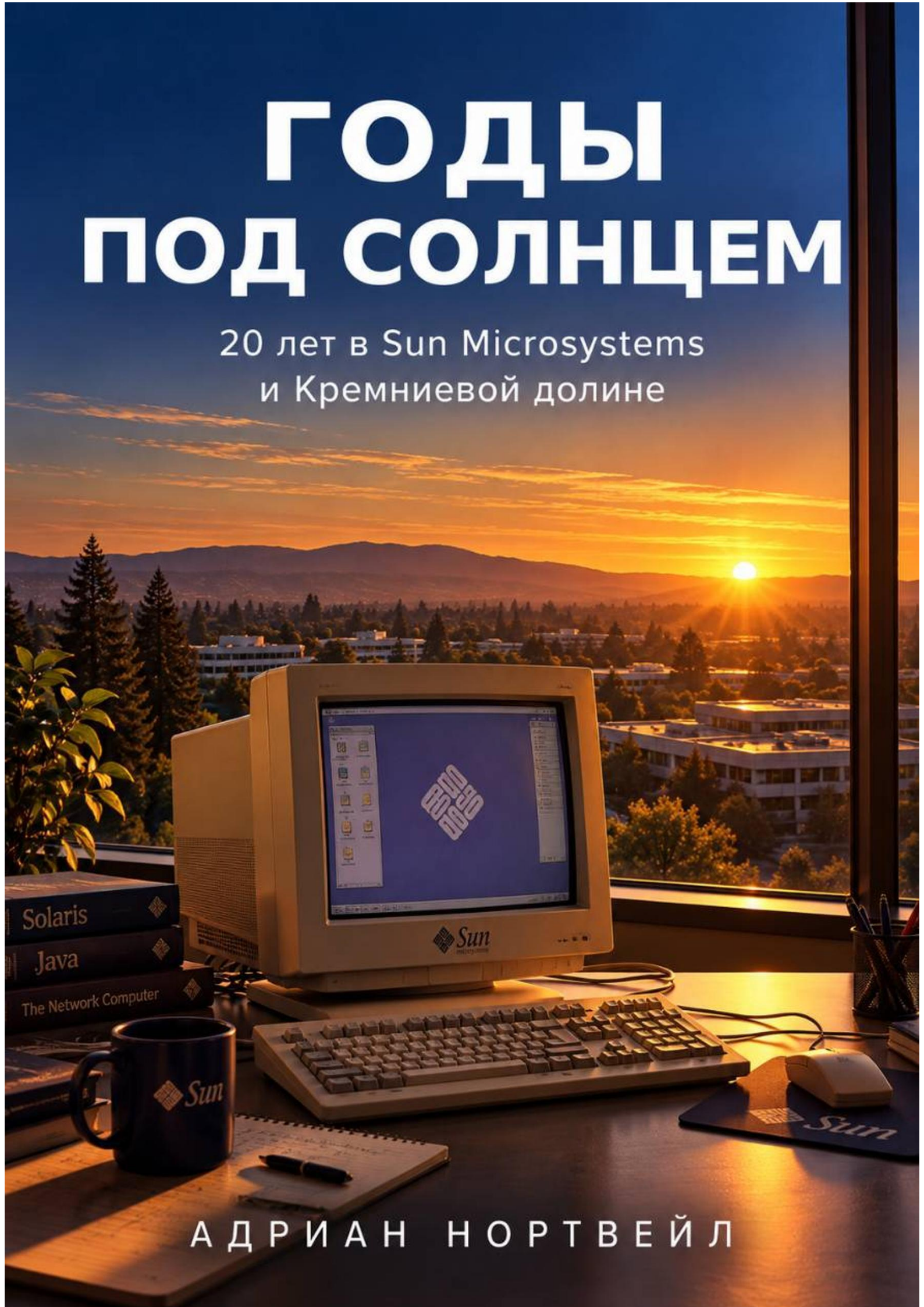


ГОДЫ ПОД СОЛНЦЕМ

20 лет в Sun Microsystems
и Кремниевой долине



АДРИАН НОРТВЕЙЛ

Адриан Нортвейл

**Годы под солнцем. 20 лет в Sun
Microsystems и Кремниевой долине**

«Автор»

2026

Нортвейл А.

Годы под солнцем. 20 лет в Sun Microsystems и Кремниевой долине
/ А. Нортвейл — «Автор», 2026

Кремниевую долину обычно рассказывают победители после победы. Здесь вы входите внутрь до того, как история стала легендой. Герой книги провел 20 лет в Sun Microsystems и видел изнутри взлет компании, инженерные войны, решения по SPARC, Java, хранению данных, чипам, поглощению Cray, эпохе доткомов и моментам, когда коммерческий выбор менял техническое будущее. Это не отполированная корпоративная хроника, а мемуары инженера и руководителя, который сидел достаточно близко к столу решений, чтобы понимать, почему все произошло именно так. Хотите знать, как на самом деле рождаются большие технологии, почему сильные компании ошибаются и чем живет Кремниевая долина за пределами презентаций? Откройте эту книгу. Здесь индустрия перестает быть мифом и становится живой. При редактировании и структурировании материала использовался ChatGPT

© Нортвейл А., 2026

© Автор, 2026

Содержание

Предисловие	5
До работы в Shengyang	7
Первая работа	7
Наконец я устроился в IBM	9
Дни стартапа	11
Переход в Sun Microsystems	14
Краткое пребывание в отделе рабочих станций	15
Проект Sunburst	17
Разные подходы	19
Усовершенствование продукта	20
Три варианта ИТ-услуг от Лю Дусу	21
Проект был отменен	22
Проект «Sun Dragon»	23
Собеседование	25
Ранние решения по проекту Sun Dragon	27
Одномодульная системная плата	28
Одна шина Dynabus	30
Удвоение количества процессоров	31
Транзисторная логика Ганнинга (GTL)	32
«Ло» проекта «Sun Dragon»	33
Опыт найма, которым я горжусь	38
Кризис нехватки процессоров	39
Конец ознакомительного фрагмента.	41

Адриан Нортвейл

Годы под солнцем. 20 лет в Sun Microsystems и Кремниевой долине

Предисловие

Летом 2016 года один из моих однокурсников, который после выхода на пенсию преподавал в Государственном университете Сан-Хосе в Калифорнии, пригласил меня преподавать там курс «Архитектура и проектирование компьютеров». В аспирантуре я изучал именно компьютерную инженерию. Учебник, который использовался на этом курсе в аспирантуре Государственного университета Сан-Хосе — «Архитектура и проектирование компьютеров» Джона Хеннесси (John Hennessy) и Дэйва Паттерсона (Dave Patterson) — был именно той книгой, которую я много лет хотел прочитать, но так и не находил на это времени. Я с радостью принял этот вызов.

Когда я читал эту книгу, мой более чем тридцатилетний опыт работы в Кремниевой долине в Калифорнии словно начал прокручиваться в моей голове сцена за сценой. Когда в 70-е годы я изучал курс «Архитектура компьютеров» в аспирантуре Университета Иллинойса в Шампейне, в качестве учебного материала использовались лишь статьи, опубликованные в научных журналах. Сегодня же книга Хеннесси и Паттерсона в основном посвящена развитию компьютерной инженерии, произошедшему после того времени. И я был одним из участников этого процесса.

Я никогда не думал о том, чтобы написать автобиографию. Я не являюсь известной общественной фигурой. Кому это может быть интересно? Однако, преподавая в Государственном университете Сан-Хосе, я заметил, что часто рассказываю студентам о «настоящих» причинах тех или иных событий. Хотя в целом техническое развитие следует логическому процессу, многие решения принимаются исходя из коммерческих соображений той или иной компании. А иногда — даже из-за личных предпочтений того или иного руководителя.

Студенты, похоже, проявляли большой интерес к этим закулисным историям. Это во многом подтолкнуло меня к тому, чтобы зафиксировать события, в которых я принимал личное участие.

Компьютерная инженерия и бизнес — это командная деятельность. Она основана на вкладе множества людей. Мне посчастливилось на протяжении всей моей карьеры работать с множеством талантов мирового уровня. В своих рассказах я использую их настоящие имена. Возможно, благодаря этому их вклад станет известен широкой публике. Разумеется, я описываю эти события, опираясь на свои воспоминания. Если в тексте есть ошибки, то это моя вина. Приношу за это извинения.

Я решил не описывать весь свой профессиональный путь. Я расскажу только о том, что делал, видел и слышал за 20 лет работы в Sun Microsystems. Если бы я описал всю свою карьеру, эта книга слишком напоминала бы мою автобиографию. В 90-е годы Sun была похожа на сегодняшние Google или Apple — она бурлила творческой энергией. Компания была основана в 1982 году Энди Бектолсхаймом (Andy Bechtolsheim), Биллом Джоом (Bill Joy), Скоттом Макнили (Scott McNealy) и Винодом Хосла (Vinod Khosla). Вначале она была поставщиком рабочих станций, собираемых из готовых компонентов. Вскоре, в 90-е годы, она превзошла таких гигантов информационной индустрии того времени, как IBM, Digital Equipment Corporation и Hewlett-Packard. В конце тех бурных 90-х она даже стала «той самой точкой в доменном име-

ни.com». Я присоединился к Sun не на самом раннем этапе. Я пришел в компанию в октябре 1988 года, уже после того, как акции Sun были выпущены на биржу.

За двадцать лет работы в Sun я стал свидетелем как взлета, так и окончательного упадка компании. Я не входил в высшее руководство Sun, но занимал достаточно высокую должность, чтобы понимать причины многих решений, а порой даже иметь возможность участвовать в их принятии и влиять на них. Записывая свои тогдашние переживания и мысли, я надеюсь предоставить молодым студентам-инженерам ориентир для будущей карьеры или дать студентам, изучающим менеджмент, возможность заглянуть в мир инженерного руководства. Что касается моих коллег по информационной индустрии из того же поколения, то, возможно, некоторые из описанных мной событий вызовут у них воспоминания и заставят улыбнуться от осознания. Читатели за пределами Кремниевой долины также смогут через различные закулисные истории, приведенные в книге, заглянуть в суть того, как на самом деле работает Кремниевая долина.

Для меня большая честь, что в моей жизни были эти годы, проведенные «под солнцем».

До работы в Shengyang

Первая работа

Когда в октябре 1980 года я получил докторскую степень в Университете Иллинойса в Шампейне, я одновременно получил предложения о работе от лаборатории IBM Watson, Центрального исследовательского института Texas Instruments, лаборатории HP, National Semiconductor, Университета Южной Калифорнии и подразделения TRW Array Processor. Я присоединился к TRW.

В то время IBM, Texas Instruments и HP были ведущими компаниями в сфере информационных технологий. Особенно IBM. Возможно, именно по этой причине IBM рассматривала первый год работы вновь нанятых исследователей как «испытательный срок». Поскольку это был испытательный срок, IBM не подавала заявления на получение постоянного вида на жительство (грин-карты) для вновь нанятых иностранных студентов, имевших только визу F-1, в течение их 18-месячного стажировочного периода. (В то время IBM предпочитала нанимать выпускников, только что закончивших учебу, поскольку их было легче подготовить и интегрировать в корпоративную культуру IBM. Кроме того, будучи компанией, практически занимавшей монопольное положение и обладавшей значительными финансовыми ресурсами, IBM не нужно было беспокоиться о том, что предыдущие работодатели новых сотрудников могут подать на неё в суд за попытку похищения коммерческой тайны.)

Я объяснил IBM, что такие условия пробного периода делают их предложение о работе неконкурентоспособным, поскольку другие компании обещали, что при удовлетворительных результатах работы они сами подадут за меня заявление. В конце концов IBM пошла на уступки.

Несмотря на это, после месяца мучительных раздумий я всё же решил поступить на работу в подразделение матричных процессоров TRW. Их зарплата, по сравнению с другими, была невысокой. И дело было не в том, что они более известны. Наоборот, среди этих компаний они были самыми малоизвестными. Но они действительно находились в Кремниевой долине в Калифорнии. Для выпускника факультета электротехники работа в Кремниевой долине означала, что даже при смене компании ему не пришлось бы переезжать всю жизнь. Но ведь у лабораторий HP и National Semiconductor есть те же преимущества. Уникальность TRW Array Processors по сравнению с другими компаниями заключалась именно в том, что она была неизвестной величиной.

Я пытался выбрать работу методом исключения. Во всех остальных вакансиях мне всегда удавалось найти какие-то недостатки. Например, как раз в тот момент, когда я собирался принять решение, в Далласе, где находится Центральный исследовательский институт Texas Instruments, разразилась двухнедельная жара с температурой выше 38 градусов по Цельсию. Даже для меня, выросшего на Тайване, это было слишком жарко. TRW Array Processor — это стартап, созданный внутри концерна TRW. Он занимался разработкой нового высокопроизводительного матричного процессора. На тот момент компания существовала не более года. Её будущее было самым непредсказуемым (или самым многообещающим?) из всех предложений. Именно эта неопределённость, напротив, не давала мне возможности придаться к чему-либо. К тому же они наняли Роберта Лоу, моего профессора по компьютерной инженерии из Иллинойса, в качестве главного архитектора. Профессор Лоу входил в тройку моих научных руководителей при написании диссертации. Это тоже оказало на меня некоторое влияние.

На протяжении многих лет я время от времени вспоминал об этом «неортодоксальном решении». Думаю, главная причина, по которой я выбрал матричный процессор TRW, заклю-

чалась в том, что в глубине души мне действительно нравилось разрабатывать новаторские передовые продукты! Однако тогда я и не подозревал, что этот интерес определит всю мою профессиональную карьеру.

К сожалению, 1 октября 1980 года, то есть в первый же день моего поступления на работу в проект массивных процессоров TRW, из-за спада в электронной промышленности, а также в связи с уходом в отставку ответственного вице-президента, компания TRW объявила о приостановке проекта массивных процессоров. После моего прихода компания перестала нанимать новых сотрудников, и пока обсуждалось, как поступить с этим проектом и с уже нанятой дюжиной сотрудников, это новое начинание временно превратилось во внутренний проект по разработке продукта.

Мы, бывшие участники проекта, под руководством менеджера продолжили разработку небольшой демонстрационной модели матричного процессора, пытаясь повлиять на окончательное решение высшего руководства компании. Из-за нехватки кадров мне пришлось в одиночку выполнять функции архитектора этой модели, инженера по логическому проектированию и разработчика микропрограмм. Через год мы провели чрезвычайно успешную демонстрацию, в ходе которой с помощью этой модели быстро проанализировали спутниковые снимки, секретность которых была снята правительством США. К сожалению, несмотря на успех демонстрации, мы поняли, что наши попытки повлиять на руководство компании потерпели неудачу. Вице-президент, ответственный за наш проект в южнокалифорнийской штаб-квартире TRW, которого мы пригласили на демонстрацию, так и не явился.

В начале 1982 года TRW наконец решила присоединить нас к своей дочерней компании ESL, расположенной в Саннивилле, Калифорния. ESL была компанией, специализирующейся исключительно на контрактах с правительством США и военными. Как сотрудник, не являющийся гражданином США, я был обязан носить идентификационный бейдж с красной точкой. Даже когда я выходил из офиса, чтобы сходить в туалет, меня должен был сопровождать кто-то из сотрудников. Очевидно, что это была не та среда, в которой я хотел работать. Я решил уйти. Когда я подал заявление об увольнении, мой менеджер, напротив, извинился передо мной.

Когда я искал новую работу, мне посоветовали попробовать устроиться в стартап в Кремниевой долине под названием Sun Microsystems, занимавшийся производством каких-то настольных компьютеров. Очевидно, я только что приехал в Кремниевую долину и ещё не смог оценить прелести небольшой компании, в которой работало всего чуть больше ста человек. Мне это не было интересно.

Наконец я устроился в IBM

Я до сих пор сожалею, что после окончания университета не принял предложение о работе в IBM. В 1960-х и 1970-х годах IBM была всемирно известной и авторитетной компанией. Я как-то пошутил со своим американским другом, что знал название IBM ещё до того, как научился *писать* по-английски «Соединённые Штаты Америки».

Теперь я уже в Калифорнии. Очевидно, они сочли меня достаточно «свежим» кандидатом, и я подал заявку на работу в Исследовательский центр IBM в Сан-Хосе и, к счастью, был принят. Не так давно IBM изобрела считывающе-записывающие головки для магнитных дисков, изготовленные из тонкопленочных материалов. Чтобы заменить существующую систему, в которой для ускорения производства контроль качества осуществлялся дорогостоящим персоналом, работающим в три смены, IBM хотела создать высокопроизводительный компьютер для контроля качества с параллельной и конвейерной архитектурой, позволяющий программно изменять целевые показатели качества. Я отвечал за аппаратную часть компьютера. Это был очень интересный проект. Мне посчастливилось работать с группой талантливых и дружелюбных коллег.

Большинство моих коллег по проекту массивных процессоров в TRW в конечном итоге покинули ESL и устроились в различные компании Кремниевой долины. Роберт Лоу и Элен Кума, отвечавший за финансы проекта массивных процессоров, добились от TRW лицензии на использование интеллектуальной собственности, разработанной нами в рамках этого проекта. Это позволило Роберту Лоу продолжить работу над своим новым изобретением — « » (очень большое слово команды) (Very Large Instruction Word; сокращённо VLIW). Он назвал её «*полициклической архитектурой*» (Polycyclic Architecture).

Роберт Лоу, Элен Кума и я вскоре начали собираться по выходным, чтобы обсуждать возможность создания суперкомпьютера на основе этой полициклической архитектуры. Спустя несколько месяцев Роберт Лоу и Элен Кума подготовили бизнес-план и начали продвигать его на Санд-Хилл-Роуд (Sand Hill Road) в Монро-Парке, Калифорния. (Если вы живёте в Кремниевой долине, вы наверняка понимаете, о чём я говорю. Улица Сэнд-Хилл-Роуд в Монро-Парке, штат Калифорния, просто кишит венчурными инвестиционными компаниями. Многие известные сегодня высокотехнологичные компании получили здесь своё начальное финансирование.)

Поскольку на тот момент у нас были лишь некоторые идеи, ни у кого из нас не было впечатляющего профессионального опыта или опыта в сфере предпринимательства. Попытки привлечь финансирование поначалу не увенчались успехом.

Наконец представился шанс выйти из этого тупика. Роберт Лоу услышал, что легенда компьютерной индустрии Гордон Белл (Gordon Bell) собирается приехать в район залива Сан-Франциско. Роберт познакомился с ним, когда преподавал. Роберт Лоу выяснил, в каком отеле остановился Белл, и, воспользовавшись случаем, рассказал ему о нашей идее. Белл сказал, что идея интересная. Возможно, он просто хотел быть вежливым. Но этого было достаточно! Во всех последующих презентациях для венчурных компаний Роберт Лоу не забывал упоминать, что Белл назвал эту идею интересной. Хотите верьте, хотите нет, но через два месяца три венчурные компании выделили по 200 000 долларов США в качестве стартового капитала, чтобы мы могли продолжить доработку нашей идеи.

Мы решили использовать UNIX версии 5 от American Telephone and Telegraph (AT&T) в качестве операционной системы для нашего нового компьютера. Нам нужен был специалист по операционным системам, чтобы установить её на наш новый компьютер. Лучшим специалистом по UNIX, которого я знал, был мой брат Янь Вэйцзюнь, работавший в то время в лабораториях Hew. С другой стороны, для эффективной работы архитектуры VLIW требовалась

мощная поддержка со стороны компилятора. Роберт Лоу пригласил в нашу команду эксперта по компиляторам Россса Тоула (Ross Towle). В июне 1984 года мы в пятером основали стартап по производству мини-суперкомпьютеров под названием «Axiom».

Поскольку это название уже было занято другой компанией, во избежание юридических споров о нарушении прав мы позже переименовали компанию в «Cydrome». Мы потратили 50 000 долларов на услуги лингвистической компании из Сан-Франциско под названием «Name Lab». Говорят, что именно эта компания в своё время придумала название для известной нефтяной компании «Эксон» (Exxon). Сначала мы в пятером пытались придумать название самостоятельно. К сожалению, никому из нас не нравились варианты, предложенные другими. Задача, казавшаяся простой, превратилась в многочасовую дискуссию. Когда «Name Lab» придумала новое слово «Cydrome», лингвисты заявили, что оно символизирует гоночную трассу для умных автомобилей. Никому из нас оно не понравилось. Но благодаря этому у нас не осталось никаких предубеждений. Деньги уже были потрачены, и в итоге мы всё-таки выбрали это название.

Дни стартапа

Этот стартовый капитал в размере 600 000 долларов США требовал от нас разработки полного набора технических спецификаций для архитектуры, аппаратного обеспечения и программного обеспечения, необходимых для создания предложенного нами мини-суперкомпьютера.

В то время рыночные перспективы таких мини-суперкомпьютеров казались весьма обнадеживающими. Популярный мини-компьютер VAX-11/780 компании Data Equipment продавался тогда по цене около 250 000 долларов США. Эта цена была вполне доступной для обычных организаций и учебных заведений. В то же время мощный суперкомпьютер Cray-1 стоил более 1 миллиона долларов США, что было недостижимо для большинства организаций. На рынке высокопроизводительного вычислительного оборудования между этими двумя моделями не было никаких других продуктов. Именно это и стало причиной, по которой компания TRW создала свой стартап по производству массивных процессоров. TRW использовала VAX-11/780 во многих своих государственных проектах. Для повышения вычислительной производительности 780 часто приходилось устанавливать ускорители, и массивный процессор AP-120B компании Floating Point Systems из *Сиэтла* стал одним из таких решений. Со временем стало очевидно, что TRW устала от этих дополнительных расходов. Поэтому она решила заняться этим самостоятельно, хотя и не довела дело до конца.

SaiZhong не была первой стартап-компанией, ориентировавшейся на эту нишу рынка. Техасская компания Convex и Alliant с восточного побережья США были основаны ещё до нас. В течение следующих двух лет появились ещё четыре-пять стартапов. Все эти компании использовали свои инновационные проприетарные архитектуры, стремясь достичь производительности, сопоставимой с Cray, при цене ниже одного миллиона долларов. Период с 1983 по 1989 год можно назвать «золотым веком» разработки компьютерных архитектур.

Мы, пятеро парней чуть за тридцать, в том числе четверо докторов наук и один бухгалтер, были полны энтузиазма, но не имели практически никакого опыта работы в промышленности. Тогда, в начале 80-х, компания Data Equipment Corporation (DEC) из Массачусетса переживала расцвет благодаря своему VAX-11/780. Мы твердо решили стать второй компанией DEC. Хотя среди основателей компании я был единственным специалистом по аппаратному обеспечению и впоследствии создал команду из более чем сорока человек, включая инженеров, техников и менеджеров, венчурный фонд не позволил мне занять должность вице-президента по аппаратному обеспечению, поскольку в IBM я был всего лишь научным сотрудником и руководителем небольших проектов. Мне пришлось нанять более опытного менеджера из DEC в качестве своего начальника.

Затратив более трёх лет, мы создали довольно крупную компьютерную систему, и в сентябре 1987 года поставили шесть прототипов заинтересованным покупателям. В наших машинах использовались новейшие на тот момент микросхемы Fairchild и Motorola, называемые *матрицами логических вентиляей* (gate arrays). По сравнению с сегодняшней плотностью логических вентиляей в микросхемах «массивы вентиляей», плотность в те времена была довольно низкой. Из-за этого в корпус шириной 19 дюймов приходилось устанавливать множество печатных плат размером 20 на 20 дюймов. Массивное использование аппаратного обеспечения привело к тому, что цена нашей машины превысила 500 000 долларов США. К сожалению, хотя впечатляющая производительность нашей машины соответствовала первоначально поставленным целям, рынок за это время уже изменился.

Все студенты-инженеры-компьютерщики, знакомые хотя бы немного с историей вычислительной техники, должны знать о радикальной реформе, произошедшей в области центральных процессоров (Central Processing Unit, или CPU) в середине 80-х годов. После десятиле-

тий стремления к созданию сложного набора команд (Complicated Instruction Set,) в попытке приблизиться к высокоуровневым языкам программирования и добиться высокой производительности, все более длительные задержки между работой логических схем процессора и оперативной памятью (memory) подтолкнули к появлению нового подхода.

В 1986 году появилась новая архитектурная концепция, получившая название «вычисления с сокращённым набором команд» (*Reduced Instruction Set Computing, RISC*). Вместо поддержки множества сложных команд, обеспечивающих прямой доступ к данным в памяти, архитектура RISC использует в процессоре большое количество регистров. Это позволяет ускорить доступ к операндам. Кроме того, кодер становится проще оптимизировать. Прямой доступ к данным в памяти ограничивается командами LOAD и STORE. Все команды имеют одинаковую длину и легко декодируются. Их выполнение также легче организовать в конвейерном режиме. Все это повышает производительность вычислений.

В 1986 году архитектуру RISC представляли процессоры SPARC, изобретённый Дейвом Паттерсоном (Dave Patterson) из Калифорнийского университета в Беркли, и MIPS, изобретённый Джоном Хеннесси (John Hennessy) из Стэнфордского университета. ARM, хотя и была разработана в то же время, изначально полностью ориентировалась на встраиваемые системы. Компания Sun Microsystems из Кремниевой долины быстро выбрала SPARC, а Silicon Graphics — MIPS для производства своих рабочих станций.

К осени 1987 года, когда множество стартапов, занимающихся мини-суперкомпьютерами, включая Saizhong, начали выпускать свою продукцию, рабочие станции Sun Microsystems и Silicon Graphics, отличавшиеся доступной ценой и высокой производительностью (10 миллионов команд в секунду), стали пользоваться популярностью на рынке. Кроме того, ходили слухи о скором выпуске их версий, способных обрабатывать 20 миллионов команд в секунду. Многие клиенты из сферы высокопроизводительных вычислений, которые в самом начале проявляли к нам большой интерес — в том числе голливудские производители цифрового кино и несколько государственных исследовательских лабораторий, — начали колебаться. Они заявили нам, что вместо того, чтобы тратить 500–600 тысяч долларов и специально разрабатывать приложения под проприетарные архитектуры различных мини-суперкомпьютеров, они предпочли бы приобрести 10–20 рабочих станций Sun или Silicon Graphics для достижения той же цели. Не говоря уже о том, что для этих популярных рабочих станций уже существовал огромный выбор прикладного программного обеспечения.

Конкуренция за тех клиентов, которые по-прежнему интересовались высокопроизводительными вычислениями, стала обостряться. Стартапы, занимающиеся производством мини-суперкомпьютеров, начали предлагать заинтересованным клиентам шестимесячное бесплатное тестирование. Таким образом, по крайней мере, серверные помещения этих клиентов были заняты, и они не могли опробовать продукты других компаний.

К сожалению, хотя клиенты временно не платили, нам всё равно приходилось тратить на техническую поддержку поставленных машин. В начале 1988 года нам позвонили из Питтсбургской лаборатории суперкомпьютеров. В их компьютерном зале произошёл небольшой пожар, и сработала автоматическая система пожаротушения на крыше. Химикат для тушения попал на наши машины Cydra-5. Возможно, они были повреждены. Мы, несколько инженеров, срочно обсудили, как справиться с этой ситуацией. Именно в этот момент наш финансовый директор Элен Кума громко рассмеялся. Он сказал, что мы скоро получим первый доход, потому что страховая компания должна будет за это заплатить!

К 1988 году компания Saizhong, пройдя три раунда инвестиций на общую сумму 50 миллионов долларов США, выросла до штата в 150 человек. К сожалению, из-за плохих рыночных перспектив мини-суперкомпьютеров наш ведущий инвестор, компания Prime Computers, решила выйти из проекта. Расположенная на шоссе 128 в штате Массачусетс, Prime Computers в то время была весьма успешным производителем мини-компьютеров. Изначально они про-

являли большой интерес к технологиям Saizhong и даже собирались использовать Cydra-5 в качестве основы для своей продукции следующего поколения. Поскольку они были лидерами в этой отрасли, многие венчурные компании, инвестировавшие в Saizhong, фактически следовали за ними. В начале 1988 года генеральный директор (CEO) Prime Computers захотел приобрести Cydra ещё до её выхода на биржу. В результате совет директоров Prime Computers направил двух опытных директоров в Калифорнию, чтобы они лично оценили компанию Cydra и рынок мини-суперкомпьютеров. Они провели два дня, осматривая компанию, беседуя с каждым ключевым сотрудником и изучая рынок. В итоге они пришли к шокирующему выводу. Они не только отклонили предложение генерального директора о слиянии, но и потребовали, чтобы «Прайм» полностью ушла с этого жестокого рынка, включая свои инвестиции.

Вывод инвестиций компанией «Прайм» полностью подорвал доверие всех остальных венчурных инвесторов. Они заморозили оставшиеся на банковских счетах «Сайчжун» средства. После четырёх подряд волн сокращений «Сайчжун» наконец закрыла свои двери в конце 1988 года. Компания HP за 50 000 долларов США приобрела все права на интеллектуальную собственность, включая три моих патента в США, а также два работающих прототипа, которые были в нашей компании. Одновременно они наняли Роберта Лоу в лабораторию HP для продолжения разработки этой архитектуры.

К концу 1989 года рынок мини-суперкомпьютеров фактически полностью исчез. Интересно, что порядок закрытия этих компаний был прямо противоположен порядку их создания. Поскольку к моменту, когда рабочие станции начали серьезно подрывать рынок мини-суперкомпьютеров, у Compath и Eyrient уже было значительное количество клиентов, им удалось продержаться ещё несколько лет за счёт доходов от сервисного обслуживания.

Переход в Sun Microsystems

Когда компания SaiZhong закрылась, я начал искать работу. Я уже находился в Кремниевой долине. Отрасль «мини-суперкомпьютеров» была фактически уничтожена «супер»-рабочими станциями. А менее десяти лет назад, когда я ещё учился в аспирантуре, термина «рабочая станция» в классификации компьютеров ещё даже не существовало! Чем же они были так удивительны? Мини-суперкомпьютеры SaiZhong для выполнения своих задач использовали целый стойку печатных плат размером 20 на 20 дюймов. А в каком волшебстве заключалась рабочая станция Sun, похожая на коробку для пиццы (pizza box) и размещенная в небольшом корпусе под монитором на столе?

У моих американских друзей есть поговорка: «Если не можешь победить их, присоединяйся к ним!» Я начал интересоваться работой в этой области. Мне удалось раздобыть имена вице-президента по инженерным вопросам SGI и вице-президента по разработке рабочих станций Sun. Я написал им напрямую.

Я до сих пор помню свою первую встречу с Говардом Ли (Howard Lee), вице-президентом по инженерным вопросам SGI. В одно прохладное утро, когда я стоял у ресторана «Хубис» на пересечении шоссе 85 и Стивен-Крик-авеню в Купертино, ожидая его, чтобы пойти вместе на завтрак, на парковку ресторана заехал небольшой ярко-жёлтый Porsche 914. Из него вышел Говард Ли. Он явно был любителем спортивных автомобилей.

Процесс собеседования в Sun Microsystems прошел очень гладко. Мне сообщили, что в Sun Microsystems в то время действовал мораторий на найм. Однако Хол всё же принял меня на работу и направил под руководство своего технического директора Кена Окина (Ken Okin), где я приступил к разработке шины, поддерживающей несколько процессоров. В Sun Microsystems мне присвоили лишь звание старшего менеджера (на одну ступень ниже директора), поскольку ранее в одной из провалившихся стартап-компаний я дослужился лишь до должности директора.

На самом деле SGI тоже предложила мне отличную работу. На рынке рабочих станций они были прямыми конкурентами Sun. Двухдневный процесс собеседований прошел настолько гладко, что я начал беспокоиться о том, как выбрать между Sun и SGI. Во второй половине дня вице-президент их инженерного отдела провел меня по лаборатории рабочих станций. Среди множества рабочих станций и прототипов SGI я, к своему удивлению, увидел две рабочие станции Sun. Это меня очень удивило. Я спросил, почему они используют продукцию своего злейшего конкурента. Они с некоторым смущением объяснили, что один или два необходимых им инструмента для проектирования доступны только на рабочих станциях Sun. Мне показалось, что Бог дает мне знак.

18 октября 1988 года я присоединился к отделу рабочих станций Sun Microsystems.

Краткое пребывание в отделе рабочих станций

Когда я только начал работать, я заметил, что на стенах многих офисов в компании висел логотип в виде кулака с вытянутым указательным пальцем, выполненный из желтой пузырчатой пленки. Я спросил коллег, что это такое. Они ответили: «Мы только что превысили отметку в 1 миллиард долларов США по выручке!»

Под руководством Огина работали четыре команды по разработке рабочих станций, занимавшиеся проектами на разных этапах. Каждая команда состояла из менеджера и пяти-восьми инженеров. Цикл разработки был очень коротким — обычно он занимал от девяти месяцев до года. В то время технологии процессоров развивались очень быстро: каждые несколько месяцев появлялись новые, более быстрые процессоры. Это позволяло Sun выпускать новые модели рабочих станций с периодичностью менее одного года. Именно по этой причине Билл Джой, один из основателей Sun, настаивал на том, чтобы компания сосредоточилась на разработке моделей с одним процессором. Не углубляясь в проектирование более сложных и трудоемких многопроцессорных систем, Sun могла идти в ногу с развитием процессоров и выпускать продукты с более высокой производительностью. Такой подход не только укреплял конкурентоспособность компании, но и позволял нам, инженерам, ежегодно обновлять рабочие станции на наших столах.

Интересно, что такой быстрый цикл выпуска продукции приносил и другие неожиданные преимущества. Однажды у одного из клиентов произошел сбой рабочей станции при одновременном запуске определённого набора приложений. Sun потратила много времени, но так и не смогла найти причину неисправности. В это время на рынок вышла новая, более быстрая рабочая станция, в которой этой проблемы уже не было. Sun заменила клиенту старую машину на новую. И производитель, и клиент остались довольны.

Я проработал в отделе рабочих станций три месяца и завершил разработку первой спецификации Sun, поддерживающей несколько процессоров, — общей шины под названием M-bus. На самом деле для меня это было очень просто. Мне достаточно было упростить часть кода, которую я разработал и тщательно протестировал в компании Saizhong (ранее называвшейся Aixion) для операционной системы мини-суперкомпьютера Saizhong, чтобы она поддерживала многопроцессорную шину A-bus.

Именно в те три месяца, пока я осваивал работу с рабочими станциями, Оу Джин, отвечавший за разработку различных моделей рабочих станций, заметил, что основные изменения от одной модели рабочей станции к следующему поколению касаются лишь процессорного чипа и связанной с ним логики, такой как кэш-память (cache). Память на печатной плате и логика, обеспечивающая работу периферийных устройств ввода-вывода (таких как монитор, клавиатура и мышь), как правило, не меняются. Ему пришла в голову гениальная идея: разместить новый процессорный чип (который, как правило, требует более высокой тактовой частоты) и соответствующий кэш-память на одном модульном блоке (небольшом фрагменте печатной платы). Достаточно было установить разъем для такого модульного блока на материнской плате рабочей станции, чтобы она могла поддерживать процессоры нескольких поколений. Это принципиальное изменение архитектуры позволило значительно ускорить процесс исследований и разработок.

Первый процессор SPARC компании Sun был создан двумя инженерами на основе микросхемы матрицы логических элементов (gate array) японской компании Fujitsu. Последующие чипы процессоров SPARC, из-за более сложной логики и необходимости использования монолитной полупроводниковой технологии для обеспечения скорости и плотности, требовали более крупной команды разработчиков. В связи с ростом числа разработчиков ЦП, а также из-за принципиальных различий между процессами разработки полупроводников и про-

цессами разработки системных продуктов, в основном основанных на печатных платах, Sun создала новый отдел по разработке процессоров SPARC.

Как ни парадоксально, но этот творческий подход к проектированию модулей ЦП привёл к возникновению политической проблемы. Кто должен владеть правами на модули: вновь созданное подразделение процессоров SPARC или прежнее подразделение системных продуктов?

В итоге победил отдел процессоров SPARC. Он перестал просто поставлять чипы ЦП команде по системным продуктам и стал предоставлять целые модули ЦП. К сожалению, их узкий подход, основанный исключительно на модулях ЦП, привёл к системным проблемам в будущем. Однако это произошло уже много лет спустя.

Проект Sunburst

У Sun в то время фактически было два подразделения системных продуктов. Помимо подразделения рабочих станций, у Sun было ещё подразделение серверов. В ту эпоху, когда рабочие станции были на пике популярности, серверы фактически рассматривались как нечто второстепенное. О серверах задумывались только в тех случаях, когда требовалось хранить большие объёмы данных, превышающие возможности рабочих станций, или выполнять задачи, требующие длительного времени, а пользователи рабочих станций не хотели, чтобы сервер слишком долго монополизировал ресурсы рабочей станции. По этой причине серверы обычно обладали значительным объемом памяти (оперативной или дисковой) и более быстрым процессором. Кроме того, поскольку они были дороже, их обычно использовали совместно. (Поэтому с точки зрения производителя объёмы их продаж были невысокими.) Когда я только пришёл в Sun, я спросил вице-президента подразделения рабочих станций Холла Ли, что такое сервер, поскольку в университете я никогда не слышал этого термина. Он ответил: «Сервер — это рабочая станция без рук (имея в виду клавиатуру) и без головы (имея в виду монитор)». Даже производственный отдел Sun в те годы, в отличие от рабочих станций, которые называли «настольными» (desktop), долгие годы настаивал на том, чтобы серверы называли «у стола» (deskside).

В январе 1989 года отдел серверов Sun потерял своего технического директора, возглавлявшего проект Sunburst. В то время основным продуктовым проектом отдела серверов была разработка сервера под названием SunUp, в котором использовались процессоры на основе технологии ECL, производимые компанией ВІТ из Сиэтла. Логические схемы ECL отличались более высокой скоростью работы и в то время применялись только в мэйнфреймах и суперкомпьютерах. По сравнению с технологией TTL, используемой в мини-компьютерах, рабочих станциях и персональных компьютерах (ПК), ECL была гораздо быстрее, но при этом потребляла значительное количество энергии и была дорогостоящей. В связи с этим серверное подразделение Sun Microsystems запустило проект по созданию производного продукта под названием Sunburst, который имел ту же архитектуру, что и SunUp, но в нецентральных областях использовал более дешёвую технологию TTL/CMOS. Поскольку это был производный проект, его инженерная команда была гораздо меньше по сравнению с командой SunUp, насчитывавшей более сорока инженеров.

Учитывая мой опыт работы с крупными системами, Хол Ли пригласил к участию в моем собеседовании вице-президента серверного подразделения Лена Хьюза (Len Hughes). Поэтому он знал меня лично —. Вместо того чтобы нанимать нового сотрудника со стороны, он попросил Хол Ли «одолжить» меня.

Я только что завершил разработку спецификации M-bus. Рабочая станция, оснащённая материнской платой с процессором, оперативной памятью и логикой ввода-вывода (все настольные рабочие станции имели всего одну печатную плату), хотя и была интересной, но мне, как компьютерному архитектору, она начала надоедать. Поэтому, когда Холл обратился ко мне с предложением, оно показалось мне очень привлекательным — как с точки зрения более сложной архитектуры крупных машин, так и с точки зрения возможности разработки собственного продукта, — и я сразу же согласился.

С тех пор я больше не возвращался в отдел рабочих станций.

Боевой дух команды Sunburst был на низком уровне. Проект отставал от графика. Через две недели после того, как я присоединился к проекту Sunburst, мне позвонил Берни Лакрут (Bernie Lacroute) — вице-президент Sun Microsystems, курировавший всю инженерную деятельность в то время и являвшийся начальником Лина Хьюза. Он спросил меня, сможет ли проектная команда завершить проектирование основного чипа матрицы и передать его произ-

водителю в срок — до 21 мая, то есть через два с половиной месяца. Оценив текущий ход разработки и объем оставшейся работы, я, несмотря на то что очень хотел заниматься собственным продуктом и понимал, что отрицательный ответ, скорее всего, поставит под угрозу само существование проекта, честно признался, что это невозможно. Он, как и следовало ожидать, ответил, что такая задержка может вынудить компанию закрыть этот проект. Я сказал ему, что в таком случае ему лучше закрыть проект прямо сейчас, поскольку в его нынешнем состоянии он ни в коем случае не сможет достичь запланированных этапов. По крайней мере в последующий период он не воплотил свою угрозу в жизнь.

Разные подходы

С одной стороны, я хотел как можно быстрее восполнить пробелы в знаниях о продукте Sunburst и наверстать отставание в графике проекта, но в то же время я понимал, что перейти от напряжённой работы к более спокойному режиму легко, а обратно — сложно. Поэтому я продолжал ходить на работу по субботам, как и в «Сайчжун». На самом деле это довольно распространённый режим работы в стартапах Кремниевой долины. Инженеры Sunburst заметили это и тоже начали выходить на работу по субботам. В полдень мы пошли вместе пообедать, я угостил всех.

Через три недели в один из дней в мой кабинет зашел инженер из Sunburst, о котором я знал, что он одновременно посещает вечерние курсы в Стэнфордском университете. Он сказал мне: «Дэвид (мое английское имя), ты не можешь так нас гонять. Так мы умрём».

Я был в шоке. Я чуть не закричал на него: «Инженеры в Sun Microsystems работали так более четырёх лет, и никто не умер!» Но в последнюю секунду я проглотил слова, которые уже были у меня на языке. Я внезапно понял, что нахожусь в совершенно иной среде и работаю с людьми, у которых совсем другой менталитет. Те, кто приходит работать в такую компанию, как Sun Microsystems, уже выходящую на биржу, отличаются от тех, кто устраивается в стартапы.

Усовершенствование продукта

По мере того как я учился, я начал пытаться усовершенствовать этот продукт, который теперь стал моим проектом. Поскольку этот продукт позиционировался как более дешевая (и более медленная) версия SunUp, я сначала занялся его стоимостью. Основная часть затрат на аппаратное обеспечение приходилась на микросхемы матриц логических элементов, производимые компанией LSI Logic. Опираясь на опыт переговоров со многими производителями аппаратного обеспечения, накопленный за время работы в SaiZhong, я пригласил торгового представителя LSI Logic для переговоров. Я представил ему сравнительные цены от нескольких других производителей матриц логических элементов, пытаясь снизить стоимость наших чипов LSI. Представитель LSI Logic, глядя на молодого и неопытного менеджера, сидевшего напротив него, широко улыбнулся и сказал: «У SunUp и LSI Logic есть договор, охватывающий все матрицы логических элементов. Цена фиксированная». Он, очевидно, был прав, и я ничего не мог изменить.

Что ещё я мог сделать, чтобы улучшить свой продукт? Я обнаружил, что CMOS-чипы памяти, которые Sunburst использовала для своего кэша, только что вышли на рынок в новом поколении: по той же цене, но с объёмом, в четыре раза превышающим прежний. Я поручил команде немедленно доработать исходный проект с учётом использования этих новых микросхем. Теперь, хотя Sunburst по-прежнему оставался более дешёвым устройством с более низкой тактовой частотой, его производительность при запуске приложений уже могла составить конкуренцию SunUp. Это вызвало напряжённость в отношениях между командами, работающими над этими двумя продуктами.

Три варианта ИТ-услуг от Лю Дусу

Помимо самого продукта, я также задумался, можно ли сократить расходы на проект. Это позволило бы выделить больше средств на покупку материалов для прототипов или наем дополнительного персонала. Я заметил, что значительная часть всех расходов проекта приходилась на сборы, взимаемые другими подразделениями Sun Microsystems. Среди них самые высокие составляли сборы за «ИТ-услуги», взимаемые ИТ-отделом головного офиса компании. В команде Sunburst был только я, менеджер, и восемь инженеров, и сборы за ИТ-услуги, которые они вносили, казались запредельно высокими. Как говорится, молодой бык не боится тигра: пользуясь тем, что я был новым менеджером, только что пришедшим в крупную компанию, я договорился о встрече с тогдашним директором по информационным технологиям (CIO) Sun Microsystems Биллом Радучелом (Bill Raduchel), чтобы «проверить» расходы на ИТ-услуги Sunburst.

После короткой беседы я объяснил цель своего визита и вежливо выразил недовольство столь высокой стоимостью ИТ-услуг. Я спросил, есть ли возможность снизить эту стоимость. Билл, улыбаясь, посмотрел на меня, медленно открыл средний ящик своего стола и достал таблицу с тремя различными вариантами ИТ-услуг. Мой взгляд сразу же упал на стоимость каждого пакета, указанную внизу таблицы. К моему удивлению, я обнаружил, что они все одинаковы!

Это сообщение было предельно ясным. Этот господин занимал одну из самых высоких должностей в Sun. Какими бы ни были алгоритмы или логика расчёта этой платы, я не смогу достичь своей цели, да и права на переговоры у меня нет. Я поблагодарил его и молча вышел из его кабинета.

Проект был отменен

Одновременное существование двух проектов — SunUp и Sunburst — продлилось недолго. Обычно SunUp завершала составление бюджета на новый финансовый год, начинающийся 1 июля, примерно в начале июня. Будучи молодой, активной и творческой компанией, она в течение года всегда запускала больше проектов, чем могла поддерживать в долгосрочной перспективе. Результаты процедуры составления бюджета на новый финансовый год обнажили реальность. Настал момент, когда проекты пришлось закрыть. Дефицит бюджета серверного отдела, судя по всему, был не столь значительным. Отказ от гораздо более мелких производных проектов казался удобным решением.

Проект «Sun Dragon»

Однажды вечером в конце июня 1989 года Эрик Шмидт (Eric Schmidt), заместитель генерального директора, сменивший Лаку на посту руководителя всего инженерного подразделения, пригласил меня в свой кабинет. (Лаку уже ушел из Sun и стал «L» в названии стартапа HAL Computer Systems, занимавшегося разработкой нового 64-разрядного микропроцессора SPARC.) «Н» — это Хеллер (Heller), другой соучредитель его компании.) Он сообщил мне, что проект Sunburst будет закрыт. Затем он рассказал мне о проекте «SunDragon», который Sun недавно запустила совместно с Xerox.

Исследовательский центр Херох в Пало-Альто (Palo Alto Research Center, сокращённо PARC) с 1985 года занимался исследованиями в области архитектуры многопроцессорных компьютеров. Группа высококвалифицированных инженеров мирового уровня, в том числе Билл Ганнинг (Bill Gunning), создавший чип интерфейса Ethernet с тактовой частотой 3 ГГц, и главный архитектор Прадип Синдху (Pradeep Sindhu), впоследствии основавший компанию Juniper Networks, уже разработала высокопроизводительную многопроцессорную архитектуру с несколькими шинами Dynabus. Dynabus — это инновационная шина с разделением циклов (split-cycle), поддерживающая когерентность кэша (cache coherence). Они даже создали набор микросхем VLSI с матрицей логических элементов для демонстрации этой технологии. Их проект в Херох назывался «Дракон» (поскольку в западной культуре дракон — это многоголовое чудовище).

К сожалению, когда в 1988 году они обратились к руководству Херох с просьбой создать проект по разработке продукта, чтобы реально изготовить эту машину, руководство ответило, что Херох не является компьютерной компанией и не будет финансировать столь масштабную разработку. Однако из уважения к этим талантливым инженерам, а также глубоко осознавая ценность их интеллектуальной собственности, руководство Херох разрешило им найти компьютерную компанию, готовую платить роялти за право использовать эту технологию. (Менеджер этих инженеров Херох рассказал мне несколько лет спустя, что на самом деле их амбиции превосходили возможности их компании. Они были убеждены, что компьютер на базе архитектуры Dragon должен обладать достаточной производительностью, многофункциональностью и надёжностью, чтобы поддерживать высокопроизводительные цветные печатные и копируемые аппараты Херох.)

Именно в это время, в 1988 году, Sun Microsystems столкнулась с сильной конкуренцией со стороны Silicon Graphics на рынке серверов. Последняя начала выпускать многопроцессорные серверы. Многопроцессорные системы позволяют значительно расширить производительность системы по сравнению с тем, что может обеспечить один процессор. Не менее важно, что многопроцессорные системы обеспечивают резервирование. Если один процессор выходит из строя, остальные процессоры могут продолжать поддерживать работу системы. Оба этих аспекта чрезвычайно важны для серверов. В то время у Sun, можно сказать, полностью отсутствовали технические знания в области многопроцессорных систем.

PARC компании Херох и Sun Microsystems находились в Кремниевой долине. Географическая близость и взаимно выгодные общие интересы способствовали созданию идеального союза. Осенью 1988 года Sun Microsystems и Херох подписали соглашение о совместной разработке. В январе 1989 года команда Херох, состоящая из девяти инженеров мирового уровня и одного менеджера, разместилась на территории кампуса Sun в Маунтин-Вью. Вместе с несколькими недавно нанятыми инженерами Sun так родился проект Sun «Dragon». В качестве руководителя проекта Sun назначила Стива Борна (Steve Bourne), изобретателя известной оболочки Bourne Shell для операционной системы Unix.

На начальном этапе крупных проектов по разработке оборудования всегда уделяется чрезмерное внимание аппаратному обеспечению, особенно в 80-х и 90-х годах. Однако Стив, очевидно, был человеком, ориентированным исключительно на программное обеспечение. Вероятно, после нескольких конфликтов мнений эти талантливые, но довольно гордые инженеры из Херох быстро утратили уважение к руководителю из Sun. Команда Херох пригрозила вернуться в Херох PARC. Проект, казалось, вот-вот рухнет.

Эрик (который сам в своё время перешёл из PARC в Sun) попросил меня присоединиться к команде Sun Long, чтобы «оценить, есть ли у этого проекта и команды реальная ценность», а затем доложить ему.

Собеседование

Я до сих пор помню тот день... когда я предстал перед Глэдис, «управляющей» в Бонне (в высокотехнологичной индустрии Кремниевой долины мы обычно называем секретарей менеджеров или высшего руководства «управляющими»). Это считается более уважительным титулом, который также отражает более широкий круг их обязанностей, поскольку им часто приходится заботиться обо всем подразделении своего начальника.), она вручила мне двухдневный график собеседований, заполненный именами, должностями, номерами кабинетов и временем встреч. Несколько имен принадлежали руководителям из других отделов. Это был полноценный график собеседований для соискателей! Они приняли мои запросы на встречи за собеседования при приеме на работу!

Я не знал, как объяснить это этой даме. Улыбаясь, я взял ручку, вычеркнул некоторые имена и добавил несколько должностей (поскольку не знал имен людей, ответственных за эти направления). Я сказал Глэдис, что мне нужно встретиться с этими людьми, и попросил её назначить время. Я до сих пор помню выражение лица этой бедной дамы. (На её месте я, скорее всего, отреагировал бы точно так же.)

Результаты этой «собеседовательной» процедуры оказались весьма приятными. Стив объяснил мне, почему он организовал это собеседование: он не знал меня и не понимал, за что именно мне следует отвечать в рамках проекта.

Практически все ключевые должности в проекте занимали сотрудники Xerox под руководством менеджера компании Жана Гастинеля (Jean Gastinel). Жан — энергичный, увлечённый, но тихо говорящий французский джентльмен — уже много лет возглавлял проект «Dragon» в Xerox. Фактически, именно он руководил и проектом «Sun Dragon». Жан очень дружелюбно и терпеливо рассказал мне об истории проекта, его архитектуре, членах проектной команды и их обязанностях. Его знания и энтузиазм по поводу проекта «Dragon» в Sun заставили меня почувствовать, что этот проект — его жизнь. Мы очень быстро стали общаться как два старых друга, и Жан оставался моим незаменимым партнёром вплоть до того момента, когда команда Xerox в 1993 году завершила свою миссию и покинула Sun.

Проди Синду был главным архитектором проекта. Еще когда я работал в отделе рабочих станций над проектированием M-bus, я несколько раз встречался с ним на корпоративных совещаниях: он представлял отдел серверов и был изобретателем Dynabus с разделяемым циклом (шины, используемой в архитектуре системы Dragon для соединения ЦП и памяти). Во время тех встреч между нами иногда возникали профессиональные споры с оттенком соперничества по поводу двух разных архитектур шин. Спустя столько лет я до сих пор не могу забыть первые слова, которые Проди сказал мне на той «собеседовательной» встрече: «Ты пришел сюда, чтобы найти недостатки в Dynabus?»

В то время все известные мне шины, поддерживающие многопроцессорные системы, включая M-bus, разработанную мной на основе NuBus от Texas Instruments, были шинами с общим циклом (shared-cycle). В системе с шиной с общим циклом, когда один процессор получает право доступа к шине и отправляет запрос на данные из памяти, он удерживает это право до тех пор, пока не получит запрошенные данные из памяти, не позволяя другим процессорам использовать шину. Такая конструкция интуитивно очень понятна. Что ещё более важно, она обеспечивает «атомарность» (atomicity) операций чтения или записи данных, то есть никакой другой процессор не может вклиниться в текущую операцию чтения или записи. Это позволяет многопроцессорной системе с несколькими кэшами легко достигать согласованности данных (data consistency).

К сожалению, ценой этой «простоты» является низкая загрузка пропускной способности шины. Пока владелец шины ожидает, пока память выполнит его запрос на чтение или

запись, шина простаивает. Таким образом, драгоценная пропускная способность шины просто тратится впустую. Низкая пропускная способность шины означает, что она может поддерживать лишь небольшое количество процессоров. В технической терминологии это называется низкой «масштабируемостью» (scalability). В 90-е годы, поскольку скорость процессоров росла гораздо быстрее, чем скорость памяти, задержка доступа к памяти (memory latency) по сравнению со скоростью процессоров становилась всё больше. Ограничения по масштабируемости становились всё более серьёзными.

Dynabus от Херох, однако, очень похож на современные сетевые технологии. Владелец шины, отправляя запрос на чтение или запись в память, передает не только адрес цели, но и свой собственный идентификатор. Затем он освобождает шину, давая возможность другим процессорам воспользоваться ею. Когда память готова вернуть запрошенные данные, управляющая логика памяти запрашивает право владения шиной и передает запрошенные данные, используя идентификатор первоначального запрашивающего устройства. Такая характеристика разделения циклов значительно увеличивает пропускную способность шины и позволяет ей поддерживать большее количество процессоров. Для шинного проектирования того времени это было поразительным прорывом в концептуальном плане. (Вспоминая об этом много лет спустя, я убеждён, что карьера Проди в сфере сетей началась не с момента основания им компании Juniper Networks в 1996 году. Сети уже существовали в его воображении за десять лет до этого.)

Конечно, с инженерно-технической точки зрения ничего не бывает бесплатно. Реализация протокола когерентности кэша для нескольких процессоров на Dynabus гораздо сложнее. Поскольку я не имел достаточного представления о том, как работает Dynabus, я скептически относился к протоколу когерентности кэша для этой системы. Проди с улыбкой сказал мне, что может формально доказать (formally proof) эффективность этого протокола. Раньше я лишь слышал о существовании такого теоретического метода формального доказательства. Но сам я не знал, как это сделать.

Это заставило меня сдаться. Я с уважением смотрел на этого исследователя из Херох. Я верил, что он сможет.

Ранние решения по проекту Sun Dragon

Как и ожидалось, Стив поручил мне руководить разработкой аппаратного обеспечения для проекта Sun Dragon. Через шесть месяцев компания Sun объединила подразделения серверов и рабочих станций. Руководителем был назначен Хол Ли. Он предложил мне заменить Стива на посту руководителя проекта Sun Dragon, а сам стал моим непосредственным начальником.

Одномодульная системная плата

Одним из решений, которые пришлось принять на самом раннем этапе проекта, было определение позиционирования продукта. Технология «Херох Dragon», над которой мои коллеги из Херох работали с 1985 года, обладала богатыми возможностями и высокой масштабируемостью. Еще до моего прихода в проект эти эксперты из Херох решили, что вся система будет построена по модульной архитектуре. В отличие от распространенного в то время в мире супермини-компьютеров подхода, включая DEC VAX-11, где использовались отдельные платы процессора, памяти и ввода-вывода, Sun Dragon имел только одну печатную плату. Два процессора, их вспомогательные кэш-памяти, часть оперативной памяти и логика ввода-вывода — всё это находилось на одной плате. Соединительная шина всей системы, Dynabus, не просто подключалась к краю платы, как в большинстве многопроцессорных систем. Эта шина проникла внутрь платы, благодаря чему вся оперативная память и устройства ввода-вывода на всех платах, независимо от того, на какой именно плате они находились, имели равный статус на шине.

Когда я присоединился к этому проекту, мне рассказали об этом решении. Я был очень рад, потому что, как, вся команда могла сосредоточиться только на одной печатной плате, за исключением задней панели корпуса. Это также упрощало масштабирование системы. Если заказчик хотел расширить систему для повышения производительности, ему достаточно было просто установить дополнительные платы SunDragon! Спустя более двадцати лет, глядя на блейд-серверы Cisco Systems «Unified Computing System» (UCS), я вдруг осознал, что система Sun Dragon фактически является прародителем современных блейд-серверов! Единственное отличие заключается в том, что системные платы Sun Dragon вставлялись в корпус вертикально, тогда как сегодня большинство плат блейд-серверов размещаются горизонтально.

Интересно, что у такой конструкции с одной системной платой есть дополнительное преимущество. В конце 80-х — начале 90-х годов Sun была лидером на рынке рабочих станций. Основатели компании явно поощряли культуру, направленную против чрезмерной сложности системного дизайна. Это отражалось в ряде неписаных правил, касающихся всех системных продуктов. Например, время загрузки компьютера не должно было превышать тридцати секунд, а в продукте могла быть только одна (материнская) плата. Чувствуете дух рабочих станций? Ни один пользователь рабочей станции не любит сидеть перед ней, долго ожидая запуска системы. Это правило кажется вполне разумным.

К сожалению, процедура запуска системы включала проверку файловой системы и тестирование оперативной памяти. Поэтому время, необходимое для запуска, было прямо пропорционально их размеру. Объем оперативной памяти и емкость устройств хранения на серверах значительно превышали аналогичные показатели рабочих станций. Тщательное самотестирование перед запуском пользовательских приложений — это именно то, что должен делать сервер. На это требуется больше времени. Мы вынуждены были нарушить это правило. К счастью, хотя пользователи рабочих станций обычно выключают их в конце рабочего дня, серверы, по крайней мере в США, обычно не выключают. (Мы, правда, слышали, что в некоторых других частях мира в то время это делали.) Нам несложно было оправдать длительность запуска Sun Dragon, составлявшую несколько минут, тем, что перезагрузка происходила нечасто.

Другое правило — «если у продукта только один тип платы, это не так легко объяснить» — объяснить гораздо сложнее. Как компания, у которой подавляющее большинство продуктов представляло собой рабочие станции в корпусе типа «коробка для пиццы» с установленным на ней монитором, Sun в принципе не любила производить супермини-компьютеры с многоплатными стоечными корпусами! Поверите или нет, но несмотря на то, что инженеры Херох разрабатывали эту лаконичную модульную конструкцию исключительно из-за возможностей Dynabus,

совершенно не подозревая о существовании такого правила, мы на соответствующих совещаниях всё же настаивали на том, чтобы проект Sun Dragon состоял только из одной платы — « »!

Продукты Sun Dragon действительно могли работать в архитектуре, состоящей из одной единственной платы с двумя процессорами, некоторой объёмом памяти и полным набором входов-выходов. Если заказчикам требовалась более высокая вычислительная мощность или больший объём памяти, им достаточно было просто установить дополнительные (такие же) системные платы. Система ввода-вывода на системной плате представляла собой шину, называемую S-Bus, проходящую исключительно по плате. Она была изобретена одним из основателей Sun Microsystems и тогдашним вице-президентом по инженерным вопросам Энди Бекто-Сян и поддерживала четыре разъёма для модулей ввода-вывода. Заказчики могли выбрать и установить необходимые модули ввода-вывода.

Одна шина Dynabus

Dynabus фактически спроектирована таким образом, что до четырёх параллельных шин могут работать одновременно как одна шина. В связи с ограничениями, налагаемыми шириной стандартного 19-дюймового корпуса, каждая шина может поддерживать пять системных плат (всего десять процессоров). Учитывая, что в 1990 году все продукты Sun были однопроцессорными, а во всей компании — от инженерного отдела до отделов продаж, маркетинга, послепродажного обслуживания и производства — отсутствовал опыт работы с крупными системами, мы решили создать только самую минимальную архитектуру Dragon/Dynabus, то есть систему с одной шиной Dynabus. В результате, несмотря на то что архитектура Dragon/Dynabus, которой мы, партнеры Хегох, так гордились, могла поддерживать большее количество процессоров, продукты Sun Dragon будут иметь максимум десять процессоров.

Через шесть месяцев я однажды поговорил с Тоддом Линчем (Todd Lynch), моим коллегой по проекту Maxray — рабочей станции на базе M-Bus с четырьмя процессорами, — который возглавлял отдел рабочих станций. Он упомянул, что полностью укомплектованную четырехпроцессорную систему Maxray можно также использовать в качестве сервера. Во время разговора я не мог не почувствовать опасения по поводу пересечения продуктовых линеек (конкуренции), особенно учитывая, что мы оба использовали одни и те же процессорные модули, поставляемые отделом процессоров. Каждый год в конце весны в Sun начинается составление бюджета на новый финансовый год, начинающийся в июле. Как я уже упоминал, будучи молодой компанией с творческой культурой, Sun время от времени генерирует новые идеи и запускает новые проекты (не всегда официально одобренные). Процесс составления бюджета выявляет реальное положение дел, что зачастую приводит к отмене проектов и реорганизации подразделений. Некоторые проекты по своему характеру слишком схожи, особенно те, которые из-за небольшого объема производства рассматриваются компанией как второстепенные серверы, и эта опасность вызывает серьезные опасения.

Удвоение количества процессоров

Опираясь на свои знания об архитектуре Sun Dragon и модульном дизайне микросхем с матрицей логических элементов, используемом нашей командой, я принял одно из лучших решений в своей жизни: Sun Dragon удвоил количество поддерживаемых шин Dynabus, тем самым увеличив максимальное количество процессоров до двадцати! При этом нам удалось сохранить концепцию «одной платы». Благодаря модульной архитектуре Xerox модифицировать логику на системной плате оказалось на удивление просто. Системная архитектура с двумя шинами не только удвоила пропускную способность всей системы (и, следовательно, максимальное количество процессоров), но и обеспечила избыточность (redundancy), которая крайне важна для серверов. Даже в случае выхода из строя одной из шин «Sun Dragon» он сможет продолжить работу. Эта модификация сделала «Sun Dragon» более устойчивым, а ещё большее отличие от «Maxiway» с четырьмя процессорами сделало оба проекта более «безопасными».

Этот «увеличенный» Sun Dragon действительно столкнулся с проблемами через два года из-за сложностей с послепродажным обслуживанием и производственными мощностями. Об этом я расскажу позже.

Модульная архитектура Sun Dragon имела ещё один интересный эффект, приведший к появлению производных продуктов. За девять месяцев до запланированной поставки Sun Dragon в мае 1993 года мы действительно создали уменьшенную версию Sun Dragon с единственным Dynabus, используя те же самые чипы матрицы Sun Dragon. Мы назвали её «Скорпион» (Scorpion). Но это уже совсем другая история.

Транзисторная логика Ганнинга (GTL)

Среди множества инноваций «Sun Dragon» я должен упомянуть физическую технологию под названием GTL, изобретённую академиком Xerox Биллом Ганнингом (Bill Gunning). Работая в Xerox PARC, Билл первым создал сеть Ethernet со скоростью 3 мегабита в секунду. Когда он присоединился к команде Xerox, переехавшей в Sun, ему было уже 83 года. Но он по-прежнему выглядел полным энергии и сохранял ясный ум. В команде его глубоко уважали.

В 80-е и начале 90-х годов наиболее распространённой в отрасли цифровой логикой была изобретённая компанией «Texas Instruments» логика «транзистор-транзистор» (Transistor-to-Transistor Logic, или TTL). В схемах TTL напряжение +5 В соответствовало логическому «1», а 0 В — логическому «0». В конце 80-х годов постепенно стала внедряться более энергоэффективная комплементарная MOS-технология (CMOS). Однако в схемах CMOS по-прежнему использовались логические напряжения, характерные для TTL. Другая, более быстрая технология — эмиттерно-связанная логика (Emitter-Coupled Logic, или ECL) — использовалась в суперкомпьютерах и мэйнфреймах. Быстродействие ECL отчасти объясняется меньшим амплитудным диапазоном напряжения: разница между логическим «1» и «0» составляет всего 0,8 вольта. Однако ECL потребляет очень много энергии.

В матрицах логических элементов, выпускаемых компанией LSI Logic и используемых в Sun Dragon для управления кэш-памятью, оперативной памятью, вводом-выводом и шиной SBus, применялась технология CMOS. К сожалению, из-за большого амплитудного колебания напряжения между логическим «1» и «0» (5 вольт) возникали отражённые волны, которые ограничивали рабочую скорость шины. Билл Ганнинг решил сотрудничать с LSI Logic, чтобы перевести входные и выходные транзисторы матриц логических элементов, используемых в Sun Dragon, с логического напряжения TTL на логическое напряжение ECL. Так появилась новая технология транзисторной логики. В знак уважения к Ганнингу мы решили назвать её «транзисторной логикой Ганнинга» (GTL). Два года спустя IEEE стандартизировала эту технологию, сделав её доступной для всех.

Попутно отмечу, что, поскольку в проекте «Sun Dragon» широко использовались крупные микросхемы матриц логических элементов компании LSI, LSI направила молодого и энергичного менеджера по прикладной инженерии для оказания помощи проекту «Sun Dragon». Этого менеджера звали Хуан Жэньсюнь (Jenson Huang). Наше знакомство в начале 90-х годов положило начало нашей профессиональной дружбе, длившейся более двадцати пяти лет. Да, этот самый Хуан Жэньсюнь — основатель и нынешний генеральный директор компании NVIDIA.

«Ло» проекта «Sun Dragon»

Благодаря Юан-Шан Гестинару — нашему руководителю, консультанту и партнеру по обсуждению — научно-исследовательские и опытно-конструкторские работы над «Shengyang Long» ведутся по четко структурированной схеме. Мы осознаем масштаб этого проекта — это крупнейший системный продукт, когда-либо созданный компанией Shengyang, — и понимаем, что модульная архитектура, совершенно новая шина с разделением циклов, а также инновационная технология ввода-вывода GTL представляют собой смелые и весьма рискованные решения. Нам необходимо соблюдать строжайшую инженерную дисциплину и, там где это возможно, делать консервативный выбор, чтобы сбалансировать риски.

Например, в связи с быстрым развитием технологий DRAM в плане плотности и производительности, особенно с учётом длительного срока службы серверов, чипы памяти в то время не устанавливались непосредственно на материнскую плату, а на ней размещались разъёмы для установки вставных модулей DRAM. Такие модули, в которых чипы DRAM размещались по обеим сторонам небольшой тонкой печатной платы, назывались DIMM (Dual Inline Memory Module — двухрядный модуль памяти с прямолинейным разъёмом). Из-за ограничений, связанных с площадью контакта, необходимой для, и размером разъема, количество медных контактов по краям платы DIMM, соприкасающихся с внутренними зажимными пружинами разъема, было весьма ограниченным.

Поскольку эти медные контакты являются единственным каналом связи между логикой материнской платы и логикой DIMM, они представляют собой ценный ресурс. Однако мы специально спроектировали систему таким образом, чтобы соответствующие выводные и вводные медные контакты по обеим сторонам DIMM передавали одинаковые сигналы. Такой подход сократил количество доступных входных и выходных сигналов вдвое, но если пружина на одной стороне разъема ослабла и не могла обеспечить надёжный контакт, другая сторона по-прежнему могла обеспечить правильное соединение цепей, что придавало системе отказоустойчивость. Как только появились прототипы DIMM «Sun Dragon», я часто носил один из них в кармане рубашки. В течение двух лет, предшествовавших выпуску продукта на рынок, мне приходилось несколько раз в неделю проводить презентации для приезжающих клиентов Sun Microsystems, знакомя их с готовящимся к выпуску SunDragon. Во время презентаций я всегда доставал этот модуль DIMM из кармана пиджака и показывал его гостям в качестве доказательства того, что при разработке SunDragon мы уделяли внимание мельчайшим деталям. Клиентам это явно нравилось. Спустя много лет один из клиентов действительно признался мне, что они купили продукт исключительно из-за моего жеста. Клиенты — люди принципиальные. Они терпеливо выслушивают рекламные речи маркетологов, но доверяют словам инженеров.

Чтобы ещё больше укрепить команду, мы наняли Стимсона Хо (Stimson Ho), ветерана компании Amdahl, специализирующейся на производстве хост-компьютеров, который возглавил команду по технологии физической упаковки. Лео Юань (Leo Yuan), выдающийся инженер по целостности сигнала, пришёл к нам из команды IBM 3081. Наш эксперт по источникам питания Чин Чен (Chin Chen) — лицензированный инженер-электротехник штата Калифорния.

Что касается логического проектирования, то модульность системы заключалась не только в том, что вся система состояла из одного-десяти одинаковых системных плат и двух шин Dynabus для связи между ними, но и логика на системных платах также модулирована на модули ЦП/кэша, логику управления кэшем, логику управления памятью, массив модулей памяти DIMM и логику управления вводом-выводом, которые соединены между собой двумя линиями Dynabus, проходящими по плате. Вышеупомянутые три вида управляющей логики, а

также управляющая логика S-Bus, совместно используемая двумя группами логики управления вводом-выводом, реализованы на основе микросхем LSI с большими матрицами логических элементов. Каждая группа по проектированию матриц логических элементов состояла из одного опытного проектировщика логических схем и одного младшего инженера. За проектирование всей системной платы отвечал увлечённый и талантливый инженер Джефф Прайс (Jeff Price). Джефф был первым сотрудником со стороны Sun Microsystems в команде Sun Dragon.

Как только этап разработки архитектуры и спецификаций проекта был завершён, начался этап детального проектирования. В отличие от традиционного подхода, при котором коллеги проверяют проекты друг друга, мы тестировали всю систему, состоящую из нескольких системных плат (а значит, с несколькими процессорами, несколькими модулями памяти и несколькими отдельными входами-выходами S-Bus), с помощью многоуровневого моделирования. Сначала группы разработчиков микросхем матриц логических элементов проводят тестирование с помощью инструментов моделирования, предоставляемых LSI Logic. Затем группа разработчиков системных плат проводит симуляцию с использованием модели всей системной платы, включающей все протестированные модели микросхем матриц логических элементов. Наконец, группа по моделированию всей системы под руководством главного архитектора Проди тестирует многопроцессорную систему, соединённую по Dynabus на задней панели. Мы проводили не только тестирование нормального функционирования, но и проверку экстремальных ситуаций, с которыми могли столкнуться отдельные части конструкции. Мы даже настаивали на том, чтобы все разработчики нашли способы обеспечить работоспособность своих решений в условиях, которые не должны были возникнуть.

Процессор SPARC, который должен был использоваться в Sun Dragon, назывался «Viking» и все еще находился в стадии разработки одной из команд подразделения процессоров, поэтому в тестах использовалась поведенческая модель (behavior model) этого процессора. В четырехпроцессорной системе Maxray, разрабатываемой подразделением рабочих станций, также использовался тот же процессор «Viking».

Чтобы скоординировать совместную работу всех этих модулей и групп в рамках моделирования, мне понадобился подробный рабочий план и график, который я мог бы показать как себе, так и всей команде. Именно тогда мне пригодился навык, который я приобрел, работая в «Сайчжун». В «Сайчжун» использовали несколько компьютеров Apple Macintosh в качестве общего сервера для хранения графических файлов и документов. В комплекте с Macintosh поставлялось приложение под названием MacProject. Оно использовало множество прямоугольников для обозначения модулей задач. На каждом прямоугольнике указывалось время выполнения задачи и даты начала и окончания. Прямоугольники задач соединялись прямыми линиями и стрелками, обозначая последовательность и зависимости (dependency). Руководители проектов могли легко отслеживать ход работы, критический путь в графике, а также взаимозависимости между задачами.

Когда компания «Сайчжун» закрылась, я купил один из Macintosh на распродаже по ликвидации и принес его домой. Я был уверен, что «MacProject» — лучший вариант для достижения моей цели. Каждый день после работы я начинал рисовать в «MacProject» график задач и стрелки зависимостей для проекта «Sun Dragon». Я нанес в диаграмму не только этап имитационного тестирования, но и весь процесс разработки продукта вплоть до даты первой поставки клиенту (First Customer Shipment, или FCS). Примерно через неделю весь рабочий план и график, включающие все задачи, даты, потребности в ресурсах и взаимозависимости, которые я только мог придумать, превратились в одну прекрасную большую диаграмму!

Если диаграмма слишком велика и не помещается на экране монитора целиком, Project позволяет просматривать её с помощью панорамирования или по уровням. На верхнем уровне отображаются только основные модули задач. Эти основные модули задач, в свою очередь, можно развернуть на следующем уровне на более детализированные модули. Я предпочитаю

прокручивать диаграмму, так как это позволяет мне лучше видеть общую картину. Уровневое отображение, возможно, больше подходит для работы за компьютером или для руководства компании.

Поскольку я хочу, чтобы все члены команды могли увидеть этот план работы и график, я решил воспользоваться функцией увеличения всей диаграммы в проекте Майка. При печати на бумаге диаграмму можно увеличить так, чтобы люди могли видеть её толстые чёрные линии и крупный шрифт даже с большого расстояния. Для печати всего плана работы и графика «Восходящего дракона» в программе «МакПроект» потребовалось около восьмидесяти листов бумаги формата А4, причем каждый лист слегка перекрывал соседние листы по четырём сторонам.

Я принес эти восемьдесят распечатанных листов в компанию, занял пустой конференц-зал с большим столом для переговоров, обрезал ненужные части перекрытия на каждом листе, а затем склеил соседние листы. Готовая диаграмма рабочего плана и графика проекта имела длину двенадцать футов и высоту пять футов! Я выбрал коридор в центре офиса всей команды и прикрепил эту схему к стене. Каждый раз, когда какая-то задача выполнялась, я закрашивал соответствующий прямоугольник желтым фломастером. Теперь все члены команды, проходя по этому коридору, могли видеть ход выполнения всего проекта!

Многoproцессорный Sun Dragon был огромной машиной, и нам требовался очень большой объем симуляционных тестов. Для этого требовались значительные вычислительные мощности. Вскоре после того, как мы перешли к этапу симуляционных тестов, самый передовой на тот момент сервер Sun 420 только-только поступил в серийное производство. Первоначально объемы производства были весьма ограниченными. В связи с накопленным спросом стандартная процедура Sun заключалась в том, что первые выпущенные машины всегда поставлялись важным клиентам. Внутренние потребности инженерно-разработческого отдела обычно приходилось ждать шесть месяцев.

Однако проект Sun Dragon не мог ждать. Боевой дух команды был высок, а темпы разработки — высокими, но быстрый прогресс на этапе проектирования поставил наши возможности в области моделирования перед серьезным испытанием. Прежде чем передать микросхемы матриц ворот и печатные платы производителям для изготовления, нам предстояло провести огромный объем тестирования. Мы не хотели тратить много времени на отладку (debug) и переделку этого оборудования. Тщательное моделирование на раннем этапе — самый простой, эффективный и всеобъемлющий подход.

Несмотря на то, что я понимал: надежды не так уж много, я всё же заказал тридцать единиц из первой партии Sun 420. Если есть хоть малейшая надежда, я всегда делаю всё, что в моих силах, а удастся ли — решать уже Богу.

Пока я ждал результатов, однажды Эрик Шмидт, тогдашний вице-президент по инженерным вопросам, внезапно, без предварительной договоренности, пришёл с инспекцией в подразделение Sun Dragon. Возможно, вы слышали об этом Эрике Шмидте — позже он много лет проработал генеральным директором и председателем совета директоров Google.

Когда мне сообщили, что Эрик пришёл, я поспешил выбежать из офиса. Он сказал, что просто хочет прогуляться и осмотреть этот отдел. Сопровождая его, я кратко рассказал, чем сейчас занимается наша команда. Конечно, я также естественным образом упомянул о проблеме, связанной с отсутствием возможностей для имитационного тестирования. Когда мы проходили мимо коридора, где были вывешены планы проектов и графики работ, он обратил на них внимание. Он спросил меня, что это такое, и я ему объяснил. Я заметил, как его лицо просветлело. Полагаю, поскольку почти все проекты по рабочим станциям Sun были небольшими — команды насчитывали менее десяти человек, а время разработки не превышало одного года — он, вероятно, никогда раньше не видел такой дисциплины в выполнении графиков.

Через два дня мне сообщили, что из 39 единиц Sun 420, выпущенных в первый месяц, нам выделили 19! Когда эти 19 ящиков размером с холодильник доставили и выстроили в ряд в коридоре возле моего кабинета, не говорите, как я был счастлив! Это было гораздо ценнее, чем похлопывание по плечу со стороны начальства. Конечно, такая щедрость также сопровождалась определенными ожиданиями. Если я скажу вам, что не чувствовал при этом давления, я вас обману.

Чтобы ещё больше стимулировать инженеров-проектировщиков на поиск ошибок в проекте с помощью тщательных имитационных тестов, я послал человека в китайский квартал Сан-Франциско за большим медным гонгом, который можно было бы повесить. У менеджера, которого я послал, был бойфренд-художник. Он добровольно нарисовал на гонге большого цветного дракона, схватившего двумя лапами двух насекомых. Мы поставили этот большой медный гонг в коридоре, где висел график, и сказали всей команде: если они найдут хотя бы одну ошибку — будь то в своём собственном проекте или в других частях системы — они могут ударить по гонгу один раз. Каждый раз, когда раздавался звук гонга, все члены команды высывали головы из своих кабинетов и аплодировали. Обнаружение ошибки в проекте — будь то собственная или чужая — является большим вкладом в проект, а не поводом для стыда!

Но этого было недостаточно: мы даже разработали желтую круглую наклейку с изображением черной ошибки, перечеркнутой большим крестом. Мы наклеивали эту наклейку на косяк двери кабинета того, кто обнаружил ошибку, и сразу же выплачивали ему или ей пять долларов в качестве вознаграждения.

Через шесть месяцев, благодаря интенсивным симуляционным тестам всей команды, дверные косяки многих инженеров были полностью заклеены желтыми наклейками. Однако совершенно неожиданное последствие в конце концов доставило мне неприятности. Когда всей проектной команде пришлось переехать в другое здание (такое часто происходит при расширении команды или реорганизации компании), менеджер по эксплуатации зданий прибежал в мой кабинет и в ярости накричал на меня. Оказалось, что эти желтые наклейки очень легко наклеивать, но чрезвычайно трудно отклеивать!

Получение этих Sun 420 не полностью решило наши потребности в вычислительных мощностях для симуляционных тестов. Всегда хочется проводить ещё больше симуляционных тестов. Нам требовались дополнительные вычислительные мощности для симуляций. Один из нанятых нами опытных консультантов по диагностике, Джим Шимандл (Jim Schimandle), придумал решение. Он написал программу, которая предоставляла свободные серверы любому инженеру, которому требовались ресурсы для моделирования. Если на тот момент в системе «» не было свободных серверов, задача моделирования добавлялась в список ожидания. Таким образом, инженеры могли отправлять запросы на моделирование в любое время, когда в этом возникала необходимость, не тратя время на поиски или ожидание. В конце концов, когда задача моделирования завершалась или, к сожалению, заканчивалась неудачей, программа уведомляла их об этом по электронной почте.

Что касается предложения серверов, помимо серверов, принадлежащих нам, я посетил все инженерные подразделения компании, серверы которых подключены к корпоративной сети. Поскольку различные проекты обычно находятся на разных этапах разработки, не все эти серверы загружены. Кроме того, наша программа позволяет серверам ограничивать время, в которое они могут использоваться в течение дня. Ни один руководитель отдела не любит, когда его серверы берут напрокат и уносят из их помещения, ведь вернуть их вовремя бывает непросто, а порой и вовсе невозможно. Однако предоставление времени использования компьютера исключительно через внутреннюю сеть компании воспринимается гораздо спокойнее, не говоря уже о том, что можно самостоятельно указать время начала и окончания использования в течение дня. Когда их проекты доходят до этапа, когда требуются серверы, мы в любой момент можем прекратить такое сотрудничество.

Наверное, это был один из тех периодов в моей профессиональной жизни, когда мне приходилось «просить милостыню».

Опыт найма, которым я горжусь

Однажды днём в конце апреля 1990 года мне позвонил Ашок Сингхал (Ashok Singhal), кандидат наук по компьютерной инженерии из Калифорнийского университета в Беркли. Этот парень рассказал мне, что, будучи занятым завершением докторской диссертации, он пропустил проведённые прошлой осенью на территории кампуса Беркли мероприятия по набору выпускников следующего лета, в которых участвовали крупные компании, включая Sun Microsystems. Он подчеркнул свой интерес к Sun Microsystems и выразил надежду на возможность пройти собеседование.

Исходя из моего уважения к программе обучения компьютерной инженерии в Калифорнийском университете в Беркли, я договорился с ним о встрече через неделю. Я обнаружил, что он был очень компетентным компьютерным архитектором. Встреча была запланирована на час, но в итоге мы беседовали более двух часов. Он принес с собой свое резюме, в котором в качестве рекомендателей были указаны три всемирно известных профессора: Дэйв Паттерсон (Dave Patterson), Йель Патт (Yale Patt) и Алан Смит (Alan Smith). Впервые в жизни я принял его на работу в группу системной архитектуры Sun Dragon прямо на месте. Позже Асук продемонстрировал такие результаты, что его прием на работу стал одним из самых гордых моментов в моей профессиональной карьере. Впоследствии он не только стал ведущим архитектором чрезвычайно успешного второго поколения серверов Sun — моделей Enterprise 6000/5000/4000/3000 — после того, как команда из Xerox вернулась в Xerox PARC, но и был соучредителем и техническим директором (CTO) компании 3Par Data, занимающейся продуктами для хранения данных, за которую в 2010 году развернулась громкая борьба между HP и Dell.

Когда Асук осенью окончил Беркли и присоединился к нам, все должности, связанные с проектированием микросхем и системных плат, уже были заняты. Мы как раз начинали переключать внимание на подсистему ввода-вывода на базе SBus. SBus — это эксклюзивная проприетарная шина ввода-вывода компании Sun. На рынке не было готовых решений. Я также хотел проверить практические навыки Асука, поэтому поручил ему разработать SCSI-контроллер дисков на базе SBus. В итоге он справился с задачей всего за три месяца.

Чтобы укрепить команду и создать системные продукты, которые действительно можно было бы вывести на рынок и которые смогли бы удержаться на нём, в конце 1992 года я пригласил своего бывшего коллегу по «Сайчжун» Гэри Бека (Gary Beck) возглавить направление «продуктовой инженерии» в рамках проекта. Гэри был опытным ветераном компании «Берроуз» (Burroughs). В 70-е годы «Берроуз» была очень успешной компанией-производителем мэйнфреймов, особенно известной в финансовом секторе. В 1985 году я фактически пригласил Гэри из «Берроуз» в «Сайчжун», чтобы он отвечал за разработку «сердца» машин «Сайчжун»: вычислительного процессора на основе архитектуры с очень длинными словами команд (VLIW). Он отлично справился с задачей и под его руководством команда успешно выполнила поставленную задачу.

Длительный срок разработки ставил проект «Sun Dragon» перед особыми испытаниями в условиях циклических колебаний бюджета, сокращения проектов и реорганизации в компании Sun Microsystems. Проект поддерживал доверие высшего руководства исключительно благодаря потенциалу продукта «Sun Dragon» на рынке, а также высокому моральному духу и прогрессу команды разработчиков. Иногда я думал, что, возможно, именно тот факт, что это был совместный проект Xerox и Sun Microsystems, затруднял любые попытки покончить с проектом «Sun Dragon».

Кризис нехватки процессоров

К весне 1992 года у нас было готово всё аппаратное обеспечение, не хватало только процессора «Викинг». В рамках возможностей аппаратного тестирования микросхемы матриц логических вентилях, системные платы, задняя панель, система питания и механическая сборка соответствовали проектным требованиям. Ничто не требовало доработки. В этом и заключалась сила тщательного моделирования и тестирования.

На этом этапе нам крайне необходим процессор «Viking» (известный под названием SuperSPARC) для полной системной интеграции и тестирования систем реального времени. Согласно общему графику компании, по идее, к моменту, когда все новые продуктовые проекты, в которых планировалось использовать процессор «Viking» (включая рабочие станции и серверы), дойдут до этапа, требующего его применения, этот процессор уже должен был быть готов. К сожалению, как и в случае со многими сложными проектами, его реализация отставала от графика. Что ещё хуже, из-за конкуренции за передовую производительность технологии следующего поколения в полупроводниковой отрасли всегда продвигаются за счёт высокопроизводительных микропроцессоров.

Компания Texas Instruments, производящая чипы для Sun, использовала именно процессоры SPARC для разработки своих новых технологий. С началом каждого нового поколения технологий производства микросхем, соответствующего закону Мура (Moore's Law), инженеры-микротехнологи команды Sun Microsystems, занимающейся процессорами SPARC, собирались вместе с инженерами Texas Instruments, чтобы совместно разработать подробные правила компоновки микросхем (layout), которые могли бы эффективно производиться с использованием новейшего производственного и испытательного оборудования Texas Instruments. Процессоры SPARC компании Sun фактически становились «подопытными кроликами» для новых технологических процессов производства микросхем «Texas Instruments».

В конечном итоге эти новые процессы могли применяться и в других продуктах «Texas Instruments». Именно по этой причине «Texas Instruments» взимала с Sun лишь половину разовых инженерных затрат (Non-Recurring Engineering, NRE) за ранние партии прототипов. К сожалению, это партнерство, позволившее Sun Microsystems сэкономить на затратах на разработку процессоров SPARC, привело к тому, что время, необходимое для освоения нового технологического процесса производства микросхем Texas Instruments, было добавлено к общему сроку разработки чипа. Многократные переделки чипа — то для исправления ошибок в проекте Sun Microsystems, то из-за незрелости технологического процесса Texas Instruments — превратили ранние поставки процессоров Viking в настоящий кошмар. Более полугода каждый четверг днём все подразделения компании, которым требовались процессоры Viking, собирались на совещание с представителем процессорного подразделения Texas Instruments Терри Ганноном (Terry Gannon), чтобы бороться за квоты — и это в тех случаях, когда удавалось получить исправные чипы. Хуже того, все проекты рабочих станций того поколения, за исключением одного, требовали всего одного исправного процессора Viking для сборки одного прототипа. Крупнейший проект рабочей станции, Maxiray, действительно требовал четырех процессоров, но Sun Dragon же был сервером на двадцать процессоров!

В мае 1992 года мы наконец получили наш первый исправный процессор Viking. В тот момент, когда однопроцессорная система Sun Dragon запустила операционную систему Solaris, вся команда сошла с ума от радости. Чтобы компьютер загрузил свою операционную систему, необходимо, чтобы практически все его системные компоненты — включая процессор, кэш, оперативную память и даже устройства ввода-вывода — работали исправно. Появившаяся на мониторе при запуске надпись «Привет, мир!» была точно как первый крик новорожденного!

Для компании, занимающейся разработкой рабочих станций, появление надписи «Привет, мир!» обычно считается важной вехой, знаменующей конец периода высокого риска и неопределённости в графике проекта. К сожалению, для крупной многопроцессорной системы работа на одном процессоре — это лишь начало более сложного процесса тестирования и поиска ошибок во взаимодействии аппаратного и программного обеспечения в многопроцессорной среде. Чтобы ускорить этот процесс, в рамках системных проектов обычно создаётся несколько прототипов, чтобы несколько групп специалистов могли одновременно тестировать свои разработки. Но это ещё больше увеличивает наш спрос на процессоры Viking.

Конец ознакомительного фрагмента.

Текст предоставлен ООО «Литрес».

Прочитайте эту книгу целиком, [купив полную легальную версию](#) на Литрес.

Безопасно оплатить книгу можно банковской картой Visa, MasterCard, Maestro, со счета мобильного телефона, с платежного терминала, в салоне МТС или Связной, через PayPal, WebMoney, Яндекс.Деньги, QIWI Кошелек, бонусными картами или другим удобным Вам способом.