
КОМПЬЮТЕРНАЯ ИНЖЕНЕРИЯ

УДК 658:512.011; 681.326; 519.713

КВАНТОВЫЕ МОДЕЛИ И ОБЛАЧНЫЕ СЕРВИСЫ ДЛЯ АНАЛИЗА И ДИАГНОСТИРОВАНИЯ ЛОГИЧЕСКИХ СХЕМ

ЕМЕЛЬЯНОВ И.В., ЛЮБАРСКИЙ М.М.,
ХАХАНОВ В.И.

Предлагаются логические структуры киберфизического компьютеринга, которые рассматриваются как компоненты облачных технологий для точного мониторинга и метрического управления объектами. Дается аналитический обзор киберфизических технологий, задекларированных в Gartner's Hype Cycle 2017, а также некоторые разъяснения, связанные с их применением в науке, образовании, транспорте, индустрии и государственных структурах. Описывается кубитный метод диагностирования неисправностей и минимизации логических схем, которые отличаются от аналогов параллельным выполнением логических операций. Предлагаются кубитные структуры данных, облачная реализация методов анализа, тестирования и моделирования цифровых систем. Даются рекомендации к использованию топ 10 компонентов суперцикла 2017 в бизнесе и научно-образовательном процессе университетов. Представляется memory-driven архитектура квантового компьютеринга, которая характеризуется использованием фотонных транзакций на структуре электронов при отсутствии логики, связанной с суперпозицией и перепутыванием состояний.

Ключевые слова: hype cycle emerging technologies, digital world, cyber-physical computing, machine learning, cloud-driven computing, quantum computing, qubit data structures.

1. Введение

Целью исследования является обоснование и разработка архитектуры квантового компьютеринга на основе использования кубитных структур данных для параллельного решения задач синтеза и анализа цифровых устройств.

Задачи: 1) Анализ современных киберфизических трендов развития цифровых технологий управления физическими и социальными процессами, представленными компанией Gartner. 2) Модернизация архитектуры квантового memory-driven компьютеринга без операций суперпозиции и перепутывания. 3) Метод трансформирования табличных моделей цифровых компонентов в кубитные структуры данных. 4) Методы квантовой минимизации, облачного анализа и диагно-

стирования булевых функций путем унитарного кодирования входных состояний.

Киберфизический компьютеринг представляет собой теорию и практику точного облачного управления виртуальными, социальными и физическими процессами и явлениями на основе использования больших данных, метрического онлайн-мониторинга цифрового состояния процесса или явления в целях улучшения качества жизни человека и сохранения экологии планеты. Существуют определенные тенденции в мире, которые создают технологическую основу для создания киберсоциального компьютеринга, как части киберфизического компьютеринга, в рамках технологического уклада Internet of Things. Компания Gartner Inc., предсказывающая технологическую кибермоду, в 2017 году добавила восемь новых трендов в свой бренд Hype Emerging Technologies Cycle: 5G, Artificial General Intelligence, Deep Learning, Deep Reinforcement Learning, Digital Twin, Edge Computing, Serverless PaaS and Cognitive Computing [1,2].

Edge computing является технологией повышения производительности облачных сервисов путем выполнения локальных вычислений по месту мобильного пользователя. Digital Twin создает киберобразы физических процессов и явлений. Как в зеркале, если нет отображения оцифрованной компании (университета) в киберпространстве, то ее нет в физическом пространстве. Serverless PaaS – бессерверная архитектура для организации вычислительных процессов на основе платформы как услуги (Platform as a Service). Экономика облачной платформы является явно выигрышной по сравнению с серверной поддержкой деятельности компаний. Поэтому весь малый и средний бизнес в течение двух лет перейдет на облачные инфраструктуры и сервисы. Как следует понимать фазы Gartner цикла? 1) Innovation Trigger – запуск инновации, где потенциально интересные для рынка прорывные технологии с еще недоказанной коммерческой состоятельностью идут на смену существующим киберфизическим укладам. 2) Peak of Inflated Expectations – пик раздутых рыночных ожиданий, где своевременная реклама создает успешные прецеденты создания инновационных технологий на фоне множества неудач. 3) Trough of Disillusionment – приход разочарования, когда интерес к технологиям угасает, эксперименты не подтверждают ожидаемую рыночную привлека-

тельность, отдельные разработчики улучшают свою продукцию и получают инвестиции. 4) Slope of Enlightenment – склон прозрения, когда появляются примеры технологий, приносящие пользу предприятию, находятся финансы для пилотных проектов. 5) Plateau of Productivity: плоскость устойчивого повышения производительности, когда создаваемые технологии, товары и услуги находят своего потребителя на международном рынке.

2. Три главных направления киберкультуры

Hype-cycle 2017 формирует киберкультуру планеты на последующие 5-10+ лет путем экспертного анализа более, чем 1800 возможных технологий, выполняемого ведущими исследовательскими и консалтинговыми компаниями.

Первые три места в Gartner-топ-цикле закреплены за следующими стратегическими направлениями: Artificial Intelligence Everywhere, Transparently Immersive Experiences и Digital Platforms.

1) *Artificial Intelligence Everywhere*. Искусственный интеллект становится самой дизрапторной технологией в последующие 10 лет благодаря наличию вычислительных мощностей, бесконечных объемов больших данных и достижений в реализации нейронных сетей для адаптации к новым ситуациям, с которыми никто и никогда не сталкивался ранее. Предприятия, которые заинтересованы в использовании искусственного интеллекта, считают полезными для себя следующие технологии: Deep Learning, Deep Reinforcement Learning, Artificial General Intelligence, Autonomous Vehicles, Cognitive Computing, Commercial UAVs (Drones), Conversational User Interfaces, Enterprise Taxonomy and Ontology Management, Machine Learning, Smart Dust, Smart Robots and Smart Workspace.

Таким образом, Artificial General Intelligence в последующие 10 лет будет проникать во все сферы человеческой деятельности, как технологическая услуга, погруженная в киберфизическое пространство, включая 30 процентов высокотехнологичных и транспортных компаний.

Smart Workspace – умное рабочее место означает быть подключенным к инфраструктуре решения производственных проблем в пространстве и времени по формату 24/7. При этом используются виртуальные частные сети, метрика измерения потенциала и результатов деятельности, наличие определенной киберкультуры и выбор наиболее удобных мест для ведения бизнеса. Высокая самомотивация к успешному и результативному выполнению задания обуславливает использование динамически изменяющегося киберфизического рабочего пространства для творчества, инвариантного к офису, дому, транспорту, местам отдыха и спорта.

2) *Transparently Immersive Experiences*. Технологии опыта прозрачного погружения становятся все более ориентированными на человека и обеспечивают: 1) прозрачность отношений между людьми, бизнесом и вещами; 2) гибкость и адаптивность связей между рабочим местом, домом, предприятием и другими людьми. Gartner-Inc. также предсказывает внедрение в практику следующих, ожидаемых всеми, критических технологий: Autonomous Vehicles, Brain-Computer Interfaces, Smart Dust, 4D Printing, Augmented Reality (AR), Connected Home, Human Augmentation, Nanotube Electronics, Virtual Reality (VR), and Volumetric Displays. Интеграция кибер-технологий направлена на обеспечение качества жизни человека путем создания: smart workspace, connected home, augmented reality, virtual reality and the growing brain-computer interface. Напри-мер, Human Augmentation технология направлена на расширение или дополнение человеческих возможностей в целях улучшения здоровья и качества жизни за счет гармоничного использования когнитивных и биотехнических улучшений, как частей человеческого тела. Volumetric Displays, как объемные дисплеи, визуализируют объекты с помощью 3D активных элементов-вокселей (voxels) в трех измерениях со сферическим углом обзора в 360 градусов, где изображение явления изменяется при перемещении зрителя. Технология 4D Printing является инновацией 3D-печати, где конструктивные материалы могут трансформироваться после производства изделия в целях адаптации продукта к потребностям человека и к окружающей среде.

3) *Digital Platforms*. Ключевые платформы технологической культуры формируются компонентами: 5G, Digital Twin, Edge Computing, Blockchain, IoT, Neuromorphic Hardware, Quantum Computing, Serverless PaaS и Software-Defined Security. Такие технологии, как Quantum Computing и Blockchain, будут создавать самые непредсказуемые и дизрапторные прорывы для человека в ближайшие 5-10 лет. Neuromorphic Hardware рассматривается как будущее искусственного интеллекта, которое направлено на создание нейроморфного компьютерного чипа, способного заменить облачные вычислительные мощности Apple Siri Data Center при решении сложных задач machine learning (Chris Eliasmith, a theoretical neuroscientist and co-CEO of Canadian AI startup Applied Brain Research) [3]. Иначе, внутри iPhone появится цифровой мозг в форме нейроморфного IP-core, оперативно и на месте решающий все задачи взаимодействия гаджета с внешним миром в реальном времени. Нейроморфный универсальный чип IBM, благодаря

спайковому асинхронизму, потребляет на три порядка меньше энергии при количестве транзисторов, превышающих в пять раз существующие аппаратные решения компании Intel. Для программирования аппаратно-ориентированных алгоритмов используются компиляторы: Nengo, Python. Путем использования компилятора Nengo сегодня уже реализованы цифровые системы на кристаллах: vision systems, speech systems, motion control, adaptive robotic controllers, а также Sprau-chip для автономного интерактивного общения компьютера с окружающей средой. Software-Defined Security (SDS) или Catbird предназначена для защиты системных объектов или логических структур в виртуальном пространстве. Это связано с тем, что сетевая безопасность уже не имеет физических границ в рамках существования логической архитектуры облачных сервисов. Поэтому создается точная и гибкая SDS в виде дополнения к инфраструктурам и центрам обработки данных без наличия специализированных аппаратных устройств защиты. Масштабирование SDS дает возможность создавать или приобретать минимально необходимые условия безопасности в определенном месте и времени, что существенно уменьшает материальные затраты на формирование качественного SDS сервиса.

3. Практика использования топ-технологий

Высокие затраты на исследования и разработки от Amazon, Apple, Baidu, Google, IBM, Microsoft и Facebook стимулируют создание оригинальных патентуемых решений в области Deep Learning и Machine Learning, среди которых следует отметить: Amazon Alexa, Apple Siri, Google Now, Microsoft Cortana. Компания Gartner Inc. уверена, что инструменты для глубокого обучения будут составлять 80% стандартных средств для ученых к 2018 году. Сегодня уже на сайтах компаний становятся доступными технологии и данные о научных исследованиях: Amazon Machine Learning, Apple Machine Learning Journal, Baidu Research, Google Research, IBM AI и Cognitive Computing, Facebook Research.

Внедрение 5G-технологии телекоммуникаций в ближайшее десятилетие предоставит рынку ожидаемые инновационные решения по безопасности, масштабируемости и производительности глобальных сетей и соединений в транспорте, IoT, индустрии, здравоохранении.

Gartner Inc. прогнозирует, что к 2020 году 3% сетевых провайдеров услуг мобильной связи запустят коммерческие сети в 5G-формате, что обеспечит качественно новые условия повсеместного внедрения телекоммуникаций для масштабируемой глобализации сервисов: IoT, cloud-transport control, UHD-телевидения. лиде-

рами 5G-внедрения в 2017-2018 году выступают: AT&T, NTT Docomo, Sprint USA, Telstra, T-Mobile и Verizon. Технология 5G представляет собой ультраширокополосную мобильную связь в миллиметровом диапазоне для Massive M2M транзакций в реальном времени с допустимыми для управления задержками (1мс), при одновременном подключении порядка 10 млн устройств на 1 кв км. 5G использует технологию множественного доступа с разделением луча (Beam Division Multiple Access – BDMA) для взаимодействия базовой станции с мобильными устройствами. Беспроводная сотовая архитектура 5G обеспечивает пропускную способность 10-50 Гбит/с в миллиметровом диапазоне частот 30-300 ГГц для приложений UHD видео и создания виртуальной реальности [4]. Инновационная технология 5G характеризуется использованием: массива приемо-передающих антенн Massive MIMO, сети Cognitive Radio, организацией непосредственной связи D2D для IoT, созданием сети радиодоступа, как облачной услуги (radio access network as a service) и облака виртуальных сетевых функций (network function virtualization cloud – NFV).

Green IoT – киберфизическая культура человеческой деятельности, направленная на обеспечение качества жизни людей и сохранение экологии планеты, энергии, ресурсов и времени. Компонентами IoT являются: Identification, Sensing, Controlling, Communication, Computation, Services Intelligent, Digital Infrastructure. Умный мир (smart world) предоставляет каждому человеку сервисы от: умных устройств (watches, mobile phones, computers), умного транспорта (aircrafts, cars, buses, trains), умной инфраструктуры (homes, hospitals, offices, factories, cities, states), умного образования (school, university).

Компания Gartner Inc. посчитала необходимым вывести из-под зонтика актуальной рыночной моды следующие технологии, как не оправдавшие ожидания IT-бизнеса: Affective Computing, Micro Data Centers, Natural-Language Question Answering, Personal Analytics, Smart Data Discovery and Virtual Personal Assistants.

Для создания успешных бизнесов и новых образовательных курсов компания Gartner Inc. рекомендует учитывать свои предположения о стратегическом планировании, которые включают 10 пунктов: 1) К 2020 году 100 миллионов потребителей будут делать покупки в расширенной реальности, в том числе с использованием Head-Mounted Displays (HMDs). 2) К 2020 году 30% сеансов просмотра веб-страниц будут выполняться без использования экрана. Более 5 из 550 миллионов обладателей Apple iPhone будут использовать AirPods для обмена голосовыми со-

общениями. Пять процентов веб-сайтов, ориентированных на потребителя, будут оснащены аудиоинтерфейсами (включая голосовые чаты с поддержкой голоса). 3) К 2019 году 20% брендов откажутся от своих мобильных приложений (в пользу MASA – Mesh App and Service Architecture). 4) К 2020 году умные алгоритмы положительно повлияют на поведение более 1 миллиарда глобальных работников. 5) К 2022 году бизнес на основе использования блокчейнов будет стоить 10 миллиардов долларов. 6) К 2021 году 20% всех видов деятельности человека будут включены, по крайней мере, в сервисы одной из семи ведущих глобальных компаний (Google, Apple, Facebook, Amazon, Baidu, Alibaba и Tencent). 7) До 2019 года каждый доллар, инвестированный в инновации, потребует дополнительных 7 долларов для основного исполнения проекта. 8) В течение 2020 года Internet of Things (IoT) на 3% увеличит спрос, связанный с data centers. Комнатные экранные устройства, такие как Amazon Echo и Google Home, будут находиться в более, чем 10 миллионах домов. 9) К 2022 году IoT и облака (Google, Amazon, Microsoft) будут экономить потребителям и предприятиям 1 триллион долларов в год, ориентированных на услуги и расходные материалы. К 2020 году около 40 миллионов автомобилей будут использовать Android Auto, а 37 миллионов транспортных средств будут использовать CarPlay. 10) К 2020 году 40% сотрудников смогут сократить свои расходы на здравоохранение, используя фитнес-трекер.

4. Инновации для архитектуры квантового компьютеринга

Физическая основа классического квантового компьютеринга (рис. 1) заключается в использовании операций суперпозиции и перепутывания над состояниями электронов (p), которых вполне достаточно для организации вычислительного процесса [11-13]. Электрон выполняет функцию памяти для хранения бита информации. Транзакции между электронами осуществляются с помощью квантов или фотонов (q). Низкая и высокая орбиты электрона соответствуют значениям нуля и единицы.

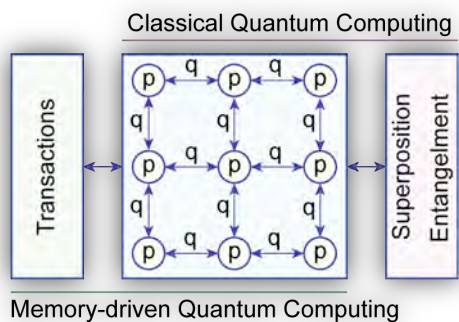


Рис. 1. Два вида квантового компьютеринга

Функционально полный базис для создания архитектуры квантового компьютеринга представлен операциями суперпозиции и перепутывания, которым можно поставить в соответствие традиционный логический базис или-не. В теории множеств данному базису ставится в соответствие изоморфизм в форме пары «объединение-дополнение». Однозначно, операция суперпозиции в квантовой физике изоморфна логической инверсии или теоретико-множественному дополнению в алгебре логики. Поэтому естественно, что каждое состояние двоичного разряда (электрона) после применения данной операции к биту информации знает друг о друге все, где бы они не находились, с точностью до инверсии a_i/\bar{a}_i . Далее на основе упомянутой пары операций-примитивов (или, не) строится более сложная система логических компонентов и устройств для организации и оптимизации вычислительных процессов. Недостатки квантового классического компьютеринга: 1) Высокая стоимость поддержания температурных условий для функционирования квантовых атомарных структур на уровне -270 градусов Цельсия. 2) Наблюдаемость результатов вычислительных процессов, приводящих к разрушению данных после их чтения.

Инновация в архитектуре квантового компьютеринга определяется устранением логики, связанной с суперпозицией и перепутыванием. Аналогом могут выступать memory-driven архитектуры классического компьютера, свободные от reusable logic. В таком компьютере нет ничего, кроме памяти, где реализуется транзакция (операции записи-считывания данных) на адресуемой памяти. Транзакции достаточно для организации любого вычислительного процесса путем использования единственного характеристического уравнения [11, 13]: $M_i = Q_i[M(X_i)]$.

Здесь M_i , Q_i , X_i представляют собой компоненты памяти: для вектора-состояния вычислительного процесса; вектора-кубита логического примитива; вектора-адреса ячейки логического примитива. Q-логика реализуется на адресной памяти, где также осуществляется ассемблирование всех примитивов с помощью интегрирующего M-вектора состояния вычислительного устройства, который формирует двоичные адреса $M(X)$ на основе использования массива номеров входных переменных X .

Инновационное предложение заключается в создании квантового memory-driven компьютеринга без квантовых операций суперпозиции и перепутывания (или, не) на основе использования приведенного выше характеристического уравнения, задающего две транзакции записи-считывания на

структуре электронов (см. рис. 1). Исключить две сложные операции из квантового компьютеринга – означает существенно упростить архитектуру и привести ее к структуре памяти на электронах для выполнения транзакций между ними с помощью квантов или фотонов.

Подтверждением состоятельности предложенной инновационной квантовой архитектуры может служить несколько свежих публикаций, которые фиксируют устойчивую тенденцию к созданию квантового компьютеринга на атомарной структуре памяти с передачей информации с помощью фотонов или квантов.

Ученые из Калифорнийского технологического института создали оптическую квантовую память [14], в которой информация передается путем кодирования данных с использованием квантового состояния фотонов. Память реализована на редкоземельных элементах и способна сохранять состояния фотонов с помощью резонаторов-посредников между атомом и светом. Размерность квантовой памяти в 1000 раз меньше, чем традиционные классические решения. Она реализована в нанополости, которая позволяет хранить информацию в очень небольшом объеме.

Практическая реализация идеи замены электронов фотонами приводит к созданию компьютеринга с быстройдействием, близким к скорости света [15]. Корейские исследователи сделали еще один шаг к оптическим вычислениям. Они создали *photon-triggered* нанопроволочный транзистор на основе кристаллического и пористого кремния, где переключение и усиление величины токового сигнала осуществляется под воздействием фотона. Использование фотонов в логических вентилях AND, OR и NAND приведет к ультракомпактным нанопроцессорам и наноразмерным фотоприемникам для получения изображений с высоким разрешением.

Ученые из Колумбийского университета провели успешные опыты по созданию транзистора из одного атома в молекулярной электронике [16]. Они реализовали геометрически упорядоченный кластер неорганических атомов с центральным ядром, состоящим из 14 атомов, которое связали с золотыми электродами, что позволило управлять транзистором под воздействием одного электрона при комнатной температуре.

Впервые достигнута передача цифровых сигналов между молекулами, что является существенным достижением на пути развития молекулярного компьютеринга [17]. Создание электронных компонентов из отдельных молекул является многообещающей стратегией для миниатюризации и интеграции электронных устройств. Однако практическая реализация молекулярных устройств и схем для передачи и обработки сиг-

налов при комнатной температуре оказалась сложной задачей, которая решена путем размещения молекул SnCl_2Pc на поверхности меди (Cu). Плоскостная ориентация молекул в межмолекулярном взаимодействии служит носителем информации. В связанных молекулярных массивах сигнал передается от одной молекулы к другой по наперед заданным маршрутам, реализующим логические операции. Явления плоскостной ориентации позволяют использовать молекулы, имеющие внутренние бистабильные состояния, для создания сложных молекулярных устройств и схем.

Теоретическое сходство классического компьютеринга с квантовым заключается в общей модели вычислительной архитектуры, которая использует: память для хранения данных и функционально полный базис примитивных элементов (или, не) = (суперпозиция, перепутывание) для реализации арифметико-логических операций над данными.

Каковы же формальные различия между классическим и квантовым компьютерингом? Первый из них последовательно обрабатывает адресуемые или упорядоченные гетерогенные данные, затрачивая на процедуру $Q=n$ тактов. Он также способен обработать гомогенные данные параллельно и за один автоматный такт. Если данные не упорядочены и представляют собой множества, то предельная вычислительная сложность их обработки на классическом компьютере зависит от мощности двух множеств и определяется как $Q = n \times m$. Например, для пересечения двух множеств: $M_1 \cap M_2 = \{Q, E, H\} \cap \{E, H, J\} = \{E, H\}$ необходимо затратить 6 автоматных тактов. Квантовый компьютеринг устраняет данный недостаток, связанный с квадратичной или мультипликативной вычислительной сложностью процедуры пересечения на классическом компьютере. Он решает задачу одновременной и параллельной обработки теоретико-множественных данных. Примером тому может служить параллельное выполнение приведенной выше операции пересечения над множествами за один автоматный такт. Для этого предварительно выполняется операция суперпозиции или объединения примитивных символов, входящих в множества $M_1 \cap M_2 = \{V\} \cap \{C\} = \{P\} = \{E, H\}$, но при этом используется замкнутый теоретико-множественный алфавит [10]: $V^*(Y) = \{Q, E, H, J, O = \{Q, H\}, I = \{E, J\}, A = \{Q, E\}, B = \{H, J\}, S = \{Q, J\}, P = \{E, H\}, C = \{E, H, J\}, F = \{Q, H, J\}, L = \{Q, E, J\}, V = \{Q, E, H\}, Y = \{Q, E, H, J\}, U = \emptyset\}$. Символы алфавита представляют собой множество всех подмножеств на универсуме Y , которые составлены путем суперпозиции примитивов.

Квантовая суперпозиция дает возможность сосредоточить в одной точке гильбертова пространства несколько дискретных состояний. Аналогично операция объединения также создает в одной точке дискретного пространства символический образ, содержащий несколько состояний. Исходя из сказанного, достаточно просто использовать многозначный замкнутый алфавит для моделирования квантовых вычислений на классическом компьютере. Но для этого необходимо предварительно создать символическую систему или алгебру множеств для кодирования состояний. Самой простой является алгебра Кантора, которая оперирует двумя дискретными состояниями и создает 4 символа: $A^k = \{0, 1, X = \{0, 1\}, \emptyset\}$. Символика данного алфавита с точностью до изоморфизма представляет собой теоретико-множественную интерпретацию кубита. Иначе, суперпозиция двух состояний одного кубита образует 4 символа. Естественно, что два кубита способны сгенерировать 16 состояний, три кубита – 64 состояния. В общем случае число состояний Q имеет зависимость от числа кубитов n , которая представлена следующей формулой: $Q = 2^{2^n}$.

Для параллельного выполнения, но уже логических операций над кубитами интерес представляет кодирование примитивных символов алфавита унарным двоичным кодом. Остальные символы получаются суперпозицией кодов примитивов. Исключение составляет код символа пустого множества, который получается путем применения операции логического умножения (пересечения). Для алгебры Кантора таблица соответствия “Symbol – Code” имеет следующий вид:

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

Платой за параллелизм выполнения логических операций над множествами в классическом компьютере является существенное увеличение разрядности (регистра, памяти) для унарного кодирования примитивных символов алфавита. Аналогичная таблица соответствия для кодирования 16-ричного алфавита $B^*(Y)$ на четырех унарных кодах примитивов имеет вид:

$a_i \in B^*$	Q	E	N	J	O	I	A	B	S	P	C	F	L	V	Y	\emptyset
$C(a_i)$	1	0	0	0	1	0	1	0	1	0	0	1	1	1	1	0
	0	1	0	0	0	1	1	0	0	1	1	0	1	1	1	0
	0	0	1	0	1	0	0	1	0	1	1	1	0	1	1	0
	0	0	0	1	0	1	0	1	1	0	1	1	0	1	0	1

Что касается табличной модели логического элемента, он изначально представлен квантоподобной совокупностью равнозначных строк или множеством дискретных отношений между

входными и выходными переменными. Взамен такого множества предлагается кубитный вектор выходных состояний, ориентированный на адресное параллельное моделирование цифровых логических схем. Замена неупорядоченного множества равнозначных строк таблицы истинности на упорядоченный вектор адресуемых состояний дает возможность создавать параллельный компьютер на классических вычислителях за счет увеличения объема памяти для унарного кодирования каждого состояния. Иначе, суперпозиция n равнозначных элементов конечного множества в квантовом (Q) компьютеринге имеет взаимно-однозначное соответствие n -мерному вектору в классическом (C) адресуемом компьютеринге (рис. 2). Данный вектор получен путем выполнения og-операции над унарными кодами примитивных элементов исходного множества. Естественно, что любое пересечение (and), объединение (or) или дополнение (not) в C-компьютере унарных кодов данных выполняется параллельно, за один автоматный такт, как и в Q-компьютере. Платой за полученное быстродействие является увеличение объема памяти (количества битов) для унарного кодирования символов относительно позиционного кодирования, которое определяется следующим выражением: $Q = n^2 / n \times \log n = n / \log n$.

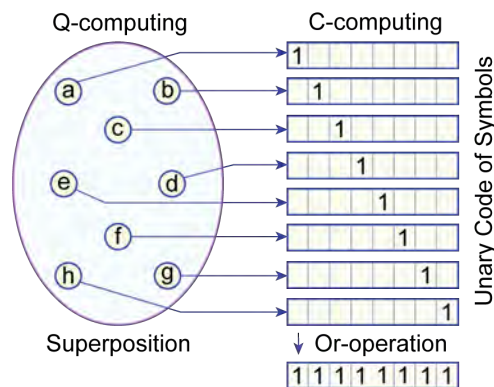


Рис. 2. Суперпозиция элементов и логическое объединение векторов

Таким образом, расширению мощности теоретико-множественного алфавита можно поставить в соответствие наращивание кубитов в квантовом компьютере. Это дает возможность параллельно и в одном автоматном такте осуществлять вычислительные процессы на основе использования логических (теоретико-множественных) операций. Суперпозиции n элементов конечного множества в квантовом (Q) компьютеринге взаимно-однозначно соответствует n -мерный вектор в классическом (C) адресуемом компьютеринге, который получается на основе использования og-операции над унарными кодами символов ис-

ходного множества, выполняемой параллельно за один автоматный такт.

Интерес представляет оптимальное решение проблемы покрытия путем использования цифровой регистровой структуры (Quantum Coverage Processor – QCP), создающей все возможные сочетания входных векторов в форме кубитных структур данных, представленных булеаном или множеством всех подмножеств (рис. 3).

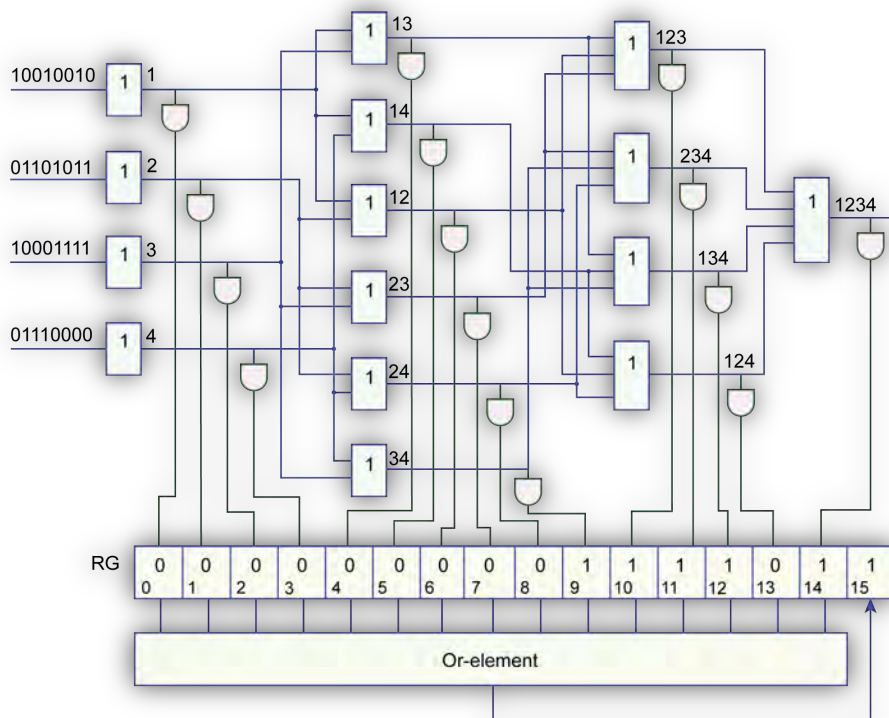


Рис. 3. QC-Processor для определения оптимального покрытия

Схема имеет логический анализатор при каждом регистровом выходе, который определяет полноту покрытия путем выполнения and-функции по всем разрядам регистровой переменной. Число таких функций-преобразователей “вектор-бит” соответствует количеству элементов и равно $Q = 2^n - 1$. Битовые результаты вычисления and-функций интегрируются в регистр анализа RG, который своими единичными разрядами идентифицирует получение покрытия. Последний разряд регистра, равный 1, свидетельствует о существовании позитивного результата, полученного в процессе поиска покрытия. При этом минимальное покрытие определяется крайней левой единицей в регистре-анализаторе RG. Схема также предназначена для определения примитивизма или уникальности входных векторов, что идентифицируется нулевыми значениями по всем разрядам регистра-анализатора, кроме последнего, равного в этом случае единице. Аппаратная сложность регистровой цифровой схемы для поиска оптимального покрытия, где n – ко-

личество строк, m – длина регистра, определяется следующим аналитическим выражением:

$Q = [2^{n+1} + 2^n] \times m + 2^n$. Вычислительная сложность совокупных процедур при поиске оптимального покрытия равна, в худшем случае, n автоматным тактам. Вместо og-операций в логических элементах можно использовать хог-функции, что позволит решать задачи по иденти-

фикации и распознаванию киберобъектов, представленных в векторной форме. Здесь используется аксиома: хог-операция двух векторов-примитивов создает их логическое объединение или суперпозицию. Таким образом, n объектов в дискретном векторном пространстве распознаются, если все хог-сочетания, кроме последнего, формируют нулевые значения на выходе and-анализаторов.

Пример. Определить минимальное покрытие единичными значениями восьми разрядов следующей таблицы:

X – Inputs	0	1	2	3	4	5	6	7
1	1	0	0	1	0	0	1	0
2	0	1	1	0	1	0	1	1
3	1	0	0	0	1	1	1	1
4	0	1	1	1	0	0	0	0

Воздействие на схему четырьмя регистровыми переменными: $X = (1,2,3,4)$ формирует следующее состояние регистра анализатора:

$$RG = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 1\ 1]$$

Единичные значения регистра показывают существование четырех возможных вариантов решения задачи: $C = \{3,4\}$, $\{1,2,3\}$, $\{2,3,4\}$, $\{1,3,4\}$, $\{1,2,3,4\}$. Минимальное покрытие обеспечивается двумя входными векторами: $C = \{3,4\}$, которое идентифицируется левой крайней единицей в регистре-анализаторе RG.

QCP-структура может быть упрощена путем устранения регистра, который выполняет функцию хранения результатов поиска оптимального покрытия. В этом случае QCP-схема становится строго логической, где позитивный результат поиска определяется уже за один автоматный такт единичным значением состояния выхода интегрального og-элемента. Оптимальное покрытие будет идентифицироваться единичным зна-

чением выхода and-элемента, который топологически ближе находится к внешним входам схемы.

5. Метод квантовой минимизации булевых функций

Квантовое представление данных в виде суперпозиции унарных кодов можно использовать для существенного упрощения метода неопределенных коэффициентов при минимизации булевых функций.

Утверждение. Любая сколь угодно сложная таблица истинности дискретного объекта может быть представлена не более, чем двумя векторами квантового покрытия при унитарном кодировании (UC) входных состояний. Процедура, иллюстрирующая данное утверждение, показана на рис. 4. Здесь представлены нулевые и единичные кубы таблицы истинности – входные состояния, которые унарно кодируются и логически объединяются (VUC). В результате получаются два вектора квантового покрытия, где каждый из которых может представлять логическую функцию в форме кубитного покрытия.

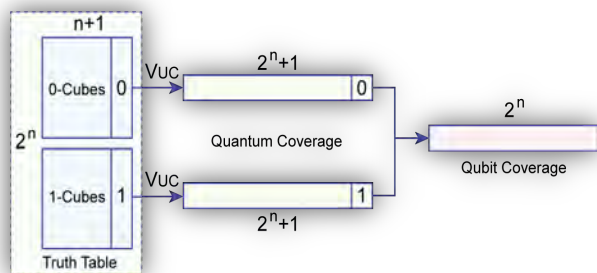


Рис. 4. Модель получения кубитного покрытия

Пусть имеется таблица $T = T_{ij}$, $i = \overline{1, m}$; $j = \overline{1, n}$. В худшем случае число состояний в каждом столбце таблицы (матрицы) равно m . Данное количество состояний может быть представлено m рядами унитарного кода. Например, два символа алфавита Кантора можно представить одной строкой (11), состоящей из двух двоичных разрядов:

$$A^k = \{0 \rightarrow 10; 1 \rightarrow 01; X \rightarrow 11; \emptyset \rightarrow 00\}.$$

В результате получается матрица размерностью m на n , где каждая ячейка содержит m -разрядный вектор. Применение операции суперпозиции или логического объединения ко всем строкам таблицы, благодаря унитарному кодированию, создает одну строку, которая формирует в компактном виде отношения, ранее представленные исходной таблицей. Например, все двоично-десятичные коды входных состояний логического элемента от трех переменных представляются вектором (11111111), если учитывать следующее кодирование: 000 – 10000000, 001 – 01000000, 010 – 00100000, 011 – 00010000, 100 – 00001000,

101 – 00000100, 110 – 00000010, 111 – 00000001. Однако таблица истинности представляет собой функциональное соответствие:

$$Y = f(X), X \rightarrow Y, X = \{x_1, x_2, \dots, x_i, \dots, x_n\}, Y = \{0, 1\}.$$

Учитывая, что функция определена на двух дискретных значениях $\{0, 1\}$, то двоичная по выходам таблица истинности может быть всегда представлена двумя строками, каждая из которых суперпозиционирует или собирает нулевые или единичные унитарные коды входных воздействий. Например, таблица истинности хог-элемента на три входа после логического объединения унарных кодов входных состояний по единичному и нулевому значению выхода будет иметь следующий вид:

a	b	c	Y	a	b	c	U-code	Y
0	0	0	0	0	0	0	10000000	0
0	0	1	1	0	0	1	01000000	1
0	1	0	1	0	1	0	00100000	1
0	1	1	0	0	1	1	00010000	0
1	0	0	1	1	0	0	00001000	1
1	0	1	0	1	0	1	00000100	0
1	1	0	0	1	1	0	00000010	0
1	1	1	1	1	1	1	00000001	1

V	VY=1	0	1	1	0	1	0	0	1	1
V	VY=0	1	0	0	1	0	1	1	0	0

Таким образом, любая таблица истинности цифрового устройства может быть представлена в явном виде двумя строками-кубами квантового покрытия, которые суперпозиционно объединяют единичные и нулевые (по состоянию выхода) входные воздействия исходной таблицы. Учитывая, что кубы квантового покрытия всегда взаимно инверсны, то для задания функциональности достаточно оставить один из них, полагая, что второй можно быстро и просто достроить с помощью операции инверсии, в случае необходимости.

С учетом использования полученной информации о возможности представления таблицы истинности двумя кубами квантового покрытия далее предлагается модернизация известного метода неопределенных коэффициентов для минимизации логических функций. Пусть имеется исходная таблица неопределенных коэффициентов для минимизации булевой функции от трех переменных [21]:

T _{ij}	x ₁ x ₂ x ₃	k ₁	k ₂	k ₃	k ₁₂	k ₁₃	k ₂₃	k ₁₂₃	f _i
0	000	k ₁ ⁰	k ₂ ⁰	k ₃ ⁰	k ₁₂ ⁰⁰	k ₁₃ ⁰⁰	k ₂₃ ⁰⁰	k ₁₂₃ ⁰⁰⁰	1
1	001	k ₁ ⁰	k ₂ ⁰	k ₃ ¹	k ₁₂ ⁰⁰	k ₁₃ ⁰¹	k ₂₃ ⁰¹	k ₁₂₃ ⁰⁰¹	0
2	010	k ₁ ⁰	k ₂ ¹	k ₃ ⁰	k ₁₂ ⁰¹	k ₁₃ ⁰⁰	k ₂₃ ¹⁰	k ₁₂₃ ⁰¹⁰	1
3	011	k ₁ ⁰	k ₂ ¹	k ₃ ¹	k ₁₂ ⁰¹	k ₁₃ ⁰¹	k ₂₃ ¹¹	k ₁₂₃ ⁰¹¹	0
4	100	k ₁ ⁰	k ₂ ⁰	k ₃ ⁰	k ₁₂ ¹⁰	k ₁₃ ¹⁰	k ₂₃ ⁰⁰	k ₁₂₃ ¹⁰⁰	1
5	101	k ₁ ¹	k ₂ ⁰	k ₃ ¹	k ₁₂ ¹⁰	k ₁₃ ¹¹	k ₂₃ ⁰¹	k ₁₂₃ ¹⁰¹	0
6	110	k ₁ ¹	k ₂ ¹	k ₃ ⁰	k ₁₂ ¹¹	k ₁₃ ¹⁰	k ₂₃ ¹⁰	k ₁₂₃ ¹¹⁰	0
7	111	k ₁ ¹	k ₂ ¹	k ₃ ¹	k ₁₂ ¹¹	k ₁₃ ¹¹	k ₂₃ ¹¹	k ₁₂₃ ¹¹¹	1

Данная таблица преобразуется к двоичному виду всех возможных комбинаторных сочетаний состояний входных переменных, которые способны сформировать значения выходов функции, представленной в последнем столбце:

T _{ij}	x ₁ x ₂ x ₃	1	2	3	12	13	23	123	f
0	000	0	0	0	00	00	00	000	1
1	001	0	0	1	00	01	01	001	0
2	010	0	1	0	01	00	10	010	1
3	011	0	1	1	01	01	11	011	0
4	100	1	0	0	10	10	00	100	1
5	101	1	0	1	10	11	01	101	0
6	110	1	1	0	11	10	10	110	0
7	111	1	1	1	11	11	11	111	1

Естественно, что полученные в ячейках таблицы комбинации входных воздействий: 0,1; 00, 01, 10, 11; 000, 001, 010, 011, 100, 101, 110, 111 три-виально трансформируются в унитарные коды двоичных состояний 10, 01; 1000, 0100, 0010, 0001; 10000000, 01000000, 00100000, 00010000, 00001000, 00000100, 00000010, 00000001 соответственно:

T _{ij}	x ₁ x ₂ x ₃	1	2	3	12	13	23	123	f
0	000	10	10	10	1000	1000	1000	10000000	1
1	001	10	10	01	1000	0100	0100	01000000	0
2	010	10	01	10	0100	1000	0010	00100000	1
3	011	10	01	01	0100	0100	0001	00010000	0
4	100	01	10	10	0010	0010	1000	00001000	1
5	101	01	10	01	0010	0001	0100	00000100	0
6	110	01	01	10	0001	0010	0010	00000010	0
7	111	01	01	01	0001	0001	0001	00000001	1

Далее выполняется раздельное логическое объединение всех единичных и нулевых строк таблицы в два интегрирующих вектора. В результате получают все возможные сочетания переменных, которые формируют единичные и нулевые значения функции:

Q	Operations	1	2	3	12	13	23	123	f
1	$Q^1 = \bigvee_{f_i=1} T_{ij}$	11	11	11	1111	1011	1011	10101001	1
2	$Q^0 = \bigvee_{f_i=0} T_{ij}$	11	11	11	1111	0111	0111	01010110	0
3	$Q = (\bigvee_{f_i=1} T_{ij}) \wedge (\bigvee_{f_i=0} T_{ij})$	00	00	00	0000	1000	1000	10101001	Y

Чтобы получить дизъюнктивную форму минимизированной размерности (строка 3 в приведенной выше Q-таблице), необходимо из единичного куба квантового покрытия логически вычесть нулевой куб по правилу, представленному в следующей формуле:

$$Q = (\bigcup_{f_i=1} T_{ij}) \setminus (\bigcup_{f_i=0} T_{ij}) = (\bigvee_{f_i=1} T_{ij}) \wedge (\overline{\bigvee_{f_i=0} T_{ij}}).$$

Дешифрирование полученного квантового куба Q в дизъюнктивную нормальную форму дает следующий результат:

$$Y = \bar{x}_1 \bar{x}_3 \vee \bar{x}_2 \bar{x}_3 \vee \bar{x}_1 \bar{x}_2 \bar{x}_3 \vee \bar{x}_1 x_2 \bar{x}_3 \vee x_1 \bar{x}_2 \bar{x}_3 \vee x_1 x_2 x_3.$$

Данная форма не является минимальной, поэтому нуждается в применении средств, позволяющих решить задачу покрытия простейших исходных единичных термов (000, 010, 100, 111) полученными решениями. Для функции Y очевидно, что первые два терма (0x0, x00) покрывают логические слагаемые 3,4,5 или (000, 010, 100), которые, в данном случае, являются избыточными в соответствии с правилом поглощения $a \vee ab = a$, что дает возможность записать минимальную дизъюнктивную нормальную форму в следующем виде:

$$Y = \bar{x}_1 \bar{x}_3 \vee \bar{x}_2 \bar{x}_3 \vee x_1 x_2 x_3.$$

Другое решение задачи покрытия связано с применением QC-процессора, имеющего 6 регистровых входов, что позволит определить минимальную ДНФ путем моделирования двоичных кодов-строк $X_i \in X$ таблицы покрытия:

T	000	010	100	111	$X_i \in X$
0x0	1	1	.	.	1100
x00	1	.	1	.	1010
000	1	.	.	.	1000
010	.	1	.	.	0100
100	.	.	1	.	0010
111	.	.	.	1	0001

Результат моделирования кодов определяет в качестве минимального покрытия три кода-строки, которые создают минимальную функцию:

$$Y = 1100 \vee 1010 \vee 0001 \rightarrow 0x0 \vee x00 \vee 111 \rightarrow \bar{x}_1 \bar{x}_3 \vee \bar{x}_2 \bar{x}_3 \vee x_1 x_2 x_3.$$

Данный метод целесообразно использовать для получения минимальной ДНФ или КНФ по таблицам истинности, где количество нулевых и единичных кубов-строк не сильно отличается друг от друга. Другое применение метода связано с существенной минимизацией дефектной области при поиске неисправностей в цифровых системах.

Вычислительная сложность Q квантового метода неопределенных коэффициентов определяется выражением, которое формирует время для унарного кодирования состояний таблицы истинности (для сравнения Q^b – есть сложность базового метода минимизации):

$$Q = 2^n \times 3^n;$$

$$Q^b = 2^n \times 2^n \times 2^n = 2^n \times 2^{n+n};$$

$$R = \frac{Q}{Q^b} = \frac{2^n \times 3^n}{2^n \times 2^{n+n}} = \frac{3^n}{2^{2n}}.$$

Таким образом, вычислительная сложность Q получения компактного квантового покрытия для минимизации булевых функций существенно

меньше по сравнению с базовым методом Q^b неопределенных коэффициентов, использующим специальной формы таблицу истинности. Исключив предварительную обработку таблицы истинности, которая заключается в унарном кодировании состояний, вычислительная сложность собственно метода минимизации булевых функций включает всего три векторных параллельных операции.

Затраты памяти H для хранения структур данных формируются размерностью таблицы, необходимой для суперпозиционного получения двух векторов квантового покрытия, где ячейки таблицы представлены унарными кодами состояний:

$$H = 2^n \times 3^n;$$

$$H^b = 2^n \times 2^n = 2^{2n};$$

$$S = \frac{H}{H^b} = \frac{2^n \times 3^n}{2^n \times 2^n} = \frac{3^n}{2^n}.$$

Таким образом, чтобы получить компактное квантовое покрытие, необходимо использовать таблицу H существенно большей размерности по сравнению с исходной таблицей истинности H^b . Предложен квантовый способ минимизации булевых функций, который отличается от метода неопределенных коэффициентов параллельным выполнением операции суперпозиции над нулевыми и единичными состояниями входных переменных, представленными унитарными кодами, что дает возможность существенно повысить быстродействие за счет избыточной памяти.

6. Кубитный метод поиска дефектов

Рассматривается кубитный метод поиска дефектов путем теоретико-множественной разности двух векторов, соответствующих единичному и нулевому значению состояний выходов, как реакций наблюдаемых выходов на входной тест проверки неисправностей:

$$F = \left(\bigcup_{\forall R_i=1} Q_{ij} \right) \setminus \left(\bigcup_{\forall R_i=0} Q_{ij} \right) = \left(\bigvee_{\exists R_i=1} Q_{ij} \right) \wedge \left(\bigwedge_{\exists R_i=0} \overline{Q_{ij}} \right).$$

Структуры данных представлены таблицей неисправностей на декартовом произведении тестовых наборов и множестве линий объекта диагностирования, где каждая ячейка представляет собой два бита: первый из них идентифицирует проверяемую константную неисправность нуля (10), а второй – константную неисправность единицы (01):

$$Q = \{F, T, L\},$$

$$Q = Q_{ij}, i = 1, m; j = 1, n;$$

$$F = (F_1, F_2, \dots, F_j, \dots, F_n),$$

$$F_j = \{10 \equiv 0; 01 \equiv 1; 11 \equiv \{ \neq 0, \neq 1 \}; 00 \equiv \emptyset\};$$

$$T = (T_1, T_2, \dots, T_i, \dots, T_m);$$

$$L = (L_1, L_2, \dots, L_j, \dots, L_n).$$

Суперпозиция неисправностей (две единицы на одной линии-ячейке) дает возможность существенно минимизировать структуры данных для хранения информации в целях последующего поиска дефектов при выполнении диагностического эксперимента в режиме online.

Для проверки метода поиска дефектов далее предлагается логическая схема, представленная на рис. 5, которая имеет 6 элементов and-not, 11 линий, 5 входов и 2 выхода.

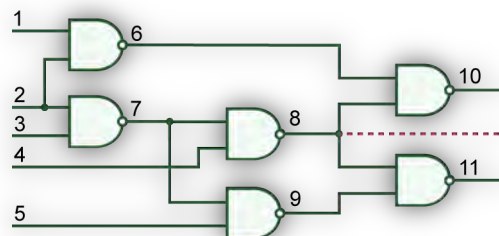


Рис. 5. ISCAS-схема для верификации

Следующая таблица иллюстрирует выполнение диагностического эксперимента в целях объединения множества дефектов, которые формируют некорректные состояния выходов на тестовых наборах $\{T1-R10; T5-R11; T6-R10, R11; T8-R11\}$:

Q = Q _{ij}	1	2	3	4	5	6	7	8	9	10	11	R ₁₀	R ₁₁
T ₁	01	10	01	00	10	00	10	10	00	10	01	1	0
T ₂	10	00	10	00	01	10	00	00	10	01	10	0	0
T ₃	00	01	01	00	00	01	10	01	01	10	10	0	0
T ₄	10	00	01	00	10	00	01	00	10	01	01	0	0
T ₅	00	10	00	01	00	01	00	10	00	10	10	0	1
T ₆	01	10	00	00	10	00	00	01	10	01	10	1	1
T ₇	01	00	00	10	00	00	01	00	10	01	01	0	0
T ₈	00	10	10	01	01	10	00	00	00	01	10	0	1
Q ₁	01	11	11	01	11	11	10	11	10	11	11	1	1
Q ₀	11	01	11	10	11	11	11	01	11	11	11	0	0
F	00	10	00	01	00	00	00	10	00	00	00	1/0	1/0

Здесь дизъюнкция строк T₁, T₅, T₆, T₈ формирует вектор Q₁, который собирает все возможные дефекты, проверяемые на тестовых наборах. Вектор Q₀ с помощью строк T₂, T₃, T₄, T₇ объединяет все невозможные, непроверяемые на тестовых наборах дефекты. Вычитание всех невозможных из всех возможных дефектов дает искомый результат в виде трех неисправностей, закодированных как F₂ = 10; F₄ = 01; F₈ = 10. Таким образом, параллельное выполнение двух регистровых or-операций на основе результатов проведенного диагностического эксперимента позволило определить три возможных неисправности, каждая из которых может иметь место в логической схеме:

$$F = \{2^0, 4^1, 8^0\}.$$

Более жестким является ограничительное условие существования в логической схеме одиноч-

ного константного дефекта, использование которого приводит к вычислению дефектов на основе следующего выражения:

$$F = \left(\bigcap_{\forall R_i=1} Q_{ij} \right) \setminus \left(\bigcup_{\forall R_i=0} Q_{ij} \right) = \left(\bigwedge_{\exists R_i=1} Q_{ij} \right) \wedge \left(\bigvee_{\exists R_i=0} \overline{Q_{ij}} \right).$$

Применение данной формулы существенно уточняет результат диагностирования и приводит его к виду: $F = \{2^0\}$ из-за противоречивости кодов дефектов по and-операции в столбцах 4 и 8. Условие наличия в логической схеме одиночной константной неисправности ставит во главу угла следующее утверждение.

Утверждение. Если в столбце таблицы неисправностей существует координата 00 или 01, которая создает на наблюдаемых выходах некорректность $R=1$, связанную с неисправностью 10 на остальных координатах столбца, то такой одиночный дефект (10) в логической схеме невозможен.

Д о к а з а т е л ь с т в о. Пусть на n тестовых наборах зафиксировано несовпадение на внешних выходах эталонных и реальных значений сигналов. При этом $n-1$ координата в рассматриваемом столбце имеет значение 10 (01) и лишь одна n -координата имеет значение 01 (10). Если предположить, что в логической схеме имеется дефект 10, то на n -координате также должен присутствовать дефект 10, который создает некорректное состояние выходов. Но по условиям моделирования такой дефект там отсутствует. Следовательно, невозможно считать, что в схеме присутствует дефект 10. Это подтверждается также формальным результатом – пустым пересечением всех координат столбца, связанных с некорректными состояниями выходов схемы:

$$F = \left(\bigwedge_{\exists R_i=1} Q_{ij} \right) = \begin{cases} 10 \wedge 10 \wedge \dots \wedge 10 \wedge 01 = 00; \\ 10 \wedge 10 \wedge \dots \wedge 10 \wedge 00 = 00. \end{cases}$$

Все сказанное относится и к состоянию n -координаты, которое идентифицируется сигналом пустого множества 00, взаимодействие с которым также делает невозможным присутствие в логической схеме одиночной константной 0-неисправности (код 10).

7. Cloud Service «Quantum Modeling»

Предлагается облачный сервис проектирования и тестирования цифровых устройств, который отличается от существующих аналогов эргономичным интерфейсом быстрого ввода структурно-функционального графического описания в виде рисунка, а также адресно-ориентированным кубитно-векторным моделированием исправного поведения и неисправностей в целях существенного повышения быстродействия интерпретативного анализа, синтеза тестов и диагностирования.

Цель – предоставление облачных сервисов проектирования и тестирования цифровых устройств широкому кругу специалистов и студентов, связанному с внедрением и отработкой новой технологии квантового моделирования на основе использования кубитных структур данных.

Задачи: 1) Структуры кубитных данных для проектирования модели цифрового устройства. 2) Структура функциональных компонентов облачного сервиса «Quantum Modeling». 3) Тестирование сервиса моделирования на реальных цифровых логических схемах. 4) Облачная имплементация сервисов моделирования и тестирования логических схем. 5) Практические результаты и выводы.

8. Структуры кубитных данных цифрового устройства

Можно ли отказаться от структурно-логической схемы описания модели при синтезе и анализе цифрового изделия? Ответ связан с методу-driven реализацией логических или комбинационных схем. Для этого необходимо создавать новые оригинальные форматы визуализации, нетрадиционные по отношению к исторически сложившимся шаблонам проектирования компьютерных компонентов. Далее рассматривается пример логической схемы (рис. 6) из библиотеки международного симпозиума ISCAS, для которой выполняется построение структур данных.

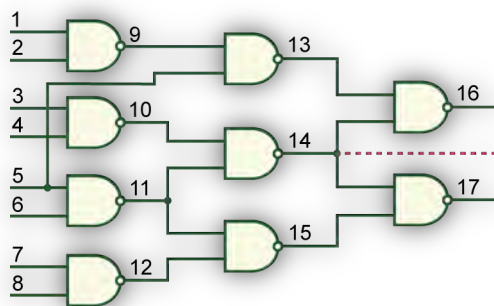


Рис. 6. Структура логических элементов из библиотеки ISCAS

Одним из возможных вариантов инновационной формы описания цифровых логических схем является суперпозиционная структура кубитных векторов, представленная на рис. 7. Здесь результат анализа определяется суперпозицией кубитных векторов, создающих вектор моделирования M , где алгоритм анализа содержит одну процедуру – сдвига векторов вверх-вниз относительно вектора моделирования. Виртуальный сдвиг технологически просто реализуется путем вычисления адресов ячеек кубитных векторов $M_i = Q_i[M(X_i)]$, из которых считывается информация и заносится в вектор моделирования исправного поведения. Структурная взаимосвязь

кубитных векторов-примитивов осуществляется посредством нумерации логических переменных: входных, внутренних и выходных. Переменные формируют своими двоичными состояниями адреса ячеек кубитных векторов для вычисления реакции цифрового устройства на входное воздействие (01001110 110101110).

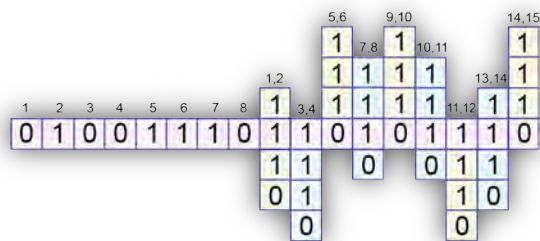


Рис. 7. Инновационные кубитные структуры данных логической схемы

Структуры кубитных данных для синтеза и анализа цифровых систем и компонентов оперируют следующей моделью описания логических схем:

$$\begin{aligned}
 S &= \{M, X, Y, Q\}, \\
 M &= (M_1, M_2, \dots, M_i, \dots, M_m), \\
 X &= (X_1, X_2, \dots, X_i, \dots, X_n), \\
 Y &= (Y_1, Y_2, \dots, Y_i, \dots, Y_k), \\
 Q &= (Q_1, Q_2, \dots, Q_i, \dots, Q_q), \\
 M_i &= Q_i[M(X_i)].
 \end{aligned}$$

Здесь представлены следующие системные компоненты: M – вектор моделирования цифрового устройства, который связывает все кубитные покрытия примитивных элементов в структуру; X – вектор входных переменных, заданных двоичными значениями; Y – вектор выходных переменных, которые формируют реакцию цифрового устройства; Q – кубитное покрытие, представленное в виде вектора выходных состояний логического элемента и предназначенное для формирования его функции. Основное характеристическое уравнение для моделирования цифрового устройства оперирует вычислением адресов для записи-считывания данных, что создает простые и быстродействующие транзакции между вектором моделирования и кубитными покрытиями:

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

Чтобы определить двоичное значение логической переменной или линии M_i , необходимо сформировать адрес ячейки кубитного покрытия, который создается конкатенацией двоичных состояний вектора моделирования M , где адреса ячеек вектора задаются номерами-идентификаторами входных переменных X . Характеристическое уравнение непосредственно влияет на быстродействие квантового метода моделирования, которое зависит от операций

конкатенации k , считывания r и записи w битов, количества q кубитных покрытий в цифровой схеме или логических примитивов, а также длины теста (входных наборов) t :

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

Таким образом, всего три транзакционных операции необходимы для обработки логического элемента любой функциональной сложности!

9. Структура облачного сервиса «Quantum Modeling»

Современная стратегия компьютерного бизнеса заключается в массовом переводе всех процессов и явлений из чисто физического в киберфизическое пространство. Это делает сервисы менее уязвимыми для киберпреступности, экономически выгодными и технологически доступными в пространстве и во времени, без каких-либо ограничений. Поэтому высокотехнологический бизнес Design and Test, определяемый компаниями, формирующими индекс капитализации NASDAQ, усиленными темпами уходит в облачный киберфизический компьютеринг, который имеет следующую примерную структуру, представленную на рис. 8.

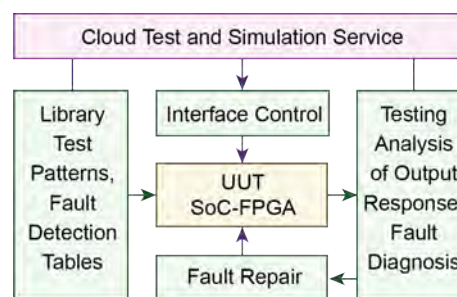


Рис. 8. Design and Test Cloud Online Service

Здесь имеется только один физический модуль с IP-адресом, Unit Under Test (любые цифровые устройства, сенсоры, актюаторы, мобильные устройства, автомобильные компьютеры, конечные гаджеты), который находится в физическом пространстве. Остальные блоки, указанные в структуре, создают облачные сервисы, доступные в режиме онлайн 24/7. Таким образом, любое физическое устройство может быть легко продиагностировано, благодаря его подключению к облачным сервисам в режиме автономного тестирования или даже в режиме рабочего функционирования. Все сказанное относится и к образовательному процессу, когда студенты могут воспользоваться сколь угодно сложной аппаратурой, подключенной через интернет к собственному гаджету для получения киберфизических сервисов, связанных с выполнением реальных технологических экспериментов на компьютерных устройствах.

Облачный сервис «Quantum Modeling» реализован на языке SWIFT (2300 строк кода) и содержит следующие структурные компоненты или модули: 1) Q-function – кубитная модель функционального примитива, входящая в состав логической схемы. 2) Data Structures – структура цифрового устройства, основанная на использовании сквозной нумерации входных внутренних и выходных переменных, которые фигурируют в качестве вход-выходных идентификаторов линий логических элементов. 3) Interface – интерфейс связи цифрового устройства с внешним окружением создает контакты для воздействия на изделие внешними функциональными или тестовыми наборами, а также для снятия двоичной информации с внешних наблюдаемых выходов схемы. 4) Run-Step – функциональные сервисы моделирования входных воздействий в режимах: пошаговое моделирование теста путем ручного задания или ввода двоичного входного набора; автоматическое моделирование исчерпывающего теста на всех $2^{*}n$ входных последовательностях. 5) Visual – сервис визуализации схемы цифрового устройства в составе: логических элементов; соединительных линий между элементами, внешними входами и выходами; визуализация внешних входных и выходных контактов; визуализация состояний логических элементов на одном входном наборе и на полном тесте; визуализация вектора моделирования на одном тестовом наборе и на полном тесте; визуализация нумерации логических элементов, входных и выходных портов, а также десятичных кодов, соответствующих двоичным кубитным векторам логических примитивов. 6) Infrastructure – инфраструктурные сервисы для обслуживания основных функций, такие как: сохранение файла со схемным описанием цифрового устройства; коррекция схемного описания путем удаления-рисования структурных компонентов, кубитных покрытий и межэлементных связей. 7) Optimizer – оптимизация визуального размещения схемных компонентов и связей путем применения волнового алгоритма и ограничений, связанных с минимальным расстоянием между линиями и элементами. 8) Fault Simulation – моделирование неисправностей цифрового устройства на основе использования кубитных покрытий логических элементов и булевых производных, дающих возможность одновременной обработки списков неисправностей по дедуктивному алгоритму. 9) Test Synthesis – синтез тестовых наборов для проверки одиночных константных неисправностей путем взятия булевых производных по кубитному покрытию для всех входных переменных цифрового устройства на основе встречного сдвига и логического суммирования симметрич-

ных соседних частей кубитного вектора. Screen-Shot экрана, который визуализирует интерфейс сервиса «Quantum Modeling», представлен на рис. 9.

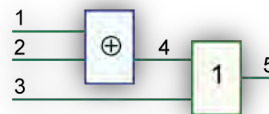


Рис. 9. Визуализация экрана с сервисом Quantum Modeling

Следующая схема (рис. 10) содержит данные о моделировании полного теста, содержащего 8 входных наборов, с помощью сервисной процедуры RUN.

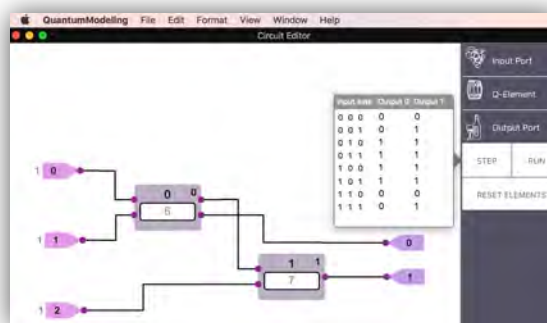


Рис. 10. Моделирование схемы на полном тесте

Для моделирования цифровой схемы вводятся кубитные покрытия в десятичном виде, что достаточно удобно для простейших логических элементов (рис 11). Чтобы наблюдать состояния выходных линий, можно воспользоваться соединением внутренних линий с выходными значками интерфейса. Это также дает возможность автоматически формировать таблицу моделирования на полном проверяющем тесте, содержащем $2^{*}n$ входных последовательностей.

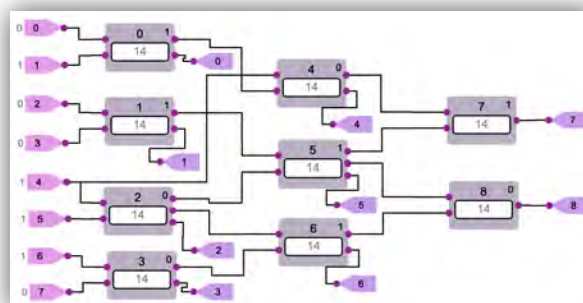


Рис. 11. Моделирование схемной структуры на тест-векторе

Реализация Хассе-процессора (рис. 12) для решения задачи покрытия содержит логические элементы, которые содержат одноходовые примитивы, двухходовые, трехходовые и один

четырёхходовой элемент. Практически за один автоматный такт осуществляется поиск оптимального покрытия, если элементы реализуют регистровые переменные по входам и выходам. Состояние ближайшего к внешним входам выхода логического элемента, равное единице, свидетельствует о наличии оптимального решения.

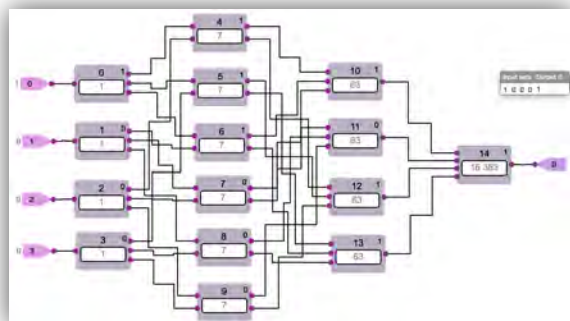


Рис. 12. Моделирование Хассе-структуры на тест-векторе

Таким образом, сервис логического моделирования способен продемонстрировать преимущества гибких кубитных структур данных и адресно-ориентированного анализа цифровых устройств, как альтернативной технологии для memory-driven квантового параллельного компьютеринга для решения задач синтеза и анализа.

Здесь следует также отметить необходимость введения в будущем: новой модели кубитных неисправностей, ориентированной на memory-driven компьютеринг; задания кубитного формата входных воздействий, аналогичного кубитным векторам логических элементов; синтеза кубитных моделей для цифровых автоматов, где уже не нужны триггеры для синтеза функций возбуждения; параллельного моделирования без наличия адресов памяти; моделирования неисправностей, привязанных к кубиту.

10. Облачная имплементация сервиса “Quantum modeling”

Реализация квантового моделирования в качестве облачного сервиса имеет ряд преимуществ перед программным приложением: 1) Инвариантность к аппаратным вычислительным средствам как к быстро устаревающей основе, на которой функционирует сервис. 2) Отсутствие необходимости в покупке аппаратных средств для реализации программного приложения как сервиса. 3) Низкий уровень финансовых затрат для аренды облачного пространства, как платформы для реализации компьютерингового сервиса. 4) Инвариантность места нахождения разработчика (-ов) для быстрой имплементации облачного сервиса. 5) Инвариантность облачного сервиса к географическим координатам всех

пользователей, находящихся на планете. 6) Доступность облачного сервиса с любой точки земного шара во времени и пространстве. 7) Высокий уровень надежности дата-центров, формирующих облачно-ориентированные платформы от компаний: Google, Amazon, Microsoft. 8) Высокий уровень software-driven кибербезопасности облачных сервисов, предоставляемых упомянутыми платформами. 9) Быстрая технологическая масштабируемость облачных сервисов, ориентированная на расширение функциональных возможностей и большое количество реальных пользователей.

Конкретно, облачный сервис “Quantum modeling” реализован на платформе Google, которая обеспечивает средства для реализации инфраструктурных микросервисов: хранения данных и доступа к ним, киберзащиты облачной функциональности и аутентификации пользователей. Инфраструктурные микросервисы от Google платформы существенно влияют на производительность и снижение времени проектирования облачных сервисов, а также на технологические удобства поддержания работающего сервиса в актуальном режиме функционирования.

Облачный сервис “Quantum modeling” представлен доменно и IP-идентифицируемой структурой киберфизических компонентов, реализующих компьютеринг-сервис моделирования цифровых устройств в контейнерно-ориентированной среде Docker, работающей в Google Computing Engine.

Цель сервиса связана с популяризацией новых технологий квантового синтеза и анализа цифровых устройств в среде студентов, ученых и специалистов, благодаря упомянутым выше преимуществам облачного компьютеринга.

Задачи: 1) Создание облачных микросервисов квантового проектирования, моделирования, диагностирования и верификации широкого класса цифровых устройств. 2) Популяризация кубитных структур данных и квантовых технологий проектирования memory-driven вычислительных устройств.

Современный рынок проектирования облачных систем показывает, что в последние три года набирают популярность гибкие и масштабируемые технологии проектирования сервисов, использующих контейнеризацию, направленную на минимизацию time-to-market. Основная идея контейнера заключается в создании защищенной и доступной виртуальной среды проектирования и эксплуатации в киберпространстве на основе использования локальной или глобальной платформы (рис. 13).

На инфраструктуру платформы устанавливается практически любая операционная система Host

operating system, удобная для разработчика, например, Ubuntu, Debian. В локально исполненной hardware инфраструктуре контейнеров можно установить также OSX и Windows.

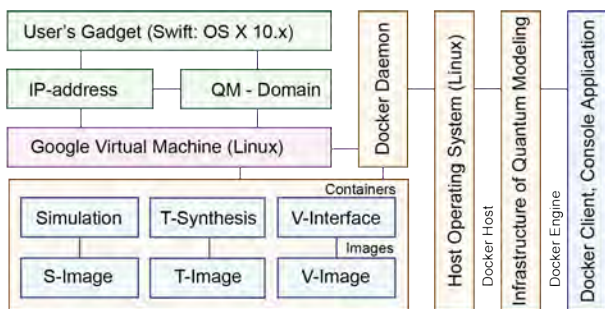


Рис. 13. Инфраструктура сервиса моделирования

Разработчик устанавливает Docker Engine, если он не установлен изначально, который с помощью Docker Daemon (реализация на Golang, C) управляет контейнерами, запущенными на основе предварительно построенных образов (Images). Контейнеры Containers представляют собой легковесную виртуальную вычислительную машину, под которой работают функциональные блоки: Modeling, Simulation, Test Synthesis, Fault Simulation, Fault Diagnosis, которые реализованы на языке Swift (Jawa) и имеют соответствующие библиотеки. Docker Engine представляет собой компьютерную систему облачной контейнеризации, состоящую из двух взаимодействующих между собой компонентов: Docker Host, Docker Client (Console Application), которые можно интерпретировать в качестве исполнительного Host и управляющего (Client) механизма облачной системы Quantum Modeling. Далее Docker Engine является компонентом, входящим в состав Google Virtual Machine, которой ставится в соответствие IP-address и Quantum Modeling domain, что делает разработку, а далее облачный сервис, доступным для широкого круга специалистов и студентов.

Последним звеном в вертикальной связке (Back-End, Front-End) является User's Gadget (Swift OS X.10x). На конечном устройстве пользователя устанавливаются Front-End приложения, способные привести результаты работы облачных сервисов моделирования к мультиоконному формату визуализации данных, удобных для восприятия человеком. Следует отметить, что намного технологичнее создать Front-End в виде приложения на JavaScript, HTML, CSS, которое будет работать через Internet Browser. Это гарантирует отсутствие блокирования клиента со стороны поставщика конкретной ОС (vendor lock). Для мобильных устройств целесообразнее использовать приложение индивидуального клиента под

каждую ОС. Все модули облачного сервиса запрограммированы на языке Swift, операционная система OSX 10.9, компилятор XCode 7. Количество исходных файлов 42, общее число строк кода – 2300.

Опытная эксплуатация Quantum Modeling сервиса показала: 1) Слушатели достаточно легко переходят на кубитный формат описания логических элементов в цифровой структуре. 2) Существенно уменьшается время проектирования цифровых устройств при использовании удобного графического интерфейса. 3) Значительно проще решаются все задачи, связанные с тестированием, моделированием и диагностированием цифровых компонентов и схем при применении кубитных структур данных и квантовых методов синтеза и анализа.

Для получения статистической информации и верификации Quantum Modeling было проведено 10 экспериментов на логических схемах из библиотеки ISCAS, а также на других структурах, приведенных ниже: 1) Adder SP. 2) Circuit Schneider. 3) Circuit C5. 4) Circuit C17. 5) RFO Circuit. 6) MUX16 Circuit. 7) DFA Circuit. 8) Hasse processor. 9) DC4-16 Circuit. 10) Circuit C432. Сравнению подлежало время ввода схемной структуры (Modeling Time), а также время проектирования диагностической информации (Designing Time), которая включает: ввод модели устройства, синтез и анализ тестов для получения таблицы неисправностей. Базовым вариантом для сравнения послужил продукт Active HDL, Aldec Inc., где информация о модели схемы вводилась на VHDL языке описания аппаратуры. Статистика сравнения двух времен: Modeling Time и Designing Time представлена на рис. 14 и 15 соответственно.

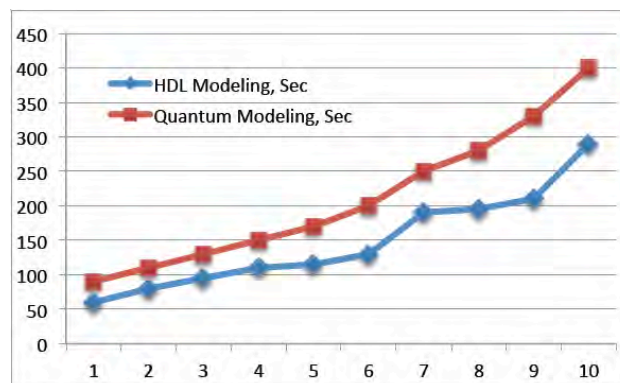


Рис. 14. Сравнительный анализ времени ввода логических схем

Она показывает преимущества визуального образного схемотехнического проектирования логических схем небольшой размерности по сравнению с вводом схем на основе HDL-описания. Особенно это приемлемо и эргономично в про-

цессе обучения студентов технологиям проектирования и верификации цифровых систем и компонентов. Однако для больших промышленных проектов пока еще целесообразно использовать языки описания аппаратуры.

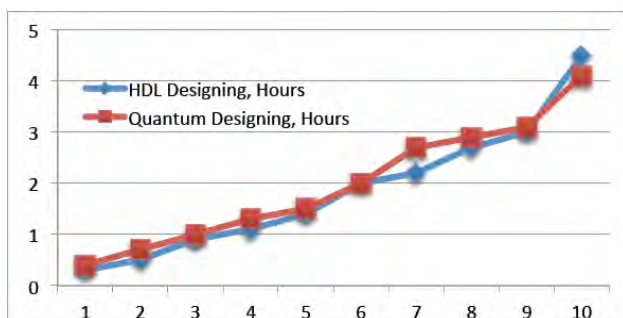


Рис. 15. Сравнительный анализ времени проектирования логических схем

Выводы. Становление новой теории проходит через ее первоначальное отторжение традиционно ориентированной научной общественностью. Тем не менее, очевидные отдельные преимущества практически направленного квантового компьютеринга находят отражение в публикациях ученых, работающих в областях, связанных с параллельными memory-driven вычислениями и комбинаторикой задач защиты информации, верификации и диагностирования. Нет лучшего способа продвинуть инновационную теорию путем использования открытого киберфизического пространства, где создаются облачные сервисы, покрывающие интересы широкой научной общественности во времени и пространстве. Поэтому соединение квантового компьютеринга с его облачным исполнением является удачной технологической связкой для решения задач киберфизического формата, ориентированного на проектирование и тестирование цифровых систем и программно-аппаратных компонентов. В рамках обозначенного научно-технологического направления были созданы следующие средства: 1) Графический интерфейс, удобный для ручного проектирования кубитных моделей цифровых устройств и компонентов, который дает возможность в режиме online осуществлять коррекцию ошибок. При высоком быстродействии современного компьютеринга главным фактором, влияющим на time-to-market, становится время ручного ввода моделей цифровых систем и компонентов. 2) Структуры данных для кубитного описания цифровых устройств и компонентов, которые отличаются компактностью и высоким параллелизмом их обработки. 3) Инфраструктура для проектирования и тестирования цифровых устройств и компонентов с функциями хранения, удаления и коррекции данных, что дает возможность осуществлять одновременное создание

цифровых схем несколькими проектировщиками. 4) Программные модули для квантового моделирования цифровых устройств и компонентов в режимах ручного и автоматического ввода входных тестовых последовательностей, что дает возможность наглядного обучения студентов методам синтеза и анализа. 5) Программные средства для синтеза таблиц неисправностей на основе дедуктивного метода квантового моделирования дефектов путем использования кубитных структур данных, которые обеспечивают поиск одиночных и кратных константных дефектов в режиме реального времени. 6) Направления будущих исследований связаны с созданием memory-driven моделей неисправностей, ориентированных на кубитные формы описания функциональностей с последующей разработкой семейства memory-driven параллельных методов квантового тестирования, моделирования и диагностирования компьютеринговых систем и компонентов.

11. Заключение

1. Кибертенденции от Gartner Inc. предоставляют возможность лидерам корпоративной архитектуры и руководителям университетов не отставать безнадежно от цифрового бизнеса в науке, образовании и индустрии, своевременно реагировать на киберфизические угрозы, возглавлять бизнес-инновации и определять эффективную цифровую бизнес-стратегию устойчивого развития государств.
2. По факту Hype-cycle представляет собой глубокую 4D-аналитику, во времени и в пространстве, состояния современного рынка устойчивого киберфизического развития умных облачных технологий на ближайшие 10-15 лет.
3. Для университетов Hype-цикл определяет жизненную необходимость инвестировать в знания студентов инновационные технологии, показанные в фазах цикла, в целях получения через 5-10 лет армии креативных специалистов, способных поднять государство из руин современного киберневежества. Иначе, Gartner цикл для университета представляет собой стратегию его киберфизического устойчивого развития во времени и пространстве. Любая стратегия, разработанная без знания темпов и направления технологических изменений, будет страдать неправильным планированием действий, разрушением бизнеса, науки и образования. Например, следует учитывать, что в 2018 году gobobossy будут точно мониторить и дистанционно online управлять 3 миллионами работников в мире с целью: метрически оценивать потенциал исполнителей, раздавать задания, логистически верно маршрутизировать их успешное выполнение, инвариантное к позиционированию работника в физическом

пространстве, оценивать качество и производительность труда, начислять заработную плату по метрическим результатам.

4. Нуре-цикл неявно дифференцирует все топ-технологии на ведущие и ведомые (master-slave), которые по факту означают, что развитию HardWare (Physical Space) платформ в сторону компактности всегда отдается приоритет, поскольку остальной виртуальный мир (Cyber Space), стремящийся к безграничному расширению SoftWare приложений, всегда будет ведомым. Взаимодействие двух миров, связанных с устойчивым развитием объемов аппаратного и программного обеспечения, формирующего киберфизическое пространство, представлено на рис. 16.

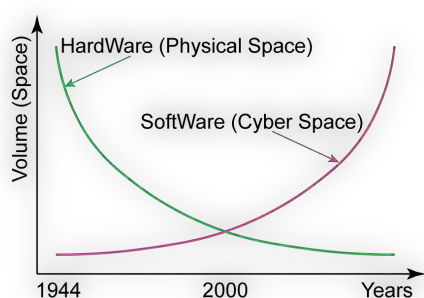


Рис. 16. Взаимодействие объемов киберфизических компонентов

5. Тем не менее, аппаратные и программные технологии представлены в Нуре-цикле (на рынке) практически в одинаковых пропорциях (50:50): Hardware-driven technologies: 4D Printing, Volumetric Displays, Nanotube Electronics, Brain-Computer Interface, Human Augmentation, Autonomous Vehicles, Cognitive Computing, Commercial UAVs (Drones), Smart Dust, Smart Robots, Smart Workspace, Connected Home, 5G, IoT Platform, Edge Computing, Neuromorphic Hardware, Quantum Computing;

Software-driven technologies: Deep Learning, Deep Reinforcement Learning, Artificial General Intelligence, Enterprise Taxonomy, Ontology Management, Machine Learning, Virtual Assistants, Cognitive Expert Advisors, Digital Twin, Blockchain, Serverless PaaS, Software-Defined Security, Virtual Reality, Augmented Reality, Augmented Data Discovery, Conversational User Interfaces, Digital Humanity, Smart Cyber Digital State.

6. Одинаковое соотношение аппаратных и программных технологий в Gartner-прогнозе означает, что уровни их капитализации на NASDAQ-рынке стремятся к паритету, ярким примером которого являются компании Apple (800 млрд долларов – индекс NASDAQ 2017) и Google (570 млрд). Эти производители существенно отличаются тем, что они полагаются на мудрость своих команд (экспертов), вооруженных доктриной:

«потребители не могут предсказывать свои собственные потребности» (consumers could not predict their own needs) [9]. Альтернативой является политика компании Microsoft (503 млрд), которая проводит обширные исследования перед запуском продукта, например, такого как Windows Phone. По оценкам Gartner доля Apple на мировом рынке мобильных телефонов составляет 14,2% против 3,3% для Microsoft. Кому доверять, экспертам или потребителям? Ответ однозначный – экспертам, в формате 4D (всегда, везде и по всем вопросам).

7. Представлена memory-driven инновационная архитектура квантового компьютеринга, которая определяется возможностью устранения логики, связанной с суперпозицией и перепутыванием состояний на основе использования характеристического уравнения, реализующего транзакции записи-считывания на структуре электронов. Исключение логических операций из квантового компьютеринга позволяет существенно упростить архитектуру до уровня структуры памяти на электронах для выполнения транзакций между ними с помощью квантов или фотонов. Показано формальное отличие квантового компьютеринга от классического, которое заключается в возможности параллельного и одновременного выполнения логических операций над множествами.

8. Предложен квантовый метод минимизации булевых функций, который отличается от метода неопределенных коэффициентов параллельным выполнением операции суперпозиции над нулевыми и единичными состояниями входных переменных, представленными унитарными кодами, что дает возможность существенно повысить быстродействие за счет избыточной памяти.

9. Предложен кубитный метод безусловного диагностирования неисправностей цифровых устройств, который отличается от аналогов параллельным выполнением логических операций, что дает возможность существенно повысить быстродействие поиска дефектов за счет унитарного кодирования данных.

10. Представлены кубитные структуры данных, облачная реализация методов и моделей анализа, что дает возможность изучать квантовые технологии тестирования и моделирования цифровых систем и компонентов в режиме online.

Литература:

1. <https://www.forbes.com/sites/louiscolombus/2017/08/15/gartners-hype-cycle-for-emerging-technologies-2017-adds-5g-and-deep-learning-for-first-time/#646a4cf34be2>
2. <http://www.gartner.com/newsroom/id/3784363>
3. <http://www.wired.co.uk/article/ai-neuromorphic-chips-brains>

4. A. Gupta and R. K. Jha, "A Survey of 5G Network: Architecture and Emerging Technologies," in IEEE Access, vol. 3. P. 1206-1232, 2015.
5. C. Zhu, V. C. M. Leung, L. Shu and E. C. H. Ngai, "Green Internet of Things for Smart World," in IEEE Access, vol. 3. P. 2151-2162, 2015.
6. K. Christidis and M. Devetsikiotis, "Blockchains and Smart Contracts for the Internet of Things," in IEEE Access, vol. 4. P. 2292-2303, 2016.
7. *Blockchains: How They Work and Why They'll Change the World* IEEE Spectrum. October 2017. <https://spectrum.ieee.org/computing/networks/blockchain-s-how-they-work-and-why-theyll-change-the-world>
8. A. Zanella, N. Bui, A. Castellani, L. Vangelista and M. Zorzi, "Internet of Things for Smart Cities," in IEEE IoT Journal, vol. 1, no. 1. P. 22-32, Feb. 2014.
9. https://www.gartner.com/doc/3471559?srcId=1-7578984202&utm_campaign=RM_GB_2017_TRENDS_QC_E2_What&utm_medium=email&utm_source=Eloqua&cm_mmc=Eloqua_-Email_-LM_RM_GB_2017_TRENDS_QC_E2_What_-0000
10. <http://www.gartner.com/smarterwithgartner/three-digital-marketing-habits-to-break-2/>
11. Vladimir Hahanov. *Cyber Physical Computing for IoT-driven Services*. New York. Springer. 2017. 243p.
12. Hahanov V.I., Bani Amer Tamer, Chumachenko S.V., Litvinova E.I. (2015) Qubit technology for analysis and diagnosis of digital devices. *Electronic modeling*. J 37 (3): 17-40.
13. Vladimir Hahanov, Wajeb Gharibi, Eugenia Litvinova, Mykhailo Liubarskyi, Anastasia Hahanova. Quantum memory-driven computing for test synthesis. *IEEE East-West Design and Test Symposium*. 2017. Novi Sad, Serbia. P. 123-128.
14. Tian Zhong1, Jonathan M. Kindem1, John G. Bartholomew1 at all. Nanophotonic rare-earth quantum memory with optically controlled retrieval. *Science*. 29 Sep 2017: Vol. 357, Issue 6358, pp. 1392-1395.
15. Jungkil Kim, Hoo-Cheol Lee, Kyoung-Ho Kim at all. Photon-triggered nanowire transistors. *Nature Nanotechnology* 12. P. 963–968 (2017).
16. Giacomo Lovat, Bonnie Choi, Daniel W. Paley at all. Room-temperature current blockade in atomically defined single-cluster junctions. *Nature Nanotechnology*, doi:10.1038/nnano.2017.156.
17. Chao Li, Zhongping Wang, Yan Lu, Xiaoqing Liu & Li Wang. Conformation-based signal transfer and processing at the single-molecule level. *Nature Nanotechnology*. (2017), doi:10.1038/nnano.2017. 179.
18. Patrick T. Hester, Kevin Adams. *Systemic Decision Making. Fundamentals for Addressing Problems and Messes*. Springer Nature. 412 p.
19. Hahanov Vladimir, Barkalov Alexander, Adamski Marian. *Infrastructure intellectual property for SoC simulation and diagnosis service*. Springer, Germany. 2011.– P. 289-330.
20. <http://dpchas.com.ua/politika/dubilet-zadumal-revolyuciyu>
21. Савельев А.Я. *Прикладная теория цифровых автоматов*. М.: Высш. шк., 1987. 272 с.
8. V. Kharchenko, Y. Kondratenko, J. Kacprzyk (Eds.), "Green IT Engineering: Concepts, Models, Complex Systems Architectures", In the book series "Studies in Systems, Decision and Control" (SSDC), vol. 1, Berlin, Heidelberg: Springer International Publishing, 2017.
9. V. Kharchenko, Y. Kondratenko, J. Kacprzyk (Eds.), "Green IT Engineering: Components, Networks and Systems Implementation". In the book series "Studies in Systems, Decision and Control" (SSDC), vol. 2, Berlin, Heidelberg: Springer International Publishing, 2017.

Поступила в редколлегию 12.12.2017

Рецензент: д-р техн. наук, проф. Кривуля Г.Ф.

Емельянов Игорь Валериевич, научный сотрудник кафедры АПВТ ХНУРЭ. Научные интересы: проектирование и тестирование цифровых систем. Хобби: путешествия. Адрес: Украина, 61166, Харьков, пр. Науки, 14, e-mail: iyemelyanov@itdelight.com.

Любарский Михаил Михайлович, соискатель кафедры АПВТ ХНУРЭ. Научные интересы: проектирование и тестирование цифровых систем. Хобби: путешествия. Адрес: Украина, 61166, Харьков, пр. Науки, 14.

Хаханов Владимир Иванович, д-р техн. наук, проф., главный научный сотрудник кафедры АПВТ ХНУРЭ. Научные интересы: проектирование и тестирование цифровых систем. Хобби: футбол, горные лыжи. Адрес: Украина, 61166, Харьков, пр. Науки, 14, e-mail: hahanov@icloud.com.

Iemelianov Igor Valerievich, Research Associate, Design Automation Department, NURE. Scientific interests: design and testing of digital systems. Hobbies: traveling. Address: Ukraine, 61166, Kharkov, Nauki Ave, 14, e-mail: iyemelyanov@itdelight.com.

Lyubarsky Mikhail Mikhailovich, PhD student, Design Automation Department, NURE. Scientific interests: project-bathing and testing digital systems. Scientific interests: design and testing of digital systems. Hobbies: traveling. Address: Ukraine, 61166, Kharkov, Nauki Ave, 14.

Hahanov Vladimir Ivanovich, Dr. of Tech. Sc., Prof., Chief Scientific Officer, Design Automation Department, NURE. Scientific interests: design and testing of digital systems. Hobby: football, downhill skiing. Address: Ukraine, 61166, Kharkov, Science, 14, e-mail: hahanov@icloud.com.