

**ОСНОВЫ
ИНФОРМАЦИОННЫХ
ТЕХНОЛОГИЙ**

**ВЫЧИСЛИТЕЛЬНЫЕ
НАНОСТРУКТУРЫ
I
ЗАДАЧИ, МОДЕЛИ, СТРУКТУРЫ**



Основы информационных технологий

ВЫЧИСЛИТЕЛЬНЫЕ НАНОСТРУКТУРЫ

I Задачи, модели, структуры

Под редакцией д.т.н., профессора Г.М. Алакоза

Учебное пособие



Интернет-Университет
Информационных Технологий
www.intuit.ru



БИНОМ.
Лаборатория знаний
www.lbz.ru

Москва
2010

Оглавление

Введение	8
Лекция 1. Междисциплинарный характер компьютерных проектов ...	23
1.1. Научно-технические и социально-экономические предпосылки зарождения вычислительных технологий	23
1.2. Базовые положения современной проектной технологии прототипирования	36
1.3. Глобальные технотронные комплексы как источник конфликт-индуцированных опасностей	42
1.4. Глобальные военные и национальные программы как средство консолидации финансовых, экономических, интеллектуальных и технических ресурсов общества для решения задач прорывных научных исследований	65
1.5. Интенсивные и экстенсивные факторы повышения пропускной способности вычислительных систем по потокам команд и данных	81
1.6. Обобщенная схема организации вычислений	88
1.7. Взаимовлияние физико-технических, схемотехнических факторов в процессе развития средств вычислительной техники	92
Системотехнические выводы по лекции 1	97
Лекция 2. Состояние и перспективы развития вычислительных систем и проектных технологий их создания	99
2.1. Основные определения и понятия системотехники	99
2.2. Классификация и основные характеристики современных ЭВМ	103
2.3. Концептуальное развитие программ создания перспективного бортового электронного оборудования (БЭО) в США	110
2.4. Обобщенная архитектура БЭО и элементная база БВС перспективных самолетов пятого поколения F-22	116
2.5. Особенности разработки БЭО самолета F-35	119
Системотехнические выводы по лекции 2	126
Лекция 3. Теория вычислений и машины Тьюринга	129
3.1. Интуитивные предпосылки формализации понятия «алгоритм»	129
3.2. Машины Тьюринга и вычислимые функции	140
3.3. Машины Тьюринга и перечислимые множества	144

3.4.	Машины Тьюринга и «количество информации»	147
	Системотехнические выводы по лекции 3	153
Лекция 4.	Нейрофизиологический и формально-логический базис нейроподобных вычислений	156
4.1	Системотехнический анализ морфофункциональных схем нейронов, нервных сетей и механизмов «спайковой» активности	156
4.2.	Базовые положения теории функциональных систем	163
4.3.	Формальная модель нейрона и сети Мак-Каллока — Питтса	170
4.4.	Перцептрон Ф. Розенблатта и теория распознавания образов.	179
4.5.	Многопороговые модели	192
4.6.	Входное преобразование формального нейрона как генератор кратных транспозиций в лексикографически упорядоченной последовательности значений свертки . . .	198
4.7.	Теоретико-групповая модель формального нейрона как операционное ядро компиляции на квантовый уровень организации вычислений.	214
	Системотехнические выводы по лекции 4	227
Лекция 5.	Базовые положения теории многофункциональных логических модулей	229
5.1.	Методы структурно-параметрической адаптации многофункциональных логических модулей	229
5.2.	Каноническая система преобразований многофункциональных логических модулей	237
5.3.	Структурно-функциональная избыточность многофункциональных логических модулей и формальных нейронов	242
5.4.	Нейроподобная модель универсальных дискретных модулей с ассоциативным управлением по «сходству» и по «отличию»	249
5.5.	PD-ассоциативные конструкции и дуализм между потоками инструкций и данных	255
	Системотехнические выводы по лекции 5	265
Лекция 6.	Современные вычислительные технологии и их программно-аппаратные платформы	268
6.1.	Особенности технологий, основанных на (сверх)массовом параллелизме	268
6.2.	RISC-архитектуры как компромисс между системотехническими требованиями и технологическими возможностями микроэлектроники . . .	275

6.3.	Особенности нейрокомпьютерных технологий	279
6.4.	Роль и место нейроподобных вычислительных систем в технотронных комплексах двойного назначения	285
6.5.	Принципы и методы МКМД-бит-потокковой организации вычислений.	293
6.6.	Факторы, ограничивающие коэффициент распараллеливания вычислений.	297
	Системотехнические выводы по лекции 6	304
Лекция 7.	Системотехнические аспекты перспективных вычислительных технологий	307
7.1.	Основные направления развития наноэлектроники	307
7.2.	Базовая модель алгоритмически ориентированных супрамолекулярных вычислителей	319
7.3.	Базовые модели квантовых компьютеров.	324
7.4.	Особенности нанометровой системотехники	344
7.5.	Особенности нанометровой схемотехники	352
7.6.	Особенности нейроподобной схемотехники	382
	Системотехнические выводы по лекции 7	394
Лекция 8.	Нейроподобные модели как формально-логический базис анализа живых систем	399
8.1	Самоорганизация и инструктированный синтез себе подобных	399
8.2.	Многопороговая модель биологического кода	404
8.3.	Криптостойкость как синоним специфичности молекулярно-биологических комплексов.	417
8.4.	PD-ассоциативные вычислительные конструкции как основа структурно-параметрического метода хранения и преобразования информации	427
8.5.	Фундаментальные константы как источник неоднозначной идентификации предметов и явлений природы	436
8.6.	«Нечисленная» идентификация как средство повышения сложности живых систем	449
	Системотехнические выводы по лекции 8	463
	Список использованных сокращений	467
	Литература	470

Введение

Термин «*технология*» традиционно ассоциируется с операционной картой, регламентирующей действия персонала с оборудованием и оснасткой в процессе производства *материальных* продуктов. Словосочетание «информационные технологии» указывает на то, что объектом производства является нематериальный продукт, «карта» которого ограничена тремя типами операций: *хранение, передача и преобразование* информации, включая ее *кодирование*. Поэтому вторая специфическая особенность информационных технологий состоит в том, что в них «информация» выступает и как исходное сырье, и как конечный продукт. В результате в информационных технологиях весьма спорными остаются ответы на вопросы о *качестве и количестве* «произведенного продукта», что требует привлечения не только алгоритмических, вероятностных или статистических, но и семантических и/или прагматических методов и средств оценки.

Тем не менее, термин «*информационные технологии*» прочно укрепился в науке и обществе, и к нему прибегают, когда требуется описать *хранение, передачу и преобразование* информации и используемые при этом вычислительные ресурсы ЭВМ: память, операционные, интерфейсные, адресные, коммутационные и управляющие устройства.

В научных и прикладных разработках информационных технологий можно выделить два взаимно обусловленных направления, одно из которых занимается созданием аппаратных средств и инструментальных платформ их использования, а другое — собственно применением вычислительной техники в конкретных прикладных областях деятельности человека. На начальном этапе развития кибернетики и вычислительной техники доминировало первое направление исследований, но уже с середины 80-х годов прошлого столетия вопросы информационных технологий «собственных нужд» стали уделом «элитарного круга» специалистов высокоразвитых стран. Этот круг был очерчен во многом благодаря успехам *кремниевой компиляции*, которые привели к повышению качества сквозного, «однопроходного», бездефектного проектирования СБИС и, как следствие, к сокращению сроков разработки и повышению процента выхода годных устройств, что резко удешевило радиоэлектронную и вычислительную аппаратуру, сделав ее массово доступной.

Однако невиданная по масштабам коммерциализация информационных технологий не обошлась без жертв и уже к середине 90-х годов прошлого столетия привела к гипертрофированному дисбалансу между аппаратно-временными затратами, расходуемыми на системные и пользовательские функции. В результате системные функции, связанные с *управлением вычислительными ресурсами* непосредственно во время вычис-

лений, и особенно так называемый «дружественный интерфейс» стали основными «пожирателями» этих же ресурсов. В таких условиях оказались несостоятельными устоявшиеся оценки эффективности работы вычислительной техники, прямо или контекстно учитывающие некоторый коэффициент полезного действия.

Ситуация в этой области еще больше усложнилась, когда в тот же период времени были развернуты ширококомасштабные *прикладные исследования* элементной базы нанометрового и супрамолекулярного диапазона, где на первое место вышли вопросы сохранения условий правильной работы (когерентности) на интервалах времени и в диапазонах изменения внешних условий, сопоставимых с возможностями современной (суб)микронной элементной базы.

При этом задним числом обратили внимание на тот факт, что при переходе от «навесной» электроники к интегральной микроэлектронике перераспределились функции в «иерархии» разработчиков средств вычислительной техники, что неминуемо привело к изменению показателей оценки качества создаваемых изделий. Действительно, в «навесной» электронике состав элементной базы, узлов и блоков диктовался разработчиками ЭВМ, а схема считалась тем лучше, чем меньше в ней активных элементов. В микроэлектронике состав блоков и устройств процессора стал определяться разработчиком СБИС, а лучшей стала считаться схема, которая занимает меньшую площадь и по максимуму использует «короткие» однородные связи. В итоге все без исключения производители микропроцессоров безоговорочно стали использовать внутренние микроконвейеры, а в процессоры RISC-архитектуры стали включать только те операционные блоки, а с ними и *ассемблерные инструкции*, которые удовлетворяют условиям *эффективной микроэлектронной* реализации.

Одновременно с переходом к микроэлектронным изделиям ультрабольшой (свыше 1 млн. транзисторов) степени интеграции даже в массово доступных процессорах развернулась борьба за повышение не только тактовой частоты, но и *функциональной насыщенности* каждого такта за счет векторно-конвейерного распараллеливания вычислений как на одном кристалле, так и в вычислительной системе в целом.

До этого методы и средства распараллеливания вычислений исследовались и использовались в рамках вычислительных систем, ориентированных на решение задач с «*критическими*» требованиями к производительности, пропускной способности, точности и отказоустойчивости, которые не менее чем на порядок превосходили физические возможности аппаратуры. Такие задачи ставились и решались в основном разработчиками систем управления в реальном времени локальными и глобальными военными комплексами, объектами атомной энергетики, опасными химическими и биологическими производствами и т. п.

Военно-политическая составляющая таких задач объективно возлагала на военные бюджеты высокоразвитых стран *консолидирующую роль* в создании и продвижении на рынок новых информационных технологий и программно-аппаратных платформ их поддержки. Наиболее четко эта тенденция проявилась в США и СССР как главных участниках гонки вооружений. Показательно, что «кремниевая долина» США вышла на самокупаемость только в начале 90-х годов прошлого столетия, а в основе ее технического перевооружения даже в субмикронной области по-прежнему лежат миллиардные военные контракты.

СССР в силу целого ряда объективных и субъективных причин отставал по технологическому уровню микроэлектронного производства от США. Поэтому в развитии информационных технологий «собственных нужд» он вынужден был делать основную ставку на распараллеливание вычислений, которое невозможно было реализовать без глубоких знаний предметной области, а значит, и без теснейшего взаимодействия с информационными технологиями, отработанными специалистами предметной области.

В итоге такого взаимодействия к середине 80-х годов прошлого столетия у нас в стране сформировалась бит-поточковая вычислительная технология сквозного системного проектирования проблемно- и алгоритмически ориентированных субпроцессорных трактов, в рамках которой:

- комплексно решались все вопросы от выбора физико-технических процессов работы элементной базы и до потоковых алгоритмов обработки данных;
- разработчик прикладных задач получил доступ к микрокомандному уровню управления (сверх)большим коллективом элементарных вычислителей, что обеспечило достижение предельно возможных (по алгоритму и аппаратуре) коэффициентов векторно-конвейерного распараллеливания и требуемой для этого пропускной способности по потокам данных;
- заданные показатели отказоустойчивости аппаратуры оказались достижимы при некратном ее резервировании, а негативное влияние распараллеливания вычислений на вычислительную устойчивость алгоритмов компенсировалось высокой адаптивностью разрядной сетки при обработке и передаче промежуточных данных.

Теоретическую базу данной технологии составили работы по однородным вычислительным средам и структурам наших отечественных ученых Э.В. Евреинова, Ю.Т. Косарева, А.В. Каляева, И.В. Прангишвили и ряда других авторов, которые своими исследованиями предопределили необходимость объединения в одной вычислительной технологии сверхмассового микрокомандного параллелизма с потоковой и ассоциативной организацией вычислений. В связи с распадом СССР данная технология

не получила широкого распространения даже у нас в стране, но обнародованные результаты *натурных исследований* успели остудить горячие головы по обе стороны «железного занавеса», которые рвались к развертыванию ударных космических группировок, запланированному *не по нашей инициативе* на 1995 год.

Стратегический характер данной разработки, а также реальные ограничения на все виды материальных, финансовых, интеллектуальных и других видов ресурсов вынудили проводить ее с учетом 40-летнего прогноза развития электроники и вычислительной техники. Простейший линейный прогноз с экспертным замедлением темпов снижения топологических норм микроэлектроники показал, что к 2015–2020 годам микроминиатюризация может достичь своих физических пределов. В результате *процессы производства и функционирования* элементной базы должны были оказаться в прямой зависимости от законов квантовой механики. Поэтому базовые положения и программные инструментальные платформы бит-поточковой технологии с микрокомандным уровнем доступа разрабатывались с учетом выхода аппаратных платформ на уровень прямых квантовых взаимодействий.

Прогноз оказался достаточно удачным, и человечество действительно стоит на рубеже освоения нанотехнологий, способных кардинально изменить его взаимоотношения с внешней средой и внутри себя. Учитывая реальность такой перспективы, авторы работы поставили перед собой задачу передать следующему поколению информационных технологий «собственных нужд» достигнутые в области бит-поточковых технологий результаты, сопоставив их с достижениями в других направлениях развития вычислительной техники.

Одним из адресатов книги является менеджерский корпус, от успешной работы которого зависит качество *однопроходного, бездефектного, сквозного системного проектирования и запуска* информационной технологии «собственных нужд». *Управляемость* такой разработки, а с ней и успех возглавляемого менеджерами проекта «собственных нужд» определяется уровнем их собственных знаний «болевыми» точек и законов их взаимодействия в рамках создаваемой технологии.

Сами «болевы» точки и особенно закономерности их взаимодействия наиболее четко проявляются при решении «критических» задач предметной области, удовлетворение требований которых вынуждает ставить во главу угла всего проекта забытую уже оценку соотношения между аппаратно-временными затратами, расходуемыми на реализацию функций пользователя и функций управления системой. В силу этих причин основу предметной области составили задачи аэрокосмических систем двойного назначения, от решения которых по объективным причинам невозможно отказаться ни в военном, ни в гражданском применении,

несмотря на запредельные и противоречивые требования, одновременно предъявляемые ими к производительности, точности и отказоустойчивости поддерживающих их вычислительных систем.

В лекции 1 показан *междисциплинарный характер* всех без исключения компьютерных проектов, начиная с работ основоположников кибернетики и вычислительной техники в лице Н. Винера и Дж. фон Неймана, которые интегрировали достижения науки и техники того периода в таких областях, как (нейро)физиология и психология, статистическая физика и фундаментальная математика, электротехника, техника связи, радиотехника и т. д. Показано, что междисциплинарный характер современных компьютерных проектов только усилился. Это наиболее четко проявляется при создании современных аэрокосмических летательных аппаратов, которые представляют собой сложные *технотронные комплексы*, — их эксплуатация и (боевое) применение в принципе невозможны без использования компьютерных технологий. Благодаря этой особенности именно в авиационно-космическом приборостроении США зародилась и продолжает развиваться проектная технология *прототипирования*, в рамках которой информационная технология «собственных нужд» адаптируется под требования технологии управления и эксплуатации создаваемого летательного аппарата, который по объективным причинам необходимо рассматривать как открытую систему. Открытость авиационно-космической техники проявляется в том, что за «время жизни» (порядка 10–20 лет) ее бортовое оборудование постоянно модифицируется как по составу, так и по показателям качества. Главная опасность такой модификации состоит в том, что на таком интервале времени может измениться не одно поколение разработчиков. В результате может потеряться «системотехнический контекст» всего проекта, так как согласно Дж. фон Нейману «...сложную систему проще сделать, чем описать...». Потеря такого контекста чревата возникновением конфликт-индуцированных ситуаций как между операторами технотронных комплексов, так и между оборудованием и оператором, что грозит потерей контроля над управляемыми объектами и процессами. Потенциальные источники таких опасностей показаны в разделе 1.3 и должны учитываться разработчиками технотронных комплексов как на этапе их создания, так и на этапах модификации. Опыт консолидации финансовых, экономических, интеллектуальных и технических ресурсов общества для решения задач прорывных научных исследований проанализирован в разделе 1.4 на примере американской программы, получившей название СОИ, и на примере японской программы ERATO, ориентированной на формирование менеджерского корпуса для прорывных научных и технологических проектов XXI века. В завершение лекции показана взаимосвязь

между комплексом *интенсивных* и *экстенсивных* физико-технических, схемотехнических факторов, определяющих развитие средств вычислительной техники.

В лекции 2 раскрыт эволюционный характер становления *технологии прототипирования* при создании боевых летательных аппаратов **F-22** и **F-35**, что, в частности, подтверждает консолидирующее влияние военного бюджета в прорывных проектных технологиях, которые невозможно реализовать только силами даже таких мощных корпораций как Boeing, McDonnell Douglas, Lockheed, Martin Marietta и т. д.

Необходимость освоения технологии прототипирования вызвана тем фактом, что развитие бортового электронного оборудования (БЭО) авиации США ориентировано на переход от *прямого* к *директивному*, а впоследствии и к *декларативному* управлению сложными технетронными комплексами, в рамках которого пилот задает только цели и режимы функционирования бортовых систем и всего летательного аппарата (ЛА). В таких условиях реализация поставленных целей и оптимизация режимов работы систем ЛА полностью возлагается на его бортовую вычислительную систему (БВС), которая и призвана решить «критические» задачи управления, вплоть до непрерывной оценки технического состояния ЛА и всего БЭО с использованием методов и средств искусственного интеллекта. Стратегический характер задач технологии прототипирования проявляется в стремлении высокоразвитых стран к завоеванию «экономического господства в воздухе», которое неминуемо должно привести к мировому распределению труда, а с ним и доходов от производства и эксплуатации ЛА, стоимостные характеристики которых уже сейчас приближаются к миллиарду долларов на 1 самолет. Механизм реализации этой стратегии закладывается в процедуры и нормы сертификации ЛА и их БЭО, которые лоббируются через международную организацию гражданской авиации ИКАО и которые кардинальным образом перераспределяют задачи обеспечения *безопасности полетов* между наземными и бортовыми комплексами в пользу последних (стратегия автономизации всех функций управления ЛА). В рамках такой стратегии получает мощный импульс развития беспилотная авиация (в основном военного назначения), которая призвана снизить собственные потери живой силы в локальных военных конфликтах со слаборазвитыми в техническом и экономическом отношениях странами. Показано также, что даже такая технологически развитая страна как США не способна удовлетворить требования «критических» задач управления ЛА на основе одной аппаратной платформы. Поэтому в БВС ЛА 5-го поколения планируется использовать **RISC**-процессоры, цифровые процессоры обработки сигналов и **FPGA**-матрицы, из которых только последние ориентированы на решение «критических» задач управления ЛА, связанных с обработкой

в реальном времени изображений и поддержкой механизмов «виртуальной реальности». (FPGA-матрица — это Field Programmable Gate Array, или программируемая пользователем вентильная матрица). В итоге можно утверждать, что целевая функция построения и работы БВС за последние 25–30 лет фактически не изменилась и разработчики по-прежнему ориентированы на создание открытых, многоуровневых, неоднородных, распределенных, (сверх)параллельных, защищенных от несанкционированного доступа БВС повышенной живучести.

В лекции 3 показана общность традиционных компьютерных технологий, в основе которых лежит теория алгоритмов и машины Тьюринга, с нейрокомпьютерными технологиями, для реализации которых достаточно «нечисленного» операционного базиса в виде «ассоциативной выборки» и «подстановки» (замещения) символов. При этом *первичность* нейрокомпьютерных технологий по отношению к компьютерным предопределяет сама логика формализации: от эмпирической таблицы к аналитическому или алгоритмическому представлению. Поэтому разница между компьютерными и нейрокомпьютерными технологиями в конечном счете состоит в уровне доступности пользователя к операционному базису. В нейрокомпьютерных технологиях пользователю доступны «элементарные операции» сетевого уровня управления. В традиционных вычислительных технологиях уровень «элементарных операций» (микрокоманд) доступен только производителю процессора — он формирует ассемблерные команды, которые для разработчика прикладных программ уже являются «неделимой единицей проекта» и поддерживаются практически независимыми микроэлектронными аппаратными блоками.

В итоге в нейрокомпьютерных технологиях любое задание можно представить тривиальной таблицей соответствия между входными воздействиями и требуемыми реакциями. В традиционных вычислительных технологиях требуется не только формализованное представление задания, но и алгоритм, регламентирующий последовательность выполнения ассемблерных команд. Показано также, что фундаментальной основой *экстенциональной* эквивалентности традиционных и нейроподобных ЭВМ служит сводимость вычислительных по Тьюрингу процедур к пересчитываемым по Тьюрингу процедурам, первые из которых составляют операционный базис ЭВМ одной из традиционных архитектур, а вторые — одной из нейрокомпьютерных архитектур. Поэтому имеются достаточные и формально доказанные основания утверждать, что любой нейро-ЭВМ можно поставить в соответствие классическую ЭВМ, эквивалентную по вычисляемой функции, и наоборот. (*Экстенциональная* эквивалентность означает эквивалентность по конечному эффекту, а не по процедуре его достижения.) Отталкиваясь от результатов А.Н. Колмогорова в области

алгоритмической меры информации, показано, что оценку алгоритмической сложности реальных (нейро)компьютерных реализаций можно провести только с определенной погрешностью и при достаточно жестких ограничениях на используемые вычислительные и/или перечислительные алгоритмы, что характерно для алгоритмически ориентированных областей применения традиционных вычислителей и «дочерних» нейро-ЭВМ, а не для машин общего назначения.

В лекции 4 проанализированы современные нейрохимические модели нейронов, которые исходят из тончайшего молекулярно-биологического взаимодействия внешнего субстрата с внутренними (метаболическими) процессами. Эти особенности работы реальных нейронов не могли быть отражены в электрофизиологической модели Ходжкина — Хаксли, положенной в основу ставших уже традиционными (много)пороговых моделей формальных нейронов. С этих позиций сети Мак-Каллока — Питтса и Ф. Розенблатта отличаются способами соединения и управления пороговыми элементами. Чтобы отразить известные уже нейрохимические особенности работы реальных нейронов в их формальных моделях, необходимо учесть последние достижения нейрофизиологии, среди которых можно выделить теорию конвергентного замыкания условного рефлекса отечественного нейрофизиолога академика П.К. Анохина. Она указывает на существование реальных «нейрокомпиляторов» с уровня межнейрональных взаимодействий, поддерживаемых *межклеточным супрамолекулярным нейросубстратом*, на уровень метаболических процессов, и наоборот. В результате можно предположить, что в реальной нервной системе «задание» доводится или по крайней мере может быть доведено до молекулярно-биологических взаимодействий, соподчиненных законам квантовой механики. Поэтому для решения центральной проблемы современной нано- и супрамолекулярной электроники, связанной с декогерентизацией квантовых систем, осмысленным является поиск методов и средств прямого отображения заданий предметной области на физико-химический уровень работы «элементной базы», что требует динамичного синтеза нанометровых и супрамолекулярных вычислительных структур с ограниченным «временем жизни».

Следуя по этому пути, удалось синтезировать теоретико-групповую модель формального нейрона, которая позволяет решать задачи оптимального синтеза нейро-ЭВМ на конечном множестве *целочисленных* вариаций весового и порогового векторов. Такая модель создает предпосылки для прямого отображения задач предметной области применения нейро-ЭВМ на квантовые процессы нанoeлектронных систем, фундаментальные свойства которых определяются и описываются группами симметрий. При этом сохраняется традиционная схема кремниевой ком-

пиляции с использованием библиотек стандартных элементов, которая и ограничивает совокупность используемых теоретико-групповых преобразований, отвечающих физико-химическому уровню описания работы нанометровой или супрамолекулярной элементной базы.

В лекции 5 показана принципиальная возможность использования в вычислительной технике упрощенных аналогов механизмов конвергентного замыкания. Для этого достаточно:

- признать тот факт, что многофункциональность «элементарных вычислителей» является не только прерогативой нейрокомпьютерных технологий и она может быть реализована методами и средствами традиционных компьютерных технологий;
- использовать микрокомандный уровень доступа и *естественный структурно-функциональный дуализм* между потоками инструкций и данных, который позволяет рассматривать мультиплексор и как коммутационный автомат, и как универсальный логический модуль, способный реализовать любую логическую функцию фиксированного числа переменных.

Структурно-функциональный дуализм предопределяет также и эффективность использования **PD**-ассоциативных методов управления, особенно в многопроцессорных вычислительных системах МКМД-типа, которые критичны к процедурам инициализации и распределения потоков инструкций. (МКМД — организация вычислений по типу «множественный поток инструкций — множественный поток данных».) В таких условиях **PD**-ассоциативное управление позволяет с помощью специально организованного потока данных и без обращения к внешней памяти *модифицировать в реальном времени инструкцию*, закрепленную за локальным вычислителем. **PD**-ассоциативное управление критично к информационным и аппаратным «сбоям», но оно адекватно условиям работы нанометровых и супрамолекулярных вычислителей, в которых «тирания» паразитных полимодальных квантовых взаимодействий способна изменить реализуемую функцию. В результате **PD**-ассоциативные конструкции позволяют комплексно использовать *структурно-параметрические методы и средства хранения и преобразования информации* во время динамического синтеза квантового «рабочего тела» проблемно-ориентированных нанометровых или супрамолекулярных вычислителей. В таких условиях физико-химический синтез становится *составной частью вычислительного процесса*, а предшествующая ему деструкция «рабочего тела» вычислителя-предка будет тем «глубже», чем выше размерность **PD**-ассоциативного управляющего вектора в вычислителе-потомке.

В лекции 6 сопоставлены возможности основных информационных технологий «собственных нужд»:

- кремниевая компиляция, которая используется для создания программируемых и заказных СБИС или УБИС;
- ПЛИС-технология, которая обеспечивает прямое отображение основных, интерфейсных и управляющих функций на (полу)заказные СБИС или УБИС (ПЛИС — программируемые логические интегральные схемы);
- бит-процессорная технология, которая обеспечивает отображение функций пользователя в «терминах» микропрограммного уровня организации вычислений в (Б)ВС с массовым параллелизмом, повышенной живучести и вычислительной устойчивости.
- ЦПОС-технология, которая обеспечивает отображение проблемно-ориентированных функций пользователя в «терминах» языка низкого уровня (ЦПОС — цифровой процессор обработки сигналов);
- RISC-технология, которая обеспечивает отображение проблемно-ориентированных функций пользователя в «терминах» языка высокого уровня многопроцессорных (Б)ВС (RISC — редуцированная система инструкций на микропрограммном уровне управления);
- нейрокомпьютерная технология, которая обеспечивает прямое отображение функций пользователя на сетевой уровень организации вычислений.

Показано:

1. В комплексной оценке качества (Б)ВС тесно переплетены как системотехнические, так и конструктивно-технологические факторы, первые из которых фактически предопределяют коэффициент использования физических возможностей параллельных (Б)ВС, а вторые — сами эти возможности.
2. Основными факторами, ограничивающими допустимый уровень параллелизма в (Б)ВС, являются не только потребляемая мощность и габариты, но и вычислительная устойчивость, которая в параллельной арифметике падает с ростом коэффициента распараллеливания вычислений. Поэтому в (сверх)параллельных (Б)ВС более перспективной можно считать последовательную (конвейерную) арифметику, обеспечивающую одинаковую и произвольно наращиваемую в ходе вычислений разрядность как регистров-аккумуляторов, так и шин обмена данными.
3. Основные системные аппаратно-временные издержки (сверх)параллельных (Б)ВС связаны с обеспечением взаимодействия многомерных потоков команд и многомерных потоков данных, в которых образуются «вихри» из-за наличия в программах операторов услов-

ных переходов, нарушающих *линейный порядок перечисления* потоков инструкций.

4. *Прямой перенос* алгоритмов и программ с последовательных на параллельные (Б)ВС *может привести к потере вычислительной устойчивости*, а значит, и к непрогнозируемому поведению автоматизированных комплексов в целом. Это чревато потерей управления сложными и опасными технотронными комплексами, функционирование которых осуществляется *на и над* собственной территорией.
5. Нейрокомпиляция является единственной альтернативой современным технологиям погружения заданий предметной области на уровень аппаратной реализации, и она *призвана* сократить финансовые, временные и интеллектуальные издержки на формирование исполняемого формата за счет замены дорогостоящих этапов алгоритмизации, программирования и компиляции обучением «дочерней» нейро-ЭВМ.
6. Нейрокомпьютерные технологии не отменяют интеллектуальный этап и связанные с ним затраты на создание формализованного описания задач. Они лишь изменяют *представление* формализованного описания в *виде репрезентативных обучающих выборок*. Обоснование и формирование репрезентативной обучающей выборки — не менее сложная интеллектуальная задача, чем получение формализованного описания задания пользователя ЭВМ.
7. Основное преимущество нейрокомпьютерных технологий сосредоточено в *стандартных процедурах обучения* или, что одно и то же, в формировании исполняемого формата задания вплоть до физико-технического или молекулярно-биологического уровня описания вычислительного процесса, а основная опасность использования нейрокомпьютерных технологий в технотронных комплексах проистекает от непрогнозируемости их поведения в быстро изменяющейся «внешней среде».
8. Успех **RISC**-технологий был обусловлен максимальной консолидацией финансовых и интеллектуальных ресурсов США под эгидой Агентства по прорывным исследовательским проектам в области обороны, но успешное продвижение инновационной технологии обеспечила коммерциализация инновационного проекта, изменившего стратегию создания изделий вычислительной техники.

В лекции 7 проанализированы наиболее вероятные изменения информационных технологий «собственных нужд», которые могут вызвать наноэлектронные или супрамолекулярные аппаратные платформы. Отталкиваясь от фундаментальных положений квантовой механики, мы показали:

1. В субмикронных твердотельных вентилях строго фиксированные детерминированные переходные процессы между уровнями *физических сигналов*, кодирующих логический «ноль» и логическую «единицу», находятся под полным контролем разработчиков и эксплуатирующего персонала средств вычислительной техники. В супрамолекулярных гетероструктурах переходные процессы уже существуют априори и носят вероятностный характер. Поэтому в супрамолекулярной электронике необходимо подавить множественные «паразитные» полимодальные взаимодействия как между субъединицами гетероструктуры, так и всей гетероструктуры с внешней средой.
2. Классическая технология кремниевой компиляции сохранит свою актуальность и в нанометровых вычислителях с той разницей, что:
 - может измениться на противоположное отношение «простой-сложный» между логическими и арифметическими функциями и реализующими их блоками, а также между комбинационными и конечными автоматами, работа которых приобретет вероятностный характер, включая и (до)определение начальных и текущих состояний;
 - размещение библиотечных бит-, слов- и поток-инструкций в тело (микро)программы практически всегда будет сопровождаться их структурно-функциональной «деформацией».
3. В нанометровых и супрамолекулярных вычислителях ассоциативная обработка приобретает атрибутивный характер, обеспечивая если и не динамичную, то по крайней мере приемлемую по времени структурно-функциональную адаптацию гипербольшого коллектива вычислителей (в случае полной деструкции вычислителя-предка) под требования как задач-наследников, так и действующей карты отказов.
4. Если в (суб)микронных изделиях электроники задача потребления энергии, отвода и рассеяния тепла решается на схмотехническом и конструктивно-технологическом уровнях проектирования, то в нанометровых вычислителях для этого придется привлекать и системотехнический уровень, обеспечивающий пространственно-временное *перераспределение парциальной энергии* за счет ее аккумуляирования и расходования в специфической для отдельных элементов, блоков и устройств физико-химической форме.

В результате таких изменений могут оказаться более эффективными:

 - а) критерии оптимальности, учитывающие не минимум аппаратно-временных затрат на решение каждой задачи, а максимум структурно-функционального сходства (родства) в общем случае не прямых алгоритмически ориентированных вычислителей-наследников и их предков;

- б) методы и средства структурно-функциональной адаптации и эволюции открытых систем, которые в явном виде учитывают энергетические (термодинамические) затраты:
- на частичную или полную деструкцию вычислителя-предка;
 - на полную или частичную консервацию термов вычислителя-предка для их дальнейшего использования в непрямом и, вообще говоря, отстоящем во времени вычислителе-наследнике;
 - на поддержание устойчивого функционирования вычислителя-наследника, «родственного» по структурно-функциональной схеме вычислителю-предку, на интервалах активности задач и т. п.;
- в) не методы классической компиляции программ, а формально-логические методы *интерпретации* задач-наследников в терминах задач-предков, которые могут изменять отношение «родства», повышая живучесть одной из задач-предков, а значит, и одного из вычислителей-предков;
- г) аналоги молекулярно-биологических технологий, в которых объединены в единый производственный цикл фактически неразрывные процессы:
- «формализации» задач пользователя;
 - «проектирования» нанометровых вычислителей путем компиляции или интерпретации алгоритмов и программ на уровне квантовых процессов;
 - адаптации или «производства» в неоднородном функционально-логическом и «технологическом» базисе гиперпараллельных вычислителей;
 - поддержания условий их правильного функционирования в широком диапазоне изменения внешних воздействующих факторов и при наличии «карт отказов» и т. п.

В лекции 8 проанализирована роль нейрокомпиляторов в создании инструментальных платформ наноэлектронных и супрамолекулярных вычислительных технологий. Показано, что аналоги молекулярно-биологических технологий должны учитывать следующие особенности прототипов:

1. *Инструктированный синтез* характерен только для этапа биологической эволюции, а *самоорганизация* и связанные с ней *синергетические* принципы и методы управления свойственны предбиологическому этапу. Оба этих метода взаимно дополняют друг друга, так как инструктированный синтез отталкивается от достаточно сложного молекулярного субстрата, который в эволюции жизни на Земле мог быть получен только с помощью самоорганизующихся нелинейных автокаталитических реакций.

2. Неоднозначная идентификация физико-химических объектов, от которой невозможно избавиться даже с привлечением фундаментальных констант:
 - предопределяет необходимость использования полуэмпирических методов и средств управления сложным нелинейным комплексом молекулярно-биологических процессов с глубокими обратными связями;
 - служит главным источником имитационного моделирования, которое обеспечивает любому организму необходимые условия восприятия и последующего анализа всех внешних воздействий.
3. Вариационный характер механизмов структурно-параметрической идентификации (узнавания) молекулярно-биологических субстратов говорит о том, что генетический код в традиционном инженерном понимании кодом не является и его следует рассматривать не как совокупность символов, а как совокупность «иероглифов», изменяющих свое «содержимое» в зависимости от контекста.
4. *Криптостойкость* генетического кода является результатом объективных эволюционных процессов — в их рамках исходный нуклеиновый субстрат синтезирован с использованием механизмов самоорганизации диссипативных структур и поэтому допускает гиперкомбинаторное множество конформационных преобразований, которое еще требуется ограничить преобразованиями, *сохраняющими отношение комплементарности* между кодонами и антикодонами, что является необходимым условием структурированного синтеза макромолекул нативного белка.
5. В нейрофизиологических системах преобразование информации осуществляется не столько в специфических операционных устройствах, сколько в распределенной и иерархически организованной ассоциативной памяти, работающей на принципах *структурно-параметрической адаптации и идентификации*.

В результате перспективный нейрокомпилятор должен комплексно использовать как методы и средства «нечисленной» нейроматематики для эффективного сопряжения нелинейных подсистем с глубокими обратными связями, так и строгую «численную» математику, которая способна точно описать поведение сопрягаемых молекулярно-биологических субъединиц, но бессильна дать точную оценку качества на всем гипербольшом многообразии альтернативных вариантов протекания управляемого процесса.

Не все положения и выводы, включенные в книгу, являются бесспорными. В частности, схематика нано- и супрамолекулярных структур может остановиться на управляемом извне *переносе в пространстве* заряженных частиц, как это имеет место в субмикронных УБИС, и не

пойти по пути их *перераспределения в пространстве*, изменяющем физико-химические свойства многоатомного субстрата, как это предполагается в разделе 7.5. Такая смена физико-технической платформы не может пройти без противоборства со сторонниками «твердотельного рая» в виде потенциальных барьеров и ям, которые сами когда-то вытеснили из вычислительной техники сторонников электронно-вакуумных приборов, считавшиеся более радиационно стойкими.

Разделы 2.1, 2.3–2,5, написаны А.А. Поповым, а разделы 1.3, 8.5 и 8.6 – совместно А.В. Котовым и Г.М. Алакозом. Разделы 2.2, 6.2 написаны А.П. Сериковым. Остальной материал книги написан Г.М. Алакозом.

После распада СССР исследования технологий микропрограммного конструирования были поддержаны командованием Военно-воздушной инженерной академии им. проф. Н.Е. Жуковского, и в первую очередь генерал-полковником авиации, дважды Героем Советского Союза, летчиком-космонавтом В.В. Коваленком, генерал-лейтенантом В.П. Кутаховым, генерал-лейтенантом В.А. Ефимовым, генерал-майором Ю.Б. Кулифеевым, полковником В.П. Харьковым и коллективом кафедры электронной автоматики.

Все перечисленным коллегам, а также корректору А. Ангилиной, редактору С. Перепелкиной и руководителю издательского отдела Н. Рахмановой авторы выражают глубокую благодарность и признательность.

Лекция 1. Междисциплинарный характер компьютерных проектов

1.1. Научно-технические и социально-экономические предпосылки зарождения вычислительных технологий

Пик дискуссий о роли и месте кибернетики и вычислительной техники в обществе пришелся на середину 50–60-х годов прошлого столетия [1,2] и был ориентирован на оценку перспектив их использования. Отсутствие видимого прогресса в области промышленно освоенных нанотехнологий, несмотря на 25-летние интернациональные усилия в этой области и многомиллиардные затраты, вынуждают провести теперь уже ретроспективный анализ причин и предпосылок, которые обусловили успешное завершение беспрецедентного по масштабам и результатам того времени междисциплинарного проекта.

Появление кибернетики и вычислительной техники в середине 40-х годов прошлого столетия было обусловлено целым рядом факторов, требовавших повышения скорости вычислений. Простейший способ ускорения вычислений — это выполнение действий на бумаге в столбик. Однако со временем такого ускорения оказалось недостаточно для расчета различного рода строительных и подвижных механических конструкций, что привело к созданию *арифмометра*, способного *механически* выполнять арифметические действия над десятичными числами. Механический арифмометр повысил производительность вычислений в несколько раз. Но в 30-х годах прошлого столетия и такая скорость оказалась недостаточной, что привело к «автоматизации» процесса выполнения *отдельных арифметических действий*. Осуществить такую «автоматизацию» удалось благодаря успехам электротехники, когда к механическому арифмометру добавили *электрический привод* и *электромеханические средства автоматизации*, обеспечившие выполнение *циклической* последовательности элементарных действий в алгоритмах умножения и деления десятичных чисел. В итоге удалось довести скорость вычислений до 1 операции за 2–5 секунд.

Вторая мировая война поставила человеческие жизни и накопленные человечеством материальные и культурные ценности в прямую зависимость от умения быстро и точно вычислять: кто быстрее и точнее принимал решения в зенитных и артиллерийских противоборствах на суше, море и в воздухе, тот и побеждал.

С позиций вычислительной техники сущность проблемы состояла в следующем. Возросшие скорости перемещения боевой техники (в первую

очередь самолетов и танков) повысили их маневренность на поле боя, что требовало повышения эффективности средств противодействия (зенитных и артиллерийских орудий), и в первую очередь на этапе прицеливания. Стрельба по подвижным целям требует определения *в темпе реального времени* «точки встречи» средства поражения (снаряда в кинетическом оружии) с целью. В свою очередь, достоверный расчет «точки встречи» требует учета целого ряда «случайных» факторов: скорости, ускорения и траектории движения цели, направления и скорости ветра, взаимного начального расположения цели и орудия и т. п. В итоге наводчику зенитного орудия необходимо было в течение нескольких секунд вычислить (*экстраполировать*) «точку встречи», направить на нее орудие и осуществить выстрел. На охоте за утками такая задача разрешима и решается охотником линейной экстраполяцией на 2–3 корпуса утки вперед. Успех такой стрельбы предопределен несопоставимой скоростью взлета утки и полета дроби, а также относительно малой дальностью стрельбы (50–100 метров), что в совокупности лишает утку возможности эффективного маневра и уклонения от встречи со средством поражения.

В случае самолета периода Второй мировой войны разность скоростей со снарядом уже была не так велика, а дальность стрельбы достигала нескольких километров, что позволяло летчику осуществлять плохо предсказуемые и поэтому достаточно эффективные противозенитные маневры. В итоге одна из наиболее развитых в мире система противовоздушной обороны (ПВО) Англии оказалась малоэффективной, а ее радиотехнические средства оказались способны решить достаточно ограниченную задачу раннего предупреждения о нападении. В итоге на начальном этапе войны Англия понесла значительные потери среди мирного населения (Ковентри), что несовместимо с основополагающими принципами построения и существования вооруженных сил любой страны, требующих, чтобы потери среди мирного населения были несопоставимо более низкими, чем в армии.

Отсюда и встала жизненно важная задача *экстраполяции в темпе реального времени* случайного процесса и достоверного вычисления «точки встречи» снаряда и цели. Она потребовала увеличить производительность вычислительных средств на несколько порядков и включить их в состав контура управления «радиолокационная станция — зенитные орудия».

Автор кибернетики Н. Винер, решая эту задачу, исходил из следующего. Точную экстраполяцию положения самолета выполнить невозможно, так как с точки зрения артиллериста траектория самолета носит случайный характер. Но при стрельбе зенитным снарядом нулевая ошибка наведения и не требуется, так как имеется коническая область разлета осколков, а не одна точка поражения, а значит, и встречи цели и снаряда. Более того, траектория полета самолета не совсем случайная и ограниче-

на такими факторами, как перегрузка летчика, инерционность системы управления, прочность летательного аппарата и т. п. Поэтому неопределенность в будущем положении самолета можно оценить приближенно с помощью статистических методов, то есть на основе предыстории наблюдения траектории полета.

Заслуга Н. Винера как раз и состояла в том, что он отказался от требования стопроцентного поражения самолета и стал решать задачу по критерию максимальной вероятности поражения цели. Однако методы экстраполяции случайных процессов требовали колоссальных по тем временам объемов вычислений, которые необходимо было осуществить в темпе реального времени, то есть в течение нескольких десятков секунд наблюдения за траекторией полета. Выполнить такой объем вычислений на электромеханических арифмометрах практически было невозможно, даже распределив работу между коллективом вычислителей, которые *должны были координировать свои действия и обмениваться информацией друг с другом по ходу выполнения алгоритма вычислений.*

Отсюда, прежде чем автоматизировать процесс наведения зенитного орудия на цель необходимо было:

- автоматизировать процесс управления ходом вычислений;
- использовать качественно новое вычислительное оборудование, работающее с производительностью порядка нескольких десятков тысяч операций в секунду, что требовало использования в вычислительной технике принципиально новых физико-технических принципов работы аппаратуры.

С позиций автоматизации вычислительный процесс представляет собой заданную программистом и корректируемую содержимым потоков обрабатываемых данных *частично упорядоченную во времени последовательность арифметико-логических действий*, которые необходимо выполнить на том или ином вычислительном устройстве над входными и промежуточными данными, чтобы получить требуемый результат вычислений. Поэтому автоматизация управления ходом вычислений прежде всего требовала формализации понятия *алгоритма*. Такая формализация была проведена к началу 30-х годов прошлого столетия для решения «внутренних» проблем математики. Поэтому Н. Винеру и его современникам осталось только материализовать требования формальных определений и понятий, которые интуитивно сводились к следующему.

Алгоритм должен представлять собой конечный набор правил выполнения некоторой процедуры, которая удовлетворяет трем главным требованиям:

1. *Массовость*, которая гарантирует выполнение не одной конкретной процедуры, а целого класса однородных и однотипных процедур. Например, процедура вызова абонента телефонной сети регламен-

тирует правила набора номера «вообще» и не зависит от конкретного номера абонента.

2. *Детерминированность*, которая гарантирует однозначное понимание каждой инструкции алгоритма, что гарантирует воспроизводимость результатов вычислений, если они выполнены над одними и теми же входными данными. В нашем случае правильная связь с нужным абонентом будет всегда установлена, если исключить ошибки набора и идентификации его номера.
3. *Результативность*, которая гарантирует конечность применения отдельных инструкций и алгоритма в целом. В случае экстраполяции траектории полета самолета результат должен быть получен за конечное число шагов $L_p \leq \Delta T / \tau_c$, где L_p — количество реально исполненных команд программы; ΔT — интервал времени, регламентируемый соотношением дальности до цели и скоростями полета цели и зенитного снаряда; τ_c — скорость выполнения арифметико-логических действий используемым вычислителем.

На рубеже XX века Давид Гильберт сформулировал задачу «автоматизации» процесса вывода математических теорем: «Можно ли построить алгоритм, строящий необходимый алгоритм решения любой точно поставленной задачи?». Программа Д. Гильберта была направлена на получение универсального алгоритма, сводившего деятельность математиков к формальной механической игре, в которой произвольную формализованную задачу (точно описанную на языке математики) можно было решить *механически*, используя процедуры универсального алгоритма. Парадокс программы Д. Гильберта обнаружился практически сразу [3] и состоит он в том, что для доказательства существования такого универсального алгоритма требуется его строгое математическое описание. Этот парадокс был обнаружен Бертраном Расселом (впоследствии научный руководитель Н. Винера в аспирантуре) и сводился к хорошо известному еще древним грекам парадоксу брадобрея, «который бреет всех тех и только тех мужчин деревни, которые не бреются сами». Вопрос «Кто бреет брадобрея?» ставит любую «формальную» машину в тупик, так как такой брадобрей существует в том и только в том случае, когда его нет. Аналогичная ситуация складывается и с «множеством всевозможных подмножеств, которые не включают себя в качестве элемента», что является необходимым условием существования универсального алгоритма Гильберта.

Чтобы обойти возникшую парадоксальную ситуацию, было предложено большое количество уточнений понятия алгоритма, а наш соотечественник академик Андрей Николаевич Колмогоров построил схему, лежащую в основе любого уточнения этого понятия. С помощью этой схемы можно строить новые уточнения как угодно долго.

В вычислительной технике остановились на уточнении понятия алгоритма, которое предложил английский математик Алан Матис Тьюринг и

которое известно под названием «машина Тьюринга». Эта «машина» до сих пор служит формально-логическим прототипом любой ЭВМ.

Машина Тьюринга состоит из [4]:

- разделенной на клетки *бесконечной* ленты;
- управляющего устройства, способного перемещаться вдоль ленты и выполнять предписанные «элементарные» действия;
- специальной таблицы управления, задающей правила функционирования всей машины.

На ленту можно записывать «слова», то есть упорядоченные последовательности символов из некоторого фиксированного и конечного алфавита.

На каждом шаге своей работы устройство управления извлекает символ, стоящий в клетке, против которой находится курсор устройства управления. Прочитав символ, устройство управления анализирует состояние, в котором оно находится в данный момент времени или, что одно и то же, на данном такте работы. Список возможных состояний устройства управления конечен, фиксирован и известен заранее, а пара «считанный символ — состояние устройства управления» однозначно идентифицирует каждую клетку таблицы управления, на основе которой и определяется реакция машины Тьюринга. Считанный символ обычно идентифицирует строку, а состояние — столбец таблицы управления. В идентифицированной таким образом клетке таблицы управления записана инструкция, которую и должна выполнить машина Тьюринга.

Каждая инструкция в явном виде содержит:

- какой символ надо записать на место считанного символа;
- состояние, в которое должно перейти устройство управления;
- указание на направление перемещения по ленте: «вправо» или «влево».

Выполнив все указания, содержащиеся в данной клетке таблицы управления, машина Тьюринга завершает один цикл своей работы.

Пусть машина Тьюринга работает по правилам таблицы 1.1 [4].

Таблица 1.1.

	G_0	G_1	G_2
\emptyset	$\emptyset, G_0, \text{Stop}$	$\emptyset, G_1, \text{Right}$	$1, G_0, \text{Stop}$
1	$1, G_0, \text{Stop}$	$\emptyset, G_2, \text{Right}$	$1, G_2, \text{Right}$
+	$+, G_0, \text{Stop}$	—	$1, G_0, \text{Stop}$

Здесь символ « \emptyset » соответствует «пустой» клетке бесконечной ленты машины. «Слова», которые понимает данная машина Тьюринга, не должны содержать подряд несколько символов «+», и этот символ не может находиться в начале и в конце «слова», то есть запрещенными считаются

[. . .]