC специфікації, що використовується при створенні модуля Sherlock, який призначений для точного моніторингу даних щодо позиціонування та функціонування автомобиля в процесі виконання заданого маршруту. Мультиверсний синтез також застосовується при створенні программних продуктів хмарного сервісу Cloud Traffic Control, а також при розв'язанні задач тестування та діагностування програмно-апаратных компонентів кіберфізичної системи. Пропонується модель мультиверсної системи захисту всіх програмно-технічних компонентів хмарного сервісу управління транспортом у межах технічного та організаційного кібер-шару проекту для подальшого виконання заходів з інтегрування реальної керуючої інфраструктури доріг та автомобілів з віртуальним кіберпростором. Виконується тестування, верифікація та діагностування програмно-апаратних засобів хмарного керування трафіком на стадіях проектування та експлуатації.

## 5.1 Вступ

Створення цифрових систем на кристалах грає ключову роль для розвитку комп'ютерних технологій, що використовуються в економіці, соціології, медицині, бізнесі, промисловості, освіті, які формують кіберекосістему планети: розподілені центри великих даних, Internet of Things, Internet of Everything, cyber physical system, cloud -mobile computing, service computing. Важливо відзначити експоненційне зростання не тільки продуктивності комп'ютерних систем, але і продуктивності інженерів,

завдяки створенню хмарних сервісів проектування апаратних і програмних додатків, інтегрованих в поняття комп'ютинг (рис. 5.1).

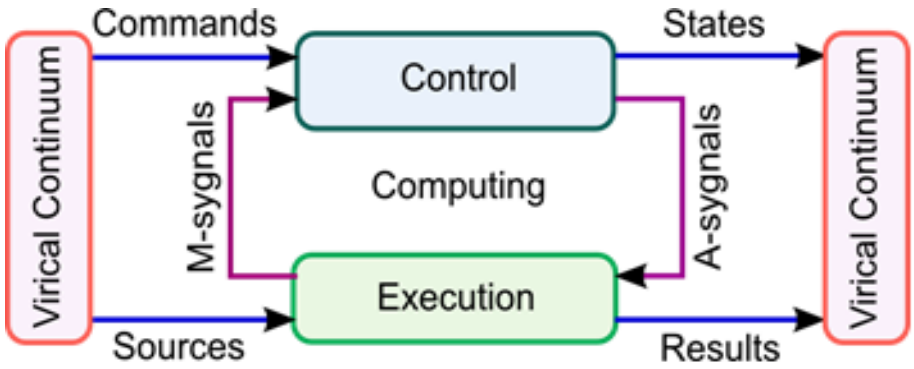



Рисунок 5.1 – Комп'ютинг для створення ціфрових систем на кристалі

Технології хмарних сервісів по синтезу цифрових систем на кристалах системного рівня дозволяють створювати виконувані специфікації – опис проекту на функціональному рівні, що можна верифікувати в віддаленому online-режимі. На функціональному рівні специфікація відображає питання «що робити?», а не «як робити?». Таким чином, вона не містить вимог до структури або архітектури цільової платформи. При цьому компонентами створення деякої цифрової системи виступають вже сервіси, а не функціональні елементи. Така заміна дає суттєвий виграш у часі проектування, що визначається швидкодією процедур верифікації та тестування, за рахунок варіювання параметрами системного, RT і вентильного рівнів, домагаючись кращих характеристик реалізації цифрового виробу на кристалі.

Мета – істотне зменшення часу проектування обчислювальних архітектур і підвищення якості цифрових виробів шляхом мультиверсного синтезу структури цифрового виробу на основі заданої специфікації в середовищі SystemC (C++) і автоматичному підборі функціональних компонентів за рахунок паралельного синтезу і верифікації архітектурних рішень системного рівня відповідно до запропонованої метрики.

## 5.2 Методи системного та архітектурного синтезу інтерфейсних структур

Модель системного рівня може не мати однозначного інтерфейсу на логічному рівні. Розроблений метод визначає інтерфейс моделі, щоб її можна було однозначно представити на логічному рівні. Метод розглядає прототипи C ++ - функцій: кількість аргументів, їх типи і типи значень, що повертаються. Представлено відповідність стандартних C++ – типів типам мови VHDL. Наступний метод синтезує протокол взаємодії з модулем на логічному рівні. Для обраного протоколу взаємодії будується граф-схема алгоритму, яка об'єднується з граф-схемою проектованого пристрою. Таким чином, синтезується пристрій, пов'язаний з іншими, що працюють на тій же самій шині даних.

Зазвичай високорівневі моделі не мають поняття модельного часу: всі обчислення відбуваються на одному умовному такті роботи. Моделі ж на логічному рівні працюють за тактами. Таким чином, можна виділити дві підзадачі: а) визначення відповідності аргументів функції на системному рівні і сигналів на логічному рівні; б) визначення протоколу взаємодії з синтезованою функцією. Тут протокол – це послідовність зміни інформаційних і керуючих сигналів схеми, яка призводить до виконання синтезованої функції.

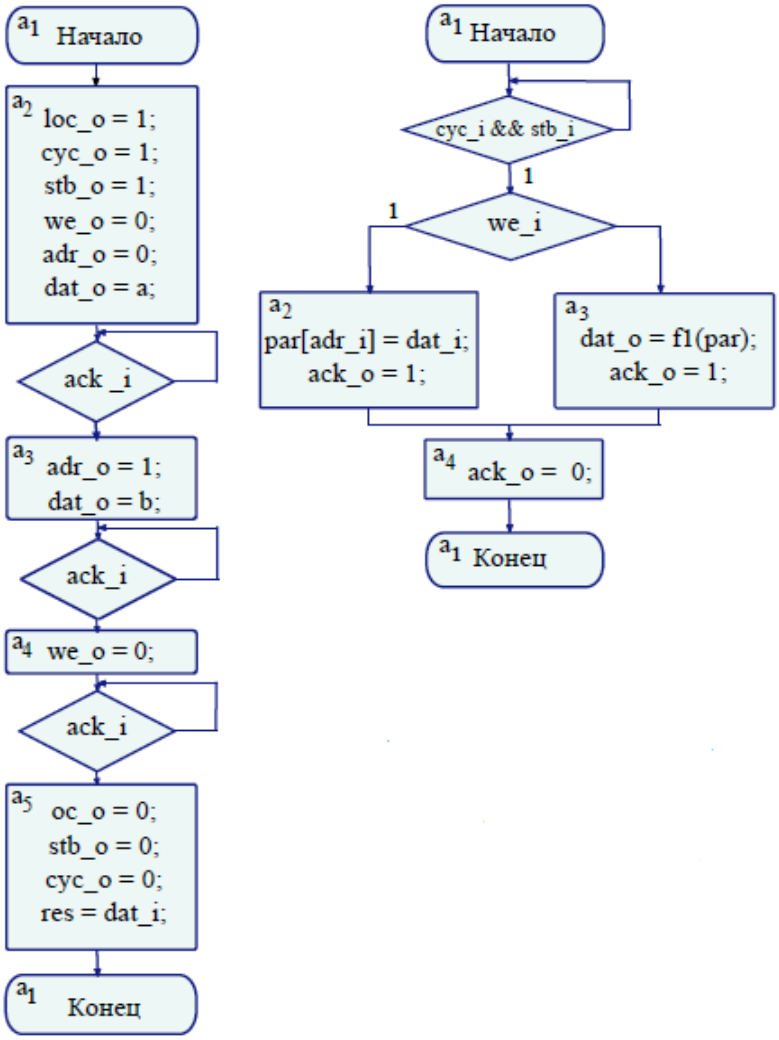
Розглянемо синтез інтерфейсів для популярного стандарту шини системи на кристалі Wishbone. Цей стандарт передбачає різні сполучення модулів в системі: з'єднання «точка-точка»; конвеєрне з'єднання (для обробки потоку даних); з'єднання «багато до багатьох» через загальну шину; з'єднання «багато до багатьох» через комутатор; з'єднання в складі мережі на кристалі.

Wishbone є шина даних, розмір якої може бути від 8 до 64 біт, але завжди кратний одному байту. Таким чином, для передачі параметрів вико-

ристовується послідовне завантаження значень у внутрішню пам'ять відомого модуля. Для цього використовується спеціальна адресація регістрів, які розташовані в інтерфейсі веденого модуля.

У тому випадку, якщо розмір шини менше 32 біт, потрібно кілька циклів читання результату з регістру R1. Можна розрахувати кількість тактів, необхідних для передачі параметрів. Нехай Pi – вектор параметрів функції, i = 1, ..., n, де n – кількість параметрів, переданих у функцію; S(Pi) – розмір параметра Pi в байтах; M – розрядність шини даних в байтах.

На рис. 5.2 показані алгоритми роботи ведучого і веденого інтерфейсів в системі Wishbone.




а б

Рисунок 5.2 – Алгоритми ведучого і веденого інтерфейсів в системі Wishbone

У стані a3 показаний виклик функції f1 з параметром par. Відповідно до цієї граф-схеми алгоритму (ГСА) обчислення функції займає рівно один такт роботи автомата. Це може бути справедливо лише для високорівневих моделей без відсутності модельного часу. У системах на структурному рівні опису такого можна домогтися тільки для комбінаційних функцій. Для автоматів, які потребують більшу кількість станів для завершення, необхідно змінити ГСА (рис. 5.2, а) наступиним чином. На місці стану a3 необхідно підставити ГСА викликається функції f1.

## 5.3 Моделі опису логічних блоків системного рівня

На першому етапі розробки моделі контейнера «вектор» системного рівня був формалізований інтерфейс SystemC-модуля. На рис. 5.3 показана система, що включає всі необхідні компоненти системного рівня. Clock – це генератор тактових імпульсів. Контейнер «вектор» – синхронний пристрій, у якого всі операції здійснюються по передньому фронті сигналу синхронізації.

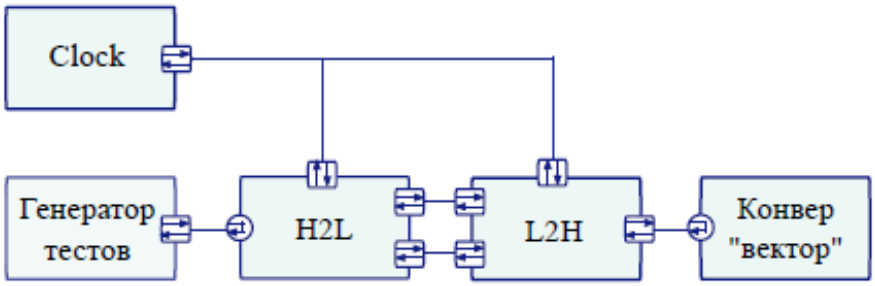



Рисунок 5.3 – Повний комплект модулів і транзактор для розробки системної моделі конвеєра "вектор"

Генератор тестів – це високорівневий модуль, який містить генератор тестів для контейнера і аналізатор відповідних реакцій. Цей генератор буде повторно використаний для верифікації моделі на логічному рівні.

H2L (high level to low level) – це транзактор, який перетворює високорівневі виклики функцій контейнера до протоколу на рівні перемикання логічних сигналів.

L2H (low level to high level) – це транзактор, який здійснює перетворення, зворотне H2L – протокол рівня логічних сигналів до високорівневих викликам функцій контейнера «вектор». На рис. 5.4 показаний детальний інтерфейс транзактору L2H. Цей же інтерфейс матиме майбутня апаратна реалізація.

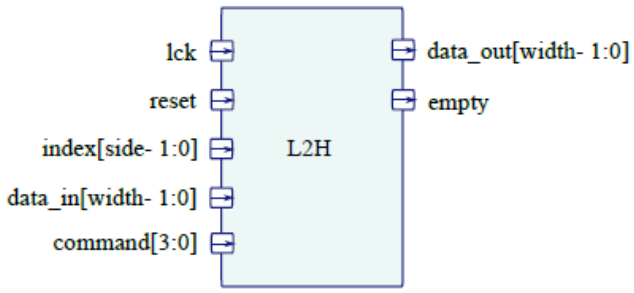



Рисунок 5.4 – Інтерфейс логічного рівня моделі контейнера "вектор"

Логічна модель контейнера «вектор» реалізована за допомогою примітивів блокової пам'яті кристала Xilinx Virtex-5. Один блок пам'яті кристала розрахований на 36 кбіт даних і може бути налаштований або як два незалежних модуля пам'яті по 18 кбіт, або як один модуль ємністю 36 кбіт. Можна користуватися наступними конфігураціями пам'яті: 32К по одному біту, 16К по 2, 8К по 4, 4К по 9, 2К по 18 або 1К по 36 біт.

Вибір кристалів Xilinx Virtex визначається наступними факторами. Ці кристали доступні на ринку в низьких партіях за прийнятні ціни. Тут вони істотно виграють перед кристалами жорсткої логіки, для яких тільки підготовчі витрати на виготовлення маски і пробну серію можуть досягати мільйонів доларів. ПЛІС Virtex також має можливість багаторазового перепрограмування, що дає можливість відчувати різні конфігурації проектовано-

го пристрою, а також виправляти логічні дефекти в проекті. Кристали жорсткої логіки такої можливості не мають.

Кристали Xilinx Virtex також мають необхідні ресурси для реалізації контейнера «вектор»: елементи блокової пам'яті. Блокова пам'ять підтримує синхронні операції запису/читання, два порти доступу повністю симетричні і працюють незалежно один від одного, розділяючи дані в пам'яті.

Обмеження логічної моделі: контейнер займає всю пам'ять, яку надає базовий елемент блокової пам'яті, використаний в реалізації; максимальна кількість елементів вектора визначається розміром одного елемента і обсягом елементів пам'яті, використаних в реалізації. Може бути обчислено за формулою $V = w \times k$, де w – розмір елемента в бітах, k – кількість елементів, V – обсяг займаної пам'яті.

Для успішної реалізації моделі контейнера “вектор” була розроблена додаткова логіка, яка реалізує необхідні операції для елементів пам'яті.

Контейнер «вектор» надає повний набір конструкторів, деструкторів і операцій копіювання. Для реалізації сукупності операцій використовується дешифратор команд, який переводить з двійкового коду інструкції в позиційний код, які активує сигнали контейнера. Логічна модель має наступні параметри: DATA_WIDTH та визначає ширину шини даних та залежить від типу данихЮ з яких створений контейнер «вектор». Параметр вибирається залежно від максимальної кількості доступних команд, число яких дорівнює $2^{\text{CMD_WIDHT}}$ ; SIZE визначає ширину шини адреси. Цей параметр впливає на максимальну кількість адресованих комірок контейнера і дорівнює $2^{\text{SIZE}}$.

Логічна модель контейнера «вектор» має інтерфейсні сигнали: глобальний синхроімпульс clk; скидання reset, при високому рівні якого відбувається скидання елементів пам'яті в нулі, що відповідає очищення контейнера; шина адреси index, розмірністю [SIZE-1: 0], яка використовується для вказівки порядкового номера елемента у векторі під час запису або читанні; шина даних data_in, розмірністю [DATA_WIDTH-1: 0], яка використовується для запису даних в вектор; шина команд command, розмірністю

[CMD_WIDTH-1: 0], яка використовується для управління контейнером; шина даних data_out, розмірністю [DATA_WIDTH-1: 0], яка використовується для читання даних з вектора; сигнал empty, високий рівень якого сигналізує про те, що вектор порожній.

Структурна модель контейнера «вектор» містить дешифратор команд dc і перетворює двійковий код в керуючі сигнали.

Блок pointer містить два регістри: адреса останнього елемента (для порожнього вектора він дорівнює 1) і адреса першого вільного елемента (для порожнього вектора дорівнює нулю). На вході блоку є три керуючих сигнали: reset, inc, dec. При високому рівні сигналу inc і передньому фронті синхроімпульса внутрішні регістри збільшуються на одиницю. При високому рівні сигналу dec і передньому фронті синхроімпульса внутрішні регістри зменшуються на одиницю.

Мультиплексор індексу управляється шиною з двох сигналів. У режимі читання / запису осередку за індексом сигнали повинні бути встановлені в 00, в режимі push_back () сигнали повинні бути встановлені в 01, в режимі back () сигнали повинні бути встановлені в 10, в режимі front () сигнали повинні бути встановлені в 11.

Для аналізу і проектування контейнера «список» розглянемо типові операції, що виконуються контейнером «вектор». Тут X – це умова, що збуджує ту чи іншу операцію, команди і опис команд: x1_begin () – посилання на перший елемент списку; x2_end () – посилання на останній елемент списку; x_bool empty () – повертає true, якщо контактів немає; x3_push_front () – додає новий елемент в початок; x4_push_back () – додає новий елемент в кінець; x5_pop_front () – видаляє елемент з початку списку; x6_pop_back () – видаляє елемент з кінця списку; x7_insert () – додає елемент перед поточним; x8_erase() – видаляє поточний елемент; x9_next () – посилання на наступний елемент; x10_prev () – посилання на попередній елемент.

Для успішної реалізації моделі контейнера «список» була розроблена додаткова логіка, яка реалізує необхідні операції.

Для верифікації отриманих моделей були використані тести, що перевіряють 100% режимів роботи пристрою. Вихідні еталони були отримані з програми на C++.

Тут розглядаються тільки функціональні тести – перевірка правильності функціонування моделі, і навантажувальні тести – виміри продуктивності і часових характеристик моделі. На рис. 5.5 показана схема процесу верифікації логічної моделі.

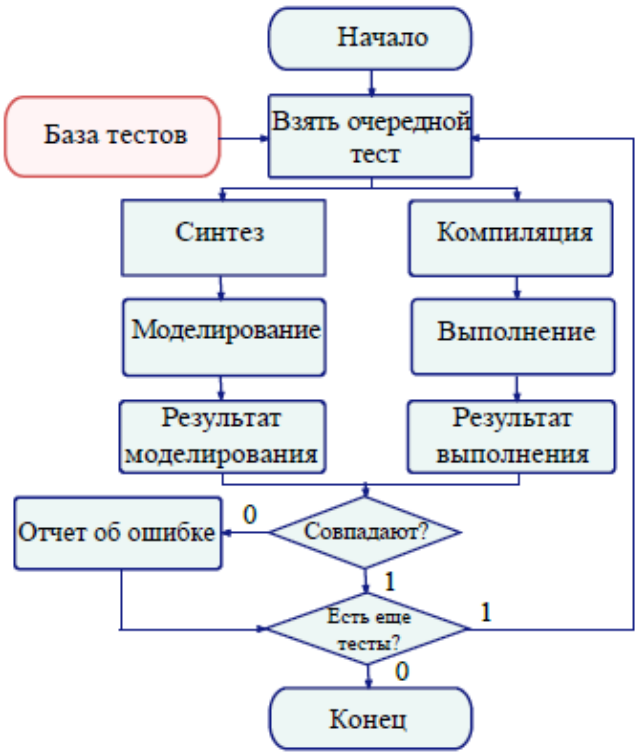



Рисунок 5.5 – Схема верифікації логічної моделі

Створюється набір тестів на C++, в яких використовуються контейнер вектор. У цих тестах присутні виклики всіх методів, які підтримує контейнер. На наступному етапі тести компілюються звичайним C++ компілятором для отримання виконуваного файлу. При виконанні цієї програми в файл зберігаються результати роботи контейнера.

З іншого боку, відбувається синтез вихідних текстів. В результаті синтезу виходить модель системного рівня, що містить контейнер «вектор», опи-

саний на мові опису апартури (МОА) VHDL. Далі здійснюється моделювання за допомогою програми Aldec Active-HDL для отримання результатів. Вони порівнюються з результатами, отриманими внаслідок роботи звичайної програми. Логічна модель складена вірно, якщо результати моделювання і виконання збігаються.

Наступним етапом верифікації є перевірка відповідності функціональної моделі і моделі з часовими затримками. В функціональній моделі відсутня інформація про час поширення сигналу від входів елементів до виходів. Мається на увазі, що сигнал поширюється миттєво або за один дельта-цикл моделювання. Програма логічного синтезу, трасування і розміщення генерує спеціальний файл затримок поширення сигналу для обраної архітектури. Цей файл може бути завантажений в САПР Aldec Active-HDL для побудови часової моделі.

Останнім етапом тестування є перевірка моделі, реалізованої в тому чи іншому кристалі. Архітектура ПЛІС Xlinix обирається як дешева і високопродуктивна мікросхема з можливістю багаторазового перепрограмування.

## 5.4 Програмно-апаратна реалізація моделей

Архітектури для System-C-driven Design Computing.

Представлений метод мультиверсного синтезу керуючих і операційних автоматів в заданій інфраструктурі проектування, орієнтованих на архітектурні рішення в метриці, що мінімізує час виконання функціональності за рахунок розпаралелювання операцій при обмеженні на апаратні витрати.

За своєю суттю програма синтезу C ++ в VHDL є машиною по трансформації вихідного опису в результуючий. Таких трансформацій відбувається кілька. Вхідна модель представлена на мові C ++. Це алгоритмічний опис, який вирішує поставлене перед інженером завдання.

Виконано тестування і верифікація програмних модулів інфраструктури проектування цифрових систем на кристалах, а також визначення ефек-

тивності запропонованих моделей, методів і структур даних при створенні реальних компонентів цифрових виробів.

Тестування системи складається з декількох етапів: розробка схеми тестування компілятора; підготовка тестів; підготовка еталонів; виконання тестів і аналіз результатів. В роботі [160] наведено опис етапів перетворення.

Проведено порівняння продуктивності розробленої апаратної моделі і програмної реалізації на процесорі експериментальним шляхом. Дані отримані в результаті роботи 10 інженерів над одним проектом. Оцінювалися такі характеристики: часові витрати на проектування, швидкодія, енергоспоживання, площа на кристалі. В результаті дослідження отримали при адитивній оцінці автоматичний метод краще в 1,301292012 раз (рис. 5.6).




Рисунок 5.6 – Адитивна оцінка ефективності

На рис. 5.7 наведено архітектуру дисертаційного дослідження, яка містить компоненти, відображені у вигляді моделей, методів і алгоритмів:

1) визначення специфікації системи;

2) парсерінг опису з мови SystemC у мову VHDL;

3) паралельний опис проекту мовами VHDL, Verilog;

4) диверсифікація проектування цифрової системи на кристалі;

5) тестування та верифікація HDL-коду;

6) розміщення, трасування та виробництво SoC;

7) бібліотеки IP-cores для паралельного та послідовностного проектування;

8) імплементація архітектури та методів у хмарний сервіс.

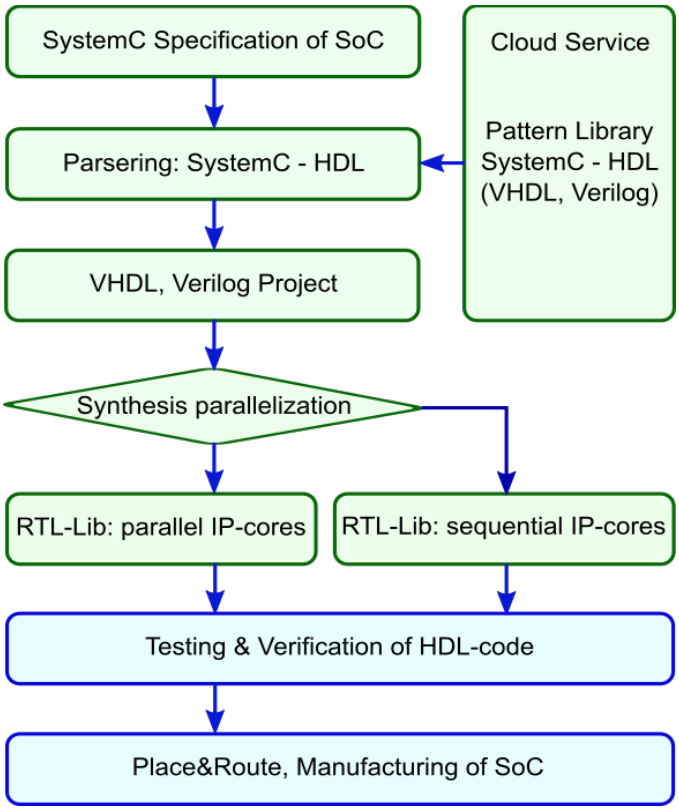



Рисунок 5.7 – Архітектура проектування SoC

Архітектури для design and test computing повинні бути орієнтовані на використання стандартів IEEE 11.49, 1500, 1687, які підтримують внутрішні формати структур даних, специфікації існуючих мов опису апаратури, методів синтезу, тестування, верифікації, аналізу, діагностування, виробництва та відновлення працездатності в режимі реального часу при відмові компонентів. Крім того, необхідно орієнтувати результати наукових досліджень на технології SoC design and test від провідних компаній планети: Synopsys, Cadence, Mentor, Aldec, ARM, які задають комп'ютингову моду і показують вченим актуальні невирішені проблеми.

Домінуючою архітектурою для TLM Design є замкнута в цикл система, заснована на використанні бібліотек нових та існуючих рішень, що створю-

ють функціонально і timing-коректні рішення (рис. 5.8). В цьому випадку можна використовувати технологію, яка укладається в чотири рівні верифікації проекту цифрового виробу, мета якої – усунути всі можливі помилки на ранніх стадіях синтезу (рис. 5.9). Більш укрупнено схема процесу верифікації може бути представлена графом, який створює причинно-наслідковий зв'язок між стадіями проходження проекту, а також визначає основні відмінності між верифікацією і валідацією (рис. 5.10).

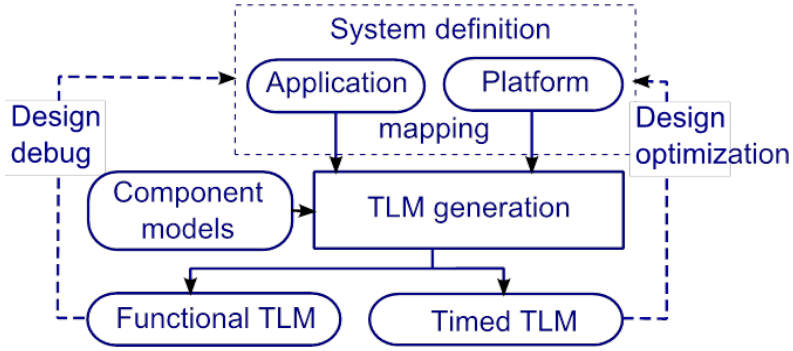



Рисунок 5.8 – Архітектура TLM Design

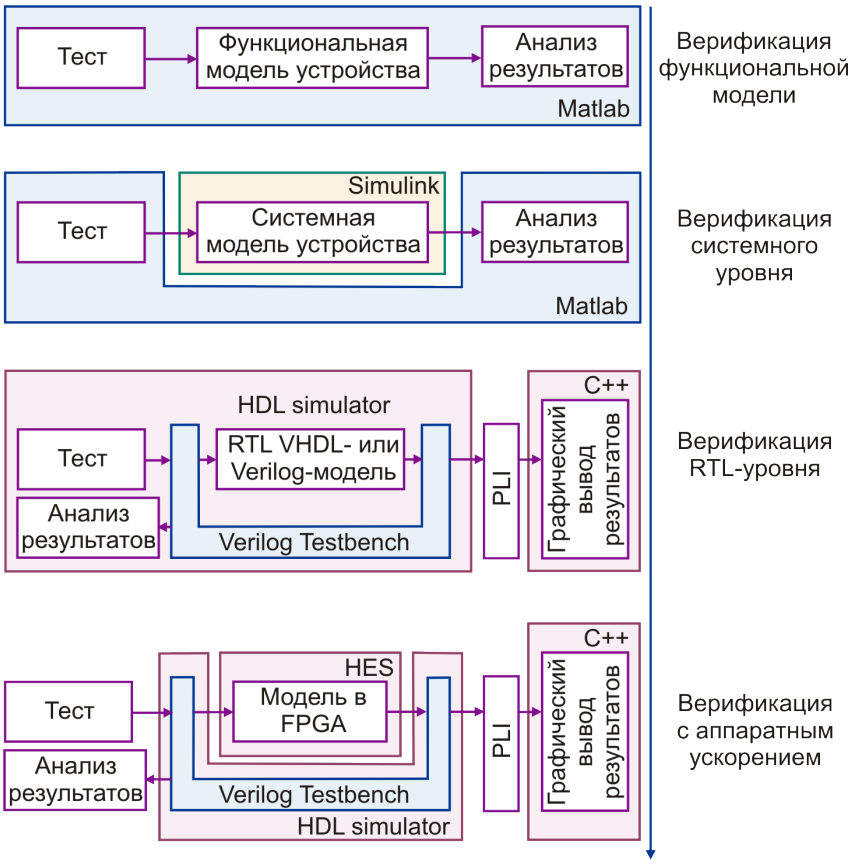



Рисунок 5.9 – Рівні верифікації TLM Design

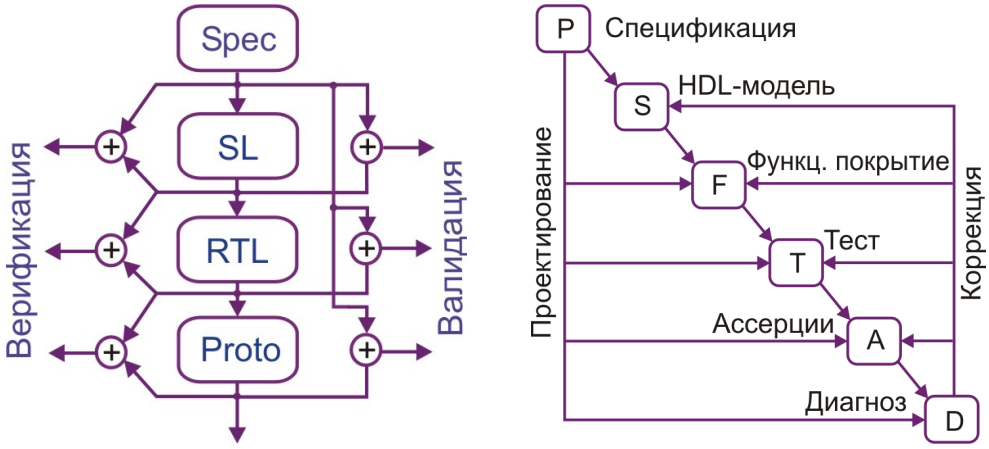



Рисунок 5.10 – Графи валідації та верифікації TLM Design

Слід зауважити, що технології проектування стають все більш симетричними щодо імплементації в апаратні або програмні рішення (рис. 5.11). Питання полягає лише в тому, який продукт бажає отримати замовник щодо продуктивності, ремонтопридатності і якості виробу. Однак завжди враховується параметр time-to-market, який є домінуючим при визначенні економічної стратегії Design Computing (рис. 5.12).

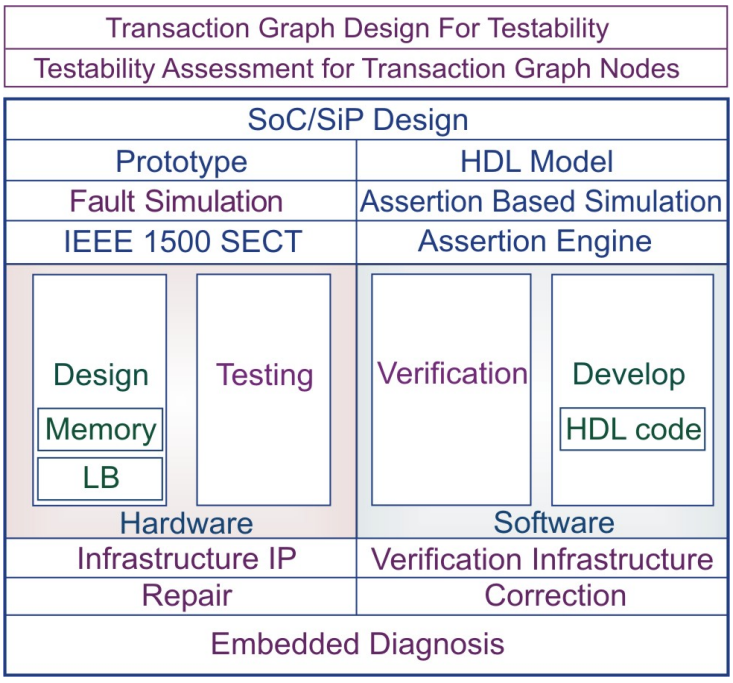



Рисунок 5.11 – Архітектура для HW / SW TLM Design

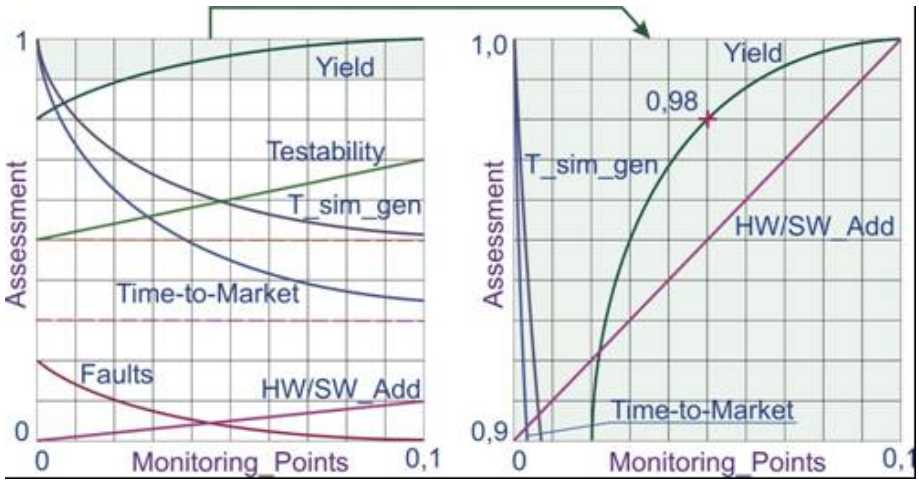



Рисунок 5.12 – Time-to-market для HW / SW TLM Design

Згідно з існуючими стандартами будь-яка Computing Design архітектура повинна містити компілятори з мов опису апаратури, включаючи System C, System Verilog, C++ (рис. 5.13). Це дає можливість диверсифікувати процес створення цифрової системи, а також підвищити якість процесу проектування за рахунок паралельного аналізу і синтезу декількох прототипів і подальшого метричного вибору кращого з них (рис. 5.14).

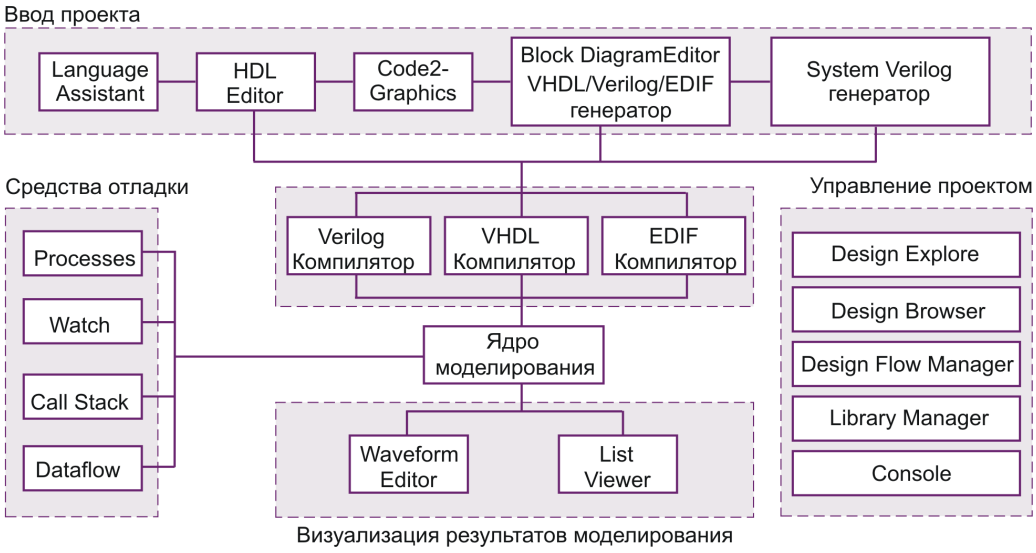



Рисунок 5.13 – Архітектура компілятора для HW / SW TLM Design

$$Q = Q(\overline{\vee((Q \wedge Q_i) \oplus Q)}) \vee Q_i(\vee((Q \wedge Q_i) \oplus Q));$$
$$Y = \vee((Q \wedge Q_i) \oplus Q);$$
$$Q = Q\overline{Y} \vee Q_i Y.$$

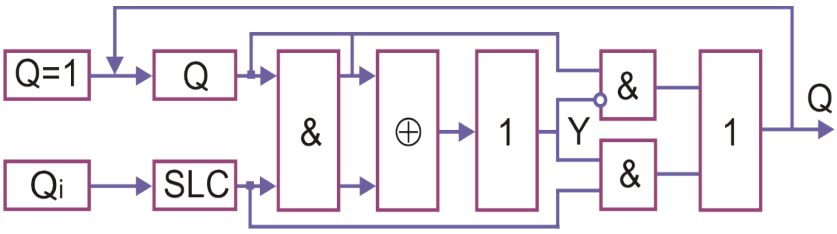



Рисунок 5.14 – Секвенсор для вибору рішення HW / SW TLM Design

Економіка технологій проектування сьогодні змушує всіх вендорів відмовлятися від пакетів дискет або флеш-пам'яті і приходити тільки до рішень хмарного комп'ютингу, який стає потенційно доступним всім жителям планети (рис. 5.15). Крім того, створення нового мікросервіса комп'ютінгового проектування забезпечує його просування в галузі SoC design and test, включаючи покупку даного сервісу, якщо він цікавий провідними компаніям.

Таким чином, хмарний сервіс-комп'ютинг (рис. 5.16) для вирішення завдань синтезу та аналізу, тестування і верифікації, як online-доступна і економічно вигідна технологія є найбільш конкурентоспроможною кібер-укладом, який буде домінувати в світі в найближчі 10 років.

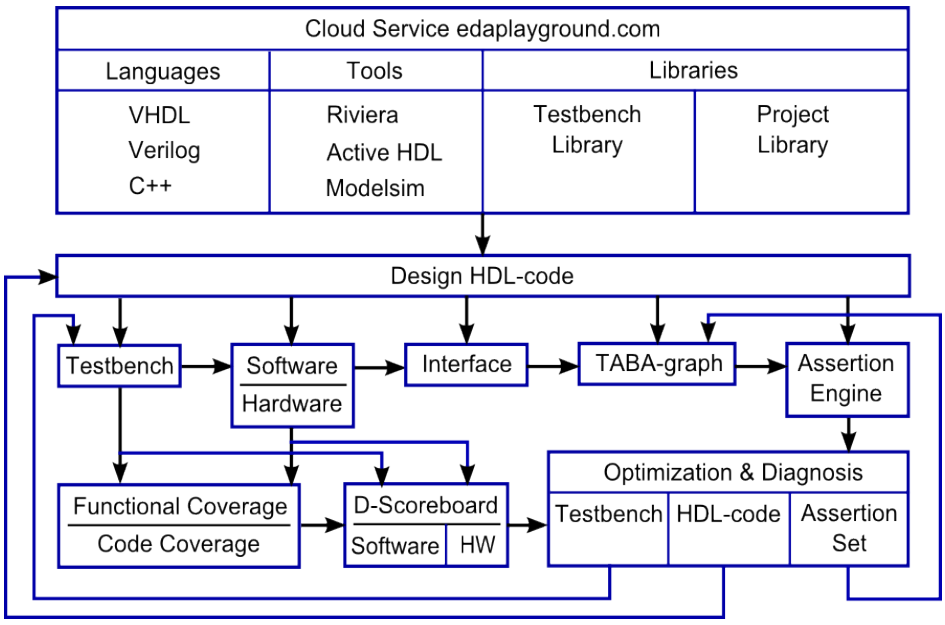



Рисунок 5.15 – Cloud-driven HW / SW TLM Design

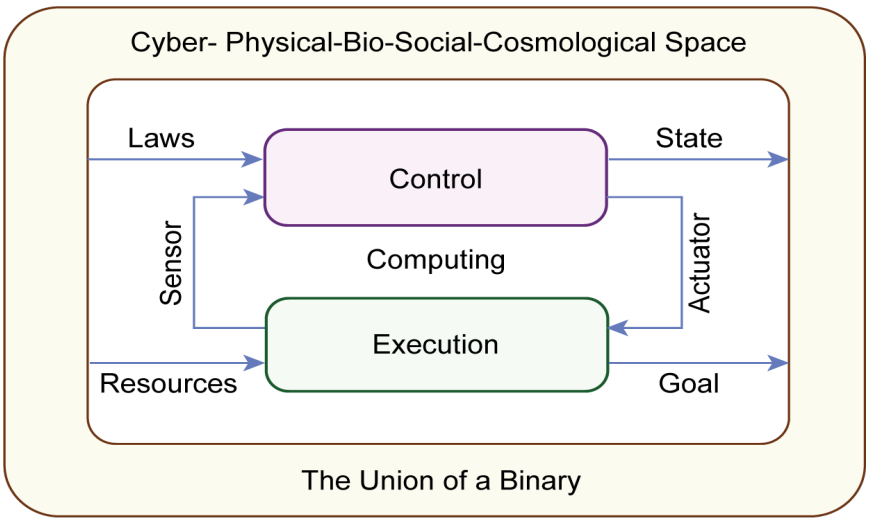



Рисунок 5.16 – Cloud-driven Computing for HW / SW TLM Design

Завдання вчених в нових незалежних державах – опанувати існуючу кіберкультуру SoC design and test від провідних компаній планети з метою подальшої пропозиції інноваційних технологічних рішень в межах існуючих стандартів за описом проектів засобами мов, включаючи System C, System Verilog. Все, що стосується хмарно-програмної імплементації інноваційних ідей, моделей, методів і алгоритмів, то ця область є доступною для кожного вченого України.

## 5.5 Висновки до розділу 5

Запропоновано мультиверсний паралельний синтез цифрових структур моніторингу та управління трафиком на основі SystemC специфікації, що використовується при створенні модуля Sherlock, який призначений для точного моніторингу даних щодо позиціонування та функціонування автомобиля в процесі виконання заданого маршруту.

Мультиверсний синтез застосовується при створенні программних продуктів хмарного сервісу Cloud Traffic Control, а також при розв'язанні задач тестування та діагностування програмно-апаратних компонентів кіберфізичної системи.

Результити можна використовувати для мультиверсної системи захисту програмно-технічних компонентів хмарного сервісу управління транспортом у межах технічного та організаційного кібер-шару проекту для подальшого виконання заходів з інтегрування реальної керуючої інфраструктури доріг та автомобілів з віртуальним кіберпростором; тестування, верифікації та діагностування програмно-апаратних засобів хмарного керування трафіком на стадіях проектування та експлуатації.

Розроблені програмно-апаратні реалізації моделей, методів і структур даних для проектування цифрових систем на кристалах, які включають процедури створення специфікації, синтезу, тестування, моделювання та верифікації на основі запропонованої інфраструктури, що містять промислові засоби компаній Aldec і Xilinx. Розглянуті питання тестування розроблених програмних продуктів на реальних цифрових проектах створення IP-Core як примітивів для реалізації цифрових систем на кристалах.

Вирішені наступні завдання:

1. Розроблено і описано метод мультиверсного синтезу керуючих і операційних автоматів в заданій інфраструктурі проектування, орієнтованих на архітектурні рішення в метриці, що мінімізує час виконання функціональності за рахунок розпаралелювання операцій при обмеженні на апаратні витрати.

2. Виконана програмна реалізація моделей і методів мультиверсної розробки операційних пристроїв в рамках інтегрованої системи проектування функціональних і архітектурних рішень SoC на основі використання продуктів верифікації та синтезу компаній Aldec і Xilinx.

3. Здійснено тестування і верифікація програмних модулів інфрастурктури проектування цифрових систем на кристалах, а також визначення ефективності запропонованих моделей, методів і структур даних при створенні реальних компонентів цифрових виробів.

Сутність ринково-орієнтованого науково-технічного підходу визначена як мультиверсне проектування архітектури цифрового виробу на основі

заданої специфікації в середовищі SystemC (C++) і автоматичному виборі синтезованих функціональних структур з метою істотного зменшення часу створення проекту і підвищенні виходу придатної продукції за рахунок паралельного синтезу і верифікації архітектурних рішень системного рівня відповідно до запропонованої метрикою. Основна інноваційна ідея полягає в паралельному автоматичному синтезі квазіоптимальної архітектури відповідно до запропонованої специфікацією і метрикою на основі підбору синтезованих функціональних структур.

# 6 КОМП’ЮТИНГОВІ МОДЕЛІ ХМАРНИХ СЕРВІСІВ

З метою розробки хмарних сервісів керування трафіком для автотранспортних компаній та водія пропонується кіберкультура мікро-макро-космо-комп'ютингу, яка формулює, пояснює і прогнозує сучасні технології моніторингу та управління процесами і явищами в фізичному, віртуальному та космологічному просторі. Представляються вербальні та структурні визначення основних типів комп'ютингу, що базуються на сучасних трендах еволюційного розвитку кіберекосистеми планети. Формулюється універсальна модель МАТ-комп'ютингу: <Memory, Address, Transactions>, яка використовує три компоненти для створення обчислювальної структури в технологічно прийнятному матеріальному середовищі. Показується інформаційно-квантовий напрям експансії людини в космічний простір, а також можливість аналогічного проникнення неземних біотехнічних об'єктів в екосистему нашої планети. Пропонується модель комп'ютингу, яка описує квазіоптимальні структури моніторингу та управління масштабованими процесами різної природи: технічними, біологічними, соціальними, віртуальними і космологічними.

## 6.1 Вступ

Розглядається автоматна модель сталого розвитку людства, задана в метриці зеленої концепції збереження планети і підвищення якості життя людства [161]. Для формування моделі слід визначити трендові аксіоми (The 10 Commitments of Greening), яким необхідно слідувати для сталого розвитку людства:

1) використання автоматної моделі комп'ютингу для моніторингу та управління всіма процесами та явищами, як найнадійнішу і просту для розуміння і виконання. Модель має мінімальну кількість компонентів і сигналів при реалізації замкнутої детермінованої комп'ютингової системи для досягнення поставленої мети. Життєвість автоматної моделі підтверджується не

тільки створенням індустрії комп'ютерів, але й адекватним комп'ютинговим описом процесів і явищ у фізиці, біології, медицині, техніці, соціології, космології;

2) механізми виконання і управління при реалізації моделі комп'ютингу не повинні перетинатися за складом компонентів. Умова регламентує суворий поділ функціональних обов'язків між механізмами управління і виконання, що також означає невтручання одного механізму в справи іншого. Порушення даного пункту є типовою системною помилкою, що породжує корупцію, і властивою управлінню соціальними групами;

3) застосування концепції Бога – все бачить і віддає належне – для моніторингу та управління індивідуумом, соціальними групами і людством в рамках створення хмарних комп'ютингових сервісів. Стійко виключати людину, як ненадійнішу ланку, з моніторингу та управління кіберфізичними, біологічними, технічними, технологічними і соціальними процесами і явищами. Умови дозволяють оптимально управляти кожною людиною, соціальними групами і державами, що виключає всі негативні наслідки від втручання людини, у тому числі війни, соціальні конфлікти, корупцію і несправедливість. Економічний ефект від реалізації концепції вимірюється десятками трильйонів доларів;

4) автоматний детермінізм і передбачуваність реакції комп'ютингу на ініціювальні або актюаторні впливи. Виключення ймовірності з подій і процесів, як «фігового листка на голому тілі нашого невігластва», за визначенням Ейнштейна. Умови орієнтовані на детермінізм комп'ютингових сервісів, що максимально виключає непередбачуваність імовірнісних методів і суттєво зменшує накладні витрати на дублювання та резервування системних компонентів і процесів;

5) тотальне оцифровування всіх просторових, кіберфізичних, біологічних, соціальних процесів, явищ і транспорту, що забезпечують точне управління на основі цифрового моніторингу та позиціонування. Умови дозволяють в глобальному масштабі вирішити проблему надійного безаварійно-

го, безпілотного управління всіма видами транспорту, зменшити витрати на виготовлення автомобільних номерів (5 млрд доларів), інфраструктуру дорожніх знаків і світлофорів (500 млрд доларів) за рахунок створення віртуальної е-інфраструктури управління транспортом. В даний час на дорогах всіх країн гине в рік 1,2 мільйона чоловік. Цифровий моніторинг природних явищ, ураганів, тайфунів, землетрусів, цунамі, потепління на основі впровадження мережі сенсорів забезпечить актюаторне управління кліматичними і геопатогенними катастрофами. Впровадження технології SlingShot в якості спеціалізованого вирішення проблеми чистої води, від нестачі якої щорічно помирає 2 мільйони жителів планети, шляхом точного моніторингу забруднюючих компонентів і їх подальшого усунення адекватними актюаторнимі впливами;

6) метричне оцінювання всіх процесів і явищ, які формують інтегральний критерій «час-гроші-якість» з метою морально-матеріального стимулювання соціально та екологічно значущих проектів. Метричне оцінювання і подальше адекватне морально-матеріальне стимулювання членів соціальних груп і колективів, що роблять істотний вплив на позитивні процеси підвищення якості життя людей і екосистеми планети. Реалізація глобального ринкового тренду e-infrastructure від програми EU Horizon 2020 стосовно всіх земних процесів і явищ, у тому числі: health care e-infrastructure, smart home e-infrastructure, smart university e-infrastructure, smart city e-infrastructure, social management and government e-infrastructure, traffic control e-infrastructure, internet driven e-infrastructure for diagnosis and repair of technics, computers;

7) повне виключення вторинних ознак ідентифікації людини в кіберфізичному просторі на основі повсюдного впровадження е-інтерфейсів для введення первинних аутентифікаторів (відбитки пальців, райдужна оболонка ока, ДНК). Умови дозволяють прибрати з обігу мільярди паперових паспортів і електронних карток, які можна підробити, втратити, що дає можливість зберегти ліси і заощадити до 10 мільярдів доларів на виготовлення документів;

8) створення глобальної е-інфраструктури надійного, захищеного доступу до кіберфізичного простору планети і розробка відповідного законодавства, що забезпечують легітимне online виконання функціональних обов'язків, не прив'язаних до робочого місця. Умови дозволяють в масштабах планети на 20 відсотків зменшити транспортний трафік, споживання бензину і шкідливих викидів завдяки виконанню функціональних обов'язків працівниками за місцем проживання;

9) впровадження електронного громадянства для всіх жителів планети і виключення паперових носіїв інформації з усіх сфер людської діяльності. Умови дозволяють зберегти не менше 20 % лісу від вирубки для виготовлення паперу, що підвищить вміст кисню на планеті і поліпшить її екологію. Електронне громадянство скоротить міграцію громадян і зробить все країни привабливими завдяки реальній небезпеці (для керівників відсталих держав) електронного переміщення всіх громадян разом з їх податками;

10) хмарний і big data комп'ютинг на основі планетарної мережі data центрів дозволяє вирішити проблему захисту інформації, сервісів, персональних даних, прибрати мільйони настільних комп'ютерів, локальних серверів і перейти до використання економічних енергозберігаючих планшетів доступу до сервісів та персональним кабінетів. Brain-Computer інтерфейси дають можливість усунути численні пристрої введення даних і перейти до безпосереднього образно-імпульсного спілкування людського мозку з комп'ютинговими терміналами. Сканування мозку на основі зовнішніх або вбудованих сенсорів дозволяє запобігати злочиннам або нелегітимним діям людини, що істотно вплине на ефективність роботи поліції і спеціальних служб.

Виконання перерахованих аксіом дозволить перемістити всі компоненти механізму управління земними процесами і явищами в хмари комп'ютингових сервісів, що звільнить істотну частину (20%) людства від виконання невластивих йому управлінських непродуктивних процесів. Передати управ-

ління хмарному комп'ютингу означає істотне зниження накладних витрат і шлях до створення зеленої планети (якість життя + чиста екологія).

Вартість реалізації перерахованих аксіом порядку 50 мільярдів доларів, економія від їх впровадження – не менше 50 трильйонів доларів, плюс якість життя кожної людини і реінкорнація екології зеленої планети.

Всі технічні проблеми для реалізації десяти трендів вже вирішені в тій чи іншій мірі. Головною перешкодою на шляху створення зеленої планети і щасливого життя є невисокий рівень кіберкультури людини. Тому проблема досяжності поставлених зелених цілей жорстко пов'язана з вихованням довіри у людей до надійності, справедливості і непідкупності комп'ютингових сервісів моніторингу та управління.

Сталий розвиток зеленого комп'ютингу представлено в метриці простору і часу трьома історичними періодами або фазами, наведеними на рис. 6.1:

1) відображення (моніторинг) фізичних процесів і явищ, представлений сингулярним (Single Computing), мережевим (Network Computing) і глобальним комп'ютингом (Global Computing - Internet). Тут також фігурують поняття Desktop, Servers, Data Base;

2) управління фізичними процесами і явищами на основі e-infrastructure, цифрового моніторингу, представлене Cloud Computing, Cyber Physical Networks, Internet of Things. Цикл теперішнього часу, де основними учасниками є Gadget, Laptop, Data Centers, Big Data;

3) створення інтелектуальних кіберфізичних процесів і явищ під керуванням кібермозку планети, представлене Brain Computing, Robotic Networks, Internet of Nature (World). Очікується поява Massive Quantum-Atomic Computing, Brain-Computer Interface, Atomic Data Center Networks, Smart Big Data Networks.

На рис. 6.1 також представлені вектори розвитку зеленого комп'ютингу, які оформлені в наступні фазові трансформації за трьома історичними періодами: 1) Single Computing – Cloud Computing – Brain Computing. 2) Network Computing – Cyberphysical Computing – Network Robotic Computing. 3)

Internet Computing – Internet of Things – Internet of Nature. 4) Data based Computing – Big Data Computing – Smart Data Computing.




Рисунок 6.1 – Комп'ютинг у сталому розвитку

Космологічна модель Всесвіту представлена жорстко пов'язаними між собою взаємодіючими компонентами: Простір і Час, Матерія і Енергія. Наведену структуру базових понять можна застосувати і до процесу сталого розвитку комп'ютингу. Для цього необхідно виділити такі компоненти: 1) маса матерії (m) для реалізації примітиву-транзистора або комп'ютера; 2) енергія (E) для виконання елементарної операції або функціонування комп'ютера; 3) швидкодія (t), як величина, зворотна часу (t=1/T) виконання однієї елементарної операції або продуктивність комп'ютера (t=I/T, кількість інструкцій за секунду); 4) простір (S) (кіберфізичний), що обслуговується комп'ютингом, шляхом надання користувачам сервісів моніторингу, управління, творення.

Використовуючи введені компоненти в якості аргументів, можна визначити на історично незначному проміжку часу (1970-2020) два досить парадоксальних і дуже оптимістично зелених закони розвитку комп'ютингу (рис. 6.2): 1) зв'язок між енергією і часом; підвищення швидкодії комп'ютера пов'язане зі зменшенням енергоспоживання; 2) зв'язок між матерією і простором; зменшення маси комп'ютера пов'язане з розширенням обслуговуваного простору; обидва закони вірні також у зворотному прочитанні.

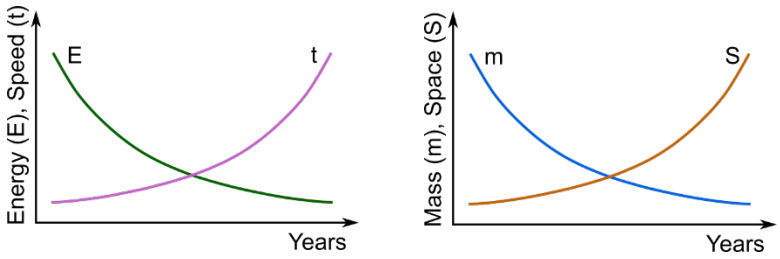



Рисунок 6.2 – Взаємодія компонентів комп'ютингу

Якщо виконати суперпозицію двох законів, то можна отримати інтегральний закон комп'ютингу: збільшення швидкодії і простору комп'ютингу жорстко пов'язане зі зменшенням його енергоспоживання і маси. Якщо умовно визначити інтервал зміни відносних значень всіх чотирьох параметрів між 0 і 1, то закон можна записати в наступному вигляді: t+E=1, S+m=1. Це означає, що адитивна оцінка швидкодії (простору) і енергоспоживання (маси) комп'ютингу на піввіковому відрізку часу є величиною постійною.

Найбільш значущі ринково-орієнтовані відкриття та інновації вчені роблять шляхом суперпозиції міждисциплінарних досліджень, а також імплементації досягнень однієї технологічної культури в суміжні галузі знань. Моделі комп'ютингу знаходять все більше застосування для моніторингу та управління процесами і явищами в усіх областях діяльності людини і природи [162-165]. Інтеграція природної і біосоціальної культури з кібертехнологічними рішеннями моніторингу та управління приводить до оригінальних

системних (кіберфізичних, біоінформаційних) наукових результатів та інновацій в традиційно консервативних галузях знань, таких як: природа, біологія, соціологія, екологія, техніка, транспорт і промисловість. Підтвердженням сказаного можуть служити модні глобальні технології, що використовують масштабовані моделі комп'ютингу [163, 164]: 1) Cyber-Physical Systems; 2) Internet of Things and Everything; 3) Web- Cloud-, Mobile-, Service-, Network-, Automotive, Big Data and Quantum computing; 4) Internet-Driven Smart Infrastructures: Enterprise, University, City and Government. Так, наприклад, використання автоматної моделі комп'ютингу для опису процесів мозку привело до точного моніторингу порушень в ньому і подальшого ефективного відновлення функциональностей шляхом застосування актюаторних впливів [166]. Вчені роблять успішні спроби в розробці комп'ютерних структур на основі використання природних рішень, які ототожнюються з мозком живих істот [167], що неодмінно приведе до створення кібермозку людства.

Поняття комп'ютингу розвивається з класичної автоматної моделі обчислювача, що об'єднує механізми управління і виконання, сигнали моніторингу та актюації, входи для введення інструкцій і даних, а також виходи стану системи і результатів. Комп'ютинг – процес досягнення поставленої мети шляхом використання механізмів управління та виконання в циклічно замкнутій системі з заданими входами і виходами, сигналами моніторингу та актюації. У вузькому сенсі, комп'ютинг – це дії, цілеспрямовані на дослідження, проектування і застосування інтелектуальних програмно-апаратних систем і мереж для моніторингу та управління кіберфізичними процесами і явищами. Область комп'ютингу покриває: нано-, мікро- і макроелектроніку, радіотелекомунікації, комп'ютерну, програмну, системну, виробничу, транспортну та соціальну інженерію, штучний інтелект і кіберуправління, комп'ютерні науки та інформаційні технології.

Слід нагадати, що інформаційні технології – це процеси, методи і способи пошуку, збору, зберігання, обробки і поширення інформації шляхом використання засобів обчислювальної техніки [168, 169]. З урахуванням наве-

дених вище визначень можна зробити висновок, що комп'ютинг поглинає досить застаріле, але, в силу інерції людського мислення, поширене поняття «інформаційні технології», яке формально ототожнюється з відображенням (моніторингом) фізичних процесів і явищ у віртуальному кіберпросторі.

Істотним видається відмінність комп'ютингу від інформаційних технологій, пов'язана з активним управлінням процесами і явищами в реальному і віртуальному світах. Можна поставити знак відповідності між поняттями змісту і форми: Internet = Information Technology і Internet of Things = Computing. Розвиток комп'ютингу має історично виражені фази (рис. 6.1): 1) сингулярний комп'ютинг; 2) мережевий комп'ютинг; 3) глобальний комп'ютинг; 4) кіберфізичний комп'ютинг; 5) сервіс-комп'ютинг – початок нового циклу. Далі передбачається поява кіберлюдського комп'ютингу, коли human brain буде безпосередньо інтегрований у кіберпростір. Розвиток комп'ютингу, основна функція якого оптимальне і надійне управління всіма процесами та явищами на основі точного цифрового моніторингу без прямої участі людини, слід розглядати тільки у взаємодії двох світів, реального і віртуального: 1) людина завжди погано керує реальним світом і створює собі в допомогу комп'ютинг; 2) як більш досконалий механізм, комп'ютинг забирає управління технологічними процесами у людей – фаза теперішнього часу; 3) щоб врятувати людство від самоліквідації, комп'ютинг, найближчим часом, повинен забрати і решту управління соціальними процесами під свою юрисдикцію; 4) людина і комп'ютинг об'єднуються в бажанні творчої зміни кіберфізичного континууму в цілях мирного співіснування і взаємного проникнення для створення нового поняття кіберлюдини (Cyber Human – CyMan). Кіберлюдина – персона, яка володіє глобальною технологічною кіберкультурою, безпосередньо (Brain-Computing-Interface) підключена до кіберфізичного простору планети в цілях виконання кіберсоціальної ролі. Комп'ютинг сьогодні перетворюється з ненадійної локальної конструкції в незнищенну глобальну субстанцію. Людина поступово буде трансформуватися з уразливого біологічного суб'єкта в кібербіологічну форму, а далі – в

кіберенергетичну, «вічно живу» інформаційно і фізично реінкорновану субстанцію.

В результаті взаємодії людини з кіберфізичним світом формується нове поняття – кіберкультура, як рівень розвитку соціально-технологічних відносин між суспільством, фізичним світом і кіберпростором, який визначається впровадженням інтернет-сервісів точного цифрового моніторингу та надійного метричного управління в усі процеси і сфери людської діяльності, у тому числі освіту, науку, медицину, виробництво і транспорт, в цілях підвищення якості життя людей і збереження екосистеми планети.

Bernard Marr (Forbes) запропонував 9 аттракторів [170], що підтверджують інтерес ринку і світового бізнесу до комп'ютингу: Big Data, Internet of Things, Mobile to computing everywhere, Cyber security, E-Assistants or Brain-Computer Interface, Social Networks, Gamification for Bisiness and Education, Cloud computing, Video communications. У найближчому майбутньому очікується масове впровадження в життя людей нових технологій, які формують Virical Continuum: Internet of Everything, wearable's, smart car, smart home, smart city, 3D printing, quantum computing, robotics, the cloud, Big Data, the maker movement, drones. Особливу увагу автор приділяє рекламі IoT. Тому, що 87 відсотків людей не чули, що це таке? Фактично – це взаємодія віртуального і реального світу: 2015 рік – 5 мільярдів з'єднаних пристроїв, 1,4 мільярда смартфонів; 2020 рік – 50 мільярдів і 6,1 зазначених виробів відповідно. Це також 250 мільйонів машин у 2020 році, які будуть керуватися без водіїв за допомогою хмарних кіберсервісів дорожнього руху. Вже сьогодні Google-машини проїжджають близько 10 000 миль на тиждень по інноваційним міським дорожнім інфраструктурам. Більше, ніж 10,2 мільйони «розумного» одягу буде вироблено до 2020 року. Також істотно зросте ринок RFID-міток для цифрової ідентифікації об'єктів і процесів з 11,1 до 21,9 мільярда доларів. У підсумку, фінансовий вплив IoT в 2025 році на світовий ринок складе суму в 11 трильйонів доларів, а рівень IoT-капіталізації – 4,6 і 14,4 трильйони в публічному і приватному секторі відповідно.

Susan Galer (Forbes) вважає, що хмарний сервіс буде найбільш цікавою моделлю IT-бізнесу в найближчі 10 років [171]. До кінця 2017 року дві третини від 2000 глобальних компаній трансформують свою діяльність під цифрові процеси моніторингу та управління. Понад 50 відсотків усіх інвестицій компьютингових (IT-) компаній будуть спрямовані на створення технологічних платформ і сервісів, пов'язаних з cloud, mobile, social business and big-data analytics. До 2020 року cloud-based капіталізація IT-компаній досягне 70 відсотків від всіх програмно-технологічних сервісів. У 2018 року будуть інстальовані 22 мільярди internet of things devices, які матимуть доступ до більш, ніж 200 000 нових internet of things додатків і сервісів. До 2018 року більше 50% компаній-розробників будуть додавати до своїх додатків пізнавальні сервіси (проти 1 відсотка сьогодні), забезпечуючи тільки USA-підприємствам понад 60 мільярдів доларів щорічної економії. До цього часу 50 відсотків усіх підприємств створять хмарні платформи для поширення власних інновацій і споживання зовнішніх пропозицій. До 2018 року 80% компаній в форматі B2B (Busines-to-Business) і 60% B2C (Busines-to-Customer) організацій перебудують свої входи (двері) на цифрові, що забезпечить збільшення споживачів і замовників на 3-4 порядки. Важливим є і те, що до 2020 року більше 30 відсотків успішно існуючих сьогодні компаній – IT-постачальників різних послуг, які не перебудуються під нові цифрові відносини в кіберпросторі, перестануть існувати. Можна резюмувати, всі підприємства та організації, у тому числі університети, міста і країни, виявляться неконкурентоспроможними без інтеграції в кіберпростір заздалегідь підготовленими цифровими правовими відносинами (digital legislations).

## 6.2 MAT-комп'ютинг для фізичного, віртуального і космологічного простору

Мета – створення загальної моделі MAT-комп'ютингу, як універсальної та масштабованої структури моніторингу та управління фізичними, віртуальними і космологічними процесами і явищами, що надає людині можливість

освоювати нові життєві простори, у тому числі космос, а також створювати ринково-орієнтовані продукти і кіберсервіси, спрямовані на підвищення якості життя людини і збереження екосистеми планети [163, 164, 172-177].

Задачі: 1) Створення універсальної і масштабованої моделі МАТ-комп'ютингу, яка описує фізичні та віртуальні процеси моніторингу та управління. 2) Класифікація існуючих технологій комп'ютингу в реальному і віртуальному просторі для створення загальної картини кіберфізичного континууму (Cyber Physical Continuum). 3) Квантова телепортація процесів, але не об'єктів, як технологія проникнення людини в космос. 4) Кіберфізичний і квантовий комп'ютинг, як методологія опису процесів розвитку природи і суспільства.

Корисними можуть бути модифікації деяких визначень в бік спрощення. Комп'ютинг – це технологія інтерактивного моніторингу та управління процесами і явищами для досягнення мети за заданою програмою без (прямої) участі людини. Основа комп'ютингу – транзакційна взаємодія адресовних даних (компонентів пам'яті) для досягнення поставленої мети. Модель комп'ютингу: <Пам'ять, Адреса, Транзакція> (рис. 6.3, зліва). Адреса – субстанція, яка визначає структуру координатами компонентів у віртуальному чи реальному просторі. Структура формується різними субстанціями: гравітацією, внутрішньо- і міжатомною взаємодією (базон Пітера Хіггса), електромагнітними полями, електричними зв'язками, свідомістю людини, адресною взаємодією. Транзакція – цілеспрямований процес прийому-передачі даних між адресовними компонентами структурованої пам'яті для реалізації функціональності або сервісу. Пам'ять – будь-яка фізична субстанція, здатна зберігати, приймати і передавати дані, як віртуальну сутність. Оскільки будь-яка матерія складається з атомів і електронів, то кожна з чотирьох форм існування будь-якої матерії має властивості пам'яті.

Рівняння фізичного комп'ютингу задає системна симетрична (xor) взаємодія трьох рівнозначних субстанцій (Memory – Address – Transactions): $M \oplus A \oplus T = 0$. Це означає, що будь-який компонент визначається за допомогою

взаємодії двох інших: M=A⊕T, A=T⊕M, T=M⊕A. Віртуальний комп'ютинг оперує тріадою: (Data – Address – Transactions). Універсальне характеристичне рівняння комп'ютингу (комп'ютера), оперує єдиною адресною транзакцією (зчитування–запис) даних між джерелом і приймачем [163, 164]: M(Yi) = Qi [M(Xi)]. Тут W=<M, Q, X, Y> – структура, яка містить адресовні комопоненти пам'яті і нічого більше: M – пам'ять комп'ютингу; Q – пам'ять функціональних примітивів - квантів [163, 164]; X – пам'ять адрес вхідних змінних примітивів; Y – пам'ять адрес вихідних змінних примітивів. Природно, що всі компоненти пам'яті та комірки в них є адресовними. Це означає, що вони можуть замінюватися і ремонтуватися в режимі remoute and online. Недолік пам'яті один – програш у швидкості виконання транзакцій перед reusable logic або комбінаційними логічними елементами. Таким чином, комп'ютинг, як обчислювальний процес, зводиться до просторово-часової структури транзакцій над адресовними даними.

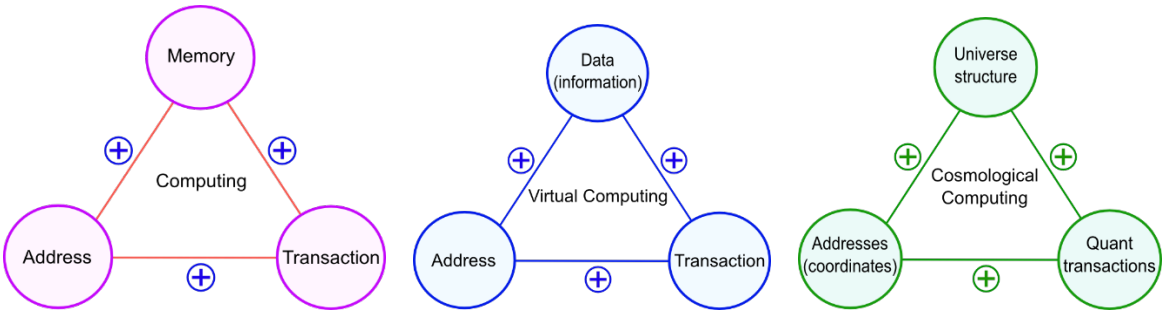



Рисунок 6.3 – Модель МАТ-комп'ютингу

Далі наведено види комп'ютингу (рис. 6.3) без участі людини, але для людини і збереження планети, із зазначенням пари: тип носія (пам'ять) – сигнатура (транзакції): 1) атомарний, стан або спін електрона - квант, фотон; 2) кристалічний, стан транзистора – електрони; 3) космологічний, стан матерії – гравітаційні і електромагнітні поля; 4) біологічний, стан клітини – електрони; 5) технологічний, стан системи – дані; 6) соціологічний, стан суспільства –

законодавство; 7) віртуальний (рис. 6.3, праворуч), стан кіберпростору – інформація.

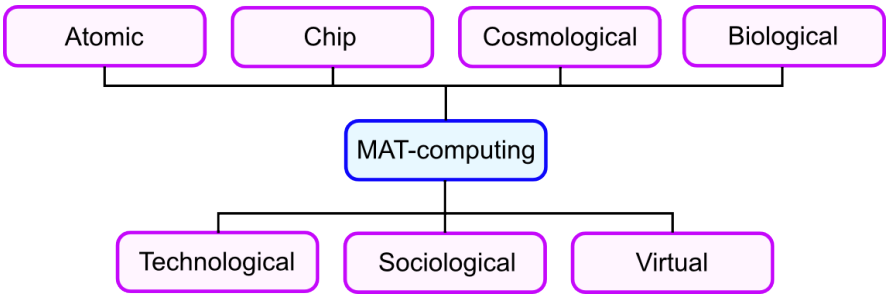



Рисунок 6.4 – Види комп'ютингу

Віртуальний або кіберкомпьютинг оперує даними (див. рис. 6.3, праворуч) замість пам'яті, розподілений в (кібер) просторі і сьогодні представлений модними структурними напрямами (рис. 6.4): 1) Cloud Computing; 2) Fog Network Computing; 3) Mobile Computing; 4) Service Computing (Web Services); 5) Social Computing; 6) Automotive Computing; 7) Internet Computing – Smart Everything; 8) Cyber Physical or Internet of Things (Everything) Computing; 9) Big Data Computing; 10) Quantum Computing (Security).

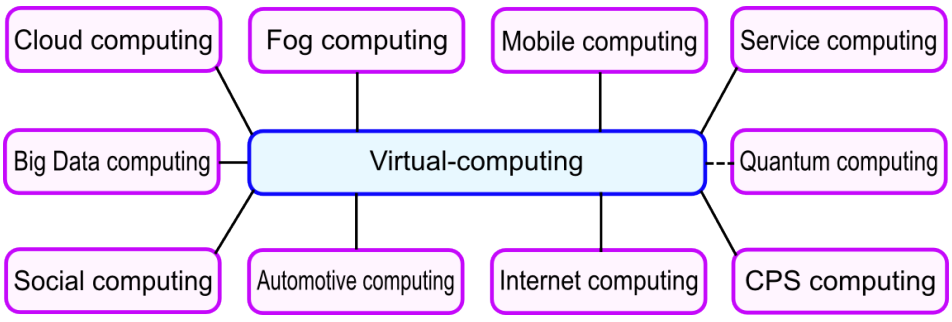



Рисунок 6.4 – Віртуальний (кібер) комп'ютинг

Фізичний комп'ютинг (рис. 6.5) сконцентрований у функціональностях, зосереджених у компактному просторі твердотільних спеціалізованих пристроїв: 1) Quantum Computers; 2) Mobile Gadgets and Laptops; 3) Automotive

Computers; 4) Smart Sensors and Actuators as MEMS; 5) Robotics; 6) Drones; 7) 3D-Printing; 8) Smart Brain-User Interfaces; 9) Security Computers; 10) Big Data Centers.

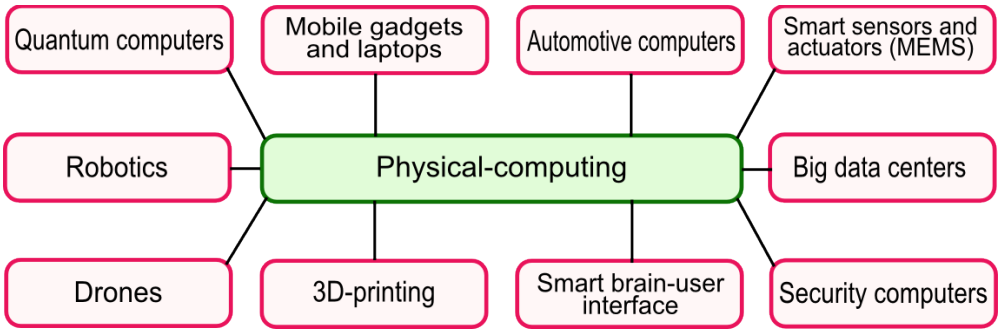



Рисунок 6.5 – Фізичний комп'ютинг

Види комп'ютингу, представлені вище, покриваються узагальненою автоматною структурою керованих фізичних і віртуальних процесів, відомих людству (рис. 6.6).

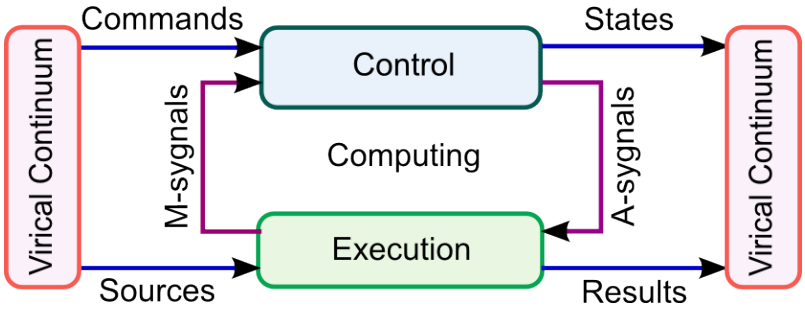



Рисунок 6.6 – Автоматна модель комп'ютингу

Тут в якості основного компонента фігурує пам'ять, як будь-яка субстанція, здатна зберігати інформацію, на якій організовані структурні компоненти комп'ютингу (Control and Execution). Функціонування і розвиток системи здійснюється за допомогою моніторингу (M-sygnals) і управління (актюації) (A-sygnals) компонентами обчислювальної системи. Входи і виходи такої системи навантажені на єдиних зовнішній віртуально-фізичний (virical)

простір (virtual-physical continuum) [Forbes, Josh Linkner]. Таке континуальне визначення кіберфізичного простору стосується не тільки Землі, але й Всесвіту, який також має програму свого еволюціонування за найпростішою і доступною для розуміння автоматною моделлю комп'ютингу. Модель адекватно, ідеально і просто описує всі процеси функціонування й еволюціонування технічних, біологічних, соціальних, віртуальних і космологічних об'єктів і структур на основі використання сигналів моніторингу та управління. При цьому представлений вище автомат споживає вхідні дані і ресурси із зовнішньої екосистеми, яка також є місцем, куди потрапляють результати або продукти діяльності конкретного комп'ютингу. Тому такий автомат, в залежності від мети, може бути перетворювачем кібер-фізичної екосистеми в Green Planet for Human або в сторону Armageddon деструктуризации.

## 6.3 Комп'ютинг квантової телепортації

Комп'ютинг квантової телепортації передбачає передачу на відстані не об'єктів, а технологічних процесів і відносин, які можна відтворити на приймальній стороні телекомунікаційного каналу. Прихід космічних прибульців на Землю не такий вже неймовірний. Сьогодні дозріває впевненість, що людство зможе продовжити життя на подібних до земних умов планетах у Всесвіті шляхом квантової передачі актюаторних потоків інформації, що описують структури і алгоритми еволюційного синтезу і розвитку земних субстанцій. Потрапивши в сприятливе середовище якої-небудь планети, квантокодовані геноми зможуть підготувати інфраструктуру, а потім виростити форсованим або еволюційним шляхом біологічні та технічні об'єкти. Здійснення такої дизрапторної експансії людини у відповідну область космічного простору реально навіть на сучасному рівні розвитку науки і технологій. При цьому космічні польоти з біологічними або технічними речовинами на борту для міжзоряних відстаней – тупиковий шлях фантастів і вчених. Таким чином, розширення життя людства в космосі реально і станеться саме шляхом квантової телепортації процесів синтезу біотехнічних об'єктів в

прийнятне планетарне середовище. Для цього вчені повинні в земних умовах вирішити технологічну проблему квантової актюації форсованого або еволюційного синтезу згаданих субстанцій. Неважко припустити, що більш розвинені цивілізації вже вирішили для себе проблему космічної експансії своїх форм життя і зробили квантову інтервенцію біотехнічних об'єктів на Землю шляхом фотонної доставки алгоритмів їх синтезу в благодатному земному середовищі.

Структурно-алгоритмічна сутність квантової телепортації об'єкта полягає в доставці даних про зв'язки компонентів в деяку точку простору, де є комплектуючі елементи, а також про послідовність дій, що становить оптимальну технологію виготовлення певної структури. Наприклад, в Монголії можна побудувати фабрику з виробництва інтегральних схем, для чого необхідно компанії Інтел найняти фахівців під систему відносин, що телепортується в дану країну, дати їм технології і техніку для виробництва чіпів. Щоб створити університет світового рівня (Кембридж, Стенфорд, Массачусетс), потрібно переміщувати в проблемний вуз тільки систему відносин (статут і традиції) від лідерів науки і освіти. Щоб створити сприятливу для населення країну, слід переміщувати гуманну людино-орієнтовану систему відносин (Конституцію), прийняту, наприклад, в США, Німеччині, Норвегії, Сінгапурі.

Таким чином, при наявності абсолютно однакових умов і компонентів ефективність і якість структури або системи визначаються тільки зв'язками або відносинами, так само як при наявності однакових кадрів, будівель і цілей ефективність університету визначається прийнятою в ньому системою відносин (статут, положення, традиції). Все сказане залишається в силі і може бути застосовано до успішності міст і країн. Можна також додати, що в разі неефективності або помилок в системі міняти потрібно не людей – безглузде заняття – люди скрізь однакові, а структуру відносин між ними. We are not looking for the people who made mistakes, but the situations where the mistakes have been possible (Stanley Hyduke).

Квантові (фотонні) транзакції космологічного комп'ютингу на структурованій гравітацією матерії у формі зірок і планет, що мають адреси в просторі, розширюють горизонти життя у Всесвіті.

Квантова структура біологічного генома проникає в космос і створює нові форми життя в інших світах. Людство не приречене бути вічно на Землі. Гравітація за допомогою бозона Хіггса створює структури, а фотони або кванти – транзакції на них. Структурні елементи Всесвіту мають адреси, виражені через просторово-часові координати завдяки гравітації. Бозон Хіггса вчені сьогодні розглядають як розчин для склеювання цегли світобудови в матеріальні форми і структури. Гравітація – сила тяжіння, що виробляється масами взаємодіючих космічних тіл – створює відносний структурний порядок у Всесвіті на короткому часовому інтервалі. Гравітації протидіє Великий вибух, сила якого служить причиною всіх змін у циклі еволюції Всесвіту до наступного Великого вибуху.

## 6.4 Висновки до розділу 6

Таким чином, з метою розробки хмарних сервісів керування трафіком для автотранспортних компаній та водія запропоновано кіберкультуру мікро-макро-космо-комп'ютингу, яка формулює, пояснює і прогнозує сучасні технології моніторингу та управління процесами і явищами в фізичному, віртуальному та космологічному просторі (у тому числі, транспортному комп’ютингу).

1) Запропоновано модель та напрями сталого розвитку кіберфізичного комп'ютингу для прогнозування аттракторів в області ІТ-індустрії, що базуються на просторово-часовому аналізі технологічних процесів, які характеризуються трьома фазами розвитку і дають можливість прогнозувати майбутні тренди розвитку кіберкультури.

2) Запропоновано кіберкультуру мікро-макро-космо-комп'ютингу, яка формулює, пояснює і прогнозує сучасні технології моніторингу та управління процесами і явищами в фізичному, віртуальному та космологічному просторі.

2) Представлено вербальні і структурні визначення основних типів комп'ютингу, що базуються на сучасних трендах еволюційного розвитку кіберекосистеми планети.

3) Сформульовано універсальну модель МАТ-комп'ютингу: <Memory, Address, Transactions>, яка використовує три компоненти для створення обчислювальної структури в технологічно прийнятному матеріальному середовищі.

4) Показано інформаційно-квантовий напрям експансії людини в космічний простір, а також можливість аналогічного проникнення неземних біотехнічних об'єктів в екосистему нашої планети.

5) Запропоновано модель комп'ютингу, яка визначає квазіоптимальні структури моніторингу та управління масштабованими процесами різної природи: технічними, біологічними, соціальними, віртуальними і космологічними.

# 7 СИНТЕЗ Q-ТЕСТІВ ЗА BLACK BOX КУБІТНИМ ОПИСОМ

Для проектування програмно-апаратних засобів керування транспортом використовуються квантові структури даних, моделі і методи логічного синтезу та аналізу цифрових пристроїв. Представлено методи і апаратно-програмні реалізації безумовного паралельного синтезу тестів на основі булевих похідних для логічних схем в black box формі, заданих кубітним покриттям. Надано теоретичне обґрунтування застосування методів і оцінки їх ефективностей для широкого класу цифрових схем, що імплементуются в кристали програмовних логічних пристроїв. Запропоновано інноваційні методи взяття булевих похідних і дедуктивного моделювання несправностей для функціональних елементів, описаних кубітними покриттями. Описано структуру процесора моделювання несправностей і справної поведінки для вирішення завдань синтезу тестів і діагностування дефектів, який може бути імплементований в інфраструктуру SoC або хмарний сервіс. Наведено приклади, що ілюструють технологічність реалізації методів в якості компонентів BIST-Infrastructure і хмарних сервісів проектування і верифікації SoC.

## 7.1 Реалізація логічних функциональностей на елементах пам'яті

Поняття адресного виконання логічних операцій, реалізованих на елементах пам'яті LUT в програмованих логічних пристроях (PLD), дає потенційну можливість створювати на кристалі тільки адресний простір, максимально технологічний для вбудованого відновлення працездатності всіх компонентів, що беруть участь у формуванні функціональності [178 – 181]. Актуальність створення адресного простору для всіх компонентів підтверджується наступним розподілом логіки і пам'яті на кристалі, представленим на рис. 7.1, де після 2020 року на чіпі буде тільки один відсоток логіки і 99 відсотків пам'яті.

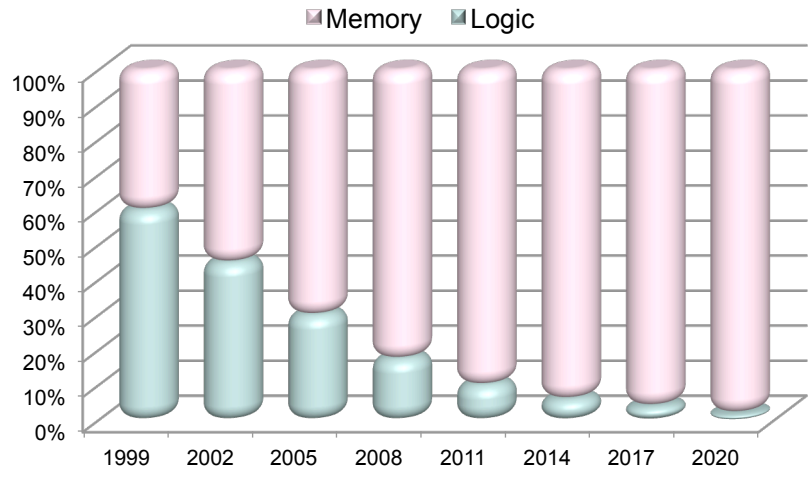



Рисунок 7.1 – Розподіл пам'яті і логіки на кристалі

Тенденція до збільшення пам'яті тягне можливість вбудованого відновлення працездатності відмовних комірок за рахунок виділених додаткових ресурсів для їх ремонту (spare logic cells). Проблема автономного усунення дефектів (самовідновлення працездатності) логічних елементів пов'язана з відсутністю у них адрес. Але вирішити її можна, якщо зв'язки між елементами логіки зробити гнучкими за допомогою програми опису структури, розміщеної у пам'яті, яка з'єднає логічні компоненти в схему. Крім структури взаємодії елементів пам'ять повинна містити порядок їх обробки. У разі виникнення дефекту в одному з адресовних логічних елементів, система вбудованого тестування відновить його працездатність шляхом переадресації на завідомо справний аналог з ремонтного запасу. Просто вирішується проблема підвищення якості та надійності цифрових систем на кристалах шляхом створення інфраструктури вбудованого тестування, діагностування, оптимізації та відновлення працездатності за рахунок апаратної надлишковості і зменшення швидкодії виконання функціональних операцій [180, 181, 188].

Як наочний приклад, архітектура FPGA Family Xilinx® UltraScaleTM Architecture Configurable Logic Block (the first ASIC-class All Programmable architecture) належить уже до класу ASIC і забезпечує продуктивність реалізованих смарт-функціональностей на рівні сотень гігабіт за секунду.

Вона виконана у вигляді 3D-on-3D об'ємному кристалі по технології 20 nm на основі FinFET транзисторів і працює як мультипроцесорна SoC (MPSoC), забезпечуючи мережеву продуктивність 1+ Tb/s обміну даними. [Www.xilinx.com/support/documentation/user_guides/ug574-ultrascale-clb.pdf].

На початку 2000-х років співвідношення продуктивності процесорної логіки і пам'яті було, як 100:1. В даний час ситуація поступово вирівнюється в сторону зменшення відмінностей, як 10:1, за рахунок підвищення швидкодії транзкцій на пам'яті. Це є істотним аргументом для прискорення процесу «меморизації» – реалізації логіки на пам'яті з метою підвищення надійності цифрових систем на кристалах за рахунок часткової втрати продуктивності. В кінцевому рахунку пам'ять і логіка мають загальну атомарну межа досконалості, коли біт пам'яті буде ототожнюватися зі спіном або орбітою електрона. Швидкодія межатомарних транзакцій на основі обміну квантами стане сумірною зі швидкістю світла. Класична транзисторна логіка, керована потоком заряджених частинок, в межі – одним електроном, має ту ж саму межу досконалості – зміна станів електрона під впливом кванта. Таким чином, перехід електронної індустрії на only-memory-computing (Memory-Address-Transaction) неминучий. Звідси випливає висновок: своєчасне створення теорії синтезу та аналізу цифрових обчислювальних пристроїв на основі МАТ-комп'ютингу може принести ринку принципово нові можливості для вирішення проблем надійності, швидкодії та енергозбереження.

Мета – істотне підвищення швидкодії синтезу тестів і дедуктивної верифікації для black box функціональностей логічних компонентів за рахунок використання компактних описів у формі кубітних покриттів і паралельного виконання мінімальної кількості регістрових логічних операцій (shift, or, not, nxor).

Задачі дослідження пов'язані з розробкою:

1) Методу генерації тестів для black box функциональностей логічних схем на основі використання кубітних покриттів і паралельного виконання регістрових логічних операцій (shift, or, not, nxor).

2) Методу взяття булевих похідних для синтезу тестів на основі використання кубітних покриттів.

3) Методу синтезу тестів на основі використання булевих похідних, представлених векторами в форматі кубітних покриттів.

4) Методу дедуктивного моделювання несправностей для функціональних елементів, представлених векторами кубітних покриттів.

5) Процесору моделювання несправностей і справної поведінки на основі кубітного опису функциональностей, який інтегрує розроблені методи в хмарний або SoC-інфраструктурний сервіс тестування.

6) Тестуванням і верифікацією методів синтезу тестів і моделювання несправностей шляхом паралельного виконання регітрових логічних операцій над кубітними описами прикладів логічних схем.

Сутність дослідження – розробка методів генерування вхідних тестових послідовностей і оцінки їх якості для функціональних логічних компонентів шляхом паралельного виконання регістрових логічних операцій (shift, or, not, nxor) над кубітним покриттям і його похідними в структурі процесора кубітного моделювання.

Тестування і діагностування запам'ятовувальних пристроїв реалізується за допомогою спеціальних алгоритмів, що генерують тести: марш – нуль і одиниця, що біжать, логарифмічного поділу [179 – 181, 188]. Якщо елементи пам'яті виконують функції логічних схем (reusable logic), то для них слід генерувати тести перевірки функциональностей, не прив'язані до фізики процесів зберігання даних. Тому далі розглядається метод синтезу тестів для функціональних схем, представлених у формі black box, опис яких задається кубітним вектором [182, 183].

Ринкова привабливість застосування квантових методів обчислень при створенні компьютингових структур в кіберпросторі визначається використанням кубітних моделей даних, орієнтованих на паралельне вирішення задач проектування, тестування, дискретної оптимізації [181], завдяки збільшенню витрат пам'яті. Не вдаючись в деталі фізичних основ

квантової механіки, що стосуються недетермінованої взаємодії атомних часток [184, 185], далі використовується поняття кубіта як двійкового вектора для спільного і одночасного завдання булеана станів в дискретної області кіберпростору на основі лінійної суперпозиції унітарних кодів, орієнтованих на паралельне виконання операцій. У теорії квантових обчислень, що швидко розвивається, вектори станів утворюють квантовий регістр з n кубітів, які формують унітарний або Гільбертів [186] простір H, розмірність якого має ступеневу залежність від кількості кубітів $\mathrm{Dim}\ \mathrm{H} = 2^{\mathrm{n}}$.

Мотивація нового підходу для проектування компьютингових систем обумовлена появою хмарних сервісів в рамках нової кіберкультури Internet of Things, яка являє собою спеціалізовані і розосереджені в просторі системи, реалізовані в апаратурі або в програмному продукті. Будь-який компонент функціональності, так само як і структура системи, може бути представлений векторною формою, впорядкованою за адресами, таблиці істинності, що реалізується за допомогою пам'яті. Логічні функції в традиційному виконанні reusable logic не розглядаються. Від цього частково зменшується швидкодія, але, з огляду на те, що 94% SoC-кристалу становить пам'ять [179, 180], решта 6% будуть імплементуватися в пам'ять, що не критично для більшості хмарних сервісів. Практично для створення ефективних компьютингових структур слід використовувати теорію, що базується на обчислювальних компонентах високого рівня абстракції: адресовна пам'ять і транзакція.

Згідно квантової теорії електрон може бути в двох місцях одночасно. З цього твердження випливає, що точка простору може одночасно містити два електрони або два стійких стани. Квант (кубіт) тут розглядається як дискретний стан точки в кіберпросторі, який визначається суперпозицією декількох примітивів універсуму.

Особливість організації даних в класичному комп'ютері полягає в адресації біта, байта або іншого компонента. Адресовність створює проблему обробки асоціації неадресовних елементів множини, які не мають порядку за визначенням. Рішенням може бути процесор, де образ універсуму з n

унітарно кодованих примітивів використовує суперпозицію для формування булеана $|B(A)| = 2^n$ всіх можливих станів [162, 187].

Існує аналогія векторного представлення булеана з кубітом квантового комп'ютера. Квантовому кубіту зберігання інформації в квантовому комп'ютері [184, 185], що дозволяє суперпозицію, можна поставити у взаємно однозначну відповідність булеан станів, який формує алфавіт Кантора. Тут примітиви алфавіту унітарно кодуються векторами: 0=(10) і 1=(01). Коди інших символів є похідними по операції суперпозиції (10)∨(01) = (11) і перетину (10) ∧ (01) = (00), що формує булеан: {(10),(01),(11),(00)} [162, 186, 187].

Коректність використання прикметника «кубітна» для моделей цифрових пристроїв базується на порівнянні лінійної і булевої алгебри Кантора з алфавітом $A^k = \{0,1,X,\emptyset\}$. Тут перші два символи – примітиви. Третій визначається суперпозицією: $X = 0 \cup 1$. В гільбертовому просторі лінійна алгебра оперує примітивами $\alpha|0\rangle, \beta|1\rangle$ кубіта, третій символ також є суперпозиція двох складових: $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$. В лінійній алгебрі немає символу, відповідного порожній множині, він є похідним від функції, зворотної по відношенню до суперпозиції. В теорії множин такою операцією є перетин, яке дає порожня множина на примітивах: $\emptyset = 0 \cap 1$. В гільбертовому просторі такою операцією є скалярний добуток або функція Дірака $\langle\alpha|b\rangle$ [192], яка має геометричну інтерпретацію: $\langle\alpha|\beta\rangle = |\alpha|\times|\beta|\times \cos\angle(\alpha,\beta)$. Якщо проекції a і b вектора квантового стану ортогональні, виходить:$\langle\alpha|\beta\rangle = |\alpha|\times|\beta|\times\cos\angle(\alpha,\beta) = |\alpha|\times|\beta|\times\cos\angle 90° = 0$. Скалярний добуток ортогональних векторів дорівнює нулю, що є аналогом символу порожньої множини в алгебрі Кантора. Таблиця відповідності алгебри множин та лінійної алгебри, наведена нижче, підтверджує властивості ізоморфізму між символами булеана і станами кубіт-вектора:

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

.

Отже, структуру даних «булеан» можна розглядати як детермінований образ квантового кубіта в алгебрі логіки, елементи якої унітарно кодуються двійковими векторами і мають властивості суперпозиції, паралелізму і переплутування. Це дає можливість використовувати пропоновані кубітні моделі для підвищення швидкодії аналізу цифрових пристроїв на класичних обчислювачах, а також без модифікації – в квантових комп'ютерах, які з'являться через кілька років на ринку електроніки.

Квантовий опис цифрових функціональних елементів. Кубіт (n-кубіт) є векторною формою унітарного кодування універсуму з n примітивів для завдання булеана станів за допомогою двійкових змінних. Якщо n=2, то 2-кубіт задає 16 станів за допомогою чотирьох змінних, при n=1, кубіт задає чотири стану на универсумі з двох примітивів (10) і (01) за допомогою двох двійкових змінних (00,01,10,11) [162]. При цьому допускається суперпозиція у векторі $2^n$ станів, позначених примітивами. Синонімом кубіта при завданні двійкового вектора логічної функції є Q-покриття (Q-вектор) [163, 164] як уніфікована векторна форма суперпозиційного завдання вихідних станів, відповідних адресним кодами вхідних змінних функціонального елемента. Формат структурного кубітного компонента цифрової схеми $Q^* = (X, Q, Y)$ містить інтерфейс (вхідні та вихідну змінні), а також кубіт-вектор Q, що задає функцію $Y = Q(X)$, розмірність якого визначається ступеневою функцією від кількості вхідних ліній $k = 2^n$. Новизна кубітної форми полягає в заміні невпорядкованих по рядках таблиць істинності функціональних елементів векторами упорядкованих станів виходів. Наприклад, якщо функціональний примітив має двійкову таблицю, то йому можна поставити у відповідність кубіт або Q-покриття: Q = (1110):

$$C = \begin{matrix} X_1 & X_2 & Y \\ 0 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{matrix} \rightarrow Q = (1110)$$

Таким чином, дизрапторна ідея, покладена в основу досліджень, полягає в заміні множини вхід-вихідних відповідностей таблиці істинності кубітним вектором адресовних вихідних станів, рис. 7.2.

Примітивізм і компактність кубітної векторної форми (Q-coverage) диктує застосування тільки простих паралельних регістрових операцій над його вмістом: (not, shift, or, and, xor) для вирішення всіх завдань синтезу та аналізу цифрових виробів.

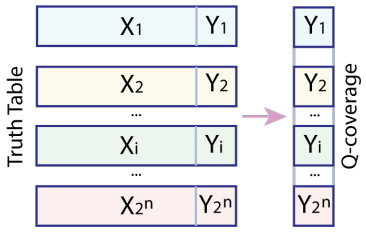



Рисунок 7.2 – Таблиця істинності та кубітне покриття

7.2 Кубітний метод синтезу тестів

Пропонується метод синтезу тестів, що використовує кубітні вектори або Q-покриття функціональних примітивів цифрових пристроїв, який характеризується компактністю опису даних і паралелізмом виконання логічних операцій.

Q-покриття є векторна форма опису поведінки цифрового пристрою, де кожен розряд має адресу, що формується двійковими станами його вхідних змінних:

$$Q = (Q_1, Q_2, ..., Q_i, ..., Q_n),\ Q_i = \{0,1\},\ Q_i = Q(i),\ i = (X_1X_2, ..., X_i, ..., X_n).$$

Q-тест є векторною формою неявного завдання тестових послідовностей цифрового пристрою, де координати вектора формують впорядковану послідовність двійкових наборів, що подаються на вхідні змінні за правилом:

$$Q_i = Q(i) = \begin{cases} 1 \to (X_1X_2, ..., X_i, ..., X_n) = i, \\ 0 \to (X_1X_2, ..., X_i, ..., X_n) = \varnothing. \end{cases}$$

Іншими словами, якщо координата вектора Q(i)=1, то тестовий набір, складений з двійкових розрядів, які формують десяткову адресу i, подається на входи пристрою. В іншому випадку, при нульовому значенні координати Q(i)=0, такий тестовий набір відсутній.

Природно, що розмірності Q-покриття і Q-тесту завжди однакові, що робить можливим виконувати паралельний аналіз їх спільної взаємодії при синтезі тестів для функціональних елементів за правилом [187]:

$$F \oplus Q \oplus T = 0, \;\; T = Q \oplus F.$$

Приклад 1. Необхідно згенерувати тест перевірки одиночних константних несправностей для логічного елемента 2and, заданого таблицею істинності. Для цього задається таблиця впливу одиночних константних несправностей вхідних змінних, з числом кубів, що дорівнює кількості входів. Кожен куб несправності має одиницю на координаті вхідної змінної і на виході, а решта координат дорівнюють нулю. Це означає, що існує залежність: зміна вхідної змінної повинна викликати зміну стану виходу.

Алгоритм синтезу тесту по таблиці істинності [187]:

1) На першому етапі виконується покоординатна xor-операція між таблицею істинності і кожним рядком матриці несправностей, яка генерує дві таблиці, по кількості вхідних змінних, що містять рядки-кандидати в тест:

$$\begin{bmatrix} X_1 & X_2 & Y \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \oplus \begin{bmatrix} F_1 & F_2 & F_Y \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} = \begin{bmatrix} X_1 & X_2 & \overline{Y}_1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \vee \begin{bmatrix} X_1 & X_2 & \overline{Y}_2 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

2) Потім рядки двох останніх таблиць упорядковуються за правилом зростання двійкових кодів вхідних змінних:

$$\begin{bmatrix} X_1 & X_2 & \overline{Y}_1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \oplus \begin{bmatrix} X_1 & X_2 & \overline{Y}_2 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} X_1 & X_2 & Y'_1 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} X_1 & X_2 & Y'_2 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

3) Після цього виконується порівняння отриманих станів виходів ($Y'_1 Y'_2$) з вектором значень виходів вихідної таблиці істинності Y за правилом

операції еквівалентності (nxor) – якщо значення однойменних координат двох кубітів рівні, то результат порівняння дорівнює 1, в іншому випадку – 0:

$$\begin{bmatrix} Y \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \overline{\oplus} \begin{bmatrix} Y_1' \\ 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} \vee \begin{bmatrix} Y_2' \\ 1 \\ 1 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} Y_1^t \\ 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} Y_2^t \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

4) На останньому кроці виконується операція логічного додавання отриманих результатів порівняння, яка дає Q-тест або T-вектор, одиничні координати якого ідентифікують тільки ті двійкові вхідні послідовності, які слід подавати на входи цифрового пристрою для його перевірки:

$$\begin{bmatrix} Y_1^t \\ 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} \vee \begin{bmatrix} Y_2^t \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} T \\ 0 \\ 1 \\ 1 \\ 1 \end{bmatrix} \rightarrow \begin{bmatrix} X_1 & X_2 & Y \\ . & . & . \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

В даному прикладі Q-тест містить три одиниці 0111, тому відповідні йому вхідні набори представлені трьома векторами: 010, 100, 111.

Приклад 2 ілюструє роботу алгоритму [187] синтезу тесту по таблиці істинності для функціонального елемента, що має три вхідних змінних. Тут також фігурує таблиця істинності, в якій останній стовпець Y є кубіт-вектором. Стовпці ($F_1F_2F_3F_Y$) формують матрицю несправностей, сутність кожного рядка якої – одновимірна активізація входу на вихід. Стовпці ($Y_1'Y_2'Y_3'$) – отримані в результаті хог-взаємодії таблиці істинності і матриці несправностей, яка визначає поведінку функціональності при внесенні на кожен вхід несправності, інверсної по відношенню до її справної поведінки. Стовпці ($Y_1^tY_2^tY_3^t$) отримані шляхом покоординатного порівняння вектора вихідних станів вхідної таблиці істинності з наборами ($Y_1'Y_2'Y_3'$), отриманими на попередньому етапі. Вектор T – логічно об'єднує набори ($Y_1^tY_2^tY_3^t$) в тест для заданої функціональності:

| $X_1$ | $X_2$ | $X_3$ | Y | $F_1$ | $F_2$ | $F_3$ | $F_Y$ | $Y_1'$ | $Y_2'$ | $Y_3'$ | $Y_1^t$ | $Y_2^t$ | $Y_3^t$ | T |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | | | | | 1 | 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 0 | | | | | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | | | | | 0 | 1 | 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | | | | | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 0 | | | | | 0 | 0 | 0 | 1 | 1 | 1 | 1 |

Останній стовпець являє собою тест перевірки несправностей зовнішніх вхідних і вихідних змінних функціонального елемента T=(10111111). Тест-вектор задає такі вхідні набори, які слід подати на входи функціонального елемента: 0001 0100, 0111 1000, 1011, 1101, 1110, щоб перевірити всі дефекти заданого класу.

Формальний Q-алгоритм синтезу тестів для константних несправностей функціонального примітиву на основі використання Q-покриття містить такі пункти:

1) Інвертування Q-покриття: $Q_i = \overline{Q}_i, i = \overline{1, 2^n}$.

2) Упорядкування інверсного Q-покриття для кожної з n вхідних змінних $Q_j = S_j(\overline{Q}), j = \overline{1, n}$. Дана процедура зводиться до виконання операцій логічного зсуву (shift) на кубітному векторі, кожна з яких має власний алгоритм для даної вхідної змінної, що ілюструється наступною схемою для трьох входів, рис. 7.3.

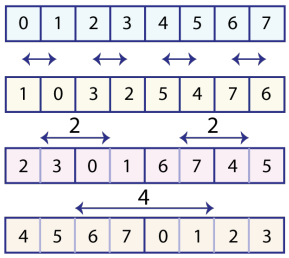



Рисунок 7.3 – Операції зустрічного зсуву на регістрах

Тут другий рядок адресними індексами задає процедуру обміну даними між сусідніми координатами кубіт-вектора шляхом зустрічного зсуву вмісту двох сусідніх координат, що формує тест-вектор для першої вхідної змінної. Третій рядок ілюструє обмін даними між сусідніми парами координат кубіт-вектора шляхом зустрічного зсуву їх вмісту, що формує тест-вектор для другої вхідної змінної. Четвертий рядок задає обмін даними між сусідніми тетрадами координат кубіт-вектора шляхом зустрічного зсуву їх вмісту, що формує тест-вектор для третьої вхідної змінної.

3) Отримання Т-векторів для кожної з n вхідних змінних шляхом порівняння з вихідним кубітним покриттям функціонального елемента:

$$T_j = Q \overline{\oplus} \overline{Q}_j, j = \overline{1, n}.$$

4) Отримання Q-тесту функціональності шляхом логічного об'єднання Т-векторів для кожної вхідної змінної:

$$T = \vee_{j=1}^{n} T_j.$$

Інтегрально алгоритм безумовного синтезу тестів для функціональних елементів, заданих кубітнимі покриттями компактно може бути записаний у вигляді такої формули:

$$T = \vee_{j=1}^{n} [Q \overline{\oplus}_{j=1}^{n} S_j(\overline{Q})].$$

Обчислювальна складність безумовного алгоритму синтезу тестів на основі послідовного використання регістрових логічних операцій (not – shift – nxor – or) для функціонального елемента, описаного кубіт-вектором, представлена наступним виразом:

$$q = n + n \times 2^n + n + n = n(2^n + 3).$$

Тут n – кількість змінних; перший доданок визначає обчислювальну складність операції 1) інверсії; другий – 2) логічний зсув для перестановки станів координат кубіт-вектора відносно кожної з n вхідних змінних; третій – 3) nxor-порівняння отриманих тест-векторів з вихідним кубіт-покриттям функціонального елемента; четвертий – 4) об'єднання тест-векторів для вхідних змінних.

Абстрагуючись від поняття таблиці істинності, далі пропонується формальний безумовний алгоритм кубітного синтезу тестів для функціональних примітивів на основі Q-покриття, стосовно раніше розглянутого прикладу:

1) Інвертування всіх розрядів Q-покриття функціонального елемента, що має три вхідних змінних:

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Q | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| $\overline{Q}$ | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |

2) Логічний зсув номерів розрядів інвертованого кубітного вектора відповідно до послідовностів номерів:

$$Q_1(X_1) = 45670123, Q_2(X_2) = 23016745, Q_3(X_3) = 10325476.$$

Тут діє просте логічне правило: для першої вхідної (молодшої) змінної виконується паралельний обмін даними між сусідніми координатами, для другої вхідної змінної реалізується обмін даними але вже між сусідніми парами координат, для третьої змінної виконується обмін даними між сусідніми тетрадами координат вектора адрес.

Узагальнення операцій зсуву для функціональності, що містить 4 змінних, представлено на рис. 7.4. Збільшення кількості змінних принципово не змінює суті зсуву даних: зустрічний зсув двох бітів, пар бітів, тетрад бітів, вісімок, ....

Процедура зустрічного зсуву даних в інверсному кубітному векторі є найбільш часовитратною в алгоритмі синтезу тестів для функціональних елементів. Тому її швидкодія матиме максимальне значення при апаратній реалізації зсувних операцій.

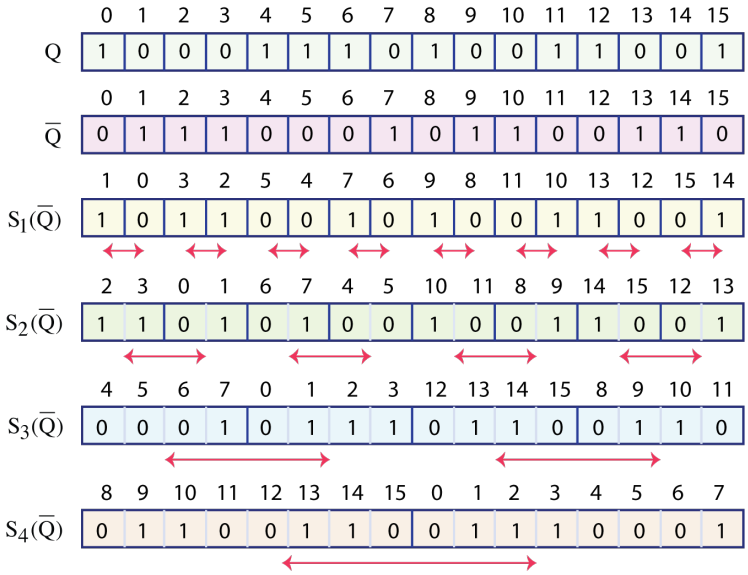



Рисунок 7.4 – Операції зсуву для чотирьох-входової функціональності

Процедурна реалізація обміну номерами для формування тест-векторів перевірки несправностей вхідних змінних, стосовно наведеного вище прикладу, має такий вигляд:

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Q | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| $\overline{Q}_1$ | 1 | 0 | 3 | 2 | 5 | 4 | 7 | 6 |
| $\overline{Q}_2$ | 2 | 3 | 0 | 1 | 6 | 7 | 4 | 5 |
| $\overline{Q}_3$ | 4 | 5 | 6 | 7 | 0 | 1 | 2 | 3 |

Цікаво, що для першої змінної існує проста формула перенумерації комірок кубіт-вектора для синтезу тестів: $j = j + (-1)^j$. Для інших змінних такої простої залежності визначити поки не вдалося.

З урахуванням введених правил зустрічного зсуву комірок кубіт-вектора реалізація даного пункту алгоритму для Q = 11010110 представлена таблицею:

$$\begin{aligned} Q &= 1\ 1\ 0\ 1\ 0\ 1\ 1\ 0 \\ \overline{Q} &= 0\ 0\ 1\ 0\ 1\ 0\ 0\ 1 \\ \overline{Q}_1 &= 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0 \\ \overline{Q}_2 &= 1\ 1\ 0\ 1\ 0\ 1\ 1\ 0 \\ \overline{Q}_3 &= 0\ 0\ 0\ 1\ 0\ 1\ 1\ 0 \end{aligned}$$

Таким чином, кожна змінна визначає розбиття кубіт-вектора на групи упорядкованих послідовностей. Перша (молодша) змінна створює зміни

станів за правилом: додати 1 до поточної парної адреси комірки: j=j+1, відняти 1 від поточної непарної адреси комірки: j=j-1. Для другої змінної розглядаються вже пари комірок, які обробляються за правилом: додати 2 до поточної парної адреси пари комірок: j=j+2, відняти 2 від поточної непарної адреси пари комірок: j=j-2. Для третьої змінної розглядаються вже тетради комірок, які обробляються за правилом: додати 4 до поточної парної адреси тетради комірок: j=j+4, відняти 4 від поточної непарної адреси тетради комірок: j=j-4.

3) Порівняння за допомогою операції еквівалентності отриманих, в даному випадку, трьох інвертованих і переупорядкованих Q-векторів з вихідним Q-покриттям функціонального елемента:

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| $Q$ | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 |
| $\overline{Q}_1$ | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 |
| $S_1(\overline{Q})$ | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| $S_2(\overline{Q})$ | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| $S_3(\overline{Q})$ | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 |
| $Q\overline{\oplus}S_1(\overline{Q})$ | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 |
| $Q\overline{\oplus}S_2(\overline{Q})$ | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 |
| $Q\overline{\oplus}S_3(\overline{Q})$ | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |

4) Диз'юнкція отриманих трьох векторів формує Q-тест, одиничні координати яких визначають тестові набори для перевірки всіх одиночних константних несправностей зовнішніх входів і виходів.

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| $Q$ | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 |
| $\overline{Q}$ | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 |
| $S_1(\overline{Q})$ | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| $S_2(\overline{Q})$ | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| $S_3(\overline{Q})$ | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 |
| $T_1 = Q\overline{\oplus}S_1(\overline{Q})$ | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 |
| $T_2 = Q\overline{\oplus}S_2(\overline{Q})$ | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 |
| $T_3 = Q\overline{\oplus}S_3(\overline{Q})$ | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| $T = T_1 \vee T_2 \vee T_3$ | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |

З огляду на істотність зустрічних регістрових зсувів, нижче пропонується універсальний алгоритм отримання перестановок кубітних фрагментів в залежності від номера вхідної змінної.

Послідовний алгоритм зустрічного зсуву даних в кубітних покриттях для отримання Q-тестів вхідних змінних має три вкладених цикли:

1) Завдання i-номера вхідної змінної або кроку $2^i$ для зустрічного зсуву даних в кубите: $i = \overline{1,n}$.

2) Формування циклу обробки кубіта з уже заданим кроком $2^i$ (2,4,8,16 ...), кратним ступеня двійки: $j = \overline{0, 2^n - 1}, 2^i$. Залежно від номера вхідної змінної і формуються такі послідовності індексу j=f(i): [(0,2,4,6 ...), (0,4,8,12 ...), (0,8,16,32 ... )].

3) Завдання циклу зустрічного зсуву даних для пари сусідніх груп: $t = j, j + 2^{i-1} - 1$. Значення індексу t=f(i,j): обробка кубіта першої змінної – (0,0; 2,2; 4,4 ...), другої змінної – (0,1; 4,5; 8,9 ...), третьої змінної – (0,3; 8,11; 16,19 ...). Виконання операцій зсуву за допомогою використання буферного регістру B (рис. 7.5):

$$(B_t = Q_{i,t+j}) \rightarrow (Q_{i,t+j} = Q_{i,t}) \rightarrow (Q_{i,t} = B_t)$$

Кінець алгоритму зсуву даних в регістрах.

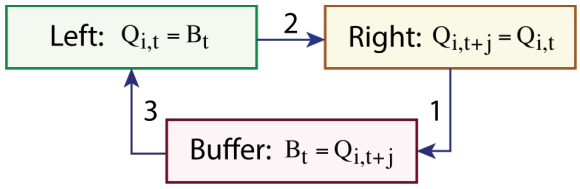



Рисунок 7.5 – Зустрічний зсув сусідніх частин кубіта

Обчислювальна складність послідовного алгоритму для обробки кубітного покриття функціональності, що має n вхідних змінних, дорівнює $q = 3n2^n$.

Приклад 3. Побудувати тест для перевірки функціональності, заданої логічним рівнянням: $Y = \overline{X}_1\overline{X}_2 \vee \overline{X}_1X_2$. Результати виконання алгоритму синтезу тесту по заданому кубіт-вектору представлено в наступному вигляді:

| Q | 0 0 1 1 |
|---|---|
| $\overline{Q_1}$ | 1 1 0 0 |
| $S_1(\overline{Q})$ | 0 0 1 1 |
| $S_2(\overline{Q})$ | 1 1 0 0 |
| $T_1 = Q\overline{\oplus}S_1(\overline{Q})$ | 1 1 1 1 |
| $T_2 = Q\overline{\oplus}S_2(\overline{Q})$ | 0 0 0 0 |
| $T = T_1 \vee T_2$ | 1 1 1 1 |

Результуючий тест містить 4 вхідних послідовності, кожна з яких перевіряє несправності для вхідної змінної $X_1$. Однак тест для змінної $X_2$ не існує, оскільки $T_2$=0000. Це означає, що змінна не може бути перевірена, а значить вона не є суттєвою. Дійсно, перетворення вихідної функції

$$Y = \overline{X}_1\overline{X}_2 \vee \overline{X}_1X_2 = \overline{X}_1(\overline{X}_2 \vee X_2) = \overline{X}_1$$

в сторону мінімізації виключає змінну $X_2$, як надлишкову або несуттєву. Таким чином, метод генерації тестів на основі кубітного покриття дає додаткову можливість визначати істотність змінних і мінімізувати (зменшувати розмірність) Q-вектор шляхом зменшення кількості змінних.

Приклад 4. Побудувати тест для перевірки функціональності, заданої логічним рівнянням: $Y = \overline{X}_1X_2 \vee X_1\overline{X}_2$. Результати виконання алгоритму синтезу тесту по заданому кубіт-вектору на основі послідовного виконання чотирьох регістрових логічних операцій (not, shift, nxor, or) представлені у вигляді:

| Q | 0 1 1 0 |
|---|---|
| $\overline{Q_1}$ | 1 0 0 1 |
| $S_1(\overline{Q})$ | 0 1 1 0 |
| $S_2(\overline{Q})$ | 0 1 1 0 |
| $T_1 = Q\overline{\oplus}S_1(\overline{Q})$ | 1 1 1 1 |
| $T_2 = Q\overline{\oplus}S_2(\overline{Q})$ | 1 1 1 1 |
| $T = T_1 \vee T_2$ | 1 1 1 1 |

Отриманий тест містить 4 вхідних послідовності, кожна з яких перевіряє несправності для вхідних змінних $X_1, X_2$. Таким чином, метод генерації тестів на основі кубітного покриття дозволяє згенерувати тест для функціонального компонента, але не надає інструменту зробити його мінімальним.

Приклад 5. Синтезувати тест для перевірки функціональності, заданої логічним рівнянням: $Y = \overline{X}_1 X_2 \overline{X}_3 \vee X_1 \overline{X}_2 \overline{X}_3 \vee X_1 X_2 X_3 \vee \overline{X}_1 X_2 X_3$. Результати виконання алгоритму синтезу тесту по Q-вектору на основі послідовного виконання чотирьох регістрових логічних операцій (not, shift, nxor, or) представлені у вигляді:

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| $Q$ | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 |
| $\overline{Q}$ | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 |
| $S_1(\overline{Q})$ | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 |
| $S_2(\overline{Q})$ | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 |
| $S_3(\overline{Q})$ | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 |
| $T_1 = Q \overline{\oplus} S_1(\overline{Q})$ | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| $T_2 = Q \overline{\oplus} S_2(\overline{Q})$ | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| $T_3 = Q \overline{\oplus} S_3(\overline{Q})$ | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| $T = T_1 \vee T_2 \vee T_3$ | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

На рис. 7.6 представлений секвенсор синтезу тестів для функціональних елементів, заданих кубітними покриттями.

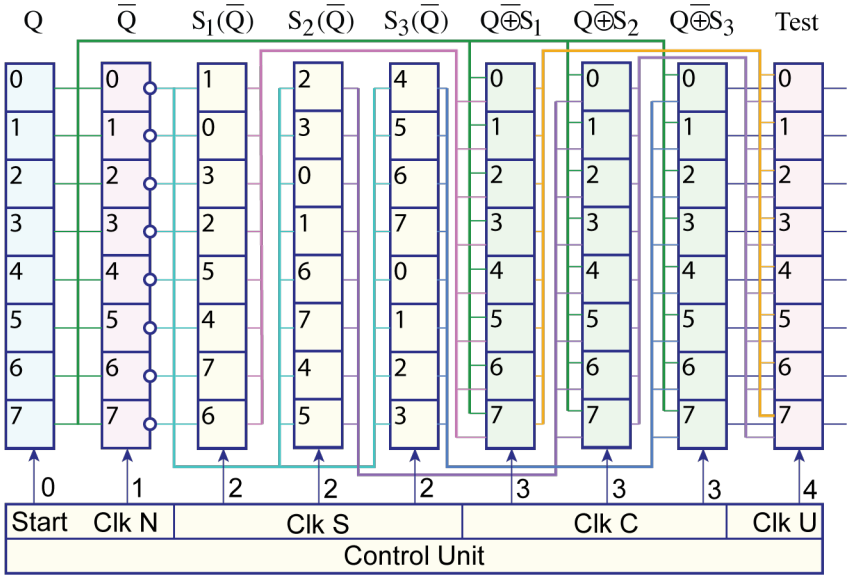



Рисунок 7.6 – Секвенсор синтезу тестів

Він містить модуль керування, який розподіляє чотири синхроімпульси, створюючи цикл генерації тесту, що передбачає виконання 4 паралельних операцій послідовно: 0) Початкова стартова операція, що

ініціюється сигналом Start, передбачає завантаження в регістр кубітного покриття. 1) Потім синхроимпульс Clk N активує виконання операції інверсії Not над вмістом регістру кубітного покриття. 2) Синхросигнал Clk S активує виконання операцій зустрічного зсуву над вмістом, в даному випадку трьох, регістрів, і отримання таких результатів: $S_1$ (not Q), $S_2$ (not Q), $S_3$ (not Q). 3) Потім синхросигнал Clk C ініціює виконання паралельних операцій порівняння (в даному випадку для трьох регістрів) отриманих кандидатів в тест з початковим кубітним покриттям: nxor (notQ, $S_1$), nxor (notQ, $S_2$), nxor (notQ, $S_3$). 4) Синхросигнал Clk U активує виконання or-операції над кандидатами в тест, у даному випадку реалізацію логічного об'єднання вмісту трьох регістрів:

$$T = \mathrm{or}[\mathrm{nxor}(\mathrm{notQ}, S_1), \mathrm{nxor}(\mathrm{notQ}, S_2), \mathrm{nxor}(\mathrm{notQ}, S_3)].$$

Якщо не економити на апаратурі, то швидкодію тестового генератора можна довести до чотирьох автоматних тактів q=4 паралельного виконання логічних регістрових операцій, що дає можливість вбудовувати секвенсор у BIST-інфраструктуру цифрових систем на кристалах для online тестування функціональних елементів. При цьому сумарна кількість $2^n$ –розрядних регістрів дорівнюватиме N(n)=1+1+n+n+1=(2n+3), а загальний обсяг регістрової пам'яті в бітах буде дорівнює N(R)=(2n+3) $2^n$, де n – кількість вхідних змінних функціонального елемента.

## 7.3 Обчислення булевих похідних для Q-синтезу тестів

Розглядається метод взяття булевих похідних по кубітному покриттю для створення умов активізації вхідних змінних при синтезі кубітних тестів. Проводиться аналогія між двома формами булевих функцій для взяття похідних: аналітичної та векторної. Дослідження методу виконується на прикладах логічних функцій:

1) $f(x) = x_1 \vee x_1\bar{x}_2$. 2) $f(x) = x_1x_2 \vee \bar{x}_1x_3$. 3) $f(x) = \bar{x}_2\bar{x}_3 \vee x_1x_2x_3$.

Питання, що підлягають вирішенню: 1) Визначення похідних першого порядку за аналітичною і кубітною формами завдання логічної функції. 2)

Верифікація отриманих умов активізації шляхом їх моделювання на одній з форм опису функціональності. 3) Синтез тестів активізації змінних логічної функції на основі обчислення похідних.

Приклад 7. Визначити всі похідні першого порядку по кубітній формі логічної функції $f(x) = x_1 \vee x_1\bar{x}_2$.

Застосування формули обчислення за аналітичним виразом

$$\frac{df(x_1,x_2,...,x_i,...,x_n)}{dx_i} = f(x_1,x_2,...,x_i = 0,...,x_n) \oplus f(x_1,x_2,...,x_i = 1,...,x_n)$$

визначає булеву похідну першого порядку як суму по модулю два нульової та одиничної залишкових функцій.

Для даної функції виходить:

$$\frac{df(x_1,x_2)}{dx_1} = f(0,x_2) \oplus f(1,x_2) =$$
$$= (0 \vee 0\bar{x}_2) \oplus (1 \vee 1\bar{x}_2) = 0 \oplus 1 = 1.$$

$$\frac{df(x_1,x_2)}{dx_2} = f(x_1,0) \oplus f(x_1,1) =$$
$$= (x_1 \vee x_1 \cdot \bar{0}) \oplus (x_1 \vee x_1 \cdot \bar{1}) =$$
$$= (x_1 \vee x_1 \cdot 1) \oplus (x_1 \vee x_1 \cdot 0) =$$
$$= (x_1 \vee x_1) \oplus (x_1 \vee 0) = x_1 \oplus x_1 = 0.$$

Нульове значення похідної означає відсутність умов активізації для змінної $x_2$, що дає підстави вважати її несуттєвою, отже, прибрати з числа змінних, які формують функціональність. Далі пропонуються аналогічні перетворення для кубітного покриття функції, заданого вектором:

| $X_1$ | $X_2$ | Y |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

$= (0011).$

Похідна по таблиці істинності може обчислюватися шляхом почергового завдання всіх нулів і одиниць в координатах стовпців, відповідних кожній змінній:

| $X_1$ | $X_2$ | Y | $Y_1^0$ | $Y_1^1$ | $Y_1'$ | $Y_2^0$ | $Y_2^1$ | $Y_2'$ |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 |
| 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 |

Таким чином, похідні за першою і другою змінною, записані в форматі кубітного покриття, дорівнюють: 1111 та 0000. Це означає, що похідна за першою змінною дорівнює 1, а за другою – дорівнює 0.

Однак такий результат можна отримати більш формально і технологічно, не розглядаючи вхідні набори таблиці істинності, використовуючи тільки логічні операції зустрічного зсуву і подальшої xor-операції над розрядами кубітного покриття {a,b}=a⊕b, де a,b – сусідні підвектори кубіта Q=(a,b):

| Y | $Y_1'$ | $Y_2'$ |
|---|---|---|
| 0 | 1 | 0 |
| 0 | 1 | 0 |
| 1 | 1 | 0 |
| 1 | 1 | 0 |

Інакше, для першої змінної необхідно xor-скласти, зсунуті один щодо одного дві половинки першого стовпчика, а результат записати в обидві зсувані симетричні області {a,b}=a⊕b, (11,11)=00⊕11. Для другої змінної слід розглядати пари сусідніх координат стовпчика а загальний результат записувати в кожні зсувані симетричні області-біти {a,b}=a⊕b: (0,0)=0⊕0=0, (0,0)=1⊕1=0. Таким чином, результат підсумовування буде загальним для кожної пари взаємодіючих підвекторів, розмірність яких визначається номером даної змінної, від 0 до $2^n - 1$.

Приклад 8. Визначити всі похідні першого порядку по аналітичній формі логічної функції $f(x) = x_1x_2 \vee \bar{x}_1x_3$. Для даної функції виконуються наступні обчислення:

$$\frac{df(x_1,x_2,x_3)}{dx_1} = f(0,x_2,x_3) \oplus f(1,x_2,x_3) =$$
$$= (0 \cdot x_2 \vee \bar{0} \cdot x_3) \oplus (1 \cdot x_2 \vee \bar{1} \cdot x_3) =$$
$$= (0 \vee 1 \cdot x_3) \oplus (x_2 \vee 0 \cdot x_3) =$$
$$= x_3 \oplus x_2 = x_2\bar{x}_3 \vee \bar{x}_2x_3.$$

$$\frac{df(x_1,x_2,x_3)}{dx_2} = f(x_1,0,x_3) \oplus f(x_1,1,x_3) =$$
$$= \bar{x}_1x_3 \oplus (x_1 \vee x_3) =$$
$$\overline{\bar{x}_1x_3}(x_1 \vee x_3) \vee \bar{x}_1x_3\overline{(x_1 \vee x_3)} =$$
$$= (x_1 \vee \bar{x}_3)(x_1 \vee x_3) \vee \bar{x}_1x_3\bar{x}_1\bar{x}_3 = x_1.$$

$$\frac{df(x_1,x_2,x_3)}{dx_3} = f(x_1,x_2,0) \oplus f(x_1,x_2,1) =$$
$$= x_1x_2 \oplus (\bar{x}_1 \vee x_2) =$$
$$\overline{x_1x_2}(\bar{x}_1 \vee x_2) \vee x_1x_2\overline{(\bar{x}_1 \vee x_2)} =$$
$$= (\bar{x}_1 \vee \bar{x}_2)(\bar{x}_1 \vee x_2) \vee x_1x_2x_1\bar{x}_2 = \bar{x}_1.$$

Для трьох змінних отримано 4 умови активізації, які відповідають чотирьом логічним шляхам в схемній структурі диз'юнктивної форми даної функції.

Обчислення трьох похідних першого порядку по таблиці істинності дає наступний результат:

| $X_1$ | $X_2$ | $X_3$ | Y | $Y_1^0$ | $Y_1^1$ | $Y_1'$ | $Y_2^0$ | $Y_2^1$ | $Y_2'$ | $Y_3^0$ | $Y_3^1$ | $Y_2'$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 |

Якщо виключити з розгляду таблицю істинності, а використовувати кубітне покриття, то на змістовному рівні процес обчислення похідних матиме такий вигляд:

| Y | $Y_1'$ | $Y_2'$ | $Y_3'$ |
|---|---|---|---|
| 0 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 |
| 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |

Пояснення. Для отримання похідної за першою змінною необхідно xor-скласти, зсунуті назустріч один одному, дві сусідні тетради кубіт-вектора Y, а результат записати в обидві тетради: {a,b}=a⊕b, (0110,0110)=0101⊕0011. Для отримання похідної за другою змінною слід послідовно xor-скласти сусідні пари кубіт-вектора Y а загальний результат записувати в кожну пару: {a,b}=a⊕b: (00,00)=01⊕01, (11,11)=00⊕11. Для отримання похідної по третій змінній слід послідовно xor-скласти сусідні біти кубіт-вектора Y а загальний результат записати в кожен біт сусідів: {a,b}=a⊕b: (1,1)=0⊕1, (1,1)=0⊕1, (0,0)=0⊕0, (0,0)=0⊕0.

Природно, що кубіт-похідна по будь-якій вхідній змінній, як вектор, має відносну симетрію рівності підвекторів за побудовою: похідна першої змінної має симетричну рівність двох тетрад, похідна другої змінної має симетричну рівність кожних сусідніх пар, похідна третьої змінної має симетричну рівність кожних сусідніх бітів.

Приклад 9. Визначити всі похідні першого порядку за таблицею істинності логічної функції трьох змінних:

$$f(x) = \bar{x}_2\bar{x}_3 \vee x_1x_2x_3.$$

Результат взяття похідних за таблицею істинності [187] наведеної функціональності на основі виконання операцій над стовпцями вхідних змінних представлена у вигляді:

$$\frac{df}{dx_1} =$$

| $x_1$ | $x_2$ | $x_3$ | Y | $Y_2^0$ | $Y_2^1$ | $Y_2^{\oplus}$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 | 1 | 1 |

$= x_2x_3.$

$$\frac{df}{dx_2} =$$

| $x_1$ | $x_2$ | $x_3$ | Y | $Y_2^0$ | $Y_2^1$ | $Y_2^{\oplus}$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 0 | 1 | 1 |

$= \bar{x}_1\bar{x}_3 \vee x_1\bar{x}_3 \vee x_1x_3 = \bar{x}_3 \vee x_1x_3.$

$$\frac{df}{dx_3} =$$

| $x_1$ | $x_2$ | $x_3$ | Y | $Y_3^0$ | $Y_3^1$ | $Y_3^{\oplus}$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 0 | 1 | 1 |
| 1 | 1 | 1 | 1 | 0 | 1 | 1 |

$= \bar{x}_1\bar{x}_2 \vee x_1\bar{x}_2 \vee x_1x_2 = \bar{x}_2 \vee x_1x_2.$

Як альтернатива, нижче наведено результати виконання процедур взяття похідних першого порядку для трьох змінних по кубітному покриттю функціональності:

| $X_1$ | $X_2$ | $X_3$ | Y | $Y_1^0$ | $Y_1^1$ | $Y_1'$ | $Y_2^0$ | $Y_2^1$ | $Y_2'$ | $Y_3^0$ | $Y_3^1$ | $Y_3'$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 |

Далі представлений процес обчислення похідних на кубітному покритті без розгляду таблиці істинності, що на формальному рівні має такий вигляд:

| Y | $Y_1'$ | $Y_2'$ | $Y_3'$ |
|---|---|---|---|
| 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 |
| 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 1 |
| 1 | 1 | 1 | 1 |

Інтерпретація отриманих кубіт-похідних. Природно, що похідні є функції, задані векторами. Вони можуть бути записані в аналітичній формі (ДНФ) за одиничними значеннями змінних, які формують адреси комірок кубіт-вектора:

$$Y_1' = 011 \vee 111 = \overline{X}_1 X_2 X_3 \vee X_1 X_2 X_3 = X_2 X_3 (\overline{X}_1 \vee X_1) = X_2 X_3.$$

$$Y_2' = 000 \vee 010 \vee 100 \vee 101 \vee 110 \vee 111 = \overline{X}_1 \overline{X}_3 \vee X_1.$$

$$Y_3' = 000 \vee 001 \vee 100 \vee 101 \vee 110 \vee 111 = \overline{X}_1 \overline{X}_2 \vee X_1.$$

Мінімізація булевих функцій, відповідних похідним, приводить до аналітичних виразів, де відсутні змінні, за якими береться похідна. Таким чином, всі результати обчислення похідних від трьох форм (аналітична, таблична, векторна) завдання функції є ідентичними. Найбільш технологічним є метод взяття похідної по кубітному покриттю. Метод має меншу обчислювальну складність в силу компактного представлення функціональності. Використання аналітичної форми передбачає істотне підвищення складності алгоритмів, пов'язаної із застосуванням законів

булевої алгебри і мінімізації функцій, що обмежує її застосування для вирішення практичних завдань.

Для порівняння далі коротко описаний метод отримання тесту $T = [T_{ij}], i = \overline{1,k}; j = \overline{1,n}$ комбінаційної функціональності, заданої кубічним покриттям або таблицею істинності, який містить пункти [205]:

1) $f'(x_i) = f(x_1, x_2, ..., x_i = 0, ..., x_n) \oplus f(x_1, x_{2,...,} x_i = 1, ..., x_n)$;

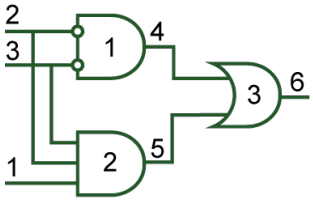
2) $T = \bigcup_{i=1}^{n} [f'(x_i) * (x_i = 0) \vee (x_i = 1)]$;

3) $T_{ij} = T_{i-1,j} \leftarrow T_{ij} = X;\ T_{1j} = 1 \leftarrow T_{1j} = X;$

4) $T = T \setminus T_i \leftarrow T_i = T_{i-r},\ r = \overline{1, i-1},\ i = \overline{2, n}.$

1) Обчислення похідних за всіма n змінними функціональності шляхом використання кубічного покриття. 2) Об'єднання всіх умов (векторів) активізації в таблицю, де кожному вектору шляхом конкатенації (*) ставиться у відповідність зміна змінної, за якою була взята похідна, що означає подвоєння кількості тестових наборів відносно загальної кількості (k) умов активізації. 3) Мінімізація тестових векторів шляхом видалення повторюваних вхідних послідовностей. Рис. 7.7 ілюструє таблиці ходу отримання тесту відповідно до пунктів 2-3 алгоритму для функціональності $f(x) = \bar{x}_2\bar{x}_3 \vee x_1x_2x_3$, представленої схемної структурою.

| $x_1$ | $x_2$ | $x_3$ | Y |
|---|---|---|---|
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 → |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 |




| $x_1$ | $x_2$ | $x_3$ | Y |
|---|---|---|---|
| 0 | 1 | 1 | 0 |
| 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 |

Рисунок 7.7 – Отримання тесту для схемної структури булевої функції

Як альтернатива описаному вище, далі пропонується технологічно простий метод синтезу тестів на основі взяття похідних по кубітним покриттям функціональних елементів без розгляду станів вхідних змінних.

1) Початкове завдання логічної функціональності кубітним покриттям.

2) Виконання операцій зустрічного зсуву частин кубіт-вектора і подальшого покоординатного xor-підсумовування для отримання векторів похідних для кожної вхідної змінної.

3) Логічне об'єднання векторів похідних, що формує тест-вектор, рівний за розміром кубітному покриттю.

4) У разі необхідності отримання мінімального тесту вирішується завдання покриття (вже на матриці кубіт-похідних) шляхом знаходження мінімальної кількості пар одиничних координат кубіт-вектора всіх змінних, де пара одиниць повинна перевіряти одиночні константні несправності кожного входу. Процедура вибору пари одиниць в кубіт-похідній визначається наявністю двох представників, по одному від будь-якої парної і будь-якої непарної частин вектора.

Далі в таблиці представлено результати синтезу тесту для логічної функції від трьох змінних, заданої рівнянням: $f(x) = \bar{x}_2\bar{x}_3 \vee x_1x_2x_3$, що має кубітне покриття (10001001),

| $X_1$ | $X_2$ | $X_3$ | Y | $Y_1'$ | $Y_2'$ | $Y_3'$ | T |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| $X_1$ | $X_2$ | $X_3$ | Y |
|---|---|---|---|
| 0 | 1 | 1 | 0 |
| 1 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 |

З урахуванням виконання пункту 3 алгоритм синтезу дає мінімальний тест перевірки вхідних змінних, який містить 5 наборів, що представлено стовпцем T, а також продубльовано в явному вигляді правою таблицею.

Інтерес представляє той факт, що результат виконання процедури взяття похідної по кубіт-вектору вже містить тест активізації кожної змінної. Об'єднаний тест перевіряє всі константні несправності вхідних змінних, а також може бути використаний для діагностування несправностей, оскільки для істотних входів всі похідні-вектори будуть різними. Фактично взяття похідної по змінній на кубіт-покритті формує Q-тест, не більше і не менше.

7.4 Дедуктивний аналіз несправностей цифрових структур

Використовується для визначення якості тесту щодо введеного класу несправностей, як правило, одиночних константних. Існує розвинена теорія дедуктивного аналізу [187], орієнтована на паралельну обробку списків несправностей. Базові поняття теорії моделювання несправностей представлені апаратом транспортування списків дефектів через примітивні функціональні логічні елементи [182, 183, 188, 189-192]. Нижче визначено дедуктивні функції паралельного моделювання несправностей на вичерпному тесті для функціональних елементів and, or, not. Отримання дедуктивного перетворювача для функції and:

$$L[T = (00,01,10,11), F = (X_1 \wedge X_2)] =$$
$$= L\{(\bar{x}_1\bar{x}_2 \vee \bar{x}_1 x_2 \vee x_1\bar{x}_2 \vee x_1x_2) \wedge [(X_1 \oplus T_{t1} \wedge X_2 \oplus T_{t2}) \oplus T_{t3})]\} =$$
$$= (\bar{x}_1\bar{x}_2)\{[(X_1 \oplus 0) \wedge (X_2 \oplus 0)] \oplus 0\} \vee (\bar{x}_1x_2)\{[(X_1 \oplus 0) \wedge (X_2 \oplus 1)] \oplus 0\} \vee$$
$$\vee (x_1\bar{x}_2)\{[(X_1 \oplus 1) \wedge (X_2 \oplus 0)] \oplus 0\} \vee (x_1x_2)\{[(X_1 \oplus 1) \wedge (X_2 \oplus 1)] \oplus 1\} =$$
$$= (\bar{x}_1\bar{x}_2)(X_1 \wedge X_2) \vee (\bar{x}_1x_2)(X_1 \wedge \bar{X}_2) \vee (x_1\bar{x}_2)(\bar{X}_1 \wedge X_2) \vee (x_1x_2)(X_1 \vee X_2).$$

Аналогічно виконуються обчислення для функції or:

$$L[T = (00,01,10,11), F = (X_1 \vee X_2)] =$$
$$= L\{(\bar{x}_1\bar{x}_2 \vee \bar{x}_1 x_2 \vee x_1\bar{x}_2 \vee x_1x_2) \wedge [(X_1 \oplus T_{t1} \vee X_2 \oplus T_{t2}) \oplus T_{t3})]\} =$$
$$= (\bar{x}_1\bar{x}_2)\{[(X_1 \oplus 0) \vee (X_2 \oplus 0)] \oplus 0\} \vee (\bar{x}_1x_2)\{[(X_1 \oplus 0) \vee (X_2 \oplus 1)] \oplus 1\} \vee$$
$$\vee (x_1\bar{x}_2)\{[(X_1 \oplus 1) \vee (X_2 \oplus 0)] \oplus 1\} \vee (x_1x_2)\{[(X_1 \oplus 1) \vee (X_2 \oplus 1)] \oplus 1\} =$$
$$= (\bar{x}_1\bar{x}_2)(X_1 \vee X_2) \vee (\bar{x}_1x_2)(\bar{X}_1 \wedge X_2) \vee (x_1\bar{x}_2)(X_1 \wedge \bar{X}_2) \vee (x_1x_2)(X_1 \wedge X_2).$$

Тут $T_t = (T_{t1}, T_{t2}, T_{t3}), (t = \overline{1,4})$ – тест-вектор, який має 3 координати, причому остання з них визначає стан виходу двовходового елемента and (or);L – вихідний список несправностей; X – список несправностей на конкретному вході примітиву; x={0,1} – логічне значення на вході

примітиву. У наступному перетворенні $T_t = (T_{t1}, T_{t2}), (t = \overline{1,2})$– тест-вектор, який має 2 координати, де друга – стан виходу інвертора:

$$L[T = (0,1), F = \overline{X_1}] = L\{(\bar{x}_1 \vee x_1)[(\overline{X_1 \oplus T_{t1}}) \oplus T_{t2}]\} =$$

$$= \bar{x}_1[(\overline{X_1 \oplus 0}) \oplus 1] \vee x_1[(\overline{X_1 \oplus 1}) \oplus 0] = \bar{x}_1\bar{\bar{X}}_1 \vee x_1\bar{\bar{X}}_1 = \bar{x}_1 X_1 \vee x_1 X_1.$$

Останній вираз ілюструє инвариантность інверсії до вхідного набору для транспортування дефектів. Вона трансформується в повторювач. Тому дана функція не фігурує на виходах дедуктивних елементів. Спільна апаратурна реалізація дедуктивних функцій для решти двовходових елементів and, or на вичерпному тесті представлена універсальною схемою (рис. 7.8) дедуктивно-паралельного аналізу несправностей.

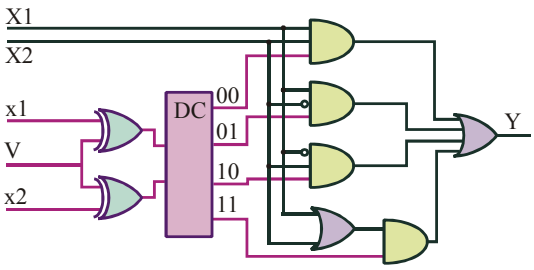



Рисунок 7.8 – Симулятор несправностей примітивів

У симуляторі представлено булеві (x1,x2) і регістрові (X1,X2) для кодування несправностей входи, змінна V вибору типу справної функції (and, or), вихідна регістрова змінна Y. Стани двійкових входів x1, x2 і змінна вибору елемента визначають одну з чотирьох дедуктивних функцій для отримання вектора Y перевіряються несправностей.

Далі пропонується технологічна реалізація дедуктивного моделювання на кубітній формі завдання функціональностей, яка відрізняється від наведеної вище паралелізмом виконання логічних операцій, а також можливістю застосування методу для будь-яких цифрових структур.

Сукупність кубіт-похідних для всіх вхідних змінних, обчислених по кубітному покриттю, являє собою кубітную матрицю для реалізації дедуктивного методу моделювання несправностей. Рядок матриці формує умови для транспортування списків несправностей від зовнішніх входів до виходу за правилом: одиничні значення створюють об'єднання вхідних

списків, а нульові сигнали вказують на входи, списки яких повинні бути відняті з результату об'єднання. Наявність всіх нульових сигналів в рядку створює умови перетину вхідних списків між собою.

Як приклад пропонується побудова дедуктивних формул транспортування списків несправностей від вхідних змінних до виходу функціональності, заданої вхідними наборами, кубітним покриттям з векторними похідними:

| $x_1$ | $x_2$ | $x_3$ | Y | $X_1$ | $X_2$ | $X_3$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 |

У загальному випадку формула дедуктивного моделювання логічних функціональностей, представлених у вигляді кубітних векторів, має такий вигляд:

$$L = \vee_{i=1}^{2^n} (x_{i1} \wedge x_{i2} \wedge \ldots \wedge x_{ij} \wedge \ldots \wedge x_{in}) \wedge (X_{i1} \vee X_{i2} \vee \ldots \vee X_{ij} \vee \ldots \vee X_{in}),$$

$$L = \vee_{i=1}^{2^n} (x_{i1} \wedge \ldots \wedge x_{in}) \wedge (X_{i1} \vee \ldots \vee X_{in}),$$

$$L = x \wedge X, x = x_{ij}, X = X_{ij}, i = \overline{1, 2^n}, j = \overline{1, n}.$$

Тут L – список вихідних несправностей; x – матриця вхідних тестових наборів; X – матриця кубітних похідних від кубітного покриття; n – кількість вхідних змінних.

Алгоритм побудови дедуктивної формули для заданої функціональності містить наступні пункти:

1) Завдання кубіт-вектора функціональності.

2) Обчислення кубіт-похідних для вхідних змінних з метою отримання відповідної матриці.

3) Формування аналітичної або матрично-векторної форми обчислення вихідних списків несправностей шляхом логічного множення матриць вхідних тестових впливів і матриці похідних.

Нижче наведено процес обчислення аналітичної і векторної форм для дедуктивного моделювання несправностей логічної функціональності:

$$T = (000,001,010,011,100,101,110,111).$$

$$Q = (011,001,010,100,011,011,011,111)].$$

$$L = (000 \wedge 011) \vee (001 \wedge 001) \vee (010 \wedge 010) \vee (011 \wedge 100) \vee$$
$$\vee (100 \wedge 011) \vee (101 \wedge 011) \vee (110 \wedge 011) \vee (111 \wedge 111).$$

$$L = (\bar{x}_1\bar{x}_2\bar{x}_3 \wedge \overline{X}_1X_2X_3) \vee (\bar{x}_1\bar{x}_2x_3 \wedge \overline{X}_1\overline{X}_2X_3) \vee (\bar{x}_1x_2\bar{x}_3 \wedge \overline{X}_1X_2\overline{X}_3) \vee$$
$$\vee (\bar{x}_1x_2x_3 \wedge X_1\overline{X}_2\overline{X}_3) \vee (x_1\bar{x}_2\bar{x}_3 \wedge \overline{X}_1X_2X_3) \vee (x_1\bar{x}_2x_3 \wedge \overline{X}_1X_2X_3) \vee$$
$$\vee (x_1x_2\bar{x}_3 \wedge \overline{X}_1X_2X_3) \vee (x_1x_2x_3 \wedge X_1X_2X_3).$$

Для порівняння технологічної складності запропонованого методу нижче представлена процедура отримання дедуктивних формул моделювання несправностей [187] на основі аналітичного завдання функциональностей:

$$L = \bar{x}_1\bar{x}_2\bar{x}_3\{[(\overline{X}_2 \oplus 0)(\overline{X}_3 \oplus 0) \vee (X_1 \oplus 0)(X_2 \oplus 0)(X_3 \oplus 0)] \oplus 1\}$$
$$\vee \bar{x}_1\bar{x}_2x_3\{[(\overline{X}_2 \oplus 0)(\overline{X}_3 \oplus 1) \vee (X_1 \oplus 0)(X_2 \oplus 0)(X_3 \oplus 1)] \oplus 0\}$$
$$\vee \bar{x}_1x_2\bar{x}_3\{[(\overline{X}_2 \oplus 1)(\overline{X}_3 \oplus 0) \vee (X_1 \oplus 0)(X_2 \oplus 1)(X_3 \oplus 0)] \oplus 0\}$$
$$\vee \bar{x}_1x_2x_3\{[(\overline{X}_2 \oplus 1)(\overline{X}_3 \oplus 1) \vee (X_1 \oplus 0)(X_2 \oplus 1)(X_3 \oplus 1)] \oplus 0\}$$
$$\vee x_1\bar{x}_2\bar{x}_3\{[(\overline{X}_2 \oplus 0)(\overline{X}_3 \oplus 0) \vee (X_1 \oplus 1)(X_2 \oplus 0)(X_3 \oplus 0)] \oplus 1\}$$
$$\vee x_1\bar{x}_2x_3\{[(\overline{X}_2 \oplus 0)(\overline{X}_3 \oplus 1) \vee (X_1 \oplus 1)(X_2 \oplus 0)(X_3 \oplus 1)] \oplus 0\}$$
$$\vee x_1x_2\bar{x}_3\{[(\overline{X}_2 \oplus 1)(\overline{X}_3 \oplus 0) \vee (X_1 \oplus 1)(X_2 \oplus 1)(X_3 \oplus 0)] \oplus 0\}$$
$$\vee x_1x_2x_3\{[(\overline{X}_2 \oplus 1)(\overline{X}_3 \oplus 1) \vee (X_1 \oplus 1)(X_2 \oplus 1)(X_3 \oplus 1)] \oplus 1\}.$$

$$L = \bar{x}_1\bar{x}_2\bar{x}_3(\overline{\bar{X}_2\bar{X}_3 \vee X_1X_2X_3}) \vee \bar{x}_1\bar{x}_2x_3(\bar{X}_2X_3 \vee X_1X_2\bar{X}_3) \vee \bar{x}_1x_2\bar{x}_3(X_2\bar{X}_3 \vee X_1\bar{X}_2X_3) \vee \bar{x}_1x_2x_3(X_2X_3 \vee X_1\bar{X}_2\bar{X}_3) \vee x_1\bar{x}_2\bar{x}_3(\overline{\bar{X}_2\bar{X}_3 \vee \bar{X}_1X_2X_3}) \vee x_1\bar{x}_2x_3(\bar{X}_2X_3 \vee \bar{X}_1X_2\bar{X}_3) \vee x_1x_2\bar{x}_3(X_2\bar{X}_3 \vee \bar{X}_1\bar{X}_2X_3) \vee x_1x_2x_3(\overline{\bar{X}_2X_3 \vee \bar{X}_1\bar{X}_2\bar{X}_3}).$$

$$L = \bar{x}_1\bar{x}_2\bar{x}_3(\overline{\bar{X}_2\bar{X}_3} \wedge \overline{X_1X_2X_3}) \vee \bar{x}_1\bar{x}_2x_3(\bar{X}_2X_3 \vee X_1X_2\bar{X}_3) \vee \bar{x}_1x_2\bar{x}_3(X_2\bar{X}_3 \vee X_1\bar{X}_2X_3) \vee \bar{x}_1x_2x_3(X_2X_3 \vee X_1\bar{X}_2\bar{X}_3) \vee x_1\bar{x}_2\bar{x}_3(\overline{\bar{X}_2\bar{X}_3} \wedge \overline{\bar{X}_1X_2X_3}) \vee x_1\bar{x}_2x_3(\bar{X}_2X_3 \vee \bar{X}_1X_2\bar{X}_3) \vee x_1x_2\bar{x}_3(X_2\bar{X}_3 \vee \bar{X}_1\bar{X}_2X_3) \vee x_1x_2x_3(\overline{X_2X_3} \wedge \overline{\bar{X}_1\bar{X}_2\bar{X}_3}).$$

$$L = \bar{x}_1\bar{x}_2\bar{x}_3[(X_2 \vee X_3) \wedge (\bar{X}_1 \vee \bar{X}_2 \vee \bar{X}_3)] \vee \bar{x}_1\bar{x}_2x_3(\bar{X}_2X_3 \vee X_1X_2\bar{X}_3) \vee \bar{x}_1x_2\bar{x}_3(X_2\bar{X}_3 \vee X_1\bar{X}_2X_3) \vee \bar{x}_1x_2x_3(X_2X_3 \vee X_1\bar{X}_2\bar{X}_3) \vee x_1\bar{x}_2\bar{x}_3[(X_2 \vee X_3) \wedge (X_1 \vee \bar{X}_2 \vee \bar{X}_3)] \vee x_1\bar{x}_2x_3(\bar{X}_2X_3 \vee \bar{X}_1X_2\bar{X}_3) \vee x_1x_2\bar{x}_3(X_2\bar{X}_3 \vee \bar{X}_1\bar{X}_2X_3) \vee x_1x_2x_3[(\bar{X}_2 \vee \bar{X}_3) \wedge (X_1 \vee X_2 \vee X_3)].$$

$$L = \bar{x}_1\bar{x}_2\bar{x}_3(\bar{X}_1X_2 \vee \bar{X}_1X_3 \vee \bar{X}_2X_3 \vee X_2\bar{X}_3) \vee \bar{x}_1\bar{x}_2x_3(\bar{X}_2X_3 \vee X_1X_2\bar{X}_3) \vee \bar{x}_1x_2\bar{x}_3(X_2\bar{X}_3 \vee X_1\bar{X}_2X_3) \vee \bar{x}_1x_2x_3(X_2X_3 \vee X_1\bar{X}_2\bar{X}_3) \vee x_1\bar{x}_2\bar{x}_3(X_1X_2 \vee X_1X_3 \vee \bar{X}_2X_3 \vee X_2\bar{X}_3) \vee x_1\bar{x}_2x_3(\bar{X}_2X_3 \vee \bar{X}_1X_2\bar{X}_3) \vee x_1x_2\bar{x}_3(X_2\bar{X}_3 \vee \bar{X}_1\bar{X}_2X_3) \vee x_1x_2x_3(X_1\bar{X}_2 \vee X_1\bar{X}_3 \vee X_2\bar{X}_3 \vee \bar{X}_2X_3).$$

Обчислення кубітних форм для дедуктивного моделювання несправностей основних логічних примітивів: or, and, xor представлено в таблиці:

| $x_1$ | $x_2$ | $Y^{\vee}$ | $X_1^{\vee}$ | $X_2^{\vee}$ | $Y^{\wedge}$ | $X_1^{\wedge}$ | $X_2^{\wedge}$ | $Y^{\oplus}$ | $X_1^{\oplus}$ | $X_2^{\oplus}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 |

Тут визначено два вектор-стовпці табличного завдання вхідних змінних $(x_1, x_2)$, кубіт-вектор функції or – $Y^{\vee}$, дві похідні $(X_1^{\vee}, X_2^{\vee})$ для кожної вхідної змінної. Далі показано: кубіт-вектор завдання функції and і дві колонки похідних, а також кубіт-вектор функції xor і два стовпці похідних по вхідним змінним. Формули дедуктивного моделювання тривіально записуються по рядках таблиці:

$$L^{\vee} = \bar{x}_1\bar{x}_2(X_1 \vee X_2) \vee \bar{x}_1 x_2(\overline{X}_1 \vee X_2) \vee x_1\bar{x}_2(X_1 \vee \overline{X}_2) \vee x_1 x_2(\overline{X}_1 \vee \overline{X}_2)$$

$$L^{\wedge} = \bar{x}_1\bar{x}_2(\overline{X}_1 \vee \overline{X}_2) \vee \bar{x}_1 x_2(X_1 \vee \overline{X}_2) \vee x_1\bar{x}_2(\overline{X}_1 \vee X_2) \vee x_1 x_2(X_1 \vee X_2)$$

$$L^{\oplus} = \bar{x}_1\bar{x}_2(X_1 \vee X_2) \vee \bar{x}_1 x_2(X_1 \vee X_2) \vee x_1\bar{x}_2(X_1 \vee X_2) \vee x_1 x_2(X_1 \vee X_2)$$
$$= (X_1 \vee X_2)$$

Тут вхідні змінні $x_i$ з'єднуються між собою знаком кон'юнкції, а похідні – змінні транспортування списків вхідних несправностей, позначені символом $X_i$, об'єднуються між собою знаком диз'юнкції, відповідно до станів координат стовпців-похідних.

Таким чином, запропонована технологія моделювання несправностей, що базується на використанні векторних кубітних форм завдання функциональностей і похідних, не має аналогів за доступністю розуміння, простотою реалізації і швидкодії. Щоб отримати дедуктивні формули транспортування дефектів будь-якої функціональності, необхідно і достатньо отримати вектор-стовпці похідних всіх вхідних змінних за допомогою операцій зсуву і xor-підсумовування. Обчислювальна складність даних операцій залежить від апаратних витрат і в межі може бути зведена до лінійної залежності від кількості вхідних змінних.

На рис. 7.9 представлений процесор кубітного моделювання цифрових пристроїв, що містить структури: справного інтерпретативного моделювання, дедуктивного аналізу несправностей, призначеного для оцінки якості тесту і

побудови таблиці несправностей, а також модулів тестування і діагностування дефектів на стадіях проектування і експлуатації. Основна відмінність від існуючих рішень полягає у використанні Q-покриття, представленого у формі вектора станів функціональності, що дає можливість істотно підвищити швидкодію моделювання за рахунок виконання паралельних регістрових операцій.

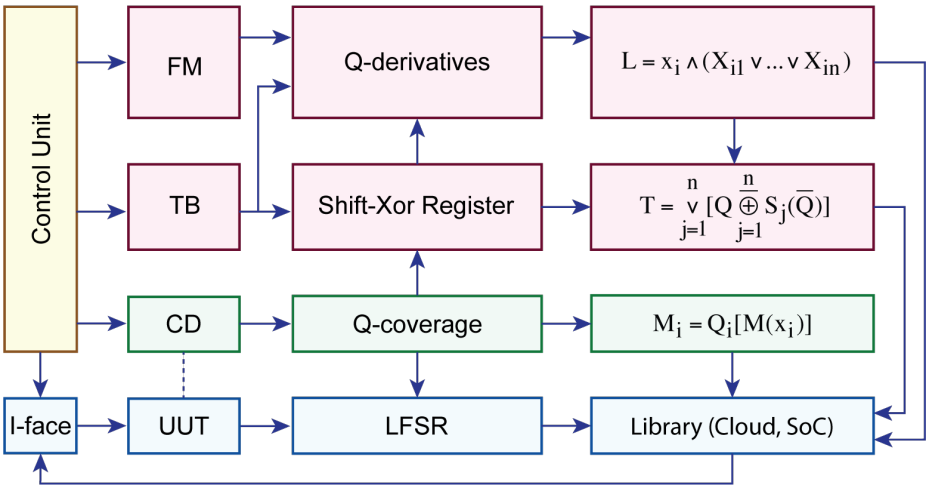



Рисунок 7.9 – Процесор кубітного моделювання цифрових пристроїв

Control Unit – пристрій управління симулятором, який синхронізує роботу блоків справного моделювання та структурних компонентів дедуктивного аналізу несправностей. FM – Fault Matrix, матриця вхідних несправностей розглянутої функціональності цифрового пристрою. TB – Test Bench, упорядкована сукупність вхідних перевіряльних послідовностей, де поточний вхідний набір ідентифікується як $x_i$. CD – Circuit Description – схемний опис цифрового пристрою, де функціональні елементи представлені кубітнимі покриттями Q-coverage. Обробка останніх здійснюється блоком справного моделювання $M_i = Q_i[M(x_i)]$, який реалізує адресні транзакції між кубітним покриттям і вектором моделювання M. Результати справного моделювання вхідних наборів формують матрицю GM (Good simulation Matrix), яка записується в Library. Блок Shift-Xor Register формує матрицю

похідних у кубітній формі (Q-derivatives), застосовуючи регістрові операції зсуву і xor. L-блок для створення вихідного списку несправностей використовує формулу аналізу вхідного набору і рядки матриці похідних $L = x_i \wedge (X_{i1} \vee \ldots \vee X_{in})$.

Результати дедуктивного аналізу формують списки несправностей, що відповідають вхідним наборам, які об'єднуються в DM – Fault Detected Matrix і заносяться в Library. Крім того, T-модуль формує Q-тест і оцінює його якість у метриці одиночних константних несправностей зовнішніх входів і виходів функціональних елементів, які заносяться в Library. Тести для функциональностей разом з матрицями несправностей формують бібліотеку Library={Signature, Q-coverage, Q-test, Quality, DM, GM}, яка може багаторазово використовуватися як хмарний або вбудований в SoC сервіс для тестування і/або діагностування функциональностей UUT (Unit Under Test) на основі використання інтерфейсу I-face, що підтримує стандарти IEEE 1500 SECT, IP (Internet Protocol). Пошук тестових сервісів в бібліотеці здійснюється за допомогою кубітного вектора, попередньо згорнутого в 16-розрядний двійковий код – сигнатуру (Sign), на основі регістра зсуву з лінійними зворотними зв'язками (LFSR), що дає можливість структурувати бібліотеку для швидкого отримання інформації та проведення тестування в режимі online. Блок LFSR виконує дві функції стиснення в 16-ковий код-сигнатуру: Q-покриття і реакції UUT на вхідні тестові впливи для подальшого діагностування місця, причини і виду дефекту.

## 7.5 Висновки до розділу 7

Нижче наведено формулювання наукової новизни і практичної значущості описаних досліджень.

1) Удосконалено векторні моделі кубітного представлення структур і компонентів цифрових систем, що базуються на адресному кодуванні вхідних сигналів і відрізняються технологічністю та швидкодією побудованих на їх основі процедур синтезу й аналізу.

2) Удосконалено методи синтезу тестів і моделювання несправностей, які відрізняються паралельним виконанням логічних регістрових операцій над кубітними покриттями схемних компонентів.

3) Вперше розроблено метод і секвенсор безумовного синтезу тестів для функціональних логічних компонентів, який характеризується паралельним виконанням регістрових логічних операцій (shift, or, not, nxor) над кубітним вектором і його похідними, що дає можливість істотно зменшити час генерування вхідних наборів і тестування пристрою в режимі embedded online.

2) Вперше розроблено метод взяття похідних для генерації тестів функціональних компонентів, який характеризується паралельним виконанням регістрових логічних операцій (shift, or, not, nxor) над кубітним вектором, що дає можливість істотно зменшити час генерування вхідних наборів і тестування пристрою за рахунок апаратної надлишковості.

3) Вперше розроблено дедуктивний метод моделювання несправностей для функціональних компонентів, який характеризується паралельним виконанням регістрових логічних операцій (shift, or, not, nxor) над кубітним вектором і його похідними, що дає можливість істотно зменшити час верифікації та тестування цифрового пристрою в режимі embedded online.

4) Запропоновано процесор кубітного моделювання цифрових пристроїв, імплементований в SoC або Cloud Service, для аналізу справної поведінки і несправностей на основі використання кубітних покриттів функціональних елементів, який відрізняється від відомих реалізацій застосуванням мінімального набору регістрових логічних операцій і високою швидкодією.

5) Практична значущість досліджень полягає в можливості хмарної реалізації швидкодіючого методу синтезу тестів і моделювання несправностей для функціональних логічних компонентів на основі паралельного виконання регістрових логічних операцій (shift, or, not, nxor) над кубітним вектором і його похідними, що дає можливість генерувати

вхідні набори і оцінювати їх якість в режимі online. Крім того, хмарний мікросервіс синтезу тестів і моделювання дефектів для функціональних логічних компонентів може бути затребуваний для навчальних і наукових цілей в процесах синтезу та аналізу цифрових архітектур.

6) Запропонований метод синтезу тестів для функциональностей на основі кубітного покриття може бути використаний в якості вбудованого BIST-компонента для сервісного обслуговування SoC на основі стандарту граничного сканування IEEE 1500 SECT або в якості хмарного online сервісу тестування апаратних модулів за допомогою IP-протоколу.

7) Подальші дослідження в цій галузі будуть спрямовані на створення програмно-апаратних генераторів тестів, симуляторів несправностей, справної поведінки, алгоритмів діагностування та бібліотечних рішень, вбудованих в інфраструктуру кристалів і/або хмарні сервіси, що використовують кубітний опис функціональності логічного компонента.

# 8 СИНТЕЗ І АНАЛІЗ КУБІТНИХ МОДЕЛЕЙ ЦИФРОВИХ СИСТЕМ

Розробка кубітних структур даних і «квантових» матричних процесорів (обчислювачів) на основі технологій Big Data для одночасного і паралельного сервісного обслуговування користувачів TCS-хмари в реальному масштабі часу є однією з складових задач. Пропонується хмарний сервіс QuaSim для моделювання і верифікації цифрових систем, що базується на транзакціях між адресовними компонентами пам'яті для реалізації будь-якої функціональності. Описано новий підхід до синтезу та аналізу цифрових систем, що використовує векторну форму (квант) завдання комбінаційних і послідовностних структур для їх імплементації в елементи пам'яті, що істотно відрізняється від загальноприйнятої теорії проектування дискретних пристроїв на основі таблиць істинності компонентів. Використовуються квантові або кубітні структури даних [193, 194-197] для реалізації обчислювальних процесів з метою підвищення швидкодії аналізу цифрових систем і зменшення обсягів пам'яті на основі унарного кодування станів вхідних, внутрішніх і вихідних змінних та імплементації кубітних векторів в елементи пам'яті FPGA, що реалізують комбінаційні і послідовностні примітиви. Впровадження квантових memory-based-only моделей опису цифрових компонентів в практику проектування обчислювальних систем безпосередньо впливає на збільшення виходу придатної продукції, підвищення надійності комп'ютерних виробів, зниження вартості проектування і виготовлення, а також забезпечує автономне відновлення працездатності в режимі remote & online без участі людини.

## 8.1 Вступ

Мета – істотне поліпшення якості та надійності обчислювальних пристроїв за рахунок адресовних схемних елементів, що дозволяє виконувати online ремонт, а також підвищення швидкодії методів моделювання, тестування і верифікації складних цифрових систем, завдяки зменшенню

розмірності моделей функціональних примітивів та адресної реалізації всіх компонентів структур даних.

Задачі: 1) Розробка автоматної моделі квантового процесора. 2) Синтез кубітних моделей цифрових примітивів: логіка, тригери, регістри і лічильники. 3) Синтез і аналіз квантових моделей цифрових схем. 4) Моделювання цифрових пристроїв на основі використання квантових векторів опису примітивів.

Мотивація і стан питання: 1) Сучасна система на кристалі містить 94% пам'яті і лише 6% логіки, яка доставляє більше 90% проблем, пов'язаних з верифікацією, тестуванням, діагностуванням та відновленням працездатності [179, 198]. Звичайно, швидкодія логічних схем на порядок вище, ніж у пам'яті, проте велика частка обчислювальних процесів припадає на обмін інформацією в структурах пам'яті. Тому переваги комбінаційної логіки в реальних обчислювальних системах обробки великих даних компенсуються великими часовими витратами (близько 90%), пов'язаними з транзакціями в пам'яті. 2) Реалізація процесора тільки на основі використання елементів пам'яті робить його однорідним за структурою і типам функціональних примітивів, що доставляє очевидні технологічні зручності процесам проектування, виробництва і експлуатації, у тому числі верифікації, вбудованому тестуванню і діагностуванню, а головне – ремонту в режимі online за рахунок використання на кристалі універсальних адресовних spare-компонентів пам'яті [179, 198]. 3) Моделювання в процесі верифікації проектованих обчислювачів на основі адресних моделей компонентів робить дану процедуру технологічно простою за рахунок регулярних структур даних і використання єдиної операції транзакції на елементах пам'яті, а також більш швидкодіючою за рахунок можливості паралельної квантоподібної обробки великих масивів однотипної пам'яті [193, 196, 197, 199, 201, 202]. 4) Енергоспоживання при заміні логіки на елементи пам'яті зростає на кілька відсотків, що насправді буде платою за перераховані вище істотні переваги, пов'язані, в кінцевому рахунку, зі збільшенням виходу придатної продукції,

підвищенням надійності обчислювальних виробів, зниженням вартості проектування і виготовлення, а також автономним відновленням працездатності в режимі remote & online, без участі людини. Однак енергозберігаючі рішення з реалізації обчислювальних процесів на пам'яті [199, 200, 206, 207] дають підстави вважати, що такого програшу взагалі не буде.

## 8.2 Квантові або кубітні структури даних

На ринку електронних технологій існує конкуренція між базами імплементації ідеї [194 – 197, 203]: 1) Гнучка (м'яка) реалізація проекту пов'язана з синтезом інтерпретативної моделі програмної форми функціональності або в апаратному виконанні програмовних логічних пристроїв на основі FPGA, CPLD; переваги полягають у технологічності модифікації проекту, недоліки – у невеликій швидкодії функціонування цифрової системи. 2) Жорстка реалізація орієнтована на використання компілятивних моделей при розробці програмних додатків або на імплементацію проекту в кристали VLSI [179, 193, 198, 206, 207]. Переваги та недоліки жорсткої реалізації є інверсними по відношенню до м'якого виконання проектів: висока швидкодія і неможливість модифікації. З урахуванням викладених базових варіантів реалізації ідеї пропонуються квантові структури даних, орієнтовані на підвищення швидкодії гнучких моделей програмного або апаратного виконання проекту, а також на можливість online ремонту в процесі експлуатації.

Квантові структури опису цифрових систем. Кубіт (n-кубіт) є векторною формою унітарного (унарного) кодування універсуму з n примітивів для завдання булеана станів $2^{2^n}$ за допомогою $2^n$ двійкових змінних. Наприклад, якщо n = 2, то 2-кубіт задає 16 станів за допомогою чотирьох змінних. Якщо n=1, то кубіт задає чотири стани на универсумі з двох примітивів (10) і (01) за допомогою двох двійкових змінних (00,01,10,11) [196, 203]. При цьому допускається суперпозиція (одночасне

існування) у векторі $2^n$ станів, позначених примітивами. Кубіт (n-кубіт) дає можливість використовувати паралельні логічні операції замість поелементних теоретико-множинних для істотного прискорення процесів аналізу дискретних систем.

Далі кубіт ототожнюється з n-кубітом або двійковим вектором, якщо це не заважає розумінню викладеного матеріалу. Оскільки квантові обчислення пов'язані з аналізом кубітних структур даних, то далі будемо застосовувати визначення «квантовий» для ідентифікації технологій, які використовують три властивості квантової механіки: паралелізм обробки (двійкових векторів), суперпозицію станів і їх переплутування. Синонімами кубіта при завданні двійкового вектора опису логічної функції є: Q-покриття, Q-вектор, квантовий вектор [196, 197, 203] як уніфікована векторна форма суперпозиційного завдання вихідних станів, відповідних адресним кодам вхідних змінних логічного елемента.

Кубіт в цифровій системі використовується в якості форми завдання структурного примітиву, інваріантної до технологій реалізації функціональності (hardware, software). Більш того, синтез цифрових систем на основі кубітних структур не прив'язаний жорстко до теореми Поста, що визначає п'ять умов (класів) існування функціонально повного базису. На пропонованому рівні абстракції n-кубіт дає більш широкі можливості для векторного завдання будь-якої n-входової функції з булеана потужністю $|B(A)| = 2^{2^n}$, який неодмінно містить всі функціональності, що задовольняють п'яти класам теореми Поста. Формат структурного кубітного компонента цифрової схеми $Q^* = (X, Q, Y)$ містить інтерфейс (вхідні та вихідну змінні), а також кубіт-вектор Q, що задає функцію $Y = Q(X)$, розмірність якого визначається ступеневою функцією від кількості вхідних ліній $k = 2^n$.

## 8.3 Синтез квант-вектора комбінаційної схеми

Кубіт (квант) комбінаційної схеми є вектор станів виходу на впорядкованій множині всіх вхідних слів, які ототожнюються з адресами комірок пам'яті вектора. Синтез Q-вектора (покриття) схемної структури (без таблиць істинності логічних елементів) на основі примітивів, заданих Q-векторами зводиться до отримання узагальненого кубіт-вектора шляхом декартової процедури виконання логічної операції над розрядами кубітних векторів. Декартова процедура для двох чотирирозрядних кубітів, які суперпозиціонуются логічною операцією or (and, xor), представлена в наступній таблиці:

| ∨,∧,⊕ | b(0) | b(1) | b(2) | b(3) |
|---|---|---|---|---|
| a(0) | c(0) = a(0) ∨ b(0) | c(1) = a(0) ∨ b(1) | c(2) = a(0) ∨ b(2) | c(3) = a(0) ∨ b(3) |
| a(1) | c(4) = a(1) ∨ b(0) | c(5) = a(1) ∨ b(1) | c(6) = a(1) ∨ b(2) | c(7) = a(1) ∨ b(3) |
| a(2) | c(8) = a(2) ∨ b(0) | c(9) = a(2) ∨ b(1) | c(10) = a(2) ∨ b(2) | c(11) = a(2) ∨ b(3) |
| a(3) | c(12) = a(3) ∨ b(0) | c(13) = a(3) ∨ b(1) | c(14) = a(3) ∨ b(2) | c(15) = a(3) ∨ b(3) |

Приклади, які використовують логічні суперпозиції двох кубітів для отримання Q-векторів схемних структур

$$c_1 = (a_1 \wedge a_2) \vee (b_1 \vee b_2),\ c_2 = (a_1 \wedge a_2) \wedge (b_1 \vee b_2),\ c_3 = (a_1 \wedge a_2) \oplus (b_1 \vee b_2),$$

представлені наступною таблицею:

| a(and) = <br> b(or) = | 0001 <br> 0111 |
|---|---|
| $c_1$ = a(and) ∨ b(or) | 0111 0111 0111 1111 |
| $c_2$ = a(and) ∧ b(or) | 0000 0000 0000 0111 |
| $c_3$ = a(and) ⊕ b(or) | 0111 0111 0111 1000 |

Тут побудовані Q-покриття трьох схем, що складаються з трьох елементів кожна, де два логічних примітиви суперпозиційно об'єднуються третім елементом (or, and, xor). В результаті виходять три вектори, кожен з яких має розмірність в 16 біт. Обчислювальна складність процедури синтезу

Q-покриття комбінаційної схеми дорівнює добутку довжин Q-векторів p примітивів, що входять в неї: $\eta = \prod_{i=1}^{p} \mathrm{card}(Q_i)$.

Складнішою є проблема синтезу Q-покриття схеми, вхідні лінії (розгалуження, що сходяться) якої мають гальванічні або проводові з'єднання (тут за змінною $a_2$): $c = (a_1 \wedge a_2) \vee (a_2 \vee a_3)$. В даному випадку після синтезу Q-покриття схеми необхідно виконати його верифікацію щодо існування суперечливих адрес на змінних $a_2$ з метою мінімізації Q-вектора шляхом подальшого виключення зазначених адрес з розгляду, що зменшує розмірність Q-покриття до $\mathrm{card}(Q) = 2^q$ координат, де q – загальна кількість вхідних змінних схеми:

| Q = | 0111 0111 0111 1111 |
|---|---|
| $a_1$ = | 0000 0000 1111 1111 |
| $a_2$ = | 0000 1111 0000 1111 |
| $a_2$ = | 0011 0011 0011 0011 |
| $a_3$ = | 0101 0101 0101 0101 |

=

| Q = | 0111 0111 0111 1111 |
|---|---|
| $a_1$ = | 0000 0000 1111 1111 |
| $a_2$ = | 00xx xx11 00xx xx11 |
| $a_2$ = | 00xx xx11 00xx xx11 |
| $a_3$ = | 0101 0101 0101 0101 |

=

| Q = | 0111 0111 |
|---|---|
| $a_1$ = | 0000 1111 |
| $a_2$ = | 0011 0011 |
| $a_2$ = | 0011 0011 |
| $a_3$ = | 0101 0101 |

=

| Q = | 0111 0111 |
|---|---|
| $a_1$ = | 0000 1111 |
| $a_2$ = | 0011 0011 |
| $a_3$ = | 0101 0101 |

Процедура синтезу Q-покриття: 1) будується таблиця відповідності адрес розрядам Q-вектора схеми; 2) далі суперечливі координати у двох рядках $a_2$ відмічаються символом x; 3) потім всі стовпці з даними символами виключаються з таблиці; 4) після чого виходять два ідентичні рядки $a_2$, які об'єднуються в одну; 5) що дає в результаті Q-вектор комбінаційної схеми, але вже значно меншої розмірності. Переваги запропонованого Q-методу синтезу обчислювальних пристроїв полягають в компактності їх опису Q-векторами і в високій швидкодії адресного моделювання логічних елементів, створюють умови для створення ринково привабливої «квантової» теорії проектування цифрових систем на кристалах, що використовує векторно-кубітную форму завдання структурних компонентів.

## 8.4 Мінімізація квант-вектора схеми

Синтез квант-вектора схеми по Q-покриттях компонентів пов'язаний з мінімізацією або зменшенням розмірності Q-вектора шляхом виключення несуттєвих змінних. Як правило істотність залежить від гальванічних з'єднань вхідних та внутрішніх ліній цифрового пристрою, які накладають обмеження, пов'язані з суперечливістю сигналів на лініях зв'язку. Тому правило мінімізації адресного простору полягає в усуненні адресних кодів, які створюють протиріччя за з'єднаними змінними. Нехай є Q-вектор схеми і його адресний простір, де змінні b,c,d (a,b,c) з'єднані гальванично. Нижче наведено таблиці перетворення або мінімізації адресного простору з метою отримання зменшеного Q-вектора:

| Q = | 0111 0111 0111 1111 |
|---|---|
| a = | 0000 0000 1111 1111 |
| b = | 0000 1111 0000 1111 |
| c = | 0011 0011 0011 0011 |
| d = | 0101 0101 0101 0101 |

| Q = | 0xxx xxx1 0xxx xxx1 |
|---|---|
| a = | 0000 0000 1111 1111 |
| b = | 0xxx xxx1 0xxx xxx1 |
| c = | 0xxx xxx1 0xxx xxx1 |
| d = | 0xxx xxx1 0xxx xxx1 |

=

| Q | 0101 |
|---|---|
| a = | 0011 |
| b = | 0101 |

| Q = | 0111 0111 0111 1111 |
|---|---|
| a = | 00xx xxxx xxxx xx11 |
| b = | 00xx xxxx xxxx xx11 |
| c = | 00xx xxxx xxxx xx11 |
| d = | 0101 0101 0101 0101 |

=

| Q | 0111 |
|---|---|
| a = | 0011 |
| d = | 0101 |

По-перше, тут слід зазначити, що в таблицях спостерігається дзеркальна осьова симетрія з інверсією сигналів на координатах адресного простору, яка створює властивість, що описується наступним виразом: $L \oplus R = 1 \rightarrow L_{ij} \oplus R_{ij} = 1.$

Дану обставину слід використовувати для зменшення розмірності аналізованого простору в два рази і відповідного зниження обчислювальної складності задачі синтезу квантової вектор-функціональності цифрової схеми. По-друге, кількість різних варіантів взаємодії на q вхідних змінних, пов'язаних з гальваничним з'єднанням різних сполучень вхідних ліній, визначається функціональною залежністю, граничні значення якої

знаходяться в інтервалі: $\mathrm{card}(Q) = [2^q - 3^q]$. Проте, існує ефективна процедура для мінімізації розмірності Q-вектора шляхом виявлення суперечностей в кодах-стовпцях на координатах (Aij), відповідних гальванично пов'язаним w-змінним за j-параметром. Таку процедуру досить виконати на половині адресного простору $\mathrm{card}(Q) = 2^q / 2$. Інша частина суперечливих стовпців видаляється відповідно до дзеркальних відображень номерів тих стовпців, які були видалені з першої половини таблиці кодів адрес:

$$\{Q_i, Q_{2^q - i}\} = \varnothing \leftrightarrow (\overset{w}{\underset{j=1}{\wedge}} A_{ij}) \oplus (\overset{w}{\underset{j=1}{\vee}} A_{ij}) = 1, \; i \le 2^q / 2$$

Якщо в стовпці Ai на групі з w пов'язаних змінних зафіксовано, що кон'юнкція їх станів дорівнює нулю, а диз'юнкція має значення одиниці, то i-стовпець і його дзеркальне відображення $2^q - i$ видаляються з адресного простору A. Це автоматично призводить до виключення з Q-вектора двох отриманих $\varnothing$-координат (в таблицях позначені символами x), які відповідають даним стовпчикам.

Природно, що також спостерігається симетрія простору векторів-відстаней за Хеммінгом, отриманих шляхом xor-взаємодії між сусідніми рядками таблиці адресного простору, для яких суперпозиція лівої і правої частин дає результат $L \oplus R = 0 \rightarrow L_{ij} \oplus R_{ij} = 0$:

| Q = | 0111 0111 0111 1111 |
|---|---|
| a ⊕ b | 0000 1111 1111 0000 |
| b ⊕ c | 0011 1100 0011 1100 |
| c ⊕ d | 0110 0110 0110 0110 |
| d ⊕ a | 0101 0101 1010 1010 |

$= (L,R); \; (L \oplus R) =$

| Q = | 0111 0111 0111 1111 |
|---|---|
| a ⊕ b | 0000 0000 |
| b ⊕ c | 0000 0000 |
| c ⊕ d | 0000 0000 |
| d ⊕ a | 0000 0000 |

$\leftrightarrow (L = \overline{R})$

Чи доцільно мінімізувати логічну функцію, описану квант-вектором? Відповідь: мінімізація Q-векторів для отримання нормальних або дужкових форм не має практичного значення, істотно тільки зменшення розмірності

вектора функціонального опису, що може бути лише наслідком визначення неістотності деяких вхідних (адресних) змінних. Проте, існує проблема розбиття квант-вектора на складові частини меншої розмірності, що пов'язано з імплементацією функціональності в конструктивні компоненти LUT FPGA. В цьому випадку виконується розбиття Q-вектора на два рівних підвектори Q=(L, R), які з'єднуються в структурно-адресну організацію функціональності за допомогою мультиплексора $Q = (\overline{a} \wedge L) \vee (a \wedge R)$. Якщо змінна мультиплексування a=0, то функціональність Q формується за допомогою комірок лівого L-вектора, в іншому випадку, коли a=1, значення функції Q формується бітами правого R-вектора. Алгоритми розбиття та імплементації складних логічних функцій є в кожній промислової системі синтезу, моделювання і верифікації компонентів SoC.

## 8.5 Імплементація куба функціональності в технологічну структуру PLD

Вирішення задачі зводиться до розбиття куба функціональності (див. Розд. 6) на сегменти, яким можна поставити у відповідність бібліотечні примітиви конкретного кристалу. На кристалі PLD (FPGA) є в наявності певна кількість чотиривходових примітивів – елементів пам'яті для зберігання таблиці істинності LUT (Look Up Table). Примітиви можуть об'єднуватися в логічні сектори (Slice), що складаються з двох примітивів, об'єднаних мультиплексором, що дає можливість збільшити розрядність вхідних змінних до п'яти. Подальше об'єднання двох секторів за допомогою мультиплексора збільшує розрядність вхідних змінних до шести. При цьому на кристалі технологічно передбачено, що кожні 2 пари секторів мультиплексуються в блоки CLB (Configurable Logic Block), що дає можливість збільшити кількість входів синтезованої логічної функції в межах одного CLB до семи змінних. Структура зазначених вище модулів (LUT, Slice, CLB) представлена на рис. 8.1. Тут значення функції від семи змінних формується на виході мультиплексора MUXF7. Подальше збільшення

розрядності вхідних змінних пов'язане з примусовою структурною організацією компонентів CLB в ієрархічні структури більш високого рівня: 1 CLB здатний формувати будь-яку функцію від 7 змінних, 2 CLB – 8 змінних, 4 CLB – 9 змінних, 8 CLB – 10 змінних. У загальному випадку функціональна залежність кількості змінних n від кількості CLB-блоків N має наступний вигляд: $n = \log_2 N + 7$.

Для оцінки апаратної складності при синтезі або реалізації обчислювальної структури необхідно мати зворотну залежність кількості логічних блоків CLB від складності логічної функції, приведеної до кількості змінних n або потужності куба функціональності $N = 2^n$. Наприклад, для функції від n=10 змінних необхідно мати на кристалі 8 CLB. У загальному випадку число логічних блоків CLB, в залежності від кількості змінних синтезованої функції, так само $N = 2^{n-7}$, $n = 7,8,9, ....$

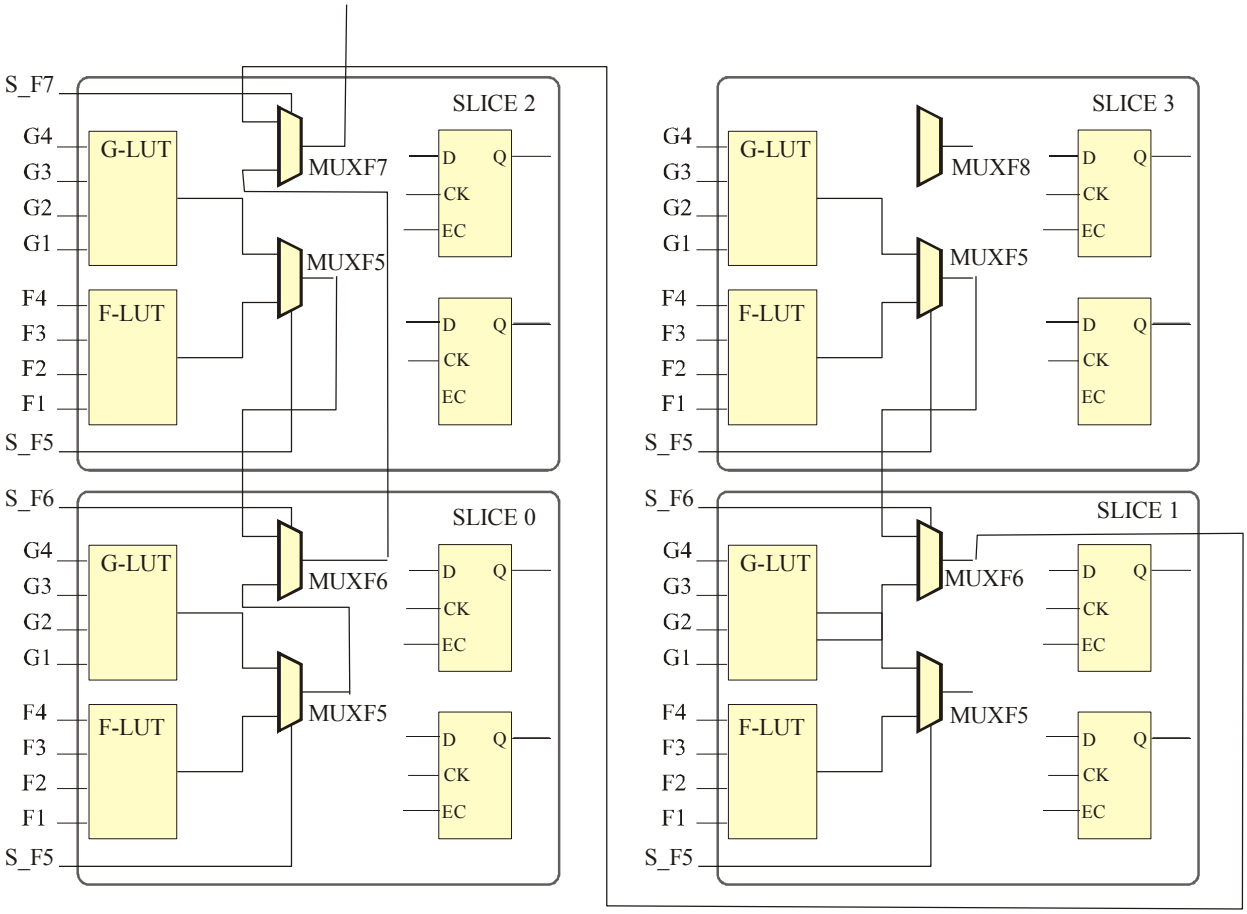



Рисунок 8.1 – Структура моделей PLD

Що стосується кількості примітивів LUT, то тут оцінка апаратної складності буде представлена наступним виразом:

$$N^* = 8L \times 2^{n-7} + 4M \times 2^{n-7} = (8L + 4M) \times 2^{n-7}.$$

Тут 8L, 4M – апаратна складність реалізації LUT і мультиплексора, що входять до складу CLB. При цьому важливі характеристики реалізації функціональності – структурна глибина, як кількість рівнів ієрархії, виражена у кількості LUT і/або мультиплексорах на найдовшому логічному шляху і його затримка D – також залежать від числа змінних (затримки мультиплексора):

$$S = (n - 3);\ D = (n - 3) \times D_M.$$

Приклад: є куб цифрової структури, що містить $p = 4096$, $(n = 12)$ бітів (змінних).

Післясинтезні характеристики такої функціональності матимуть вигляд:

$$N = 2^{12-7} = 32;$$
$$N^* = (8L + 4M) \times 2^{n-7} = (8L + 4M) \times 2^5 = 32 \times (8L + 4M);$$
$$S = (12 - 3) = 9;$$
$$D = 9 \times D_M.$$

При наявності часових характеристик примітивів для конкретного кристалу програмовної логіки, а також оцінок апаратної вартості реалізації структурних компонентів (LUT, Slice, CLB) можна підрахувати точніше параметри функціональності, імплементованої в кристал PLD (FPGA).

Запропоновано кубітні (квантові) структури даних і обчислювальних процесів для істотного підвищення швидкодії при вирішенні задач дискретної оптимізації та відмовостійкого проектування. Представлено суперпозиційний метод синтезу куба функціональності для її імплементації в кристали програмовної логіки.

## 8.6 Модель квантового процесора

Квантовий процесор може бути будь-якої кінцевої розмірності: вектор, матриця, куб. Для структури, що містить два виміри, він представлений матрицею стовпців або Q-векторів, які формують відповідні їм комірки M-вектора моделювання (рис. 8.2, ліва частина). Вектор M, спільно з X-вектором кортежів вхідних змінних примітивів створює структуру взаємних зв'язків між стовпцями-елементами. Адреса комірки Q-покриття, що формує стан невхідного і-розряду M-вектора, визначається вмістом комірок M-вектора, знайдених за адресами, заданими і-кортежем вектору вхідних змінних. Кожен вектор Qi, так само як і кортеж Xi вектора номерів вхідних ліній, має адресний зв'язок з Mi-коміркою вектора моделювання. Квантовий процесор може входити компонентом до складу більш складної системи. Квантова модель процесора має таку структуру:

$$
\begin{aligned}
&W =< Q, M, X >, \\
&Q = (Q_1, Q_2, ..., Q_i, ..., Q_n), \\
&Q_i = (Q_{i1}, Q_{i2}, ..., Q_{ij}, ..., Q_{ik_i}); \\
&M = (M_1, M_2, ..., M_i, ..., M_n); \\
&X = (X_1, X_2, ..., X_i, ..., X_n), \\
&X_i = (X_{i1}, X_{i2}, ..., X_{ij}, ..., X_{im_i}); \\
&M_i = Q_i[M(X_i)];\ k_i = 2^{m_i}.
\end{aligned}
$$

В аналітичній моделі W представлені: 1) Упорядкована адресно-доступна Q-сукупність квантових примітивів, які формують функціональність системи. 2) Вектор моделювання M, що зв'язує всі примітиви в єдину систему на основі ідентифікації еквіпотенційних ліній, які створюють формат з істотних змінних: вхідних, внутрішніх і вихідних. 3) Вектор X кортежів упорядкованих номерів вхідних змінних для кожного квантового примітиву, які формують адреси доступу до комірок Q-векторів примітивів (рис. 8.2, ліва частина). Вектор кількості вхідних змінних примітиву $|X|$ формує адресний простір або довжину кожного Q-покриття. Його можна представити у вигляді таблиці кортежів вхідних змінних, які

формують номери ліній вектора моделювання для обчислення адрес доступу до квантових покриттів (рис. 8.2, середня частина). Таблицю кортежів можна також представити у вигляді матриці масок входів, визначених у форматі вектора моделювання, для паралельного формування адрес і одночасного зчитування вихідних станів примітивів з матриці Q-покриттів (рис. 8.2, права частина). Зі структури X-матриці вхідних ліній видно, що кванти, які формують виходи: (8, 9, 10), (11, 12) і (13, 14) можна обробляти паралельно. 4) Характеристичне рівняння, що задає алгоритм функціонування квантового процесора на основі використання тільки операцій транзакції (зчитування-запис) між Q-векторами примітивів і вектором моделювання.

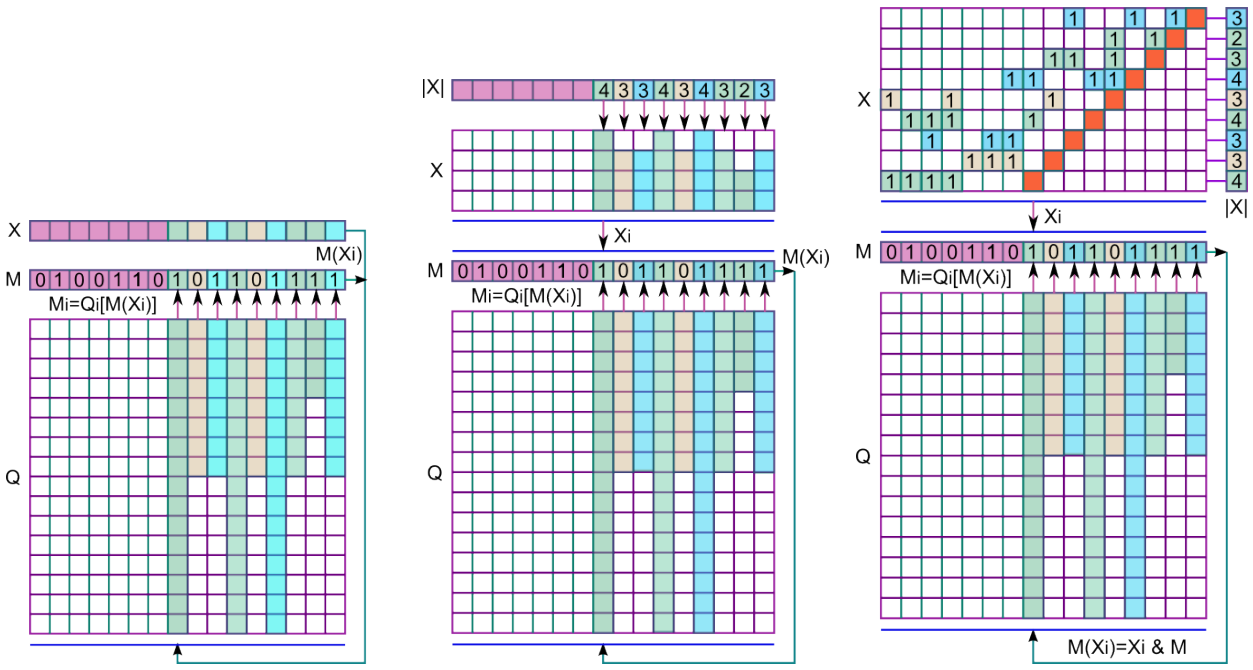



Рисунок 8.2 – Кубітні структури даних квантового процесору

Схема цифрового пристрою, яка відповідає наведеному вище опису структур даних: M-вектор моделювання, X-матриця входів і Q-матриця покриттів, представлена на рис. 8.3. Вона містить 9 примітивів, кожен з яких має Q-покриття в формі квант-вектора, що реалізує певну функціональність. Особливість квантових структур даних, що представляють модель цифрової

схеми, полягає в повній адресовності всіх компонентів пристрою без гальванічних проводових з'єднань.

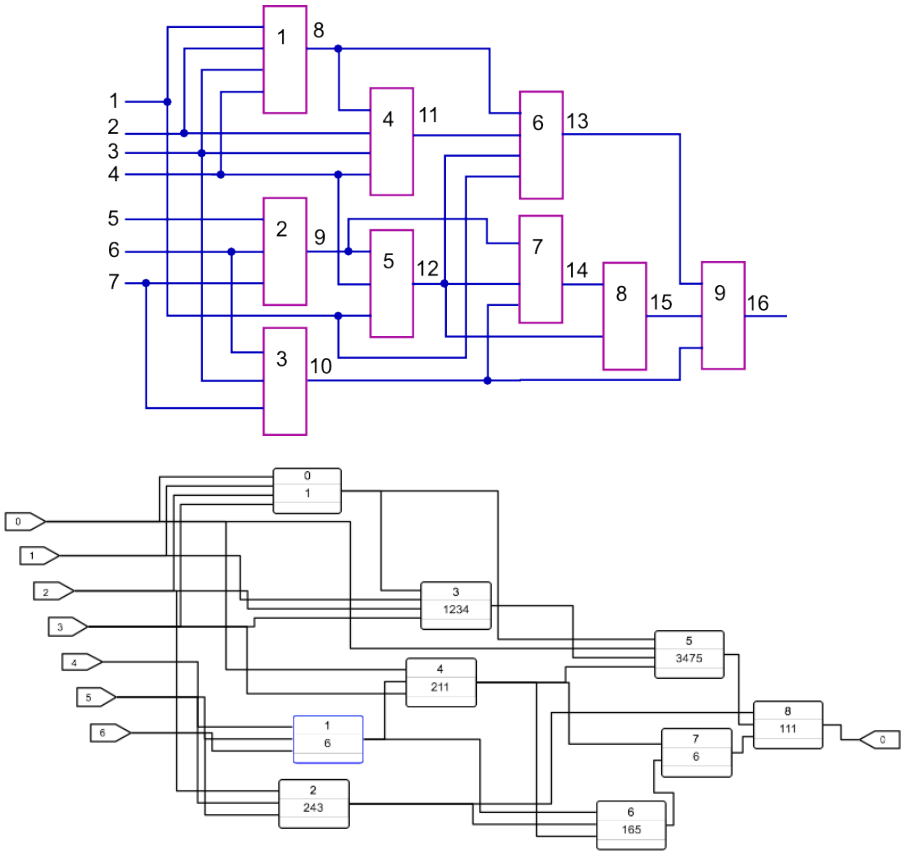



Рисунок 8.3 – Схемна модель цифрового пристрою

Аксіоми квантового (only memory-based) процесора: 1) В квантовому процесорі немає нічого, крім адресовної пам'яті. 2) Обчислювальний процес представлений єдиною універсальною транзакцією між адресовними компонентами пам'яті Mi = Qi [M(Xi)]. 3) Транзакція є універсальною процедурою зчитування-запису даних на непорожній множині адресовних елементів пам'яті. 4) Всі компоненти пам'яті є online-repaired, завдяки їх псевдогальванічній адресній (address-connected) зв'язності. 5) Комбінаційні логічні елементи (reusable logic), так само як і послідовні (sequential components), виконуються на елементах пам'яті. 6) З'єднання всіх компонентів в обчислювальну систему здійснюється за допомогою (цифрової) ідентифікації псевдо-гальванічних з'єднань вхід-вихідних змінних

компонентів схеми, які формують вектор моделювання, який зберігає стани всіх істотних ліній цифрової системи. 7) Всі компоненти квантової моделі цифрової системи: W=<Q,M,X>, у тому числі функціональні модулі, вектор моделювання, вектор адрес вхідних змінних, є online перепрограмовними, тобто – online ремонтопридатними. 8) Примітив цифрової системи має формат W=<Q,Y,X>, оскільки окремий елемент не має зв'язків і вектора M, що створюють систему з окремих компонентів.

Згідно введеної квантової моделі, описи послідовних примітивів (тригери, регістри, лічильники) можна представляти Q-покриттями або кубітними векторами, які мають псевдозмінні для завдання внутрішнього стану. Наприклад, функціональний опис SR-тригера трансформується в квантовий примітив, заданий Q-покриттям, а потім реалізується на адресовному елементі пам'яті FPGA з діаграмами перевірки, які наведено на рис. 8.4.

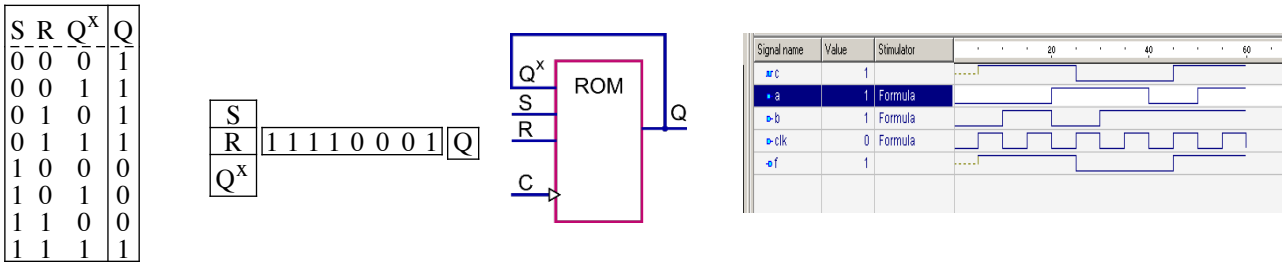



Рисунок 8.4 – Реалізація SR-тригера на елементі пам'яті

Таблиця істинності тригера представлена в формі вектора вихідних станів $Q(S,R,Q^X) = (11110001)$, який записується в елемент постійної пам'яті, що підтримує три адресних входи, сигнал синхронізації, а також зворотний зв'язок, який з'єднує вихід елемента пам'яті з одним адресним входом. HDL-реалізація в системі проектування Active HDL 9.1 (Aldec Inc.), а також результати верифікації синтезованого SR-тригера підтверджують коректність схемотехнічного рішення.

Інший приклад пов'язаний з синтезом на елементі постійної пам'яті синхронного DV-тригера. Таблиця істинності тригера трансформована у вектор вихідних станів $Q(D,V,Q^X)=(01000111)$, який записується в елемент пам'яті, що має три адресних входи, сигнал синхронізації, а також зворотний зв'язок, якій з'єднує вихід примітиву пам'яті з одним адресним входом. Зазначені компоненти, у тому числі часові діаграми верифікації HDL-коду моделі DV-тригера представлені на рис. 8.5.

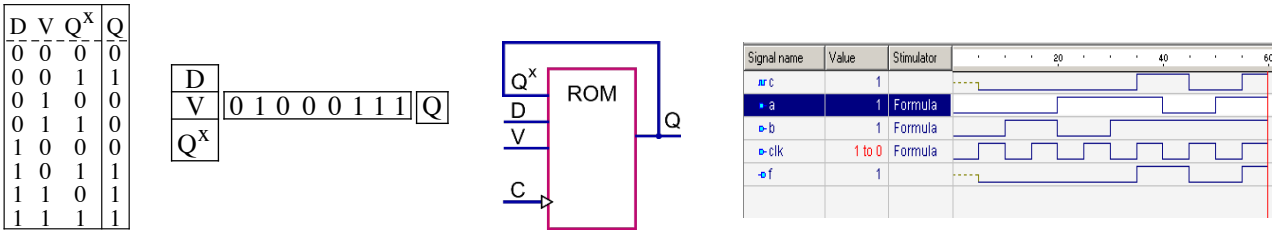



Рисунок 8.5 – Реалізація DV-тригера на елементі пам'яті

На рис. 8.6 наведено моделі двох послідовних примітивів: дворозрядного регістра та лічильника. Їх відмінність полягає в завданні двох виходів, стани яких формуються однією і тією ж множиною вхідних змінних.

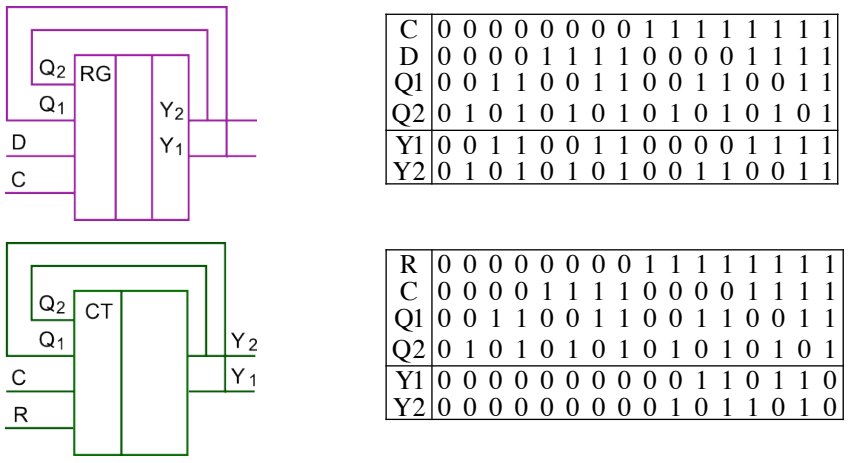



Рисунок 8.6 – Memory-based моделі регістра та лічильника

Регістр на змінних (C, D, Q1, Q2, Y1, Y2) виконує функцію зсуву вправо інформації від входу D за розрядами: (D-Y1-Y2), при R=1, і збереження даних при C=0. Лічильник, визначений на змінних (R, C, Q1, Q2, Y1, Y2), реалізує функцію інкремента за розрядами (Y1, Y2), при RC=(11), а також режим зберігання інформації, при (R or C=0). Таким чином, для реалізації дворозрядного регістра або лічильника необхідно два 16-бітових елементи пям'яті, що працюють синхронно від одних і тих же входів:

| | | |
|---|---|---|
| C | | |
| D | 0 0 1 1 0 0 1 1 0 0 0 0 1 1 1 1 | Y1 |
| Q1 | 0 1 0 1 0 1 0 1 0 0 1 1 0 0 1 1 | Y2 |
| Q2 | | |

| | | |
|---|---|---|
| R | | |
| C | 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 1 0 | Y1 |
| Q1 | 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 0 1 0 | Y2 |
| Q2 | | |

Тут кожна квантова модель представлена двома векторами, де кожен з них формує функцію розряду регістра або лічильника, як стан комірки вектора, отриманої при формуванні адреси A вхідними змінними: {Y1, Y2} = A(C, D, Q1, Q2), {Y1, Y2} = A(R, C, Q1, Q2) відповідно. Моделювання примітиву зводиться до тривіальної процедури формування адреси, за якою знаходиться стан виходу примітиву, як вміст комірки квантового вектора.

## 8.7 Алгоритм моделювання квантових покриттів цифрових компонентів

Використовує memory-based only моделі для адресного аналізу цифрових систем з метою їх верифікації. Реалізація таких структур пов'язана з комиркамі пам'яті (LUT (Look Up Table) FPGA), які здатні зберігати інформацію у вигляді Q-вектора, де кожен біт або розряд має свою адресу, що ототожнюється з вхідним словом. Програмна реалізація алгоритму моделювання таких структур стає конкурентоспроможною за швидкодією на ринку проектування цифрових систем на кристалах за рахунок адресації функціональних примітивів.

Одновимірній Q-вектор опису функціональності можна прив'язати до вихідної (внутрішньої) лінії пристрою, стан якої формується в процесі моделювання розглянутого Q-покриття. Тоді регістрова реалізація комбінаційного пристрою може бути представлена вектором моделювання

M, невхідні лінії якого безпосередньо пов'язані з виходами функціональних елементів. Впорядковані значення вхідних змінних задають адресу біта Q-вектора, що формує стан розглянутої невхідної лінії. Якщо функціональності описуються одновиходовими примітивами, то кожен з них можна ототожнити з номером або координатою невхідної лінії, на яку навантажений даний елемент. Якщо функціональність багатовиходова, то Q-покриття являє собою таблицю з кількістю рядків, що дорівнює кількості виходів. Ефект від такого примітиву полягає в паралелізмі одночасного обчислення станів кількох виходів за одне звернення до матриці за поточною адресою. Дана обставина є істотним аргументом на користь синтезу узагальнених кубітів для фрагментів цифрового пристрою або всієї схеми з метою їх паралельної обробки на одному часовому такті. Модель функціонування цифрової структури спрощується до обчислення двох адрес при формуванні вектора моделювання $M_i = Q_i[M(X_i)]$ шляхом виключення складної адреси виходу примітиву в процесі запису станів виходів в координати M-вектора.

Алгоритм моделювання квантових примітивів цифрової системи використовує аналітичну структуру (k – кількість вхідних змінних і-примітиву, * – операція конкатенації бітів, A – адреса біта Q-вектора):

$$\boxed{M_i = Q_i(A)} \leftarrow \boxed{A = \overset{k}{\underset{j=1}{*}} M(X_{ij})} \leftarrow \begin{array}{|c|}\hline M \\ \hline X_i \\ \hline Q_i \\ \hline \end{array}$$

Даному аналітичному виразу можна поставити у відповідність наступні пункти алгоритму формування двійкових станів M-вектора моделювання цифрової схеми, зображені на рис. 8.7: 0) Ініціювання початкових умов і параметрів. 1) Завдання чергового набору двійкових станів на вхідних координатах вектора моделювання. 2) Визначення і-номера чергового оброблюваного примітиву шляхом виконання операції інкрементування. 3) Виконання процедури конкатенації станів бітів M-вектора, відповідних номерам вектора вхідних змінних Xi. Зчитування відповідного біта з

функціонального кубіт-покриття Qi за допомогою бінарної вектор-адреси сконкатенованих бітів М-вектора. Занесення зчитаного з кубіта біта у вектор моделювання М за адресою і. (М-вектор може мати координати з символами X, що дає можливість виконувати потрійне моделювання цифрових пристроїв для вирішення задач тестування і верифікації.) 4) Якщо не всі примітиви оброблені i<n, виконується перехід до пункту 2 алгоритму. 5) Якщо не всі вхідні набори оброблені t<m, виконується перехід до пункту 1. 6) Кінець моделювання.

Виходячи з характеристичного рівняння квантової моделі цифрової системи можна зробити висновок, що сучасний <MQT> (Memory-Quant-Transaction) процесор слід представляти як адресну організацію структури функціональних примітивів пам'яті без гальванічних або проводових зв'язків, на яких визначені адресні транзакції даних в часі і просторі для досягнення поставленої мети.

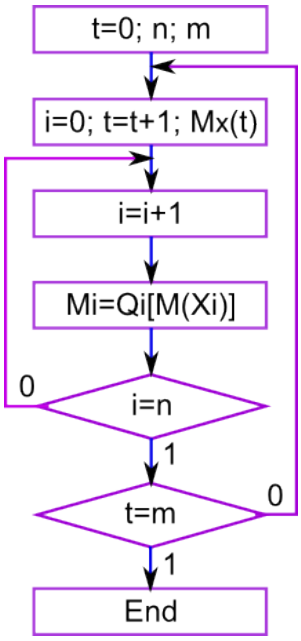
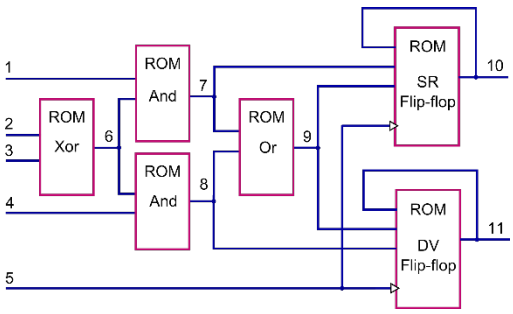



Рисунок 8.7 – Алгоритм моделювання квантових покриттів цифрової системи

На рис. 8.8 представлена схема з тригерами і комбінаційної логікою, яка також описана у вигляді елементів пам'яті, куди занесено вихідні стани таблиці істинності кожного логічного елемента. Структури даних, необхідні для моделювання цифрового пристрою зведено в таблицю, де основними

компонентами є: M – вектор моделювання або стану занумерованих ліній, який в даному випадку має 5 вхідних, 6 внутрішніх і вихідних ліній, стани яких підлягають визначенню; X – вектор кортежів номерів вхідних ліній примітивів, які необхідні для формування адреси з метою отримання по ній стану виходу елемента Qi, функціональність якого задається Q-вектором.




| L | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 1( |
|---|---|---|---|---|---|---|---|---|---|---|
| M | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
| X | . | . | . | . | . | 2 | 1 | 6 | 7 | 1( |
| | . | . | . | . | . | 3 | 6 | 4 | 8 | 7 |
| | . | . | . | . | . | . | . | . | . | 9 |
| Q | . | . | . | . | . | 0 | 0 | 0 | 0 | 1 |
| | . | . | . | . | . | 1 | 0 | 0 | 1 | 1 |
| | . | . | . | . | . | 1 | 0 | 0 | 1 | 1 |
| | . | . | . | . | . | 0 | 1 | 1 | 1 | 1 |
| | . | . | . | . | . | . | . | . | . | 0 |
| | . | . | . | . | . | . | . | . | . | 0 |
| | . | . | . | . | . | . | . | . | . | 0 |
| | . | . | . | . | . | . | . | . | . | 1 |

Рисунок 8.8 – Memory-based комбінаційна схема з тригерами

Приклад виконання алгоритму моделювання схемної квантової структури. Всі примітиви повинні бути впорядковані за принципом: черговий елемент аналізується, якщо все попередники для нього були оброблені. В процесі моделювання адресно витягнутий стан комірки поточного Q-покриття заноситься в розряд Mi вектора моделювання. Результати послідовної обробки всіх Q-векторів схемної структури формують стани ліній M-вектора, для наведеного вище прикладу комірки (6-11). Початкові стани невизначеностей на псевдовходах функціональних примітивів довизначають сигналами нуля або одиниці в залежності від внутрішньої технологічної культури компанії, що надає промислові засоби моделювання і верифікації. Кількість вхідних змінних примітиву q пов'язана з довжиною Q-вектора співвідношенням: $\mathrm{card}(Q) = 2^q$. Правильність роботи алгоритму моделювання була верифікована на тестових і реальних схемах із залученням засобів Active HDL 9.1 (Aldec Inc.). Особливість структурно-функціонального завдання цифрової системи полягає в поданні всіх примітивів елементами пам'яті, куди записуються Q-вектори вихідних станів.

Висновки: 1) Будь-які структурні компоненти обчислювальних пристроїв, комбінаційні і/або послідовностні, а також системи в цілому можна описувати кубітними Q-векторами і реалізовувати в елементах пам'яті FPGA, CPLD або VLSI. Це надає ринку електронних технологій можливість не використовувати комбінаційну reusable логіку при синтезі обчислювальних пристроїв, яка доставляє розробникам серйозні проблеми, пов'язані з тестуванням, верифікацією і ремонтом жорсткої проводової реалізації цифрових виробів. 2) Memory-based інтерпретативне адресно-орієнтоване моделювання комбінаційних і послідовностних примітивів цифрових пристроїв стає порівнянним за швидкодією з компілятивним аналізом дискретних об'єктів. Крім того, стає можливим реалізовувати на програмовних логічних пристроях апаратне моделювання цифрових систем, де комбінаційні і послідовносні функціональні примітиви будуть представлені стандартними елементами пам'яті, в які зашиваються Q-вектори.

## 8.8 Моделювання синхронних цифрових схем

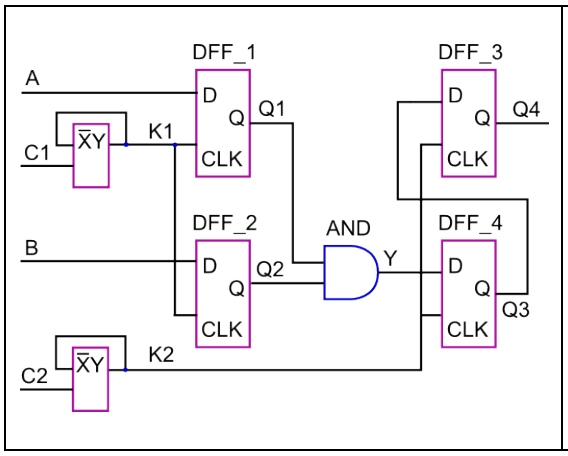
Сигнали синхронізації доставляють певні незручності для опису моделей послідовностних компонентів (тригери, регістри, лічильники) і реалізації алгоритмів аналізу. Це пов'язано зі схемотехнічним виконанням управління за переднім або заднім фронтом, які дозволяють виконання транзакцій між master-slave компонентами. Іншими словами, синхронні примітиви мають два послідовно з'єднаних елементі, орієнтовані на низький і високий рівні сигналів запису даних в перший і другий момент часу відповідно. Однак для логічного моделювання урахування подробиць, пов'язаних зі схемотехнічними рішеннями, може істотно уповільнити час аналізу цифрових схем. Тому тут необхідні логічно адекватні моделі реальних процесів, що приводять до підвищення швидкодії алгоритмів обробки компонентів. При цьому накладається обмеження, пов'язане тільки з адресним характером аналізу всіх компонентів схеми. Для забезпечення

можливості розгляду синхровходу, як логічної змінної, що формує адресу квантового вектора, пропонується модель розбиття послідовностного примітиву на два елементи: 1) Логічний квант видачі дозвільного сигналу при формуванні переднього (заднього) фронту в двох часових модельних тактах. 2) Квант реалізації штатної функціональності (тригера, регістра, лічильника) послідовностного компонента. Таким чином, модель синхронного D-тригера може бути описана у формі двох Q-покриттів, адресно обчислюючих стани виходів:

| CLK(t – 1) / CLK(t) | 0 1 0 0 | C(t) |
|---|---|---|

| Q(t – 1) / D(t) / C(t) | 0 0 0 1 0 0 1 1 | Q(t) |
|---|---|---|

С урахуванням викладеного вище, модель цифрової схеми з двома сигналами синхронізації, представлена на рис. 8.9, буде мати структури даних, що складаються з сукупності Q-покриттів, які формують поточний вектор моделювання M, але з урахуванням значень координат даного вектора в попередній момент часу M(t-1). Збільшення кількості змінних за рахунок введення двох елементів синхронізації зменшує сукупну розмірність таблиці квантових векторів, яка при 7 змінних матиме 56 координат.




| L | A | $C_1$ | B | $C_2$ | $K_1$ | $Q_1$ | $Q_2$ | Y | $K_2$ | $Q_3$ | $Q_4$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $M^{t-1}$ | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $M^t$ | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| X | . | . | . | . | $C_1^{t-1}$ | $Q_1^{t-1}$ | $Q_2^{t-1}$ | $Q_1$ | $C_2^{t-1}$ | $Q_3^{t-1}$ | $Q_4^{t-1}$ |
| | . | . | . | . | $C_1^t$ | A | B | $Q_1$ | $C_2^t$ | Y | $Q_3$ |
| | . | . | . | . | . | $K_1$ | $K_1$ | . | . | $K_2$ | $K_2$ |
| Q | . | . | . | . | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | . | . | . | . | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| | . | . | . | . | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | . | . | . | . | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| | . | . | . | . | . | 0 | 0 | . | . | 0 | 0 |
| | . | . | . | . | . | 0 | 0 | . | . | 0 | 0 |
| | . | . | . | . | . | 1 | 1 | . | . | 1 | 1 |
| | . | . | . | . | . | 1 | 1 | . | . | 1 | 1 |

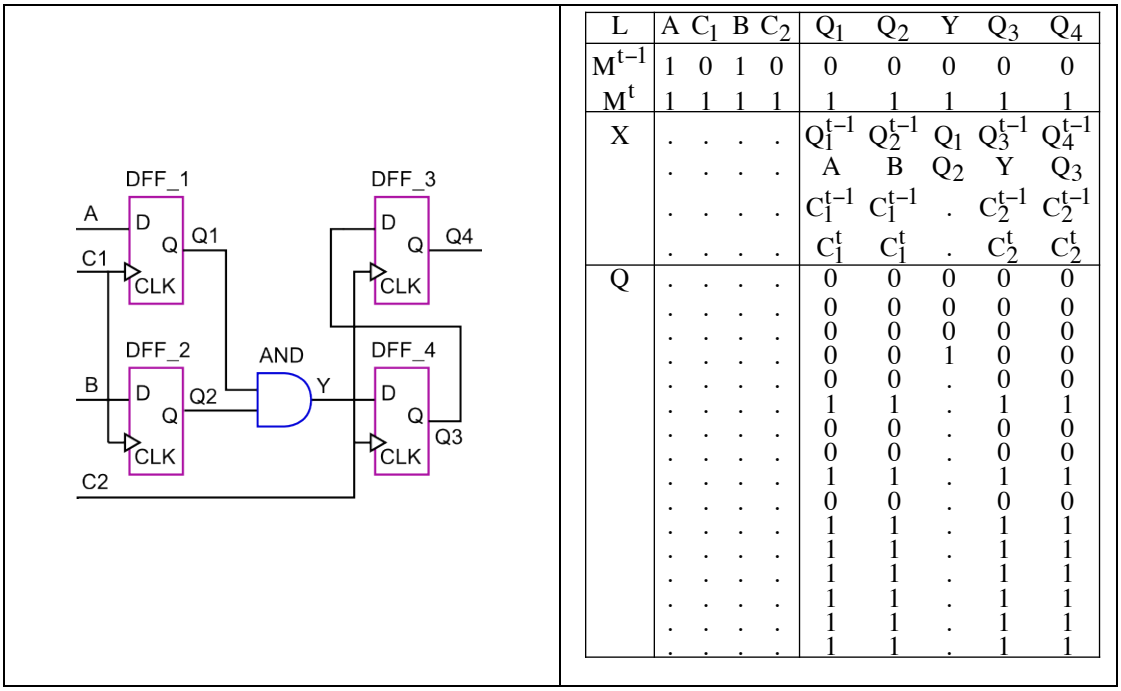
Рисунок 8.9 – Схема з D-тригерами і елементами синхронізації

Якщо не вводити дві додаткові змінні (елементи синхронізації), то обсяг пам'яті для Q-покриттів буде дорівнювати 80 коміркам (рис. 8.10).

Наведена схемна реалізація максимально орієнтована на структури даних промислових засобів моделювання і верифікації. Однак квантові вектори для завдання функциональностей тригерів створюють необхідні умови для підвищення швидкодії інтерпретативного аналізу, тестування і діагностування схемних компонентів.

Проблема зменшення сукупного обсягу Q-покриттів схеми пов'язана з кількістю змінних, які формують адреси координат Q-вектора. Природно, що будь-яке розбиття кількості змінних на дві рівних упорядкованих підмножини дає можливість істотно зменшити розмірність пам'яті для запису вже двох Q-векторів. У загальному випадку функціональна залежність зменшення розмірності вихідного Q-вектора, визначеного на n-змінних, при розподілі на 2 підсхеми з рівним числом результуючих змінних (n/2), має наступний вигляд:

$$Q = \frac{2^n}{2^{n/2} + 2^{n/2}} = \frac{2^n}{2 \times 2^{n/2}} = \frac{2^n}{2^{n/2+1}} = 2^{n-(n/2+1)} = 2^{n/2-1}.$$




| L | A | $C_1$ | B | $C_2$ | $Q_1$ | $Q_2$ | Y | $Q_3$ | $Q_4$ |
|---|---|---|---|---|---|---|---|---|---|
| $M^{t-1}$ | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| $M^t$ | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| X | . | . | . | . | $Q_1^{t-1}$ | $Q_2^{t-1}$ | $Q_1$ | $Q_3^{t-1}$ | $Q_4^{t-1}$ |
| | . | . | . | . | A | B | $Q_2$ | Y | $Q_3$ |
| | . | . | . | . | $C_1^{t-1}$ | $C_1^{t-1}$ | . | $C_2^{t-1}$ | $C_2^{t-1}$ |
| | . | . | . | . | $C_1^t$ | $C_1^t$ | . | $C_2^t$ | $C_2^t$ |
| Q | . | . | . | . | 0 | 0 | 0 | 0 | 0 |
| | . | . | . | . | 0 | 0 | 0 | 0 | 0 |
| | . | . | . | . | 0 | 0 | 0 | 0 | 0 |
| | . | . | . | . | 0 | 0 | 1 | 0 | 0 |
| | . | . | . | . | 0 | 0 | . | 0 | 0 |
| | . | . | . | . | 1 | 1 | . | 1 | 1 |
| | . | . | . | . | 0 | 0 | . | 0 | 0 |
| | . | . | . | . | 0 | 0 | . | 0 | 0 |
| | . | . | . | . | 1 | 1 | . | 1 | 1 |
| | . | . | . | . | 0 | 0 | . | 0 | 0 |
| | . | . | . | . | 1 | 1 | . | 1 | 1 |
| | . | . | . | . | 1 | 1 | . | 1 | 1 |
| | . | . | . | . | 1 | 1 | . | 1 | 1 |
| | . | . | . | . | 1 | 1 | . | 1 | 1 |
| | . | . | . | . | 1 | 1 | . | 1 | 1 |
| | . | . | . | . | 1 | 1 | . | 1 | 1 |

Рисунок 8.10 – Схема з D-тригерами на основі внутрішньої синхронізації

Наприклад, розбиття вектора з восьми змінних на два Q-покриття з 4 змінними дає зменшення обсягу пам'яті у 8 разів. Однак слід мати на увазі, що кожне розбиття функціонального модуля на k підсхем відносно зовнішніх входів потребує k додаткових d-циклів для обчислення стану виходу всієї схеми $T = d(k+1)$. Зниження швидкодії розбиття функціональності є платою за істотне зменшення обсягу пам'яті для зберігання структур даних цифрової системи. У загальному випадку розбиття функціональності на k однакових частин призводить до отримання такої залежності виграшу обсягу пам'яті від числа розбиттів на підмножини вектора вхідних змінних:

$$Q = \frac{2^n}{k \times 2^{n/k}} = \frac{1}{k} \times 2^{n-n/k}.$$

Тут параметр розбиття k приймає значення, кратні ступеню двійки: 2, 4, 8, 16. Однак значення k не повинно бути більше, ніж n/2. Однак слід зауважити, що необов'язково кількість розбиття має приймати значення, кратні ступеню двійки. У загальному випадку, на векторі вхідних змінних може існувати m розбиттів, кожне з яких має більше однієї змінної. При цьому виконується умова розбиття, що сума всіх змінних, які беруть участь у розбитті, не може бути більше n:

$$Q = \frac{2^n}{2^{k_1} + 2^{k_2} + \ldots + 2^{k_i} + \ldots + 2^{k_m}}, \; k_1 + k_2 + \ldots + k_i + \ldots + k_m = n.$$

Формула показує виграш від розбиття функціональності у вигляді відношення розмірності вихідного квант-вектора до сукупного обсягу Q-векторів, отриманих після розбиття. Щоб оцінити ефективність розбиття функціональності на схемні фрагменти, необхідно враховувати не тільки зменшення обсягу пам'яті для зберігання структур даних, але й негативні наслідки, пов'язані з вартістю аналізу збільшеної кількості схемних компонентів, яка регламентується в кожному конкретному випадку коефіцієнтом d:

$$Q = \frac{2^n}{d \times m \times (2^{k_1} + 2^{k_2} + \ldots + 2^{k_i} + \ldots + 2^{k_m})}.$$

Підводячи підсумок в частині модифікації теорії справного моделювання (fault-free) цифрових систем, можна відзначити наступні факти. Автомат моделювання синхронних цифрових пристроїв, як правило, представлений моделлю Мура:

$$S(t) = f[X(t), S(t-1)];$$
$$Y(t) = f[X(t), S(t)].$$

Тут фігурують вхідні (X) і внутрішні стани автомата в двох сусідніх часових фреймах S(t), S(t-1), а також правила визначення вихідних значень Y(t) для ініціювання обчислювальних процедур. Пропонується модифікація зазначеної моделі автомата Мура для аналізу цифрових систем, сутність якої полягає в заміні функціональних відносин адресними (A) операціями:

$$S(t) = A[X(t), S(t-1)];$$
$$Y(t) = A[X(t), S(t)].$$

Адресний або квантовий автомат, визначений вище дозволяє: 1) Уникнути жорстких гальванічних міжз'єднань між елементами комбінаційних і послідовностних схем при їх апаратній імплементації тільки в елементи пам'яті. 2) Отримати властивість гнучкої замінюваності компонентів цифрової системи в режимі online, завдяки їх адресовності. 3) Істотно спростити всі процеси моделювання, верифікації та тестування шляхом використання тільки процедур обчислення адреси компонента схеми або комірки його пам'яті. 4) Уніфікувати процеси проектування цифрових виробів шляхом їх зведення до формування функциональностей на основі обчислення адрес або до транзакцій на елементах пам'яті. 5) Підвищити ефективність процедур моделювання цифрових схем за рахунок зменшення обсягу інтерпретативних моделей і спрощення способу їх обробки, коли замість вичерпного аналізу таблиць пропонується обчислення адреси комірки квантового вектора. 6) Виконувати всі обчислювальні процедури на основі використання квантових покриттів і вектора моделювання, заданого в двох сусідніх автоматних тактах, згідно з визначенням квантового автомата. 7) Технологічно простіше стає використовувати інфраструктуру [193, 198] стандартів тестопридатного проектування (IEEE 1500, 1149) для

покомпонентного тестування, діагностування та відновлення працездатності адресно доступних функціональних блоків в режимі online.

## 8.9 Структура хмарного сервісу QuaSim для моделювання цифрових пристроїв

QuaSim є засобом для аналізу, тестування і верифікації цифрових проектів невеликої розмірності і призначений для використання в навчальному процесі в якості хмарного сервісу, доступного для студентів з будь-якого мобільного пристрою або комп'ютера.

Мета – істотне підвищення якості навчального процесу за рахунок надання технологічних мікро-сервісів аналізу цифрових пристроїв з одночасною візуалізацією схем, тестів, результатів моделювання і таблиць істинності функціональних елементів.

Задачі: 1) Створення структури хмарного моделювання цифрових пристроїв на платформі компьютингових сервісів Google. 2) Розробка модуля (мікросервіса) Q-element, що реалізує створення моделі та візуалізацію логічного або функціонального примітиву схеми. 3) Проектування модуля генерування кубітних моделей примітивів і більш складних цифрових пристроїв, а також засобів їх оперативної візуалізації. 4) Розробка модуля або панелі управління, що інтегрує симулятор, генератор логічних елементів, схемних конструкцій і вхід-вихідних портів. 5) Розробка власне модуля для аналізу цифрових схем на основі рекурсивної обробки логічних елементів. 6) Створення службових бібліотек для зберігання: готових функціональних елементів, складних цифрових схем, тестів і результатів їх аналізу, а також службової інформації. 7) Тестування і верифікація хмарного сервісу Q-simulator, призначеного для інтерпретативного моделювання цифрових пристроїв.

Сутність квантового методу аналізу полягає в адресній реалізації всіх функціональних компонентів цифрових систем і структур даних, що дає можливість істотно підвищити швидкодію інтерпретативного моделювання

та якість обслуговування проектів за рахунок швидкої заміни некондиційних логічних елементів шляхом їх переадресації.

Структура хмарного сервісу містить наступні основні мікросервіси: 1) Q-element генерує квантові описи логічних елементів в структурі цифрової функціональності. 2) Модуль View виконує візуалізацію схемних елементів, портів входів і виходів на екрані монітора. 3) Модуль Collapse керує вікнами монітора та їх розмірами за допомогою відповідних іконок. 4) Контролер Split візуалізує роботу всіх контролерів при складанні схеми на екрані, а також здійснює масштабування деталей проекту. 5) Функція Evaluate формує стан виходу поточного елемента шляхом вибору вмісту з комірки кубіта за її адресою. 6) Модуль Q-sim реалізує власне алгоритм моделювання всіх ліній схеми шляхом побудови на першому кроці рекурсивної моделі для послідовно-паралельної обробки елементів. На другому кроці обчислюються стани всіх виходів логічних елементів. Моделювання закінчується після обробки всіх тестових вхідних послідовностей. Якщо схема має глобальні або локальні зворотні зв'язки, то моделювання здійснюється до фіксації однакових значень сигналів на всіх лініях схеми. Якщо схема не встановлюється в стійкий стан на вхідних наборах, то фіксується генераторний режим після виконання n (= 20) ітерацій. В цьому випадку всім постійно змінюваним лініям присвоюється значення двійкової невизначеності X={0,1}. 7) Модуль керування бібліотекою елементів, схем і проектів здійснює зчитування, запис і підключення фрагментів. 8) Модуль часових діаграм здійснює візуалізацію тесту з вихідними сигналами на моніторі в формі безперервних сигналів, розділених на такти в абсолютному або модельному часі. 9) Всі модулі хмарного сервісу запрограмовані мовою Swift, з використанням операційної системи OSX 10.9, компілятора XCode 7. Кількість вихідних файлів 36, загальна кількість рядків коду - 1450.

На рис. 8.11 наведено візуалізацію результатів графічного проектування схеми з тригерами на моніторі комп'ютера. Схема повністю відповідає функціональності, представленої на рис. 8.3. Вона містить чотири

вхідних порти для подачі робочих або тестових впливів, а також два вихідних порти. Структура містить чотири логічних елементи і два тригери. Мнемонічний опис компонентів схеми приведено до універсальної формі прямокутника, він різниться тільки номером примітиву в складі пристрою, а також типом функціональності, яка задається кубіт-вектором, представленим десятковим числом. У схемі, представленій на рис. 8.11, елементи мають порядкові номери (у верхній частині) і цілі числа для ідентифікації функциональностей: 0/6 - 0110, 1/1 - 0001 2/1 - 0001 3/7 - 0111, 4/143 - 11110001, 5 / 226 - 01000111. Тут двійковий вектор відповідає десятковому еквіваленту числа для завдання функціональності. Оскільки кубіт-вектор не має в явному вигляді задання вхідних наборів, то його можна розглядати як неявну або компактну форму теоретико-множинної по суті таблиці істинності. Навіщо явно вказувати вхідні значення, якщо вони складають строго послідовну адресацію вихідних значень. Таким чином, таблиця істинності, як сукупність вхідних сигналів і відповідних їм вихідних значень завжди програє перед кубіт-векторною формою подання функциональностей в плані обсягу і швидкодії аналізу даних. Не існує принципових відмінностей між описами комбінаційного елемента, схеми або послідовностного примітиву, оскільки всі вони формально представлені кубіт-векторами, які розміщуються в адресовній пам'яти. Більш того, всі примітиви схеми також є адресовними, а структура схеми також може бути описана у вигляді кубіт-вектора. Таким чином, можна прийти до такої реалізації обчислювального пристрою, де немає нічого крім адресної пам'яті або кубітних векторів різної довжини, в яких функціональності визначаються впорядкованими наборами нульових і одиничних сигналів. Переваги даного сервісу QuaSim кубітного опису і моделювання цифрових пристроїв полягають в наступному: 1) усі функціональні елементи і схеми задаються Q-векторами, що уніфікує процедури синтезу та аналізу цифрових пристроїв; 2) технологічно просто міняти або корегувати функціональність схеми або будь-якого примітиву шляхом заміни окремих бітів Q-вектора; 3) уніфікація кубітної форми опису

примітивів схеми дає можливість застосувати до них єдину процедуру аналізу функціональностей, яка зводиться до обчислення адреси $M_i = Q_i[M(X_i)]$, що робить процес програмування хмарного сервісу QuaSim технологічно простим у виконанні і не залежним від функціональної та структурної складності цифрових структур; 4) простий і зрозумілий початківцю графічний інтерфейс робить хмарний сервіс конкурентоспроможним на ринку освітніх послуг, де складні та важкі засоби моделювання від провідних компаній планети є недоступними для університетів у зв'язку з їх високою вартістю, а для студентів – часовитратними за складністю підготовки HDL-специфікацій при розгляді невеликих навчальних проектів; 5) уніфікація форми опису примітивів створює умови для технологічного вирішення задач синтезу, моделювання несправностей, тестування, верифікації та діагностування, на основі операцій з кубіт-векторами; 6) недоліком кубітної або квантової технології опису та аналізу цифрових структур можна вважати деяке зменшення швидкодії моделювання в порівнянні з існуючими промисловими компіляторами, для ASIC і VLSI проектів, де обсяг reusable logic є домінуючим для досягнення високої швидкодії.

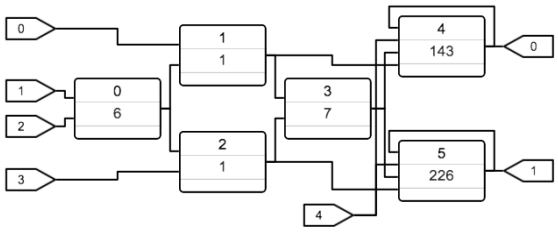



Рисунок 8.11 – Скріншот структури з тригерами

Структура взаємодіючих компонентів хмарного сервісу QuaSim представлена на рис. 8.12. Квантове або кубітне представлення моделі цифрового пристрою разом з інтерпретативним симулятором складають ядро системи, інтегрованої у великі дані кіберпростору або Інтернету. Це дає можливість використовувати в якості вихідних даних відкриті специфікації і тестбенчі, описані мовами VHDL, Verilog. Такі дані і/або тестові приклади є

практично у всіх провідних компаніях, університетах і тематичних конференціях IEEE, TTTC, ISCAS. Крім того, занурення QuaSim сервісу в інтернет-простір передбачає також вивантаження результатів його роботи, пов'язаних з аналізом і синтезом навчальних або ринково орієнтованих проектів в сервіси зберігання даних на платформах Google, Amazon, Microsoft, IBM, Facebook. Природно, що інтеграція хмарного сервісу з кіберпространством передбачає наявність парсер-мікросервісів для перетворення специфікацій з мов опису апаратури у внутрішню мову QuaSim, а також має існувати і зворотнє перетворення даних з кубітного представлення в стандарти HDL-мов. Парсерізація забезпечує можливість використання відкритих в інтернеті проектів для їх вивчення і порівняння в системі моделювання QuaSim, а також робить доступними внутрішні проектні рішення QuaSim для всіх бажаючих на ринку освітніх сервісів.

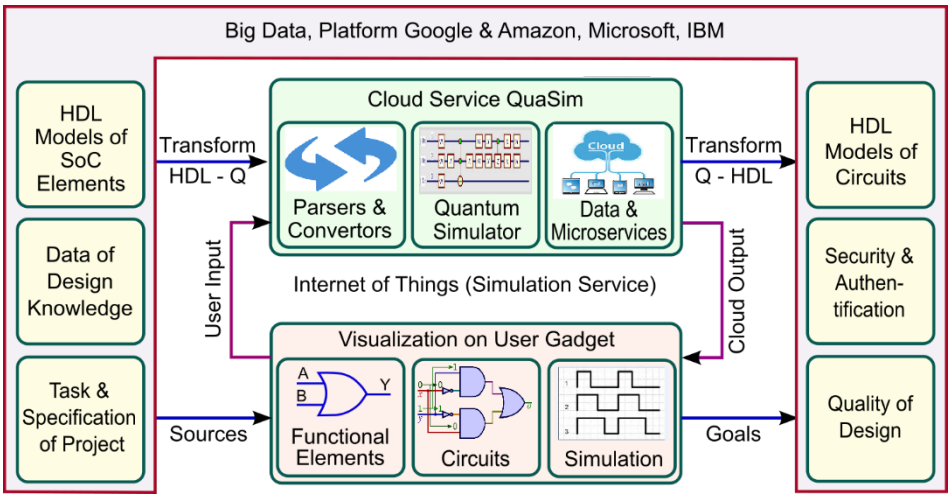



Рисунок 8.12 – Хмарний сервіс моделювання цифрових пристроїв

Блок Security контролює доступ користувачів з метою їх статистичного обліку і передбачає аутентифікацію кожного на основі пароля, прізвища, імені, доповнене будь-яким дійсним (корпоративним) атрибутом зі списку: {електронний цифровий підпис, e-mail, цифровий ключ, номер телефону}.

Тестування і верифікація хмарного сервісу моделювання цифрових систем здійснювалася окремо для кожного мікросервісу, а потім у взаємодії всіх модулів: 1) перевірка правильності генерування логічних і більш

складних функціональних елементів; 2) перевірка структурного синтезу цифрової схеми і засобів візуалізації; 3) верифікація і тестування алгоритмів двійкового і трійкового синхронного інтерпретативного справного моделювання вхідних впливів на 40 схемах, комбінаційного та послідовностного типів. При цьому використовувалися тестові набори, алгоритмічно генеровані і складені користувачем; 4) перевірка сервісних модулів, що забезпечують працездатність основних мікросервісів: бібліотеки елементів і схем, аутентифікація користувача, модуль формування статистичних даних за проектами і користувачами; 5) верифікація інтерфейсних мікросервісів, що забезпечують інтеракції між хмарними back-end і призначеними для користувача front-end модулями.

## 8.10 Хмарний сервіс тестування

Представлена доменно і IP-ідентифікована структура взаємодії кіберфізичних компонентів, що реалізують хмарний комп'ютинг-сервіс моделювання і тестування цифрових пристроїв в контейнерно-орієнтованому середовищі компанії Docker, що працює в середовищі Google Computing Engine.

Мета – зменшення часу проектування і верифікації мікросервісів за рахунок контейнерної технології, а також популяризація нових методів кубітного синтезу та аналізу цифрових пристроїв в середовищі фахівців, завдяки своїй хмарної доступності.

Задачі: 1) Створення хмарного середовища проектування та верифікації на основі контейнерів. 2) Розробка мікросервісів моделювання і тестування цифрових пристроїв. 3) Верифікація структури і функціональностей хмарного сервісу.

Дослідження ринку проектування сучасних програмних продуктів показали, що в останні три роки набирають популярність гнучкі і масштабовані технології проектування програмних продуктів, що використовують контейнеризацію, орієнтовані на мінімізацію циклу time-to-

maret. Основна ідея контейнера полягає у створенні захищеного і доступного віртуального середовища проектування і експлуатації в кіберпросторі, на основі використання локальної або глобальної апаратури Infrastructure, рис. 8.13.

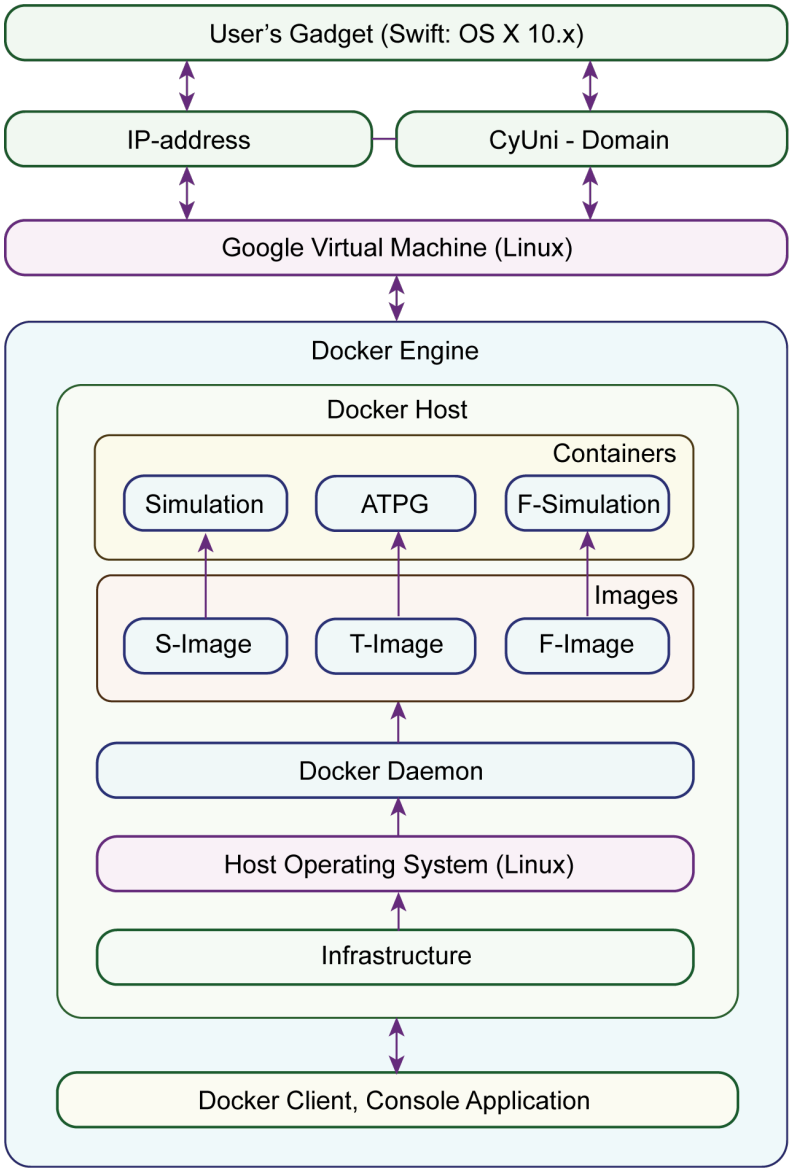



Рисунок 8.13 – Хмарний сервіс тестування і верифікації цифрових пристроїв

На інфраструктуру встановлюється практично будь-яка операційна система Host operating system, зручна для розробника, наприклад, Ubuntu, Debian. У локально виконаній hardware інфраструктурі контейнерів можна інсталювати також OSX і Windows.

Далі розробник інсталює Docker Engine, якщо він не встановлений спочатку, який за допомогою Docker Daemon (реалізація на Golang, C), керує контейнерами, запущеними на основі попередньо побудованих образів (Images). Контейнери Containers являють собою легку віртуальну машину, на якій працюють функціональні блоки: Simulation, ATPG, F-simulation, реалізовані на Swift (або Jawa), і відповідні бібліотеки. Docker Engine представляє собою компьютингову систему контейнеризації, яка складається з двох взаємодіючих між собою компонентів: Docker Host, Docker Client (Console Application), які можна інтерпретувати як виконавчий Host і керуючий (Client) механізми хмарної системи тестування і моделювання. Далі Docker Engine є компонентом, який входить до складу Google Virtual Machine, якій ставиться у відповідність IP-address and CyUni domain, що робить розробку, а далі хмарний сервіс, доступним для фахівців і студентів.

Останньою ланкою у вертикальній зв'язці (Back-End, Front-End) є User's Gadget (Swift OS X.10x). На кінцевому пристрої користувача установлюються додатки, здатні привести результати роботи хмарних сервісів моделювання і тестування до мультивіконного формату візуалізації даних, зручного для сприйняття людиною. Слід зазначити, що набагато більш технологічним є створення Front-End додатку на JavaScript, HTML, CSS, який буде працювати через Internet browser. Це гарантує відсутність блокування клієнта з боку постачальника конкретної ОС (vendor lock). Для мобільних пристроїв доцільніше використовувати додаток індивідуального клієнта під кожну ОС.

Переваги структурної організації хмарного сервісу тестування і моделювання цифрових пристроїв: 1) захищеність і доступність контейнерів для командної розробки і верифікації моделей і методів синтезу та аналізу обчислювальних виробів; 2) глобальна доступність сервісів в кіберпросторі планети для вивчення і застосування інноваційних технологій проектування, що базуються на кубітно-векторних моделях опису цифрових компонентів; 3) контейнерна розширюваність хмарного комп'ютингу при реалізації нових мікросервісів, відповідних додаткових методів і алгоритмів тестування і

моделювання; 4) просторово-часова інваріантність проектування і експлуатації кібер-фізичної системи, спрямована на доступність хмарних сервісів з будь-якої точки планети в будь-який час доби; 5) інфрструктурна незалежність хмарного віртуального комп'ютингу забезпечує безвідмовну роботу невидимої апаратури, її своєчасну замінюваність на стороні Google (Amazon, IBM) без додаткових фінансових витрат для власника сервісів.

Тестування і верифікація хмарних сервісів кубітного синтезу тестів, аналізу несправностей і справної поведінки було виконано на стандартних прикладах Benchmark circuits (c5, c17, c432, c499, c880, c1355, c1908, c2670, c3540, c5315) з бібліотек конференції ISCAS. Приклад опису схемної структури c17 на мові Verilog представлений у вигляді наступного лістингу:

```
// Verilog
// c17
// Ninputs 5
// Noutputs 2
// NtotalGates 6
// NAND2 6
module c17 (N1, N2, N3, N6, N7, N22, N23);
input N1, N2, N3, N6, N7;
output N22, N23;
wire N10, N11, N16, N19;
nand NAND2_1 (N10, N1, N3);
nand NAND2_2 (N11, N3, N6);
nand NAND2_3 (N16, N2, N11);
nand NAND2_4 (N19, N11, N7);
nand NAND2_5 (N22, N10, N16);
nand NAND2_6 (N23, N16, N19);
endmodule
```

Результати порівняльного тестування програмної реалізації методів за основними параметрами: час моделювання справної поведінки і несправностей, представлено діаграмами на рис. 8.14 і 8.15.

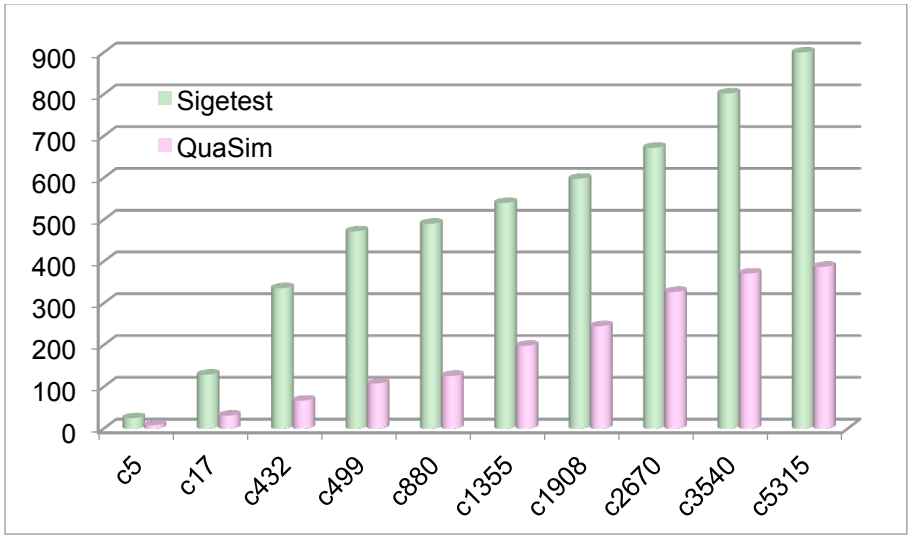



Рисунок 8.14 – Порівняльний аналіз двох методів справного моделювання

Моделювання здійснювалося для схем (c5, c17, c432, c499, c880, c1355, c1908, c2670, c3540, c5315) на 10 000 вхідних наборів: 1) базовий метод – програмний продукт SiGeTest [204]; 2) кубітний метод моделювання, реалізація QuaSim. Одиниця виміру – мілісекунда. Порівняльний аналіз синтезу тестів не проводився, оскільки новий метод синтезу тестів реалізований тільки для black box функциональностей, заданих кубітним покриттям.

Виграш у швидкодії справного моделювання на 10 прикладах цифрових схем – не менше, ніж в два рази. Обидва методи використовують інтерпретативний алгоритм, орієнтований на імплементацію в програмовну логіку. Перевага програмної реалізації зазначеного алгоритму полягає в інтерактивному характері модифікації схемної структури в процесі виконання програми моделювання.

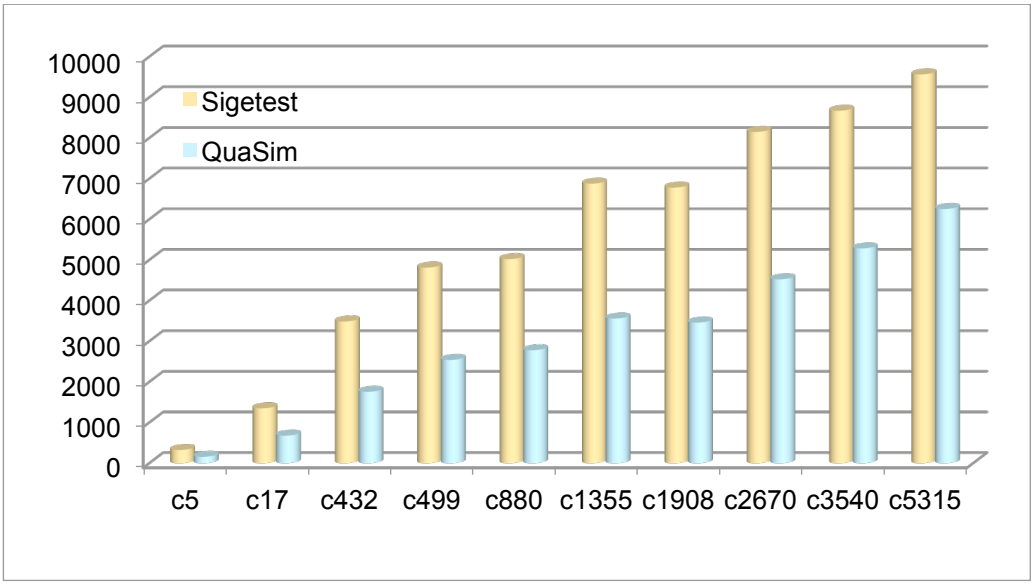



Рисунок 8.15 – Порівняльний аналіз двох методів моделювання несправностей

Дедуктивні методи моделювання несправностей, при їх програмній реалізації, на порядок складніші, ніж алгоритми справного моделювання. Тому кількість мілісекунд тут на порядок більше, ніж на попередній діаграмі. Діаграма показує також, що за швидкодією майже в два рази виграє нова реалізація дедуктивного методу QuaSim, яка використовує паралельні операції над кубітами функціональних елементів.

Таким чином, порівняльний аналіз нових методів моделювання, в порівнянні з базовим, показав суттєві переваги у швидкодії, які, поряд з високою технологічністю паралельних операцій, ставлять їх у ряд перспективних напрямів memory-driven проектування цифрових систем на основі використання хмарного комп'ютингу. Інтеграція моделей і методів кубітного аналізу із середовищем моделювання Riviera фірми Aldec Inc представлена на рис. 8.16. Нові структури даних і методи синтезу тестів, справного моделювання та аналізу несправностей, додані в систему, дозволяють удосконалити існуючий процес верифікації та на 12% скоротити час розробки цифрового виробу (time-to-market).

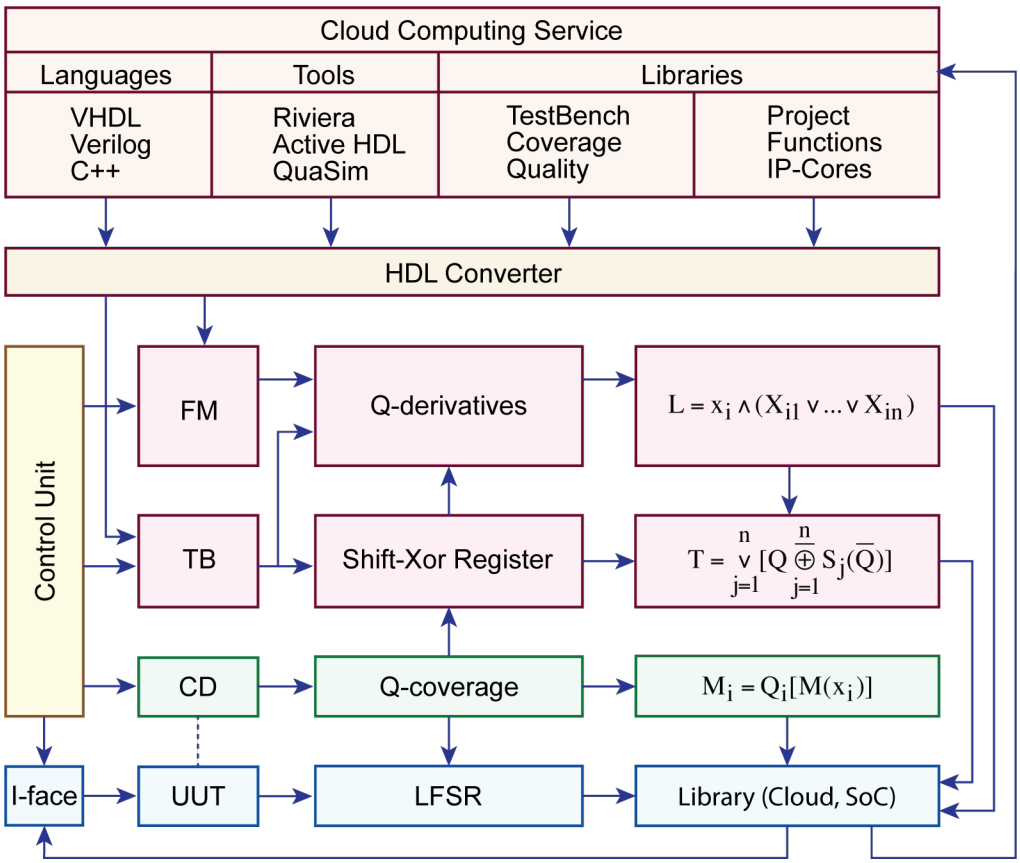



Рисунок 8.16 – Інтеграція методів з продуктом Riviera, Aldec

Таким чином, кіберфізична організація взаємодії користувачів з хмарним компьютинговим сервісом є найбільш перспективною формою проектування, тестування цифрових виробів, яка орієнтована на вільний доступ до віртуальних кіберпродуктів з будь-якої точки фізичного простору.

## 8.11 Висновки до розділу 8

Запропоновано новий підхід до проектування цифрових пристроїв, який характеризується: 1) удосконаленим методом інтерпретативного паралельного моделювання компонентів цифрових систем на кристалах, який відрізняється застосуванням автоматної МАТ-моделі, що використовує тільки адресовні структури пам'яті і операції транзакції; 2) методами синтезу та аналізу, що базуються на суперпозиції кубіт-векторних примітивів завдання всіх типів функциональностей, імплементованих в елементи пам'яті, що дає можливість істотно підвищити швидкодію засобів моделювання,

тестування і верифікації, а також значно спростити процедури створення реальних і віртуальних комп'ютерних систем; 3) оригінальними структурами даних для моделювання і верифікації цифрових систем, які дають можливість суттєво спростити алгоритми реалізації та підвищити їх швидкодію за рахунок адресовних функціональних квантів і паралельності обробки компонентів; 4) реалізацією всіх обчислювальних структур і процесів на основі використання введеного квантового адресного автомата, який дає можливість безпосередньо використовувати інфраструктуру стандартів тестопридатного проектування для підвищення виходу придатної продукції за рахунок online ремонту функціональних примітивів.

Практична значущість нового підходу синтезу та аналізу цифрових систем полягає в наступному: 1) реалізація процесора тільки на основі використання елементів пам'яті робить його однорідним за структурою і типам функціональних примітивів, що доставляє очевидні технологічні зручності процесам проектування, виробництва і експлуатації, у тому числі верифікації, вбудованого тестування і діагностування, а головне – ремонту в режимі online за рахунок використання на кристалі універсальних адресовних spare-компонентів пам'яті; 2) моделювання в процесі верифікації проектованих обчислювачів на основі адресних моделей компонентів робить дану процедуру технологічно простою за рахунок регулярних структур даних і використання єдиної операції транзакції на елементах пам'яті, а також більш швидкодіючою за рахунок можливості паралельної квантоподібної обробки великих масивів однотипної пам'яті; 3) імплементація квантових only memory-based моделей опису цифрових компонентів і систем безпосередньо пов'язана зі збільшенням виходу придатної продукції, підвищенням надійності обчислювальних виробів, зниженням вартості проектування і виготовлення, а також автономним відновленням працездатності в режимі remote & online, без участі людини.

Практичні висновки за елементами кубітної теорії комп'ютингу. Згідно квантової теорії, електрон може перебувати в двох місцях одночасно. Дане твердження не суперечить відомому рівнянню Шредінгера, де координата кіберфізічного простору здатна мати в один і той же час два електрони або два стійких стани. Кубіт $Q = (Q_1, Q_2, \dots, Q_i, \dots, Q_n)$ – дискретний стан точки у кіберфізичному просторі, який визначається суперпозицією скінченної кількості примітивів (носіїв) універсуму, але представлених вектором. Квант функціональності (XQY) – універсальна за формою елементарна структура (частка), призначена для створення як завгодно складних компьютингових систем. Квант функціональності – кубіт з ідентифікаторами вхідних і вихідних змінних, які формують адреси примітивів (комірок пам'яті) для цілеспрямованого виконання транзакцій (зчитування-запис). Квант функціональності призначений для реалізації як завгодно складних memory-driven обчислювальних архітектур, вільних від логіки. Структура пам'яті, що використовує манхеттенську або трикутну топологію, створює квазіоптимальні умови для підвищення швидкодії адресних транзакцій між кубіт-векторами функциональностей. Послідовність обробки квантів функциональностей обчислювальної архітектури задається кубітом алгоритму, заданого орієнтованим графом, який також реалізується на пам'яті. Створюються два memory-driven механізми керування і виконання компьютингової системи, що використовує тільки кубітну векторну форму опису функцій і структур. Адреса доступу до комірки кубіта формується на основі конкатенації станів вхідних змінних (X) кванта функциональностей. Вміст розряду кубіта Q[M(X)] за сконкатенованою адресою M(X) визначає стан кванта функціональності, який записується в комірку (Y) вектора стану (M) обчислювальної системи. Таким чином, computing (рис. 8.17), глобально і локально, віртуально і фізично, програмно і апаратно, являє собою реалізацію в пам'яті тривіальної транзакції: M(Y)=Q[M(X)] – рівняння Memory-Address-Transaction комп'ютингу. І нічого більше.

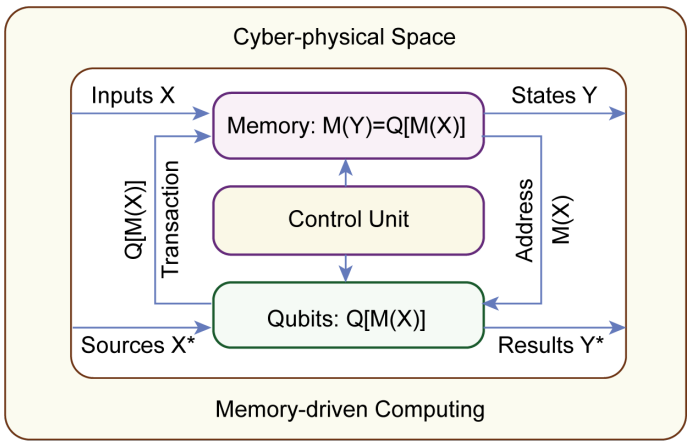



Рисунок 8.17 – MAT-комп'ютинг

Іншими словами, будь-який обчислювальний процес зводиться до адресних операцій зчитування-запису на пам'яті. Практична значущість даної парадигми полягає у виключенні апаратної логіки з комп'ютингу, що дає можливість: 1) зробити регулярною архітектуру комп'ютера на основі використання тільки матриць пам'яті; 2) виконувати всі арифметичні, логічні та спеціальні операції на матричній структурі пам'яті, без звернення до зовнішнього модулю АЛУ, якій значно зменшує швидкодію; 3) технологічно вирішувати всі проблеми надійності комп'ютерних систем на основі вбудованого online тестування і ремонту відмовних модулів пам'яті шляхом їх переадресації на справні з ремонтного запасу; 4) мати практично необмежені можливості в межах одного кристала для паралельного виконання логічних операцій будь-якої регістрової розмірності; 5) істотно зменшувати час проектування спеціалізованих процесорів за рахунок виключення складних і нерегулярних блоків АЛУ з шинами обміну інформацією; 6) перепрограмувати в режимі online логіку елементів пам'яті для виконання інших операцій.

В даний час спостерігається стійке домінування інноваційної кіберкультури, що містить нано-Адитивні, біо-інформаційні, кібер-фізичні, соціально-когнітивні технології, на ринку науки, освіти, індустрії, транспорту і забезпечення якості життя.

Основні атрактори досліджень пов'язані з ключовими словами: Qubit Function, Qubit Structure, Qubit Coverage, Qubit Vector, Quantum of Functionality, Memory-Address-Transaction, Quantum Simulation, Qubit Design and Testing, Memory-Driven Computing, Cloud Computing, IoT Computing, Big Data Computing, Cyber-Physical Computing and Cosmological Computing.

В роботі вирішено проблему усунення протиріччя між двовимірним класичним описом функцій і структур і одновимірним поданням регістрових змінних в апаратному компьютингу за рахунок приведення функцій і структур до одновимірної кубітно-векторної метрики в цілях технологічного вирішення задач синтезу і аналізу цифрових систем на основі паралельних логічних операцій, що створюють memory-driven computing.

Сутність представленого науково-технологічного дослідження полягає у створенні векторних структур даних і кубітних методів синтезу, тестування і моделювання, інтегрованих в хмарну інфраструктуру сервісного обслуговування компонентів цифрових систем на кристалах з метою підвищення якості комп'ютингу і виходу придатної продукції за рахунок адресовності всіх обчислювальних процесів і явищ.

Основна інноваційна ідея запропонованої MAT-моделі комп'ютингу полягає в синтезі і аналізі векторних цифрових структур на основі адресовних елементів пам'яті, що виключають використання reusable or new logic.

Мета дослідження була спрямована на суттєве підвищення виходу придатної програмно-апаратної продукції і якості комп'ютингу за рахунок створення інфраструктури хмарних сервісів моделювання, тестування і відновлення працездатності на основі кубітно-векторних структур даних опису функціональних елементів і підвищення швидкодії кубітних методів синтезу та аналізу.

# 9 ПЕРЕВАГИ ХМАРНИХ СЕРВІСІВ КІБЕРСИСТЕМИ

Впровадження СТС надасть можливість суттєво поліпшити комфортність пересування водіїв і пасажирів, зберегти сотні мільярдів доларів, матеріалів і часу, а також зберегти екологію планети для майбутніх поколінь. Переваги впровадження точного цифрового моніторингу і хмарного управління транспортом мають глобальний і локальний характер і полягають у наступному:

1) Збереження екології планети і регіонів за рахунок зменшення забруднення навколишнього середовища, підвищення якості життя людей за кермом, економія паливних ресурсів і скорочення часу руху завдяки вибору оптимального маршруту, зменшення кількості і складності трафіку за рахунок розумної інфраструктури та інтелектуальних світлофорів.

2) Повне виключення дорожньої поліції для ідентифікації автомобілів, які порушують правила дорожнього руху, точний моніторинг позиціонування транспортних засобів в часі і просторі, включаючи викрадення, колізії, несанкціоновані маршрути. Дистанційне керування через хмарні сервіси екстреним вимиканням двигуна автомобіля, при створенні реальної небезпеки іншим учасникам дорожнього руху. Істотне зниження аварійності за рахунок моніторингу та прорахунку рівня безпеки маневрів, зменшення наслідків дорожньо-транспортних пригод, підвищення безпеки і комфорту учасників дорожнього руху.

3) Моніторинг позиціонування і пересування транспортних засобів в автомобільних компаніях, оптимальне виконання замовлень з перевезення пасажирів і вантажів з позиції мінімізації матеріальних і / або часових витрат.

4) Надання сервісів водієві, пов'язаних з прокладанням оптимальних маршрутів і графіка руху з урахуванням негативних дорожніх факторів в існуючій інфраструктурі для мінімізації матеріальних і часових витрат в online режимі; істотне зниження аварійності за рахунок цифрового

моніторингу дорожньої обстановки, закритою для візуального перегляду ділянок руху і прорахунку рівня безпеки маневрів.

5) Надання сервісів пасажирам з моніторингу позиціонування і руху пасажирських транспортних засобів на зупиночних або транспортних терміналах завдяки використання стаціонарних моніторів або мобільних гаджетів, пов'язаних з відповідними хмарними сервісами. Візуалізація на моніторі автомобіля критичних точок маршруту руху транспортного засобу в реальному часі шляхом використання камер відеоспостереження. Замовлення гібридного маршруту руху до пункту призначення шляхом використання різних транспортних засобів.

6) Автоматична і точна цифрова реєстрація в часі і в просторі всіх порушень правил дорожнього руху та дорожньо-транспортних пригод, які тягнуть зняття з рахунку грошового еквівалента штрафів і формують історію водія в страховій компанії, що дає можливість виключити участь поліції в процедурах огляду ДТП.

## 9.1 Технічні і функціональні можливості TCS

1) Моніторинг (24/7) реальної швидкості руху всіх транспортних засобів та інформування про зони швидкісного режиму. Цифровий моніторинг проїзду на заборонні знаки і сигнали світлофорів.

2) Економія палива, зменшення забруднення навколишнього середовища, скорочення часу руху, завдяки вибору оптимального маршруту, наданого хмарою.

3) Зменшення кількості та складності пробок за рахунок планування руху транспортних засобів, що враховує майбутні маршрути учасників. Коригування маршруту руху транспортного засобу в реальному часі при зміні дорожньої ситуації.

4) Інтелектуальне управління циклами перемикання світлофорів залежно від дорожньої обстановки на перехрестях.

5) Генерування аналітичних, статистичних звітів та рекомендацій щодо поліпшення інфраструктури доріг, розстановки віртуальних знаків, світлофорів та централізоване програмування циклів їх перемикання.

6) Протидія угону транспортного засобу і самовільне залишення місця ДТП, завдяки моніторингу місця розташування кожного автомобіля. Хмарна цифрова реєстрація для страхових компаній всіх необхідних деталей і динаміки ДТП, не пов'язаних з травмами, без участі дорожньої поліції.

7) Інформування за допомогою віртуальної тривожної кнопки спеціальних служб про події, що сталися на дорогах або з автомобілем.

8) Попередження водія про потенційну небезпеку на замовленому маршруті, інформація про яку отримана від хмари в процесі руху.

## 9.2 Проблеми, які вирішуються за допомогою цифрової ідентифікації

Уже сьогодні існує досить багато актуальних і практично орієнтованих задач, які можна вирішити за допомогою радіоцiфрових паспортів:

1. Ідентифікація виробу (об’єкта або суб’єкта) в локальній або глобальній системі координат.
2. Збереження параметрів, що характеризують основні властивості об'єкта.
3. Накопичення і збереження історії життєвого циклу об'єкта.
4. Передача інформації про об’єкт або явище за санкціонованою вимогою в хмару управління.
5. Прийом санкціонованої інформації, що дає можливість модифікувати окремі властивості електронного паспорта об’єкта.
6. Санкціонована взаємодія з електронними паспортами (гаджетами) інших об’єктів, що знаходяться в полі радіовідимості об’єкта.
7. Передача інформації про взаємодію об’єкта з іншими ідентифікаторами в межах радіовідимості.

Таким чином, електронний цифровий ідентифікатор об’єкта є автономною цифровою системою на кристалі з прийомо-передавачем, який

здатний зберігати інформацію про ідентифікований об'єкт, модифікувати її за командою центру управління, а також зберігати інформацію про взаємодії з навколишнім середовищем і іншими цифровими ідентифікаторами з можливістю передачі даних в хмару управління. Інші варіанти ID комунікацій пов’язані з використанням: 1) мережі мобільної телефонії; 2) супутникових систем для прийому і передачі інформації.

## 9.3 Аргументи проти впровадження хмари в масштабах країни

1. «Порушення права на недоторканність приватного життя, оскільки теоретично TCS хмара здійснює тотальний моніторинг всіх транспортних засобів». Насправді сьогодні існує система законного перехоплення телекомунікацій, реалізована відповідно до міжнародних вимог. Перехоплення телефонних переговорів абонента використовується тільки в ході слідства і з санкції суду. Є можливість відстежувати місце знаходження абонента спеціальними службами. Даний факт для законослухняних громадян ніякої проблеми не створює, якщо служби укомплектовані чесними чиновниками.

2. «Додаткові витрати на придбання апаратно-програмних мобільних гаджетів ідентифікації і орендна плата, близько 100 доларів в рік, за використання сервісів TCS-системи». Уже сьогодні практично всі мешканці планети мають такі пристрої, а економічні переваги хмари, пов'язані з економією палива і зменшенням часу поїздки цілком гідно компенсують витрати на придбання сервісів.

3. «Ненадійність віртуальних світлофорів, пов’язана з падінням хмарного сервісу». Однак, реальні світлофори більш уразливі і менш надійні, ніж віртуальні, з причин відключення електроенергії, фізичних порушень через аварії, відмов компонентів світлофорів. Віртуальні світлофори знищити практично неможливо.

## 9.4 Практичні приклади впровадження компонентів TCS-системи

1. Програмний додаток управління корпоративними перевезеннями використовується для оптимального планування рейсів з доставки вантажів, що приводить до зменшення часових і матеріальних витрат за рахунок: 1) зниження витрат на паливно-мастильні матеріали; 2) оптимального розподілу замовлень між автомобілями; 3) прогнозування поставок товарів для зменшення складських витрат; 4) економії робочого часу персоналу або скорочення штатних співробітників; 5) зменшення числа автомобілів для виконання заданого обсягу перевезень; 6) моніторингу та оперативного управління автомобілями при доставці вантажів в реальному часі. Ринкова привабливість хмарного сервісу транспортної логістики: оптові компанії, регіональні дистриб'ютори продовольчих і промислових товарів (хлібозаводи, молокозаводи, м'ясокомбінати, пиво-безалкогольні комбінати, промислові підприємства, автотранспортні підприємства, торгові мережі, логістичні оператори, транспортно-експедиторські компанії, вендингові компанії, швидка допомога, інкасаторські служби, кур'єрські служби, інтернет-магазини, клінінгові компанії).

2. Дистанційний модуль "SHERLOCK" призначений для побудови розподілених систем моніторингу та управління об'єктами, включаючи мобільні. Являє собою електронний виріб, побудований на основі використання трьох новітніх технологій Mobile-to-Mobile, GPS і GPRS. Задачі, які вирішуються за допомогою модуля: 1) автоматичне визначення місцезнаходження транспортних засобів; 2) управління автомобілями транспортного парку, логістика; 3) автоматизація служб таксі; 4) моніторинг маршруту і розкладу руху транспорту; 5) моніторинг режимів експлуатації транспортних засобів. Технічні характеристики: GPS – багатоканальний приймач з високою чутливістю і малим енергоспоживанням, спеціально призначений для роботи в умовах міської забудови та наявності відбитих сигналів http://gps.rfid.com.ua. Для отримання доступу до сервісу користувач повинен авторизуватися за допомогою логіна і пароля. На головній сторінці

сервісу більшу частину площі екрану займає вікно з картою, на яку накладаються дані про місцезнаходження мобільних об’єктів. Для візуалізації використовується картографічна інформація компанії ВІЗІКОМ. При візуалізації на карті піктограмами відображається стан об’єктів і маршрут руху за обраний інтервал часу, а також тривалі стоянки. Розмір і положення карти можна змінювати за допомогою миші і елементів управління. У нижній частині головної сторінки розташовуються елементи управління, що дозволяють швидко перемикатися між частинами маршруту і між об’єктами, а також статистична інформація. При виборі відображення тільки одного об’єкта доступна функція розрахунку відстані. Комплект поставки телеметричного модуля: антена GPS; антена GSM; кабель з’єднувальний; інструкція з експлуатації; SIM-карта.

## 9.5 Технології високоточної ГНСС-навігації транспортних засобів

Навігація – визначення місця розташування об’єкта, що рухається в заданій системі координат, а також вектора швидкості і кутової орієнтації з точною прив’язкою часу до навігаційних параметрах. Оскільки функції кіберсистеми – не тільки моніторинг транспортних засобів, а й управління їх рухом, то вимоги до точності і надійності навігаційних операцій повинні задовольняти самим сучасним і перспективним вимогам державних радіонавігаційних планів, що пред’являються до критичних об’єктів управління в типових і в складних дорожніх умовах. При цьому для управління використовуються сигнали глобальних навігаційних супутникових систем (ГНСС) – GPS (США), ГЛОНАСС (РФ), BeiDou / Compass (Китай), Galileo (ЄС), також їх регіональних функціональних доповнень (РФД) - WAAS (США), EGNOS (ЄС), MSAS (Японія), QZSS (Японія), GAGAN (Індія). Як показує аналіз, основні вектори розвитку навігаційного ринку в світі визначаються застосуванням супутникових навігаційних технологій на транспорті, в першу чергу автомобільному. Хорошим прикладом використання супутникових технологій слугує проект

екстреного реагування при аваріях на дорогах «ЕРА-ГЛОНАСС» (РФ). Перспективи навігаційних супутникових технологій – у високоточній навігації транспорту, які стануть масово і матеріально доступними через 5 років завдяки диференційним технологіям RTK (Real Time Kinematic), PPP-RTK визначення траєкторії руху об’єкта з сантиметровою (міліметровою) точністю. Метод, який використовує широкозонні корекції регіональних функціональних доповнень – WAAS (США), EGNOS (ЄС), MSAS (Японія), QZSS (Японія), GAGAN (Індія) формує WADGNSS-системи (Wide Area Differential Systems). Тут виконується трансляція із супутників диференційних поправок, що дозволяє підвищити точність визначення планових координат до 1,0 м. Традиційний диференційний метод корекції похибок ГНСС-вимірювань шляхом використання диференційних корекцій від окремих ГНСС-станцій дозволяє збільшити точність визначення планових координат до рівня 0,5-0,8 метрів з меншою надійністю навігаційних визначень в порівнянні з методом широкозонной навігації.

Бортове навігаційне обладнання транспортного засобу – радіонавігаційні приймач і радіомодем має забезпечувати: 1) визначення координат об'єкту з точністю 0,5 ÷ 1,0 м в плані і 1,0 ÷ 2,0 м за висотою; 2) визначення складових вектора швидкості з точністю 0,05 ÷ 0,10 м/с; 3) визначення параметрів кутової орієнтації об’єкта з точністю 0,15 ÷ 0,30 кут. град. за курсом, або 0,5 ÷ 1,0 кут. град. по тангажу і / або крену; 4) визначення відходу бортової шкали часу з точністю 50 ÷ 100 нс. При управлінні рухомими об’єктами бортове навігаційне обладнання повинне забезпечити визначення координат динамічних об’єктів з точністю 0,05 м (в плані) і 0,1 м (по висоті). Зазначені вище значення точностних характеристик (рівні допустимих похибок) навігаційних визначень відповідають 95%-му вірогідністному довірливому інтервалу. Вимоги до надійності навігаційних визначень: 1) доступність або експлуатаційна готовність (Availability) – з ймовірністю 0,999; 2) цілісність (Integrity) включає «рівень тривоги» (Alert Limit) – 2 м і інтервал часу до повідомлення про вихід характеристик за

допустимі межі (Time to Alarm) – 5 секунд; 3) безперервність обслуговування (Continuity) – з ймовірністю 0,999/год. У разі необхідності отримання даних про параметри кутової орієнтації об'єктів можливо використовувати ГНСС-технологію визначення кутових параметрів спільно з інерційної підсистемою, наприклад, одного зі світових лідерів – компанії NovAtel. Реалізація технології RTK вимагає використання телекомунікаційних засобів (мобільний Інтернет, GSM/GPRS) для отримання диференційних корекцій від мережі референтних станцій. Така мережа вже існує і на Україні та забезпечує практично повне покриття її території. Навігаційна технологія WADGNSS використовує прийом поправок споживачем як від геостаціонарних супутників-ретрансляторів корекцій на частоті L1 GPS, так і по мережі Інтернет в умовах міських забудов і висотних будівель при маскуванні сигналів від супутників-ретрансляторів корекцій. У разі неможливості отримати RTK-корекції, наприклад, при перервах зв’язку з мережею референтних ГНСС-станцій або при виїзді об’єкта із зони дії найближчій ГНСС RTK-мережі, приймач споживача переходить в режим роботи WADGNSS, отримуючи сигнали корекції безпосередньо від геостаціонарних супутників на приймальню ГНСС-антену. Приймаються одночасно як сигнали ГНСС, так і коригувальні сигнали від супутників-ретрансляторів. При цьому точність навігаційних визначень зменшується з кількох сантиметрів до 1 метра за плановими координатами.

Інтегрування ГНСС + INS дозволяє виключити втрати навігаційної інформації там, де відсутня стеження за сигналами навігаційних супутників: під мостами, в тунелях, в умовах міських «каньйонів» і «допомогти» ГНСС-приймачу здійснити миттєве відновлення стеження за навігаційними сигналами ГНСС. Датчик ГНСС, в свою чергу, забезпечує отримання початкових умов (поточні координати і складові вектора швидкості об’єкта) для роботи інерційної підсистеми. Крім того, інтегрування ГНСС + INS в разі підтримки RTK-визначень дозволяє забезпечити необхідні характеристики (Availability, Integrity, Continuity) надійності навігаційних визначень. На

сучасному етапі окремо RTK-технологія не забезпечує необхідної надійності навігаційних визначень для задач кіберуправління. Інерціальні пристрої INS можуть включати в себе системи датчиків-акселерометрів і датчиків-гіроскопів в різних конфігураціях для визначення місця розташування та параметрів кутової орієнтації. В останні роки для транспортних додатків використовуються недорогі інерціальні мікромеханічні датчики MEMS, які інтегруються з ГНСС OEM-модулями.

Таким чином, найкращим варіантом побудови бортової підсистеми навігаційних визначень для моніторингу та управління транспортом є: 1) інтегрування ГНСС + INS на основі технологій MEMS (координатні і, за необхідністю, кутові визначення); 2) мульти-системність – одночасний прийом сигналів всіх чотирьох ГНСС + коригувальних сигналів РФД (WADGNSS); 3) RTK / WADGNSS / DGNSS технології точного позиціонування / навігації; 4) здійсненність кутових визначень за сигналами ГНСС з інтеграцією з INS. Устаткування, що реалізує перераховані функції з заданим якістю, існує, його ціна становить $ 15-20 тисяч. Через 5 років можна очікувати зниження ціни до рівня ~ $ 1,0 тисячі при масовому випуску апаратури. Розробку прототипу бортової апаратури управління автомобілем і проведення експериментальних досліджень в рамках проекту можуть виконати співробітники НДЛ «Супутникові мережеві технології високоточного позиціонування» ХНУРЕ, які мають багаторічний досвід з розробки супутникових технологій точного позиціонування і навігації: FP7 «EEGS – EGNOS Extension to Eastern Europe» – грант № 247698; «EEGS2 - EGNOS Extension to Eastern Europe Applications» – грант № 287179.

## 9.6 Система управління корпоративними перевезеннями

Система вже використовується для оптимального планування рейсів з доставки вантажів, що приводить до зменшення часових і матеріальних витрат за рахунок: 1) зниження витрат на паливно-мастильні матеріали (ПММ); 2) оптимального розподілу замовлень між автомобілями; 3)

прогнозування поставок товарів для зменшення складських витрат; 4) економії робочого часу персоналу або скорочення штатних співробітників; 5) зменшення числа автомобілів для виконання заданого обсягу перевезень; 6) моніторингу та оперативного управління автомобілями при доставці вантажів в реальному часі.

Ринкова привабливість хмарного сервісу транспортної логістики: оптові компанії, регіональні дистриб'ютори продовольчих і промислових товарів (хлібозаводи, молокозаводи, м'ясокомбінати, пиво-безалкогольні комбінати, промислові підприємства, автотранспортні підприємства, торгові мережі, логістичні оператори, транспортно-експедиторські компанії, вендингові компанії, швидка допомога, інкасаторські служби, кур'єрські служби, інтернет-магазини, клінінгові компанії). Більше 7500 підприємств тільки в Україні.

Технології логістики. Послуги з перевезення вантажів – це комплексна, багатокритеріальна задача, що включає велику кількість параметрів, від яких залежить ефективність виконання договору з замовником, а значить, і прибуток підприємства. Транспортна задача є NP-повною, де кількість варіантів знаходиться в експоненційній залежності від числа вхідних значень. Точне рішення може бути отримано методом повного перебору всіх можливих варіантів. Для реальних завдань бізнесу використовуються квазіоптимальні методи, які не дають точного рішення, а значить, і максимально можливої економії коштів. Пропонується оптимальний метод вирішення транспортної задачі на основі оригінального алгоритму, що істотно зменшує час, який стає прийнятним для аналізу більшості практичних ситуацій на картах регіонів.

Бізнес-моделі: 1) Продаж ліцензій на використання програмного продукту з подальшою оплатою сервісного супроводу. 2) Продаж сервісів відповідно до абонентської плати за час користування функціональностями хмари дорожнього руху.

## 9.7 Організація зв'язків «хмара – автомобіль», «хмара – інфраструктура»

Найважливішим аспектом технологічної (технічної) реалізації CPS є організація зв'язку між чотирма компонентами системи (рис. 9.1), інтегрованими з хмарою: Cloud Servers – сервери, що створюють хмару довготривалого зберігання розподілених даних і сервісів; Buffer Computers – буферні комп'ютери, що забезпечують збір даних від моніторів інфраструктури та доставку сервісів управління дорожнім контролерам; C-RFID – комп'ютерні блоки радіочастотної ідентифікації транспортних засобів; I-CMC – інфраструктурні контролери моніторингу та управління дорожнім рухом на основі радіочастотної ідентифікації транспортних засобів.

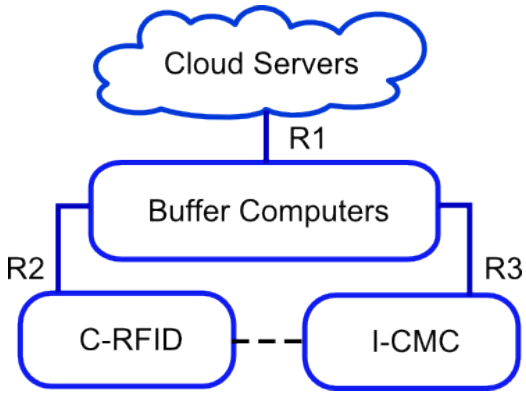



Рисунок 9.1 – Структура зв'язків між компонентами ІІДД

Структура комунікаційної інтеграції чотирьох компонентів ІІДД представлена транзакціями: (R1 * R2) = (SC, BC, C-RFID) – доставка хмарних сервісів до споживача; (R1 * R3) = (SC, BC, I-CMC) – доставка сигналів управління до контролерів дорожнього руху. Маршрут першого типу використовує традиційні технології GPRS, HSPA, Wi-Fi, WiMAX на основі мережі Internet. Для другого типу транзакцій, зважаючи на їх надзвичайну важливість, а також високі вимоги до надійності, завадостійкості та

захищеності, необхідні додаткові науково-технічні дослідження в процесі створення масштабованого прототипу.

Передбачається, що в блоці C-RFID будуть записані індивідуальний код транспортного засобу (CID), електронний код реєстрації за місцем проживання (NID), а також код водія (DID), що виконує експлуатацію даного автомобіля в поточний момент часу. Зчитувати тріаду кодів (CND-ID) повинні радіопристрої, якими будуть оснащені всі світлофорні об’єкти, мости, тунелі, залізничні переїзди та інші пункти вулично-дорожньої мережі, істотні з позиції управління дорожнім рухом, в тому числі, деякі критичні контрольні точки. Структура блоку C-RFID представлена на рис. 9.2, де модулі (CND-ID, CT, SP, ALB, M, D, CU) позначають: універсальний код автомобіля, приймач, модуль захисту, арифметико-логічний пристрій, модуль пам’яті, дисплей і модуль управління.

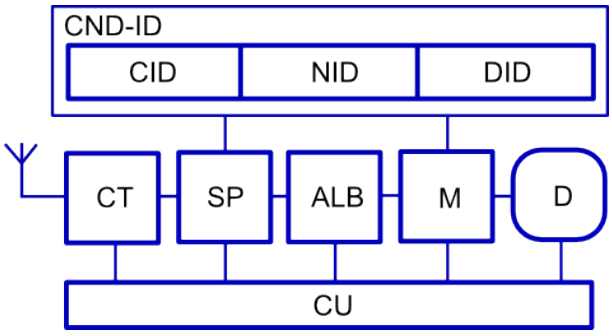



Рисунок 9.2 – Структура блоку C-RFID

Світовий досвід застосування засобів радіочастотної ідентифікації в сфері транспорту дозволяє робити оптимістичні прогнози щодо впровадження подібних технологій в Україні. Так, у 2012 році МВС Росії успішно провело випробування з постановки RFID-міток на номери автомобілів в рамках проекту "Розумне місто". В даному випадку чіп RFID був інтегрований в табличку з номерним знаком, вироблений пітерським ВАТ "Авангард". У Малайзії обов’язкова постановка RFID-чіпа на автомобільний номер введена з 2007 року. Дорожній поліцейський тут може перевірити будь-який автомобіль, навіть не зупиняючи його, як зі

стаціонарного поста, так і з мобільної патрульної машини. У США з початку 1990-х років діяла система 3M GM Automotive Adhesive, яку можна вважати прототипом сучасної технології радіочастотної ідентифікації.

Дослідження, проведені Московським технологічним університетом в 2001 році, показали, що технологія RFID дозволяє ідентифікувати як стаціонарні об'єкти, так і рухомі транспортні засоби з досить високою точністю і достовірністю, а також має високу надійність, довговічність і захищеність. Однак поряд з численними перевагами цієї технології мають місце також її недоліки. Перш за все, дальність дії RFID-міток залишає бажати кращого. Проте, опубліковані в Components & Technologies результати досліджень російських вчених претендують на дальність до 300 метрів. Відзначається також негативний вплив електронних чіпів на живу і неживу матерію. Так, в червні 2008 року Journal of the American Medical Association опублікував результати впливу RFID на медичну техніку. Електронні перешкоди від радіоміток скидають налаштування внутрішньовенних крапельниць, перепрограмують електронні кардіостимулятори і вносять збої в роботу медичної техніки. Більше третини проведених тестів дійсно виявили збої в роботі медичного обладнання, що знаходилося на відстані від сантиметра до шести метрів від джерела RFID. В іншій третині тестів були зафіксовані серйозні порушення в роботі апаратів штучної вентиляції легенів, інфузійних насосів, апарати для гемодіалізу, моніторах електрокардіограм (ЕКГ). Негативний вплив транспондерів на живі організми і людину час від часу мусується в ЗМІ та мережі Інтернет, що створює певні труднощі реальної перспективи електронної паспортизації населення.

У запропонованій системі RFID враховані зазначені вище фактори. Використовуються активні RFID мітки з двома каналами передачі даних – радіочастотним і оптичним. При застосуванні активної мітки дальність дії обмежена, перш за все, вихідною потужністю мітки при незмінному коефіцієнті спрямованої дії антени, чутливості приймального тракту.

Система RFID має можливість оперативного регулювання вихідної потужності передавача при обов'язковому обмеженні за максимумом рівня + 4дБм. Це виключає будь-який вплив на живу і неживу матерію, оскільки на кілька порядків менше норми допустимої SAR (Specific Absorption Rate) – за питомим коефіцієнтом поглинання електромагнітного випромінювання організмом людини. Вимірюється SAR у Ватах на кілограм (Вт/кг). Федеральною комісією зі зв'язку в США (FCC), Міністерством промисловості Канади (IC), а також регулюючими органами деяких інших країн прийнята норма SAR рівна 1,6Вт/кг. У країнах Європейського Союзу прийнята норма SAR рівна 2Вт/кг. Вихідна потужність пропонованої RFID не перевищує одиниць міліватт на відміну від мобільних телефонів з вихідною потужністю до двох ват. Крім того, RFID модуль встановлюється на значній відстані від водія і пасажирів, що повністю виключає негативний вплив високочастотного випромінювання.

Що стосується перешкод медичного обладнання, то, по-перше, на автодорогах таке обладнання відсутнє, а в разі реанемобіля швидкої допомоги медичне обладнання знаходиться всередині екрануючого кузова автомобіля. По-друге, перешкоди створює в основному потужне електромагнітне поле, створюване рідером для живлення транспондера (мітки), і в нашому випадку воно відсутнє, оскільки живлення транспондерів здійснюється не за рахунок поля рідера, а за рахунок бортової мережі автомобіля або резервного акумулятора транспондера. В крайньому випадку, радіоканал транспондера може бути відключений і залишиться тільки оптичний канал.

Доповненням технології RFID може стати GPS навігація. Сучасні GPS приймачі на базі чіпсета SiRF Star III, фіксують сигнал навіть в ангарах і цехах із залізобетонним перекриттям. Приймачі останнього покоління підтримують як сучасну Європейську систему глобального позиціонування Galileo, так і російську систему ГЛОНАСС, що розвивається. Недоліком GPS навігації є неможливість передачі даних про позицію ТЗ на супутник. Таким

чином, при розробці підсистеми позиціонування ІІДД можна одночасно реалізувати обидві зазначені технології з метою їх детального дослідження для застосування до важливості справ моніторингу та управління транспортними потоками.

## 9.8 Структура автомобільного блоку CAR-ID

В основу пропонованої концепції CAR-ID покладені принципи, які використовуються в системі управління повітряним рухом ADS-B. Суть CAR-ID полягає в тому, що транспондер транспортного засобу періодично передає широкомовне повідомлення, яке включає ідентифікаційну інформацію і дані про координати і швидкість транспортного засобу, одержуваних від вбудованого GPS приймача. Крім того, контролер CAR-ID веде протокол динаміки руху транспортного засобу, отримуючи інформацію від датчика прискорень.

Передача повідомлення здійснюється за двома каналами – бездротовому і/або оптичному. Повідомлення приймаються транспортними засобами або стаціонарними станціями, які перебувають в зоні оптичної або радіо- відимості. Стаціонарні станції об’єднані в мережу і розташовані в місцях, де є електроживлення (світлофори). Після отримання повідомлення контролер CAR-ID перевіряє його наявність в «історії» і в разі відсутності додає його в пам’ять контролера. При попаданні в зону дії стаціонарного монітора (станції) відбувається скидання всієї інформації, накопиченої з моменту попереднього зчитування, з пам’яті контролера в пам’ять станції. Вона формує інформаційні пакети і періодично передає їх до хмари.

Для забезпечення високої перешкодозахищеності, структурної скритності сигналу і виключення впливу перешкод іншим радіотехнічним засобам блок CAR-ID використовує технологію прямого розширення спектра DSSS. Модуль може працювати в неліцензованому ISM діапазоні з вихідною потужністю 0 – 4дБм. Цього достатньо для забезпечення зони радіо-відимості до 100 метрів при застосуванні ненапрямлених антен.

Вся інформація, передана по відкритих каналах, попередньо кодується. Як антиколізійний метод в блоці застосовується метод Slotted – ALOHA. При необхідності вся інформація, накопичена контролером за добу, може бути зчитана поліцейськими або іншими фіскальними службами при наявності спеціального рідера. Таким чином, створюється розподілена інтелектуальна бездротова мережа на основі використання блоку RFID (рис. 9.3), перевагою якої є наявність пристроїв розподіленого зберігання і оперативного обміну інформацією.

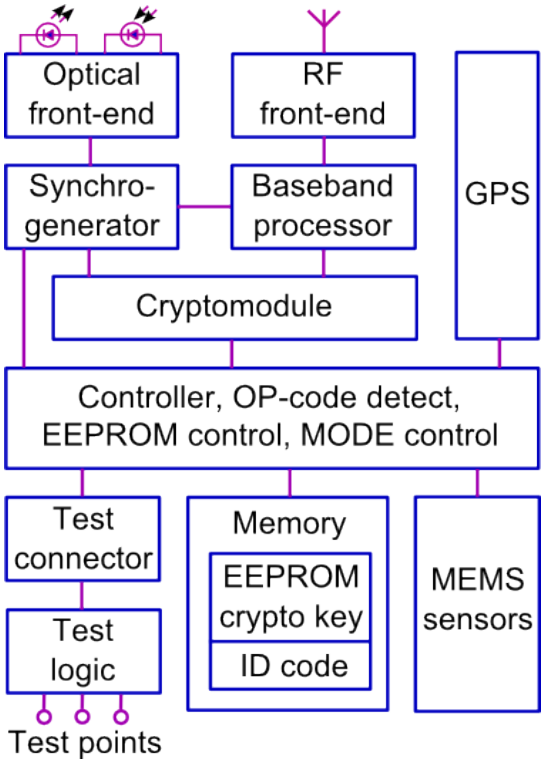



Рисунок 9.3 – Детальна структура блоку C-RFID

Структура блоку CAR-ID містить наступні модулі: Optical front-end – оптичний інтерфейс; RF front-end – радіочастотний інтерфейс; Synchrogenerator – генератор частот; Baseband processor – обробки сигналів після демодулірованія; GPS – модуль позиціонування; Cryptomodule – модуль криптографічного захисту; Controller, OP-code detect, EEPROM control, Mode control – система управління блоком; Test connector – перемикач тестування модулів; Test logic (Test points) – модуль управління тестуванням і програмуванням; Memory (EEPROM crypto key, ID code) – модуль пам’яті

для зберігання даних і службової інформації; MEMS sensors – модуль сенсорних датчиків.

## 9.9 Засоби моніторингу та управління дорожнім рухом

Сучасні великі міста мають складну дорожню інфраструктуру, де управління дорожнім рухом здійснюється за допомогою світлофорного регулювання із застосуванням систем управління дорожнім рухом (СУДР), що включають в себе сотні світлофорних об’єктів. Тут під світлофорним об’єктом будемо розуміти підсистему СУДР, що забезпечує моніторинг і управління транспортними потоками на окремій ділянці вулично-дорожньої мережі. Центральну частину цієї підсистеми (див. рис. 9.4) складає дорожній контролер, як правило, спеціалізований з вбудованими комутованими силовими ланцюгами для управління світлофорами. Слід зауважити, що контролери SITRAFFIC C800 спроможні опитувати до 84 детекторів транспорту індуктивного типу і управляти 48 групами сигналів сумарною потужністю 4 кВт в режимі реального часу з максимальним допустимим циклом в 300 секунд. Контролер C800VX підтримує використання до 120 таких модулів в одному сегменті управління, кожен сегмент здатний функціонувати автономно, інтегрується в мережу СУДР на основі бездротових технологій (GPRS, WiMAX) та централізовано управляється з центру управління рухом.

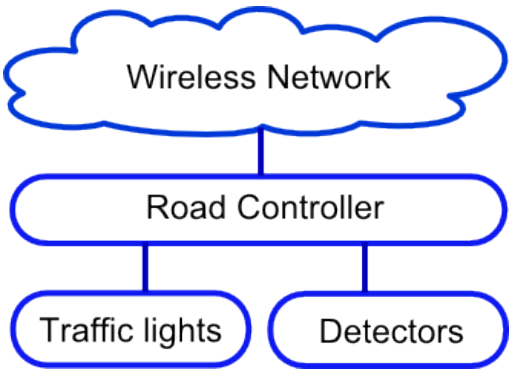



Рисунок 9.4 – Загальна структура світлофорного об'єкта

При збереженні існуючих тенденцій розростання дорожньої інфраструктури стає ясно, що використання подібних рішень ставить під загрозу надійність експлуатації таких систем. Відомо, що з ростом структури СУДР (фактично це виражається в збільшенні числа світлофорних об’єктів, контрольованих системою) її надійність падає. Тому актуальною науковою і виробничою задачею є розробка більш надійних структур СУДР, серед яких особливе місце займає структура розподіленої автоматизації. Встановлено, що для підвищення надійності системи поряд із забезпеченням інформаційних і керуючих функцій хмари доцільно організувати СУДР по централізовано-децентралізованим принципом. В такому випадку буферний комп'ютер ІІДР на рис. 9.1 виконує функції сервера даних і забезпечує зв'язок з периферійними робочими станціями, які керують кількома контролерами, вся сукупність яких сегментируется (10 - 20 світлофорних об'єктів на сегмент) за географічним принципом. Така архітектура суддю дозволяє розташувати сервери в будь-якій точці міста і організувати мобільний центр управління, що забезпечує збереження координованої роботи всіх дорожніх контролерів на об'єкті при виході з ладу центральної частини системи. Структура компонента I-CMC (див. рис. 9.1) представлена у вигляді матриці (див. рис. 9.5),

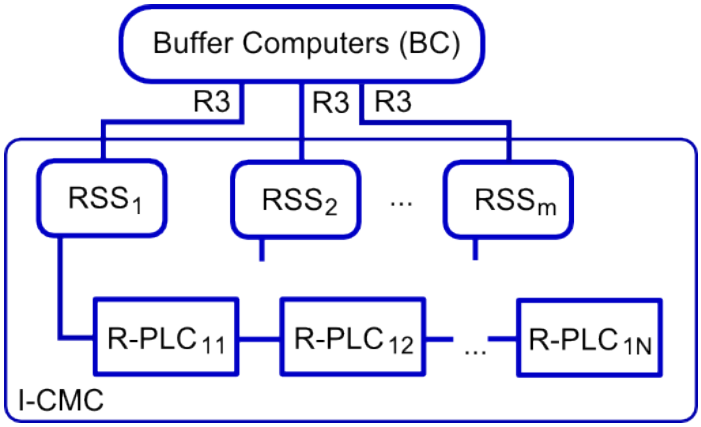



Рисунок 9.5 – Структура блоку I-CMC з рис. 9.1

Тут модуль RSS являє надійний комп'ютер промислового виконання, а компонент R-PLC побудований на базі компактного і досить потужного програмованого логічного контролера SIMATIC S7-1200 фірми SIEMENS для програмування технологічних процесів для вирішення завдань автоматичного регулювання та керування рухом. Характеристика PLC S7-1200: 1) висока надійність, середній час напрацювання на відмову – більше 30 років; 2) можливість перепрограмування під час роботи контролера; 3) сервісне обслуговування за місцем розташування об'єкта; 4) продуктивність – 105 команд в секунду з періодом 15 нс; мова програмування STEP-7 Basic з вбудованими засобами нечіткої логіки.

### 9.10 Дистанційний модуль "SHERLOCK"

Призначений для побудови розподілених систем контролю і управління об’єктами, включаючи мобільні. Являє собою електронний виріб, побудований на основі використання трьох новітніх технологій Mobile-to-Mobile, GPS і GPRS.

Задачі, які вирішуються за допомогою модуля: 1) Автоматичне визначення місцезнаходження транспортних засобів (AVL). 2) Управління автотранспортним парком, логістика. 3) Автоматизація служб таксі. 4) Моніторинг маршруту і розкладу руху транспорту. 5) Моніторинг режимів експлуатації транспортних засобів. Технічні характеристики: GPS – багатоканальний приймач з високою чутливістю і малим енергоспоживанням, спеціально призначений для роботи в умовах міської забудови та наявності відбитих сигналів. GSM – трьохдіапазонний GSM / GPRS модуль, працює у всіх існуючих на Україні GSM мережах. Входи – 8 цифрових / 1 аналоговий. Виходи – 7 цифрових (відкритий колектор). Інтерфейси – CAN 2.0 шина для підключення до бортової мережі транспортного засобу, управління виконавчими пристроями і опитування додаткових датчиків. Пам'ять – 512 кБ вбудованої пам'яті для зберігання телеметричної інформації. Вбудований датчик температури. Вбудований

апаратний самоконтроль. Автомобільний діапазон робочих температур і живлячих напруг.

Дистанційний модуль "SHERLOCK" конструктивно виконаний в компактному пластиковому корпусі, має один 24-контактний системний роз’єм для підключення до джерела живлення, виконавчих пристроїв і датчиків. Два високочастотних SMA роз’єми призначені для підключення GPS і GSM антен. Дистанційний контролер для роботи в режимі GPRS вимагає конфігурації імені точки доступу (APN, Access Point Name), імені або IP-адреси сервера, і номера порту.

Функціонування модуля. Спроби виходу на зв’язок за допомогою GPRS робляться кожні 10 хв. Дані про зміну координат, прийнятих GPS приймачем, передаються на сервер з інтервалом від 10 до 90 секунд в залежності від швидкості руху об’єкта, на якому встановлено пристрій. Віддалене управління пристроєм здійснюється за допомогою SMS-команд: 1) Запит стану пристрою. 2) Конфігурація режимів роботи по GSM / GPRS. 3) Управління виходами. 4) Запит на виконання USSD команд. 5) Онлайн сервіс моніторингу.

Доступ до онлайн сервісу моніторингу здійснюється цілодобово із сторінки http://gps.rfid.com.ua. Для отримання доступу до сервісу користувач повинен авторизуватися за допомогою логіна і пароля. На головній сторінці сервісу більшу частину площі екрану займає вікно з картою, на яку накладаються дані про місцезнаходження мобільних об'єктів. Для візуалізації використовується картографічна інформація компанії ВІЗІКОМ. Вгорі головної сторінки передбачені елементи управління, за допомогою яких можливий вибір одного, двох, або всіх належних користувачу об'єктів, і часового діапазону, для якого необхідно переглянути інформацію про переміщення. При візуалізації на карті піктограмами відображається стан об’єктів і маршрут руху за обраний інтервал часу, а також тривалі стоянки. Розмір і положення карти можна змінювати за допомогою миші і елементів управління. У нижній частині головної сторінки розташовуються елементи

управління, що дозволяють швидко перемикатися між частинами маршруту і між об’єктами, а також статистична інформація. При виборі відображення тільки одного об’єкта доступна функція розрахунку відстані.

Налаштування. На сторінці налаштувань користувач може ввести інформацію про свою електронну адресу, змінити пароль, розмір карти і переглянути зведену інформацію про налаштування об’єктів, розсилок. Об’єкти: з даної сторінки можлива зміна назви, опису об’єктів, а також їх параметрів. Правила розсилки: пункт меню, призначений для завдання правил розсилки повідомлень про переміщення об’єктів між окремими географічними областями, введеними користувачами в систему. Дозволяє в інтерактивному режимі додавати правила розсилки у відповідь на такі події, як входження об’єкта в область, вихід з неї, а також перехід з однієї області в іншу. Області: в даному пункті меню з’являється запрошення визначити замкнуті області на карті, які можуть бути використані при конфігурації розсилок. Координати: на даній сторінці буде відображено короткий звіт статистична інформація про місцезнаходження мобільних об’єктів в поточний момент часу, а також інформація про найближчий географічний об’єкт, відомому системі. У базі даних системи знаходиться інформація про координати декількох десятків тисяч адрес по місту Києву. Зв’язок: даний пункт меню призначений для отримання статистичної інформації про роботу системи. У табличному вигляді відображаються останні координати мобільних об’єктів, час, що минув з моменту їх отримання, інша телеметрична інформація, а також інформація від геокодера про область, в якій знаходиться кожен з об’єктів. Комплект поставки телеметричного модуля: антена GPS; антена GSM; кабель з’єднувальний; інструкція з експлуатації; SIM-карта, а також інформація про найближчому географічному об’єкті, відомому системі. У базі даних системи знаходиться інформація про координати декількох десятків тисяч адрес по місту Києву.

Реальний світ, потребуючи здійснених процесів хмарного управління, давно усвідомив необхідність абсолютно точної радіочастотної цифрової

ідентифікації всієї виробленої продукції і природних об’єктів на планеті, включаючи людину і тварин. Наступний крок – створення хмарних цифрових моделей об’єктів реального світу для точного симулювання, моніторингу та управління будь-якими відносинами: природними, соціальними, технічними, технологічними.

MAT (Memory – Address – Transaction) Computing. Комп’ютер (квантовий) майбутнього є пам’ять і адресні транзакції. Пам’ять організовується на будь-якій формі існування матерії. У пам’яті реалізуються механізми управління і виконання. Адресні транзакції створюють всі процеси і алгоритми. Доставити одну інструкцію з control unit в пам'ять ефективніше, ніж передати величезні обсяги інформації в АЛУ і назад. Очевидно пляшкове горлечко існує 70 років. Дані (big data) слід обробляти там, де вони існують. Це – найближче майбутнє кібер-фізичного комп’ютингу.

Кібер-соціальний комп’ютинг. Повна аналогія. Мегаполіси засмоктують мільйони людей вранці через пляшкове горлечко трафіку жахливих доріг для їх екзекуції в офісах, а ввечері випльовують їх назад по домівках, в близькі й далекі села. Гігантські матеріальні, часові та фінансові витрати на бензин, оренду офісів і переміщення в реальному просторі, стреси в трафіку, забруднення атмосфери. Рішення проблеми – кібер фізичний моніторинг і управління. Доставляти інструктивні впливи через гаджети співробітникам в їх особисті віртуальні кабінети, завдяки узаконеній online організації і створенню максимального комфорту для роботи співробітників, інваріантних до географічній точці позиціонування людини (будинок, готель, подорожі, відпочинок). Умови оплати праці – своєчасне і якісне виконання завдання. Освіта і наука повинні виконуватися в режимі online. On-site зустрічі, лекції та семінари розглядаються як розкіш живого спілкування. Сьогодні існують всі компоненти е-інфраструктури для реалізації кібер соціального комп'ютингу: хмарний сервіс управління, edge gadgets користувачів і розумні речі для реалізації интеракций. Єдине, чого немає –

законодавства, що легалізує мульти-мільярдну над-інновацію в масштабах країни і планети.

### 9.11 Сьогодення і цифрове майбутнє планети

Аддитивна наноелектроніка, IoT-комп'ютинг і кіберпростір складають сьогодні три еволюціонують рівня ієрархії цифровий планети. На лані даної технологічної культури в даний час працює третина всіх інтелектуальних ресурсів людства, створюючи цифрову карту дзеркального і хмарного відображення фізичних процесів і явищ реального світу з метою підвищення якості життя людства і створення «зеленої» кіберекосістеми. Це означає оцифровування не тільки всіх мобільних і стаціонарних об'єктів, а й «чистого» структурованого земного простору для точного завдання координат процесів і явищ, що відбуваються на реальній планеті.

Прискорення розвитку знань про природу останнім часом пов'язується з технологічною сингулярностью або вибухом в розумінні законів мікро- і макросвіту на короткому проміжку часу, що неодмінно призведе до створення через два десятиліття глобального кібермозку для управління людством, а також процесами і явищами реального світу. Це стає можливим завдяки розвитку трьох компонентів: біоінженерії, штучного інтелекту кіберпростору і нанотехнологій адитивного виробництва промислових виробів. Тут мається на увазі: 1) Вбудований інтерфейс безпосереднього зв'язку мозку людини з комп'ютером і/або кіберпростором шляхом усунення послідовних мовних інтерфейсів між ними; 2) Створення штучного інтелекту для самонавчання і самовдосконалення небіологічних (комп'ютерних) структур, програм і процесів; 3) Нано-вирощування комп'ютера шляхом адитивного структурування атомів. 4) Найцікавіше рішення пов'язане з невідворотністю відмови людства від функцій управління біологічними, соціальними і технічними об'єктами і процесами на користь кіберсистем.

Ринкова привабливість «зелених» нанотехнологій висхідного проектування – побудова або вирощування (квантового) комп'ютера шляхом

структурування атомів – полягає в безвідходності, мікроміниатюрності, наднизькому енергоспоживанні, абсолютно мінімальній витратності матеріалів, а в майбутньому і вартості, надвисокій швидкодії і необхідній масштабованості, сумірною з класом сервісів, що доставляються. Сучасні технології дозволяють сьогодні не тільки сканувати атомні структури, а й послідовно будувати або вирощувати їх методом 3D-друку. Однак на шляху вирішення проблеми нано-технологічного напрямку розробки квантового комп'ютера на ринку є три привабливих, але ще не вирішених завдання: 1) Відкриття технологій з високою швидкодією вирощування необхідних гетерогенних атомних структур відповідно до заданої програмної специфікації обчислювача; 2) Створення ефективного транзакційного механізму для реалізації простого моніторингу та управління квантовими станами вирощених атомних обчислювальних структур з адресованими компонентами; 3) Забезпечення необхідної в часі стабільності станів компонентів атомної структури, що реалізує пам'ять.

Стадії еволюції кіберпростору планети ілюструють послідовні періоди переходу науково-технологічної моди від моніторингу (відображення) фізичних, біологічних і соціальних процесів до їх управління на основі взаємодії реальних і віртуальних структур: 1) 1980-і роки – формування парку персональних комп'ютерів; 2) 1990-ті роки – впровадження Інтернет-технологій у виробничі процеси і побут людини; 3) 2000-і роки – підвищення якості життя за рахунок повсюдного використання мобільних пристроїв і хмарних сервісів; 4) 2010-і роки – створення цифрової інфраструктури моніторингу, управління і взаємодії між собою стаціонарних і рухомих об'єктів, включаючи повітряний, морський, наземний транспорт і роботів; 5) 2015-і роки – створення глобальної цифрової інфраструктури кіберпростору. При цьому досить ясно простежується процес інтелектуалізації та інтеграції фізичних і віртуальних систем: Embedded Systems – Networked Embedded Systems – Cyber Physical Systems – Internet of Things, Data and Services для вирішення наукових, технологічних, економічних, політичних і соціальних

проблем. Конкретно, під структурну дикцію кіберфізичних систем підпадають такі галузі: автомобільна промисловість, медицина, енергетика, автоматизація виробництва, мобільний зв'язок, локація і навігація, сільське господарство, транспортна логістика, створення розумних міст, будівель і будинків, соціальні мережі та спільноти, організація дорожнього руху, управління фізичними і віртуальними процесами.

## 9.12 Ринкова привабливість і техніко-економічна характеристика проекту TCS

Кожні чотири роки локально змінюється технологія виробництва товарів і сервісів в певному ринковому сегменті, яка вимагає перенавчання кадрів, зміни інфраструктури, системи управління і відносин, але головне – напрямки руху на ринку товарів і науково-освітніх послуг. Кожні 20 років в передових галузях глобально змінюються технологічні уклади типових виробництв, які вимагають мільярдних капіталовкладень, непідйомних навіть для лідерів в ринкових сегментах, які формують основні капітал-індекси на Уолл-стріт. Це означає, що новий технологічний уклад формується тільки на основі взаємної кооперації тих компаній і країн, які мають вільні капітали, підготовлені кадри і ідеї, точно потрапляють в "десятку" мішені ринкової моди наступного 20-річчя. Саме сьогодні компанії та університети України можуть зробити технологічний ривок на ринку товарів і послуг, включаючи саму модну тріаду (біо- і нано-технології, штучний інтелект), яка формує найважливіший Cyber-Physical-Space-сегмент Nasdaq-ринку: Internet of Things, Smart Everything and Cyber-Physical Systems.

Україна повинна приєднатися кадровим потенціалом до науково-технологічних перегонів євро-американо-азіатських компаній, щоб зайняти в майбутньому призове місце на п'єдесталі економічного зростання Європи. Потенціал країни фантастично великий (кадри, центрова територія, науково-освітня культура). Для його використання в модному ринковому напрямку Cyber-Physical-Space повинна бути вироблена політична воля – керуючий

вплив, що створює всебічні умови для формування науково-технологічної культури, як основи європейського майбутнього України. Найцікавіші глобальні проекти сьогодні виконуються під егідою об’єднання двох просторів в єдине ціле. Кібер-фізичний простір – метрика взаємодії (взаємної інтеграції) фізичних, біологічних і соціальних об'єктів, процесів і явищ з віртуальними або хмарними технологіями управління - покликане інтегрувати найбільш перспективні кібер-фізичні технологічні рішення. Найцікавіше з них пов’язано з невідворотністю відмови людства від функцій управління біологічними, соціальними і технічними об’єктами і процесами на користь киберсистем.

Джерела фінансування проекту: 1) Державні структури (міністерства: освіти і науки, транспорту, внутрішніх справ); 2) Приватні інвестори України та зарубіжжя; 3) Європейські та американські міждержавні програми; 4) Start-up пропозиції для інверстіційних програм провідних компаній планети (Google, IBM, Microsoft, Apple); 5) Венчурні і автомобільні компанії розвинених країн; 6) Власні кошти розробників і учасників проекту.

Джерела отримання прибутку: 1) Орендна плата за хмарний сервіс управління трафіком з боку водіїв; 2) Плата за розміщення реклами; 3) Орендна плата за корпоративний хмарний сервіс моніторингу і управління трафіком транспорту підприємства; 4) Відрахування державних служб за спеціальний сервіс моніторингу та управління транспортними засобами; 5) Відрахування муніципальних державних служб за сервіс моніторингу транспортними засобами та управління вуличними (віртуальними) світлофорами міської інфраструктури; 6) Продаж акцій, завдяки капіталізації бізнесу та відкритого акціонування активів компанії-розробника хмарного сервісу; 7) Продаж Start-up TCS-проекту однієї з компаній Google, Apple або Samsung за 150 мільйонів доларів.

**Техніко-економічне обґрунтування проекту** в масштабах України і світу: 1) Країна має сьогодні 120 тисяч фірм, серед яких 70 тисяч – суб’єкти ІТ-індустрії, понад 7 тисяч – транспортні компанії і державні структури,

зацікавлені в сервісах моніторингу та управління корпоративних транспортних засобів. Крім того, 9 мільйонів автомобілів по країні потенційно потребують хмарних сервісах, що забезпечують моніторинг і управління дорожньої обстановки по замовленому маршруту руху. Далі, обслуговування та експлуатаційні витрати одного світлофора обходяться казні приблизно в 50 доларів на рік, а його установка коштує від 1000 до 10000 доларів. Кількість таких пристроїв по Україні дорівнює 80 000 штук. Їх разова установка і експлуатаційні витрати (за 10 років служби) складають в середньому: (5500 + 500) х 80000 = $ 480 000 000 бюджетних грошей. Півмільярда доларів на світлофори можна перетворити для бюджету країни практично в нульові витрати на віртуальне розміщення та експлуатацію в хмарному сервісі пристроїв управління трафіком. У масштабах всієї планети можна звільнити для потреб людства близько 500 мільярдів доларів. До того ж весь арсенал земних світлофорів "пожирає" мільйони кіловат-годин електроенергії, що робить планету зеленішою. Сюди ж можна додати сотні тисяч тонн металу, сотні тисяч кілометрів кабелів і проводів. Без усього перерахованого можна обійтися, простим перенесенням світлофорів в кіберпростір управління трафіком, накладених на карту інфраструктури земних доріг. Економічний ефект від перенесення в масштабі України – 500 мільйонів доларів тільки за рахунок ліквідації витрат на утримання і установку пристроїв управління перехрестями. 2) Друга частина ринково орієнтованої інновації проекту полягає у 15% економії палива за рахунок пропозиції з боку хмарного сервісу квазіоптимальних маршрутів в реальному часі, а також в 15% зменшенні часу виконання поїздки, що істотно вплине на собівартість перевезення вантажів та пасажирів. Економія від використання хмарного сервісу для одного водія на рік становить: 15000 гривень – середня вартість річного споживання бензину х 0,15 = 2250 гривень або 200 доларів. В цілому по країні це дасть можливість зберегти в гаманці власників автомобілів кругленьку суму, порядку: 200х9000000 = 1 800 000 000 доларів. Приблизно така ж сума виходить за рахунок 15% економії часу проходження

замовленого маршруту, що по країні створює інтегральну економію коштів трудящих у розмірі 3,6 мільярда доларів на рік. 3) Витрати на створення промислового макета масштабованої хмарної кібер-фізичної системи управління трафіком міського фрагмента (вулиця Клочківська, місто Харків) включають покупку і інсталяцію програмно-апаратного забезпечення (М1=$200000), розробку і верифікацію нових програмно-апаратних компонентів хмарного управління, інтегрованих зі згаданими вище промисловими покупними виробами (М2=$500000), оснащення 100 автомобілів, що беруть участь в експериментах, моніторами для хмарного управління транспортом (М3 = $ 100000). Оснащення перехресть міського фрагмента інфраструктури мікроконтролерами з приймально-передавачами для створення дублюючого вулицю хмарного макета, синхронно працює з реальними світлофорами (М4 = $ 100000). 4) Сумарні витрати для проведення реальних експериментів на фрагменті міської інфраструктури складають: М = М1 + М2 + М3 + М4 = $ 900000. Накладні витрати – $ 100000. У масштабі України інтегральні витрати не будуть носити адитивного характеру в процесі масштабування, оскільки система управління створюється інваріантною до розмірності інфраструктурної ділянки доріг, але яка потребує створення потужного дата центру, вартістю $ 10 000 000. Крім того, були здійснені витрати, пов’язані з нанесенням і актуалізацією всіх дорожніх світлофорів (дорожніх знаків) на існуючі карти, близько $ 1000000. Таким чином, інтегральні витрати на створення масштабованого програмно-технічного комплексу хмари кібер-фізичної системи управління, інтегрованого з фрагментом міської інфраструктури та з автомобільною експериментальною групою складають: 900000 + 10000000 + 1000000 + 100000 = 12 000 000 доларів. 5) Доходи компанії-резидента, в доларах, від впровадження хмарної системи управління трафіком будуть визначатися наступними складовими: надходженням орендної оплати за хмарний сервіс від транспотних компаній 7000х1000=7000000. Оплатою трафік-сервісів, що надходить від водіїв особистих автомобілів 100 х 9 000 000 = 900 000 000. З

огляду на те, що кіберсистема хмарного управління транспортом стане за значимістю рівною Google-сервісам, то рівень капіталізації такого Інтернет-шару очікується близько 10-50 мільярдів доларів. При цьому очікувані надходження від реклами складуть не менше 5 відсотків (500 мільйонів доларів). Через 5-10 років автомобільний кібер-шар Інтернету об'єднає в інтелектуальну мережу один мільярд автомобілістів разом з автомобілями і 4 мільярди жителів планети. Рівень капіталізації всіх компаній, які формують кібер-шар управління транспортом, може інтегрально досягти 300-500 мільярдів доларів. Автомобіль з монітором, комп'ютером та Інтернетом стане робочим місцем і зоною відпочинку, де водій буде проводити до 20% свого життя, вирішуючи виробничі і соціальні проблеми спілкування в процесі пересування. Економічно від впровадження сервісу управління виграють всі: водії – від економії палива і колійного часу, компанії – від зниження собівартості доставки вантажів і пасажирів, держава і природа – від зниження злочинності, транспортних правопорушень, рівня вихлопних газів і підвищення безпеки руху транспорту.

Стадії інновації на стороні сервер-хмари і гаджет-користувача: 1) Створення квантових структур даних для паралельного зберігання даних і матричного аналізу Big Data форматів кіберпростору. 2) Розробка квантоподобних віртуальних паралельних процесорів для швидкого пошуку, розпізнавання та прийняття квазіоптимальних рішень з управління транспортом та інфраструктурою міст. 3) Створення комплексу програм для реалізації хмарних сервісів моніторингу та управління транспортом і дорожнім рухом в реальному часі на основі використання віртуальних хмарних світлофорів. 4) Розробка і практична реалізація моделі експериментальної ділянки міста для управління дорожнім рухом за допомогою хмарних світлофорів, моніторингу та управління транспортними засобами з анімаційної візуалізацією всіх масштабованих процесів на екрані монітора.

Власна оцінка науково-технічного рівня проекту: 1) Не має аналогів в

Україні – в частині створення хмарного сервісу моніторингу та управління дорожнім рухом, інфраструктурою вуличних світлофорів і надання сервісів прокладки оптимальних маршрутів в реальному масштабі часу. 2) На рівні існуючих в світі аналогів – в частині інтелектуального управління вуличними світлофорами, що враховує дорожню ситуацію на кожному перехресті в режимі реального часу. 3) Не має аналогів в світі – в частині реалізації керування транспортними потоками на основі використання розумних (віртуальних) хмарних світлофорів, виведених на екрани моніторів автомобілів і керованих реальними транспортними потоками.

Створення математичного кібер-шару проекту в формі моделей і методів, Big Data структур даних, нових віртуальних хмарних (Memory-Transaction) МТ-обчислювачів і засобів анімаційного моделювання кіберпростору управління транспортом. Розробка і реалізація алгоритмів квазіоптимального управління світлофором, оптимізації маршрутів руху з використанням історії. Створення алгоритмів ергономічної анімації з метою візуалізації процесів моніторингу та управління дорожніх ситуацій, пов’язаних з перехрестями на стороні хмари (сервера). Розробка алгоритмів ергономічної візуалізації керуючої інфраструктури на існуючих картах для виведення графічних даних на монітори автомобілів. Програмна імплементація алгоритмів в хмарні сервіси транспортного шару кіберпростору. Розробка технологічного кібер-шару проекту в формі взаємодії програмно-апаратних компонентів кібер-фізичної системи, що інтегрує існуючі і нові розробки: картографію, світлофори, автомобільні комп’ютери в єдину автоматну модель управління транспортом в реальному часі. Проектування інформаційного, методичного та лінгвістичного забезпечень кіберсистеми. Тестування і верифікація віртуальної частини кіберсистеми. Створення технічного і організаційного кібер-шарів проекту для подальшого виконання заходів по інтегруванню реальної керуючої інфраструктури доріг та автомобілів з віртуальним кіберпростору. Розробка глибокої мультиверсної системи захисту всіх програмно-технічних

компонентів хмарного сервісу управління транспортом і подальша верифікація промислового зразка системи хмарного управління трафіком на фрагменті міської інфраструктури з використанням спеціально оснащених автомобілів і розумних хмарних світлофорів, синхронізованих з реальними. Створення дата актуалізації картографічного шару шляхом віртуального нанесення на електронну карту інфраструктури світлофорів і їх синхронізація з реальними пристроями управління перехрестями на перехідний період. Створення Інтернет-системи банківської підтримки для акумулювання постійних, періодичних орендних і разових платежів. Проектування документації супроводу проекту, рекламна кампанія за пріоритетами кіберсистеми хмарного управління дорожнім рухом з використанням прес-релізів, телебачення, Інтернету, міжнародних конференцій.

### 9.13 Завдання, що підлягають вирішенню для реалізації TCS-хмари

1) Впровадження технології радіочастотного цифрового позиціонування транспортного засобу (гаджета) з точністю до 2 – 0,2 метрів.

2) Створення нової системи мнемонічних і супроводжуючих звукових сигналів на моніторі, що становлять хмарні правила дорожнього руху.

3) Розробка операційного та керуючого автоматів, які об’єднують всі компоненти TCS в єдину хмарну систему, що має входом і виходом гаджету G(k) (комп’ютер автомобіля), як єдиний інтерфейс зв’язку з хмарою, що позиціонується в реальному масштабі часу точними координатами k на карті M інфраструктури зі світлофорами L для управління пересуванням учасника дорожнього руху.

4) Проектування масштабованої системи TCS для світлофора, проспекту, району, міста, країни, планети.

5) Розробка сервісів (програмних додатків) на стороні клієнта (автомобіль, пішохід, велосипедист, мотоцикліст) для замовлення опцій квазіоптимального управління виконанням маршрутів пересування, включа-

ючи систему відео- і аудіо-сигналів, прив’язаних до інфраструктури і доставляються з TCS в реальному масштабі часу.

6) Проектування сервісів на стороні хмари (сервера) для вирішення завдань оптимізації при прокладці маршрутів, управлінні світлофором, модифікації хмарної і реальної інфраструктури, а також сервісів оперативного управління хмарними світлофорами на основі моніторингу дорожнього руху.

7) Розробка кубітних структур даних і «квантових» матричних процесорів (обчислювачів) на основі технологій Big Data для одночасного і паралельного сервісного обслуговування користувачів TCS-хмари в реальному масштабі часу, число яких в межі має бути рівною кількості жителів планети.

8) Створення інтелектуальних моделей, методів синтезу та аналізу віртуальної інфраструктури для оцінювання якості дорожнього руху, моделювання та візуалізації трафіку, пропозиції оптимального маршруту з урахуванням технічних, кліматичних, соціальних факторів, якості доріг, кількості світлофорів, лівих поворотів з метою створення нових і реконструкції існуючих інфраструктур дорожнього руху.

9) Розробка хмарних сервісів для автотранспортних підприємств з метою підвищення якості обслуговування пасажирів, перевезення вантажів, оптимізації часових і матеріальних витрат.

10) Створення хмарних сервісів для водія з метою підвищення якості проїзду за заданим маршрутом і оптимізації часових і матеріальних витрат.

11) Збір статистичної інформації (інтелектуалізація глобальної, корпоративної та персональної інфраструктури) шляхом накопичення історії трафіку, зміни його параметрів у часі і в просторі для прокладання квазіоптимальних маршрутів майбутніх поїздок.

12) Створення засобів захисту інформації та санкціонованого доступу до персональних і корпоративних даних в хмарі. Кожен користувач бачить на карті тільки свій автомобіль в хмарі і знеособлені транспортні потоки. Всі ідентифікатори транспорту можуть бути доступні за рішенням суду або постановою слідчих органів тільки спеціальним державним службам.

13) Створення специфікації TCS-хмари як start-up проекту для його подальшої пропозиції або продажу компаніям Apple і Google.

14) Пошук валідних партнерів і заможних інвесторів в розвинених країнах (Росія, США, Німеччина, Естонія, Іран, Саудівська Аравія, Польща, Франція, Швеція, Норвегія) для реалізації спільного проекту Smart Cloud Traffic Control шляхом отримання грантів в рамках європейських та національних програм на виконання масштабованого прототипу системи в межах міста, країни.

15) Патентування в Україні, США та інших країнах віртуальних: технології та інфраструктури управління дорожнім рухом на основі використання в реальному часі хмарних світлофорів, що відображаються на дисплеях гаджетів, одночасно використовуваних як сенсори і актюатори.

# 10 ІНШІ РЕЗУЛЬТАТИ, ОТРИМАНІ ТА РЕАЛІЗОВАНІ В МЕЖАХ НДР

## 10.1 Результативність виконання НДР

Таблиця 10.1 – Кількісні показники виконання НДР

| № з/п | Критерії | Заплановано (відповідно до запиту) | Виконано (за результатами НДР) | % виконання |
|---|---|---|---|---|
| | | кількість | кількість | % |
| **1.** | **Публікації колективу виконавців НДР:** | | | |
| | 1.1. Статті у журналах, що входять до наукометричних баз даних (*). | 12 | 29 (3 – Scopus) | 240% перевиконання |
| | 1.2. Публікації в матеріалах конференцій, що входять до наукометричних баз даних (*). | 15 | 30(Scopus) | 200% перевиконання |
| | 1.3. Статті у журналах, що включені до переліку наукових фахових видань України. | 12 | 24 | 200% перевиконання |
| | 1.4. Публікації у матеріалах конференцій, тезах доповідей та виданнях, що не включені до переліку наукових фахових видань України. | 26 | 110 | 423% перевиконання |
| | 1.5. Монографії, опубліковані за рішенням Вченої ради ВНЗ (наукової установи). | 1 | 4 (2 – Scopus) | 400% перевиконання |
| | 1.6. Підручники, навчальні посібники з грифом МОН України. | 1 | 1 | 100% |
| | 1.7. Навчальні посібники без грифу МОН України. | 1 | 2 | 200% перевиконання |
| | 1.8. Словники, довідники. | Не заплановано | – | – |
| **2.** | **Підготовка наукових кадрів:** | | | |
| | 2.1. Захищено докторських дисертацій за тематикою НДР. | | Заплановано Термін докторантури (контрактний докторант) завершується у вересні 2018 | |
| | 2.2. Подано до розгляду у спеціалізовану вчену раду докторських дисертацій за тематикою НДР. | Не заплановано | – | – |
| | 2.3. Захищено кандидатських дисертацій за тематикою НДР. | 3 | 4<br>1 | 130% |
| | 2.4. Подано до розгляду у спеціалізовану вчену раду кандидатських дисертацій за тематикою НДР. | Не заплановано | 13 | 100% перевиконання |
| | 2.5. Захищено магістерських робіт за | 12 | | 100% |

| № з/п | Критерії | Заплановано (відповідно до запиту) | Виконано (за результатами НДР) | % вико-нання |
|---|---|---|---|---|
| | | кількість | кількість | % |
| | тематикою НДР. | | | |
| **3.** | **Охоронні документи на об'єкти права інтелектуальної власності, які створено за тематикою НДР:** | | | |
| | 3.1. Отримано патентів (свідоцтв авторського права) України. | 1 | 3 | 300% |
| | 3.2. Подано заявок на отримання патенту України. | Не заплановано | – | – |
| | 3.3. Отримано патентів (свідоцтв авторського права) інших держав. | Не заплановано | – | – |
| | 3.4. Подано заявок на отримання патенту інших держав. | Не заплановано | – | – |
| **4.** | **Участь з оплатою у виконанні НДР:** | | | |
| | 4.1. Студентів. | Не заплановано | – | – |
| | 4.2. Молодих учених та аспірантів. | 4 | 5 | 200% |

10.2 Монографії та навчальні посібники

Монографії:

1) Хаханов В.І. Бібліографічний показчик. 2015. Монографія. Харків: ХНУРЕ. 284 с. (включены основные публикации). Розділ «Киберфизическая система – «облачное управление транспортом» (Cyber Physical System – Smart Cloud Traffic Control)». С. 47-82.

2) Hahanov V., Litvinova E., Chumachenko S. Green Cyber Physical Computing as Sustainable Development Model // Розділ у колективній монографії «Green IT Engineering: Components, Networks and Systems Implementation». Під редакцією V. Kharchenko. Springer-Verlag Berlin Heidelberg. 2017. Pp. 65-85. [Scopus]

3) Hahanov V., Litvinova E., Chumachenko S. Cyber Physical Computing for IoT Driven Sevices. Springer International Publisher. eBook ISBN 978-3-319-54825-8. Switzerland. 250 p.

4) Баранник В.В., Твердохлеб В.В., Хаханова А.В., Харченко Н.А. Методологическая база построения алгоритмов контроля битовой скорости видеопотока с обеспечением требуемых характеристик качества.

Монография / Под общей редакцией В. М. Безрука, В. В. Баранника. – Х. : Издательство «Лидер», 2017. — С.9-19 (ISBN 978-966-2732-78-8).

Підручники та посібники:

1) Підручник – Хаханов В. І., Хаханова І.В., Литвинова Є.І., Лобур М.В., Хаханова Г.В. Схемотехнічне проектування і моделювання НВІС (VLSI Schematic Designand Simulation) – Х., 2016. 522 с.

2) Навчальний посібник – Яворський Н., Теслюк В., Литвинова Є. Комп'ютерні методи в інженерії мікроелектромеханічних систем. – Львів, 2015. –280 с.

3) Навчальний посібник – Хаханов В.І., Чумаченко С.В., Литвинова Є.І. “Microsystem reliability and Testability”. Харків. 2016. 180 с.

Підготовлені навчальні матеріали:

1) Навчальний посібник – Хаханов В.І., Чумаченко С.В., Литвинова Є.І. “Microsystem reability and Testability”. Харків. 2016. 180 с.

2) Методичні вказівки до лабораторних робіт з дисципліни «Надійність та тестопридатність мікросистем» для студентів усіх форм навчання напряму 6.050102 „Комп’ютерна інженерія” / Упоряд. В.І. Хаханов, Є.І. Литвинова, С.В. Чумаченко – Харків:ХНУРЕ, 2016. – 31 с.

3) Методичні вказівки до лабораторних робіт з дисципліни «Microsystems Testing Techniques» для студентів усіх форм навчання спеціальності 123 „Комп’ютерна інженерія” / Упоряд. В.І. Хаханов, С.В. Чумаченко, Є.І. Литвинова – Харків: ХНУРЕ, 2016. – 68 с.

4) Методичні вказівки до лабораторних робіт з дисципліни «Автоматизація проектування комп’ютерних систем» для студентів усіх форм навчання напряму 6.050102 «Комп’ютерна інженерія» [Електронне видання] / Упоряд. І.В. Хаханова, Є.І. Литвинова. – Харків: ХНУРЕ, 2016. – 80 с.

10.3 Патенти

Отримано 3 патенти:

1) Патент на корисну модель № 102262 «Перетворювач двійково-к-значного коду в двійковий код», 26.10.2015 // Какурін М.Я., Хаханов В.І., Литвинова Є.І., Вареца В.В., Макаренко Г.М.

2) Патент на корисну модель № 111884. Перетворювач двійково-К-значного коду у двійковий код. 25.11.2016 // Какурін М. Я., Хаханов В. І., Литвинова Є. І., Вареца В. В., Макаренко Г. М.

3) Патент на корисну модель №119896. Перетворювач двійково-К-значного коду d двійковий код. 10.10.2017. Бюл. №19 // Какурін М. Я., Хаханов В. І., Литвинова Є. І., Вареца В. В., Макаренко Г. М.

10.4 Наукові статті

1. *Хаханов В.И., Багхдади Аммар Авни Аббас, Литвинова Е.И., Шкиль А.С. Кубитные структуры данных вычислительных устройств // Электронное моделирование. 2015. Т. 37. №1. С. 49-73.

2. Шкиль А.С. Поиск ошибок проектирования в поведенческих HDL-моделях методом обратного прослеживания // Радиоэлектроника и информатика. 2015. № 2. С. 19-25.

3. Зайченко С.А., Гуреев Б.Н., Шляхтун М.М., Хаханов В.И., Литвинова Е.И. Киберфизические структуры для анализа больших данных // Радиоэлектроника и информатика. 2015. № 2. С. 25-29.

4. Зайченко С.А., Мищенко А.С., Хаханов И.В., Хаханов В.И. Процессорные логические структуры для анализа BIG DATA //АСУ и приборы автоматики. 2014. Вып. 169. С.4-15.

5. Кривуля Г.Ф., Шеремет Е.В. Экспертная система функционального диагностирования технических объектов с нейронечеткой базой знани // Проблеми інформаційних технологій. 2015. № 17. С. 29-36.

6. Кривуля Г.Ф., Шеремет Е.В. Система функционального диагностирования в информационно-управляющих системах //

Інформаційно-керуючі системи на залізничному транспорті. 2015. №4. С. 18-19.

7. Кривуля Г.Ф., Шеремет Е.В. Информационные технологии для больших данных // Інформаційно-керуючі системи на залізничному транспорті. 2015. №4. С. 39-40.

8. *Багдади А.А., Хаханов В.И., Меликян В.Ш., Чумаченко С.В. Квантовые модели данных описания цифровых систем // Вестник государственного инженерного университета Армении. 2014. Т. 11. № 1. С.122-128.

9. *Manikandan Palanichamy, Areef Mohammad, Bjorn B. Larsen, Vladimir Hahanov. Selective Algorithms for Built-In Self-Test and Self-Diagnosis in Embedded SRAMS // J. Low Power Electron. (Франция). 2015. № 11. P. 541-

10. Емельянов И.В., Tamer Bani Amer, Хаханов В.И., Литвинова Е.И., Чумаченко С.В. Процессорные структуры для анализа Big Data // Радіоелектронні і комп'ютерні системи. 2016. № 6 (80). С. 163-175.

11. Хаханов И.В., Литвинова Е.И. Синтез и анализ «квантовых» моделей цифровых систем //АСУ и приборы автоматики. 2015. Вып. 172. С. 56-70.

12. Tamer Bani Amer, Любарский М.М., Хаханов В.И. Компьютинговые модели облачных сервисов // АСУ и приборы автоматики. 2015. Вып. 173. С. 48-67.

13. Tamer Bani Amer, Емельянов И.В., Хаханов И.В., Литвинова Е.И. Синтез и анализ кубитных моделей цифровых систем // АСУ и приборы автоматики. 2016. Вып. 174. С. 26-31.

14. Хаханов И.В., Литвинова Е.И. Квантовый компьютинг для проектирования цифровых систем // Радиоэлектроника и информатика. 2015. № 4(71). С. 42-45.

15. Tamer Bani Amer, Емельянов И.В., Чумаченко С.В. Кубитная форма описания вычислительных структур // Радиоэлектроника и информатика. 2016. № 1(72).С. 47-52.

16. Серокурова А.С., Шкиль А.С., Кулак Э.Н. Диагностирование HDL-моделей микропрограммных автоматов // АСУ и приборы автоматики. 2015. Вып. 172. С. 43-52.

17. Шкиль А.С. Поиск ошибок проектирования в поведенческих HDL-моделях методом обратного прослеживания // Радиоэлектроника и информатика. 2015. № 2(69). С. 19-25.

18. Гребенюк А.С., Мирошник М.А., Пахомов Ю.В., Филиппенко И.В. Методы синтеза легкотестируемых цифровых автоматов // Інформаційно-керуючі системи на залізничному транспорті. 2016. № 5. С. 28-39.

19. Krivoulya G.F., Lipchansky A.I. Using Wireless Sensor Networks and Unmanned Vehicles for precision agriculture // Науково-технічний журнал «Енергетика та комп'ютерно-інтегровані технології в АПК». Харків. 2016. № 1(3). С.68-70.

20. Tamer Bani Amer, Емельянов И.В., Любарский М.М., Хаханов В.И. Синтез Q-тестов по кубитному описанию функциональностей // Радиоэлектроника и информатика. 2016. № 2(72). С. 38-48.

21. Обризан В.И. Инфраструктура проектирования SOC для метода мультиверсного синтеза // Радиоэлектроника и информатика. 2016. № 2(72). С. 48-60.

22. Шкиль А.С., Мирошник М.А., Кулак Э.Н., Гребенюк А.С., Кучеренко Д.С. Обнаружение ошибок проектирования в HDL-моделях конечных автоматов с использованием синхронизирующих последовательностей // Радиоєлектроника и информатика. 2016. №3. С. 39-46.

23. Зиарманд А.Н. Модели и методы мониторинга и управления транспортом [Текст] / А.Н. Зиарманд, В.И. Хаханов // Радиоэлектроника и информатика. – 2016. – №3(74). – С. 71-87. (Входить до міжнародних наукометричних баз Index Copernicus, Google Scholar, OECSP, OAJI, Scholar Steer, SIS, Cyberleninka, CiteFactor, TIU Hannover, I2OR).

24. *Hahanov V. Cloud-Driven Traffic Monitoring and Control Based on Smart Virtual Infrastructure [Text] / V. Hahanov, Wajeb Gharibi, E. Litvinova, S.

Chumachenko, A. Ziarmand, I. Englesi, I. Gritsuk, V. Volkov, A. Khakhanova // SAE Technical Paper. USA. 2017-01-0092, 2017. 6 p. doi:10.4271/2017-01-0092. (Входить до міжнародної наукометричної бази Scopus, ORCID).

25. Баранник В.В., Сидченко С.А., Баранник Д.В. Метод криптокомпрессионного представления изображений на основе двухкаскадного обобщенного позиционного кодирования в базисе по верхним границам // Радиоэлектроника и информатика. 2017. №1. С. 22-27.

26. Твердохлеб В.В., Хаханова А.В., Баранник В.В. Методологическая база построения алгоритмов контроля битовой скорости видеопотока с обеспечением требуемых характеристик качества // Радиоэлектроника и информатика. 2017. № 2. С. 27-33.

27. *Зиарманд А.Н. Модель облачного сервиса для поиска оптимального пути [Текст] / С. В. Чумаченко, Е. И. Литвинова, В. И. Хаханов, А. Н. Зиарманд // Paradigmata poznání. Vědecko vydavatelské centrum «Sociosféra-CZ», s.r.o., Praha, Česká republika. – 2017. – Вып. 3. – С. 63-83. doi: 10.24045/pp.2017.3.6 (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).

28. *Iemelianov Igor, Liubarskyi Mykhailo, Hahanov Vladimir; Gharibi Wajeb, Chumachenko Svetlana, Litvinova Eugenia. Quantum Data Structures for SoC Component Testing // International Journal of Design, Analysis & Tools for Integrated Circuits & Systems. Oct2017, Vol. 6. Issue 1. P.23.

29. Бояджян А.Г., Котляров А.С. Синтез кубитных покрытий для цифровых систем // Радиоэлектроника и информатика. 2017. №3. С.28-32.

30. Литвинова Є.І., Ємельянов І.В., Хаханов І.В. Хмарний сервіс для тестування і верифікації систем на кристалах // Радиоэлектроника и информатика. 2017. №3. С. 60-69.

31. *Хаханов В.И., Емельянов И.В., Любарский М.М., Чумаченко С.В., Литвинова Е.И., Тамер Бани Амер. Кубитный метод дедуктивного анализа

неисправностей для логических схем // Электронное моделирование. 2017. Том 39, N 6. 29 с. [Cambridge Scientific Abstracts, Computer and Information Systems Abstracts, INIS Collection, Inspec, ВИНИТИ РАН]

10.5 Тези доповідей на конференціях

1. Серокурова А.С. Научный руководитель – доц. Шкиль А. С. Стратегия верификации HDL-моделей цифровых устройств // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С.7-8.

2. Щербин Д.А., Пащенко А.А. Научный руководитель – д-р техн. наук, проф. В.И. Хаханов. Кубит-векторная модель вычислительного автомата // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С.11-12.

3. Грек Ю.Ю., Панасенко А.Н. Научный руководитель – канд. техн. наук, доц. А.В. Хаханова. Квантоподобные структуры данных для построения высокопроизводительных детерминированных вычислителей // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С.13-14.

4. Климов А.В. Научный руководитель – д-р техн. наук, проф. В.И. Хаханов. Аппарат факторного анализа в компьютерном зрении // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С.15-16.

5. Сила И.В., Хаханова Ю.В. Научный руководитель – д-р техн. наук, проф. Е.И. Литвинова. «Квантовый» подход для проектирования виртуальных компьютеров // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С.17-18.

6. Мороз К.Э., Мусиенко Б.Г. Научный руководитель – д-р техн. наук, проф. Литвинова Е.И. Анализ HDL-моделей систем на кристаллах //

Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С.21-22.

7. Зайченко В.Г. Научный руководитель – канд. техн. наук, доцент Зайченко С.А. Методы структурно-функциональной верификации пересечений тактових доменов // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С. 25-26.

8. Климов А.В. Научный руководитель – д-р техн. наук, проф. Кривуля Г.Ф. Компьютерные средства для обеспечения автономного функционирования дронов // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С.28-29.

9. Обризан В. И., Ломова Ю.В. Мультиарендные облачные сервисы // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С. 34-35.

10. Ларченко Б.Д. Научный руководитель – д-р техн. наук, проф. Чумаченко С.В. Архитектура частной облачной системы // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С.36-37.

11. Хаханов И.В., Ильченко А.М. Научный руководитель – д-р техн. наук, проф. С.В. Чумаченко. Формальна модель хмарного управління транспортом // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С.38-39.

12. Скоробогатый М.В., Таранович В.П. Научный руководитель – д-р техн. наук, проф. С.В. Чумаченко. Big Data analysis // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С.40-41.

13. Васильджагаз А.И., Куприков А.В. Научный руководитель – д-р техн. наук, проф. В.И. Хаханов. Киберинформационная система мониторинга

и управления // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С.42-43.

14. Щербак В.К., Пономарева В.И. Научный руководитель – д-р техн. наук, проф. Е.И. Литвинова. Кібер-метрика управління персоналом для ринка України // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С.44-45.

15. Мизь В.А., Ломова Ю.В. Научный руководитель – д-р техн. наук, проф. С.В. Чумаченко. Big Data driven киберфизическая система // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С.46-47.

16. Руденко О.А., Палюх Г.Н. Научный руководитель – канд. техн. наук, доц. А.В. Хаханова. Технологии Big Data для киберпространства // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С.48-49.

17. Хаханова А.В., Чугуров И.И. Научный руководитель – д-р техн. наук, проф. С.В. Чумаченко. Модель кіберуправління «Smart Cyber University» // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С.50-51.

18. Арефьев А.А. Научный руководитель – д.т.н., проф. Чумаченко С.В. Энергоэффективное управление виртуальными машинами в облачных датацентрах // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С.52-53.

19. Мазур В.Ю., Лемешев А.С. Научный руководитель – канд. техн. наук, доц. Ларченко Л. В. Облачная технология управления ресурсами хостинг-компании // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С.54-55.

20. Мусиенко Б.Г., Мороз К.Э. Научный руководитель – д-р техн. наук, проф. Чумаченко С.В. Киберфизическая система – «умный город» // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С. 62-63.

21. Данильченко В.В. Научный руководитель – канд. техн. наук, проф. Немченко В.П. Анализ методов моделирования сенсорных сетей, использующихся в технологии Internet of things // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С.66-67.

22. Грек Ю.Ю., Шульга Я.Н. Научный руководитель – канд. техн. наук, доц. Филиппенко И.В. Облачный сервис распознавания речи Yandex Speechkit Cloud // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С.74-75.

23. Макаревич А.О. Научный руководитель – д-р техн. наук, проф. Хаханов В.И. Большие данные: технологии и проблемы. Обзор // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С.84-85.

24. Шестак А.С. Научный руководитель – д-р техн. наук, проф. Кривуля Г.Ф. Интеллектуальная компьютерная система управления автотранспортными средствами // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 20-22 апреля 2015. Ч. 5. С.178-179.

25. *Hahanov I., Dementiev S., Arefiev A., Hahanova Irina, Tamer Bani Amer. Automation MQT-model for virtual computer design // Матеріали XIII Міжнародної науково-технічної конференції CADSM 2015 «Досвід розробки та застосування приладо-технологічних САПР в мікроелектроніці». 24-27 лютого 2015. Львів – Поляна. С.66-69.

26. *Adamov A., Sorudeykin K., Hahanov V., Chumachenko S., Litvinova E. Structures for information retrieval in Big Data // Матеріали XIII Міжнародної науково-технічної конференції CADSM 2015 «Досвід розробки та

застосування приладо-технологічних САПР в мікроелектроніці». 24-27 лютого 2015. Львів – Поляна. С.70-75.

27. *Miz V., Mishenko A., Shcherbin D., Hahanov V., Litvinova E. Big Data driven cyber physical systems // Матеріали XIII Міжнародної науково-технічної конференції CADSM 2015 «Досвід розробки та застосування приладо-технологічних САПР в мікроелектроніці». 24-27 лютого 2015. Львів – Поляна. С.76-80.

28. *Hahanova Yu.,Yemelyanov I., Obrizan V., Krulevska D., Skorobogatiy M., Hahanova A. Metric for Analyzing Big Data // Матеріали XIII Міжнародної науково-технічної конференції CADSM 2015 «Досвід розробки та застосування приладо-технологічних САПР в мікроелектроніці». 24-27 лютого 2015. Львів – Поляна. С.81-83.

29. *Ziarmand A., Pryimak A., Chumachenko S., Shkill A., Hahanova A. Quantum Data Structures for SoC Design // Матеріали XIII Міжнародної науково-технічної конференції CADSM 2015 «Досвід розробки та застосування приладо-технологічних САПР в мікроелектроніці». 24-27 лютого 2015. Львів – Поляна. С.355-357.

30. *Yemelyanov I., Obrizan V., Hahanov I., Hahanov V. «Quantum» diagnosis and simulation of SoC // Proc. of XIth international conference “Perspective technologies and methods in mems design”. Lviv-Polyana. 2-6 September 2015. P.58-60.

31. Krivoulya G. F., Lipchansky A. I., Stephen N. The use unmanned vehicles to collect data for smart farm // Матеріали Міжнародної науково-практичної інтернет-конференції «Комп'ютерні технології в міському та регіональному господарстві». Харків: ХНУМГ. № 5. 2015.

32. Вербенець В. А., Величко Д.С., Кривуля Г. Ф. Применение беспроводной технологии M2M для сбора информации в производственных объектах // Матеріали Міжнародної науково-практичної інтернет-конференції «Комп'ютерні технології в міському та регіональному господарстві». Харків. ХНУМГ. № 5. 2015.

33. Шевцов О.И., Тищенко Д.В., Кривуля Г.Ф. Мониторинг промышленных предприятий с использованием беспроводных технологи // Матеріали Міжнародної науково-практичної інтернет-конференції «Комп'ютерні технології в міському та регіональному господарстві». Харків. ХНУМГ. № 5. 2015. С.

34. Копин, В. И., Юрченко В. Ю., Немченко В. П. Использование технологии Internet of Things в городском хозяйстве // Матеріали Міжнародної науково-практичної інтернет-конференції «Комп'ютерні технології в міському та регіональному господарстві». Харків. ХНУМГ.№ 5. 2015. С.

35. Изотов А. С., Яценко А. П., Немченко В. П. Использование булевых производных в тестировании сетевых протоколов // Матеріали Міжнародної науково-практичної інтернет-конференції «Комп'ютерні технології в міському та регіональному господарстві». Харків. ХНУМГ.№ 5. 2015. С.

36. Кривуля Г.Ф., Шеремет Е.В. Нейронечеткая система функционального диагностирования для мониторинга аномалий в промышленных системах // Материалы 15 международной научно-технической конференции «Проблеми інформатики та управління». Харьков–Одесса, 2015. С. 5.

37. *Igor Iemelianov, Dmitry Shcherbin, Vladimir Hahanov, Wajeb Gharibi «Quantum» Processor for Digital Systems Analysis // Proc. of IEEE East-West Design and Test Symposium. Batumi. 26-29 September 2015. P.104-110.

38. *Ivan Hahanov, Wajeb Gharibi, Vladimir Hahanov, Eugenia Litvinova. «Quantum». Structures for Digital Systems Synthesis // Proc. of IEEE East-West Design and Test Symposium. Batumi. 26-29 September 2015. P.115-121.

39. *Sergiy Zaychenko, Pavlo Leshtaev, Bogdan Gureev, Maksym Shliakhtun. Structural CDC Analysis Methods // Proc. of IEEE East-West Design and Test Symposium. Batumi. 26-29 September 2015. P.122-125.

40. *Alexander Adamov, Anders Carlsson. A Sandboxing Method to Protect Cloud Cyberspace // Proc. of IEEE East-West Design and Test Symposium. Batumi. 26-29 September 2015. P.126-128.

41. *Alexander Mishchenko, Mazen Abdelrahman Abdelaziz Hussein, Anastasya Hahanova, Inna Filippenko. Vladimir Hahanov, Svetlana Chumachenko, CyUni Service – Smart Cyber University // Proc. of IEEE East-West Design and Test Symposium. Batumi. 26-29 September 2015. P.149-153.

42. *Konstantin Gerasimenko, Aleksey Pryimak, Tamer Bani Amer. Method for Functional Testing Critical Control Systems // Proc. of IEEE East-West Design and Test Symposium. Batumi. 26-29 September 2015. P.122-125.

43. *Barannik V.V., Ryabukha Yu.N., Hahanova A.V., Musienko A.P. The methods of intellectual processing of video frames to enhance their semantic integrity and efficiency of delivery in aeromonitor systems // Proc. of IEEE East-West Design and Test Symposium. Batumi. 26-29 September 2015. P.379-382.

44. *Vladimir Barannik, Vladimir Larin, Anna Hahanova, Sergey Podlesny. A mathematical model of the process of transferring video // Proc. of IEEE East-West Design and Test Symposium. Batumi. 26-29 September 2015. P.383-385.

45. *Alexander Adamov. Android ransomware: turning cryptolocker into crypto unlocker // Proc. of the 25th virus bulletin international conference. Prague.30 September – 2 October 2015. P. 220-223.

46. *Ivan Hahanov, Vladimir Hahanov, Svetlana Chumachenko, Tamer Bani Amer. Cloud-Driven Traffic Control: Feasibiliti and Advanteges // Proc. of the 4th Mediterranean Conference on Embedded Computing (MECO). 14-18 June 2015. Budva. Montenegro. IEEE Catalog Number 978-1-4799-18999-7. P.17-20.

47. *Vladimir Hahanov, Alexey Zhalilo, Wajeb Gharibi, Eugenia Litvinova. Cloud-Driven Traffic Control: Formal Modeling and Technical Realization // Proc. of the 4th Mediterranean Conference on Embedded Computing (MECO). 14-18 June 2015. Budva. Montenegro. IEEE Catalog Number 978-1-4799-18999-7. P.21-24.

48. *Vladimir Hahanov, Svetlana Chumachenko, Eugenia Litvinova. Diagnosis Method for SoC HDL Model // Proc. of Microtechnology and Thermal Problems in Electronics (Microtherm). Лодзь. 23-25 June 2015. P. 72-77.

49. *Ivan Hahanov, Vladimir Hahanov, Tamer Bani Amer. MQT-model for Virtual Computer Design // Proc. of Microtechnology and Thermal Problems in Electronics (Microtherm). 23-25 June 2015. P. 182-185.

50. *Vladimir Hahanov, Eugenia Litvinova, Wajeb Gharibi, Svetlana Chumachenko. Big Data Driven Cyber Analytic System // IEEE International Congress on Big Data. New-York. USA. 2015. P.615-622.

51. Хаханов В., Гариби В., Чумаченко С., Литвинова Є. (Vladimir Hahanov, Wajeb Gharibi, Svetlana Chumachenko, Evgeniya Litvinova). Кіберфізична система хмарного управління транспортом (Cyber Physical System of Cloud Traffic Control) // The 4th International Academic Congress “Science and Education in the Modern World”. Vol. II. New Zealand, Auckland. 5-7 January 2015. P. 1011-1020.

52. Хаханов В., Тамер Бані Амер, Чумаченко С., Литвинова Є. (Vladimir Hahanov, Bani Amer Tamer, Svetlana Chumachenko, Evgeniya Litvinova). Розумне управління ресурсами (Smart Resources Control) //The 4th International Academic Congress “Science and Education in the Modern World”. Vol. II. New Zealand, Auckland. 5-7 January 2015. P. 1021-1034.

53. Володченко Д.Д., Журба С.В. Научный руководитель – д-р техн. наук, проф. Кривуля Г.Ф. Сенсорная сеть IC эклогичной лаборатории // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5. С.5-6.

54. Дерявый Д.С., Величко Д.С. Научный руководитель – д-р техн. наук, проф. Хаханова И.В. Методология верификации // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5. С.7-8.

55. Семенко О.С. Науковий керівник – к.т.н, проф. В.П. Немченко. Підхід до моделювання мережених протоколів // Материалы XX

Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5. С.9-10.

56. Hahanov I., Hahanova A. Научный руководитель – д-р техн. наук, проф. Е.И. Литвинова. «Quantum» processor for digital systems analysis // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5.С.11-12.

57. Бражникова М.Н. Научный руководитель – д.т.н., проф. Хаханов В.И. Молекулярная электроника – будущее компьютинга // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5. С.13-14.

58. Сердюк С.С. Научный руководитель – д-р техн. наук, проф. Е.И. Литвинова. Эффективное использование облачных сервисов в мобильной разработке // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5.С.15-16.

Стасюк Я.В. Науковий керівник – д.т.н., проф. Кривуля Г.Ф. Обнаружение аномалий в промышленных объектах с использованием беспроводных технологий // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5. С.17-18.

59. Серокурова А. С. Научный руководитель – доц. Шкиль А.С. Стратегия диагностирования HDL-моделей микропрограммных автоматов // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5.С.19-20.

60. Гончаренко Д.А. Научный руководитель – доц. Филипенко И.В. Критерии выбора микроконтроллера // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5. С.21-22.

61. Пенькова А.М. Научный руководитель – доц. Филипенко И.В. Особенности контроля и отладки МПС на различных этапах жизненного

цикла // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5. С.23-24.

62. Tamer Bani Amer, Емельянов Игорь. Научныи руководитель – д.т.н., проф. Хаханов В.И. Кибер-компьютинг – новый бренд IoT-рынка // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5. С.36-37.

63. Сурмеева Ю.С. Научный руководитель – доц. Филипенко И.В. Обзор современных технологий для беспроводных сенсорних сетей // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5. С.38-39.

64. Гуртовой Н.А. Научный руководитель – ст. пр. каф. АПВТ Обризан В. И. DOCKER-виртуализация приложений // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5. С.40-41.

65. Сиваков А.Ю. Научный руководитель – доц. Филлипенко И.В. Особенности применнения технологий беспроводной сети BLUETOOTH // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5. С.42-43.

66. Котляров А.С. Научный руководитель – кандидат техн.наук, Литвинова Е.И. Виды микроэлектромеханических систем // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5. С.46-47.

67. Чуприна А.А. Научный руководитель – д.т.н., проф. Литвинова Е.И. Модели надежности МЭМС // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5. С.48-49.

68. Загреба С.Н., Попов В.А. Научный руководитель – д.т.н., проф.В.И. Хаханов, д.т.н., проф. Чумаченко С.В. Модель киберсистемы для управления транспортом // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5. С.50-51.

69. Ломова Ю.В., Межевич А.А. Научный руководитель – кандидат техн.наук, доц. Хаханова А.В. Микроэлектромеханические системы // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5. С.52-53.

70. Гуртовой Н.А. Научный руководитель – ст. пр. каф. АПВТ Обризан В. И. Google cloud platform– инструменты разработчика // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5. С.54-55.

71. Копін В.І., Яценко О.П. Науковий керівник – к.т.н., проф. Немченко В.П. Використання технології Internet of Things в міському господарстві // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5. С.56-57.

72. Яценко О.П., Копін В.І. Науковий керівник – к.т.н., проф. Немченко В.П. Використання булевих похідних в тестуванні мережевих протоколів // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5. С.58-59.

73. Величко Д.С., Дерявий Д.С. Научный руководитель – к.т.н, проф. Немченко В. П. Застосування бездротових технологій для збору та передачі інформації про стан виробничого об'єкта // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5. С.62-63.

74. Соклакова Т.И. Научный руководитель – д.т.н., проф., акад. Хаханов В.И. Основные компоненты Big Data аналитики // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5. С. 66-67.

75. Межевич А. А. Научный руководитель – ст. пр. Обризан В. И. Internet computing – smart everything // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5. С.72-73.

76. Ломова Ю. В. Научный руководитель – ст. пр. каф. АПВТ Обризан В. И. Микросервисная облачная архитектура // Материалы XX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 19-21 апреля 2016. Ч. 5. С.74-75.

77. *Soklakova T., Iemelianov I., Tamer Bani Amer, Hahanov I. Technological culture of Big Data // Матеріали XIII Міжнародної конференції TCSET 2016.23-26 лютого. 2016. Львів – Славське. С.549-554.

78. Гребенюк А.С., Шкиль А.С., Кулак Э.Н. Шаблон среды верификации для диагностирования HDL-моделей микропрограммных автоматов // Сучасні напрямки розвитку інформаційно-комунікаційних технологій та засобів управління: матеріали 6-ї міжнародної наук.-практич. конф., Полтава, 21-22 квітня, 2016. Полтава, ПНТУ, 2015. С. 20.

79. Кривуля Г. Ф., Липчанский А.И. Непрерывный мониторинг информационных систем как средство повышения кибербезопасности // Матеріали першої міжнародної науково-практичної конференції «Проблеми науково-технічного та правового забезпечення кібербезпеки у сучасному світі –ПНПЗК-2016. Харків.С.7-8.

80. Krivoulya G.F., Lipchansky A.I. The new way of collecting data using Wireless Sensor Network and Unmanned Aerial Vehicle // Матеріали міжнародної науково-практичної конференції "Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління". Полтава. 2016. С.12.

81. Krivoulya G.F., Lipchansky A.I. Диагностирование распределенных обьектов с применением летающих сенсорных сетей // Матеріали міжнародної науково-практичної конференції "Проблеми інформатики та моделювання". Харків-Одеса. 2016. С. 44.

82. Кривуля Г.Ф., Липчанский А.И. Интеллектуальный беспроводный мониторинг объектов железнодорожного транспорта // Матеріали міжнародної науково-практичної конференції "Інформаційно-керуючі

системи на залізничному транспорті" Чорноморськ, 27-29 вересня 2016р. С. 25-26.

83. Lipchansky A. New approach for collecting data in Wire less Sensor Network // Матеріали міжнародної науково-практичної конференції "Інформаційно-керуючі системи на залізничному транспорті". Чорноморськ, 27-29 вересня 2016. С. 12-13.

84. Гребенюк А.С., Шкиль А.С., Кулак Э.Н. Шаблон среды верификации для диагностирования HDL-моделей микропрограммных автомато // Сучасні напрямки розвитку інформаційно-комунікаційних технологій та засобів управління: матеріали 6-ї міжнародної наук.-практич. конф., Полтава, 21-22 квітня, 2016. Полтава, ПНТУ, 2015. С. 20.

85. *Mykhailo Liubarskyi, Vladimir Hahanov, Svetlana Chumachenko, Eugenia Litvinova. Qubit Description of the Functions and Structures for Computing // Proc. of IEEE East-West Design and Test Symposium. Yerevan. 14-17 October 2016. P.88-93.

86. *Ivan Hahanov, Igor Iemelianov, Wajeb Gharibi, Tamer Bani Amer. QuaSim – Cloud Service for Digital Circuits Simulation // Proc. of IEEE East-West Design and Test Symposium. Yerevan. 14-17 October 2016. P.363-370.

87. *Artur Ziarmand, Dariya Kucherenko, Tetiana Soklakova. Transport Monitoring and Control Systems // Proc. of IEEE East-West Design and Test Symposium. Yerevan. 14-17 October 2016. P. 474-477.

88. *Vladimir Obrizan, Tetiana Soklakova. Multiversion Parallel Synthesis of Digital Structures Based on SystemC Specification // Proc. of IEEE East-West Design and Test Symposium. Yerevan. 14-17 October 2016. P. 482-486.

89. *Alexander Adamov, Anders Carlsson. Cloud Incident Response Model // Proc. of IEEE East-West Design and Test Symposium. Yerevan. 14-17 October 2016. P.487-489.

90. *Oleg Kulitsa, Vitaly Tverdokhleb, Dmitry Okladnoy, Anna Hahanova. The Development Method for Evaluating the Saturation of Video Frame Blocks to

Reduce the Processing Time of the Video Stream // Proc. of IEEE East-West Design and Test Symposium. Yerevan. 14-17 October 2016. P.487-489.

91. *Dmitry Barannik, Ali Bekirov, Oleg Frolov, Oleh Suprun. The New Method of Secure Data Transmission on the Indirect Steganography Basis // Proc. of IEEE East-West Design and Test Symposium. Yerevan. 14-17 October 2016. P.578-581.

92. *Vladimir Hahanov, Mazen Abdelrahman Abdelaziz Hussein, Anna Hahanova, Ka Lok Man. Cyber Physical Computing // Proc. of IEEE East-West Design and Test Symposium. Yerevan. 14-17 October 2016. P. 611-618.

93. *Sheremet Ye., Krivoulya G., Lipchansky A. Expert Diagnosis of Computer Systems using Neuro-Fuzzy Knowledge Base // Proc. of IEEE East-West Design and Test Symposium. Yerevan. 14-17 October 2016. P. 619-623.

94. Адамов А.С. Digital Forensics vs. OpenStack // OpenStack Summit. Barcelona. 25-28 октября 2016. [https://www.openstack.org/summit/barcelona-2016/summit-schedule/events/16177/digital-forensics-vs-openstack]

95. *Hahanov I., Iemelianov I., Tamer Bani Amer, Timofieiev D., Chumachenko S., Hahanov V., Larchenko L. Deductive qubit fault simulation // Матеріали XIV Міжнародної науково-технічної конференції CADSM 2017 «Досвід розробки та застосування приладо-технологічних САПР в мікроелектроніці». 21-25 лютого 2017. Львів – Поляна. С.256-259.

96. *Zaychenko S.,Hurieiev B., Shliakhtun M. CDC analysis for Xilinx FPGA designs // Матеріали XIV Міжнародної науково-технічної конференції CADSM 2017 «Досвід розробки та застосування приладо-технологічних САПР в мікроелектроніці». 21-25 лютого 2017. Львів – Поляна. С.269-271.

97. Бояджян А.Г., Котляров А.С. Научный руководитель – д.т.н., проф. Хаханова И.В. Продвинутые методы верификации: Assertion In Simulation, Coverage, Acceleration On Demand // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.5-6.

98. Гога М.В. Науковий керівник – доц. Шкіль О.С. Верифікація HDL-моделей кінцевих автоматів з використанням аналізу еквівалентності ТПВ // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.7-8.

99. Изотов Е.А. Научный руководитель – Филипенко И.В. Микроконтроллеры в системах релейной защиты и автоматики // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.17-18.

100. Кондрюков С.Э., Лавров В.И. Научный руководитель – к.т.н, доц. Кулак Э.Н. Регулировка микроклимата в помещении с помощью ПИД-регулятора // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.19-20.

101. Корниенко В.Р. Научный руководитель – к.т.н., доц. Филиппенко И.В. Оптимизация решений микроконтроллерных систем // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.21-22.

102. Костюк С.А., Кондрюков С.Э. Научный руководитель – к.т.н, доц. Филиппенко И.В. Конфигурируемые системы на кристаллах на примере продуктов CYPRESS PSOC // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 2017. часть 5. 25-27 апреля. С.23-24.

103. Кульчицкий Р.А., Лавров В.И. Научный руководитель – к.т.н, доц. Филиппенко И.В. Автономная система энергообеспечения // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.25-26.

104. Ломова Ю.В., Тройно М.С., Межевич А.А. Научный руководитель – д.т.н., проф. Немченко В.П. Проектирование систем на кристаллах // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 2017. часть 5. 25-27 апреля. С.27-28.

105. Орлик В.В. Научный руководитель – доц. Филиппенко И.В. Применение нейронных сетей для аутентификации в системе умный дом // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.29-30.

106. Сурмеева Ю.С. Научный руководитель – Филипенко И.В. Сравнительная оценка сред разработки устройств на AVR микроконтроллера // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.35-36.

107. Alexander Mishchenko, Nikolay Shavlak Scientific superviser – Dr.Sc., prof. Hahanov V.I. e-Document Circulation Structure // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.42-43.

108. Aram Avetisyan (1State Engineering University of Armenia), Ivan Hahanov (NURE), Artem Lavrov (NURE) Supervisors – Dr. of Sc., Prof. Vazgen Melikyan1, Dr. of Sc., Prof. Vladimir Hahanov. Deductive Qubit Fault Simulation For The Black Box Function // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.44-45.

109. Mykhailo Liubarskyi (USA), Ivan Hahanov. Scientific supervisor – Dr.Sc., Prof. Hahanov V.I. IoT Computing Evolution // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.46-47.

110. Баранник Д.В., Залозный М. Научный руководитель – д.т.н., проф. Хаханов В.И. MAT-Computing For Physical And Virtual Space // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.50-51.

111. Барыбан М.А., Ласуков В.С. Научный руководитель – д.т.н., проф. Чумаченко С.В. Triangular Metric of the Cyber Space For Big Data Analysys // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.52-53.

112. Богатыренко А.М., Мошенский Н.Н. Научный руководитель – д.т.н., проф. Чумаченко С.В. Quantum Computing // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.56-57.

113. Данильченко В. В., Грек Ю. Ю. Научный руководитель: к.т.н., проф. Немченко В. П. Analysis of the Perfomance Wireless Local Area Network Sharing 802.11G AND 802.11N standards // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 2017. часть 5. 25-27 апреля. С.66-67.

114. Егоров Д.А. Научный руководитель – доц. Филиппенко И.В. Автоматизация тестирования в SELENIUM // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». 2017. часть 5. 25-27 апреля. С.68-69.

115. Емельянов И.В., Тимофеев Д.И., Ларченко Б.Д. Научный руководитель – д.т.н., проф. Хаханов В.И. Models of Quantum Sequential Primitives // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.70-71.

116. Ерченко А.В., Таранов В.Ю. Научный руководитель – д.т.н., проф. Литвинова Е.И. State-of-The-Art of Cyber Physical Computing // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.72-73.

117. Куликовская Ю.С., Скоробогатый М.В. Научный руководитель – д.т.н., проф. Литвинова Е.И. Commitments of Computing Development // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.76-77

118. Межевич А.А., Ломова Ю.В., Тройно М.С. Научный руководитель – д.т.н., проф. Немченко В.П. Безопасность среды облачных вычислений // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.78-79.

119. Пшеничный К.Ю., Ахундов А.Я. Научный руководитель – д.т.н., проф. Чумаченко С.В. Integral Metric For Solution Estimation // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.82-83.

120. Рустинов В.А., Мова А.Ю. Научный руководитель – д.т.н., проф. Литвинова Е.И. Quantum Description of Graph Structures // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.84-85

121. Тимофеев В.В.Научный руководитель: канд. техн. наук, проф. Шкиль А.С. Диагностировпание HDL-Модели автомата Мили с помощью WEB-Интерфейса // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.88-89.

122. Тройно М.С., Ломова Ю.В., Межевич А.А. Научный руководитель – д.т.н., проф. Немченко В.П. Модели облачных вычислений // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.90-91.

123. Усков Е.В., Мамишев Р.И. Научный руководитель – д.т.н., проф. Чумаченко С.В. Cyber-Physical Computing // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.92-93.

124. Шакура А.Г., Коврижный А.В. Научный руководитель – д.т.н., проф. Хаханов В.И. Virtual And Physical Computing Development // Материалы XXI Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». Ч. 5. 25-27 апреля 2017. С.94-95.

125. *Timofeev D. Iemelianov, I., Hahanov, I., Chumachenko, S., Hahanov, V., Larchenko, L. Deductive qubit fault simulation // Матеріали XIV Міжнародної науково-технічної конференції CADSM 2017 «Досвід розробки та застосування приладо-технологічних САПР в мікроелектроніці». 24-27 лютого. 2017. Львів – Поляна. С.256-259.

126. Хаханов В.І., Чумаченко С.В., Литвинова Є.І., Хаханова І.В. Квантовий Хасе-процесор оптимального покриття // Матеріали III Всеукраїнської науково-практичної конференції «Комп'ютерне моделювання та програмне забезпечення інформаційних систем і технологій». 28-30 вересня. 2017. Рівне. С. 88-90.

127. *Gharibi, W., Ziarmand A., Khakhanova A., Hahanov, V., Litvinova, E., Chumachenko, S., Englesi I., Gritsuk I., Volkov V. Cloud-Driven Traffic Monitoring and Control Based on Smart Virtual Infrastructure // SAE. Detroit, 3-5 April 2017.

128. *Guo, H., Man, K.L., Ren, Q., Huang, Q.,Hahanov, V., Litvinova, E., Chumachenko, S. FPGA implementation of VLC communication technology // Proceedings of the 31st IEEE International Conference on Advanced Information Networking and Applications Workshops, WAINA. 2017. 16 May 2017. P. 586-590.

129. *Mykhailo Liubarskyi, Anastasia Hahanova, Vladimir Hahanov, Eugenia Litvinova, Wajeb Gharibi. Quantum Memory-driven Computing for Test Synthesis // Proc. of IEEE East-West Design and Test Symposium. Novi Sad. October 2017. P.63-68.

130. *Dariia Kucherenko, Maryna Miroschnyk, Yuriy Pakhomov, Edward German, Alexander Shkil, Elvira Kulak. Design Automation of Testable Finite State Machines // Proc. of IEEE East-West Design and Test Symposium. Novi Sad. October 2017. P.203-208.

131. *Artur Ziarmand, Svetlana Chumachenko, Eugenia Litvinova, Vladimir Hahanov. Cloud-Driven Traffic Control: Route Service Metric // Proc. of IEEE East-West Design and Test Symposium. Novi Sad. October 2017. P.224-230.

132. *Igor Iemelianov, Ivan Hahanov, Vladimir Hahanov, Svetlana Chumachenko, Irina Hahanova. Quantum Sequencer for the Minimal Test Synthesis of Black-box Functionality // Proc. of IEEE East-West Design and Test Symposium. Novi Sad. October 2017. P.445-450.

133. *Artur Ziarmand, Vladimir Hahanov, Svetlana Chumachenko, Eugenia Litvinova. Cloud Traffic Control: Smart Traffic-Driven Streetlight // Proc. of IEEE East-West Design and Test Symposium. Novi Sad. October 2017. P.451-458.

134. *Vladimir Barannik, Yuriy Ryabukha, Dmitry Okladnoy, Mykhailo Lytvinenko, Anna Hahanova, Albert Lekakh. Calculation of the Input Coefficients of the OFDM Symbol, the Researching of the Data's Quality Reception Depending on the Type of Signal Code Construction // Proc. of IEEE East-West Design and Test Symposium. Novi Sad. October 2017. P.563-567.

135. *Ivan Hahanov, Anastasia Hahanova, Vladimir Hahanov, Eugenia Litvinova, Svetlana Chumachenko, Mykhailo Liubarskyi. Qubit deductive fault simulation of black box functionalities // Proc. of 5th Prague Embedded Systems Workshop. June 29-30, 2017. Roztoky u Prahy, Czech Republic. P.83-91.

136. *Igor Iemelianov, Mykhailo Liubarskyi, Vladimir Hahanov, Eugenia Litvinova, Svetlana Chumachenko. Qubit test synthesis for the black box functionalities // Proc. of 5th Prague Embedded Systems Workshop. June 29-30, 2017. Roztoky u Prahy, Czech Republic. P. 45-51.

137. Adamov A. Open Stack Summit. USA, Boston, 08.05.2017 – 25.05.2017.

138. Anastasia Hahanova. International Test Conference. USA. Dallas. 01.11.2017 – 07.11.2017.

139. Adamov A. 27 Virus Bulletin. Spain. Madrid. 04.10.2017 -06.10.2017

## 10.6 Договори з зацікавленими організаціями

У 2015 році укладено угоди про співробітництво з Університетом м. Павія (Італія) та Батумським державним університетом (Грузія).

## 10.7 Організація і проведення конференцій та семінарів

З 26 по 29 вересня 2015 р. в Батумі кафедрою АПОТ сумісно з Батумським державним університетом організовано 13-й Міжнародний науковий симпозіум «IEEE East-West Design & Test Symposium – 2015». Подано доповідей – 105, прийнято – 97, кількість авторів – 228, кількість країн – 27, кількість замовлених доповідей – 3, кількість університетів та компаній – 70, кількість міст – 87. Кількість учасників від кафедри АПОТ – 5, поданих доповідей – 6.

З 14 по 17 жовтня 2016 р. в Єревані кафедрою АПОТ сумісно з Єреванським інженерним університетом Арменії організовано 14-й Міжнародний науковий симпозіум «IEEE East-West Design & Test Symposium – 2016». Подано доповідей – 146, прийнято – 135, кількість авторів – 384, кількість країн – 30, кількість замовлених доповідей – 25, кількість університетів та компаній – 92, кількість міст – 115. Кількість учасників від кафедри АПОТ – 5, поданих доповідей – 11.

З 29 вересня по 2 жовтня 2017 р. в м. Нові Сад (Сербія) сумісно з університетом м. Нові Сад організовано 15-й Міжнародний науковий симпозіум «IEEE East-West Design & Test Symposium – 2017». Прийнято доповідей – 152, кількість авторів – 368, кількість країн – 32, кількість замовлених доповідей – 7, кількість університетів та компаній – 94, кількість міст – 117. Кількість учасників від кафедри АПОТ – 6, поданих доповідей – 8.

При кафедрі Автоматизації проектування обчислювальної техніки ХНУРЕ діє щотижневий науковий семінар «Актуальні проблеми кіберпростору», який об’єднує дослідників, викладачів, студентів та аспірантів кафедри, університету, а також зацікавлених вчених університетів Харкова та України. У межах семінару обговорюються питання, пов’язані з розвитком квантових обчислень, движків кіберпростору, хмаровими обчисленнями, захистом інформації у кіберпросторі, персональним віртуальним кіберкомп’ютером. Протягом 2015 року проведено 31 засідання семінарів за актуальними науково-технічними напрямками розвитку ІТ-індустрії, 2016

року – 28 семінарів за актуальними науково-технічними напрямками розвитку ІТ-індустрії, у 2017 році – 29 семінарів.

10.8 Виконані інноваційні проекти та проекти, що виконуються

1) Виконано проект TEMPUS «Curricula Development for new speciality Master of Engineering in MEMS/NEMS Design" (MastMEMS)» 530785-TEMPUS-1-2012-1-PL-TEMPUS-JPCR сумісно з університетом «Львівська політехніка», Київським національним університетом, Технічним університетом м. Лодзь (Польща), Ліонським університетом (Франція), Університетом м. Ільмінау (Німеччина), Університетом м. Павія (Італія) на 2012 – 2016 рр. – керівники: д.т.н., проф. Хаханов В.І., д.т.н., проф. Чумаченко С.В.

3) Співробітники кафедри взяли участь у виконанні проекту ТЕМПУС ENGENSEC “Підготовка наступного покоління експертів з кібер-безпеки: нова визнана ЄС магістерська програма” (“Educating the Next generation experts in Cyber Security: the new EU-recognized Master’s program (ENGENSEC)”) 544455-TEMPUS-1-2013-1-SE-TEMPUS-JPCR, Blekinge Institute of Technology (BTH), Karlskrona, Sweden, 2014-2017 рр. (виконавці від кафедри АПОТ – ас. Адамов А.С., ст. викл. Обрізан В.І.), координатор від України – каф. ТКС ХНУРЕ.

10.9 Отримані гранти

Таблиця 10.2 – Гранти, отримані у межах НДР

| Тип гранту | Кількість грантів |
|---|---|
| **На стажування/підвищення кваліфікації:**<br>**2015 рік**<br>**1) 4 гранти** на магістерське стажування в Лодзинському політехнічному університеті (Польща) – магістранти Загреба С., Попов В., Сердюк С., Черкашин В. листопад 2015.<br>**2) 1 грант** на стажування в Плімутському університеті (Великобританія, грудень 2014 р. – січень 2015 р.) – д-р техн. наук, проф. Кривуля Г.Ф. Мета – вивчення сучасних методів проектування та діагностики комп’ютерних систем та мереж. | **15+23+1** |

| | |
|---|---|
| **3) 2 гранти** на інтенсивні технологічні магістерські курси в Ліонському університеті (Франція) TEMPUS MastMst – магістранти Ларченко Б., Мороз К. 17-30 травня 2015.<br>**4) 2 гранти** на магістерське стажування в Лодзинському політехнічному університеті (Польща) – магістранти Мусієнко Б., Гурєєв Б., квітень-червень 2015.<br>**5) 6 грантів** на участь у Літній школі TEMPUS MastMst – 4 студенти та 2 викладачі, 28.08.2015 – 13.09.2015<br>**2016 рік**<br>**1) 4 гранти** на магістерське стажування в Лодзинському політехнічному університеті (Польща) – магістранти Загреба С., Попов В., Сердюк С., Черкашин В. листопад 2015.<br>**2) 2 гранти** на інтенсивні технологічні магістерські курси в Ліонському університеті (Франція) TEMPUS MastMst – магістранти Володченко Д., Крамаренко О. травень 2016.<br>**3) 2 гранти** на магістерське стажування (дипломування) в Університеті Павія (Італія) – магістранти Велічко Б., Максимов М., червень-липень 2016.<br>**4) 7 грантів** на участь у Літній школі TEMPUS MastMst – 5 студентів та 2 викладачі, 28.08.2016 – 13.09.2016<br>**5) 1 грант** на стажування викладачів у Лодзинському політехнічному університеті (Польща) – доц. Ларченко Л.В., 19.09.2016 – 02.10.2016.<br>**6) 2 гранти** на стажування інженерного персоналу в Лодзинському політехнічному університеті (Польща) – інж. Зіарманд А.Н., ст.. лаб. Залозний М., 09.10.2016 – 14.10.2016.<br>**7) 4 особи** (проф. Хаханов В.І., проф. Чумаченко С.В., проф. Литвинова Є.І., доц. Хаханова Г.В.) – стажування за освітнім проектом TEMPUS MastMst, Ільменау, Німеччина, червень 2016 р.<br>**8) 1 грант** на читання лекцій у межах виконання проекту ТЕМПУС ENGENSEC в університеті Блекінге, Швеція, вересень-жовтень 2016 – ст. викл. каф. АПОТ Адамов О.С.<br>**2017 рік**<br>**1) 1 грант** на читання лекцій у межах виконання проекту ТЕМПУС ENGENSEC в університеті Блекінге, Швеція, вересень-жовтень 2016 – ст. викл. каф. АПОТ Адамов О.С. | |
| **На організацію науково-навчальних заходів:**<br>**2015 рік**<br>**1) 1 грант** на організацію 13-го Міжнародного наукового симпозіуму «IEEE East-West Design & Test Symposium – 2015», що пройшов 26 - 29 вересня 2015 у м. Батумі за ініціативою Харківського національного університету радіоелектроніки та за підтримкою Батумського державного університету.<br>**2) 3 гранти** на участь у нараді університетів-партнерів щодо виконання проекту ТЕМПУС 25-27.02.2015 – д-р техн. наук, проф. Хаханов В.І.; д-р техн. наук, проф. Чумаченко С.В.; д-р техн. наук, проф. Литвинова Є.І.<br>**3) 1 грант** на читання лекцій у межах виконання проекту ТЕМПУС ENGENSEC в університеті Блекінге, Швеція, вересень 2015 – ст. викл. каф. АПОТ Адамов О.С.<br>**4) 1 грант** на поїздку до м. Брюсель (Бельгія) для узгодження питань міжуніверситетського співробітництва – проф. Немченко | **6+10+1** |

| | |
|---|---|
| В.П., березень 2015.<br>**2016 рік**<br>**1) 1 грант** на організацію 14-го Міжнародного наукового симпозіуму «IEEE East-West Design & Test Symposium – 2016», що пройшов 14 - 17 жовтня 2016 у Єревані за ініціативою Харківського національного університету радіоелектроніки та за підтримкою Єреванського інженерного університету Вірменії.<br>**2) 4 особи** (проф. Хаханов В.І., проф. Чумаченко С.В., проф. Литвинова Є.І., доц. Хаханова Г.В.) – участь у нараді університетів-партнерів за освітнім проектом TEMPUS MastMst, Павія, Італія, травень 2016 р. (накази №342К від 26.04.2016, №363К від 29.04.2016)<br>**3) 3 гранти** на участь у нараді університетів-партнерів щодо виконання проекту ТЕМПУС у м. Київ 22-24.09.2016 – д-р техн. наук, проф. Хаханов В.І.; д-р техн. наук, проф. Чумаченко С.В.; д-р техн. наук, проф. Литвинова Є.І.<br>**4) 2 гранти** на участь у нараді університетів-партнерів щодо виконання проекту ТЕМПУС у м. Лодзь 04-08.10.2016 – д-р техн. наук, проф. Хаханов В.І.; д-р техн. наук, проф. Чумаченко С.В.<br>**2017 рік**<br>**1) 1 грант** на організацію симпозіуму EWDTS 2017 Сербія, 29.09.2017 – 02.10.2017 | |
| **На участь у конференції/олімпіаді:**<br>**2015 рік**<br>**1) 1 грант** на участь в олімпіаді з мікроелектроніки, м. Єреван, Вірменія, жовтень 2015 (студент гр. КІ-13-4 Бояджян А.)<br>**2) 3 гранти на участь** у конференції MECO**,** Чорногорія, – д-р техн. наук, проф. Хаханов В.І.; д-р техн. наук, проф. Чумаченко С.В., д-р техн. наук, проф. Литвинова Є.І. 14-21 червня 2015;<br>**3) 1 грант на участь у конференції Microterm** Лодзь 21-24.06.2015 - д.т.н., проф. Литвинова Є.І.;<br>**4) 2 гранти на участь у конгресі BigData,** Нью-Йорк, США, 25.06.2015-04.07.2015 – д-р техн. наук, проф. Хаханов В.І.; д-р техн. наук, проф. Литвинова Є.І.<br>**5) 3 гранти на участь у симпозіумі EWDTS2015,** Батумі, – д-р техн. наук, проф. Хаханов В.І.; д-р техн. наук, проф. Чумаченко С.В., д-р техн. наук, проф. Литвинова Є.І. 26-29 вересня 2015.<br>**6) 1 грант Адамов О.С.** на участь у конференції «Design Summit», Токіо, Японія, 27.10.2015-02.11.2015<br>**2016 рік**<br>**1) 1 грант** на участь в олімпіаді з мікроелектроніки, м. Єреван, Вірменія, жовтень 2016 (студент гр. СКСм-16-1 Бояджян А.)<br>**2) 1 грант** на участь у конференції, Барселона, Іспанія (ст. викл. Адамов О.С.).<br>**3) 3 гранти на участь у симпозіумі EWDTS2016,** Єреван, – д-р техн. наук, проф. Хаханов В.І.; д-р техн. наук, проф. Чумаченко С.В., д-р техн. наук, проф. Литвинова Є.І. 14-17 жовтня 2016;<br>**4) 1 грант Адамов О.С. –** участь у нараді університетів-партнерів щодо виконання проекту ТЕМПУС ENGENSEC Saint Petersburg State University of Telecommunications (SUT) 4-8 квітня 2016;<br>**5) 1 грант (ст. викладач Адамов О.С.) –** участь у конференції «OpenStack Summit», Остін, США, наказ № 324К від 19.04.2016; | **11+17+21** |

| | |
|---|---|
| **6) 1 особа (наук. співроб. Міщенко О.С.) –** участь у конгресі BigData 2016, Сан-Франциско, США, наказ № 65Кн від 23.06.2016.<br>**7) 3 гранти на участь у конгресі BigData,** Лас Вегас, США, 26.07.2016-05.08.2016 – д-р техн. наук, проф. Хаханов В.І.; д-р техн.. наук, проф.. Чумаченко С.В., д-р техн. наук, проф. Литвинова Є.І.<br>**8) 3 гранти Соклакова Т., Зіарманд А., Хаханов І.** на участь у конференції «Design Summit», Відень, Австрія, 30.11.2016-05.12.2016<br>**2017 рік**<br>**1) 1 грант** Ст. викл. Адамов О.С. – участь у конференції Нью-Йорк, США, травень 2017<br>**2) 5 грантів** проф. Хаханов В.І., проф. Чумаченко С.В., проф. Литвинова Є.І., 3 гранти США, м. Детройт, участь у науковому конгресі "WCXTM 17: SAE World CongressExperience" 02.04.2017 – 12.04.2017.<br>**3) 3 гранти** проф. Хаханов В.І., проф. Чумаченко С.В., проф. Литвинова Є.І., на участь у науковому симпозіумі та виставці IEEE Europien Test Symposium, Кіпр, Лімасол, 27.05.2017 – 02.06.2017<br>**4) 5 грантів** проф. Хаханов В.І., Чумаченко С.В., проф. Литвинова Є.І. + 2 студенти, Чехія, м. Розтокі, Прага, участь у 5th Prague Embedded Systems Workshop (PESW 2017). June 29-30, 2017.<br>**5) 6 грантів** проф. Хаханов В.І., Чумаченко С.В., проф. Литвинова Є.І., 3 студенти на участь у симпозіумі EWDTS 2017 Сербія, 29.09.2017 – 02.10.2017.<br>6) **1 грант** на ITC/TTTC Gordon Students Service Award, Dallas, USA Магістрант Хаханова А. | |
| **РАЗОМ:** | **105 грантів** |

10.10 Теми захищених та поданих дисертацій

1) Аспірант Кучеренко Д.Ю. Моделі та методи підвищення якості експертного діагностування комп'ютерних систем з використанням продукційних правил. – Дисертація на здобуття наукового ступеня кандидата технічних наук за спеціальністю 05.13.05 – комп'ютерні системи та компоненти (наук. керівник – канд. техн. наук, доц. Шкиль О.С.). 2015 рік. Отримано диплом.

Анотація. У дисертаційній роботі вперше запропоновано моделі та методи аналізу продукційних правил на логічну коректність в системах нечіткого виводу шляхом їх перевірки на повноту, несуперечливість, зв'язність та мінімальність, що дозволило формалізувати процес формування та аналізу бази знань для підвищення якості експертного діагностування комп'ютерних систем; вдосконалена модель представлення знань експерта в

експертних системах діагностування у вигляді продукційних правил, представлених в кубічній формі з використанням кубітного кодування символів багатозначного кубічного алфавіту, що, на відміну від існуючих, дало можливість використовувати логічні операції та значно прискорити обчислення при аналізі продукційних правил; отримали подальший розвиток моделі представлення діагностичних ознак різних компонентів комп'ютерних систем у вигляді лінгвістичних змінних, що, на відміну від існуючих, дозволяє використовувати єдиний підхід до побудови бази знань експерта і розширити перелік діагностичних ознак в експертних системах діагностування; отримали подальший розвиток моделі та методи синтезу продукційних правил нечіткого логічного виводу в кубічній формі, які, на відміну від існуючих, дозволяють експерту формалізувати процес і зменшити час на створення продукційних правил з довільним числом діагностичних ознак і діапазонів їх значень.

2) Аспірант Багдаді Аммар Авні Аббас. Кубітні моделі і методи аналізу та діагностування цифрових пристроїв. – Дисертація на здобуття наукового ступеня кандидата технічних наук за спеціальністю 05.13.05 – комп’ютерні системи та компоненти (наук. керівник – д-р техн. наук, проф. Хаханов В.І.). 2015 рік. Отримано диплом.

Анотація. Мета дисертаційного дослідження – істотне підвищення швидкодії програмних і апаратних засобів аналізу цифрових пристроїв за рахунок збільшення розмірності структур даних і пам'яті на основі використання кубітних покриттів функціональних елементів, методів адресного паралельного моделювання, убудованого діагностування та ремонту, що дозволяє значно підвищити вихід придатної продукції завдяки створенню інфраструктури сервісного обслуговування. Основні результати: кубітні моделі опису цифрових систем і компонентів, які характеризуються компактністю опису таблиць істинності у формі Q-покриттів і дозволяють підвищити швидкодію програмних і апаратних засобів інтерпретативного моделювання обчислювальних пристроїв; матрична модель кубітних

примітивів для реалізації комбінаційних схем, яка характеризується адресним об'єднанням Q-покриттів на елементах пам'яті, з'єднаних у цифрову схему за допомогою вектору станів ліній, і дає можливість відновлювати працездатність відмовних логічних примітивів; автоматна MQT-модель цифрового пристрою, яка характеризується використанням тільки адресованих структур пам'яті та операції транзакції для програмної і апаратної реалізації комбінаційних і послідовносних функціональностей та дозволяє створювати швидкодіючі та надійні обчислювачі для проектування сервісів кіберпростору; Q-метод інтерпретативного справного моделювання цифрових схем, який характеризується використанням компактних Q-покриттів замість таблиць істинності та дозволяє істотно підвищити швидкодію аналізу схеми.

3) Ст. викладач Обрізан В.І. Мультиверсний паралельний синтез цифрових структур на основі SystemC специфікації. – Дисертація на здобуття наукового ступеня кандидата технічних наук за спеціальністю 05.13.05 – комп’ютерні системи та компоненти (наук. керівник – д-р техн. наук, проф. Хаханов В.І.). Захищено 05.07.2017. Отримано диплом.

Мета дисертаційного дослідження суттєве зменшення часу проектування обчислювальних архітектур і підвищення якості цифрових виробів шляхом мультиверсного синтезу структури цифрового виробу на основі заданої специфікації в середовищі SystemC (C++) і автоматичному підборі функціональних компонентів за рахунок паралельного синтезу і верифікації архітектурних рішень системного рівня відповідно до запропонованої метрики. Основні результати: 1) Вперше запропоновано метод синтезу інтерфейсних структур і протоколів виконання транзакцій RT-рівня на основі аналізу специфікації SoC-архітектури системного рівня, який характеризується використанням двобічної стандартної шини Wishbone обміну даними між функціональними модулями, що дозволяє здійснювати мультиверсне створення компонентів цифрових систем на кристалах. 2) Вперше запропоновано метод синтезу RTL-моделей функціональностей,

який характеризується однозначним перетворенням C++ і SystemC-описів цифрових блоків системного рівня в алгоритми і структури даних автоматної моделі Мура, заданої синтезованою підмножиною мовних конструкцій VHDL, що дає можливість істотно зменшити час виконання процесів проектування, тестування і верифікації. 3) Удосконалено структури даних для опису функціональних примітивів системного рівня, які відрізняються орієнтацією на використання семантичних і синтаксичних конструкцій мови C ++ і SystemC, що дозволяє здійснювати паралельний синтез і верифікацію архітектурних рішень. 4) Удосконалено метод мультиверсного синтезу керуючих і операційних автоматів, орієнтованих на архітектурні рішення в метриці, що відрізняється мінімальним часом виконання функціональності за рахунок розпаралелювання операцій при обмеженні на апаратні витрати.

4) Аспірант Tamer Bani Amer. Хмарний сервіс-комп’ютинг для тестування та моделювання SoC компонентів. – Дисертація на здобуття наукового ступеня кандидата технічних наук за спеціальністю 05.13.05 – комп’ютерні системи та компоненти (наук. керівник – д-р техн. наук, проф. Хаханов В.І.). Захищено 05.07.2017. Отримано диплом.

Анотація. Робота пов'язана з хмарною реалізацією нової технології проектування і тестування цифрових систем хмарного управління на основі використання векторної, кубітної, квантової форми опису функцій і структур, орієнтованої на імплементацію функціональностей в елементах пам'яті. Переваги: 1) Спрощення архітектури комп'ютингу для хмарного управління транспортом, яка зводиться до однорідної регулярної моделі Memory-Address-Transaction (MAT). 2) Технологічність програмування завдань, пов'язаних з обробкою великих даних (big data analytics) на одному надпотужний memory-driven computer без взаємодії з логікою. 3) Найвищий паралелізм SIMD виконання логічних і арифметичних операцій на пам'яті, яка практично не має меж розмірності за кількістю бітів. 4) Ринок отримує нову технологію дата-центрів для обробки Yottabyte (1024) байтів інформації. 5) Індустрія отримує надійну архітектуру, оскільки будь-який несправний

компонент пам'яті, на відміну від логіки, може бути переадресовано на справний, в режимі online.

5) Зіарманд Артур Нісарович. Моделі і методи кіберфізичного комп’ютингу для цифрового моніторингу і хмарного управління транспортом. На правах рукопису. Дисертація на здобуття наукового ступеня кандидата технічних наук (доктора філософії) за спеціальністю 05.13.05 – комп'ютерні системи і компоненти. Харківський національний університет радіоелектроніки, Міністерство освіти і науки України, Харків, 2017. (Прийнято до захисту. Захист у січні 2018).

Мета дослідження – підвищення якості та безпеки дорожнього руху за рахунок створення кіберфізичної моделі комп’ютингової online взаємодії водія з хмарними сервісами керування автомобілем на основі цифрового моніторингу дорожньої інфраструктури і транспортних потоків, при використанні розумних сенсорів, засобів телекомунікації та навігації. Основні результати: 1) Вперше запропоновано модель транспортного комп’ютингу, яка характеризується кіберфізичною взаємодією автомобіля з хмарним сервісом за допомогою еволюційного переміщення світлофора з фізичного у віртуальний простір для цифрового моніторингу транспортних потоків і квазіоптимального управління дорожнім рухом. 2) Вперше запропоновано архітектуру розумного хмарного світлофора, яка характеризується використанням логічних операцій і часом простою зеленого сигналу, що дає можливість істотно збільшити пропускну здатність транспортних потоків на перехресті доріг. 3) Удосконалено метрика і критерії оцінювання якості інфраструктури, яка відрізняється можливістю online аналізу кіберфізичного простору для пошуку квазі-оптимального маршруту і зменшення часу його виконання. 4) Удосконалено алгоритм Дейкстри, який відрізняється можливістю аналізу кіберфізичної інфраструктури дорожнього руху для online пошуку квазіоптимального маршруту транспортного засобу в умовах виникнення колізій. Практична реалізація моделей і методів цифрового моніторингу і хмарного керування

автомобілем виконана в рамках створення і верифікації програмних компонентів кіберфізичної архітектури дорожнього руху «Cloud Traffic Control» з подальшим тестуванням модельних потоків транспортних засобів на ділянках дорожньої інфраструктури.

## 10.11 Виставки

Таблиця 10.3 – Участь у виставках

| № | П.І.Б. розроблювачів експонату | Назва експонату | Назва виставки | Дата й місце проведення виставки | Нагороди |
|---|---|---|---|---|---|
| **2015** | | | | | |
| 1 | Хаханов В.И., Зайченко А.С., Гуреев Б.Н. (Kharkov National University of Radioelectronics Aldec Inc) | Topological Verification Rules For Multi-Chip FPGA-Devices | Выставка научных разработок ХНУРЭ | ХНУРЕ, 21.12.2014 | |
| 2 | Хаханов В.И., Зиарманд А.Н. | Киберфизическая система «Green Wave Traffic Control» | Выставка научных разработок ХНУРЭ | ХНУРЕ, 21.12.2014 | |
| 3 | Хаханов В.И., Чумаченко С.В. Мусиенко Б. | TEMPUS MastMst - европейский образовательный проект (магистерские стажировки, моделирование встроенных микросистем) | Выставка научных разработок ХНУРЭ | ХНУРЭ, 21.12.2014 | |
| 4 | Зиарманд Артур Нисарович, ассистент каф. АПВТ, Мусиенко Б.Г., магистрант каф. АПВТ ХНУРЭ. Научный руководитель: д.т.н., проф. Хаханов, кафедра АПВТ. | ОБЛАЧНОЕ УПРАВЛЕНИЕ ТРАНСПОРТОМ – Smart Cloud Traffic Control | Выставка научных разработок ХНУРЭ в рамках Международного радиоэлектронного форума | ХНУРЕ, 20.04.2015 | |
| 5 | Руководитель - д.т.н., проф Хаханов В.И. Презентер - Зиарманд Артур Нисарович, ассистент каф. АПВТ. | Green Wave Traffic Control (Зеленая волна - облако управления дорожным движением). | Выставка ко Дню открытых дверей | ХНУРЕ, 08.02.2015 | |
| 6 | Хаханов В.И., Зиарманд А.Н. | Киберфизическая система «Green Wave Traffic Control» | Виставка «Тиждень науки» | ХНУРЕ, 24.05.2015 | |

| № | П.І.Б. розроблювачів експонату | Назва експонату | Назва виставки | Дата й місце проведення виставки | Нагороди |
|---|---|---|---|---|---|
| 7 | Зиарманд Артур Нисарович, ассистент каф. АПВТ, Мусиенко Богдан Геннадиевич, магистрант каф. АПВТ ХНУРЭ. Научный руководитель: д.т.н., проф. Хаханов, | ОБЛАЧНОЕ УПРАВЛЕНИЕ ТРАНСПОРТОМ - Smart Cloud Traffic Control | Виставка «Ніч науки» | ХНУРЕ, 26.09.2015 | |
| **2016** | | | | | |
| 8 | Зиарманд А. Н. | ОБЛАЧНОЕ УПРАВЛЕНИЕТРАНСПОРТОМ - Smart Cloud Traffic Control | Виставка в межах Міжнародного молодіжного форуму “Радіоелектроніка і молодь в XXI столітті” | ХНУРЕ, Квітень 2016 | |
| 9 | Гурєев Б.М., Керівники - д.т.н., проф. Хаханов В.І., к.т.н., доц. Зайченко С.А. | Топологічна верифікація правил проектування для багатокристальних FPGA-устройств (сумісна розробка з компанією Aldec Inc.). Демонстраційні матеріали: кольоровий плакат 80x80, презентація на ноутбуці. | «Освіта Слобожанщини та навчання за кордоном – 2016» | Виставковий центр «Палац Спорту», Харків, Квітень 2016 | |
| 10 | Загреба С. Керівники – д.т.н., проф. Хаханов В.І., д.т.н., проф. Чумаченко С.В. | MastMst - європейський освітній проект (модернізація навчальних планів, магістерські стажування). | «Освіта Слобожанщини та навчання за кордоном – 2016» | Виставковий центр «Палац Спорту», Харків, Квітень 2016 | |
| 11 | Костюк С. та ін. Керівник – к.т.н., доц.. Філіпенко І.В. | Розумний дім | Виставка в межах Міжнародного молодіжного форуму “Радіоелектроніка і молодь в XXI столітті” | ХНУРЕ, Квітень 2016 | **2 місце** |
| 12 | Зиарманд А. Н. | Transport Monitoring and Control Systems | Ніч науки, ХНУРЕ | ХНУРЕ, Вересень 2016 | |
| 13 | Загреба С. Керівники – д.т.н., проф. Хаханов В.І., д.т.н., проф. Чумаченко С.В. | MastMst - європейський освітній проект (модернізація навчальних планів, магістерські стажування). | «Освіта Слобожанщини та навчання за кордоном – 2016» | Виставковий центр «Палац Спорту», Харків, Жовтень 2016 | |
| 14 | д.т.н., проф. Хаханов В.І., д.т.н., проф. Чумаченко С.В. д.т.н. Литвинова Є.І., асп. | MastMst - європейський освітній проект (модернізація навчальних планів, магістерські стажування). | EWDTS 2016, Єреван, Вірменія | «AGBU Palac», Жовтень 2016 | |

| № | П.І.Б. розроблювачів експонату | Назва експонату | Назва виставки | Дата й місце проведення виставки | Нагороди |
|---|---|---|---|---|---|
| | Соклакова Т. | | | | |
| 15 | Зиарманд А. Н., Хаханов В.І. | Transport Monitoring and Control Systems | EWDTS 2016, Єреван, Вірменія | «AGBU Palac», Жовтень 2016 | |
| 16 | Костюк С., Куликівська Ю., та ін. Керівник – к.т.н., доц. Філіпенко І.В. | Розумний дім | Kharkiv Mini Maker Fair. ХНУРЕ | ХНУРЕ, 5.11.2016 | |
| 17 | д.т.н., проф. Хаханов В.І., д.т.н., проф. Чумаченко С.В., д.т.н., проф. Литвинова Є.І., асп. Соклакова Т. | MastMst - європейський освітній проект (модернізація навчальних планів, магістерські стажування). | «Освіта Слобожанщини– 2016» | Виставковий центр «Палац Спорту», Харків, 10.11.2016 – 12.11.2016 | |
| **2017** | | | | | |
| 18 | Кульчицкий Р. гр. КИ-13-5, Лавров В. КИ-13-6 | Автономная система энергообеспечения | Виставка в межах Міжнародного молодіжного форуму “Радіоелектроніка і молодь в XXI столітті” | Квітень 2017 | |
| 19 | Куликовская Ю., Костюк С. КИ-14-4 | Интерактивная бесконтактная панель | Виставка в межах Міжнародного молодіжного форуму “Радіоелектроніка і молодь в XXI столітті” | Квітень 2017 | |
| 20 | Кондрюков С., Давыдов Д. Гр. КИ-14-4, Архиреев Р. Гр. КИ-14-7 | Универсальная модульная автономная система транспортирования объектов в закрытом помещении | Виставка в межах Міжнародного молодіжного форуму “Радіоелектроніка і молодь в XXI столітті” | Квітень 2017 | |
| 21 | Корниенко В., Иванов М. Гр. КИУКИ-16-4 | Система инерциальной навигации на основе MEMS – гироскопа и акселерометра с обработкой даннях цифровым фильтром | Виставка в межах Міжнародного молодіжного форуму “Радіоелектроніка і молодь в XXI столітті” | Квітень 2017 | |
| 22 | Хаханов В.И., Чумаченко С.В., Литвинова Е.И. | Qubit test synthesis for the black box functionality | 22nd IEEE European Test Symposium, Limassol, CYPRUS \| | May 22-26, 2017 | |

## 10.12 Теми захищених магістерських робіт за тематикою НДР

1. Данильченко В.В. Засоби моделювання бездротових сенсорних мереж на базі протоколу ZigBee (гр. СКСм-14-1, керівник – проф. Немченко В.П.). 2015 рік, червень.

2. Зайченко В.Г. Методи структурно-функціональної верифікації перетинів тактових доменів (гр. СКСм-14-1, керівник – проф. Хаханов В.І.). 2015 рік, червень.

3. Клімов О.В. Спеціалізовані комп'ютерні засоби для забезпечення автономного функціонування безпілотних апаратів (гр. СКСм-14-1, керівник – проф. Кривуля Г.Ф.). 2015 рік, червень.

4. Ларченко Б.Д. Інтегрована структура мобільного та хмарного програмного забезпечення (англійською мовою, гр. СКСм-14-1, керівник –проф. Чумаченко С.В.). 2015 рік, червень.

5. Мороз Х.Е. Діагностування HDL-моделей цифрових систем на кристалі (гр. СКСм-14-1, керівник – проф. Литвинова Е.І.). 2015 рік, червень.

6. Володченко Д.Д. Сенсорна комп'ютерна мережа для екологічного моніторинга високотехнологічної лабораторії (гр. СКСм-15-1, керівник – проф. Кривуля Г.Ф.). 2017 рік, січень.

7. Загреба С.М. Формальна модель кібер системи на основі хмарних технологій для управління транспортною системою (гр.СКСм-15-1, керівник – проф. Хаханова І.В.) 2017 рік, січень.

8. Копін В.І. Аналіз і використання методів моделювання бездротових динамічно реконструйованих сенсорних мереж (гр. СКСм-15-1, керівник – проф. Немченко В.П.). 2017 рік, січень.

9. Роменський В.В. Моделі та методи підвищення якості обслуговування в мультисервісних мережах зв'язку (гр. СКСм-15-1, керівник – проф. Шкіль О.С.). 2017 рік, січень.

10. Семенко О.С. Аналіз захисту протоколів SSL/TLS при виконанні криптопримітивів (гр. СКСм-15-1, керівник – проф. Немченко В.П.) 2017 рік, січень.

11. Сердюк С.С. Методи обробки та аналізу графічних даних (гр. СКСм-15-1, керівник – проф. Літвінова Е.І.). 2017 рік, червень.

12. Стасюк Я.В. Виявлення аномалій на промислових об'єктах за використанням бездротових сенсорних мереж (гр. СКСм-15-1, керівник – проф. Кривуля Г.Ф.). 2017 рік, січень.

13. Яценко О.П. Використання теорії булевих похідних в системі генерації тестових послідовностей для повірки конформності мережевих протоколів (гр. СКСм-15-1, керівник – проф. Немченко В.П.). 2017 рік, січень.

10.13 Залучення позабюджетних коштів

У межах підтримки фінансування НДР кафедрою автоматизації проектування обчислювальної техніки було залучено кошти Європроекту «Curricula Development for New Specialization: Master of Engineering in Microsystems Design 530785-TEMPUS-1-2012-1-PL-TEMPUS-JPCR» для оснащення лабораторії проектування мікросистем загальною сумою 11900.00 Євро:

| Повний опис товару | Код по УКТЗЕД | Країна походження | Кількість |
|---|---|---|---|
| Apple (monoblock) iMac (MD093) (Apple iMac 21,5" MD093) | 8471410000 | EU | 6 |
| Microsoft Windows 8.1 PRO 32/64 wersja BOX EN/UA – lic. | 8471703000 | EU | 20 |

У вересні – жовтні 2016 отримано обладнання для лабораторії «Тестування мікросистем» межах реалізації проекту TEMPUS MastMst за кошти Європроекту загальною сумою **695267.78 грн.**

| № | Повний опис товару | Країна походження | Кількість |
|---|---|---|---|
| 1 | Цифровий мультиметр 34972A Agilent + Мультиплексор34901A+R-51B-001-3C Agilent | EU | 3 |

| № | Повний опис товару | Країна походження | Кількість |
|---|---|---|---|
| 2 | Лабораторний блок живлення DP832 | EU | 3 |
| 3 | Цифровий осцилограф Rigol DS1054Z | EU | 3 |
| 4 | Генератор сигналу довільної форми DG1022 | EU | 3 |
| 5 | Спектральний аналізатор Signal Hound USB-SA44B | EU | 2 |
| 6 | Мультиметр Rigol 3058E | EU | 3 |
| 7 | Мульиметричний гвинт-супор | EU | 2 |
| 8 | 3D Printer Flashforge Creator PRO+росхідні матеріали | EU | 1 |
| 9 | Плата для розробки ProC DEVELOPMENT KIT CY8CKIT-001C PSOC1/3/5LP | EU | 3 |
| 10 | Додаткові компоненти до плати CY8CKIT-001C | EU | 3 |
| 11 | Плата для розробки BIG8051 на основі C8051F040 | EU | 4 |
| 12 | Плата для розробки EasyMx PRO v7 for STM32 | EU | 4 |
| 13 | Модуль безпровідного зв'язку Zigbee/802,15,4 Modules Digi International XBP24BZ7SIT-004 | EU | 4 |
| 14 | GSM модуль для мікроконтролера | EU | 4 |
| 15 | GPS приймач SILVANA 68677-00 GPS-receiver | EU | 4 |
| 16 | GPS антена для мікроконтролера 66800-52 SMA 5M antenna GPS | EU | 1 |
| 17 | Набор вимірювальних лабораторних пристроїв (60 шт) | EU | 1 |
| | Разом: **695267.78 грн.** | | |

## 10.14 Розробки

Таблиця 10.4 – Розробки за тематикою виконаної НДР, що впроваджені за межами ХНУРЕ

| № з/п | Назва та автори розробки | Важливі показники, які характеризують рівень отриманого наукового результату; переваги над аналогами, економічний, соціальний ефект | Місце впровадження (назва організації, відомча належність, адреса) | Дата акту впровадження | Практичні результати, які отримано від впровадження (обладнання, обсяг отриманих коштів, налагоджено співпрацю для подальшої роботи тощо) |
|---|---|---|---|---|---|
| 1 | Методи та моделі кіберфізичної транспортної хмарної інфраструктури. Архітектура розумного хмарного світлофора | Архітектура розумного хмарного світлофора характеризується використанням логічних операцій і часом простою зеленого сигналу, що дає можливість на 10-30% зменшити час простою зеленого світлофора і істотно збільшити пропускну здатність | Перший інститут надійного програмного забезпечення | Довідка від 28.09.2017 | Впровадження розумних світлофорів в масштабах міста, дозволить скоротити інтегральний час знаходження транспорту на маршруті і витрати на енергоносії не менше, ніж на 10%. |

| № з/п | Назва та автори розробки | Важливі показники, які характеризують рівень отриманого наукового результату; переваги над аналогами, економічний, соціальний ефект | Місце впровадження (назва організації, відомча належність, адреса) | Дата акту впровадження | Практичні результати, які отримано від впровадження (обладнання, обсяг отриманих коштів, налагоджено співпрацю для подальшої роботи тощо) |
|---|---|---|---|---|---|
| | | транспортних потоків на перехресті доріг. У рамках практичної реалізації моделей і методів цифрового моніторингу і хмарного керування автомобілем з метою створення і верифікації програмних компонентів кіберфізичної архітектури дорожнього руху «Cloud Traffic Control» виконано тестування модельних потоків транспортних засобів на ділянках дорожньої інфраструктури. Достовірність результатів підтверджується проведеними експериментами, тестуванням і верифікацією запропонованих моделей і методів моніторингу та управління дорожнім рухом. Вперше запропоновано модель транспортного комп'ютингу, яка характеризується кіберфізичною взаємодією автомобіля з хмарним сервісом | | | |

| № з/п | Назва та автори розробки | Важливі показники, які характеризують рівень отриманого наукового результату; переваги над аналогами, економічний, соціальний ефект | Місце впровадження (назва організації, відомча належність, адреса) | Дата акту впровадження | Практичні результати, які отримано від впровадження (обладнання, обсяг отриманих коштів, налагоджено співпрацю для подальшої роботи тощо) |
|---|---|---|---|---|---|
| | | за допомогою еволюційного переміщення світлофора з фізичного у віртуальний простір для цифрового моніторингу транспортних потоків і квазіоптимального управління дорожнім рухом. | | | |
| 2 | Хмарні комп'ютингові сервіси тестування і верифікації цифрових пристроїв метою підвищення якості проектування цифрових виробів. | Моделі і методи кубітного синтезу та аналізу цифрових пристроїв, хмарних комп'ютингових сервісів тестування і верифікації дозволило на 12 % зменшити загальний час розробки, необхідний для створення і виходу придатної продукції на ринок. Встановлено, що розроблені моделі і методи кубітного тестування, моделювання та оцінки якості вхідних наборів реалізовані у вигляді хмарних мікросервісів, потенційно дозволяють на 5-6 % підвищити вихід придатної продукції за рахунок online відновлення | ООО «Альдек-КТС» | Довідка про впровадження від 25.11.2016 | Основними результатами, впровадженими в виробничий процес слід вважати: векторні структури даних і кубітние методи синтезу, тестування і моделювання, інтегровані в хмарну інфраструктуру сервісного обслуговування компонентів цифрових систем на кристалах з метою підвищення якості виробів і виходу придатної продукції за рахунок адресованих всіх обчислювальних процесів і явищ. |

| № з/п | Назва та автори розробки | Важливі показники, які характеризують рівень отриманого наукового результату; переваги над аналогами, економічний, соціальний ефект | Місце впровадження (назва організації, відомча належність, адреса) | Дата акту впровадження | Практичні результати, які отримано від впровадження (обладнання, обсяг отриманих коштів, налагоджено співпрацю для подальшої роботи тощо) |
|---|---|---|---|---|---|
| | | працездатності memory-driven логічних схем. | | | |
| 3 | Метод мультиверсного проектування HDL-моделей цифрових пристроїв | Мультиверсний метод проектування HDL-моделей цифрових пристроїв, представлених на мовах опису апаратури, дозволяє знизити трудомісткість синтезу HDL-моделей цифрових пристроїв функціонально-блокового рівня при автоматизованому проектуванні. Розроблена методика проведення діагностичного експерименту при верифікації HDL-коду дозволила автоматично застосовувати різні алгоритми пошуку дефектів в залежності від форми специфікації і стилю написання HDL-моделі. | Алдек-КТС | Довідка про впровадження від 15.04.2016 | Розроблена на основі результатів дослідження методика проведення діагностичного експерименту застосовується на фірмі ООО Aldec-КТС при автоматизованому проектірованні цифрових пристроїв з використанням САПР Active-HDL і системи імітаційного моделювання Riviera ™ компанії Aldec Inc. (USA), що дозволило на 20-30% відсотків скоротити час на пошук помилок проектування. |

1) Кіберсистема "Хмарне керування транспортом" (Traffic Cyber Physical Systems – Smart Cloud Traffic Control), автори – Хаханов В.І.,

Чумаченко С.В., Литвинова Є.І., Мізь В.В., Жаліло О.О., Зіарманд А.Н. (Диплом 2 місце).

Пропонується кіберфізична система для хмарного управління транспортом як інтелектуальна віртуальна інфраструктура моніторинга та управління дорожнім рухом в реальному масштабі часу на основі використання глобальних систем позиціонування і навігації, автомобільних комп'ютерів, мобільних гаджетів, розумних світлофорів та Інтернету з метою підвищення якості та безпеки пересування транспортних засобів, мінімізації часових і матеріальних витрат при русі автомобілів за заданими маршрутами. Основна інноваційна ідея – поступове перенесення світлофорів з поверхні землі у віртуальний хмарний простір для управління транспортом, обладнаним мобільним гаджетом або автомобільним комп'ютером, на екран (лобове скло) якого виводяться: карта з дорожніми знаками, маршрут прямування, координати учасника дорожнього руху та реальні керуючі сигнали віртуального розумного світлофора. Пропонується комплекс інноваційних технологій для вирішення соціальних, гуманітарних, економічних, паливно-енергетичних, страхових, кримінальних та екологічних проблем на основі створення та застосування хмарного сервісу цифрового моніторингу та управління транспортом. Технології та функціональні компоненти інтегруються в просту для реалізації автоматну модель кіберфізичної системи інтерактивної взаємодії у реальному часі.

2) «Хмарове цифрове керування дорожнім рухом (Smart Cloud Traffic Control)». Автори: д-р техн. наук, проф. Хаханов В.І., д-р техн. наук, проф. Литвинова Є.І., д-р техн. наук, проф. Чумаченко С.В., асп. Зіарманд А.Н., асп. Мізь В.О., доц. каф. ТКС Філіпенко О.І., студент гр. КІ-12-4 Чугуров І.І.

Основні характеристики, суть розробки: Сутність розробки полягає у створенні інтелектуальної інфраструктури дорожнього руху (ІІДР) – хмарного сервісу моніторингу інфраструктури і керування дорожнім рухом у реальному масштабі часу «Зелена хвиля» на основі розробки віртуальної інфраструктури дорожнього руху, інтегрованої з вуличними дорожніми

контролерами, засобами радіочастотної ідентифікації автомобілів з метою підвищення якості й безпеки пересування транспортних засобів, мінімізації часових і матеріальних витрат при виконанні заданих маршрутів. Наукова новизна визначається системною інтеграцією хмари моніторингу і керування, блоків радіочастотної ідентифікації транспорту, а також засобів моніторингу і керування дорожньої інфраструктури, що дає можливість автоматизувати процеси оптимального керування транспортними засобами і дорожнім рухом у режимі реального часу для рішення соціальних, гуманітарних, економічних і екологічних проблем. Інноваційна привабливість розробки полягає у тому, що запропонована інтелектуальна система (інфраструктура, транспорт, хмара) моніторингу і керування дорожнім рухом відрізняється від існуючої структурною інтеграцією трьох взаємозалежних інтерактивних компонентів: 1) Існуючі сервіси електронної картографії із засобами радіолокації і радіонавігації, 2) Новий хмарний сервіс моніторингу і керування дорожнім рухом на основі дорожніх контролерів, 3) Удосконалені засоби радіочастотної ідентифікації автомобіля і доступу до хмарних сервісів для комфортного й безпечного пересування за маршрутами, оптимізації часових і матеріальних витрат.

3) «Застосування квантових обчислень для пошуку найкоротшого шляху (Q-route)». Автори: д-р техн. наук, проф. Хаханов В.І., асп. Мізь В.О.

Запропоновано новий підхід знаходження найкоротшого маршруту на графі доріг. Він заснований на теорії графів і використовує основні принципи квантових обчислень. Однією з численних переваг квантових обчислень є висока швидкодія при виконанні паралельних операцій. Запропонований підхід дозволяє знаходити декілька можливих шляхів одночасно і вибирати оптимальний маршрут на основі спеціальних метрик і параметрів. Інноваційна привабливість розробки полягає в тому, що запропонована інтелектуальна система пошуку найкоротшого маршруту відрізняється від існуючої застосуванням квантових обчислень.

4) Створення системи “Розумні дороги” для організації інфраструктури дорожнього руху (Smart Roads Infrastructure).

Запропонована інтелектуальна система (інфраструктура, транспорт, хмара) моніторингу і керування дорожнім рухом відрізняється від існуючих структурною інтеграцією трьох взаємозалежних інтерактивних компонентів: існуючі сервіси електронної картографії із засобами радіолокації і радіонавігації; новий хмарний сервіс моніторингу і керування дорожнім рухом на основі дорожніх контролерів; удосконалені засоби радіочастотної ідентифікації автомобіля і доступу до хмарних сервісів для комфортного й безпечного пересування за маршрутами, оптимізації часових і матеріальних витрат.

5) Створення системи багатоканального зв'язку «multi-link» для оптимізації дорожнього руху.

Створення системи багатоканального зв'язку «Multi -Link» для оптимізації дорожнього руху, що складається з комплексу програмно - апаратних засобів, до складу якого входять пристрої прийому - передачі даних з використанням кабельних, стільникових і високошвидкісних бездротових технологій, що використовують стаціонарні світлофори та електронні табло як стаціонарні точки контролю трафіку з метою оптимізації трафіку, забезпечення безпеки пішоходів, інвалідів, дітей, зниження кількості ДТП, зниження навантаження на всіх ділянках доріг, включених в дану систему, скорочення числа пробок і часу простою транспорту в містах, з одночасним збільшенням пропускної здатності доріг. Інноваційні аспекти та переваги проекту (розробки): пропонована система багатоканального зв'язку не має аналогів, крім того приваблива інтеграцією 3-х технологій передачі даних, здатної забезпечити не тільки оптимізацію трафіку, але швидку надійний зв'язок для контролю транспорту.

6) Створення системи «розумний світлофор» для оптимізації дорожнього руху (smart lights system).

Створення системи «Розумний світлофор» для оптимізації дорожнього руху, що складається з комплексу програмно - апаратних засобів управління дорожнім рухом, до складу якого входять пристрої радіочастотної ідентифікації автомобілів та організації контролю дорожнього руху, що використовують стаціонарні світлофори, з метою забезпечення безпеки пішоходів, інвалідів, дітей, зниження кількості ДТП, зниження навантаження на всіх ділянках доріг, включених в дану систему, скорочення числа пробок і часу простою транспорту в містах, з одночасним збільшенням пропускної здатності доріг. Інноваційні аспекти та переваги проекту (розробки): пропонована система управління дорожнім рухом не має аналогів, має високу привабливість, особливо при інтеграції з системою Smart Roads, здатної забезпечити не тільки оптимізацію трафіку, а й єдиний сервіс моніторингу та контролю всього транспорту.

7) Застосування RFID-технології для створення електронного паспорта автомобіля.

Застосування радіочастотної ідентифікації автомобілів для створення електронних RFID-паспортів з метою ведення дистанційного обліку транспорту і забезпечення постійного контролю через центральну базу даних автомобілів, що перебувають в угоні , обліку водіїв , що не сплатили штрафи, а також транспорт з вичерпаним терміном реєстрації, крім того для швидкого зчитування всієї необхідної інформації в разі ДТП. Інноваційні аспекти та переваги проекту (розробки): дана система покликана забезпечити оперативність і зручність отримання інформації про автомобілі і власника органам контролю дорожнього руху , спростити процес проходження технічного обслуговування і надати можливості для комфортного і безпечного пересування по маршруту. Система інноваційна і не має попередніх аналогів в Україні , а ті системи обліку, які існували до неї не мали такої повноти функціональних можливостей.

## 10.15 Використання результатів НДР у навчальному процесі

Отримані результати НДР використовуються у навчальному процесі ХНУРЕ, а саме:

– моделі, методи і архітектури кіберфізичної інфраструктури впроваджені в навчальний процес Харківського національного університету радіоелектроніки (акт про впровадження від 03.09.2017) при читанні курсів «Дискретна математика», «Cloud-Fog кіберфізичні системи»;

– мультиверсний метод – у навчальній дисципліні «Проектування та тестування цифрових систем на ПЛІС» для бакалаврів напрямку «Комп'ютерна інженерія» у лекційному матеріалі та практичних заняттях по темі «Сучасні методи проектування цифрових систем на ПЛІС» (довідка від 09.03.2017).

– моделі і методи кубітного тестування, моделювання та оцінки якості вхідних наборів реалізовані у вигляді хмарних мікросервісов, придатних для їх використання в навчальних цілях для курсів: «Основи проектування цифрових систем», «Прикладна теорія цифрових автоматів» і «Design and Test of Computers». Основними результатами, впровадженими в навчальний процес слід вважати: векторні структури даних і кубітние методи синтезу, тестування і моделювання, інтегровані в хмарну інфраструктуру сервісного обслуговування компонентів цифрових систем на кристалах з метою підвищення якості виробів і виходу придатної продукції за рахунок адресованих всіх обчислювальних процесів і явищ (довідка від 09.03.2017).

1) Яворський Н., Теслюк В., Литвинова Є. Комп'ютерні методи в інженерії мікроелектромеханічних систем. Львів, 2015. 280 с. (Навчальний посібник).

2) Хаханов В.І., Чумаченко С.В., Литвинова Є.І. "Microsystem reliability and Testability". Харків. 2016. 180 с. (Навчальний посібник).

3) Хаханов В. І., Хаханова І.В., Литвинова Є.І., Лобур М.В., Хаханова Г.В. Схемотехнічне проектування і моделювання НВІС (VLSI Schematic Designand Simulation). – Х., 2016. 522 с. (Підручник).

4) Методичні вказівки до лабораторних робіт з дисципліни «Надійність та тестопридатність мікросистем» для студентів усіх форм навчання напряму 6.050102 „Комп'ютерна інженерія" / Упоряд. В.І. Хаханов, Є.І. Литвинова, С.В. Чумаченко – Харків: ХНУРЕ, 2016. – 31 с.

5) Методичні вказівки до лабораторних робіт з дисципліни «Microsystems Testing Techniques» для студентів усіх форм навчання спеціальності 123 „Комп'ютерна інженерія" / Упоряд. В.І. Хаханов, С.В. Чумаченко, Є.І. Литвинова. Харків: ХНУРЕ, 2016. 68 с.

6) Методичні вказівки до лабораторних робіт з дисципліни «Автоматизація проектування комп'ютерних систем» для студентів усіх форм навчання напряму 6.050102 «Комп'ютерна інженерія» [Електронне видання] / Упоряд. І.В. Хаханова, Є.І. Литвинова. Харків: ХНУРЕ, 2016. 80 с.

7) Alexander Adamov. Malware Analysis // "Підготовка наступного покоління експертів з кібер-безпеки: нова визнана ЄС магістерська програма" ("Educating the Next generation experts in Cyber Security: the new EU-recognized Master's program (ENGENSEC)") 544455-TEMPUS-1-2013-1-SE-TEMPUS-JPCR

## 10.16 Доробок авторів проекту до початку виконання НДР

1. Volodymyr Miz, Vladimir Hahanov. Smart traffic light in terms of the Cognitive road traffic management system (CTMS) based on the Internet of Things // Proceedings of IEEE East-West Design & Test Symposium (EWDTS'2014).– 2014. – Kiev, Ukraine. – P. 103 – 107.

2. Alexander Adamov, Vladimir Hahanov, Anders Carlsson. Discovering New Indicators for Botnet Traffic Detection // Proceedings of IEEE East-West Design & Test Symposium (EWDTS'2014).– 2014.– Kiev, Ukraine.– P. 281 – 285.

3. Хаханов В.И. Киберфизическая система, облачное управление транспортом // Конференция AI Ukraine-2014. Харьков. 25 октября. 2014.

4. Vladimir Hahanov, Wajeb Gharibi, Abramova L.S., Svetlana Chumachenko, Eugenia Litvinova, Anna Hahanova, Vladimir Rustinov, Vladimir Miz, Aleksey Zhalilo, Artur Ziarmand. Cyber Physical System – Smart Cloud Traffic Control // Proceedings of IEEE East-West Design & Test Symposium (EWDTS'2014). – 2014. – Kiev, Ukraine. – P. 49 – 66.

5. Hahanov V.I., Litvinova E.I., Gharibi W., Chumachenko S. Cyber Physical System – iCloud Traffic Control // Proceedings of the 11th International Conf. on Computer Modelling and Simulation – UKSim - AMSS 2014, 26-28 March. 2014.Cambridge, United Kingdom. P.158-161.

6. Hahanov V.I., Litvinova E.I., Gharibi W., Chumachenko S. iCloud Traffic Control and Monitoring // Proceedings of the 16th International Conference on Information Technology – ITNG 2014, 7-9 April. 2014. Las Vegas, Nevada. P.62-65.

7. Hahanov V.I. Пленарный доклад "Cyber Physical System – Smart Cloud Traffic Control" на конференции "Digital System Technology: Education, Research, and Industrial Aspects". Тегеран. Иран. 6-10 февраля 2014.

8. Hahanov V., Miz V., Ziarmand A., Hahanova A., Priymak A. Cloud service for traffic control // Матеріали XIII Міжнародної конференції TCSET 2014. 25 лютого -1 березня. 2014. Львів – Славське. С. 57.

9. Хаханов Владимир, Гариби Ваджеб, Чумаченко Светлана, Литвинова Евгения, Мизь Владимир, Зиарманд Артур. Интернет-слой для киберуправления транспортом (Internet-driven Cyber Control of Traffic) // Australian science review. 2014. 12 с.

10. Хаханов В.И., Литвинова Е.И., Чумаченко С.В., Филиппенко О.И. Интеллектуальное облако управления движением (Smart Cloud Traffic Control // Радиоэлектроника и информатика. 2013. №2. С.67-76.

11. Хаханов В., Гариби В., Чумаченко С., Литвинова Є. Кіберфізична система хмарного управління транспортом // II Международный научный конгресс “Фундаментальные и прикладные исследования в Америке, Европе и Азии”. 27.09.2014. 7 с.

12. Hahanov V., Gharibi W., Baghdadi Ammar Awni Abbas, Chumachenko S., Guz O., Litvinova E. Cloud traffic monitoring and control // Proceedings of the IEEE 7th International Conference on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS).– Berlin, Germany.– September 12-14, 2013.– P. 244-248.

13. Бондаренко М.Ф., Хаханов В.И., Энглези И.П., Лобур М.В., Чумаченко С.В., Литвинова Е.И., Гузь О.А. Облако мониторинга и управления дорожным движением – зеленая волна // Материалы международного научного симпозиума «Наука в жизни современного человека».– Одесса. – 2013.– С. 80-100.

14. Hahanov V., Gharibi W., Man K.L., Litvinova E., Chumachenko S., Guz O. Intelligent Road Control and Monitoring // Lecture Notes in Electrical Engineering, 235 LNEE. – 2013.– P. 327-335.

15. Hahanov V.I., Chumachenko S.V., Litvinova E.I., Dahiri F., Dementiev S. Intellectual Traffic Control on Cloud // HPC – UA Conference «Parallel and Distributed Computing Systems». – Kharkiv. – March 13-14, 2013.– P. 130-142.

16. Хаханов В.И., Чумаченко С.В., Литвинова Е.И., Хаханова А.В. Метрика качества дорожной инфрастуктуры // Материалы Международной научной конференции “Интеллектуальные системы принятия решений и проблемы вычислительного интеллекта (ISDMCI’2013)”.– Евпатория.– 20–24 мая, 2013.– С. 313-315.

17. Хаханов В.И., Энглези И.П., Гузь О.А., Полетайкин А.Н. Современные инфраструктурные средства управления дорожным движением

в крупных городах // Матеріали 3 Міжнародної науково-практ. конф. «Проблеми підвищення рівня безпеки, комфорту та культури дорожнього руху».– ХНАДУ.– 2013.– С. 208-210.

18. Хаханов В., Чумаченко С., Мизь В., Щербин Д. Облачное управление дорожным движением // 5 Міжнародна науково-практ. конф. «Сучасні інформаційні та інноваційні технології на транспорті (MINTT-2013)».– Херсон.– 28 – 30 травня, 2013.– С. 41-43.

19. Хаханов В.И., Дементьев С.П. Интеллектуальная система «Инфраструктура – транспорт – облако» // Матеріали XIX Всеукраїнської науково-практичної конференції «Інноваційний потенціал української науки – XXI сторіччя».– Запоріжжя.– 2013.– Т. 2.– С. 79-84.

20. Hahanov V., Gharibi W., Lobur M., Litvinova E., Chumachenko S., *Saatchyan A.*, Guz O., Filippenko O., Poletaykin A., Shakhov D. Cloud «Green Wave traffic monitoring and control» // Матеріали XII Міжнародної науково-технічної конференції CADSM 2013 «Досвід розробки та застосування приладо-технологічних САПР в мікроелектроніці».– Львів – Поляна.– 19-23 лютого, 2013. – С.120-126.

21. Hahanov V.I., Guz O.A., Ziarmand A.N., Ngene Christopher Umerah, Arefjev A. Cloud Traffic Control System // Proc. of IEEE East-West Design and Test Symposium.– Rostov-on-Don, Russia. – 27-30 September, 2013.– P.72-76.

22. Хаханов В.И., Энглези И.П., Литвинова Е.И., Чумаченко С.В., Гузь О.А., Хаханова А.В. Облачная инфраструктура мониторинга и управления дорожным движением // Радіоелектронні і комп'ютерні системи.– 2013.– №5 (64).– С. 106-111.

23. Хаханов В.И., Меликян В.Ш., *Саатчян А.Г.*, Шахов Д.В.«Зеленая волна» – облако мониторинга и управления дорожным движением // Вестник

«Информационные технологии, электроника, радиотехника».– Армения.– Вып. 16 (№1).– С. 53-60.

24. Бондаренко М.Ф., Хаханов В.И., Энглези И.П., Лобур М.В., Чумаченко С.В., Литвинова Е.И., Гузь О.А. Перспективные технологии XXI века «Облако мониторинга и управления дорожным движением – зеленая волна».– Украина, Одесса. – 2013.– 149 с.

25. Хаханов В.И., Бондаренко М.Ф., Энглези И.П., Убар Р., Лобур М.В., Меликян В., Дохов А.И., Филатов В.А. Тевяшев А.Д., Ткаченко В.Ф., Филиппенко О.И., *Саатчян А.Г.*, Чумаченко С.В., Литвинова Е.И., Хаханова И.В., Белоус Н.В., Полетайкин А.Н., Чугуров И.Н., Шахов Д.В., Гузь О.А. Зеленая волна – облако мониторинга и управления дорожным движением (Green Wave Traffic on Cloud) // АСУ и приборы автоматики. – 2012. – Вып. 160. – С. 4-21.

Нагороди:

1) Золота медаль та диплом Харківського міського фонду інновацій за кращий проект в області хмарних обчислень «Cloud Green Traffic Control – інтелектуальна хмара керування інфраструктурою дорожнього руху» (2013);

2) Диплом Конкурсу інноваційних проектів за 3 місце у номінації «Кращий проект в області геоінформаційного забезпечення транспорту», проект «Створення систем «Розумні дороги» для організації інфраструктури дорожнього руху (Smart Roads Infrastracture)» (2013).

3) Диплом за кращу презентацію на Міжнародній студентській конференції та конкурсі наукових робіт з питань інформаційної безпеки «CyberSecurity for the Next Generation», 20-23.02.2014, "Kaspersky Office", Москва, Росія.


THE INNOVATION PROJECTS COMPETITION
it
KHARKIV
WINNER
2013
it
KHARKIV
Лучший проект
в области облачных вычислений
Cloud Green Traffic Control – интеллектуальное облако
управления инфраструктурой дорожного движения
Мизь Владимир Александрович



IT-KHARKIV

КОНКУРС ІННОВАЦІЙНИХ ПРОЕКТІВ

INNOVATION PROJECTS COMPETITION

# ДИПЛОМ

НАГОРОДЖУЄТЬСЯ

**Чумаченко Світлана Вікторівна**
завідувач кафедри

**Литвинова Євгенія Іванівна**
професор кафедри

**Зіарманд Артур Нісарович**
асистент кафедри
Кафедра автоматизації проектування обчислювальної техніки
Харківського національного університету радіоелектроніки

ЗА 3 МІСЦЕ У НОМІНАЦІЇ

**Кращий проект в області геоінформаційних систем та інформаційного забезпечення транспорту**

ПРОЕКТ

**Створення системи «Розумні дороги» для організації інфраструктури дорожнього руху (Smart Roads Infrastructure)**

*Голова організаційного комітету конкурсу інноваційних проектів «IT-KHARKIV»* — *А.В. Моченков*

it-kharkiv.com.ua

28 листопада 2013 р.



Kaspersky® Academy

# CyberSecurity

## for the Next Generation

International Student Conference

# CERTIFICATE

## Artur Ziarmand

participated in Russia and CIS Round 2014 of "CyberSecurity for the Next Generation", the international student conference on computer security issues held in Kaspersky Lab Office on 19–21 February 2014

and submitted the research paper:
**iCloud Traffic Control**

Moscow, 2014

Eugene Kaspersky,
CEO Kaspersky Lab
Chairman of the Conference

KASPERSKY lab


Kaspersky® Academy
CyberSecurity
for the Next Generation
International Student Conference
DIPLOMA
Artur Ziaemand
Best presentation
in Russia and CIS Round of "CyberSecurity for the Next Generation",
the international student conference on computer security issues
21 February, 2014
Moscow
Eugene Kaspersky,
CEO Kaspersky Lab
Chairman of the Conference
KASPERSKY lab

# ВИСНОВОК

Запропоноване дослідження частково вирішує *фундаментальну проблему* створення кіберфізичної системи цифрового моніторингу та оптимального управління транспортними засобами на основі використання віртуальних світлофорів, розробки надійних масштабованих комп'ютингових сервісів цифрового моніторингу і хмарного управління трафіком, яка інтегрує два найактуальніших і затребуваних наукових напрямки: транспорт і комп'ютинг для отримання якісно нових умов життя в розумному автомобілі, що використовує кіберфізичну інфраструктуру дорожнього руху. Інтелектуальне хмарне управління дорожнім рухом (Smart Cloud Traffic Control) має метою включення параметра часу в цифрову карту планети, а також поступове перенесення дорожніх знаків і світлофорів в хмари, що радикально озеленює інфраструктуру дорожнього руху на землі і створює потенційні можливості для економії: тисяч тон металу на виготовлення світлофорів і непотрібних автомобільних номерів, мільйонів кіловат годин-електроенергії на підтримку працездатності, мільйонів доларів на установку світлофорів, дорожніх знаків та експлуатаційні витрати, а також зменшення часу установки і актуалізації світлофорів у віртуальній інфраструктурі міст до декількох хвилин. Вирішені наступні завдання:

1) Розробка і реалізація алгоритмів квазіоптимального управління світлофором, оптимізації маршрутів руху з використанням історії.

2) Розробка технологічного кібер-шару проекту у формі взаємодії програмно-апаратних компонентів кібер-фізичної системи, що інтегрує існуючі та нові розробки: картографію, світлофори, автомобільні комп'ютери в єдину автоматну модель управління транспортом у реальному часі.

3) Створення технічного та організаційного кібер-шарів проекту для подальшого виконання заходів по інтегруванню реальної керуючої інфраструктури доріг та автомобілів з віртуальним кіберпростором.

Були отримані, науково обґрунтовані та доведені наступні результати для застосування та використання: 1) Створення квантових структур даних для паралельного зберігання даних і матричного аналізу Big Data форматів кіберпростору. 2) Розробка квантоподібних віртуальних паралельних процесорів для швидкого пошуку, розпізнавання та прийняття квазіоптимальних рішень з управління транспортом та інфраструктурою міст. 3) Створення комплексу програм для реалізації хмарних сервісів моніторингу та управління транспортом у реальному часі на основі використання віртуальних хмарних світлофорів. 4) Розробка і практична реалізація моделі експериментальної ділянки міста для управління дорожнім рухом за допомогою хмарних світлофорів, моніторингу та управління транспортними засобами з анімаційною візуалізацією всіх масштабованих процесів на екрані монітора. 5) Верифікація та впровадження програмних і технічних засобів хмарного сервісу моніторингу та управління дорожнім рухом шляхом використання віртуальних світлофорів, а також імплементація сервісних програмних продуктів для водія при русі.

*Ринкова привабливість дослідження.* Впровадження хмарного сервісу управління транспортом призведе до збереження екології планети і до зменшення: 1) часу проходження замовлених маршрутів, 2) споживання енергетичних ресурсів і матеріальних витрат на створення і експлуатацію світлофорів, дорожніх знаків, автомобільних номерів, 3) числа аварій і крадіжок автомобілів.

Для досягнення поставленої *мети* – підвищення якості та безпеки дорожнього руху за рахунок створення кібер-фізичної системи – інтелектуальної інфраструктури дорожнього руху (TCS), що включає моніторинг і управління у реальному масштабі часу на основі використання мобільних гаджетів транспортних засобів і хмарних (віртуальних) світлофорів, що дає можливість підвищити якість життя водія, мінімізувати часові і матеріальні витрати при організації дорожнього руху, створювати інноваційні

рішення соціальних, гуманітарних, економічних та екологічних проблем, – були вирішені завдання, що мають *наукову новизну*, а саме:

1) *Вперше запропоновано* модель транспортного комп’ютингу, яка характеризується кіберфізичною взаємодією автомобіля з хмарним сервісом за допомогою еволюційного переміщення світлофора з фізичного у віртуальний простір для цифрового моніторингу транспортних потоків і квазіоптимального управління дорожнім рухом.

2) *Вперше запропоновано* архітектуру розумного хмарного світлофора, яка характеризується використанням логічних операцій і часом простою зеленого сигналу, що дає можливість істотно збільшити пропускну здатність транспортних потоків на перехресті доріг.

3) *Удосконалено* метрика і критерії оцінювання якості інфраструктури, яка відрізняється можливістю online аналізу кіберфізичного простору для пошуку квазі-оптимального маршруту і зменшення часу його виконання.

4) *Удосконалено* алгоритм Дейкстри, який відрізняється можливістю аналізу кіберфізичної інфраструктури дорожнього руху для online пошуку квазіоптимального маршруту транспортного засобу в умовах виникнення колізій.

5) *Запропоновано* мультиверсний паралельний синтез цифрових структур моніторингу та управління трафиком на основі SystemC специфікації, що використовується при створенні модуля Sherlock, який призначений для точного моніторингу даних щодо позиціонування та функціонування автомобиля в процесі виконання заданого маршруту. Мультиверсний синтез застосовується при створенні программних продуктів хмарного сервісу Cloud Traffic Control, а також при розв'язанні задач тестування та діагностування програмно-апаратних компонентів кіберфізичної системи.

6) *Запропоновано* нову модель сталого розвитку кіберфізичного комп'ютингу та напрями її використання для прогнозування аттракторів в області ІТ-індустрії на основі просторово-часового аналізу технологічних

процесів, які характеризуються трьома фазами розвитку і дають можливість прогнозувати майбутні тренди розвитку кіберкультури планети.

8) *Удосконалено* кубітно-векторні моделі для опису структур і компонентів цифрових систем на базі адресного кодування вхідних сигналів, які відрізняються від аналогів технологічністю і швидкодією побудованих на їх основі процедур тестування, логічного синтезу та аналізу.

9) *Удосконалено* метод паралельного регістрового моделювання компонентів цифрових систем на кристалах, який відрізняється від аналогів застосуванням автоматної МАТ-моделі, що використовує тільки адресовні структури пам'яті і операції транзакції. Запропоновано новий підхід до проектування цифрових пристроїв, який характеризується застосуванням методів синтезу та аналізу, які базуються на суперпозиції кубіт-векторних примітивів візначення всіх типів функциональностей, імплементованих в елементи пам'яті, що дає можливість істотно підвищити швидкодію засобів моделювання, тестування і верифікації, а також значно спростити процедури створення реальних і віртуальних комп'ютерних систем.

10) *Удосконалено* методи синтезу тестів на основі булевих похідних і дедуктивного моделювання несправностей, які відрізняються паралельним виконанням логічних регістрових операцій над кубітними покриттями схемних компонентів, що дає можливість істотно зменшити час верифікації та тестування цифрового пристрою.

11) *Створено* хмарну інфраструктуру сервісного обслуговування для online проектування, тестування і верифікації цифрових проектів, яка відрізняється візуалізацією цифрових схем і доступністю мікросервісів моделювання для зменшення періоду налагодження HDL-коду.

12) *Здійснено* тестову верифікацію компонентів хмарної інфраструктури сервісного обслуговування, а також методів тестування, моделювання, синтезу та аналізу на реальних прикладах цифрових схем і елементів. Використано мови програмування: SWIFT, C++, Verilog, Python 2.7 і платформи: Microsoft Windows, X Window і Macintosh OS X. Наукові

результати дозволили підвищити швидкодію процедур аналізу структур даних, тестування та інтерпретативного адресного моделювання цифрових схем, на 15% зменшити час налагодження HDL-проектів в процесі проектування SoC.

*Практична реалізація* моделей і методів цифрового моніторингу і хмарного керування автомобілем виконана в рамках створення і верифікації програмних компонентів кіберфізичної архітектури дорожнього руху «Cloud Traffic Control» з подальшим тестуванням модельних потоків транспортних засобів на ділянках дорожньої інфраструктури.

Отримані в процесі виконання дослідження наукові висновки і результати є достовірними, що підтверджується проведеними експериментами, тестуванням і верифікацією запропонованих моделей і методів моніторингу та управління дорожнім рухом. *Практична значимість наукових досліджень* підтверджується істотним зменшенням виконання маршрутів руху при моделюванні реальних процесів на фрагментах моделей дорожньої інфраструктури.

Результати в складі моделей, методів і архітектури кіберфізичної інфраструктури впроваджені в навчальний процес Харківського національного університету радіоелектроніки (ДОДАТОК Б) при читанні курсів: «Дискретна математика», «Cloud-Fog кіберфізичні системи». Отримані моделі, методи та озроблена архітектура розумного хмарного світлофора, а також модель транспортного комп’ютингу можуть бути імплементовані як компонент проекту при синтезі хмарного сервісу (ДОДАТОК Б).

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. The e-IRGSP4 project for e-IRG, Best Practices for the use of e-Infrastructures by large-scale research infrastructures. – 2015 February 24. – http://e-irg.eu/.

2.http://e-irg.eu/documents/10920/277005/Best+Practices+for+the+use+of+e-Infrastructures+by+large-scale+research+infrastructures.pdf

3. Geoffrey C. Fox, Web 2.0, grids and cyberinfrastructure / International Symposium on e-infrastructure Collaborative Technologies and Systems (CTS 2007). http://ieeexplore.ieee.org/document/4621721/]

4. Sy Holsinger, Sergio Andreozzi, EGI: Implementing service management in a largescale e-Infrastructure // IEEE Network Operations and Management Symposium (NOMS). – 2014. – http://ieeexplore.ieee.org/document/6838371/

5. David C. H. Wallom, Matteo Turilli, Michel Drescher, Federating Infrastructure as a Service Cloud Computing Systems to Create a Uniform E-infrastructure for Research // IEEE 11th International Conference on e-Science (e-Science). – 2015. – http://ieeexplore.ieee.org/document/7304286/

6. E-infrastructure: The ecosystem for innovation, www.bis.gov.uk

7. Richard O. Sinnott, Gerson Galang, Martin Tomko, Robert Stimson. Towards an e-Infrastructure for Urban Research across Australia // IEEE 7th International Conference on E-Science (e-Science). – 2011. – http://ieeexplore.ieee.org/document/6123291/

8. Geoffrey C. Fox, Web 2.0, grids and cyberinfrastructure / Collaborative Technologies and Systems e-infrastructure (CTS 2007). – 2007. International Symposium on, http://ieeexplore.ieee.org/document/4621721/.

9. Luong Thi Thu Phuong, Nguyen Trung Hieu, Jin Wang, Sungyoung Lee, Young-Koo Lee, Energy Efficiency based on Quality of Data for Cyber Physical Systems // IEEE International Conferences on Internet of Things, and Cyber, Physical and Social Computing. – 2011. – P.232-241.

10. Jane Hunter, Charles Brooking, Wilfred Brimblecombe, Ross G. Dwyer, Hamish A. Campbell, Matthew E. Watts, Craig E. Franklin, OzTrack. E-

Infrastructure to Support the Management, Analysis and Sharing of Animal Tracking Data, eScience (eScience) // 2013 IEEE 9th International Conference, http://ieeexplore.ieee.org/document/6683901/

11. Sy Holsinger, Sergio Andreozzi, EGI: Implementing service management in a largescale e-Infrastructure, Network Operations and Management Symposium (NOMS), 2014 IEEE, http://ieeexplore.ieee.org/document/6838371/.

12. Sheikh Mahbub Habib, Sebastian Ries, Max Muhlhauser, Towards a Trust Management System for Cloud Computing // IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom). –2011.

13. Mohemed Almorsy, John Grundy, Amani S. Ibrahim. Collaboration-Based Cloud Computing Security Management Framework // 2011 IEEE International Conference on Cloud Computing (CLOUD), http://ieeexplore.ieee.org/document/6008731/

14. Mahmoud Al-Ayyoub, Yaser Jararweh, LoAi Tawalbeh, Elhadj Benkhelifa, Anas Basalamah, Power Optimization of Large Scale Mobile Cloud Computing Systems // 3rd International Conference on Future Internet of Things and Cloud (FiCloud). – 2015. – http://ieeexplore.ieee.org/document/7300885/

15. Zhen Jiang Li and Cheng Chen, Cloud Computing for Agent-Based Urban Transportation Systems // Intelligence transportation systems, published by IEEE Computer Society. – January/February 2011. – P.73-79.

16. Drescher M. MS518 EGI Platform Roadmap // June 2014, [online] Available: https://documents.egi.eu/document/2232.

17. Habib, Sheikh Mahbub et al. “Trust as a facilitator in cloud computing: a survey.” *Journal of Cloud Computing: Advances, Systems and Applications*. – 2012. – P. 1-18. DOI: 10.1186/2192-113X-1-19

18. Sasko Ristov, Marjan Gusev, Aleksandar Donevski, "Security Vulnerability Assessment of OpenStack Cloud", Conference on Sixth International Computational Intelligence Communication Systems and Networks (CICSyN) – 2014. – Pp. 95-100,

19. New England E-infrastructure: http://www.rcuk.ac.uk/research/xrcprogrammes/OtherProgs/eInfrastructure/

20. Sinnott R. O., Bayliss C., Galang G., Greenwood P., Koetsier G., Mannix D., Morandini L., Nino-Ruiz M., Pettit C., Tomko M., Sarwar M., Stimson R., Voorsluys W., Widjaja I. A data-driven urban research environment for Australia // IEEE 8th International Conference on E-Science (e-Science). – 2012. – Pp. 1-8.

21. Tawalbeh L., Jararweh Y., Ababneh F., Dosari F., Large scale cloudlets deployment for efficient mobile cloud computing // Journal of Networks. – 2015. – Vol. 10, no. 01.

22. Chronopoulos A.T. , Johnston C.M. A real-time traffic simulation using a communication latency hiding parallelization // IEEE Transactions on Vehicular Technology. – 2002. – Vol. 51, Issue 3. – Pp. 498 – 510. DOI: 10.1109/TVT.2002.1002499

23. Swapnaja Hiray, Rajesh Ingle. Context-Aware Middleware in Cyber Physical Cloud (CAMCPC) // International Conference on Cloud & Ubiquitous Computing & Emerging Technologies (CUBE). – 2013. – Pp. 42-47, 2013.

24. Arini Widhiasi, Vasuky Mohanan, Muhammad Fermi Pasha, Rahmat Budiarto. Vertical Handover Scheme for Car-to-Car Communication based on IEEE 802.21 Standard // Second International Conference on Computer Engineering and Applications. – 2010. – P. 143-147.

25. Deyi Li, Xiaodong Wang, Wen He, Mu Guo, Tianlei Zhang, Study on Interaction Behaviours of Micro-Autonomous Vehicles // Tenth International Symposium on Autonomous Decentralized Systems. – 2011. – P. 399-406.

26. Zheng Li, Nenghai Yu, Zhuo Hao, A Novel Parallel Traffic Control Mechanism for Cloud Computing // 2nd IEEE International Conference on Cloud Computing Technology and Science. – 30 Nov. – 3 Dec. 2010. – P.376-382.

27. Visit Hirankitti, Jaturapith Krohkaew, An agent approach for intelligent traffic-light control // Proc. of the First Asia International Conference on Modelling & Simulation (AMS'07). – 27-30 March 2007.

28. Tian Hai, Zhang Yong, Cui Gui-mei, Design and realization of intelligent traffic light monitor and control system based on wireless control // 2012 31st Chinese Control Conference (CCC). – 25-27 July 2012. – http://ieeexplore.ieee.org/document/6390937/.

29. Smith R. H. and Chin D. C., Evaluation of an adaptive traffic control technique with underlying system changes // Winter Simulation Conference. – 1995. – P. 1124–1130.

30. Chen W., Chen L., Long Chen Z., and Liang Tu S., A realtime dynamic traffic control system based on wireless sensor network // ICPP Workshops. – 2005. – P. 258 – 264.

31. Hirankitti V., Krohkaew J., and Hogger C. J., A multi-agent approach for intelligent traffic-light control // World Congress on Engineering. – 2007. – P. 116–121.

32. Mohandas B., Liscano R., and Yang O., Vehicle traffic congestion management in vehicular ad-hoc networks // LCN. – 2009. – Pp. 655 – 660.

33. Hong Y. and Yang O. W. W., Design of adaptive pi rate controller for best-effort traffic in the internet based on phase margin // IEEE Trans. Parallel Distrib. Syst. – 2007. – Vol. 18, no. 4. – Pp. 550–561.

34. Cheng Hu, Yun Wang, A Novel Intelligent Traffic Light Control Scheme // Ninth International Conference on Grid and Cloud Computing. – 2010. – Pp. 372- 376.

35. Jubair Mohammed Bilal, Don Jacob. Intelligent Traffic Control System // ICSPC. – 2007. – Pp. 496-499.

36. Haimeng Zhao, Xifeng Zheng, Weiya Liu. Intelligent Traffic Control System Based on DSP and Nios II // International Asia Conference on Informatics in Control, Automation and Robotics. – 2009. – Pp. 90-94.

37. Wu Hejun, Miao Changyun. Design of intelligent traffic light control system based on traffic flow // International Conference on Computer and Communication Technologies in Agriculture Engineering. – 2010. – Pp. 368-371.

38. Liang-Tay Lin, Hung-Jen Huang, Jim-Min Lin, Fongray Frank Young. A New Intelligent Traffic Control System For Taiwan // ITST 2009. – Pp. 138-142.

39. Chen Zhaomeng. Intelligent Traffic Control Central System of Beijing-SCOOT // MACE. – 2010. – Pp. 5067 – 5069.

40. Vivek Kumar Sehgal, Samruddh Dhope, Prince Goel. An Embedded Platform for Intelligent Traffic Control // UKSim Fourth European Modelling Symposium on Computer Modelling and Simulation. – 17-19 Nov. 2010. – P. 541- 545.

41. Lawrence Y. Deng, Nick C. Tang, Dong-liang Lee, Chin Thin Wang, Ming Chih Lu, Vision Based Adaptive Traffic Signal Control System development // Proceedings of the 19th International Conference on Advanced Information Networking and Applications (AINA). – 2005.

42. Jianwu Li, Zhengxian Huangpu. Design of fuzzy control system for city single highway intersection // Electrical Drive Automation. 2000. – Vol. 22, No. 2, Pp. 22-24.

43. Bingham E. Reinforcement learning in neuro-fuzzy traffic signal control // European Journal of Operational Research. – 2001. – 131. – Pp. 232-241.

44. Abdulhai B., Pringle R. Machine learning based adaptive signal control using autonomous Q-learning agent // Proceeding of the IASTED International Conference. Intelligent Systems and Control. – Honolulu, Hawaii, USA. – August 14-16, 2000. – Pp. 320-327.

45. Xiangjun Cheng, Zhaoxia Yang, Intelligent Traffic Signal Control Approach Based on Fuzzy-Genetic Algorithm // Fifth International Conference on Fuzzy Systems and Knowledge Discovery. – 18-20 Oct. 2008. – P. 222-225.

46. Mingshu Li, John Hallam, Louise Pryor, Stanley Chan, Ken Chong. Cooperative Intelligent System for Urban Traffic Problems // Proceedings of the IEEE International Symposium on Intelligent Control. – Dearborn, MI. – September 15-18, 1996. – Pp. 162-167.

47. Goldberg, D.E., Genetic algorithms in search optimization, and machine learning. – Addison Wesley. – 1989.

48. Jeffrey L. Adler, Victor J. Blue. A Cooperative Multi-Agent Transportation Management and Route Guidance System // Transportation Research. – 2002. – Part C, No.10. – Pp. 433-454,

49. John France and Ali A. Ghorbani. A Multiagent System for Optimizing Urban Traffic / Proceedings of the IEEE WIC International Conference on Intelligent Agent Technology (IAT'03). – 2003.

50. CHENG Xiangjun, YANG Zhaoxia, Distributed Traffic Signal Control Approach Based on Multi-Agent // Sixth International Conference on Fuzzy Systems and Knowledge Discovery. – 2009. – P.582-587.

51. Baogui Cao, Zhaosheng Yang, Haiyan Zhu, Fusion and Integration Framework Study on Road Network and Dynamic Traffic Data in Vehicle Navigation // Third International Symposium on Intelligent Information Technology Application Workshops. – 2009. – P.70-73.

52. Mohamed M. Atia, Shifei Liu, Heba Nematallah. Integrated Indoor Navigation System for Ground Vehicles with Automatic 3-D Alignment and Position Initialization // IEEE Transactions on Vehicular Technology. – 2015. – Vol. 64, Issue 4. – Pp. 1279 – 1292, http://ieeexplore.ieee.org/document/7027835/.

53. Ming Li, Zh. H. Liu, J.A. Huang. Artificial Landmark Positioning System Using Omnidirectional Vision for Agricultural Vehicle Navigation // Second International Conference on Intelligent System Design and Engineering Application (ISDEA). – 2012. – http://ieeexplore.ieee.org/document/6173294/.

54. Liu Dawei, Application of Assisted TDOA Technology in Vehicle Positioning and Navigation System // TENCON 2005. IEEE Region 10, http://ieeexplore.ieee.org/document/4085074/.

55. Ying Ji, Man Zhang, Gang Liu, Positions research of agriculture vehicle navigation system based on Radial Basis Function neural network and Particle Swarm Optimization // Natural Computation (ICNC). – 2010 Sixth International Conference on, http://ieeexplore.ieee.org/document/5583145/.

56. Mingke Fang, Lei Li, Wei Huang, Research of Hybrid Positioning Based Vehicle Interactive Navigation System // International Conference on Multimedia

Information Networking and Security (MINES). – 2010. – http://ieeexplore.ieee.org/document/5671177/.

57. Kwak S.H., McKeon J.B., Clynch J.R. Incorporation of global positioning system into autonomous underwater vehicle navigation // Proceedings of the Symposium on Autonomous Underwater Vehicle Technology (AUV'92). – 1992. – http://ieeexplore.ieee.org/document/225224/.

58. Song Ying, Yu Yang, Study on Vehicle Navigation System (VNS) with Real-time Traffic Information // International Conference on Computer Science and Software Engineering. – 2008. – P.1079-1082.

59. Kim S.-B., Bazin J.-C., Lee H.-K. Ground vehicle navigation in harsh urban conditions by integrating inertial navigation system, global positioning system, odometer and vision data // IET Radar, Sonar & Navigation. – 2011. – Vol. 5, Issue 8. – P. 814 – 823, http://ieeexplore.ieee.org/document/6036233/.

60. Masaya Yoshikawa, Hidekazu Terai, IEEE. Car Navigation System based on Hybrid Genetic Algorithm // World Congress on Computer Science and Information Engineering. – 2009. – P. 62-64.

61. Holland J., Adaptation in Natural Artificial Systems. – University of Michigan Press. – 1992.

62. Fei-Yue Wang, Agent-Based Control for Networked Traffic Management Systems // IEEE Intelligent Transportation Systems. – 2005. – P.92-96.

63. Yu Shaowei, Li Xiuhai, Liu Qingling, Zhang Tongzhu, A Car-Steering Model Based on an Adaptive Cloud Neural Controller // International Conference on Digital Manufacturing & Automation. – 2010. – P. 7-10.

64. Kiattisin Kanjanawanishkul, Andreas Zell, Path following for an omnidirectional mobile robot based on model predictive control // IEEE International Conference on Robotics and Automation (ICRA'09). – 2009. –, http://ieeexplore.ieee.org/document/5152217/.

65. Sandor P. Fekete, Christiane Schmidt, Axel Wegener, Stefan Fischer, Recognizing Traffic Jams with Hovering Data Clouds // Second International

Symposium on Leveraging Applications of Formal Methods. Verification and Validation. – 15-19 Nov 2006. – P. 199-203.

66. Yin Zhu1, Junli Wang1, Huapu Lu, Study on the Intelligent Pre-control Model of Urban Arterial // IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application. – 2008. – P.114-118.

67. You-Ren Chen, Keng-Pin Chen, Pao-Ann Hsiungy. Dynamic traffic light optimization and Control System using model-predictive control method // 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC) – http://ieeexplore.ieee.org/document/7795937/.

68. Yang Zhao-sheng, Zhou Xi-Yang, Tian Chun-Lin, Research on traffic signal cycle optimization method based on bi-level programming model // 3rd International Conference on Computer Science and Network Technology (ICCSNT). – 2013. – http://ieeexplore.ieee.org/document/6967178/.

69. Xiao-Hua Yu, Stubberud A.R. Markovian decision control for traffic signal systems // Proceedings of the 36th IEEE Conference on Decision and Control. – 1997. – http://ieeexplore.ieee.org/document/649773/.

70. Ludovica Adacher, Marco Tiriolo, Distributed urban traffic signal optimization based on macroscopic model // Sixth International Conference on Innovative Computing Technology (INTECH). – 2016 – http://ieeexplore.ieee.org/document/7845123/.

71. Glenn T. Donovan, Position Error Correction for an Autonomous Underwater Vehicle Inertial Navigation System (INS) Using a Particle Filter // IEEE Journal of Oceanic Engineering. – July 2012. – Vol. 37, Issue 3. – Pp. 431 – 445. – http://ieeexplore.ieee.org/document/6200890/

72. Lee H., Optimal estimation of vehicle position in an acoustic transponder navigation system // IEEE International Conference on Engineering in the Ocean Environment. – Ocean 73. – http://ieeexplore.ieee.org/document/1161299/

73. Pedro Batista, Carlos Silvestre, Paulo Oliveira, Position and Velocity Navigation Systems for Unmanned Vehicles // IEEE Transactions on Control Sys-

tems Technology. – May 2009. – Vol. 17, Issue 3. – P. 707 – 715. – http://ieeexplore.ieee.org/document/4798174/

74. Du J., Masters J., Barth M. Lane-level positioning for in-vehicle navigation and automated vehicle location (AVL) systems // Proceedings of the 7th International IEEE Conference on Intelligent Transportation Systems. – 2004. – http://ieeexplore.ieee.org/document/1398868/

75. Braden K., Browning C., Gelderloos H. Integrated inertial navigation system/Global Positioning System (INS/GPS) for manned return vehicle autoland application // Position Location and Navigation Symposium (IEEE PLANS). – 1990. – A Decade of Excellence in the Navigation Sciences. – http://ieeexplore.ieee.org/document/66160/

76. Satoh K., Hozumi H., Okada S. Autonomous mobile patrol system for nuclear power plants: field test report of vehicle navigation and sensor positioning // Proceedings of the 1996 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 96). – http://ieeexplore.ieee.org/document/571045/

77. Jian Bu, Rui Sun, Hongyang Bai. Integrated method for the UAV navigation sensor anomaly detection // IET Radar, Sonar & Navigation 2017. – Vol. 11, Issue 5. – P. 847 – 853 – http://ieeexplore.ieee.org/document/7914109/

78. Weishan Li, Ming Bai, Runda Lu. The design of high accuracy differential positioning vehicle terminal based on BeiDou Navigation System // http://ieeexplore.ieee.org/document/7867337/

79. Hoang Van Dung, Trinh Minh Chi, Nguyen Thi Nga, A vehicle positioning and navigation system design using Mini2440 ARM and Linux // Third World Congress on Information and Communication Technologies (WICT). – 2013. – http://ieeexplore.ieee.org/document/7113112/

80. Azura Che Soh, Mohammad Hamiruce Marhaban, Marzuki Khalid, Modelling and Optimisation of a Traffic Intersection Based on Queue Theory and Markov Decision Control Methods // First Asia International Conference on Modelling & Simulation (AMS 2007). – 2007.

http://ieeexplore.ieee.org/document/4148707/.

81. Chin D.C., Spall J.C., Smith R.H. Evaluation of system-wide traffic signal control using stochastic optimization and neural networks // Proceedings of the American Control Conference. – 1999.http://ieeexplore.ieee.org/document/786341/.

82. Samuel Coogan, Murat Arcak, Calin Belta. Formal Methods for Control of Traffic Flow // IEEE Control Systems Automated Control Synthesis from Finite-State Transition Models. – April 2017. – Vol. 37, Issue 2. – P. 109 – 128. – http://ieeexplore.ieee.org/document/7879917/

83. Kale M.M., Chipperfield A.J. Reconfigurable flight control strategies using model predictive control // Proceedings of the IEEE International Symposium on Intelligent Control. – 2002. – http://ieeexplore.ieee.org/document/1157736/

84. Trodden P., Richards A. Robust distributed model predictive control using tubes // American Control Conference. – 2006.

http://ieeexplore.ieee.org/document/1656519/

85. Center for automotive research // Annual report 2016.

86. Hitachi Inspire the Next, Hitachi automotive systems, Ltd. Guide to Hitachi Automotive Systems. – Printed in Japan. – 2016. – P. 5-12.

87. http://www.hitachi-automotive-mm.com/en/

88. Vejlupek Josef, Trailer backing-up assistant using ultrasound sensors based control units to safely back-up the car with trailer // 17th International Conference on Mechatronics – Mechatronika (ME). – 7-9 Dec. 2016.

http://ieeexplore.ieee.org/document/7827834/.

89. Sandor Marcon, Yasuhiro Komatsu, Akitomo Yamanaka. Linear motor coils as brake actuators for Multi-car elevator // International Conference Electrical Machines and Systems (ICEMS). – 8-11 Oct. 2007.

http://ieeexplore.ieee.org/document/4412276/.

90. Jakob Lombacher, Markus Hahn, Jürgen Dickmann, Detection of arbitrarily rotated parked cars based on radar sensors // 16th International Radar Symposium (IRS). – 24-26 June 2015.

http://ieeexplore.ieee.org/document/7226281/.

91. Chan Wei Hsu, Tsung Hua Hsu, Kuang Jen Chang. Implementation of car-following system using LiDAR detection // 12th International Conference ITS Telecommunications (ITST). – 5-8 Nov 2012.

http://ieeexplore.ieee.org/document/6425157/.

92. Iuliia Goncharova; Stefan Lindenmeier. An interoperable antenna for GPS and GLONASS services on a car // IEEE International Symposium on Antennas and Propagation & USNC/URSI National Radio Science Meeting. – 2015. – http://ieeexplore.ieee.org/document/7304393/.

93. Gerd Saala, Jochen Hopf, Stefan Lindenmeier. Small satellite car antenna for simultaneous reception of LHCP and RHCP signals // 3rd European Conference on Antennas and Propagation (EuCAP). – 2009. – http://ieeexplore.ieee.org/document/5068167/authors.

94. Tobias Haberle, Lambros Charissis. The Connected Car in the Cloud: A Platform for Prototyping Telematics Services // IEEE Software. – 2015. – Vol. 32, Issue 6. – Ph.11 – 17. – http://ieeexplore.ieee.org/document/7310999/.

95. Arini Widhiasi, Vasuky Mohanan, Muhammad Fermi Pasha, Rahmat Budiarto, Vertical Handover Scheme for Car-to-Car Communication, based on IEEE 802.21 Standard // Second International Conference on Computer Engineering and Applications. – 2010. – P. 143-147.

96. Deyi Li, Xiaodong Wang, Wen He, Mu Guo, Tianlei Zhang, Study on Interaction Behaviours of Micro-Autonomous Vehicles // Tenth International Symposium on Autonomous Decentralized Systems. – 2011. – P. 399-406.

97. Хаханов В.И., Меликян В.Ш, Саатчян А.Г., Шахов Д.В. «Зеленая волна» – облако мониторинга и управления дорожным движением // Армения. Вестник «Информационные технологии, электроника, радиотехника». 2013. – Вып. 16(№1). – С.53-60.

98. Hahanov V.I., Guz O.A., Ziarmand A.N., Ngene Christopher Umerah, Arefjev A. Cloud Traffic Control System // Proc. of IEEE East-West Design and Test Symposium. Rostov-on-Don. – 27-30 September 2013. – P.72-76.

99. Hahanov V., Gharibi W., Baghdadi Ammar Awni Abbas, Chumachenko S., Guz O., Litvinova E. Cloud traffic monitoring and control // Proceedings of the 2013 IEEE 7th International conference on intelligent data acquisition and advanced computing systems (IDAACS). – Berlin. – September 12-14, 2013. – P. 244-248.

100. Бондаренко М.Ф., Хаханов В.И., Литвинова Е.И. Структура логического ассоциативного мультипроцессора // Автоматика и телемеханика. – 2012. – № 10. – С. 71-92.

101. Lu Antao, Li Yushan, Sun Yufang, Cao Chongzhen, Gao Kuigang, Xu Jing. Research on the Integrated Management of Highway Based on Radio Frequency Identification Technology // Third International Conference on Measuring Technology and Mechatronics Automation (ICMTMA). – 2011. – Vol.3. – P. 116-119.

102. Pandit A.A., Talreja J., Mundra A.K. RFID Tracking System for Vehicles (RTSV) // First International Conference on Computational Intelligence, Communication Systems and Networks. – 2009. – P.160-165.

103. Jiang Lin-ying, Wang Shuai, Zhang Heng, Tan Han-qing. Improved Design of Vehicle Management System Based on RFID // International Conference on Intelligent System Design and Engineering Application (ISDEA). – 2010. – Vol. 1. – P. 844-847.

104. Chen Xue-Mei, Wei Zhong-Hua. Vehicle management system based on multi-node RFID cards // 30th Chinese Control Conference (CCC).– 2011.– P. 5497-5499.

105. Дудников С., Боенко И. Бесконтактная идентификация транспорта, основанная на RFID // Компоненты и технологии №1. – 2007. – http://www.kit-e.ru/assets/files/pdf/2007_01_140.pdf

106. Manikonda P., Yerrapragada A.K., Annasamudram S.S. Intelligent traffic management system // IEEE Conference on Sustainable Utilization and Development in Engineering and Technology (Student).– 2011. – P. 119-122.

107. Samad T. Perspectives in Control Engineering Technologies, Applications, and New Directions // Intelligent Transportation Systems: Roadway Applications. – Wiley-IEEE Press. – 2001. – P. 348 -369.

108. Schutte J. Recent trends in automatic train controls // IEEE Intelligent Transportation Systems. – 2001. – P. 813 -819.

109. Zingirian N., Valenti C. Sensor clouds for Intelligent Truck Monitoring // IEEE Intelligent Vehicles Symposium (IV). – 2012.– P. 999-1004.

110. Branisso L.B., Kato E.R.R., Pedrino E.C., Morandin O., Tsunaki R.H. An Intelligent Autonomous Vehicle Management System // Second Brazilian Conference on Critical Embedded Systems (CBSEC). – 2012. – P. 42-47.

111. Brizgalov V.V., Chukhantsev V., Fedorkin E., Architecture of traffic control systems using cloud computing // International Conference and Seminar on Micro/Nanotechnologies and Electron Devices (EDM).– 2010.– P. 215-216.

112. Frank van Diggelen and Kathy Tan (2014) Interchangeability Accomplished // Tri-Band Multi-Constellation GNSS in Smartphones and Tablets – GPS World. – June 2014. – Pp. 46-52.

113. Бабаков В.Н. Высокоточная навигация. http://www.gisa.ru/102286.html

114. Роффе А. Возможности высокоточной навигации – http://www.gisa.ru/102624.html

115. http://www.gisa.ru/103133.html

116. http://top.rbc.ru/economics/28/05/2014/926718.shtml

117. Leandro, R., Landau H., Nitschke M., Glocker M., Seeger S., Chen X., Deking A., Ben Tahar M., Zhang F., Stolz R., Talbot N., Lu G., Ferguson K., Brandl M., Gomez Pantoja V., Kipka A.. GNSS Positioning Trimble TerraSat. Positioning: the Next Generation of cm-accurate Real-time. – GmbH, Germany. – 2011. – RTX, ION-GNSS-2011, 2011, Portland, OR, USA.

118. Vollath, U., Buecherl, A., Landau, H. Pagels, C., Wagner, B. Multi-Base RTK using Virtual Reference Stations // Proceedings of ION GPS, Salt Lake City. – 2000. – P. 19-22.

119. http://www.leica-geosystems.com/en/Leica-GNSS-Spider_83498.htm

120. http://www.systemnet.com.ua/ru/network/preimushchestva

121. http://gpsinformation.net/exe/waas.html

122. http://www.esa.int/Our_Activities/Navigation/The_present_-_EGNOS/What_is_EGNOS

123. http://www.gisa.ru/103059.html

124. http://www.novatel.com/about-us/news-releases/news

125. http://www.novatel.com/products/span-gnss-inertial-systems/mems-interface-card/

126. http://www.novatel.com/products/span-gnss-inertial-systems/span-combined-systems/span

127. http://www.kharkovgnssgroup.net/

128. Ariane Hellinger, Ariane Hellinger, Heinrich Seeger. Cyber-Physical Systems. Driving force for innovation in mobility, health, energy and production. – Acatech. National Academy of Science and Engineering. – 2011. – 48p.

129. Alessandro Bazzi, Alberto Zanella, Barbara M. Masini, Gianni Pasolini. A Distributed Algorithm for Virtual Traffic Lights with IEEE 802.11p. // IEEE European Conference Networks and Communications (EuCNC). – 2014. – P. 1 – 5.

130. Ferreira, M., d'Orey, P.M. On the Impact of Virtual Traffic Lights on Carbon Emissions Mitigation // IEEE Transactions on Intelligent Transportation Systems. – 2012. – Volume 13, Issue 1. – P. 284 – 295.

131. Conceicao H., Ferreira M., Steenkiste P. Virtual traffic lights in partial deployment scenarios // IEEE Intelligent Vehicles Symposium (IV). – 2013. – P. 988 – 993.

132. Hahanov Vladimir, Wajeb Gharibi, Abramova L.S., Chumachenko Svetlana, Litvinova Eugenia; Hahanova Anna, Rustinov Vladimir, Miz Vladimir, Zhalilo Aleksey, Ziarmand Artur. Cyber physical system – smart cloud traffic control // Proceedings of IEEE East-West Design & Test Symposium (EWDTS 2014). 2014. Pages: 1 - 18, DOI: 10.1109/EWDTS.2014.7027107

133. Hahanov, V., Gharibi, W., Man, K.L., Litvinova, E., Chumachenko, S., Guz, O. Intelligent road control and monitoring // Lecture Notes in Electrical Engineering. Volume 235 LNEE, 2013, Pages 327-3352013 International Conference on Future Information and Communication Engineering, ICFICE 2013; Shenyang; China; 24 June 2013 - 26 June 2013; Code 98013.

134. Gorobets A., Chugurov I., Scherbin D., Chumachenko S. Dijkstra Algorithm for cyber structures analysis // Published in: Experience of Designing and Application of CAD Systems in Microelectronics (CADSM), 2013. 12th International Conference on the 19-23 Feb. 2013 (Матеріали XII Міжнародної науково-технічної конференції CADSM 2013 «Досвід розробки та застосування приладо-технологічних САПР в мікроелектроніці». 19-23 лютого 2013. Львів – Поляна). – С.61-65.

135. Dijkstra E. W. A note on two problems in connexion with graphs // Numerische Mathematik. – 1959. – Vol. 1. – P. 269-271.

136. Кормен Т. Х., Лейзерсон Ч. И., Ривест Р. Л., Штайн К. Алгоритмы: построение и анализ = Introduction to Algorithms. М.: Вильямс, 2006. – 1296 с.

137. Левитин А. В. Глава 9. Жадные методы: Алгоритм Дейкстры // Алгоритмы: введение в разработку и анализ = Introduction to The Design and Analysis of Algorithms. – М.: Вильямс, 2006. – С. 189-195.

138. Кузнецов Н.А., Фетисов В.Н. Алгоритм Дейкстры с улучшенной робастностью для управления маршрутизацией в IP-сетях // Автоматика и телемеханика. – 2008. – № 2. – С. 80–85.

139. Томас Т.М. Структура и реализация сетей на основе протокола OSPF. Руководство Cisco = OSPF Network Design Solutions. – М.: Вильямс, 2004. – 816 с.

140. Alessandro Bazzi, Alberto Zanella, Barbara M. Masini, Gianni Pasolini. A Distributed Algorithm for Virtual Traffic Lights with IEEE 802.11p. Networks and Communications (EuCNC), IEEE European Conference. – 2014. – P. 1 – 5.

141. Ferreira, M., d'Orey, P.M. On the Impact of Virtual Traffic Lights on Carbon Emissions Mitigation // IEEE Transactions on Intelligent Transportation Systems,. – Vol. 13, Issue 1. – 2012. – P. 284 – 295.

142. Немченко В.П. Моделирование сетевых протоколов при построении тестовых последовательностей / В.П. Немченко, *А.Н. Зиарманд*, Ю.А. Чепелев // Науково-технічний журнал «Інформаційно-керуючі системи на залізничному транспорті». – 2011. – №5(90). – С. 18-21. (Індексується GoogleScholar, Реферативна база "Наукова періодика України", РИНЦ (eLibrary), National Library of Ukraine named after Vernadsky).

143. Немченко В.П. Использование энергосберегающих технологий в современных сетях / В.П. Немченко, А.Н. Зиарманд, А.С. Изотов // Вестник Херсонского национального технического университета. – 2012. – №1(44). – С. 146-148. (Індексується РИНЦ (eLibrary), Google Scholar, National Library of Ukraine named after Vernadsky).

144. Зиарманд А.Н. Модели и методы мониторинга и управления транспортом / А.Н. Зиарманд, В.И. Хаханов // Радиоэлектроника и информатика. – 2016. – №3(74). – С. 71-87. (Входить до міжнародних наукометричних баз Index Copernicus, Google Scholar, OECSP, OAJI, Scholar Steer, SIS, Cyberleninka, CiteFactor, TIU Hannover, I2OR).

145. Vladimir Hahanov. Internet-driven Cyber Control of Traffic / Vladimir Hahanov, Wajeb Gharibi, Svetlana Chumachenko, Evgeniya Litvinova, Vladimir Miz, Arthur Ziarmand // Australian Journal of Scientific Research. – 2014. – Volume IV, No.1(5). – Pp. 217-224.

146. Hahanov V. Cloud-Driven Traffic Monitoring and Control Based on Smart Virtual Infrastructure / V. Hahanov, Wajeb Gharibi, E. Litvinova, S. Chumachenko, A. Ziarmand, I. Englesi, I. Gritsuk, V. Volkov, A. Khakhanova // SAE Technical Paper. USA. 2017-01-0092, 2017. 6 p. doi:10.4271/2017-01-0092. (Входить до міжнародної наукометричної бази Scopus, ORCID).

147. Зиарманд А.Н. Модель облачного сервиса для поиска оптимального пути / С. В. Чумаченко, Е. И. Литвинова, В. И. Хаханов, А. Н.

Зиарманд // Paradigmata poznání. Vědecko vydavatelské centrum «Sociosféra-CZ», s.r.o., Praha, Česká republika. – 2017. – Вып. 3. – С.63-83. doi: 10.24045/pp.2017.3.6 (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).

148. Зиарманд А.Н. Теоретическая суть проекта «Smart Roads» / А.Н. Зиарманд // Автоматизированные системы управления и приборы автоматики. – 2013. – Вып. 162. – С. 28-34. (Входить до міжнародних наукометричних баз Google Scholar, Cyberleninka).

149. Hahanov V. I. Cloud traffic control system / V. I. Hahanov, O. A. Gus, A. Ziarmand, Ngene Christopher Umerah, A. Arefjev // Proc. of the IEEE East-West Design & Test Symposium (EWDTS). – Rostov-on-Don, Russia. – 2013. – P. 72-76. (Входить до міжнародних наукометричних баз Scopus, IEEE Xplore).

150. Ziarmand A. Smart road infrastructure / A. Ziarmand // Proc. Of the IEEE East-West Design & Test Symposium (EWDTS). – Rostov-on-Don, Russia. – 2013. – P. 430-434. (Work in Progress). (Входить до міжнародних наукометричних баз Scopus, IEEE Xplore).

151. Ziarmand A. Cloud Service for Traffic Control / Artur Ziarmand, Vladimir Hahanov, Volodymyr Miz, Anastasya Hahanova, Aleksey Priymak // Proc. of the XII International IEEE Conference “Modern Problems of Radio Engineering, Telecommunications, and Computer Science”. 2014. – Lviv-Slavske, Ukraine. – P. 557-559. (Входить до міжнародних наукометричних баз Scopus, IEEE Xplore).

152. Hahanov V. Cyber physical system – smart cloud traffic control / V. Hahanov, Wajeb Gharibi, L.S. Abramova, S. Chumachenko, E. Litvinova, A. Hahanova, V. Rustinov, V. Miz, A. Zhalilo, A. Ziarmand // Proc. of the IEEE Design & Test Symposium. – 2014. – Kiev, Ukraine. – P. 49-66. (Входить до міжнародних наукометричних баз Scopus, IEEE Xplore).

153. Ziarmand A. Transport monitoring and control systems / A. Ziarmand, D. Kucherenko, T. Soklakova // Proc. of the IEEE East-West Design & Test Symposium (EWDTS). – Yerevan, Armenia. – 2016. – P. 474-477. (Входить до міжнародних наукометричних баз Scopus, IEEE Xplore).

154. Зиарманд А.Н. Формальная модель киберсистемы / А.Н. Зиарманд, И.И. Чугуров // Материалы XVIII Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». – Харьков, Украина. – 2014. – С. 65.

155. Зиарманд А.Н. Формальна модель хмарного управління транспортом / А.Н. Зиарманд, К.Э. Мороз // Материалы XIX Международного молодежного форума «Радиоэлектроника и молодежь в XXI веке». – Харьков, Украина. – 2015. – С. 4.

156. Зиарманд А.Н. I-Cloud Traffic Control / А.Н. Зиарманд, В.А. Мизь // 6-я Международная студенческая конференция и конкурс научных работ по вопросам информационной безопасности «CyberSecurity for the Next Generation». – 2014. – “Kaspersky Office”, Москва, РФ. – С. 13.

157. Chumachenko S. Quantum data structures for SoC design / S. Chumachenko, A. Shkil, A. Hahanova, A. Ziarmand, A. Pryimak // Proc. of the 13th International Conference “IEEE Experience of Designing and Application of CAD Systems in Microelectronics (CADSM)”. – 2015. Polyana-Lviv. – Ukraine. – P. 355-357. (Входить до міжнародних наукометричних баз Scopus, IEEE Xplore).

158. Soklakova T. Big data visualization in smart cyber university / T. Soklakova, A. Ziarmand, S. Osadchyieva // Proc. of the IEEE East-West Design & Test Symposium (EWDTS). – Yerevan, Armenia. – 2016. – P. 469-473. (Входить до міжнародних наукометричних баз Scopus, IEEE Xplore).

159. Зиарманд А.Н. Анализ компетентности пользователя компьютерной системы как диагностический эксперимент / Зиарманд А.Н., Кучеренко Д.Е. // Материалы XV Международного молодежного форума

«Радиоэлектроника и молодежь в XXI веке». – Харьков, Украина. – 2011. – С. 40-41.

160. Обрізан В.І. Інфраструктура проектування SOC для методу мультиверсного синтезу / В.І. Обрізан // Радіоелектроніка та інформатика. – 2016. – №2. – С. 48-60.

161. Kharchenko V. Concepts of Green IT Engineering: Taxonomy, Principles and Implementation / V. Kharchenko, O. Illiashenko. – Springer-Verlag Berlin Heidelberg. – 2016.

162. Бондаренко М.Ф. Структура логического ассоциативного мультипроцессора / М.Ф. Бондаренко, В.И. Хаханов, Е.И. Литвинова // Автоматика и телемеханика. – 2012. – С. 71-92.

163. Хаханов В.И. Кубитные структуры данных вычислительных устройств / В.И. Хаханов, В. Гариби, Е.И. Литвинова, А.С. Шкиль // Электронное моделирование. – 2015. – № 1. – С. 76-99.

164. Хаханов В.И. Кубитные технологии анализа и диагностирования цифровых устройств / В.И. Хаханов, Т. Бани Амер, С.В. Чумаченко, Е.И. Литвинова // Электронное моделирование. – Том 37, № 3. – С. 17-40.

165. Hahanov V. Big Data Driven Cyber Analytic System / V. Hahanov, E. Litvinova, W. Gharibi, S. Chumachenko // 2015 IEEE Int. Congress on Big Data.– New York, USA. – P. 615-622.

166. Dehorter N. Tuning of fast-spiking interneuron properties by an activity-dependent transcriptional switch / N. Dehorter, G. Ciceri, G. Bartolini, L. Lim, I. del Pino, O. Marín. – Science.– 11 Sept. 2015: 349 (6253). – P. 1216-1220.

167. P.A. Merolla A million spiking-neuron integrated circuit with a scalable communication network and interface / Paul A. Merolla et all // Science, 8 August 2014. – Vol. 345, no. 6197. – P. 668-673.

168. ГОСТ 34.003-90. Автоматизированные системы. Термины и определения.

169. ISO / IEC 38500: 2008. Corporate governance of information technology.

170. Marr B. Forbes. 9 Attractors in the Computer Market. 17 'Internet Of Things' Facts Everyone Should Read. Електронний ресурс: http://www.forbes.com/sites/bernardmarr/2015/10/27/17-mind-blowing-internet-of-things-facts-everyone-should-read/

171. Galer S. Forbes. IDC Releases Top Ten 2016 IT Market Predictions. Електронний ресурс: http://www.forbes.com/sites/sap/2015/11/05/idc-releases-top-ten-2016-it-market-predictions/

172. Бондаренко М.Ф. Квантовые технологии реализации мозгоподобных вычислительных структур академика В.М. Глушкова / М.Ф. Бондаренко, В.И. Хаханов // Матеріали XIX Міжнародної конференції з автоматичного управління. – Київ. – 26-28 вересня, 2012. – С. 32-33.

173. Палагин А.В. Архитектурно-структурная организация компьютерных средств класса “Процессор-в-памяти” / А.В. Палагин, Ю.С Яковлев., Б.М. Тихонов, И.М. Першко // Математичні машини і системи. – 2005. – № 3. – 14 с.

174. Палагин А.В. Системная интеграция средств компьютерной техники. / А.В. Палагин, Ю.С. Яковлев. – Винница: “Универсум”.– 2005.

175. Большой взрыв. Рождение и развитие Вселенной. Електронний ресурс: http://ru-universe.livejournal.com/712762.html?page=1

176. Kelly K.F. The booms and busts of molecular electronics / K.F. Kelly, C.C.M. Mody // IEEE Spectrum. – 2015.– Vol. 52, Iss. 10. –P. 52-60.

177. Сет Л. Программируя Вселенную: Квантовый компьютер и будущее науки / Л. Сет. – М: Альпина нон-фикшн. – 2014.– 256 с.

178. Рябцев В.Г. Метод и средства визуализации алгоритмов тестов диагностирования запоминающих устройств / В.Г. Рябцев, Д.Н. Муамар // Электронное моделирование.– 2010. – Том 32, № 3. – С. 43-52.

179. Zorian Y. Test solutions for nanoscale Systems-on-Chip: Algorithms, methods and test infrastructure / Y. Zorian, S. Shoukourian // Nineth International Conference on Computer Science and Information Technologies Revised Selected Papers.– Yerevan.– 2013.– P. 1-3.

180. Tshagharyan G. Overview study on fault modeling and test methodology development for FinFET-based memories / G. Tshagharyan, G. Harutyunyan, S. Shoukourian, Y. Zorian // 2015 IEEE East-West Design & Test Symposium (EWDTS).– Batumi.– 2015.– P. 1-4.

181. Abramovici M. Digital systems testing and testable design / M. Abramovici, M.A. Breuer, A.D. Friedman. – Computer Science Press. – 1998. – 652 p.

182. Hadjitheophanous S. Scalable parallel fault simulation for shared-memory multiprocessor systems / S. Hadjitheophanous, S.N. Neophytou, M.K. Michael // 2016 IEEE 34th VLSI Test Symposium (VTS) .– Las Vegas, NV.– 2016.– P. 1-6.

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

184. Nielsen M.A. Quantum Computation and Quantum Information / M.A. Nielsen, I. L. Chuang.– Cambridge University Press.– 2010. – 676 p.

185. Nakahara M. Quantum Computing. An Overview / M. Nakahara. – Higashi-Osaka: Kinki University.– 2010. – 53 p.

186. Курош А.Г. Курс высшей алгебры / А.Г. Курош. – Москва: Наука. – 1968. – 426 с.

187. Проектирование и тестирование цифровых систем на кристаллах / В. И. Хаханов, Е. И. Литвинова, И. В. Хаханова, О. А. Гузь. – Харьков: ХНУРЭ, 2009. – 484 с.

188. Molnar L. Fault simulation methodes / L. Molnar, A. Gontean // 2016 12th IEEE International Symposium on Electronics and Telecommunications (ISETC). – Timisoara.– 2016.– P. 194-197.

189. Ubar R. Combinational fault simulation in sequential circuits / R. Ubar, J. Kõusaar, M. Gorev, S. Devadze // 2015 IEEE International Symposium on Circuits and Systems (ISCAS).– Lisbon.– 2015.– P. 2876-2879.

190. Gorev M. Fault simulation with parallel exact critical path tracing in multiple core environment / M. Gorev, R. Ubar, S. Devadze // 2015 Design, Automation & Test in Europe Conference & Exhibition (DATE).– Grenoble.– 2015. – P. 1180-1185.

191. Pomeranz I. Fault simulation with test switching for static test compaction / I. Pomeranz // 2014 IEEE 32nd VLSI Test Symposium (VTS).– Napa, CA.– 2014.– P. 1-6.

192. Mirkhani S. EAGLE: A regression model for fault coverage estimation using a simulation based metric / S. Mirkhani, J.A. Abraham // 2014 International Test Conference.– Seattle, WA.– 2014.– P. 1-10.

193. Hahanov V.I. Qubit Model for solving the coverage problem / V.I. Hahanov, E.I. Litvinova, S.V. Chumachenko et al // Proc. of IEEE East-West Design and Test Symposium. – Ukraine. – Sept., 2012. – P. 142 - 144.

194. Metodi T. Quantum Computing for Computer Architects / T. Metodi, F. Chong.– Synthesis Lectures on Computer Architecture.– Morgan & Claypool.– 2006.– 154 p.

195. Stig S. Quantum approach to informatics / S. Stig, Kalle-Antti Suominen. – John Wiley & Sons, Inc.– 2005.– 249 p.

196. Hahanov V.I. Qubit data structure of computing devices / V.I. Hahanov, W. Gharibi, E..I Litvinova, A.S. Shkil // Electronic modeling.– № 1.– 2015.– P.76-99.

197. Hahanov V. MQT-model for Virtual Computer Design / V. Hahanov, T. Bani Amer, I. Hahanov // Proc. of Microtechnology and Thermal Problems in Electronics (Microtherm).– 2015.– P. 182-185.

198. Zorian Y. Embedded-memory test and repair: infrastructure IP for SoC yield / Y. Zorian, S. Shoukourian // IEEE Design & Test of Computers. – Vol. 20, Issue 3. – P. 58 – 66.

199. Dugganapally I.P. Multi-level, Memory-Based Logic Using CMOS Technology / I.P. Dugganapally, S.E. Watkins, B. Cooper // 2014 IEEE Computer Society Annual Symposium on VLSI (ISVLSI) .– Tampa, FL.– P. 583-588 .

200. Yueh W. A Memory-Based Logic Block With Optimized-for-Read SRAM for Energy-Efficient Reconfigurable Computing Fabric / W. Yueh, S. Chatterjee, M. Zia, S. Bhunia, S. Mukhopadhyay // IEEE Transactions on Circuits and Systems II: Express Briefs.– vol. 62, iss. 6.– P. 593-597.

201. Matsunaga S. MTJ-based nonvolatile logic-in-memory circuit, future prospects and issues / S. Matsunaga, J. Hayakawa, S. Ikeda, K. Miura, T. Endoh, H. Ohno, T. Hanyu // Design, Automation & Test in Europe Conference & Exhibition, DATE '09.– 2009. – P. 433 - 435.

202. Harada S. Design of a Logic-in-Memory Multiple-Valued Reconfigurable VLSI Based on a Bit-Serial Packet Data Transfer Scheme / S. Harada, Xu Bai, M. Kameyama, Y. Fujioka // 2014 IEEE 44th International Symposium on Multiple-Valued Logic (ISMVL).– 2014.– P. 214 - 219.

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

204. Хаханов В.І., Обрізан В.І., Мірошниченко Я.В., Мельникова О.В. SIGETEST – система моделювання тестів перевірки несправностей цифрових пристроїв // Каталог анотацій на розробки за матеріалами першого українсько-китайського форуму «Наука-виробництво».– Харків, ХНУРЕ.– 2007.– С. 25-26.

205. Автоматизированное проектирование цифровых устройств / С.С.Бадулин, Ю.М.Барнаулов и др./ Под ред. С.С. Бадулина.– М.: Радио и связь.– 1981.– 240 с.

206. Melikyan V.Sh. A method of eliminating false paths during statistical static analysis of timing delays of digital circuits / V.Sh. Melikyan / Elektronica i svyaz. – 2009. – Vol. 2-3, no. 1.– P. 93-96.

207. Melikyan V.Sh. Interconnections model delays for the logic analysis of ECL circuits / V.Sh. Melikyan, A.O. Vatyan // SUAB. – Vol. 2. Computer Engineering. – Moscow.– 1997.– P. 187-194.

# ДОДАТОК А

## НАГОРОДИ, ДИПЛОМИ, СЕРТИФІКАТИ



SAE INTERNATIONAL

IN RECOGNITION OF

VLADIMIR HAHANOV

for a substantial contribution
as a Presenter for the
WCX™ 17: SAE World Congress Experience
April 4 - 6

H. Robert Welge
2017 SAE Engineering Meetings
Board Chair

Jim Forlenza
Engineering Meetings Board
Staff Representative

SAE INTERNATIONAL FOUNDED SEAL NEW YORK

P1771613



22ND IEEE EUROPEAN TEST SYMPOSIUM

22-26 May, 2017
Limassol, Cyprus

ETS 17

CERTIFICATE OF ATTENDANCE

This is to certify the distinguished participation of

VLADIMIR HAHANOV

at the IEEE European Test Symposium in May 2017.

MARIA K. MICHAEL
GENERAL CHAIR

HARALAMPOS G. STRATIGOPOULOS
PROGRAM CHAIR

НАУКОВИЙ ПАРК РАДІОЕЛЕКТРОНІКА ТА ІНФОРМАТИКА

*Конкурс*
*«Радіоелектроніка та інформатика 21 віку»*

# ДИПЛОМ

## за II місце

**Номінація**

***Геоінформаційні системи***

**Проект**

***Кіберсистема - "Хмарне керування транспортом"***
***(Traffic Cyber Physical Systems -***
***Smart Cloud Traffic Control)***

**Розробники**

Хаханов Володимир Іванович, д.т.н., проф.
Чумаченко Світлана Вікторівна, д.т.н., проф.
Литвинова Євгенія Іванівна, д.т.н., проф.
та інші

Генеральний директор — Прохоров В.П.

Секретар Експертної ради — Меркулова К.В.

22 липня 2015 р.



НАУКОВИЙ ПАРК РАДІОЕЛЕКТРОНІКА ТА ІНФОРМАТИКА

*Конкурс*
*«Радіоелектроніка та інформатика 21 віку»*

# ДИПЛОМ

## ПЕРЕМОЖЦЯ

**Номінація**

***Комп'ютерні технології***

**Проект**

***Віртуальний квантовий комп'ютер***

**Розробники**

Хаханов Володимир Іванович, д.т.н., проф.
Чумаченко Світлана Вікторівна, д.т.н., проф.
Литвинова Євгенія Іванівна, д.т.н., проф.
та інші

Генеральний директор Прохоров В.П.

Секретар Експертної ради Меркулова К.В.

22 липня 2015 р.





tttc

# Gerald W. Gordon Award

## for Student Volunteer Service

Presented to

*Anastasia Hahanova*

Kharkov National University of Radioelectronics

Chen-Huan Chiang, TTTC Chair

Oct 31, 2017

Date

IEEE

test technology technical council

IEEE computer society


CYBER
B2S DeSSerT S2B
FORUM
INTERNATIONAL SCIENTIFIC TECHNICAL CONFERENCE
10 YEARS
DeSSerT 2016 UA
DEPENDABLE SYSTEMS, SERVICES & TECHNOLOGIES
Xradiy
IEEE
ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
SDM ЦМСО
May 21 - 23, 2016
Ukraine, Chernivtsi
CERTIFICATE OF PARTICIPATION
presented to
Eugenia Litvinova
This is to acknowledge your presentation at International Conference Dependable Systems, Services & Technologies CYBER FORUM DESSERT B2S-S2B in Yuriy Fedkovych Chernivtsi National University, Chernivtsi, Ukraine.
We cordially thank you and look forward to further cooperation.
General Chair of DESSERT 2016
DrS, Professor
V. Kharchenko




May 21 - 23, 2016
Ukraine, Chernivtsi

# CERTIFICATE OF PARTICIPATION

presented to

**Svetlana Chemachenko**

This is to acknowledge your presentation at International Conference Dependable Systems, Services & Technologies CYBER FORUM DESSERT B2S-S2B in Yuriy Fedkovych Chernivtsi National University, Chernivtsi, Ukraine.

We cordially thank you and look forward to further cooperation.

**General Chair of DESSERT 2016**
**DrS, Professor** **V. Kharchenko**

УКРАЇНА

# ПАТЕНТ

## НА КОРИСНУ МОДЕЛЬ

## № 119896

**ПЕРЕТВОРЮВАЧ ДВІЙКОВО-К-ЗНАЧНОГО КОДУ В ДВІЙКОВИЙ КОД**

Видано відповідно до Закону України "Про охорону прав на винаходи і корисні моделі".

Зареєстровано в Державному реєстрі патентів України на корисні моделі **10.10.2017.**

Заступник міністра економічного розвитку і торгівлі України М.І. Тітарчук

МІНІСТЕРСТВО ЕКОНОМІЧНОГО РОЗВИТКУ І ТОРГІВЛІ УКРАЇНИ * ІДЕНТИФІКАЦІЙНИЙ КОД 37508596 *



Міністерство освіти і науки України
Одеський національний політехнічний університет

ДИПЛОМ

ВСЕУКРАЇНСЬКА
СТУДЕНТСЬКА ОЛІМПІАДА

2016 / 2017 навчального року
з дисципліни
«Технічна діагностика обчислювальних пристроїв та систем»
За III місце
в особистому заліку
нагороджується

Лук`янець Валентин Іванович

Голова оргкомітету
13 квітня 2017 р.

С.А. Нестеренко



Міністерство освіти і науки України
Одеський національний політехнічний університет

ДИПЛОМ

СІЛА
Ігор Володимирович
нагороджується
за II місце
II етапу Всеукраїнської студентської олімпіади
2016/ 2017 навчального року
з дисципліни «Технічна діагностика
обчислювальних пристроїв та систем»

Голова оргкомітету
13 квітня 2017 р.

С.А. Нестеренко



Міністерство освіти і науки України
Одеський національний політехнічний університет

ДИПЛОМ

ВСЕУКРАЇНСЬКА
СТУДЕНТСЬКА ОЛІМПІАДА

2016 / 2017 навчального року
з дисципліни
«Технічна діагностика обчислювальних пристроїв та систем»
За II місце
в особистому заліку
нагороджується

Сіла Ігор Володимирович

Голова оргкомітету
13 квітня 2017 р.

С.А. Нестеренко

Міністерство освіти і науки України
Одеський національний політехнічний університет

# ДИПЛОМ

## ВСЕУКРАЇНСЬКА СТУДЕНТСЬКА ОЛІМПІАДА

**2015 / 2016 навчального року**
**з дисципліни**
**«Технічна діагностика обчислювальних пристроїв та систем»**

**За II місце**
**в особистому заліку**
**нагороджується**

**Сімакін Владислав Андрійович**

**Голова оргкомітету** **С.А. Нестеренко**
**14 квітня 2016 р.**



Міністерство освіти і науки України
Одеський національний політехнічний університет

# ДИПЛОМ

## ВСЕУКРАЇНСЬКА СТУДЕНТСЬКА ОЛІМПІАДА

**2015 / 2016 навчального року**
**з дисципліни**
**«Технічна діагностика обчислювальних пристроїв та систем»**

**За II місце**
**в особистому заліку**
**нагороджується**

**Котляров Антон Сергійович**

**Голова оргкомітету** **С.А. Нестеренко**
**14 квітня 2016 р.**



Міністерство освіти і науки України
Одеський національний політехнічний університет

# ДИПЛОМ

## ВСЕУКРАЇНСЬКА СТУДЕНТСЬКА ОЛІМПІАДА

2015 / 2016 навчального року
з дисципліни
«Технічна діагностика обчислювальних пристроїв та систем»

**За II місце**

в особистому заліку
нагороджується

**Лук'янець Валентин Іванович**

Голова оргкомітету С.А. Нестеренко
14 квітня 2016 р.



Міністерство освіти і науки України
Одеський національний політехнічний університет

# ДИПЛОМ

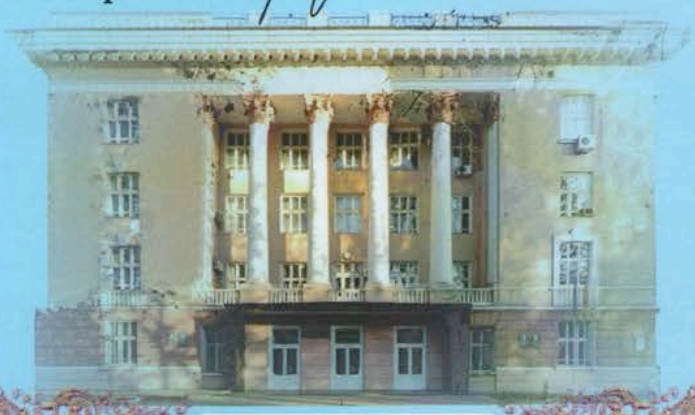
## ЛУК'ЯНЕЦЬ
## Валентин Іванович

**нагороджується**

**за II місце**

**II етапу Всеукраїнської студентської олімпіади**
**2015/ 2016 навчального року**
**з дисципліни «Технічна діагностика**
**обчислювальних пристроїв та систем»**

**Голова оргкомітету** **С.А. Нестеренко**
**14 квітня 2016 р.**



Міністерство освіти і науки України
Одеський національний політехнічний університет

# ДИПЛОМ

## ВСЕУКРАЇНСЬКА СТУДЕНТСЬКА ОЛІМПІАДА

**2015 / 2016 навчального року**
**з дисципліни**
**«Технічна діагностика обчислювальних пристроїв та систем»**

**За II місце**
**в особистому заліку**
**нагороджується**

**Сіла Ігор Володимирович**

**Голова оргкомітету** **С.А. Нестеренко**
**14 квітня 2016 р.**



Міністерство освіти і науки України
Одеський національний політехнічний університет

# ДИПЛОМ

## ВСЕУКРАЇНСЬКА СТУДЕНТСЬКА ОЛІМПІАДА

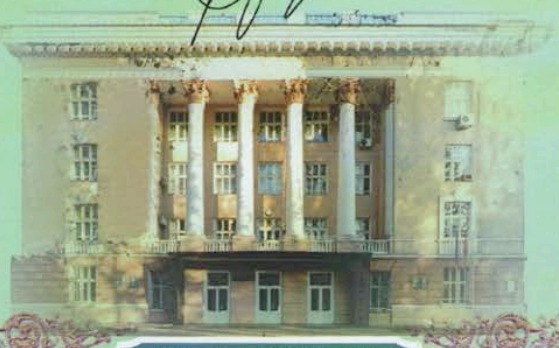
**2015 / 2016 навчального року**
**з дисципліни**
**«Технічна діагностика обчислювальних пристроїв та систем»**
**нагороджується команда**
**Харківського національного університету радіоелектроніки**

**за II місце**

**у складі:**
**Лук'янець Валентин Іванович**
**Сімакін Владислав Андрійович**
**Бояджян Армен Гамлетович**
**Котляров Антон Сергійович**
**Сіла Ігор Володимирович**

**Голова оргкомітету** **С.А. Нестеренко**

**14 квітня 2016 р.**



# Міністерство освіти і науки України

## Вінницький національний технічний університет

**Дипломом переможця II ступеня нагороджується**

**Хаханов Іван Володимирович**

**Сергієнко Владислав Ігорович**




**Харківський національний університет радіоелектроніки**

Всеукраїнський конкурс
студентських наукових робіт з напрямку
“Інформатика, обчислювальна техніка та автоматизація”

Вінниця 24 березня 2016 р.

Голова конкурсної комісії,
проректор з наукової роботи ВНТУ, д.т.н., проф. С. В. Павлов

Tempus
This project is funded by the European Union

Curricula Development for New Specialization:
Master of Engineering in Microsystems Design

530785-TEMPUS-1-2012-PL-TEMPUS-JPCR

# CERTIFICATE OF TRAINING

This certifies that

**Prof. Lina Larchenko**

from

**Kharkiv National University of Radioelectronics**

has successfully completed

**professional staff training, 20.09.2016 - 02.10.2016**

organized in the frame of Tempus project *"Curricula Development for New Specialization: Master of Engineering in Microsystems Design" 530785-TEMPUS-1-2012-PL-TEMPUS-JPCR*

at Department of Semiconductor and Optoelectronics Devices,

Lodz University of Technology

Tempus Project Coordin...
Project No. 530785
Prof. [illegible] Ph.D. D.Sc.

Lodz 07.10.2016

Lodz University of Technology




This project is funded by the European Union

Curricula Development for New Specialization:
Master of Engineering in Microsystems Design

530785-TEMPUS-1-2012-PL-TEMPUS-JPCR

# *CERTIFICATE OF TRAINING*

This certifies that

**dr Artur Ziarmand**

from

**Kharkiv National University of Radioelectronics**

has successfully completed

**professional staff training, 10.10.2016 - 14.10.2016**

organized in the frame of Tempus project *"Curricula Development for New Specialization: Master of Engineering in Microsystems Design" 530785-TEMPUS-1-2012-PL-TEMPUS-JPCR*

at Department of Semiconductor and Optoelectronics Devices,

Lodz University of Technology

Lodz 14.10.2016

Tempus Project Coordinator
Prof. Zbigniew [illegible], PhD, D.Sc.



Lodz University of Technology

VIVACELL MTS

# Certificate of Participation

ՄԱՍՆԱԿՑՈՒԹՅԱՆ ՎԿԱՅԱԿԱՆ

*presented to*

շնորհվում է

## Kostiuk Serhii

October
Հոկտեմբերի 24, 2017

Ukraine
Ուկրաինա

TWELVE ANNUAL INTERNATIONAL MICROELECTRONICS OLYMPIAD OF ARMENIA

ՄԻԿՐՈԷԼԵԿՏՐՈՆԻԿԱՅԻ ՀԱՅԱՍՏԱՆՅԱՆ ՏԱՍՆԵՐԿՈՒԵՐՈՐԴ ԱՄԵՆԱՅՆ ՄԻՋԱԶԳԱՅԻՆ ՕԼԻՄՊԻԱԴԱ

**Armen Baldryan**
**Արմեն Բալդրյան**
President of Organizing Committee
Կազմկոմիտեի նախագահ

**Sci. Dr. Vazgen Melikyan**
**Տ.գ.դ.Վազգեն Մելիքյան**
President of Program Committee
Ծրագրային կոմիտեի նախագահ

**РІШЕННЯ**
Вченої ради
Харківського національного університету радіоелектроніки

28 жовтня 2016 року м. Харків № 13/9-6

**Про відкриття нової спеціалізації «Проектування вбудованих мікросистем» за спеціальністю 123 «Комп'ютерна інженерія» для підготовки фахівців освітнього рівня «магістр», затвердження освітньо-наукової програми і навчального плану**

Заслухавши та обговоривши доповідь декана факультету КІУ Хаханова Володимира Івановича щодо відкриття нової спеціалізації «Проектування вбудованих мікросистем» за спеціальністю 123 «Комп'ютерна інженерія» для підготовки фахівців освітнього рівня «магістр» у Харківському національному університеті радіоелектроніки на кафедрі АПОТ, затвердження освітньо-наукової програми і навчального плану, Вчена рада

**ПОСТАНОВИЛА:**

1. Інформацію декана факультету КІУ Хаханова Володимира Івановича щодо відкриття нової спеціалізації «Проектування вбудованих мікросистем» за спеціальністю 123 «Комп'ютерна інженерія» для підготовки фахівців освітнього рівня «магістр» на кафедрі АПОТ, затвердження освітньо-наукової програми і навчального плану взяти до відома.
2. Затвердити відкриття нової спеціалізації «Проектування вбудованих мікросистем» спеціальності 123 «Комп'ютерна інженерія» для підготовки фахівців овітнього рівня «магістр».
3. Затвердити освітньо-наукову програму та навчальний план підготовки магістрів за спеціалізацією «Проектування вбудованих мікросистем» спеціальності 123 «Комп'ютерна інженерія».
4. Контроль за виконанням рішення покласти на декана факультету КІУ Хаханова В.І.

**Голова Вченої ради** **Е.Ю. Рубін**

**Учений секретар** **І.В. Магдаліна**

# ДОДАТОК В

ДОКУМЕНТИ, ЩО ПІДТВЕРДЖУЮТЬ ВПРОВАДЖЕННЯ РЕЗУЛЬТАТІВ

«ЗАТВЕРДЖУЮ»
Проректор з НМР ХНУРЕ
_____ Рубан І.В.
"08" 05 2017 р.

**АКТ**

про впровадження у навчальний процес ХНУРЕ результатів дисертаційної роботи Зіарманда Артура Нісаровича «Моделі і методи кіберфізичного комп'ютингу для цифрового моніторингу та хмарного управління транспортом», представленої 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 Шкіля О.С., проф. каф. АПОТ Литвинова Є.І. розглянула матеріали дисертаційної роботи Зіарманда А.Н., які використовуються в навчальному процесі кафедри АПОТ ХНУРЕ у 2016/2017 навчальному році, і прийшла до наступного висновку.

Розроблені у дисертаційній роботі методи і моделі киберфізичної інфраструктури, а саме:

– удосконалений алгоритм Дейкстри, який відрізняється можливістю аналізу кіберфізичної інфраструктури дорожнього руху для online пошуку квазіоптимального маршруту транспортного засобу в умовах виникнення колізій;

– удосконалені метрика і критерії оцінювання якості інфраструктури, яка відрізняється можливістю online аналізу кіберфізичного простору для пошуку квазі-оптимального маршруту і зменшення часу його виконання;

впроваджені у навчальний процес кафедри АПОТ ХНУРЕ та використовуються у таких навчальних дисциплінах:

1. У навчальній дисципліні «Дискретна математика» для бакалаврів напрямку «Комп'ютерна інженерія» у лекційному матеріалі та у практичних заняттях за темою «Алгоритм Дейкстри».

2. У навчальній дисципліні «Cloud-Fog кіберфізичні системи» для магістрантів спеціалізацій «Спеціалізовані комп'ютерні системи», «Проектування вбудованих мікросистем» у лекційному матеріалі за темами «Аналіз кіберфізичного простору», «Internet of Things. Основні поняття IoT як складової кіберфізичних систем. Інтернет речей і розумні сенсори. Фізичний Binary Computing», «Computing sustainable development matrix. Метрика для кіберфізичного комп'ютингу. Структури секвенсорів для прийняття рішень», «Трикутна топологія кіберпростору. Оцінка якості кіберфізичної топології», а також у лабораторному практикумі за темою «Синтез логічної схеми управління світлофором».

Зав. каф. АПОТ проф. Чумаченко С.В.,

Доц. каф. АПОТ Шкіль О.С.,

Проф. каф. АПОТ Литвинова Є.І.

Institute of Reliable Software

**ТОВ "Перший інститут надійного програмного забезпечення"**

# ДОВІДКА

## про можливість впровадження методів та моделей кіберфізичної транспортної хмарної інфраструктури

Розроблена на кафедрі Автоматизації проектування обчислювальної техніки Харківського національного університету радіоелектроніки за участю (30%) здобувача Зіарманда Артура Нісаровіча архітектура розумного хмарного світлофора характеризується використанням логічних операцій і часом простою зеленого сигналу, що дає можливість на 10-30% зменшити час простою зеленого світлофора і істотно збільшити пропускну здатність транспортних потоків на перехресті доріг. Впровадження розумних світлофорів в масштабах міста, дозволить скоротити інтегральний час знаходження транспорту на маршруті і витрати на енергоносії не менше, ніж на 10%.

У рамках практичної реалізації моделей і методів цифрового моніторингу і хмарного керування автомобілем з метою створення і верифікації програмних компонентів кіберфізичної архітектури дорожнього руху «Cloud Traffic Control» виконано тестування модельних потоків транспортних засобів на ділянках дорожньої інфраструктури. Достовірність результатів підтверджується проведеними експериментами, тестуванням і верифікацією запропонованих моделей і методів моніторингу та управління дорожнім рухом.

Вперше запропоновано модель транспортного комп'ютингу, яка характеризується кіберфізичною взаємодією автомобіля з хмарним сервісом за допомогою еволюційного переміщення світлофора з фізичного у віртуальний простір для цифрового моніторингу транспортних потоків і квазіоптимального управління дорожнім рухом.

Розроблені методи і моделі киберфізичної інфраструктури:

– удосконалений алгоритм Дейкстри, який відрізняється можливістю аналізу кіберфізичної інфраструктури дорожнього руху для online пошуку квазіоптимального маршруту транспортного засобу в умовах виникнення колізій;

– удосконалені метрика і критерії оцінювання якості інфраструктури, яка відрізняється можливістю online аналізу кіберфізичного простору для пошуку квазі-оптимального маршруту і зменшення часу його виконання;

можуть бути імплементовані як компонент проекту при синтезі хмарного сервісу.

28.09.2017 Директор _____ /Чертков П.М./

Голова комісії: _____ /Чертков П.М./

«ЗАТВЕРДЖУЮ»
Перший проректор ХНУРЕ
_____ Ключник І.І.
"09" 03 2016 р.

**С П Р А В К А**

про впровадження в навчальний процес ХНУРЕ результатів дисертаційної роботи Обрізана Володимира Ігоровича «Мультиверсний паралельний синтез цифрових структур на основі SystemC специфікації», представленої 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 Литвинова Є.І. розглянула матеріали дисертаційної роботи Обрізана В.І., які використовуються в навчальному процесі кафедри АПОТ ХНУРЕ у 2015/2016 навчальному році, і прийшла до наступного висновку.

Розроблений у дисертаційній роботі мультиверсний метод проектування у HDL-моделях цифрових пристроїв дозволяє суттєво скороти час на синтез HDL-моделей, особливо в умовах системної специфікації. Зазначений метод використовується в системах верифікації HDL-моделей при автоматизованому проектуванні цифрових пристроїв.

У навчальному процесі кафедри АПОТ ХНУРЕ результати дисертаційної роботи Обрізана В.І. використовуються у таких навчальних дисциплінах:

1. У навчальній дисципліні «Логічне моделювання» для бакалаврів напрямку «Комп'ютерна інженерія» у лекційному матеріалі по темі «Верифікація моделей цифрових пристроїв» та у лабораторних роботах по темі «Побудова мовних моделей цифрових пристроїв».
2. У навчальній дисципліні «Проектування та тестування цифрових систем на ПЛІС» для бакалаврів напрямку «Комп'ютерна інженерія» у лекційному матеріалі та практичних заняттях по темі «Сучасні методи проектування цифрових систем на ПЛІС».

12.05.2016

Зав. каф. АПОТ проф. Чумаченко С.В.,
Доц. каф. АПОТ Шкиль О.С.,
Проф. каф. АПОТ Литвинова Є.І.

ЗАТВЕРДЖУЮ»
Перший проректор ХНУРЕ
_____ Ключник І.І.
"06" 03 2016 р.

**СПРАВКА**

про впровадження в навчальний процес ХНУРЕ результатів дисертаційної роботи Тамер Бані-Амер (Tamer Bani-Amer) «Хмарний сервіс-комп'ютинг для тестування і моделювання SoC-компонентів», представленої на здобуття наукового ступеня кандидата технічних наук за спеціальністю 05.13.05 – комп'ютерні системи та компоненти

Голова комісії: Литвинова Є.І.

Члени комісії: Чумаченко С.В, Шкіль О.С.

В період з 05.10.2016 по 10.10.2016 комісія провела роботу з визначення впровадження в навчальний процес хмарних компьютингових сервісів тестування і верифікації цифрових пристроїв, розроблених на кафедрі АПОТ ХНУРЕ при 30% участі здобувача Тамера Бані Амера.

Комісією встановлено, що розроблені моделі і методи кубітного тестування, моделювання та оцінки якості вхідних наборів реалізовані у вигляді хмарних мікросервісов, придатних для їх використання в навчальних цілях для курсів: «Проектування та тестування цифрових систем на ПЛІС» для бакалаврів напрямку «комп'ютерна інженерія».

Основними результатами, впровадженими в навчальний процес слід вважати: векторні структури даних і кубітні методи синтезу, тестування і моделювання, інтегровані в хмарну інфраструктуру сервісного обслуговування компонентів цифрових систем на кристалах з метою підвищення якості виробів і виходу придатної продукції за рахунок адресованих обчислювальних процесів і явищ. Основна інноваційна ідея запропонованої МАТ-моделі обчислень полягає в синтезі і аналізі векторних цифрових структур на основі адресованих елементів пам'яті, що виключають використання reusable or new logic. Приведення опису функції і структури до єдиного одномерного формату, означає технологічно вирішувати всі завдання синтезу та аналізу для функціональностей і графів у кубітно-векторній метриці, що створює memory-driven computing на основі виконання паралельних логічних операцій.

Хмарні компьютингові сервіси тестування, моделювання несправностей і справної поведінки цифрових пристроїв є працездатними, перевіреними на представницькій вибірці стандартних тестових схем з бібліотеки конференції ISCAS.

Комісія рекомендує використання нової кубітно-векторної технології і хмарних сервісів проектування, синтезу тестів і моделювання цифрових пристроїв в навчальному процесі для студентів факультету КІУ.

Голова комісії проф. каф. АПОТ Литвинова Є.І.
Члени комісії зав. каф. АПОТ, проф. Чумаченко С.В.,
доц. каф. АПОТ Шкиль О.С.

ALDEC
The Design Verification Company

**ООО «Алдек-КТС»**
ул. Космическая 23а
61145, г.Харьков, Украина

ЕГРПОУ 36374067
Р/с 26008154763 в ВАТ ХОД «Райффайзен Банк Аваль»,
МФО 380805
Тел.: (+38 057) 760-47-25

# СПРАВКА

о внедрении метода мультиверсного проектирования HDL-моделей цифровых устройств в научно-исследовательскую и производственную деятельность ООО «Алдек-КТС»

Мультиверсный метод проектирования HDL-моделей цифровых устройств, представленных на языках описания аппаратуры, который разработан на кафедре Автоматизации проектирования вычислительной техники Харьковского национального университета радиоэлектроники при участии (30%) соискателя Обризана Владимира Игоревича, позволяет снизить трудоемкость синтеза HDL-моделей цифровых устройств функционально-блочного уровня при автоматизированном проектировании.

Разработанная методика проведения диагностического эксперимента при верификации HDL-кода позволила автоматически применять различные алгоритмы поиска дефектов в зависимости от формы спецификации и стиля написания HDL-модели.

Предложенный метод мультиверсного синтеза структуры цифрового изделия на основе заданной спецификации в среде SystemC (C++) и автоматическом подборе функциональных компонентов за счет параллельного синтеза и верификации архитектурных решений системного уровня в соответствии с предложенной метрикой существенно уменьшает время проектирования вычислительных архитектур и повышает качество цифровых изделий.

Разработанная на основе результатов исследования методика проведения диагностического эксперимента применяется в работе ООО “Алдек-КТС” при автоматизированном проектированбиии цифровых устройств с использованием САПР Active-HDL и системы имитационного моделирования Riviera™ компании Aldec Inc. (USA), что позволило на 20-30% процентов сократить время на поиск ошибок проектирования.

Директор ООО «Алдек-КТС», к.т.н.: _____ / Зайченко С.О. /

15.04.2016г.

ALDEC
The Design Verification Company

ООО «Алдек-КТС»
ул. Космическая 23а
61145, г.Харьков, Украина

ЕГРПОУ 36374067
Р/с 26008154763 в ВАТ ХОД «Райффайзен Банк Аваль»,
МФО 380805
Тел.: (+38 057) 760-47-25

**СПРАВКА**

о внедрении облачных компьютинговых сервисов тестирования и верификации цифровых устройств в методологическое и технологическое обеспечение компании ООО «АЛДЕК-КТС» в целях повышения качества проектирования цифровых изделий.

Внедрение в научно-исследовательскую и производственную деятельность моделей и методов кубитного синтеза и анализа цифровых устройств, облачных компьютинговых сервисов тестирования и верификации, разработанных на кафедре АПВТ, ХНУРЭ при 30 процентном участии соискателя Тамера Бани Амера позволило на 12 процентов уменьшить общее время разработки, необходимое для создания и выхода годной продукции на рынок.

Установлено, что разработанные модели и методы кубитного тестирования, моделирования и оценки качества входных наборов, реализованные в виде облачных микросервисов, потенциально позволяют на 5-6 процентов повысить выход годной продукции за счет online восстановления работоспособности memory-driven логических схем. Основными результатами, внедренными в производственный процесс следует считать: векторные структуры данных и кубитные методы синтеза, тестирования и моделирования, интегрированные в облачную инфраструктуру сервисного обслуживания компонентов цифровых систем на кристаллах в целях повышения качества изделий и выхода годной продукции за счет адресуемости всех вычислительных процессов и явлений.

Основная инновационная идея предложенной «Memory-Address-Transaction» модели вычислений заключается в синтезе и анализе векторных цифровых структур на основе адресуемых элементов памяти, исключающих использование reusable or new logic.

Приведение описания функции и структуры к единому одномерному формату, означает – технологично решать все задачи синтеза и анализа для функциональностей и графов в кубитно-векторной метрике, создающей memory-driven computing на основе выполнения параллельных логических операций.

Облачные компьютинговые сервисы тестирования, моделирования неисправностей и исправного поведения цифровых устройств являются работоспособными, проверенными на представительной выборке стандартных тестовых схем из бибилиотек конференции ISCAS.

Разработанные модели и методы кубитно-векторной технологии и облачных сервисов проектирования, синтеза тестов и моделирования цифровых устройств интегрированы в состав методологического обеспечения среды моделирования Riviera компании Aldec Inc. и используются в процессе проектирования цифровых систем на кристаллах, имплементируемых в блоки FPGA, CPLD.

Директор ООО «Алдек-КТС», к.т.н. _____ / Зайченко С.О. /

25.11.2016г.