

# Взлом Xbox

*Введение в обратную разработку*

*(Reverse Engineering)*

Эндрю «Банни» Хуанг

by KONOVALOV



**Книгу перевел инженер-программист Коновалов Денис.**

«За время перевода данной книги я успел – сменить фамилию, отчество и жениться. Перевод выполнен максимально **точно**, без отсебятин. Все фразы переданы дословно. Например, в книге Линуса Торвальдса - 'Just for Fun' в официальном русском переводе первая глава названа «Рождение Хакера», при том, что Линус никакого отношения к хакерам не имеет и в оригинале глава называлась — «Рождение ботаника (отличника)». Такой отсебятину я, как переводчик не приемлю.

Однако... в книге будут комментария и дополнения.»

{мои комментарии написаны в фигурных скобках}

**Данная книга не является руководством по взлому консолей. Автор перевода не имеет никакого отношения ко взлому. Книга предназначена для тех, кто хочет изучить принципы работы аппаратного обеспечения.**

**Взлом консолей преследуется законом!**

“Я должен подчеркнуть, что эта книга не нарушает авторские права Microsoft, и представленные в ней знания не позволяют напрямую обойти защиту авторских прав. Чтобы совершить акт нарушения, нужно значительно улучшить свои навыки и применить значительное количество навыков и специальных знаний и ноу-хау, направленных специально на обход контроля за соблюдением авторских прав. Утверждение, что эта книга является инструментом обхода защиты, равносильно утверждению, что все книги о схемах, встроенном программном обеспечении или криптографии также являются инструментами обхода защиты.” – автор книги “Банни”.

С уважением, Коновалов Д.А.

[https://github.com/KONOVALOVda/HackingTheXbox\\_RUS](https://github.com/KONOVALOVda/HackingTheXbox_RUS)

Уважаемый Читатель,

Спасибо, что скачали и прочитали эту книгу.

No Starch Press и я решили выпустить эту бесплатную электронную версию книги «Hacking the Xbox» в честь Аарона Шварца

*«Аарон покончил с собой в январе 2013 года, подвергнувшись преследованию пострадавших органов за скачивание объявлений о научных статьях из студенческой сети Массачусетского технологического института (МТИ), нарушив при этом десятки статей уголовного кодекса»*

Читая эту книгу, я надеюсь, что вы вспомните, насколько важна свобода для хакерского сообщества, и что вы, по-видимому, поддерживаете дела, в которые верил Аарон.

Я согласен выпустить эту книгу бесплатно, частично потому, что отношение МТИ к Аарону мне не чуждо. В этой книге вы узнаете историю о том, как, будучи аспирантом Массачусетского технологического института, я извлек ключи безопасности из оригинальной Microsoft Xbox. Вы также прочтете о сокрушительном разочаровании, которое я получил, получив письмо от судебной службы Массачусетского технологического института, отказывающегося от какой-либо связи с моей работой, фактически оставив меня одним помощником Microsoft.

Разница заключалась в том, что преподаватели моей лаборатории (Лаборатории искусственного интеллекта), были возмущены таким отношением ко мне. Преподаватели открыто выступили против юридического отдела Массачусетского технологического института и предложили опубликовать мою работу в качестве официальной документальной лаборатории «Memo AI Lab», что

дало мне больше шансов на более успешный уровень с Microsoft. Microsoft, осознавая возможную негативную реакцию общественности в случае иска против юридического академического исследователя, согласилась на мировое решение вопроса.

*«Эндрю был студентом МПГ, когда начал экспериментировать со взломом игровой приставки Microsoft. Он опубликовал в интернете ключи безопасности, которые ему удалось извлечь из оригиналной Xbox. После этого студент получил письмо из юридического отдела МПГ с уведомлением, что институт не будет заступаться за него, и Эндрю предстоит самостоятельно разбираться с юридическими претензиями Microsoft»*

Меня печалит то, что так называемое правительство Америки, созданное «для народа, от народа и во имя народа», проявляет меньшие сострадания и просвещенности к своим согражданам, чем корпорация. Я был участником юридического давления со стороны других организаций, и мне хорошо известно, насколько отвратительным и душераздирающим может быть судебный процесс с высокими ставками. К счастью, ставки в моих делах были не так высоки, и мои противники не были столь грозными, как у Аарона, иначе я тоже мог бы поддаться безнадежности и страху. Несколько лет назад я начал восстанавливать свою жизнь за границей и нахожу некоторое утешение в мысли, что мое проживание за границей делает меня немного труднодоступным для судебных исков.

Хотя правовая система США стремится к справедливости, правила системы создают асимметричную войну, которая благоприятствует тем, у кого есть ресурсы. Одним из наиболее эффективных методов принуждения к заключению соглашения, независимо от того, правильное оно или нет, против небольшого игрока, является простое выкачивание из него ресурсов и воли к борьбе через досудебные манипуляции. Кажется, что вся ваша жизнь находится под пристальным наблюдением, и любая мелочь превращается в повод для тяжбы — с ходатайствами, встречными заявлениями, запросами и вызовами в суд. Каждый такой шаг добавляет тысячи долларов к вашим юридическим расходам. Ваши друзья, коллеги, работодатели и семья оказываются вовлечеными в этот цирк унижения в качестве свидетелей. Хуже того, вам советуют не говорить откровенно ни с кем, чтобы их не вызвали в суд в качестве свидетелей против вас. Изолированному и испуганному, вам в конечном итоге становится разумнее пойти на мировую, чем рисковать проиграть дело технически против более обеспеченного противника, независимо от справедливости.

Правительство США — это, без сомнения, самая богатая и устрашающая сила в мире, а законы об авторском праве предусматривают крайне жестокие наказания. Я лично не знал Аарона, но считаю, что то давление, которому он подвергался, сыграло роль в его решении покончить с жизнью.

Я разделяю мнение Ларри Лессига о том, что юридическая система США нуждается в стыде.

{Ларри Лессиг, известный профессор права и основатель Creative Commons, высказывался о необходимости того, чтобы правовая система США испытывала чувство стыда за свои действия, особенно в контексте чрезмерных и несправедливых наказаний, связанных с делами об авторских правах и других формах юридического преследования. Одна из его наиболее известных цитат по этому поводу:

"There is no justice in following unjust laws. But there is great power in saying no to them." -

«Нет справедливости в соблюдении несправедливых законов. Но есть великая сила в том, чтобы сказать им "нет".»}

Для такого стороннего наблюдателя, как я, кажется, что некоторые прокуроры в правительстве США одержимы идеей сделать себе имя за счет преследуемых ими лиц. Выигранные дела приносят им признание и авторитет, необходимые для повышения по службе и назначения на все более громкие дела. Для них дело не в справедливости, а в победе и самовозвеличании.

Эта система стимулов способствует бесстыдному издевательству над отдельными людьми и малыми организациями, у которых хватает смелости встать и сделать что-то смелое. У отдельных людей отнимают волю и силу бороться за то, что они считают правильным, поскольку сам факт преследования может быть таким же наказанием, как и приговор. В результате я опасаюсь, что эпоха гражданского неповиновения может подойти к концу.

Как люди, как личности, как хакеры, мы должны противостоять этой тенденции и продолжать делать то, что мы чувствуем в глубине души правильным. Хотя история Аарона подошла к трагическому концу, я надеюсь, что в этой книге вы найдете обнадеживающую историю со счастливым концом. Без права

возиться и исследовать мы рискуем стать рабами технологий; и чем больше мы будем пользоваться правом на взлом, тем сложнее будет отнять его сразу.

Сингапур, март 2013 г.

## Взлом Xbox

Введение в обратную разработку

**Неограниченное издание**

# Взлом Xbox

Введение в обратную разработку

## Неограниченное издание

Эндрю «Банни» Хуанг



## No Starch Press, Inc.

Сан-Франциско

ВЗЛОМ XBOX. Авторские права © 2003 Xenatera LLC.

Некоторые права защищены. Эта работа лицензирована в соответствии с  
лицензией Creative Commons Attribution-NonCommercial-ShareAlike. Чтобы  
просмотреть копию этой лицензии, посетите

<http://creativecommons.org/licenses/by-nc-sa/1.0/> или отправьте письмо по адресу Creative Commons, 559 Nathan Abbott Way, Stanford, CA 94305, USA.

Издатель: Уильям Поллок

Управляющий редактор: Кароль Хурадо

Дизайн и верстка: Xenatera LLC

No Starch Press и логотип No Starch Press являются зарегистрированными товарными знаками No Starch Press, Inc. Другие названия продуктов и компаний, упомянутые здесь, могут быть товарными знаками их соответствующих владельцев. Вместо того, чтобы использовать символ товарного знака при каждом упоминании товарного знака, мы используем эти названия только в редакционной манере и в интересах владельца товарного знака, без намерения нарушить товарный знак.

Для получения информации о дистрибуторах книг или переводах обращайтесь напрямую в компанию No Starch Press, Inc.:

No Starch Press, Inc.

Ринголд Стрит, 38, Сан-Франциско, Калифорния 94103 США

Телефон: 415-863-9900; Факс: 415-863-9950; [info@nostarch.com](mailto:info@nostarch.com) ;

<http://www.nostarch.com>

Информация в этой книге распространяется на условиях «как есть», без гарантии. Хотя при подготовке этой работы были приняты все меры предосторожности, ни автор, ни No Starch Press, Inc. не несут никакой ответственности перед любым лицом или организацией в отношении любых потерь или ущерба, причиненных или предположительно причиненных прямо или косвенно содержащейся в ней информацией.

ISBN 1-59327-029-1

*В память об Аароне Шварце*



## Благодарности

Я хотел бы поблагодарить моих преданных и заботливых родителей за то, что они воспитали меня таким человеком, каким я являюсь сегодня.

Я также хотел бы поблагодарить сообщество интернет-хакеров за их советы и рекомендации, особенно тех, кто вынужден действовать анонимно из-за страха преследования со стороны правительства или мести со стороны своего работодателя.

Ли Тьен из Electronic Frontier Foundation, Джозеф Лю из Boston College Law School, а также доктор Том Найт и профессор Хэл Абельсон из лаборатории искусственного интеллекта Массачусетского технологического института заслуживают особой благодарности за помочь в процессе публикации моей оригинальной статьи о системе безопасности Xbox. Я бы никогда не опубликовал ее, если бы не их поддержка и советы.

Я также в долгу перед командой Xbox-Linux: Майклом Штейлом, Мишем Мериаком, Францем Ленером (спасибо за все подробные технические обзоры!) и замечательным Энди Грином (он же numbnut) за то, что они так подробно рассказали о последних хаках Xbox и предоставили такой интересный материал для книги. Огромное вам спасибо, ребята; продолжайте в том же духе. Я также хотел бы поблагодарить Дэна Джонсона (он же SiliconIce), основателя XboxHacker.net BBS, за создание XboxHacker.net BBS и за его интересные материалы для книги, а также за его очень полезный технический обзор, советы и поддержку. Также спасибо Герхарду Фарфеледеру за предоставление фотографии команды Xbox-Linux.

Благодарим Тимоти Чена из Via Technologies, Inc. за вклад Материнская плата P4M266 для сравнения Xbox и ПК и за его увлекательный взгляд на индустрию ПК. Я также хотел бы поблагодарить Xilinx за их щедрые пожертвования FPGA через университетскую программу Xilinx.

Вы знаете, кто вы, и вы знаете, как вы мне помогли: xor, adq, luc, head, visor, roastbeef, kgasper, xerox, lordvictory, pixel8, El (GCN), tom из Гонконга и sween (Scotch!).

Еще больше благодарностей за неограниченный тираж этой книги: Билл Поллок из No Starch Press заслуживает особой благодарности за то, что смело взялся за печать и распространение книги; Пол Юн заслуживает сердечной благодарности за многочисленные исправления опечаток; а Раэль Дорнфаст и Тим О'Reilly & Associates, Inc. также заслуживают особой благодарности за их полезные советы и поддержку.

*{а также хочу поблагодарить ведущего канала «Restart» - Илью Геймера, за то, что сообщил, что такая прекрасная книга еще не переведена}*

# Пролог README.1ST

Консоль Xbox™ от Microsoft® — это действительно крутое устройство, и дело не только в том, что она запускает новейшие видеоигры. Мощная и доступная по цене Xbox может использоваться не только как игровая консоль, но и как ПК, мультимедийный центр или даже веб-сервер. К сожалению, очень мало книг, которые учат, как изучать и модифицировать современное электронное оборудование, такое как Xbox. Большинство учебников по электронике сосредоточены на теории, но для реального хакерства нужны практические навыки и знания. К тому же, те немногие практические книги по взлому железа, которые вдохновляли меня в детстве, давно устарели из-за стремительного развития технологий. Эта книга написана, чтобы восполнить пробел и стать практическим руководством для понимания и обратного проектирования современных компьютеров — настоящим пособием для нового поколения хакеров.

Главное, что даёт взлом Xbox, — это обучение. Как говорится, «Дай человеку рыбу, и он будет сыр один день; научи его ловить рыбу, и он будет сыр всю жизнь». Поэтому в этой книге особое внимание уделено основным методам хакерства — пайке, обратному инжинирингу, отладке — для начинающих хакеров, при этом приводятся справочники и советы, которые могут быть полезны и более опытным хакерам. Xbox послужила хорошим учебным материалом как для специалистов по безопасности, так и для хакеров: не потому, что она пример отличной защиты, а потому, что это массовый продукт, созданный крупной компанией, которая недавно заявила, что её приоритетом стала безопасность. Опыт работы с Xbox показывает, что создание надежных устройств в условиях враждебной пользовательской среды — задача сложная, даже для большой и хорошо финансируемой компании. Один из выводов заключается в том, что сложность и риск создания дешёвых и надёжных устройств накладывают ограничения на важность секретов, которые можно доверить такому оборудованию. Кроме того, Xbox служит хорошим примером для обучения, так как на момент написания книги в мире насчитывалось почти 10 миллионов почти одинаковых устройств. Сходство архитектуры Xbox с обычным ПК делает взлом Xbox ещё более ценным с образовательной точки зрения, поскольку большая часть обсуждений в этой книге также применима к более широким темам, связанным с ПК.

Ещё один интересный аспект взлома Xbox — это подпольное сообщество хакеров, которые следуют за этой консолью. Люди, которые взламывали Xbox и приобрели соответствующий опыт, будут актуальны ещё долго после того, как Xbox превратится в пыльную вещь на распродаже. Поэтому в этой книге уделено внимание социальному аспекту. Я включил профили некоторых личностей, занимавшихся взломом Xbox. Надеюсь, что эти примеры вдохновят людей взять в руки отвертку и паяльник и начать заниматься хакерством. Привить такой исследовательский дух молодым поколениям важно в долгосрочной перспективе, чтобы сохранить резерв талантливых инженеров, которые привели технологическую революцию к тому, где она находится сегодня. Многие из

сегодняшних инженеров начинали с взлома и возни с радиолюбительскими устройствами, телефонами и компьютерами, которые тогда поставлялись с полными схемами и исходным кодом. Этот резерв инженерных талантов необходим для поддержания здоровой экономики и национальной безопасности в эпоху компьютеров.

## The Рынок игровых консолей

2002 год был отмечен потрясениями не только за рубежом, но и на рынке технологий; продажи ПК упали, серверный бизнес сократился {в I квартале 2002 года мировой рынок наиболее мощных серверов сократился до 10,5 млрд. долл., что означает 15%-ное падение}, а рынок телекоммуникаций, за некоторыми исключениями, выглядел удручающе.

{В России наоборот был прирост. По данным IDC, российский рынок ПК в 2001 году вырос на 22%, составив 1,8 млн. компьютеров. На 2002 год IDC прогнозировала продолжение устойчивого роста российского рынка ПК, однако несколько медленнее, чем в предыдущие два года. И эти прогнозы оправдались.}

Несмотря на медвежий рынок технологий, рынок оборудования, программного обеспечения и аксессуаров для видеоигр имел знаменательный год, достигнув общего объема продаж в размере 10,3 млрд долларов — на 10% больше, чем в 2001 году.<sup>1</sup> Это сопоставимо с объемом продаж звукозаписывающей индустрии в США в 2001 году, составившим 13 миллиардов долларов.

Хотя рынок видеоигр велик, вести прибыльный бизнес по продаже консолей — задача нелегкая. Покупатели видеоигр разборчивы, следят за модой и экономят деньги. Они требуют, чтобы консоль была мощной и привлекательной, но стоила как ужин в ресторане или визит к врачу. Такое сочетание экономии и ожидания высокой производительности вынуждает производителей консолей продавать своё оборудование в убыток. В результате, компании используют бизнес-стратегию «закрытой консоли»: консоль продаётся с минимальной прибылью или даже в убыток, а основная прибыль поступает от последующих продаж игр для этой консоли. Такая стратегия требует значительных первоначальных вложений в разработку аппаратного обеспечения консоли и в рекламу. Именно на производителе консоли лежит ответственность за создание рынка для своего оборудования, чтобы разработчики игр были уверены в том, что их вложения в платформу оправдаются.

Уловка-22 в том, что никто не захочет покупать консоль, на которой мало игр. Таким образом, риск создания и разработки миллионов единиц оборудования, а также сотни миллионов долларов предварительных убытков, связанных с этим оборудованием, практически полностью ложится на производителя консоли. В результате на сегодняшний день в бизнесе игровых консолей есть только три игрока: Sony, Nintendo и Microsoft. Из этих трех игроков Sony лидирует на рынке

---

<sup>1</sup>Источник: NPDFunworld

### Взлом Xbox: Введение в обратную разработку

консолей, в то время как Nintendo, выпустив линейку Gameboy, закрепилась на рынке портативных консолей. Microsoft - новый игрок на рынке игровых консолей. Гонка за второе место пока не определена. В начале 2003 года продажи Gamecube опережали продажи Xbox в Японии и Европе, в то время как Xbox сохранял преимущество над Gamecube на огромном североамериканском рынке.

{Фраза "Уловка-22" (Catch-22) происходит из одноименного романа американского писателя Джозефа Хеллера, впервые опубликованного в 1961 году. В книге "Уловка-22" описывается ситуация во время Второй мировой войны, когда американские летчики сталкивались с парадоксальной и абсурдной логикой военного руководства.

- Летчик, который продолжает выполнять боевые вылеты, несмотря на опасность, считается сумасшедшими и может быть отстранен от полетов, если он сам подаст соответствующее заявление.
- Однако, если летчик подает заявление, чтобы его освободили от полетов, это доказывает, что он здравомыслящий и заботится о своей безопасности, что делает его вменяемым и способным к службе.
- Следовательно, такой человек не может быть освобожден от полетов, потому что он здравомыслящий, несмотря на то, что подал заявление.}

Решающее значение для успеха закрытой модели бизнеса консоли имеет идея привязывать потребителей к покупке только одобренных игр с выплатой роялти. Другими словами, пиратство и неодобренные игры могут разрушить прибыльность бизнеса. Следовательно, консоль должна использовать механизмы безопасности, которые препятствуют копированию игр и разработке и распространению неодобренных игр. Провал Sega Dreamcast является ярким примером того, что происходит, когда механизмы безопасности выходят из строя.

Dreamcast был выпущен в Японии в ноябре 1998 года. Производственные проблемы с чипом NEC PowerVR2 DC, графическим ускорителем, используемым в Dreamcast, ограничили первоначальные поставки. Последующие три года стали для Dreamcast настоящими американскими горками. Такие популярные игры, как Soul Caliber, Dead or Alive 2, Resident Evil, Crazy Taxi и Shenmue, поддерживали популярность Dreamcast, в то время как запуск Playstation2 подорвал

продажи Dreamcast и, в конечном счете, доверие разработчиков программного обеспечения. По иронии судьбы, качество графики Dreamcast было эквивалентно или превосходило качество ранних игр Playstation2, таких как Dead or Alive 2, несмотря на дополнительную мощность, заложенную в Playstation2. (Playstation2 сложно программировать, и разработчикам потребовалось несколько лет, чтобы полностью раскрыть ее потенциал.)

Последний гвоздь в гроб Dreamcast был забит весной и летом 2000 года. Немецкая хакерская группа Team Utopia обнаружила черный ход внутри маски-ROM BIOS Dreamcast, который позволял Dreamcast загружаться со стандартного CD-ROM. Номинально Dreamcast использует фирменный формат под названием «GD-ROM»

для распространения игр. Формат GDROM нельзя скопировать с помощью стандартных CD или DVD-приводов. Однако черный ход в ROM BIOS Dreamcast позволил пиратам в конечном итоге создавать монолитные образы CD-ROM видеонгр, которые можно было загрузить без какой-либо необходимости в модификации оборудования. Кто будет платить за игру, если ее можно бесплатно скачать в Интернете? Последовавшее за этим безудержное пиратство снизило продажи игр, отбив у разработчиков желание разрабатывать игры для консоли и нанося ущерб бизнесу Sega. Было продано шесть миллионов единиц, и примерно через три года после ее запуска Dreamcast была снята с рынка. Теперь Sega занимается исключительно разработкой игр и даже делает игры для своих бывших конкурентов Sony и Nintendo, а также Microsoft.

Хотя из опыта Dreamcast можно извлечь много уроков, этот посыл ясен: возможность запускать код из почти бесплатных источников, таких как CD-R, DVD-R или сеть, без существенных изменений оборудования — это поцелуй смерти для любого консольного бизнеса, основанного на закрытой модели консоли. Это жестокая проблема для Microsoft Xbox, поскольку он построен на стандартном оборудовании ПК, изначально разработанном для открытого использования и запуска кода, загруженного из многочисленных источников. Следовательно, судьба Microsoft на рынке консолей тесно связана с успехом и надежностью системы безопасности Xbox. Система безопасности пока держится довольно хорошо: все обнаруженные уязвимости требуют, как минимум установки беспаечной модификации, которая аннулирует гарантию. Необходимость в изменениях оборудования ограничивает практическое влияние этих уязвимостей, поскольку большинство пользователей боятся снимать крышку со своих устройств. Однако существует сильное желание со стороны множества групп, законных и незаконных, заставить Xbox запускать код из произвольных источников без изменений оборудования.

Xbox является жертвой своей собственной конструкции: выбор использования стандартного оборудования ПК значительно увеличивает ценность «открытого» Xbox как для хакеров, так и для пиратов. Xbox является довольно удовлетворительной целью для хакеров выходного дня и любителей по той же причине, по которой Microsoft приняла архитектуру ПК для Xbox: существующие программы ПК легко портируются на Xbox. Кроме того, существует обширная и глубокая база знаний об оборудовании ПК, поэтому кривая обучения для взлома Xbox не такая крутая, как для других консолей. С другой стороны, Playstation2 и Gamecube имеют крутую кривую обучения, и они также имеют архитектурные ограничения, которые затрудняют портирование большинства приложений ПК. Xbox также является популярной целью для пиратов из-за простоты портирования устаревших игровых эмуляторов, а также из-за его высокого статуса и простоты получения совместимого отладочного и тестового оборудования.

Кроме того, сходство архитектуры Xbox с архитектурой ПК делает Xbox хорошим образовательным средством. Знания, полученные из этой книги, применимы не только к встроенному оборудованию или игровым консолям; вы сможете применить большую часть знаний из этой книги непосредственно к ПК. Кроме того, обширные ресурсы документации, применимые к Xbox, унаследованные от мира ПК, удобно индексируются поисковыми системами в Интернете.

**Взлом Xbox: Введение в обратную разработку**

Доступность документации поможет мотивированным читателям развивать знания, содержащиеся в этой книге.

Xbox также является более привлекательным образовательным примером, чем обычный ПК. Слишком много различий между деталями оборудования

Реализации для ПК позволяют создавать полезные пошаговые руководства по взлому для ПК, в то время как пошаговые руководства для Xbox гарантированно точны для миллионов устройств, которые можно легко приобрести практически в любом торговом центре или магазине электроники.

## О хакерах и взломе

Это книга о взломе в традиционном смысле: о процессе и методах исследования. Некоторые могут удивиться, что в этой книге нет глав, посвященных рипу игр и исправлению определенных проверок безопасности — в конце концов, разве не в этом суть взлома? На самом деле, термин «хакер» довольно сильно изменился за эти годы, поскольку осведомленность общественности о технологиях возросла, а сенсационные СМИ продолжают описывать общественное мнение о хакерах.

В начале хакером был тот, кто работал страстно ради любопытства и исследования. Были хакеры оборудования, которые брали на себя задачу снять крышки с компьютеров, чтобы оптимизировать их конструкцию (ранние компьютеры были построены из дискретных компонентов, поэтому их можно было модифицировать осмысленным образом с помощью простых инструментов), и были хакеры программного обеспечения, которые трудились, чтобы сделать максимально компактный и элегантный код, поскольку вычислительные ресурсы были редкими и медленными. Были хакеры, которые исследовали все входы и выходы телефонной системы, и те, кто исследовал крыши и туннели зданий университетских кампусов. Довольно часто ранние хакеры занимались всеми этими видами деятельности. Хакеры свободно делились своими открытиями или результатами (взломами) друг с другом, поскольку их вознаграждение не было финансовым, а достигалось за счет удовлетворения их интеллектуального любопытства и энтузиазма их коллег. В результате хакеры, как правило, объединялись в меритократические группы, где членство и продвижение основывались исключительно на способности человека к взлому.

По мере развития технологий, когда компьютеры становились быстрее и более интегрированными, хакеры обнаружили, что усилия, затраченные на взлом оборудования, не стоят выигрыша. Интересные части компьютеров быстро оказывались глубоко закрытыми в герметичные керамические корпуса, выгравленными в кремниевых структурах, которые было трудно увидеть даже с помощью хорошего микроскопа. Сложный взлом оборудования, который мог бы удвоить производительность компьютера, был признан нецелесообразным в течение нескольких месяцев законом Мура.

С другой стороны, программный хакер начал больше фокусироваться на приложениях и меньше на алгоритмах или оптимизации. Компактность или элегантность программы больше не были напрямую важны, поскольку память и мощность процессора стали дешевыми и многочисленными. Кроме того, технология компиляторов также улучшилась до такой степени, что скомпилированный код выполнялся почти так же быстро, как ручная сборка. К концу 80-х годов термин «хакер» стал подразумевать человека, который мог писать тома кода на языке С во сне и создавать блестящие приложения за одну ночь. Старые хакеры оборудования либо превращались в хакеров программного обеспечения, либо уходили в университетские лаборатории и корпорации, которые могли позволить себе поддерживать свои дорогие хобби.<sup>2</sup>

Термин «хакер» в то время все больше ассоциировался с людьми, которые взламывали пароли и программы, чтобы получить доступ к машинам и программному обеспечению, которые в противном случае были бы недоступны. Голливуд был частично ответственен за этот стереотип, с массой фильмов, которые изображали подростков, ставящих мир на грань ядерного уничтожения несколькими нажатиями клавиш, или скрытых гениев, создающих искусственных интеллектуальных кибермонстров в своих подвалах.<sup>3</sup> К сожалению, гипербола этих сюжетов фильмов была утеряна широкой публикой, и это мрачное впечатление о хакерах в конечном итоге стало доминирующей частью стереотипа хакера. Неточность этого стереотипа способствовала созданию термина для хакеров, который в первую очередь фокусируется на взломе систем и программ — «кракеры».

Технология формирует современного хакера так же, как хакеры сформировали технологию. Новые поколения хакеров должны усердно работать, чтобы проникнуть в «дружественные» пользовательские интерфейсы и в медийный и маркетинговый блеск, который окружает компьютерные технологии сегодня. Все используют компьютеры и ожидают от них безупречной и интуитивной работы, но мало кто действительно понимает, что происходит под капотом.

Технология вычислений стала настолько сложной, что новички все больше напоминают притчу о семи слепых и слоне.  
Семь слепых мудрецов услышали, что в их деревню привели слона, и решили узнать, что это за существо. Поскольку они были слепыми, каждый мог исследовать слона только на ощупь. Первый мудрец прикоснулся к боку слона и сказал: «Слон подобен стене». Второй потрогал его бивень и воскликнул: «Нет, слон как острое копьё». Третий мудрец взялся за хобот и уверенно заявил: «Слон похож на змею». Четвёртый, нащупав

---

<sup>2</sup>Хорошней новостью является то, что в последнее время технология взлома оборудования догоняет закон Мура, что приводит к ренессансу взлома оборудования. Появились доступные услуги по изготовлению печатных плат, а рождение Интернета упростило процесс приобретения компонентов. Кроме того, такие услуги, как Mosis chip foundry service и FIB (focused ion beam) services начали выводить взлом интегральных схем в сферу финансовых возможностей для отдельных энтузиастов оборудования.

<sup>3</sup>Родни Брукс, директор лаборатории искусственного интеллекта Массачусетского технологического института, однажды сказал, что голливудская идея о сумасшедшем изобретателе, создающем в подвале существо с искусственным интеллектом, примерно эквивалента попытке кого-то построить реактивный самолёт Boeing 747 у себя во дворе.

### Взлом Xbox: Введение в обратную разработку

ногу слона, сказал: «Слон как могучий столб». Пятый мудрец дотронулся до уха и сказал: «Слон как большой веер». Шестой исследовал хвост и заключил: «Слон как верёвка». Седьмой, который стоял рядом и слушал всех, сказал: «Все вы правы и не правы одновременно. Слон — это и то, и другое, и третье, но вы каждый видите лишь часть».

Некоторые новички начинают свой путь хакера с исследования Интернета.

Другие начнут с изучения операционной системы своего компьютера.

Другие же начнут с того, что заглянут под крышку своего компьютера. Каждый человек может потратить год на изучение своей грани, но в итоге у каждого будет совершенно разное представление о компьютерных технологиях.

Культурный разрыв между молодыми хакерами и старой гвардией стал для меня очевидным, когда самопровозглашенный хакер-новичок в МИТ усмехнулся: «Где все эти компьютеры с Windows[98]? . . . все, что у вас есть, это эти жалкие компьютеры Sun, на которых даже нет AOL! Я думал, что в МИТ будет хороший доступ в Интернет». Он, казалось, не понимал того факта, что «жалкие компьютеры Sun» были довольно мощными рабочими станциями, работающими под управлением одной из самых надежных операционных систем в мире, и что Интернет существует и за пределами AOL — более того, что кампус МИТ был одним из мест рождения Интернета, с правами на большее количество IP-адресов, чем у большинства интернет-провайдеров, и прямым подключением к магистрали Интернета.

Проникновение компьютерных технологий во все уголки повседневной жизни усилило стереотипы о хакерах. В частности, изображение хакеров в СМИ как современных Робин Гудов каким-то образом неизбежно связывало хакерство с аспектами, связанными с безопасностью или доступом к компьютерным ресурсам. Теперь стереотипный хакер несет ответственность за варез, Code Red и пинг-флуд, в то время как «разработчики» несут ответственность за Linux и BSD. Хакеры — это 31337 d00fs, которые 0\\n jh00r b0x0r, а хакер оборудования разгоняет и модифицирует корпус своего компьютера неоновыми огнями. Хакерство стало модным, и многие стремятся соответствовать стереотипу, созданному СМИ. Сегодня очень трудно убедить людей, что я взломал Xbox только потому, что он был там для взлома: это было сложно, и это было ново. Точно так же людям трудно понять, почему я с тех пор не работал над Xbox. После взлома системы безопасности Xbox остается только стандартный ПК, на котором, как мне кажется, не так уж интересно работать, и он определенно не стоит риска судебного иска от Microsoft.

## Политика хакерства

Введение в действие Закона об авторском праве в цифровую эпоху (DMCA) в 1998 году вывело криптографию из сферы влияния хакеров — теперь закон гласит, что только исследователи, «занимающиеся законным курсом обучения, имеющие работу или прошедшие соответствующую подготовку или опыт»<sup>4</sup> разрешено исследовать криптографические методы защиты прав доступа к работам. В результате взлом Xbox

---

<sup>4</sup>17 USC § 1201(g)(3), Факторы, определяющие освобождение. Конечно, значение фразы «соответствующим образом обученный или опытный» не определено. Я думаю, что лучшая подготовка для прикладных криптографических исследований должна включать практический опыт взлома реальных крипtosистем.

стал политически заряженной темой. Это битва между хакерами и законодателями за сохранение криптографии в рамках законных прав хакеров.

Похвальная реакция Microsoft на хакеров Xbox — то есть отсутствие преследования или попыток закрыть проекты по взлому Xbox — надеюсь, послужит примером для других, кто думает об использовании DMCA для прекращения хакерской деятельности. Несмотря на все взломы Xbox, Microsoft по-прежнему наслаждается устойчивыми продажами игр. Весь интерес и шумиха, вызванные взломом Xbox, могли увеличить продажи Microsoft больше, чем пиратство навредило им. (Конечно, я симпатизирую хакерам, поэтому моя интерпретация ситуации предвзята. Более субъективный и обоснованный юридический анализ обратного проектирования можно найти в главе 12 «Caveat Hacker» Ли Тьена из Electronic Frontier Foundation.)

Самым тревожным аспектом DMCA для хакеров является то, что он воплощает заблуждение, что единственные источники инноваций, приносящих пользу обществу, находятся в залах исследовательских институтов и корпораций. Внезапно стало преступлением изучать, не выходя из дома, занимаясь своим хобби, криптографические методы, используемые для защиты прав доступа. Ограничение исследований таких технологий только устоявшимися институтами исключает возможность разработки технологий независимыми лицами. Если бы не свобода исследовать и разрабатывать технологии в собственном гараже, где бы сегодня были такие люди, как Билл Хьюлетт и Дэйв Паккард, или Стив Джобс и Стив Возняк? Были бы у нас Linux и netBSD, если бы право хакеров свободно выражать себя в коде регулировалось?

*Каждая схема защиты авторских прав, которая была взломана хакером, учит кого-то важному уроку о том, как создать более надежную защиту.* Принятие законов, регулирующих исследование технологических мер, защищающих авторские права, и распространение таких результатов — это признание того, что технология защиты авторских прав не совершенна и никогда не может быть усовершенствована, и что единственный возможный исход разрешения простым людям понять технологию контроля за соблюдением авторских прав — это крах этой технологии. Я предлагаю другое мнение: некоторые из лучших отзывов, которые я получил на свою работу по взлому Xbox, пришли не из академического сообщества. Они пришли от отдельных хакеров со всего мира — особенно из зарубежных стран, — которые могли свободно изучать и понимать технологии контроля доступа. Более строгие законы в США и склонность корпораций к судебным искам уже негативно повлияли на позиции США в области электронной безопасности, и это только начало.

Социальное влияние DMCA ощущают хакерские сообщества по всему миру. В ходе моей работы над Xbox мне посчастливилось познакомиться с блестящими хакерами по всему миру. Хакеры в Америке были одними из самых робких в этой группе, и хотя они были талантливыми инженерами, они не хотели применять свои навыки для решения таких проблем из-за страха преследования. В результате некоторые из самых интересных результатов взлома Xbox получены европейскими и азиатскими хакерскими сообществами. Примечательно, что эти результаты не очень известны в Америке, поскольку у этих хакеров мало мотивации делиться своими открытиями с американцами. На самом деле, многие иностранные хакеры прилагают сознательные усилия, чтобы их открытия не покидали пределы их сообществ, по ряду причин, в том

---

### Взлом Xbox: Введение в обратную разработку

числе из-за страха возмездия со стороны американских корпораций. Эта «утечка мозгов» мало способствует укреплению компетентности Америки в технологии, столь важной, как справедливый и эффективный контроль за цифровыми авторскими правами. А в сегодняшней глобальной экономике американские корпорации не могут выжить, делая вид, что ведут бизнес в вакууме.

Можно указать на успешную публикацию моей статьи о системе безопасности Xbox как на пример того, как DMCA защищает как права на свободу слова, так и экономические интересы в технологии контроля авторских прав. Моя ситуация была нетипичной для большинства хакеров в США. Поскольку в то время я был аспирантом, у меня не было семьи, о которой нужно было бы беспокоиться, или значительных активов, которые можно было бы потерять, если бы я оказался вовлеченным в судебный процесс по поводу моей работы. Я также получил щедрую юридическую помошь от Electronic Frontier Foundation (EFF), которая помогла мне пройти через юридическое минное поле. EFF помогла представить мою статью в максимально возможном юридическом свете, проинформировав меня о моих правах и обязанностях в соответствии с DMCA.

Например, от меня требуется «приложить добросовестные усилия для получения разрешения [от Microsoft] перед обходом».<sup>5</sup> (Обратите внимание, что разрешение не требуется, но добросовестные усилия требуются.) EFF помог мне составить такое письмо для исследования. Мне также пришлось бороться с МИТ, чтобы разрешить опубликовать мое исследование в качестве аффилированного лица. Все прямые усилия по обратному проектированию безопасности Xbox финансировались из моего собственного кармана, проводились в моей квартире и делались после работы в мое личное время. МИТ изначально воспользовался этим фактом, чтобы отделить себя от моей работы, заставив меня обратиться за советом в EFF. МИТ в конце концов сдался и разрешил мне опубликовать мою работу как студенту МИТ после долгих уговоров сочувствующих профессоров и после того, как я получил конструктивное, не угрожающее письмо от Microsoft о моем исследовании.

Свобода слова не должна требовать адвоката, а свобода мысли не должна включать в себя доверенности на исследование. Я боролся за публикацию своей статьи, потому что мне нечего было терять, и потому что я верил в возможность сделать заявление о своих правах как хакера. К сожалению, есть молчаливое большинство хакеров, у которых есть семьи, которые нужно кормить, и работа, которую можно потерять, и не всем может повезти, чтобы EFF им помогла.

Книга, которую вы читаете, — еще один пример того, как DMCA оказывает сковывающее воздействие на свободу слова. Первоначально заказанная техническим издателем John Wiley & Sons, Ltd., эта книга была отменена в последний час из-за опасений судебных исков и ответной реакции со стороны Microsoft. Такая цензура расстраивает и обескураживает, и, возможно, некоторые авторы остановились бы на этом и позволили бы своему голосу замолчать из-за страха. Я иду на юридический и финансовый риск самостоятельной публикации этой книги, чтобы заявить о своем праве на свободное и беспрепятственное слово как хакера. Однако даже этот путь не

---

<sup>5</sup>17 USC § 1201(g)(2), Допустимые действия по исследованию шифрования

свободен от препятствий. Процесс предварительного заказа книги был приостановлен на второй день, поскольку первоначальный поставщик услуг электронной коммерции Americart «отказался предложить [мне] услугу корзины для продажи хакерских материалов... 15 долларов в месяц не окупают того, чтобы мы рисковали быть названными в иске DMCA».

Я должен подчеркнуть, что эта книга не нарушает авторские права Microsoft, и знания, представленные в ней знания не позволяют напрямую обойти защиту авторских прав. Чтобы совершить нарушение, нужно отточить свои навыки и применить значительное количество дополнительных навыков и ноу-хау, направленных специально на обход контроля авторских прав. Утверждать, что эта книга является инструментом обхода, было бы равносильно утверждению, что все книги о печатных платах, встроенном программном обеспечении или криптографии также являются инструментами обхода.

Область действия DMCA в отношении «добросовестного использования» оборудования — еще одна важная политическая тема с огромными экономическими последствиями. Является ли незаконным изменение или обход криптографически защищенной последовательности загрузки с целью запуска альтернативного, законно приобретенного или созданного программного обеспечения? Этот вопрос может быть частично решен судьбой хакеров Xbox. Страгическое толкование исключения обратного проектирования DMCA<sup>7</sup> раскрывает веские аргументы в пользу того, чтобы сделать такие акты обхода незаконными.

В частности, обратная разработка разрешена только для взаимодействия, где взаимодействие означает «способность компьютерных программ обмениваться информацией, а таких программ — взаимно использовать информацию, которой они обмениваются». Но это определение содержит две потенциальные мины: во-первых, обход мер безопасности на основе оборудования, возможно, отличается от обхода мер безопасности программы (программного обеспечения). Это может быть не очень сильным аргументом с технической точки зрения, но, насколько мне известно, этот пункт еще не прошел юридическую проверку. Во-вторых, цель на самом деле не в обмене информацией с мерами безопасности оборудования, а в их обходе.

Последним аргументом против разрешения обратного проектирования аппаратных механизмов безопасности является случайный обход авторских прав.

---

<sup>7</sup>, Обратное проектирование

Информация, полученная в процессе обратного проектирования, может быть в равной степени применена для создания устройств обхода авторских прав. Другими словами, базовые исследования, которые обеспечивают совместимость, по крайней мере в случае Xbox, могут также косвенно применяться к тем, кто хочет создать устройства обхода. Как оказалось, некоторые весьма специфические недостатки дизайна в Xbox позволяют обойти защиту загрузки, не обязательно позволяя обойти авторские права, хотя эти недостатки могут быть исправлены в ближайшем будущем, что поставит нас лицом к лицу с нашим первоначальным вопросом.

Существуют значительные экономические последствия, если окажется, что «добросовестное использование» не охватывает обратную разработку безопасности

### Взлом Xbox: Введение в обратную разработку

Xbox с целью запуска альтернативных приложений. Наиболее существенным последствием является то, что Microsoft может продавать юридически ограниченное оборудование конечным пользователям, запирая пользователей в своей программной базе. Это может быть использовано для создания нерушимой монополии на компьютерное оборудование и программное обеспечение. Например, Microsoft может предложить субсидии поставщикам, которые решат защитить свое оборудование для работы операционной системы Microsoft. Этот финансовый стимул будет передан клиентам, которые будут мотивированы покупать оборудование со скидкой. Как только значительная часть установленной базы оборудования будет заблокирована в операционных системах Microsoft, Microsoft сможет устанавливать цены на свою продукцию на рынке без конкуренции, поскольку было бы незаконно запускать любую другую операционную систему на заблокированном оборудовании.

В реальности, этот сценарий может быть трудно реализовать для Microsoft, даже если DMCA действительно ограничит добросовестное использование оборудования, поскольку государственные и общественные органы пристально следят за деятельностью Microsoft на предмет монополистического поведения. Однако на других развивающихся рынках, таких как умные ячейки



Автор на своем рабочем месте.

телефоны, КПК и телевизионные приставки, для поставщика может быть вполне реалистичным попытаться получить преимущество над конкурентами с помощью такой тактики низкого мяча. По крайней мере, такая тактика может быть использована для остановки конкуренции на время судебного разбирательства, которое может быть достаточно долгим, чтобы нанести непоправимый ущерб рыночной позиции конкурента. Именно из-за этих опасений многие хакеры Xbox сознательно действуют,

чтобы выразить свои политические убеждения посредством своих инженерных усилий.

## Люди, стоящие за взломом

В этой книге я привожу профили различных хакеров, которые согласились дать интервью. Этот набор хакеров ни в коем случае не единственный набор хакеров; на самом деле, это группа, которая отбирает себя сама, поскольку многие хакеры работают тайно из-за страха преследования или потому что они работают в компаниях, имеющих тесные связи с Microsoft. Цель этих интервью — немного рассказать о людях, стоящих за взломом, и рассказать об их мотивах и методах, чтобы способствовать пониманию и вдохновить новых хакеров присоединиться к нашим рядам.

Позвольте мне начать с представления себя. Я Эндрю «Банни» Хуан; большинство людей называют меня «Банни» {Кролик}. На момент написания этой статьи мне было 27 лет, я сын Эндрю С. и Маргарет Хуан. Я родился и вырос в Каламазу, штат Мичиган, но в настоящее время живу в Сан-Диего, штат Калифорния, со своей замечательной невестой Никки Джастис. Недавно я окончил Массачусетский технологический институт, получив докторскую степень по электротехнике. Одна из причин, по которой меня выбрали для написания этой книги о взломе Xbox, заключается в том, что я обнаружил и опубликовал первую известную уязвимость в системе безопасности Microsoft Xbox.

В целом я занимаюсь хакерством, потому что мне очень приятно знать, что чья-то жизнь стала лучше благодаря чему-то, что я построил. Я чувствую, что это моя обязанность — применить свои таланты и вернуть обществу то, что оно дало мне. Мне также нравится вызов исследования. Я хочу понять электроннику как можно глубже. Черные ящики меня расстраивают; ничто не возбуждает мое любопытство больше, чем ящик, который мне не разрешено открывать или понимать. В результате у меня есть доверительный интерес к криптографии и методам безопасности.

Я взламываю оборудование, потому что мне нравится эстетика электроники; есть что-то удовлетворяющее в том, чтобы в конце дня получить осязаемый артефакт, а не эфемерные кусочки программного кода. Это может показаться немного глупым, но одно из моих увлечений — разбирать электронные устройства и «читать» печатные платы. Есть что-то волнующее в запахе новенького электронного оборудования, только что вынутого из антистатических пакетов; я думаю, это запах нового разворачивающегося приключения. Он манит, как стопка чистой бумаги: интересно, что я буду делать с этими пустыми страницами. Стопка чистой белой бумаги стоит там и бросает мне вызов, чтобы заполнить ее полезной информацией.

Моя любознательная натура берет свое начало в детстве. Когда мне было около семи лет, мой отец купил клон Apple II. Он купил только материнскую плату, так что у него не было корпуса. Я до сих пор помню, как он впервые достал его из коробки — зеленая печатная плата, блестящие чипы и все разноцветные резисторы и конденсаторы. Мне хотелось поиграть с ним! Как бы мне ни было любопытно по поводу Apple II, мне не разрешалось прикасаться к материнской плате. Конечно, это означало, что когда мои

---

**Взлом Xbox: Введение в обратную разработку**

родители не смотрели, я вынимал чипы из гнезд на материнской плате и делал глупости, например, вставляя их наоборот, чтобы посмотреть, что произойдет.

После того, как я несколько раз чуть не уничтожил компьютер, мои родители купили мне набор для экспериментов с электроникой 200-в-1 от Radio Shack и мою первую книгу по электронике, *Getting Started in Electronics*, by Forrest Mims, III. Это было для меня отличным введением в электронику, так как они удовлетворяли мое желание играть со схемами и компонентами. Мой дядя также дал мне свой старый экземпляр *Art of Electronics* от Horowitz and Hill, вместе с парой книг о микропроцессорах. Я подписался на журнал Byte, который в то время включал регулярные колонки о проектах по созданию оборудования, полные схем и изображений.

В конце концов, я развел в себе достаточно понимания электроники, чтобы начать понимать схемы и листинги ПЗУ, включенные в руководства пользователя Apple II. (Я все еще считаю, что компьютеры должны поставляться с полными схемами и исходным кодом.) К восьмому классу я развел в себе достаточно понимания, чтобы иметь возможность построить собственную плату расширения для Apple II. На плате был синтезатор речи General Instruments SPO-256, который я купил в Radio Shack. Я также добавил аналого-цифровой преобразователь к своему Apple II и написал приложение, которое превратило мой Apple II в говорящий вольтметр. Я продолжал собирать аппаратное обеспечение, и до того, как меня приняли в МГТ, я построил свой собственный работающий встроенный компьютер, используя микропроцессор 80188.

Во время учебы в МГТ я избегал рутинные школьные занятия, создавая забавные маленькие проекты, такие как дистанционно управляемый выключатель света и реагирующие на музыку праздничные огни для моего братства ZBT. Именно в эти годы я впервые познакомился с доступными услугами прототипирования и инструментами САПР печатных плат, такими как те, что обсуждаются в Приложении С, «Введение в компоновку печатных плат».

Рост услуг по изготовлению печатных плат, доступных по бюджету студента колледжа, является знаковым событием для хакеров оборудования. Наконец, инструмент для накрутки проводов можно убрать, а компоненты поверхностного монтажа и сложные схемы доступны обычным любителям.

На протяжении многих лет я считал обязательным описывать свои проекты на своей веб-странице (<http://www.xenatera.com/bunnie>), чтобы каждый мог извлечь пользу из моего опыта. Многие из моих проектов доступны со схемами, файлами Gerber и исходным кодом, хотя некоторые из моих последних проектов были консалтинговыми работами, поэтому я, к сожалению, не могу поделиться этими результатами с миром.

Пока я удерживаю ваше внимание, я хотел бы прояснить одну вещь. Я не получил свою докторскую диссертацию в МГТ за взлом Xbox. Взлом Xbox был на самом деле отклонением от моей диссертации, которое было косвенно связано, но не являлось центральной темой моей диссертации.

Моя диссертация по суперкомпьютерам<sup>6</sup> сосредоточена на архитектуре по эффективному коду и миграции данных. Мой интерес к игровым консолям проистекает из моего естественного любопытства ко всему оборудованию в сочетании с поддержкой моего научного руководителя, доктора Тома Найта. Игровые консоли представляют собой вершину производительности на стоимость, а стоимость является существенной проблемой для суперкомпьютеров сегодня. Поэтому меня вдохновили взглянуть на все игровые консоли, чтобы узнать, что я могу узнать о создании экономически эффективного оборудования. Тот факт, что Xbox также имела интересную систему безопасности, был бонусом; поскольку правительственные учреждения проявляют большой интерес к суперкомпьютерным технологиям, безопасность суперкомпьютеров всегда является темой для рассмотрения. (На самом деле, очень интересная статья о создании надежных компьютеров<sup>7</sup> (Написано коллегами из моей исследовательской группы; рекомендую прочитать, если вас интересуют альтернативы криптографически защищенным доверенным вычислительным платформам, таким как Palladium и TCRP.)

Мой лучший совет начинающим хакерам оборудования — быть настойчивыми и основательными. Важно отметить, что настойчивость и основательность приходят естественным образом, если вы любите то, что делаете. Кроме того, часть работы хакера оборудования — быть спекулянтом. Покупка нового оборудования обходится непомерно дорого, поэтому я по привычке накапливаю сломанное и изношенное оборудование и инструменты, даже если я точно не знаю, что с ними делать или смогу ли их починить. Оказывается, попытка починить тестовое оборудование сама по себе является обучающим опытом и может быть весьма полезной, даже если в итоге придется выбросить эту чертову штуку на запчасти.

Цитируя бывшего евангелиста Apple и нынешнего руководителя Garage Technology Ventures Гая Кавасаки, «Ешь как птичка — наваливай как слон». Кавасаки отмечает, что колибри съедает эквивалент 50% своего веса каждый день. Следовательно, если вы едите как птица, у вас должен быть бесконечный аппетит к информации. Подпишитесь на бесплатные журналы по электронике, просматривайте веб-сайты (но будьте избирательны в выборе сайтов, которые вы просматриваете, — вы то, что вы едите), посещайте бесплатные выставки и подписывайтесь на все каталоги и периодические издания, которые попадутся вам в руки; разберите каждую электронную деталь, которая есть у вас и ваших друзей, и постараитесь узнать все, что можете, из их конструкции.

В хакерстве оборудования половину самых сложных проблем можно решить или облегчить, просто используя правильный выбор компонентов или методов. «Какать как слон» означает делиться информацией и открытиями с другими хакерами.

---

<sup>6</sup>Текст моей докторской диссертации можно найти по адресу <http://www.xenatera.com/bunnie/phdthesis.pdf> {или в переведенном виде на гитхабе переводчика - [https://github.com/KONOVALOVda/HackingTheXbox\\_RUS](https://github.com/KONOVALOVda/HackingTheXbox_RUS) }

<sup>7</sup>«Минимальная доверенная вычислительная база для динамического обеспечения безопасного потока информации» Джереми Брауна и Тома Найта можно найти по адресу <http://www.ai.mit.edu/projects/aries/Documents/Memos/ARIES-15.pdf> {или - [https://github.com/KONOVALOVda/HackingTheXbox\\_RUS](https://github.com/KONOVALOVda/HackingTheXbox_RUS) [github.com] }

---

### Взлом Xbox: Введение в обратную разработку

Неважно, сколько информации вы усвоите, вы никогда не сможете знать ее всю. Свободный обмен своими открытиями приглашает к совету и доброй воле других хакеров и ведет к синергии умов. Особенно в хакерстве оборудования, где все результаты имеют основу в осозаемые артефакты, скрытие ваших методов и результатов означает лишь то, что другие люди в конечном итоге переосмысят вашу работу без вашей помощи. С другой стороны, проявляйте некоторую рассудительность в том, что вы говорите или чем делитесь; у людей ограниченный диапазон возможностей, и они будут слушать более внимательно, если вы поделитесь результатами, которые являются новыми или интересными в каком-то смысле.

Итак, берем отвертку и начинаем взламывать!

{Опытным пользователям рекомендую переходить сразу ко второй главе –  
“мышление внутри коробки”}

# CHAPTER 1 Аннулирование гарантии

## Инструменты для работы

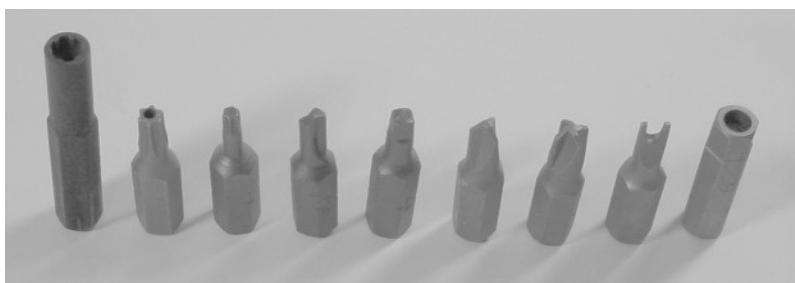
Взлом оборудования может показаться сложным на первый взгляд из-за сложных инструментов, которые требуются для некоторых проектов. К счастью, большинство базовых проектов можно выполнить, вложив лишь небольшие средства в инструменты, сопоставимые со стоимостью одной-двух видеоигр. Приложение A, «Где взять оборудование», содержит рекомендуемый список стартовых инструментов и инструкции по их заказу.

В этой главе будут рассмотрены основные инструменты, которые вам понадобятся для серьезного взлома оборудования, включая инструменты для вскрытия, присоединения и удаления электронных компонентов, диагностики и зондирования схем и проектирования печатных плат. Из этих инструментов качественные версии первых двух можно приобрести по довольно разумным ценам. Диагностические и тестовые инструменты, такие как осциллографы и логические анализаторы, стоят своего веса в золоте, но вы обнаружите, что они очень тяжелые и станут внушительной инвестицией. Что касается инструментов для проектирования печатных плат, некоторые из лучших инструментов могут продаваться по удивительно доступным ценам.

Эта глава завершится пошаговым иллюстрированным руководством по открытию Xbox. Более опытные хакеры оборудования могут пропустить следующие несколько глав.

## Инструменты для открытия вещей

Первый шаг взлома чего-либо — снятие крышки. Большинство электронных приборов можно открыть с помощью набора крестовых и плоских отверток, но для самых интересных коробок потребуется набор специальных бит безопасности.



---

### Взлом Xbox: Введение в обратную разработку

**Рисунок 1-1:** Выбор бит безопасности. Слева направо: Nintendo 4,5 мм, security torx, standard torx, clutch, Robertson или квадрат, tri-wing, torq, гаечный ключ и security allen или шестигранник.

На рисунке 1-1 показан ряд некоторых распространенных битов безопасности. Удивительно, но наборы битов безопасности доступны и их легко получить. MCM Electronics ([www.mcmelectronics.com](http://www.mcmelectronics.com)) продает набор из 105 битов безопасности (номер заказа MCM 22-3495) менее чем за двадцать долларов, а набор из 32 штук (номер заказа MCM 22-1875) менее чем за десять долларов. Они стоят своих вложений. Биты безопасности Nintendo продаются отдельно. Вы можете получить большую биту безопасности Nintendo, используемую в Nintendo Gamecube, за несколько долларов (номер заказа MCM 22-1150, «4,5-миллиметровая бита безопасности»). Уменьшенная версия биты (номер заказа MCM 22-1145, «3,8-миллиметровая бита безопасности») также используется в старых системах Nintendo и их игровых картриджах.

Xbox использует стандартные биты Торх (шестиконечная звезда) размером T10, T15 и T20. Эти биты довольно распространены и могут быть куплены в хозяйственных магазинах, таких как Home Depot. Вы также можете найти магнитный удлинитель для бит, который будет удобен для доступа в несколько узких мест вокруг жесткого диска и DVD-привода в Xbox.

Не прилагайте чрезмерных усилий при снятии крышки с оборудования. Если вы думаете, что вы удалили все винты, но крышка все еще застряла, скорее всего, вы либо пропустили винт, либо вам нужно нажать на некоторые фрикционные заплечики. Также часто винты спрятаны под резиновыми ножками на нижней части оборудования или под наклейкой. Чтобы найти винты, спрятанные под наклейками, тщательно потрите поверхность наклейки. Вы почувствуете мягкое место в любом месте, где под ним находится винт. (Разрыв такой наклейки для доступа к винту мгновенно аннулирует гарантию на оборудование, но не бойтесь: большая часть оборудования рассчитана на обслуживание, поэтому простое снятие крышки редко приводит к каким-либо повреждениям.)

Иногда вы столкнетесь с упрямой сборкой, которая отказывается разъединяться. Если крышка или панель прогибается по краям или кажется, что у нее есть некоторая свобода движения, возможно, есть какой-то фрикционный замок,держивающий крышку. Фрикционные замки, как правило, представляют собой конструкции с выступом и прорезью, имеющие такую форму, что вставить выступ гораздо проще, чем вынуть его. В этом случае найдите выступ, наблюдая, где корпус, кажется, застрял, и нажмите на выступ небольшой плоской отверткой, одновременно

осторожно потяните корпус вверх. Если таких защелок несколько, вставьте какой-нибудь клин, например, другую отвертку или скрепку, чтобы защелка не вошла в зацепление, когда вы откроете другие защелки.

Если крышка или панель отказывается двигаться даже немного, когда вы прикладываете сильное давление, она также может быть прикреплена kleem или даже заварена. Например, блоки питания «wall-wart» (квадратные черные коробки, которые вы подключаете непосредственно к розеткам) часто запечатаны таким образом. Разобрав такое оборудование, вы можете никогда не собрать его обратно в его первоначальную форму.

## Инструменты для крепления и снятия компонентов

Электронные компоненты крепятся к платам пайкой. При пайке легкоплавкий сплав, известный как припой, нагревается и обтекает соединяемые металлы. Припой и металлы образуют локальный сплав. После остывания соединения компоненты электрически и механически соединяются.

Основными инструментами для пайки являются паяльник, припой, флюс и оплетка для распайки. (Пара тонких пинцетов также весьма удобна для работ, связанных с мелкотяговыми компонентами или мелкими деталями.) Паяльник — это ручной инструмент, состоящий из нагревательного элемента и наконечника; наконечник используется для плавления припоеv посредством проводимости или прямого контакта, в отличие от других инструментов, которые используют горячие газы или интенсивное инфракрасное излучение. Тип наконечника паяльника, необходимый для оптимальной теплопередачи, зависит от ситуации. Например, плоский наконечник «зубило» или «конический зубило» будет работать лучше, чем простой заостренный наконечник при пайке большинства мелких компонентов поверхностного монтажа.

Существует также множество марок паяльников. Самые дешевые стоят около десяти долларов и поставляются с большими, громоздкими наконечниками и не имеют регулировки температуры; они просто нагреваются настолько, насколько могут. Лучшие паяльники стоят дороже и имеют датчик, который активно регулирует температуру наконечника. Регулировка температуры делает работу инструмента более последовательной и продлевает срок службы наконечника. Лучшие паяльники также поставляются с более широким выбором наконечников, которые могут включать очень тонкие, подходящие для работы с крошечными компонентами, которые встречаются в

---

### Взлом Xbox: Введение в обратную разработку

большинстве современных электронных устройств. Для нечастого использования достаточно качественного паяльника с прямым подключением и хорошим наконечником. Однако, если вы планируете собирать платы и действительно заняться хакингом оборудования, сотня долларов за качественный паяльник с регулируемой температурой, такой как Weller WTCPT или Weller WES50, вполне окупится.

Припои бывают самых разных видов. Для большинства целей достаточно эвтектического припоя из сплава Pb-Sn с флюсовым сердечником, не требующим очистки или очищаемым водой. Эвтектические сплавы предпочтительны, поскольку они переходят из жидкой фазы в однородное твердое вещество. Kester — крупный производитель припоев; их стандартные проволочные припои с сердечником, Formula 245 и 331, оба довольно хороши. Formula 245 использует флюс, не требующий очистки, но при желании можно использовать ватный тампон с изопропиловым спиртом, чтобы удалить остатки. Formula 331 имеет флюсовый сердечник, который работает с большим количеством материалов, чем 245. Однако с 331 вам нужно промыть плату водой вскоре после пайки, иначе остатки флюса станут липкими и, возможно, поменяют работе схемы. Многие дистрибуторы продают припой Kester; например, Kester 246337-8802 (припой Formula 245 калибра 25 в катушке весом 1 фунт) — это DigiKey ([www.digikey.com](http://www.digikey.com)) номер детали KE1410-ND. Тип припоя, продаваемый в большинстве Radio Shack, также довольно хорош для пайки, хотя их припой имеет тенденцию оставлять липкий черный налет и требует очистки органическими растворителями.

Припой также может быть в виде пасты с крошечными шариками припоя, взвешенными в матрице флюса. Паяльная паста может быть очень полезна при присоединении мелкошаговых компонентов поверхностного монтажа. (См. Приложение B, «Техники пайки», для получения дополнительной информации.)

Если паяное соединение не поддается формированию, флюс — это панацея. Всегда держите немного флюса под рукой. Если соединение формируется неправильно, небольшая капля флюса, нанесенная непосредственно на соединение, обычно решает проблему. Флюс также выпускается в виде большого количества паст и жидкостей, каждая из которых требует своего метода очистки. Удобным решением для нанесения флюса является флюс-карандаш, например, Kester 83-1000-0951, флюс-карандаш Formula 951 noclean. Вы можете купить этот флюс-карандаш у Digi-Key, номер детали KE1804-ND, всего за несколько долларов. Radio Shack также продает флюс-пасту в тюбике, но их паста грязная и требует очистки.

Наконец, распаивающие оплетки полезны для очистки любых паяльных беспорядков или ошибок, которые вы можете сделать.

Распивающая оплетка представляет собой тонкую плетеную медную проволоку, обычно пропитанную сухим флюсом. Чтобы использовать ее, поместите ее между паяльником и соединением, которое вы хотите очистить; как только оплетка нагреется, излишки припоя на соединении впитаются в капилляры распивающей оплетки. Даже если оплетка может быть предварительно покрыта флюсом, нанесение капли флюса на оплетку перед использованием все равно помогает процессу. Chemtronics выпускает хорошую линейку распивающих оплеток; примером является Chemtronics 60-3-5 «No-Clean Solder-Wick» (номер детали Digi-Key 60-3-5-ND).

Я расскажу об основных методах пайки в начале главы 2, где вы узнаете, как установить синий светодиод на переднюю панель Xbox.

## Инструменты для тестирования и диагностики

Электронное испытательное оборудование существует в стольких формах, сколько и электронных продуктов. Для новичка основным «обязательным» инструментом является цифровой мультиметр. Цифровые мультиметры (DMM) стали очень функциональными и доступными за последние несколько лет; типичный прибор может измерять сопротивление, напряжение, ток, емкость, полярность диода и непрерывность цепи по цене около пятидесяти долларов. Radio Shack и Jameco ([www.jameco.com](http://www.jameco.com)) предлагают разумный выбор мультиметров начального уровня. (Приложение А, «Где взять свое хакерское снаряжение», содержит предложение по мультиметру начального уровня.)

Для проектов базовой модификации и сборки комплектов цифровые мультиметры полезны для проверки закороченных соединений и проверки базовой исправности схемы до и после подачи питания. Режим непрерывности в цифровом мультиметре может быть полезен, когда вы чувствуете, что могли испортить паяное соединение.

В режиме непрерывности цифровой мультиметр будет издавать звуковой сигнал всякий раз, когда между щупами для проверки существует путь с низким сопротивлением. Таким образом, функция непрерывности полезна как для проверки целостности паяного соединения, так и для проверки коротких замыканий с соседними соединениями. Не следует использовать режим непрерывности для проверки коротких замыканий в цепи питания, поскольку некоторые платы обычно имеют достаточно низкое сопротивление между питанием и землей (около десяти Ом), чтобы вызвать звуковой сигнал непрерывности. Таким образом, перед подачей питания на

---

### Взлом Xbox: Введение в обратную разработку

любую недавно модифицированную или собранную плату используйте режим измерения сопротивления, чтобы проверить и убедиться, что на линиях питания нет коротких замыканий (нулевое сопротивление).

Для обратного проектирования и более сложных проектов вам понадобятся основные инструменты: осциллограф и иногда логический анализатор. Осциллографы полезны для захвата подробной формы электрических сигналов. С помощью осциллографа можно диагностировать проблемы синхронизации, шума и помех.

Основные определяющие характеристики осциллографа — это количество каналов или форм сигналов, которые он может отображать одновременно, и его максимальная электрическая полоса пропускания. Высококачественные осциллографы обычно имеют четыре канала и более 500 МГц полосы пропускания; уцененные или бывшие в употреблении осциллографы часто имеют только два канала и где-то между 20 МГц и 100 МГц полезной полосы пропускания. Главным ограничением всех осциллографов является то, что они могут отображать только короткий сегмент электрической формы сигнала.

Логические анализаторы полезны для захвата больших объемов цифровых данных. Они жертвуют способностью захватывать форму сигнала ради расширенных возможностей анализа данных и регистрации. Логические анализаторы полезны для диагностики сложных цифровыхшин и схем. Основными определяющими характеристиками логического анализатора являются количество цифровых каналов, которые он может оцифровывать, максимальная частота дискретизации и максимальная глубина выборки. Типичный современный логический анализатор может иметь несколько десятков каналов, частоту дискретизации в сотни мегагерц и глубину выборки в пару мегабайт. Другими функциями, обнаруженными в логических анализаторах, являются программируемые алгоритмы запуска и способность обнаруживать сбои или рант-импульсы.

К сожалению, средняя цена нового осциллографа или логического анализатора составляет от нескольких тысяч до десятков тысяч долларов. Хорошая новость заключается в том, что для большинства проектов не потребуются новейшие и самые лучшие технологии тестирования, поэтому вы можете обойтись и подержанным оборудованием. Свопфесты — это отличные места, где можно дешево купить старый осциллограф или анализатор; на eBay также время от времени бывают хорошие предложения. Если вам нужно сделать выбор между покупкой осциллографа и логического анализатора, я бы рекомендовал сначала купить осциллограф; логический анализатор далеко не так универсален, как осциллограф,

и, как правило, стоит дороже. Осциллографы можно заставить захватывать ограниченное количество логических данных, тогда как логический анализатор никогда не может быть использован для измерения аналогового сигнала. Кроме того, проще построить свой собственный самодельный логический анализатор с использованием ПЛИС и специальных плат, чем построить осциллограф сопоставимого качества. Самодельные логические анализаторы можно создать для работы в высокопроизводительных высокоскоростных приложениях, при этом затраты будут относительно дешевыми. (В главе 8 описывается, как я построил самодельный логический анализатор для прослушивания критически важной высокоскоростной шины в Xbox.)

В крайнем случае, очень простое устройство захвата цифровых трасс может быть построено примерно за пятьдесят долларов в деталях Radio Shack. Однажды мне нужно было захватить данные на порту клавиатуры PS/2, но у меня не было никакого тестового оборудования, и мне нужно было захватить данные немедленно. Макетная плата с несколькими светодиодами столбчатой диаграммы, подключенными к набору 8-битных регистров (номер детали 74НСТ574), подключенных для сдвига данных, сделала свое дело — все компоненты я купил в Radio Shack. Фактическая конструкция довольно проста, но поскольку ее использование очень ограничено, я избавлю вас от подробностей. Суть в том, что вы можете создавать свои собственные устройства для захвата цифровых данных — это то, что следует учитывать, прежде чем выкладывать несколько тысяч долларов за логический анализатор.

## Инструменты для дизайна плат

Последний набор инструментов, необходимый для завершения коллекции любого хакера, — это набор инструментов для электронного проектирования печатных плат и ПЛИС. Тема проектирования печатных плат и ПЛИС обсуждается в приложениях, но здесь стоит упомянуть, что качественные версии этих инструментов можно приобрести практически за бесценок. В результате можно спроектировать и построить полную печатную плату со сложными реконфигурируемыми аппаратными компонентами по цене ниже стоимости Xbox — включая стоимость инструментов для проектирования и строительства.

Проектирование печатных плат раньше было очень дорогим занятием; инструменты стоили тысячи долларов, а простой цикл производства платы обошелся бы в несколько сотен долларов. Сегодня новичок может изготовить простую плату менее чем за семьдесят долларов. Что касается инструментов проектирования печатных плат, Altium, ранее называвшийся Protel, продает

---

## Взлом Xbox: Введение в обратную разработку

инструмент CircuitMaker2000. Хотя я не использовал CircuitMaker2000 широко, мое первое впечатление таково, что он очень похож на теперь уже снятый с производства Protel 99SE от Altium. Загрузите бесплатную 30-дневную демоверсию или их бесплатную студенческую версию (с ограничениями), которая идеально подходит для первого проекта по проектированию, с сайта <http://www.circuitmaker.com>. После того, как вы спроектируете свою первую плату с помощью бесплатного инструмента, вы можете изготовить ее у поставщика, например Sierra Proto Express (<http://www.sierraprotoexpress.com>), примерно за 30 долларов за плату на момент написания этой статьи, с минимальным заказом в две платы. Как видите, цена больше не является серьезным препятствием, и я рекомендую вам попробовать создать один или два проекта, используя собственные печатные платы.

## Статическое электричество: убийца цепей

Статическое электричество, также известное как электростатический разряд (ESD), является бичом интегральных схем. Современные ИС особенно чувствительны к ESD; для разрушения открытого транзистора достаточно нескольких вольт. Поскольку вы не чувствуете разряды статического электричества до диапазона в сотни или тысячи вольт, вы можете разрушить такие устройства, не зная об этом.

Хорошей новостью является то, что большинство микросхем построены со специальными структурами, которые помогают сделать их более устойчивыми к электростатическому разряду. Тем не менее, лучше не участвовать добровольно в их тестировании. Чтобы нейтрализовать статическое электричество на вашем теле, всегда прикасайтесь к заземленному металлическому предмету, прежде чем прикасаться к печатной плате или микросхеме. Голый металл на корпусе компьютера, подключенного к правильно подключенной бытовой розетке, является хорошей отправной точкой.

Ношение антистатического браслета, который можно приобрести практически в любом компьютерном магазине, сведет к минимуму риск повреждения Xbox электростатическим разрядом. Чтобы браслет был эффективным, его необходимо прикрепить к заземленному объекту.

Если вы чувствуете, что живете на грани, работа босиком на не покрытом ковром бетонном полу также позволит вам быть заземленными. Голые бетонные полы на удивление проводят ток, до такой степени, что вы можете получить удар током или ожог от длительного контакта с электронным оборудованием, подключенным к неправильно подключенным розеткам. Линолеум и полы из твердых пород дерева также могут быть эффективными точками заземления, в зависимости от типа плитки или воска, использованного на полу. Можно наносить специальные токопроводящие воски или спреи, чтобы гарантировать, что пол достаточно проводящий.

FPGA — программируемые пользователем вентильные матрицы — являются решением для недорогого кремниевого прототипирования. FPGA состоит из большого массива вентилей и элементов хранения с программируемым соединением. В результате FPGA могут реализовывать все виды цифровых устройств, ограниченные только емкостью вентилей и проводов FPGA. Более крупные FPGA с емкостью в несколько миллионов вентилей могут содержать целые системы, укомплектованные микропроцессорами и периферийными устройствами. FPGA также очень доступны: 100 000 вентилей Xilinx Spartan II FPGA стоит около 20 долларов в единичных партиях. И что еще лучше, вы можете получить неограниченные среды проектирования и синтеза для FPGA Xilinx бесплатно! У Xilinx есть бесплатный продукт под названием «ISE WebPack», доступный на их веб-сайте ([www.xilinx.com](http://www.xilinx.com)), который включает такие функции, как синтез Verilog и VHDL, генерация тестового стенда HDL и программное обеспечение для анализа мощности. Verilog — это язык, подобный C, для проектирования оборудования; его можно рассматривать как строго типизированный, многопоточный C. Это отличная новость для хакеров программного обеспечения, которые хотели бы заняться оборудованием. Существуют даже сообщества по проектированию оборудования с открытым исходным кодом, такие как [www.opencores.org](http://www.opencores.org), где вы можете скачать код для микропроцессоров и других интересных цифровых компонентов, опять же бесплатно.

## Разборка Xbox

Теперь, когда мы обсудили некоторые инструменты, которые вам понадобятся для взлома, давайте займемся взломом. Первый шаг взлома Xbox — открыть коробку. Вот инструменты, которые вам понадобятся для снятия крышки:

- Биты Torx T10 и T20 (биты в форме шестиконечной звезды)
- Рукоятка отвертки для бит
- Антистатическое защитное снаряжение (см. «Статическое электричество: убийца электрических цепей»)
- Маленькая плоская отвертка (полезна, но не обязательна)

### Примечание



Прежде чем начать разбирать Xbox, помните о нескольких вещах: во-первых, всегда есть риск необратимого повреждения при разборке, а разборка Xbox аннулирует гарантию. Во-вторых,

---

Взлом Xbox: Введение в обратную разработку  
**обязательно прочтайте весь раздел, прежде чем продолжить. И в-третьих, получайте удовольствие.**

## Шаг 1: Безопасность прежде всего

**Отключите Xbox. Если оставить Xbox подключенным, вы подвергнетесь воздействию опасного, возможно, смертельно опасного напряжения.**

## Шаг 2: Открутите винты корпуса

Переверните Xbox вверх дном и осмотрите дно. Верхнюю и нижнюю половины внешней оболочки удерживают шесть винтов, и все они скрыты под этикетками или резиновыми ножками. На рисунке 1-2 показано расположение всех винтов.

Резиновые ножки приклесены к Xbox с помощью прочного клея. Для снятия ножек обычно требуется небольшая помощь плоской отвертки. Рисунок 1-3 иллюстрирует эту процедуру. После того, как вы подденьете край резиновой ножки, отклейте ее как можно ровнее, чтобы сохранить клейкую подложку на ножке. Если вы будете осторожны, вы сможете снова прикрепить ножку позже, хотя после нескольких циклов снятия клей потеряет свою липкость. В качестве замены вы можете купить резиновые ножки в любом хозяйственном магазине и прикрепить их, если вы используете Xbox на скользкой или чувствительной к царапинам поверхности.

Используйте биту Torx размера T20, чтобы удалить четыре винта, которые были под резиновыми ножками. Винты довольно длинные, но их резьба короткая, поэтому удаление должно быть быстрым.

Последние два винта спрятаны под этикеткой серийного номера и этикеткой сертификации продукта. Рисунок 1-4 иллюстрирует расположение этих винтов.



**Рисунок 1-2:** Расположение винтов корпуса Xbox. Это вид нижней части Xbox.



**Рисунок 1-3:** С помощью небольшой плоской отвертки подденьте край резиновых ножек Xbox, затем осторожно снимите их.

Чтобы найти их, сильно потрите пальцем область, где должны быть винты. Этикетка слегка вдавится в отверстия для винтов. Проколите

---

### Взлом Xbox: Введение в обратную разработку

этикетку битой и проведите или поверните биту вокруг, пока она не зацепится за отверстие для винта, и продолжайте выкручивать винт. Если вам это нужно



**Рисунок 1-4:** Расположение винтов скрыто под наклейками с серийным номером и сертификацией продукта.

Что касается косметической целостности этикеток, придерживайте их, пока выкручиваете винт, в противном случае они отклеятся или порвутся.

#### Совет



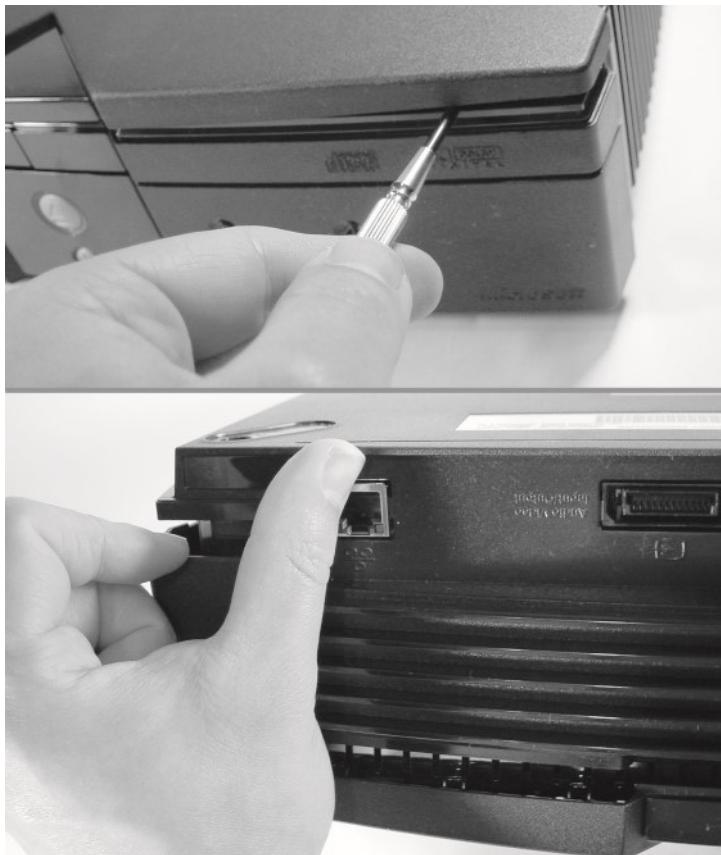
**Держите небольшой лоток или пластиковый пакет для хранения винтов, чтобы не потерять их. Винты, которые скрепляют Xbox, довольно уникальны, и вам может быть сложно купить подходящую замену в местном хозяйственном магазине.**

## Шаг 3: Снимите верхнюю крышку.

На этом этапе вы должны были снять шесть одинаковых длинных винтов. Переверните Xbox правой стороной вверх и осторожно возьмитесь за боковины коробки, используя открытые ладони рук, и попытайтесь снять крышку, слегка встяжнув ее. Если крышка не снимается этим способом, вам может потребоваться «запустить» крышку, поддев корпус пальцами сзади. В некоторых редких случаях вам также придется поддеть отверткой спереди, но будьте осторожны и аккуратны, когда вы это делаете. На рисунке 1-5 показаны некоторые

из точек, которые вы можете использовать, чтобы помочь снять корпус.

Не прикладывайте силу к крышке корпуса. Если описанные выше методы поддевания не дали результата, возможно, после публикации этой книги был добавлен дополнительный винт. Попробуйте найти винт, нашупав этикетки на задней панели Xbox.



**Рисунок 1-5:** В некоторых местах можно попытаться открыть неподатливую крышку.

#### Шаг 4: Переместите дисководы

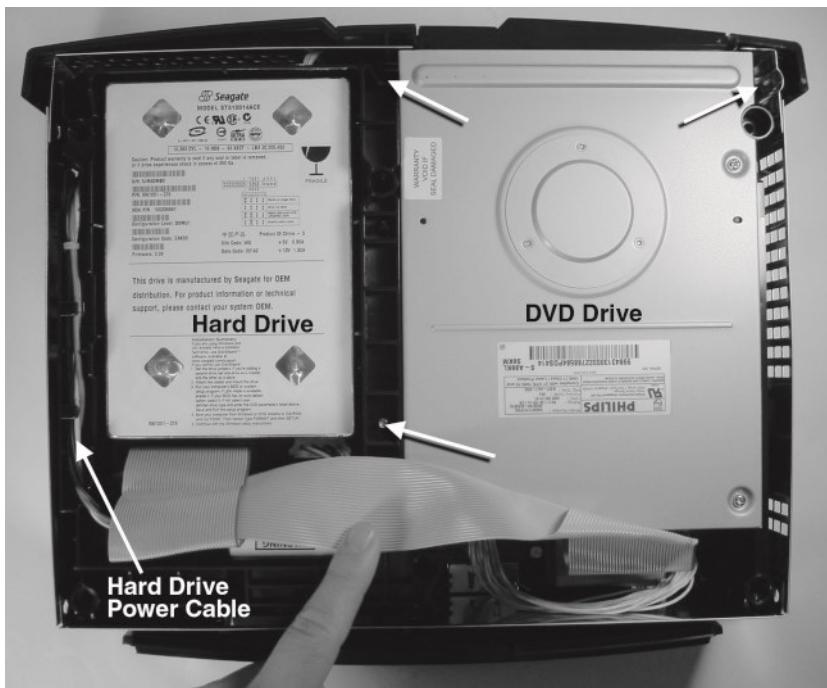
Теперь, когда вы внутри, вы должны увидеть два привода, установленных на черных пластиковых держателях. Чтобы получить доступ к материнской плате, вам нужно будет переместить (не

---

**Взлом Xbox: Введение в обратную разработку**  
обязательно снять) дисководы. Вам не нужно отсоединять кабели приводов, но вам нужно будет открутить держатели приводов.

Три винта T10 Torx удерживают держатели. Рисунок 1-6 иллюстрирует расположение этих винтов. Один спрятан под серым шлейфом IDE около задней части корпуса, а два утоплены примерно на дюйм ниже поверхности приводов около передней части корпуса. Вам может понадобиться фонарик или прямое верхнее освещение, чтобы увидеть утопленные винты.

Утопленные винты могут оказаться немного сложными, если у вашей отвертки нет намагниченного держателя бит, так как бита будет стремиться выскользнуть, когда вы поместите ее над винтом. Бита также достаточно мала, чтобы она могла зацепиться за пространство между винтом и пластиковым держателем. Если держатели привода остаются жесткими, даже если вы думаете, что вы сняли



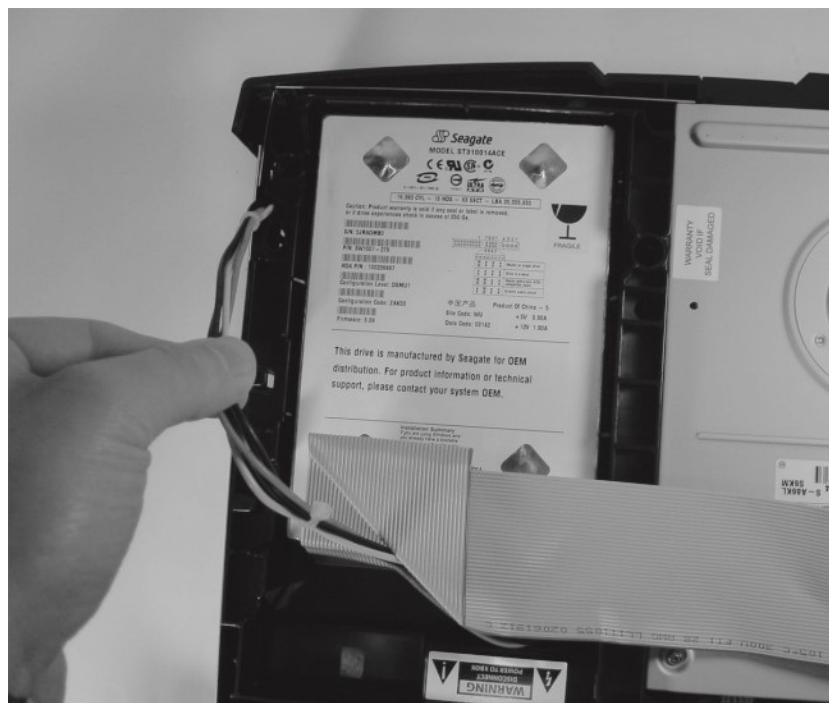
**Рисунок 1-6:** Расположение трех винтов держателя привода. Обратите внимание, что серый шлейф IDE поднят для фотографии. Коробка слева — это жесткий диск, а коробка справа — DVD-привод.

винты, дважды проверьте, чтобы убедиться, что они действительно откручены. Вы должны иметь возможность слегка приподнять держатели без чрезмерного усилия.

После того, как вы удалите винты, вам нужно будет освободить кабель питания жесткого диска, иначе он будет мешать извлечению держателей дисков. Кабель питания представляет собой черный, желтый и красный жгут проводов, идущий к внешней стороне жесткого диска. Жесткий диск — это устройство слева на рисунке 1-6. Он удерживается в выемке вдоль края держателя. Осторожно вытащите кабель из выемки так, чтобы на кабеле оставалось несколько дюймов провисания. Рисунок 1-7 иллюстрирует, какое провисание должно быть, когда вы закончите.

После того, как кабель будет освобожден, вы сможете поднять жесткий диск из места хранения. Поднимите жесткий диск и положите его на DVD-привод. Обеими руками поднимите DVD и жесткие диски из корпуса и сложите их наружу так, чтобы они свисали в сторону, как показано на рисунке 1-8. Кабели, соединяющие приводы, должны легко складываться без сопротивления. (Помните о желтом кабеле, идущем с задней стороны DVD-привода; его можно легко вытащить из гнезда, если вы не будете осторожны.)

Теперь у вас есть полный обзор материнской платы Xbox и блока питания, без отключения каких-либо приводов. Такое расположение будет полезным для тестирования Xbox после любых аппаратных модификаций.



---

Взлом Xbox: Введение в обратную разработку

**Рисунок 1-7:** Освободите кабель жесткого диска из его удерживающей выемки. После освобождения кабеля у вас должно быть несколько дюймов провисания.



**Рисунок 1-8:** Расположение дисководов с сохранением электрических соединений для тестирования и экспериментов.

**Всегда будьте осторожны с источником питания:** в нем есть напряжение, которое может травмировать или убить вас, если вы к нему прикоснетесь. Помните, что он «работает», пока Xbox подключен, даже если Xbox выключен. Также обратите внимание, что нижняя часть держателя жесткого диска и DVD-привода имеет вторичное назначение в качестве воздуховодов внутри Xbox. Работа Xbox с отключенными по бокам дисками в течение длительного времени может привести к перегреву Xbox. Кроме того, помните о больших алюминиевых радиаторах на CPU и GPU. Они могут стать неприятно горячими — потенциально достаточно горячими, чтобы обжечь вас — во время работы Xbox.

Для проекта в следующей главе (замена светодиода на передней панели Xbox) вам не нужно будет отключать жесткие диски, хотя это желательно сделать. Это предотвратит чрезмерную нагрузку на кабели при манипуляциях с корпусом Xbox.

## Шаг 5: Извлеките дисководы (необязательно)

Во многих случаях будет удобнее и безопаснее полностью извлечь дисководы.

Для этого сначала отсоедините серый шлейф IDE от материнской платы Xbox. Затем отсоедините желтый дискретный проводной кабель, подключенный к DVD-приводу от материнской платы. Этот желтый кабель подает питание на DVD-привод и передает информацию о состоянии лотка DVD-привода на материнскую плату. Не дергайте ни за один провод в желтом дискретном проводном кабеле, иначе вы можете отсоединить провод от головки кабеля. Предпочтительный способ отсоединения кабеля — взяться за белый разъем и потянуть; однако, если ваши пальцы недостаточно малы, чтобы пролезть в узкое пространство, вы можете взяться за весь жгут проводов и осторожно потянуть, чтобы снять разъем. Отложите DVD-привод в сторону.

Затем отсоедините разъем питания жесткого диска. Вы увидите, что разъем оченьочно закреплен в жестком диске. Если вы просто потянете разъем напрямую, вы рискуете пораниться об острые края корпуса, когда разъем выйдет из диска. Чтобы избежать травмы, используйте небольшую плоскую отвертку, чтобы аккуратно отсоединить разъем от корпуса жесткого диска, как показано на рисунке 1-9.

Теперь вы можете полностью снять блок дисковода. Серый ленточный кабель IDE по-прежнему будет охватывать отдельные блоки дисковода; вы можете снять их, если хотите, но запомните его ориентацию, чтобы позже вы могли снова подключить диски к Xbox.

## Сборка Xbox

Теперь, когда вы разобрали Xbox, переходите к следующим разделам, чтобы попробовать несколько интересных проектов. Когда закончите, прочитайте этот раздел, чтобы узнать, как собрать Xbox.



**Рисунок 1-9:** Отсоедините разъем питания жесткого диска с помощью плоской отвертки.

Прежде чем прикреплять что-либо к Xbox, переверните его вверх дном и осторожно встряхните, чтобы убедиться, что нет ослабленных винтов или деталей, которые вы могли случайно уронить в Xbox. Ослабленный винт означает конец вашей игровой консоли и представляет потенциальную опасность возгорания, так что это стоящая проверка, если у вас есть какие-либо сомнения.

Первым шагом в повторной сборке Xbox является повторное присоединение дисководов. Если вы следовали процедуре из предыдущего раздела, ваш DVD-привод и жесткий диск уже должны быть присоединены серым шлейфом IDE. Если это не так, присоедините конец кабеля с наименьшим количеством складок к жесткому диску, а затем присоедините средний разъем кабеля к DVD-приводу. Кабели будут присоединяться только в одном направлении; обратите внимание на выступ на разъеме и положение выемки на разъемах привода.

Подключите оставшийся свободный конец серого ленточного кабеля IDE к материнской плате Xbox. Кабель будет подключаться к материнской плате только в одном направлении; обратите внимание на выступ в середине разъема IDE и выемку на гнезде материнской платы. Теперь подключите желтый кабель DVD к материнской плате

Xbox. Этот кабель также будет подключаться к разъему материнской платы только в одном направлении, и у него также есть выступы на кабеле и соответствующие выемки на гнезде материнской платы. Теперь установите DVD-привод в Xbox. Проверьте, что он установлен заподлицо и ровно, посмотрев, как лоток DVD-привода с логотипом Xbox расположен относительно края корпуса. Вероятно, вам придется попробовать установить привод пару раз, прежде чем он сядет правильно. Используйте складки на сером ленточном кабеле IDE, чтобы помочь вам, если вы запутались в ориентации различных частей.

Вы почти закончили. Положите жесткий диск на место рядом с DVD-приводом. Опять же, этот привод, вероятно, придется немного покачивать, чтобы он встал на место. Привод должен быть на одном уровне с

DVD-приводом и заподлицо с краями металлического экрана электромагнитного излучения вокруг корпуса Xbox. Пропустите кабель питания жесткого диска через его фиксирующую выемку в лотке для дисковода и подключите его к жесткому диску. Вам нужно будет приложить изрядное усилие к разъему, чтобы создать надежное соединение; вы должны почувствовать легкий щелчок, когда разъем полностью войдет в зацепление. Наконец, пропустите серый ленточный кабель IDE через его оригинальный фиксирующий крючок на держателях дисковода.

На этом этапе хорошей идеей будет подключить Xbox, подключить его к телевизору и проверить, запускается ли Xbox должным образом. Вы можете запускать Xbox неограниченное количество времени со снятой крышкой, так как все охлаждающие каналы, образованные нижней частью держателей дисков и DVD-приводом, находятся на месте. Если диски не подключены должным образом к Xbox, консоль все равно загрузится, но на ней отобразится сообщение о том, что вашей консоли требуется обслуживание. Внимательно проверьте ваши соединения, если появится это сообщение.

Теперь пришло время прикрутить приводы на место. На этом этапе у вас должно быть девять винтов: три коротких винта T10 для держателей приводов и шесть длинных винтов T20 для крышки корпуса. Не паникуйте, если вам не хватает пары винтов или если у вас есть пара лишних. Xbox все равно будет держаться вместе, даже если вам не хватает одного или двух винтов. Прикрепите винты и крышку корпуса в обратном порядке их снятия. (См. рисунки выше в этом разделе, если вам нужно напоминание о том, где они находятся.)

После установки крышки корпуса вы снова готовы к использованию Xbox!

## CHAPTER2. Мысление внутри коробки

Обратное проектирование можно рассматривать как очень сложную, но очень полезную игру. Чтобы победить, вам понадобится немало мастерства и немало удачи. И как в любой игре, чтобы развить свои навыки, вам просто нужно играть, играть, играть.

Первый шаг в развитии навыков хакера — развить интуицию к материалу. В случае с оборудованием хороший способ почувствовать венцы — снять крышки со всего и попытаться выяснить, что представляют собой все компоненты и что они могут делать. Также полезно заказать бумажный каталог у поставщика деталей, например Digi-Key, Jameco или Newark Electronics, и просто листать страницы в свободное время. Сначала чтение каталога деталей может показаться чтением словаря, но по мере того, как вы будете рассматривать все больше и больше печатных плат, вы постепенно обнаружите, что все имеет смысла.

Следующим по мощности инструментом обратного проектирования является сопоставление с образцом. Все инженеры-разработчики оборудования ограничены одними и теми же законами природы, и все инженеры-разработчики оборудования используют одни и те же типы строительных блоков. Инженеры также любят модулировать и повторно использовать существующие проекты. В результате во многих проектах можно найти один и тот же мотив дизайна. Распознавание мотивов дизайна позволит вам определить функцию схем, даже если вы не знаете ни одного номера детали. Аналогично, можно добиться значительных успехов в обратном проектировании без какой-либо формальной подготовки в области электротехники.

Последний инструмент обратного проектирования — эксперимент. Когда интуиция и сопоставление с образцом не раскрывают секреты схемы, необходимо прибегнуть к зондированию и возмущению системы и попытаться вывести функцию на основе наблюдаемых ответов. Хотя экспериментирование может привести к отказу оборудования, можно утешиться тем фактом, что большинство потребительского оборудования разработано для зондирования и тестирования в качестве требования для производства. Кроме того, в случае Xbox можно утешиться тем фактом, что новый Xbox относительно недорог. Покупка двух коробок заранее и обращение с одной из них как с «жертвенной» коробкой помогает устраниить психологический барьер, который в противном случае мог бы возникнуть при проведении агрессивных экспериментов с оборудованием.

В этой главе вы познакомитесь с основами обратного проектирования, уделив особое внимание базовым методам, таким как чтение печатных

плат для формирования интуитивного представления, а также немного рассмотрев промежуточные методы, такие как сопоставление с образцом и распознавание основных мотивов дизайна.

## Чтение печатной платы

Первое, что вы видите, когда снимаете крышку типичного электронного устройства, — это печатная плата. Обычно окрашенный в зеленый или бежевый цвет, этот многослойный сэндвич из меди, стекловолокна и эпоксидной смолы содержит точный список схемных соединений в своих дорожках. Другими словами, следуя дорожкам, можно точно определить, как подключен каждый компонент. Размещение компонентов и компоновка дорожек также содержат подсказки, которые могут пролить свет на мыслительный процесс проектировщика.

## Основа печатных плат

Типичная печатная плата состоит из нескольких слоев узорчатой меди, разделенных тонкими листами стекловолокна, пропитанного эпоксидной смолой. Цвет необработанной печатной платы беловатый или коричневый с медными следами; однако почти все печатные платы покрыты тонким полимером, называемым паяльной маской, который придает печатным платам их знакомый зеленый цвет. Расплавленный припой не прилипает к паяльной маске, поэтому во время производства излишки припоя не прилипают к плате и не вызывают коротких замыканий. Паяльная маска имеет отверстия для соединений с компонентами. Эти отверстия обычно имеют серебристый цвет из-за тонкого покрытия оловом или припоеем, которое наносится для предотвращения окисления меди и улучшения паяемости.

Поверх паяльной маски обычно находится слой белых букв, называемый шелкографией. Каждый компонент на печатной плате имеет контур и уникальное обозначение на слое шелкографии. Обозначение позволяет людям быстро связать компонент на печатной плате с компонентом на схеме. Вы можете использовать обозначение, чтобы помочь угадать функцию компонента на основе схемы наименования компонентов. Таблица 2-1 суммирует схему наименования компонентов, используемую в Xbox.

### Совет



**Материнская плата Xbox включает в себя удобную систему координат, напечатанную по краям платы в слое шелкографии. На стороне компонентов платы координаты идут от A до G по бокам и от 1 до 8 по верху и низу. Обратная сторона платы имеет координаты от M до V по бокам. Обратите внимание, что буквы I, O, Q и S пропущены, поскольку их можно спутать с цифрами 0, 1 и 5.**

**Обозначения компонентов на материнской плате Xbox кодируются с использованием этой системы координат; таким образом, J7D1, отладочный порт LPC, можно найти на верхней стороне в координатах 7D. В этой книге эта система координат будет часто использоваться вместе с обозначениями компонентов для обозначения конкретных компонентов.**

Соединения между слоями проводки выполняются с помощью заполненных медью отверстий, называемых переходными отверстиями. Поскольку стоимость печатной платы растет с количеством слоев, большинство потребительских электронных устройств проектируются так, чтобы свести количество слоев к минимуму. Радиоприемники и аудиоусилители обычно используют односторонние платы, тогда как новейшие материнские платы ПК могут иметь до шести или восьми слоев. Материнская плата Xbox имеет четыре слоя. Верхние два слоя предназначены для передачи информации между чипами, а внутренние два слоя предназначены для подачи питания. Материнская плата Xbox на первый взгляд покажется непрозрачной, поскольку внутренние слои питания в основном представляют собой сплошные листы меди. Хорошей новостью для обратного проектирования является то, что мы можем отследить каждое соединение на материнской плате Xbox с помощью обычного визуального осмотра, поскольку все сигнальные слои находятся снаружи непрозрачных слоев питания. Конструкция Xbox контрастирует с материнскими платами, которые скрывают два или четыре сигнальных слоя внутри слоев питания. Скрытые сигнальные слои могут затруднить отслеживание сигналов. (Обратите внимание, что решение скрыть сигналы внутри силовых

Обозначитель	Тип компонента
C	Конденсатор
R	Резистор
U	Интегральная схема или транзистор
L	Индуктор
RP	Резисторный пакет
Q	Транзистор
CR	Диод
J	Соединитель или перемычка
RT	Самовосстанавливающийся предохранитель
Y	Кристалл

Таблица 2-1: Схема именования компонентов Xbox.

(Разделение слоев обычно обусловлено не соображениями безопасности, а скорее физикой взаимодействия электрических сигналов между слоями печатной платы.)

Отследить сигнал довольно просто. Начиная с соединения исходного компонента с платой, следуйте медной дорожке. Если дорожка пересекается с окружностью, то есть большая вероятность, что сигнал продолжается на противоположной стороне платы. Если дорожка заканчивается и нет соединения с другой стороной платы, то есть большая вероятность, что дорожка подключена к одной из силовых плоскостей.

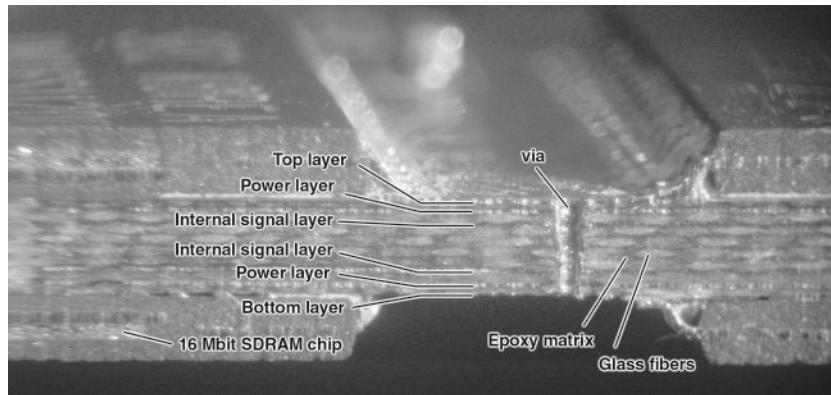


Рисунок 2-1: Поперечное сечение типичной печатной

#### платы. **Попробуйте это**



Попробуйте отследить некоторые сигналы на материнской плате Xbox. На материнской плате Xbox взгляните на разъем J8C1, 40-контактный разъем IDE в секторе 8С. Почти все сигналы от разъема IDE идут на одну микросхему, MCPX, на материнской плате через несколько наборов резисторов. Какой вывод вы можете сделать об этой микросхеме? Обратите внимание, как некоторые из дорожек, идущих от разъема IDE, извиваются туда-сюда. Это метод, используемый для того, чтобы попытаться гарантировать, что все провода имеют одинаковую длину. См. боковую панель «Почему дорожки на печатной плате извиваются везде?» для получения дополнительных объяснений.

## Компоненты

Теперь, когда у вас есть небольшой опыт отслеживания сигнала, пришло время узнать, как выглядят некоторые основные компоненты.

Компоненты классифицируются как пассивные и активные. Грубо говоря, пассивные компоненты не могут усиливать сигнала, поэтому они обычно

имеют всего два вывода. Иногда несколько пассивных компонентов упакованы вместе, поэтому один пакет пассивных компонентов будет иметь несколько выводов. Пассивные компоненты включают конденсаторы, резисторы и индукторы. Наиболее распространенными пассивными компонентами на материнской плате Xbox являются конденсаторы. Конденсаторы хранят энергию в виде электрического заряда; в Xbox они

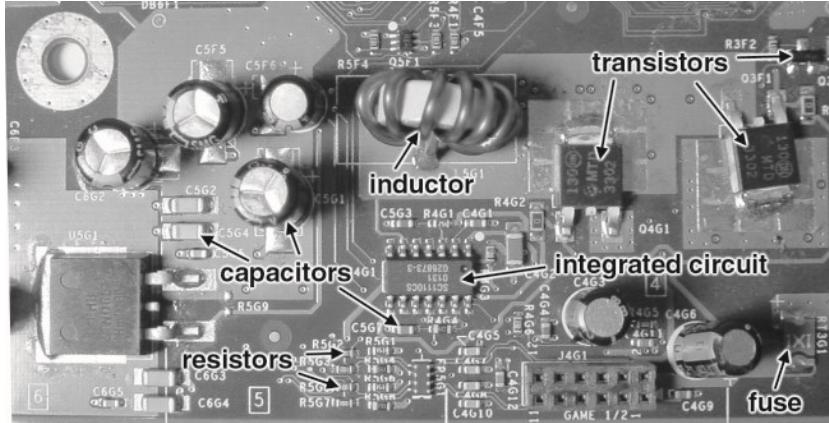


Рисунок 2-2: Типичные пассивные компоненты Xbox.

### Почему дорожки на печатной плате повсюду извиваются?

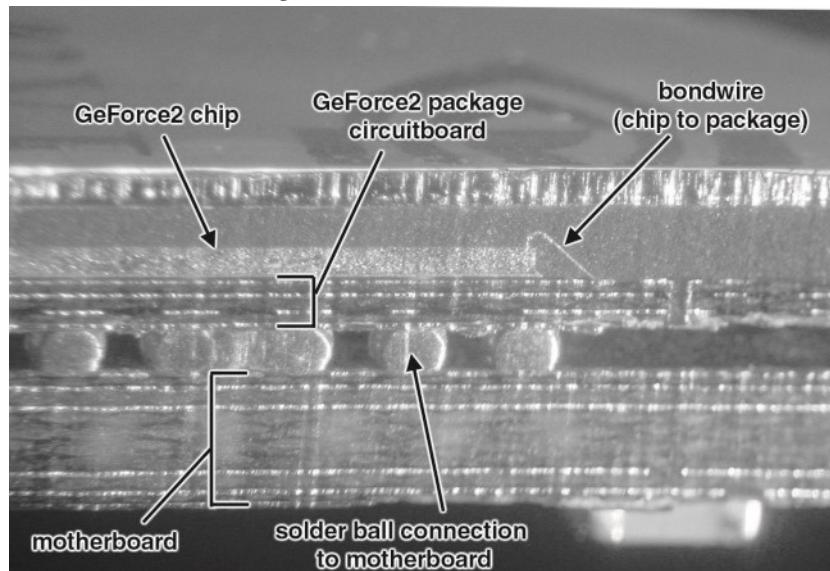
Посмотрев на несколько печатных плат, вы, вероятно, начнете замечать, что дорожки на печатной плате часто извиваются повсюду, иногда возвращаясь туда и обратно несколько раз, прежде чем соединиться с местом назначения. Это кажется бессмысленным, когда прямая дорожка справилась бы с задачей. Однако редко вы найдете структуру на печатной плате, которая была размещена как каприз разработчика. Оказывается, скорость сигналов в большинстве высокопроизводительных электронных устройств, примерно  $\frac{1}{4}$  скорости света, медленнее по сравнению со временем, необходимым для того, чтобы сигнал достиг своего места назначения. Например, сигнал пройдет всего 3 дюйма по печатной плате за один такт процессора с частотой 1 ГГц (один такт на частоте 1 ГГц составляет длительность 1 миллиардной доли секунды, или одну наносекунду). Таким образом, два сигнала, исходящие из одного и того же чипа, прибудут к месту назначения в совершенно разное время, если длины дорожек сильно различаются. Чтобы бороться с этим, разработчики добавляют дополнительные изгибы в более короткую дорожку, чтобы эффективная длина дорожки была такой же, как и у более длинной.

В основном используется для сглаживания локальных колебаний мощности, возникающих при переключении цифровой логики КМОП, а также для подавления высокочастотных шумов.

Другие крупные пассивные компоненты, обнаруженные на материнской плате Xbox, включают индукторы и резисторы. Большие тороидальные (в форме пончика) индукторы с проволочной обмоткой, обнаруженные на материнской плате Xbox, являются частью подсистемы питания. Индукторы хранят энергию в виде магнитного потока. Электрические свойства индуктора являются дополнительными к свойствам конденсатора. Комбинации индукторов и конденсаторов с транзисторными переключателями между ними используются для создания очень эффективных регуляторов мощности. Большинство резисторов на материнской плате Xbox используются либо для поглощения избыточной энергии в конце сигнальных дорожек, либо для смещения провода на определенный логический уровень.

Существует два способа идентификации пассивного устройства на материнской плате Xbox. Первый — по форме корпуса. Распознавание формы корпуса осуществимо, поскольку существует очень мало основных разновидностей пассивных деталей. На рисунке 2-2 приведены некоторые изображения конденсаторов, индукторов и резисторов, которые вы можете увидеть на материнской плате Xbox. Второй способ — прочитать этикетку рядом с деталью на материнской плате и сделать вывод о функции детали по ее условному обозначению, используя таблицу 2-1 в качестве руководства.

Активные компоненты могут усиливать сигналы и иметь три или более выводов. Простейший активный компонент — транзистор с тремя, а иногда и четырьмя выводами (иногда дискретные транзисторы «MOSFET» имеют явный четвертый



**Рисунок 2-3:** Поперечный разрез детали в корпусе BGA (GeForce2), установленной на материнской плате.

## Что делают все эти резисторы и конденсаторы на цифровой плате?

Шаблон, который вы заметите на многих печатных платах, — это преобладание резисторов и конденсаторов. Конденсаторы повсюду, потому что они помогают свести шум к минимуму и стабилизировать напряжение питания. Они необходимы, потому что медные плоскости, используемые для распределения питания, имеют небольшое сопротивление и индуктивность. Эти небольшие паразитные элементы могут вызвать большие проблемы, когда через источник питания переключается большой ток. Точное размещение и выбор конденсаторов считается чем-то вроде черного искусства. Если вы случайно уроните один из крошечных конденсаторов размером с песчинку на печатной плате во время работы с ней, есть вероятность, что вы сможете обойтись без его замены. Однако, учитывая дефект такого рода, наиболее вероятной проблемой, с которой вы столкнетесь, будут периодические проблемы с надежностью.

В то время как конденсаторы повсюду обеспечивают локальное хранение энергии для всех компонентов, резисторы удаляют избыток энергии. Быстрые сигналы на материнской плате несут много энергии, и если энергия не рассеивается на приемнике контролируемым образом с помощью резистора, энергия сигнала будет отражаться обратно на передатчик и вызывать проблемы. Это явление похоже на явление звука в спортзале. Когда вы говорите в пустом спортзале, возникает эхо. Если вы говорите слишком быстро, люди не смогут вас понять, потому что эхо начнет мешать вашей речи. Однако, если вы покроете стены спортзала пеной, эхо будет поглощаться пеной, и вы сможете говорить без помех от вашего эха.

Резисторы похожи на акустическую пену, которую вы наносите на стены, чтобы заглушить эхо, чтобы схемы могли общаться друг с другом на высокой скорости. В отличие от большинства конденсаторов, если вы случайно заденете один из этих резисторов во время игры, вам придется заменить его, чтобы схема работала правильно. Эти «согласующие резисторы» часто упаковываются по четыре или восемь штук в пакет, поэтому они выглядят почти как небольшие интегральные схемы. Вы можете отличить блоки резисторов от других компонентов, потому что они блестящие, слегка бугристые, имеют белую окантовку и рядом с ними будет префикс условного обозначения «RP». При отслеживании сигнала через блок резисторов можно с уверенностью предположить, что сигналы проходят напрямую, так что соединение с одной стороны идет прямо к штырю на другой стороне.

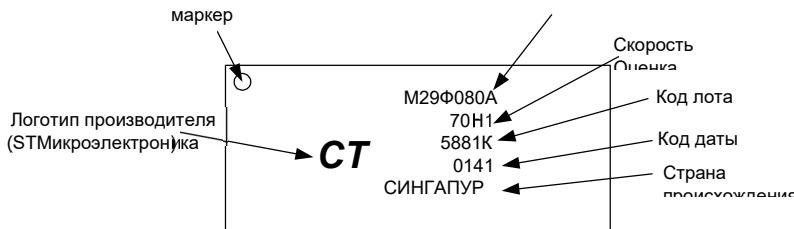
терминал «гело»). Наиболее сложными активными компонентами являются интегральные схемы, такие как микросхемы ЦП и памяти, с сотнями, иногда тысячами выводов. Интегральные схемы выпускаются в самых разных корпусах, и иногда соединения скрыты под корпусом, как в случае с корпусом Ball Grid Array (BGA). Графический чип, MCPX и ЦП на материнской плате Xbox используют корпуса BGA. На рисунке 2-3 показано поперечное сечение устройства BGA, показывающее скрытые под ним соединения.

Определение функции конкретной интегральной схемы сложнее, чем определение функции пассивного устройства. Функционально идентичный кремний можно приобрести в различных корпусах, которые могут выглядеть совершенно по-разному. В некоторых случаях вы можете угадать функцию устройства, наблюдая за тем, к чему подключено устройство или как оно выглядит, но самый надежный метод — считать номер детали с чипа и найти его в Интернете. (Обычно детали имеют какой-то логотип или префикс номера детали, который идентифицирует производителя, который вы можете использовать, чтобы найти больше данных об устройстве, посетив веб-сайт производителя.) Если вы не узнаете логотип или префикс номера детали, перечисленные ниже службы могут помочь вам найти функции детали.

1. [www.findchips.com](http://www.findchips.com) может взять номер детали или части номера детали и выполнить поиск по инвентарным запасам многих дистрибуторов для совпадений по инвентарным запасам. Наиболее распространенные детали будут отображаться в FindChips, и предоставленные ссылки часто приведут вас не только к краткому описанию детали, но также к информации о ценах и заказе.
2. [www.google.com](http://www.google.com) индексирует все в Интернете, и номера деталей не являются исключением. Google также можно использовать для поиска веб-сайтов производителей, если вы делаете запрос по буквам в логотипе и описательному термину, например, «полупроводники». На веб-сайте производителя вам, вероятно, понадобится найти специализированную поисковую систему по деталям, скрытую на веб-сайте, или перейти на подстраницу полупроводниковой продукции, чтобы выполнить поиск номера детали. Функция поиска на главной странице веб-сайта компании иногда находит номера деталей, но чаще всего она индексирует только бесполезные корпоративные и маркетинговые страницы.
3. Если ни один из этих сервисов не помог вам, попробуйте убрать некоторые префиксы и суффиксы из номера детали. В нашем примере M29F080A запрос только номера детали 29F080 приведет вас на веб-страницы нескольких производителей, которые выпускают детали, функционально совместимые с деталью STMicroelectronics.

Штифт 1

Номер детали



**Рисунок 2-4:** АнATOMия типичного номера детали ИС. На схеме изображен чип в месте расположения U7D1 на материнской плате Xbox.

### Попробуйте это



Давайте попробуем найти номер детали Xbox. Найдите U7D1 на материнской плате Xbox. Рисунок 2-4 иллюстрирует то, что вы можете найти. Номер детали, как правило, является самым длинным номером на чипе и часто начинается с одного или двух буквенных символов. Микросхемы памяти и процессоры также часто имеют суффикс класса скорости или качества после номера детали. Кроме того, почти все чипы имеют код даты. Коды даты обычно представляют собой четырехзначное число в формате YY-WW, где YY — год изготовления чипа, а WW — рабочая неделя. В нашем примере наша деталь M29F080A была изготовлена в 41-й неделе 2001 года в Сингапуре, и имеет класс скорости 70N1. Оставшийся номер, 5881K, является кодом партии, значение которого различается у разных производителей, но в целом связывает чип с конкретной кремниевой пластиной или номером отслеживания партии кремниевой пластины на производственном предприятии. Логотип «ST» указывает, что производителем этого чипа является STMicroelectronics, и, к счастью, веб-сайт этого производителя можно быстро найти через Google или угадывая, поскольку URL-адрес компании просто [www.st.com](http://www.st.com). Ввод номера детали M29F080A в поле поиска на главной странице приведет вас непосредственно к результатам поиска, которые включают подробные технические описания и описания этой детали — 8-мегабитной флэш-памяти Uniform Block Single Supply.

## Контрольные точки

Почти все печатные платы в потребительской электронике имеют структуры, разработанные для ускорения тестирования готовой платы на заводе. Эти «контрольные точки» существуют, чтобы справиться с печальной реальностью производственных дефектов. Xbox не является исключением, когда дело касается контрольных точек и производственных дефектов. Нижняя часть материнской платы Xbox заполнена сотнями контрольных точек — крошечных серебристых кружочков — которые позволяют контактному щупу получить доступ почти к каждому интересному сигналу в Xbox. Эти контрольные точки — желанный подарок для реверс-инженеров и людей, которые хотят модифицировать свое оборудование, потому что они обеспечивают легкий доступ к сигналам, которые в противном случае потребовали бы микроскопа и твердой руки.

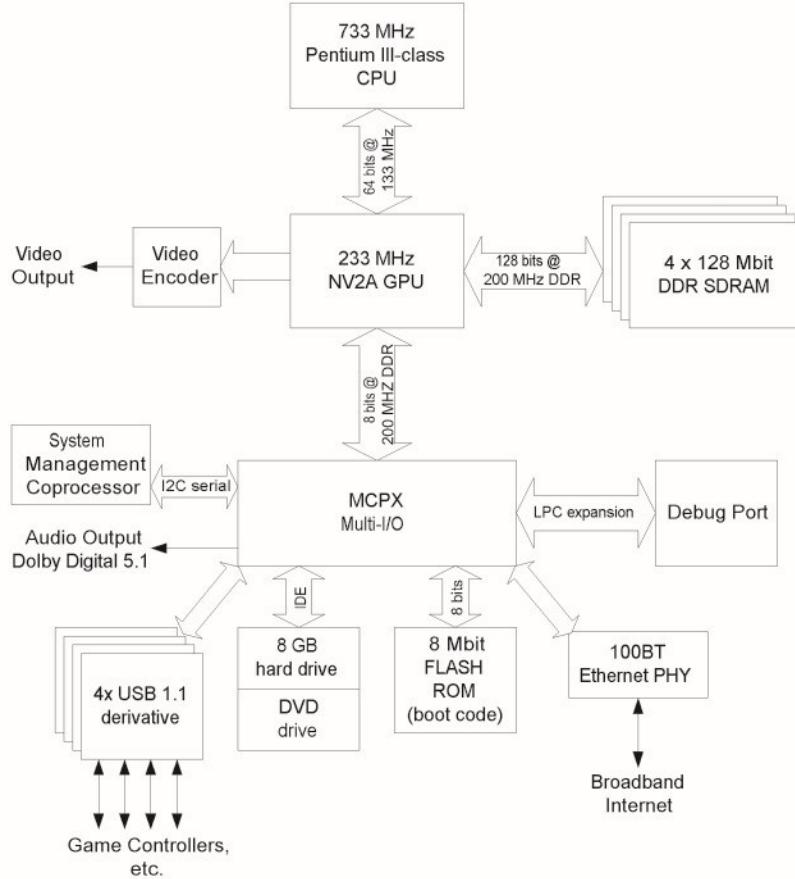
Набор контрольных точек одновременно проверяется на производственной линии с помощью оборудования, называемого «тестером с гвоздями». Тестер с гвоздями, как и следовало ожидать, состоит из сотен подпружиненных структур «пружинных штифтов». Материнская плата выравнивается по испытательному стенду и зажимается либо механическими плунжерами, либо вакуумным зажимом. Аналогичным образом вы можете использовать пружинные штифты для самостоятельной модификации материнской платы Xbox без пайки, используя контрольные точки. Вам нужно будет собрать собственные печатные платы (см. Приложение), но результатом будет плата, которую вы сможете установить, просто прикрутив ее — пайка не потребуется!

## Архитектура Xbox

Прежде чем погрузиться в примеры сопоставления шаблонов, нам понадобится ссылка на шаблон. Давайте воспользуемся этой возможностью, чтобы изучить внутреннюю архитектуру Xbox в качестве ссылки на шаблон, и в конечном итоге сравним архитектуру Xbox с ПК и с другой игровой консолью.

### Высокий уровень организации

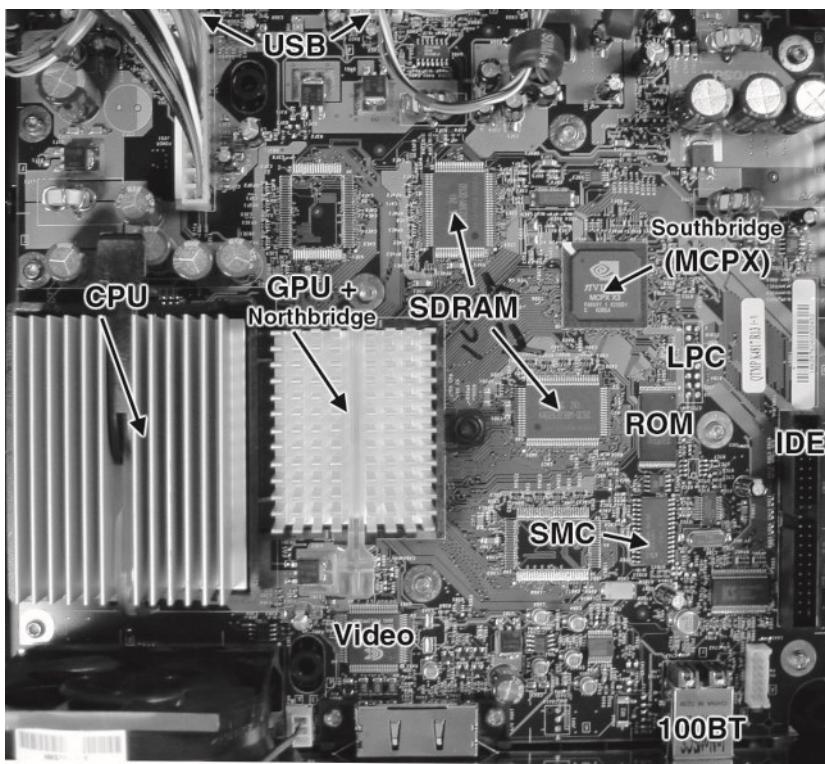
Xbox имеет процессор класса Pentium-III, работающий на частоте 733 МГц в качестве центрального процессора. Номер «S-Spec» на центральном процессоре наиболее близок к номеру Mobile Celeron. Процессор подключен через стандартную переднюю шину P6 133 МГц (FSB) к графическому процессору (GPU) и комбинированному чипу северного моста, называемому NV2A от nVidia. Его ближайший родственник для ПК — чип nForce IGP от nVidia. Поскольку логика северного моста и GPU объединены



**Рисунок 2-5:** Архитектурный вид Xbox высокого уровня.

В одном чипе ЦП и графические процессоры могут совместно использовать общий банк памяти. Это называется «унифицированная архитектура памяти» (UMA).

По сравнению с традиционной архитектурой с разделением видео/основной памяти, UMA обходится дешевле, поскольку устраняет необходимость в выделенной видеопамяти. Однако в определенных ситуациях UMA имеет более низкую производительность, поскольку вводит конкуренцию за доступ к памяти между основным процессором и графическим процессором. Чтобы смягчить часть этой конкуренции, системная память часто разделяется на несколько банков. Например, nForce IGP разделяет память на два банка, к которым могут независимо обращаться как GPU, так и CPU через коммутационную сеть.



**Рисунок 2-6:** Фотография материнской платы Xbox с маркировкой основных компонентов.

Графический процессор подключен к кухонному чипу под названием «MCPX» через быструю узкую шину, называемую шиной HyperTransport. MCPX объединяет чип южного моста и почти все периферийные устройства Xbox, включая контроллеры USB, устаревший интерфейс загрузочного ПЗУ, аудиопроцессор Dolby Digital, контроллер IDE для массового хранения, контроллер Ethernet и интерфейсы для функций управления системой.

Подключение всех основных блоков в Xbox показано на рисунке 2-5, а рисунок 2-6 иллюстрирует расположение этих блоков на реальной материнской плате Xbox.

## Функциональные детали

В следующих разделах представлен краткий обзор частей, составляющих архитектуру Xbox. Мы уделяем особое внимание деталям, необходимым для понимания того, как провести обратную разработку механизмов безопасности Xbox.

## Процессор

Центральный процессор (ЦП) — это вычислительное сердце обычного компьютера. Тема архитектуры ЦП заслуживает отдельной книги, поэтому мы рассмотрим только материал, необходимый для понимания того, как провести обратную разработку Xbox. В частности, мы рассмотрим, как получить контроль над ЦП Xbox.

Процессор считывает последовательности инструкций, хранящихся в памяти — программы — которые говорят процессору выполнять различные вычисления или принимать решения на основе доступных данных.

Инструкции хранятся в памяти в виде чисел, называемых кодами операций. Коды операций принимают операнды в качестве аргументов. Программисты используют алфавитные мемоники при написании низкоуровневого машинного кода, чтобы им не приходилось запоминать сотни номеров кодов операций. Например, своего рода инструкция вычитания размером в байт имеет код операции 0010.1000 (двоичный) или 0x28 (шестнадцатеричный) и мемонику «SUB». Требуемый код операции вычитания варьируется в зависимости от источника и ширины данных вычитания. Отслеживание всех правил кодов операций в операнды является непосильной задачей, поэтому процесс перевода мемоник и операндов в номера инструкций выполняется с помощью программы, называемой ассемблером. Аналогично, процесс перевода номеров инструкций обратно в мемоники выполняется с помощью дизассемблера. Примечательно, что большинство программ не пишутся на языке ассемблера; обычно используется язык более высокого уровня, такой как С. Эти языки высокого уровня транслируются в машинные инструкции с помощью компиляторов. Автоматическая декомпиляция машинных инструкций обратно в язык высокого уровня может быть затруднена, поскольку процесс компиляции — особенно оптимизированной компиляции — отбрасывает большую часть высокоуровневой структурной информации, содержащейся в исходном коде.

Процессор отслеживает, какая инструкция выполняется, с помощью указателя инструкций (IP). IP также упоминается как счетчик программ (PC) в некоторых контекстах. IP обычно продвигаются по программе по одной инструкции за раз, если только не встречается инструкция ветвления. Инструкция ветвления дает программе возможность принять решение, проверив данные внутри ЦП и перейдя в новое место на основе результата.

Проверка. Понимание движения указателя инструкций является центральной частью обратного проектирования Xbox. Возможность манипулировать IP равносильна контролю над тем, что Xbox может делать, а что нет. Меры безопасности, реализованные в архитектуре программного обеспечения Xbox, пытаются гарантировать, что IP всегда выполняет только код, одобренный Microsoft, всегда криптографически проверяя фрагмент кода на подлинность перед его запуском.

## Двоичные и шестнадцатеричные числа

Цифровые схемы используют 1 и 0 для представления чисел. Эта двоичная, или «основанная на 2», нотация является отражением способа использования электрических сигналов для представления чисел: два диапазона уровней напряжения используются для определения одного или другого логического состояния. Можно построить электрические системы, которые представляют информацию, используя более двух уровней напряжения, но только за счет мощности и сложности. Современные модемы, например, используют несколько уровней напряжения и информацию о фазе для представления нескольких бит данных в одной единице времени.

Состав чисел и арифметика в двоичной системе следуют тем же правилам, что и наше знакомое десятичное представление («основание 10»). В десятичной системе 0 используются в качестве заполнителей для запоминания переполнений. Например, 1 больше 9 приводит к переполнению, потому что нет ни одной цифры больше 9. Следовательно, число 10 записывает, что у нас было одно переполнение самой правой десятичной позиции. Аналогично, в двоичной системе 1 больше 1 равно 10, поскольку самая большая отдельная цифра в двоичной системе равна 1.

Таким образом, в десятичной системе счисления значение четырехзначного десятичного числа  $d_4d_3d_2d_1$  можно разбить следующим образом:

$$d_4 * 10^3 + d_3 * 10^2 + d_2 * 10^1 + d_1 * 10^0 = d_4 * 1000 + d_3 * 100 + d_2 * 10 + d_1 * 1$$

Аналогично, четырехзначное двоичное число  $b_4b_3b_2b_1$  можно разбить следующим образом:

$$b_4 * 2^3 + b_3 * 2^2 + b_2 * 2^1 + b_1 * 2^0 = b_4 * 8 + b_3 * 4 + b_2 * 2 + b_1 * 1$$

Например, число  $1010 = 1 * 8 + 0 * 4 + 1 * 2 + 0 * 1 = 10$  в десятичной системе.

Отслеживание чисел в прямом двоичном формате может быстро стать обременительным; например, для представления десятичного числа 968 вам понадобится десять двоичных цифр. Чтобы сэкономить место на экране, двоичные числа преобразуются в восьмеричные или шестнадцатеричные. Восьмеричный формат, или «основание 8», был популярен на заре компьютеров, но с тех пор стал редкостью. Шестнадцатеричная, или «основание 16», является фактической системой исчисления. В шестнадцатеричном формате 16 цифр, поэтому шестнадцатеричные цифры, соответствующие десятичным числам от 10 до 15, представлены буквами от A до F. Таблица 2-2 суммирует преобразование между двоичным, десятичным и шестнадцатеричным форматами для первых 16 положительных целых чисел. Чтобы отличить шестнадцатеричные числа от десятичных, многие люди используют соглашение языка C, где `0x[число]` представляет шестнадцатеричное число, а `[число]` неявно является десятичным числом. Двоичные числа не имеют аналогичного стандарта в C, поэтому некоторые люди используют стандарт Verilog, `[цифры]b[число]`, где `[цифры]` — это количество цифр в двоичном числе. Сuffix «b» после строки из 1 и 0, например, `1010.1100.1110b`, также используется для обозначения двоичного числа. Обратите внимание, как «.» использовалось

для группировки двоичных цифр в наборы по четыре; это помогает мысленно перевести двоичное число в шестнадцатеричное: 0xACE.

(сделаем паузу)

## Двоичные и шестнадцатеричные числа (продолжение)

Корзина Дек Гекс

Бин Дек Хекс

0000	0	1000	8 8
0001	1	1001	9 9
0010	2	1010	10 А
0011	3	1011	11 Б
0100	4	1100	12 С
0101	5	1101	13 Д
0110	6	1110	14 В
0111	7	1111	15 Ф

**Таблица 2-2:** Таблица преобразования двоичных, десятичных и шестнадцатеричных чисел.

Сердцем ЦП является крошащая, но очень быстрая память, называемая регистровым файлом. Несколько фрагментов данных могут быть записаны и считаны из регистрового файла за каждый такт процессора. Данные из регистрового файла подаются в исполнительное устройство, называемое арифметико-логическим устройством (АЛУ). Функция, вычисляемая АЛУ, управляет инструкциями, извлеченными из памяти. После того, как данные были обработаны АЛУ, их можно либо записать обратно в регистровый файл, либо сохранить в памяти.

Одной из важных характеристик производительности почти каждого современного ЦП является ускоритель доступа к памяти, называемый кэшем. Кэши — это небольшие, быстрые запоминающие устройства, которые хранят копии данных и фрагменты инструкций, которые, вероятно, будут использоваться в ближайшем будущем ядром ЦП. Кэши медленнее, чем файлы регистров, но быстрее, чем основная память; аналогично, кэши хранят больше данных, чем файл регистров, но хранят меньше данных, чем основная память.

Важной особенностью кэша процессора Xbox, о которой следует знать, является то, что это кэш обратной записи. Кэши обратной записи позволяют копиям данных, хранящимся внутри процессора, не синхронизироваться с тем, что находится в основной памяти. Эта разница во времени может усложнить попытки отследить выполнение процессора, наблюдая только за трафиком внешней памяти. Кэш-память также может использоваться процедурами безопасности для скрытия промежуточных результатов вычислений от кого-либо, наблюдающего зашиной памяти.

## Северные и южные мосты

Термины «северный мост» и «южный мост» являются жаргонными, специфичными для архитектуры ПК. Они относятся к двум основным вспомогательным микросхемам, которые есть практически в каждом ПК. Микросхема северного моста соединяет ЦП с основной памятью, а также с любыми высокопроизводительными шинами расширения, такими как AGP и PCI. Микросхема южного моста крепится к микросхеме северного моста и содержит все дополнительные периферийные устройства, которые есть в типичном ПК — параллельный, последовательный, USB, мышь, клавиатуру, контроллеры IDE, аудиокодеки и многое другое. Разделение архитектуры ПК на эти три основных модуля — ЦП, северный мост и южный мост — позволяет разработчикам ПК смешивать и сопоставлять различные виды архитектур памяти с разнообразным выбором процессоров и периферийных устройств.

Соединение между чипсетами северного и южного мостов различается в зависимости от чипсета. В случае Xbox в качестве соединения между функциональным эквивалентом чипов северного и южного мостов используется высокопроизводительная узкая параллельная шина, называемая HyperTransport. Ширина шины составляет всего 8 бит в каждом из двух направлений, но она тактируется на частоте 200 МГц, и данные выбираются на каждом фронте тактового сигнала, поэтому эффективная пиковая скорость передачи данных составляет 400 Мбайт/с в каждом направлении. Чип северного моста подключен к ЦП через шину, называемую Front Side Bus (FSB). В случае Xbox FSB представляет собой 64-битную шину с частотой 133 МГц, которая использует логические уровни AGTL+.

Знание и понимание типов соединений между чипами имеет решающее значение в обратном проектировании, поскольку тип соединения будет определять, насколько сложно перехватить данные, проходящие между различными компонентами. Подробности относительно простой для прослушивания шины, шины HyperTransport, обсуждаются в Главе 8, «Обратное проектирование безопасности Xbox».

В Xbox южный мост — это чип, разработанный nVidia и называемый MCPX; это производная от nVidia nForce MCP Multimedia and Communications Processor. Чип северного моста также был разработан nVidia и называется NV2A GPU. Чипы северного и южного мостов были произведены TSMC (Taiwan Semiconductor Manufacturing Corporation). NV2A объединяет как графический процессор (Graphics Processing Unit), так и традиционные контроллеры памяти и шины расширения, которые есть в большинстве чипов северного моста. Как объяснялось ранее, объединение графического процессора и северного моста позволяет разработчикам систем объединить графическую память с основной памятью, что приводит к некоторому снижению производительности.

## OЗУ

Материнская плата Xbox использует 64 МБ DDR SDRAM для основной памяти. {этого не хватит, даже чтобы нормально пользоваться WINDOWS XP на простом ПК. А тут игровая консоль...} DDR SDRAM означает Double Data-Rate Synchronous Dynamic Random Access Memory (синхронная динамическая память с произвольным доступом с двойной скоростью передачи данных). Благодаря объединению методов синхронизации и DDR совокупная пропускная способность основной памяти Xbox достигает 6,4 гигабайт/сек.

ОЗУ — это, по сути, таблица информации, индексируемая ЦП. Каждое местоположение в ОЗУ имеет уникальный индексный номер, называемый адресом, и, как следует из названия «произвольный доступ», нет никаких ограничений на порядок доступа к данным в ОЗУ.<sup>8</sup>

Термин «динамический» применяется к оперативной памяти, которая должна постоянно обновляться для сохранения целостности данных. Например, оперативная память, используемая в Xbox, должна считывать и записывать каждое местоположение примерно тридцать раз в секунду. Потеря производительности не так велика, как кажется, поскольку в современные чипы DRAM встроено специальное оборудование, помогающее оптимизировать процесс.

Префикс «синхронный» означает, что внутри DRAM процедура доступа к данным разбита на ряд шагов. Каждый из этих шагов независим и может происходить параллельно, так что несколько запросов данных могут выполняться одновременно. Внешний сигнал синхронизации, известный как часы, используется для синхронизации движения запросов доступа к данным через различные шаги внутри DRAM. В результате запросы доступа к данным проходят через каждый шаг, как вода через трубу, и этот метод также известен как конвейеризация. Синхронные DRAM имеют более высокую пропускную способность, чем их предшественники, поскольку конвейеризация позволяет обрабатывать несколько запросов одновременно. Однако время, необходимое с момента, когда доступ впервые выдается к SDRAM, до момента, когда данные наконец появляются на выходе — задержка доступа — не улучшается конвейеризацией.

Термин «Double Data Rate» относится к способу передачи синхронных данных относительно синхронизирующих часов. Тактовая волна состоит из повторяющегося шаблона высоких и низких сигналов. В традиционных системах данные передаются только при переходе от низкого к высокому уровню тактовой волны. В системе DDR данные передаются как при переходе

---

<sup>8</sup>На самом деле, SDRAM могут иметь несколько ограничений на шаблоны доступа к памяти (например, режимы страниц и пакетные режимы) по соображениям производительности. Название «случайный» призвано отличать RAM от памяти типа First-In, First-Out (FIFO) и Last-In, First-Out (LIFO), где доступ к данным осуществляется с использованием строгого набора правил упорядочивания.

от низкого к высокому, так и при переходе от высокого к низкому уровню. Таким образом, при той же тактовой частоте можно передать вдвое больше данных. Мнемоника производительности, указанная поставщиками DDR SDRAM, такими как DDR266, относится к скорости передачи, поэтому фактическая тактовая частота составляет половину мнемоники производительности, или 133 МГц в этом случае.

## ПЗУ

Каждому компьютеру необходимо иметь некую постоянную или энергонезависимую память для хранения программы запуска или загрузки. DDR SDRAM, обсуждавшаяся выше, не подходит для этого приложения, поскольку все данные в DDR SDRAM теряются при отключении питания. Текущие версии Xbox используют FLASH ROM вместо этого для хранения данных, которые должны сохраняться даже при отключении питания. ROM означает память только для чтения, а FLASH относится к определенному стилю элемента хранения, который можно перепрограммировать электронным способом. Память типа FLASH удобна в ПК, поскольку конечный пользователь может перепрограммировать ее для исправления ошибок в загрузочном коде. Однако в Xbox программирование FLASH ROM конечным пользователем намеренно отключено. Сигнал записи, необходимый для программирования, отключается путем опускания перемычки, расположенной на задней стороне материнской платы Xbox в месте расположения компонента R7R4 (для получения дополнительной информации см. боковую панель под названием «Включение аппаратного обеспечения программирования FLASH ROM»). В случае Xbox перепрограммируемость FLASH в первую очередь используется как удобство для Microsoft во время разработки и производства. Вполне вероятно, что через несколько месяцев Xbox будет

### Включение программирования FLASH ROM

#### Аппаратное обеспечение

Исправление сигнала, который был отключен Microsoft для предотвращения внутрисистемного программирования FLASH ROM, является довольно простой процедурой. Сигнал записи FLASH ROM был отключен путем исключения одного резистора, номер компонента R7R4, расположенного на нижней стороне материнской платы Xbox в секторе 7R. Вы можете припаять кусок провода между двумя серебряными контактами резистора или даже просто соединить контактные площадки большим количеством припоя. Обратите внимание, что даже несмотря на то, что программирование FLASH ROM включено в аппаратном обеспечении этим исправлением, у вас все еще нет программы, которая фактически выполняет перепрограммирование. Запуск такой программы является гораздо более сложной задачей из-за криптографической системы безопасности программного обеспечения, внедренной Microsoft.

использовать более дешевые жестко запитые «маскирующие ПЗУ», как только Microsoft посчитает, что она готова записать свою загрузочную программу и ядро в камень (или кремний, в зависимости от обстоятельств).

Загрузочное ПЗУ играет ключевую роль в обратном проектировании любого компьютера, поскольку оно содержит критический код, который отвечает за инициализацию всей системы. В случае Xbox загрузочное FLASH ROM играет еще более важную роль, поскольку оно частично отвечает за реализацию жесткой системы безопасности программного обеспечения. Точная роль FLASH ROM в системе безопасности будет объяснена позже, но сейчас важно помнить, что FLASH ROM управляет инициализацией оборудования в Xbox, а также содержит начальный образ ядра операционной системы.

## Всякая всячина

Xbox оснащен небольшим 8-битным сопроцессором, который называется System Management Controller (SMC). SMC — это полноценный миниатюрный компьютер с RAM, ROM и процессором в одном корпусе. Процессор внутри SMC использует архитектуру PIC (Peripheral Interface Controller), изначально разработанную в Гарвардском университете около 1975 года и адаптированную General Instruments для коммерческой продажи. Arizona Microchip Technology (теперь называется Microchip Technology, [www.microchip.com](http://www.microchip.com)) приобрела линейку продуктов PIC в 1985 году и продает ее с тех пор. SMC можно найти в секторе 7B на Xbox, а его обозначение — U7B2. SMC контролирует кнопку питания на передней панели Xbox, поэтому SMC должен работать даже при выключенном ЦП. В результате блок питания Xbox имеет слаботочную линию питания 3,3 В «резервную» линию, которая всегда активна, когда Xbox подключен. SMC также отвечает за управление индикаторами вокруг кнопки питания на Xbox, а также управляет механизмом извлечения DVD. Наконец, SMC имеет функцию, которая контролирует работоспособность ЦП и перезагружает ЦП в случае его сбоя. Функция мониторинга SMC должна быть отключена, если вы хотите запустить собственную операционную систему на Xbox. SMC взаимодействует с ЦП через MCPX через 1-битный последовательный интерфейс, известный как I2C.

Еще одной важной особенностью Xbox является порт отладки LPC. Порт отладки LPC представляет собой 4-битную шину, работающую на частоте 33 МГц. LPC означает «Low Pin Count» (низкое количество выводов), и изначально он был разработан как метод подключения большого количества медленных устаревших устройств, таких как клавиатуры, последовательные порты, параллельные порты и загрузочные ПЗУ, к чипу южного моста через простую промежуточную микросхему трансляции. Порт отладки предоставляется на Xbox предположительно для целей производственных испытаний подрядчиком по оборудованию Microsoft. Когда Xbox приближается к финальным стадиям производства, порт отладки LPC используется для загрузки программы загрузки, которая выполняет тесты, диагностику и прогрев на материнской плате Xbox. Порт отладки LPC более подробно обсуждается в главе 11, но сейчас важно знать, что можно заставить Xbox считать свой начальный образ загрузочного ПЗУ через порт отладки LPC, подключив устройство ПЗУ, совместимое с LPC, и закоротив один из контактов данных (D0) на FLASH ROM на определенное напряжение (ноль вольт). Это, пожалуй,

самый простой способ заставить Xbox загрузить ваш собственный код — при условии, что вы знаете, как обойти секретный загрузочный код, который запирает Xbox.

## Сопоставление с образцом

Теперь, когда мы знакомы с архитектурой Xbox, у нас есть точка отсчета для, возможно, одного из самых мощных инструментов обратного проектирования — сопоставления с образцом. Возможность делать обоснованные предположения о функции различных деталей, просто наблюдая за их соединением, размещением и формой — это первый шаг к тому, чтобы стать крутым обратным проектировщиком. Чтобы продемонстрировать силу сопоставления с образцом, мы сравним материнскую плату Xbox с материнской платой ПК и материнской платой Nintendo Gamecube.

Изучение множества шаблонов — лучший способ стать хорошим сопоставителем шаблонов. Я разбираю каждое купленное мной оборудование и изучаю печатные платы, чтобы попытаться узнать, что знают другие дизайнеры, «читая» печатную плату. Каждая печатная плата рассказывает историю о процессе ее разработки; редко встретишь необычную функцию схемы, которая не имеет какой-то определенной цели.

### Осторожность



При разборке любого электронного оборудования обязательно сначала отключите его от сети и подождите минуту, чтобы разрядился заряд на больших конденсаторах в блоке питания. Также обязательно используйте соответствующие меры контроля статического электричества, описанные в главе 1!

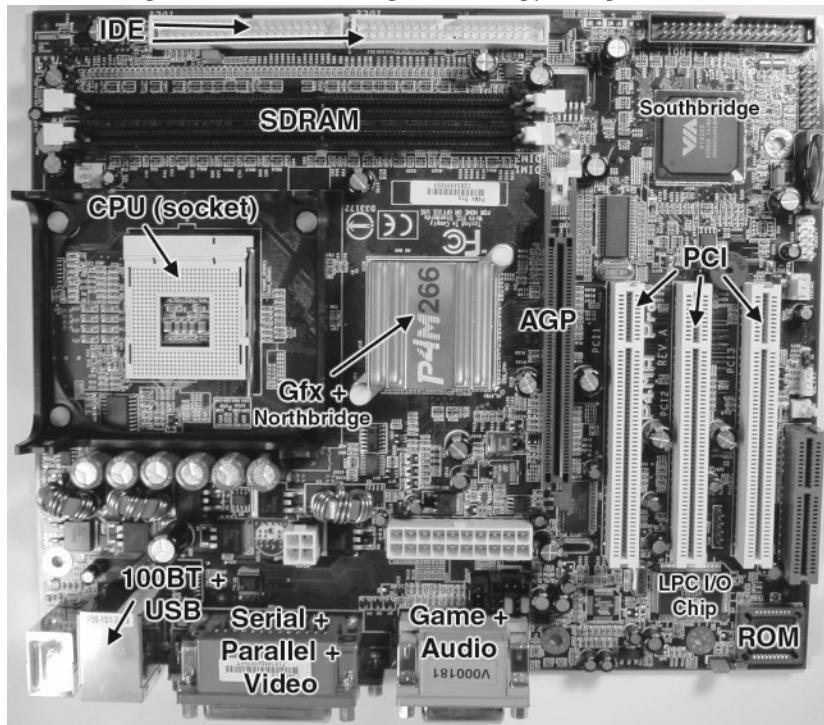
## Сравнение: Xbox против ПК

Сходство Xbox с ПК — это благо для хакеров, поскольку платформа ПК очень хорошо документирована. Каждая часть Xbox имеет аналог в типичном ПК, поэтому практически на любой вопрос высокого уровня можно ответить, просто прочитав о похожей части ПК. Таким образом, стоит более подробно рассмотреть сходства между материнской платой Xbox и стандартной материнской платой ПК. Еще одним преимуществом является то, что большая часть информации в этой книге будет применяться непосредственно к ПК, поэтому вы сможете легко применить то, чему научитесь, взламывая Xbox, к большому количеству ситуаций.

Ближайшими родственниками Xbox являются системы на базе чипсетов, использующих унифицированную архитектуру памяти, такие как nVidia nForce или Via

ProSavageDDR от Technology. Архитектурная схема, представленная в предыдущем разделе, была получена путем прочтения опубликованных спецификаций Xbox и материалов, доступных на веб-сайте nVidia о чипсете nForce. В этом разделе мы сравним Xbox с материнской платой P4M266 на базе Via Technology ProSavageDDR. Xbox сравнивается здесь с материнской платой на чипсете не от nVidia, чтобы подчеркнуть большое сходство Xbox с ПК.

Рисунок 2-7 показывает изображение материнской платы ПК, Via P4M266. Несмотря на то, что чипсет произведен другим производителем, сходство



**Рисунок 2-7:** Через материнскую плату P4M266 с интегрированной графикой.

между P4M266 и Xbox поразительны. Почти весь материал, рассмотренный в предыдущем разделе, применим к этой материнской плате ПК. Основные отличия — несколько разных портов и разъемов, а также наличие высокопроизводительных портов расширения PCI и AGP. У Via P4M266 также отсутствует явный разъем отладки LPC, поскольку все устаревшие периферийные устройства напрямую реализованы микросхемой LPC multi-I/O.

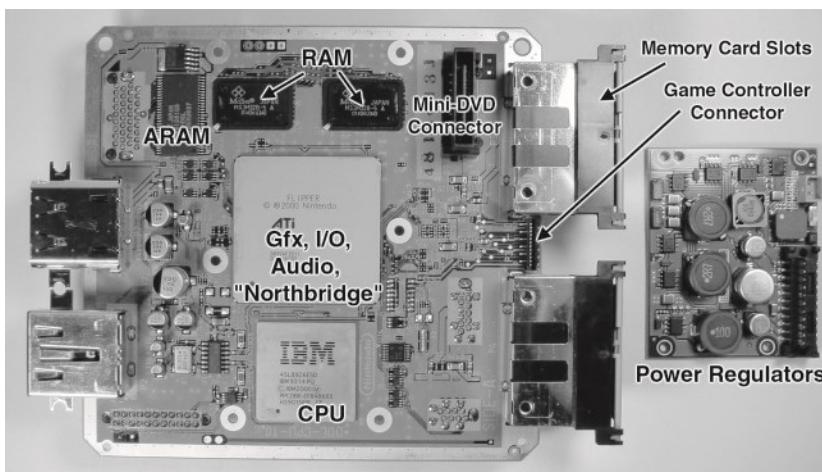
## Контраст: Xbox против Gamecube

Nintendo Gamecube интересна в сравнении с Xbox. Gamecube — это машина, разработанная для той же цели, что и Xbox — игр, — но с совершенно другой

философией дизайна. Xbox и Gamecube используют одну и ту же общую архитектуру — центральный процессор, графический сопроцессор, немного памяти и несколько вспомогательных микросхем, — но на этом сходства заканчиваются. Дизайн Gamecube демонстрирует пристальное внимание к деталям и стоимости. Материнская плата Gamecube небольшая и простая, количество компонентов сведено к минимуму, а конструкция теплоотвода и охлаждения очень проста. Чистая, прямая компоновка большинства дорожек печатной платы на материнской плате Gamecube отражает тот факт, что почти каждая микросхема специально разработана для Gamecube. В результате Gamecube является гораздо более экономичной платформой для сборки, чем Xbox.

Можно распознать общую организацию Gamecube, выведя функцию каждого чипа из базовой маркетинговой информации, которую предоставляет Nintendo. Дополнительные подробности об архитектуре Gamecube трудно вывести, поскольку он использует так много специальных компонентов, которые не имеют аналогов в стандартном ПК. По схеме дорожек на материнской плате можно было бы предположить, что большой чип в центре платы, чип «Flipper», является эквивалентом интегрированного графического чипа северного моста в ПК. Это почти верно. Ключевое отличие заключается в том, что даже несмотря на то, что чип Flipper объединяет и контроллер памяти, и графический контроллер в одном корпусе, графическая функция по-прежнему имеет свою собственную выделенную память, встроенную в тот же чип. Такой тип организации позволяет графическому движку использовать очень высокопроизводительную память, при этом компромисс памяти немного меньше, чем при использовании внешней памяти. Меньший размер встроенной памяти частично компенсируется использованием чрезвычайно быстрой внешней памяти.

Gamecube не использует DDR SDRAM, как Xbox; вместо этого он использует то, что называется 1-T SRAM. 1-T SRAM — это память DRAM, которая эмулирует очень быстрый тип памяти, известный как статическая оперативная память (SRAM). У SRAM гораздо меньшие задержки случайного доступа, чем у DRAM, и им также не требуется, чтобы каждая ячейка памяти обновлялась 30 раз в секунду, как это делает DRAM. Фактическая магия того, как DRAM может маскироваться под быструю SRAM, довольно сложна и выходит за рамки этой книги, но вы можете найти больше информации на веб-сайте производителя 1-T SRAM, [www.mosys.com](http://www.mosys.com).



**Рисунок 2-8:** Материнская плата Gamecube плюс плата регулятора питания. Материнская плата примерно в два раза меньше материнской платы Xbox.

Gamecube также имеет еще один фрагмент памяти, известный как ARAM, который медленнее, чем память 1-T SRAM, и используется для хранения таких вещей, как аудиосэмплы, которые не требуют доступа с высокой пропускной способностью. Наличие разрозненной архитектуры памяти означает, что Gamecube может выжимать более постоянный объем производительности из каждой подсистемы, что важно для сведения задержки кадров к минимуму. Однако компромисс заключается в том, что Gamecube может быть сложнее программировать, а неправильное управление несколькими фрагментами памяти может привести к проблемам с производительностью.

Еще одно важное различие между Gamecube и Xbox заключается в том, что Gamecube потребляет гораздо меньше энергии, чем Xbox. Потребление энергии может показаться неважным на первый взгляд, поскольку обе консоли предназначены для подключения к розетке, но более низкая мощность Gamecube требует меньшего количества теплопередающих компонентов и меньших блоков питания, что позволяет экономить на стоимости. На рисунке 2-8 для справки приведено изображение регулятора мощности Gamecube; регулятор мощности занимает часть объема блока питания Xbox плюс локальные импульсные регуляторы на материнской плате Xbox.

Справедливо ради, отметим, что Gamecube имеет небольшой внешний преобразователь переменного тока в постоянный, в то время как Xbox подключает консоль к сети напрямую. Кроме того, электронные компоненты деградируют гораздо быстрее при повышенных температурах, как описано в правиле Аррениуса. Например, повышение рабочей температуры на 10 градусов Цельсия примерно удваивает частоту отказов компонента. Таким образом, Gamecube должен быть более надежным с годами, чем Xbox, поскольку

Gamecube выделяет меньше тепла, а его система терморегулирования так же хороша, если не лучше, чем у Xbox.

Наконец, интересно отметить, что Gamecube везде использует фирменные интерфейсы ввода-вывода. Формат игрового диска — это формат mini-DVD, а DVD-ридер подключается к материнской плате через фирменный разъем. Использование меньшего DVD-носителя позволяет Nintendo сократить задержку поиска данных, что означает более короткое время загрузки игр. Игровые контроллеры и карты памяти также используют фирменный формат сигналов. Все в Gamecube чем-то похоже на наш знакомый ПК, но ничто не было напрямую включено в конструкцию без изменений.

Помимо оптимизации технологичности и стоимости Gamecube, использование в основном фирменных чипов и стандартов делает консоль гораздо более сложной для обратного проектирования, чем Xbox. Например, обратите внимание, что на рисунке 2-8 в Gamecube нет очевидного чипа ПЗУ. Таким образом, чтобы даже начать смотреть на код Gamecube, нужно выследить и извлечь ПЗУ, спрятанное где-то в одном из чипов на материнской плате! Это один из редких случаев, когда безопасность через неизвестность работает. Даже если бы на Gamecube вообще не было никакой безопасности, стоимость и усилия, затраченные на попытку записать свой собственный код на пользовательский формат DVD Nintendo, просто не стоят того для отдельного энтузиаста.

# CHAPTER 3 Установка синего светодиода

Теперь, когда вы сняли крышку с вашего Xbox, пришло время сделать несколько стартовых проектов. В следующих главах вы познакомитесь с некоторыми элементарными модификациями и ремонтами, которые вы можете выполнить на вашем Xbox. Эти проекты разработаны и написаны для читателей, у которых мало или совсем нет опыта в аппаратном взломе. Более продвинутые темы по Xbox можно найти в последующих главах.

В этой главе вы узнаете, как заменить обычно зеленый светодиод на передней панели Xbox на синий светодиод. Этот проект требует минимальной пайки; большая часть усилий направлена на снятие передней панели и сборки светодиодной цепи. Давайте начнем!

## Примечание



В стандартной Xbox используется комбинированный светодиод зеленого/красного цвета, но красный светодиод используется только для индикации состояний ошибки. Процедура, описанная в этой главе, преобразует индикатор передней панели Xbox в синий светодиод. Правильной заменой будет комбинированный светодиод синего/красного цвета (T-1, линза диаметром 3 мм). Однако их трудно найти, поэтому в инструкциях они не используются. Инструкции дадут вам необходимые базовые знания, чтобы вы могли импровизировать и внедрить собственное решение светодиода, если у вас есть к этому склонность.

## Что вам понадобится

Ниже приведен список оборудования, которое вам понадобится для выполнения этого проекта:

- Маломощный паяльник с тонким наконечником
- Припой
- Очиститель флюса и жала паяльника (оциально)
- Маленькая плоская отвертка
- Отвертка Torx с битой T-10
- Два низковольтных (3 вольта) синих светодиода в корпусе T-1 (3 мм)
- Маскирующая лента для фиксации деталей во время пайки

Вы можете заменить светодиод любого цвета, который вам нравится, но он должен включаться при напряжении около 3 вольт и иметь корпус в стиле T-1. Будьте внимательны при покупке светодиодов, потому что многие синие и белые светодиоды рассчитаны на работу только при 5 вольтах. Светодиод, используемый здесь, — это Lumex SSL-LX3044USBC, который вы можете купить через Digi-Key ([www.digikey.com](http://www.digikey.com)); номер детали 671747-ND. (Для экономных Digi-Key может отправить светодиоды почтой США первого класса. Обратите внимание, что для заказов через Digi-Key стоимостью менее 25 долларов взимается сбор в размере 5 долларов.)

Если ограничение минимального заказа Digi-Key является для вас проблемой, у Mouser Electronics ([www.mouser.com](http://www.mouser.com)) также есть линейка синих светодиодов, и у них нет минимального заказа. Примером может служить синий светодиод Kingbright в корпусе T-1 с прозрачной линзой; складской номер Mouser — 604L7104PBC/H для более яркой версии с немного более высоким напряжением или 604L7104QB/D для версии, которая работает при более низком напряжении, но с более низкой номинальной яркостью.

Вы также можете использовать двухцветный светодиод, если хотите сохранить функциональность светодиода состояния ошибки. Для Xbox требуется двухцветный светодиод с общим катодом в корпусе T-1 с тремя выводами. К сожалению, двухцветные светодиоды с синим элементом в меньшем корпусе T-1 очень трудно найти.

## Снятие передней панели Xbox

Передняя панель Xbox представляет собой формованный кусок АБС-пластика, который крепится на месте четырьмя винтами Torx T-10 и тремя формованными фрикционными замками. Электроника на передней панели подключается к материнской плате Xbox через один девятивыводной разъем, который проходит через отверстие в металлическом экране электромагнитных помех.

Откройте Xbox, как указано в Главе 1. Поднимите и переместите жесткий диск и DVD-привод вверх и к задней части коробки ровно настолько, чтобы открыть передний край материнской платы Xbox. Вам не придется



**Рисунок 3-1:** Расположите дисководы так, чтобы был виден передний край материнской платы. Отсоедините все кабели дисководов. На рисунке 3-1 показано, как должен выглядеть ваш Xbox после выполнения этих шагов.

#### Примечание



Старые модели Xbox будут иметь вертикально установленную печатную плату около передней части Xbox. Эту печатную плату можно снять, взявшись за плату и вытащив ее из гнезда. Вы можете обнаружить, что снятие вертикально установленной печатной платы полезно при попытке освободить средний фрикционный фиксатор передней панели. Не забудьте вернуть печатную плату на место, когда закончите!

Открутите четыре винта, которые удерживают переднюю панель на месте (см. рисунок 3-2).

Отсоедините разъем провода передней панели от материнской платы Xbox, как показано на рисунке 3-3. Для этого разъема достаточно сильного, равномерного усилия. (Не дергайте разъем, так как вы можете повредить провода.)



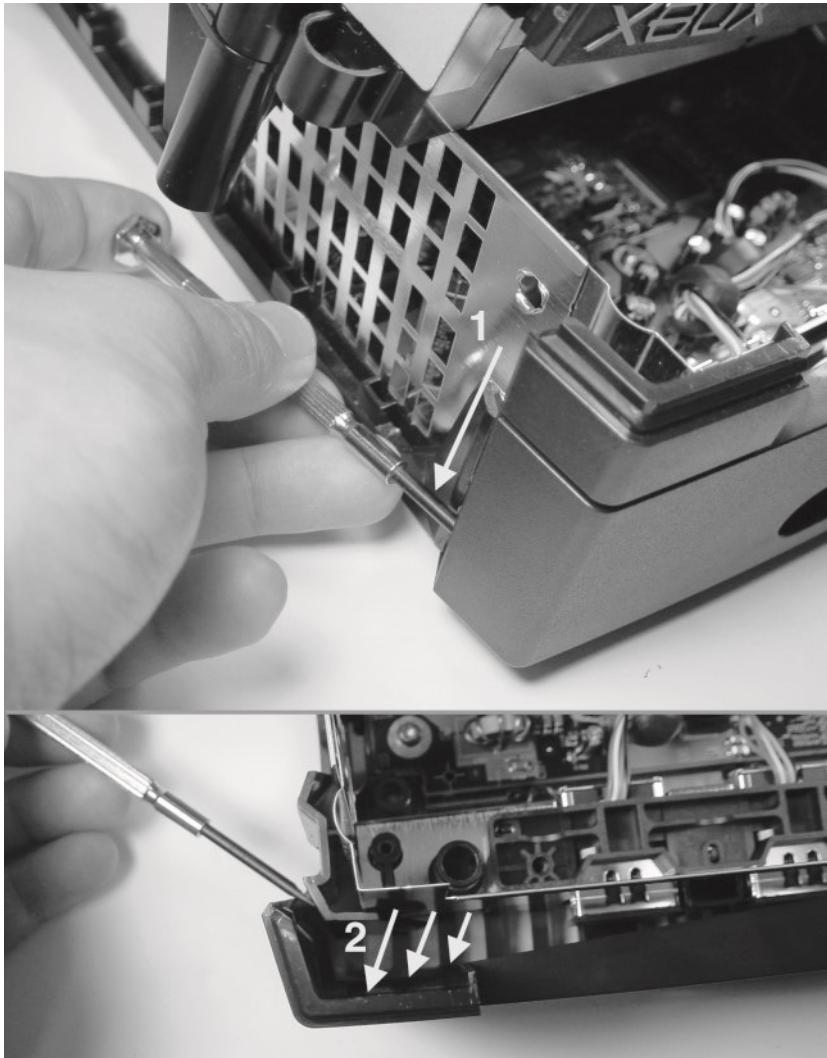
Рисунок 3-2: Расположение четырех крепежных винтов на передней панели.

Теперь о самом сложном: фрикционных замков. Фрикционный замок — это крючок из пластика, который удерживает детали вместе. Крючок имеет такую форму, что его легко вставить, но трудно вытащить. Для освобождения фрикционного замка обычно требуется согнуть или надавить на пластик.



Рисунок 3-3: Отсоедините разъем провода передней панели от материнской платы Xbox.

Три фрикционных замка удерживают переднюю панель на месте: один по краям передней панели и один посередине, протыкающий металлический экран электромагнитных помех. Сначала ослабьте фрикционные замки по краю с помощью тонкой плоской отвертки, как показано на рисунке 3-4. Эти замки очень тугие, и вам, возможно, придется ослаблять их по частям, начиная с верхней части. Вставьте кончик отвертки в пространство вдоль стороны между панелью и основным корпусом и подденьте, пока не почувствуете легкую податливость. Уберите отвертку и повторите процесс около нижней части корпуса. Возможно, вам придется попробовать несколько раз, прежде чем замок будет



**Рисунок 3-4:** Ослабьте фиксаторы кромок с помощью плоской отвертки.

(1) Начните работать сверху и двигайтесь вниз; (2) как только панель освободится, она должна отогнуться наружу от корпуса.

свободно. Не прикладывайте излишних усилий к корпусу, так как вы можете треснуть или поцарапать пластик. Когда край передней панели освободится, вы сможете отогнуть его от корпуса. Повторите этот процесс для обоих краев.

Как только оба края будут свободны, потяните вверх средний фрикционный фиксатор (показан на рисунке 3-5), и передняя панель должна сняться.

После того как передняя панель будет освобождена, проденьте соединитель провода передней панели через отверстие в металлическом экране электромагнитных помех и положите панель на стол внешней стороной вниз.

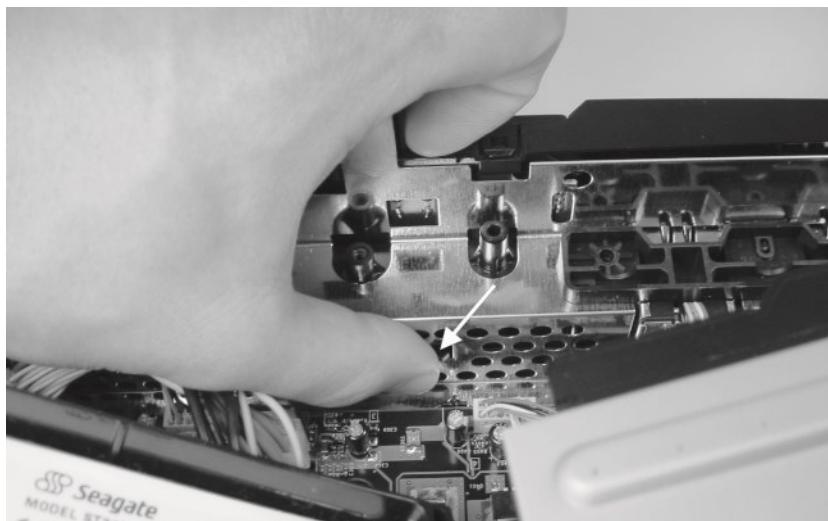
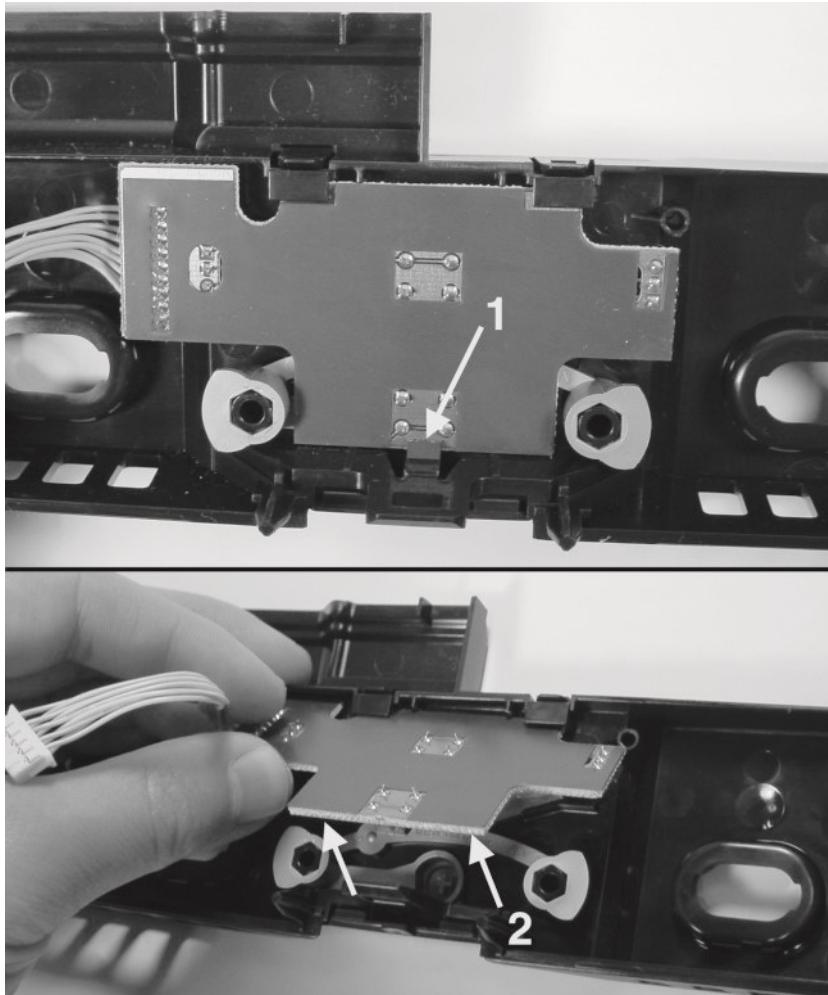


Рисунок 3-5: Большой палец нажимает на средний фрикционный фиксатор.

## Снятие платы передней панели

Передняя панель Xbox содержит небольшую печатную плату, удерживаемую на месте одним фрикционным фиксатором, как показано на рисунке 3-6. Пальцем или отверткой нажмите на фрикционный фиксатор и вытащите переднюю панель из гнезда.

Положите сборку печатной платы на стол зеленой стороной вниз. Вы должны увидеть два прозрачных светодиода и два плоских кнопочных переключателя.



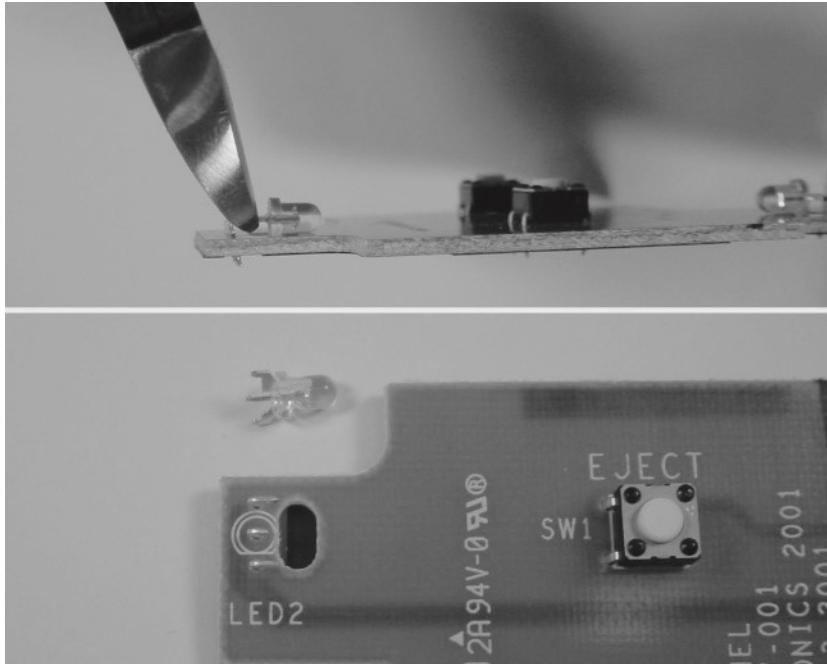
**Рисунок 3-6:** Поднимите печатную плату из передней панели. (1) Нажмите на фиксатор фрикционного замка, при необходимости используя отвертку, (2) затем поднимите сборку из передней панели.

## Установка синего светодиода

Теперь, когда плата удалена, пришло время установить синий светодиод. Это потребует некоторой пайки, поэтому включите паяльник и дайте ему нагреться. Помните, что важно использовать паяльник с тонким жалом и использовать очиститель для жала паяльника, чтобы подготовить жало перед его использованием. См. Приложение А, «Где взять хакерское

снаряжение», если вам нужно что-либо из этого; вы можете оснастить себя за немного большую сумму, чем стоимость видеонигры.

Удалите оба существующих светодиода с помощью кусачек для резки заподлицо. Сохраните как можно больше металлических ножек, выходящих из светодиодов, поскольку позже вы будете использовать их для крепления синего светодиода. На рисунке 3-7 показано, как должна выглядеть печатная плата, когда вы закончите.



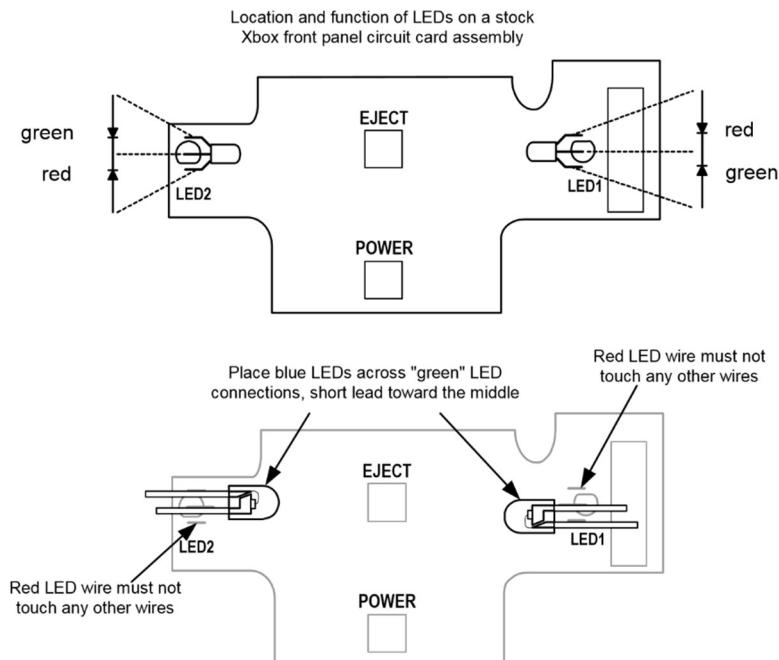
**Рисунок 3-7:** Отрежьте имеющиеся светодиоды от узла платы. (Вверху) отрежьте светодиод как можно ближе к корпусу; (внизу) светодиод удален. Используйте эту процедуру для удаления обоих светодиодов.

Чтобы облегчить пайку, приклейте плату к плоской поверхности с помощью куска клейкой ленты, чтобы она не двигалась. Расположите синие светодиоды так, чтобы их ножки касались металлических штырьков старых светодиодов на плате. (На рисунке 3-8 показано, как определить полярность светодиода и его правильную ориентацию на плате. См. врезку «Анатомия светодиода», если вы не уверены в том, как определить полярность светодиода.)

Приклейте линзовую часть светодиодов на место с помощью клейкой ленты, чтобы они не скатывались, пока вы их припаиваете. Вам нужно будет согнуть или отрезать ножки светодиода, который будет установлен с правой стороны платы, поскольку желтый проводной разъем будет мешать.

**Предупреждение**

Обратите особое внимание на полярность светодиода. Если вы установите светодиод наоборот, свет не будет излучаться. См. боковую панель «Анатомия светодиода», если вы не уверены, как определить полярность светодиода.



**Рисунок 3-8:** Размещение синих светодиодов на плате передней панели. Обратите внимание на асимметрию цветов светодиодов на каждой стороне платы.

Рисунок 3-8 также иллюстрирует полярность и функцию светодиодов Xbox. Любителям приключений предлагается импровизировать и устанавливать несколько светодиодов или светодиодные пакеты поверхностного монтажа, чтобы попытаться получить большие цвета и функциональности. Можно установить светодиоды, которые немного больше, чем пакет T-1, используемый в Xbox, предварительно отшлифовав края светодиода.

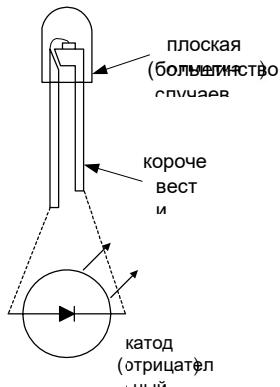
После того, как вы дважды проверили полярность светодиодов и убедились, что короткий вывод на обоих светодиодах примыкает к остаткам центрального вывода исходного светодиода, припаяйте светодиоды на место. На рисунке 3-10 показано, как светодиоды припаиваются на место. (Если вы никогда раньше не паяли, вам может быть полезно прочитать Приложение В, «Методы пайки», прежде чем продолжить.)

Перед использованием паяльника расплавьте немного припоя, чтобы убедиться, что наконечник достаточно горячий. Припой должен расплавиться мгновенно, если наконечник достаточно горячий. Если паяльник слишком холодный, вы не сможете сформировать хорошее соединение и рискуете повредить печатную плату.

Держите горячий наконечник паяльника напротив вывода синего светодиода и вставьте вывод в металлический пытры на плате. Пока вывод нагревается, нанесите немного припоя на точку, где вывод синего светодиода встречается с металлическим пытрыем. Поверхностное натяжение расплавленного припоя должно вызвать

## Анатомия светодиода

Светодиоды, или светоизлучающие диоды, являются поляризованными устройствами, которые пропускают ток только в одном направлении. Это означает, что они не будут работать, если их подключить наоборот. На рисунке 3-9 показана анатомия светодиода. Более короткий вывод на стороне корпуса светодиода с небольшой плоской частью называется катодом. Катод должен быть подключен к потенциалу, более отрицательному, чем другой вывод, анод, для того, чтобы светодиод функционировал.



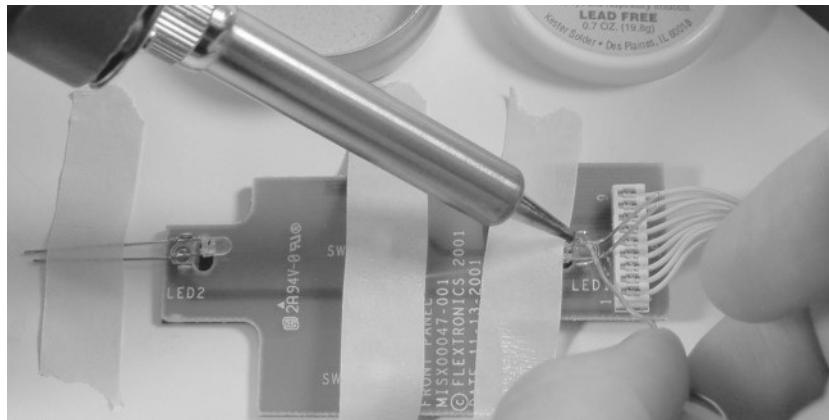
**Рисунок 3-9:** Анатомия светодиода.

Различные светодиоды обычно требуют разного количества прямого напряжения для включения. Красные светодиоды обычно требуют 1,7 вольта, зеленые светодиоды требуют около 2,1 вольта, а синие светодиоды требуют 3,5 вольта и выше. Ранние синие светодиоды требовали почти 5 вольт прямого напряжения, но достижения в области технологий снизили их напряжение, что упростило их интеграцию в электронику с питанием от батареек и низким напряжением. При покупке светодиода для этого проекта помните о требуемом прямом напряжении. Если вы установите синий светодиод на 5 вольт, его световой поток будет очень тусклым, поскольку максимальное прямое напряжение, генерируемое драйверами Xbox, составляет около 3 вольт.

припой, чтобы смочить выводы синего светодиода и металлический штырь на плате. Если этого не произошло, уберите паяльник и нанесите немного флюса на соединение, и попробуйте снова.

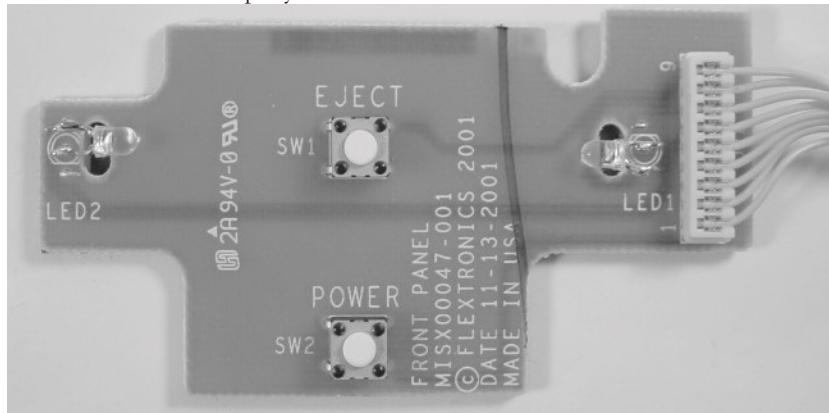
Не держите жало паяльника у металлического стержня в течение длительного времени, иначе припой, удерживающий стержень на месте, расплавится. Вы поймете, что паяное соединение стержня с платой расплавилось, когда стержень начнет свободно качаться. Если это произошло, прижмите жало паяльника к плате и медленно потяните жало паяльника в сторону.

Потягивание жала предотвратит вытягивание стержня из платы вместе с паяльником. Подождите, пока стержень остынет, и согните его обратно в нужное положение.



**Рисунок 3-10:** Припаиваем светодиоды на место. Обратите внимание, как плата и светодиоды приклеены на место с помощью клейкой ленты.

После того, как вы припаяли все четыре соединения на двух светодиодах, обрежьте лишние выводы светодиодов как можно короче, не обрезая металлические заглушки. Готовая плата должна выглядеть примерно так, как показано на рисунке 3-11.



**Рисунок 3-11:** Готовая сборка платы.

## Сборка передней панели

Запечелкните узел печатной платы обратно на переднюю панель, выровняв верхние края под удерживающими зажимами и вставив плату во фрикционный замок.

Теперь возьмите всю переднюю панель и соедините ее с Xbox. Сначала пропустите соединительный провод через оригинальное овальное отверстие через экран электромагнитных помех. Затем вставьте переднюю панель в Xbox, и все три фрикционных замка должны запцелкнуться на месте.

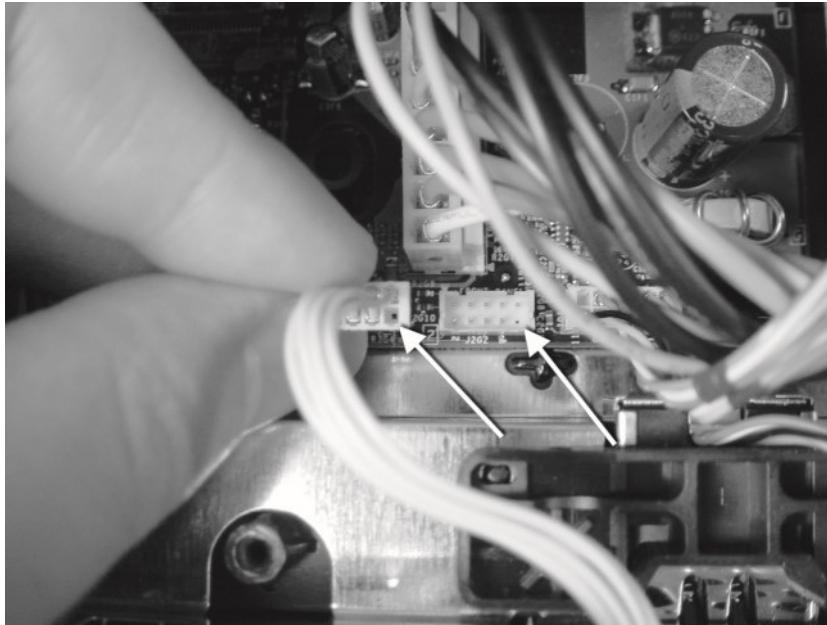
Снова подсоедините разъем провода передней панели к материнской плате Xbox. Разъем имеет такую форму, что его можно вставить только в одном направлении, но пластик, используемый в формовке разъема, мягкий, и вы можете вставить его в обратном направлении, если надавите достаточно сильно, что может привести к его необратимому повреждению. Проверьте правильность ориентации, совместив отсутствующий штырь на гнезде платы с отсутствующим проводом на заголовке разъема, как показано на рисунке 3-12.

Если у вас старая модель Xbox, снова прикрепите вертикальную сборку печатной платы, вставив ее в гнезда. Эта печатная плата является интерфейсной картой для игровых контроллеров Xbox, поэтому вам определенно нужно, чтобы она была установлена правильно.

Теперь вы готовы протестировать недавно модифицированную сборку передней панели.

Верните на место дисководы в отсеки и проверьте надежность подключения кабеля питания и шлейфа. Подключите Xbox, и вы увидите голубое свечение, исходящее от передней панели Xbox. (Если вы не видите того, что ожидаете, не паникуйте. В следующем разделе «Отладка»лагаются решения некоторых возможных проблем.)

Когда вы будете удовлетворены модификацией вашего Xbox, выключите Xbox и вкрутите четыре фиксирующих винта в переднюю панель. (Вам нужно будет снова снять дисководы, чтобы получить доступ к отверстиям для винтов.) Замените дисководы, включите Xbox еще раз, чтобы убедиться, что все работает нормально, а затем соберите остальную часть Xbox, как описано в Главе 1.



**Рисунок 3-12:** Особенности ориентации разъема провода передней панели. Стрелки указывают положение пустого поляризационного штифта.

## Отладка

Иногда что-то идет не так. По моему опыту, чаще всего что-то идет не так, чем нет. Самое главное, что нужно помнить, если что-то не работает, — не паниковать! Сохраняйте рассудок, наблюдайте за тем, что идет не так, и пытайтесь предположить причину неисправности. Для справки, Таблица 3-1 содержит список распространенных проблем и их возможных причин. Приложение Е, «Отладка: подсказки и советы», содержит более подробное обсуждение методов отладки.

Проблема	Возможная причина
Xbox не включается.	<ul style="list-style-type: none"><li>• Разъем провода передней панели вставлен неправильно.</li><li>• Xbox не подключен.</li><li>• Поврежден разъем провода передней панели или поврежден узел платы передней панели.</li></ul>

Xbox включается и работает normally, но светодиоды не светятся или горят только половина светового круга вокруг кнопки извлечения.	<ul style="list-style-type: none"><li>• Один или несколько светодиодов установлены наоборот.</li><li>• Один или несколько светодиодов установлены на металлических штырьках красного светодиода вместо металлических штырьков зеленого светодиода. Проверьте это, включив Xbox без подключенного видеокабеля. Это заставит Xbox отправить мигающий сигнал как на красный, так и на зеленый светодиоды.</li></ul>
Xbox включается и выполняет начальную анимацию, но консоль сообщает, что ей требуется обслуживание.	<ul style="list-style-type: none"><li>• Разъемы жесткого диска или DVD отсоединились. Убедитесь, что серый ленточный кабельный разъем полностью вставлен в каждый диск и что разъем питания для каждого диска полностью вставлен.</li><li>• На старых моделях Xbox проверьте, правильно ли установлена вертикальная интерфейсная плата игрового контроллера.</li></ul>
Xbox включается, но игровые контроллеры не реагируют.	<ul style="list-style-type: none"><li>• На старых моделях Xbox проверьте, правильно ли переустановлена вертикальная интерфейсная плата игрового контроллера.</li></ul>
Xbox включается, но DVD не извлекается.	<ul style="list-style-type: none"><li>• Убедитесь, что соединительный провод между передней панелью и материнской платой Xbox вставлен правильно.</li><li>• Убедитесь, что разъем питания DVD-проигрывателя вставлен правильно.</li></ul>

Таблица 3-1: Руководство по отладке для установки синего светодиода.



# CHAPTER4. Создание USB-адаптера

Создание кабеля является обязательной частью взлома оборудования. Многие модификации и эксперименты, проводимые с частями оборудования, потребуют специального кабеля для адаптации существующих разъемов к тому, что вам нужно.

В этой главе вы узнаете, как сделать USB-кабель-адаптер для Xbox, который не продается в большинстве розничных магазинов (хотя некоторые интернет-продавцы, такие как Lik-Sang ([www.lik-sang.com](http://www.lik-sang.com)), продают этот товар). USB-адаптер позволяет подключать стандартные USB-концентраторы, клавиатуры и мыши к Xbox для запуска Linux. (В этой главе в виде руководства представлены основы создания надежного кабеля; опытные хакеры оборудования могут свободно пролистать или пропустить ее.)

## Необходимые материалы

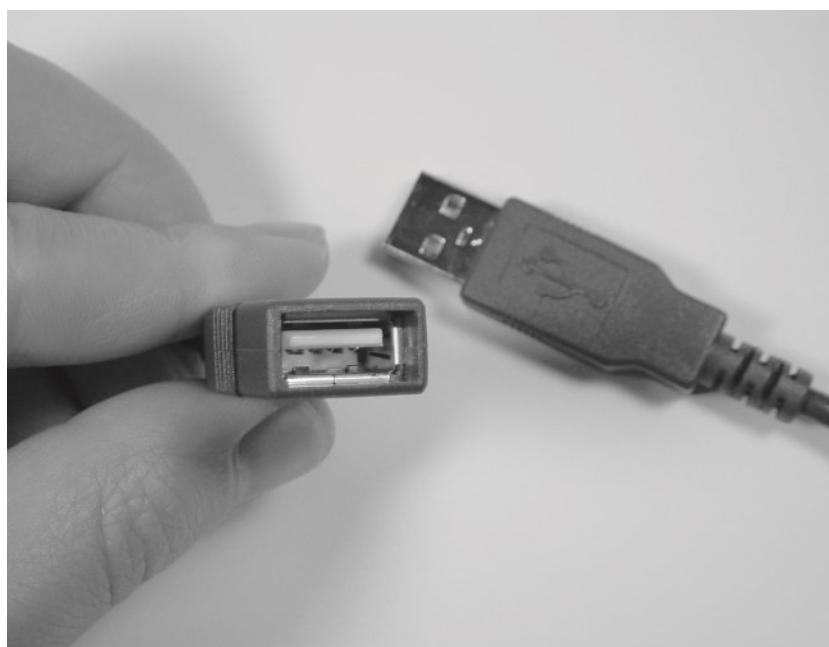
Для этого проекта требуются следующие материалы:

- Один сменный кабель для разрыва игрового контроллера Xbox или удлинительный кабель для игрового контроллера. (См. рис. 4-1.) • Один удлинительный кабель USB типа А или гнездо USB типа А.  
(Изображение гнезда USB типа А показано на рисунке 4-2.)
- Паяльник, припой и флюс.
- Диагональные кусачки и инструмент для зачистки проводов.
- Изолента.
- (Необязательно) Термоусадочная трубка 3/8" и горячий клей.
- (Необязательно) Помощь при пайке третьим лицом.

Пошаговое описание в этой главе использует сменный кабель (доступный в любом магазине видеонгр) и удлинитель USB. Если вы хотите использовать что-то другое, см. Приложение F, «Справочник по оборудованию Xbox», для распиновки различных разъемов, используемых Xbox.



**Рисунок 4-1:** (Слева) удлинительный кабель игрового контроллера Xbox. (Справа) отсоединяемый кабель Xbox.



**Рисунок 4-2:** USB-разъем типа A, гнездо.

## Стратегия

При сборке кабеля-адаптера Xbox USB основная идея заключается в том, чтобы разрезать разделительный кабель Xbox и удлинительный кабель USB пополам и соединить соответствующие концы двух кабелей. К счастью, существует стандартный код разводки для кабелей, совместимых с USB. Красный — это питание +5 В, черный — заземление, белый — данные (-), а зеленый — данные (+). Чтобы соединить кабели вместе, просто соедините провода одинаковых цветов. (Обратите внимание, что кабель Xbox будет иметь дополнительный желтый провод, который несет копию композитного видеосигнала синхронизации для использования в игровых интерфейсах типа светового пистолета. Этот дополнительный желтый провод можно спокойно игнорировать.) В этой главе вы шаг за шагом пройдете через подключение проводов и герметизацию кабеля.

### Предупреждение



**Некоторые производители USB-кабелей не соблюдают стандартный код USB-проводки. Наиболее распространенным отклонением является перепутывание белого и зеленого проводов, хотя в редких случаях цветовой код полностью игнорируется. Всегда полезно использовать измеритель целостности, чтобы убедиться, что ваш удлинительный USB-кабель действительно соответствует цветовому коду USB.**

## Выполнение

Сначала отрежьте отсоединительный кабель Xbox около (примерно 2") конца разъема Xbox, а затем отрежьте удлинительный кабель USB около конца разъема «мама». Выбросьте половину удлинительного кабеля USB и половину кабеля Xbox с меньшим разъемом.

Затем с помощью диагональных кусачек сделайте надрезы в изоляции каждого кабеля шириной 1/2 дюйма, как показано на кадре 1 рисунка 4-3. Снимите изоляцию, чтобы обнажить провода, которые будут запищены металлической оплеткой и металлической фольгой. Снимите металлическую оплётку и фольгу и отрежьте лишнюю изоляцию и экран. Зачистите концы красного, зеленого, белого и черного проводов так, чтобы было видно около 1/8 дюйма оголенного проводника. (Не зачищайте желтый провод в отводном кабеле Xbox.) Окуните оголенные концы проводников в немного паяльного флюса. (См. рисунок 4-3.)

Если вы обеспокоены прочностью или безопасностью сборки кабеля-адаптера, наденьте на один из кабелей отрезок термоусадочной трубы размером 1-1/4 дюйма. (Эта трубка позже будет надета на оголенные паяные соединения и заполнена горячим kleem для создания прочного соединения.) Экранирование не является обязательным, но кабель не

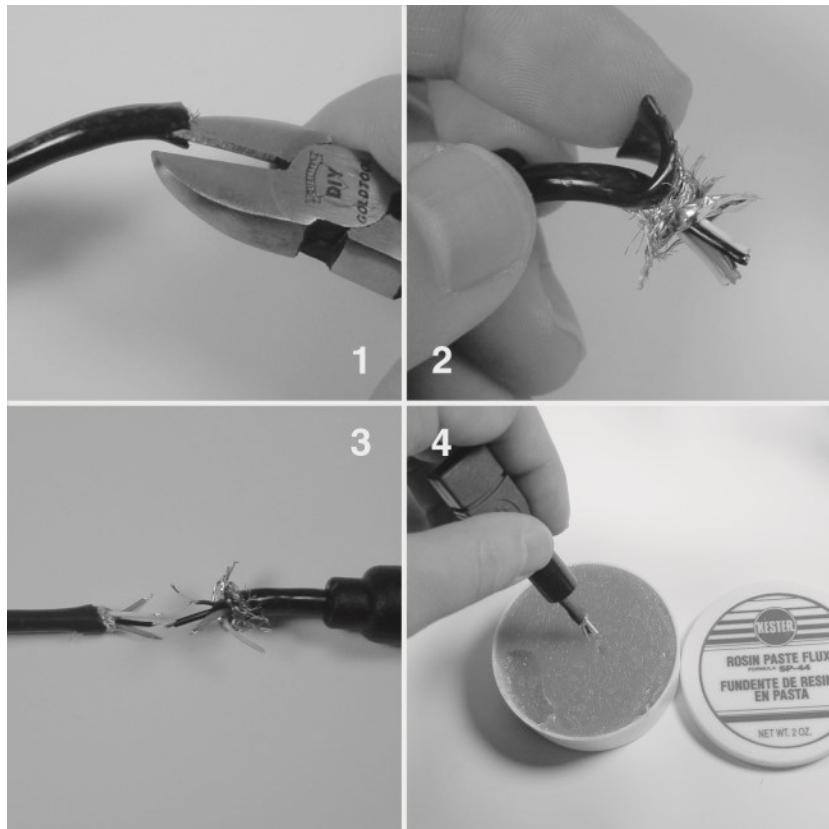
## 0

### Взлом

#### ЭХ: Введение в обратную разработку

будет таким прочным без термоусадочной арматуры; он будет подвержен поломке при многократном изгибе или растяжении.

Продолжайте процесс сборки кабеля, спаивая провода одинаковых цветов, как показано на рисунке 4-4. Попросите друга помочь вам удерживать кабели на месте, пока вы их спаиваете, или используйте инструмент «третьей руки» (заказ Jameco

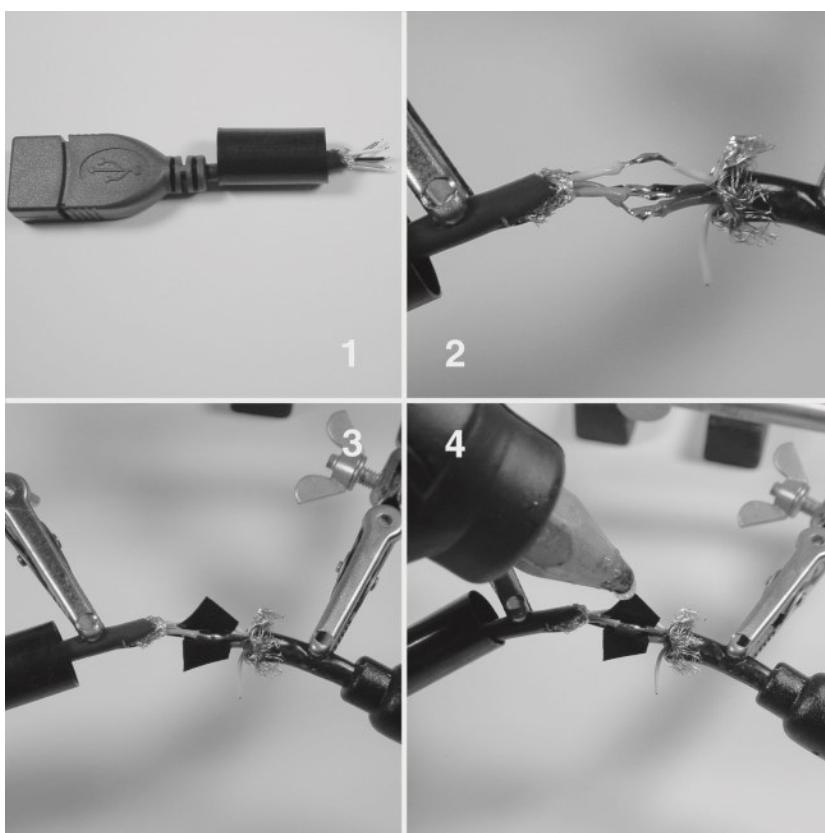


**Рисунок 4-3:** (1) Сделайте прорези в изоляции кабеля и (2) снимите изоляцию, чтобы обнажить провода и экранирование внутри. (3) Отрежьте лишнюю изоляцию и экранирование и зачистите 1/8 дюйма от конца проводов. (Обратите внимание, что желтый провод в соединительном кабеле Xbox справа не зачищен.) (4) Наконец, окуните зачищенные концы проводов в паяльный флюс.

номер 26690) с зажимами типа «крокодил», чтобы удерживать кабель на месте. Все паяные соединения должны выглядеть гладкими и блестящими. Сильно потяните за каждое соединение, чтобы убедиться, что паяное соединение хорошее. Оберните небольшой кусочек изоляционной ленты вокруг открытых соединений, чтобы

предотвратить замыкание открытых соединений. Проверьте кабель, прежде чем переходить к следующему шагу, где соединения будут постоянно изолированы.

После того, как кабель был протестирован и подтвержден как хороший, пришло время надеть прочный кожух вокруг паяных соединений для механического усиления, как показано на рисунке 4-5. Нанесите несколько небольших капель горячего клея на открытые паяные соединения, чтобы закрепить их на месте и они не закоротили друг друга, наденьте термоусадочную трубку на паяные соединения, затем заполните обе стороны трубки горячим клеем. Трубка должна сжаться от тепла клея и образовать прочный, постоянный конформный корпус над



**Рисунок 4-4:** (1) Наденьте кусок термоусадочной трубы на кабель перед пайкой. (2) Спаяйте провода вместе, одинаковый цвет с одинаковым цветом. (3) Оберните провода изолентой, чтобы предотвратить короткое замыкание. Проверьте кабель. (4) Нанесите небольшие капли горячего клея на проверяемый кабель, чтобы закрепить изоленту и провода на месте для следующего шага.

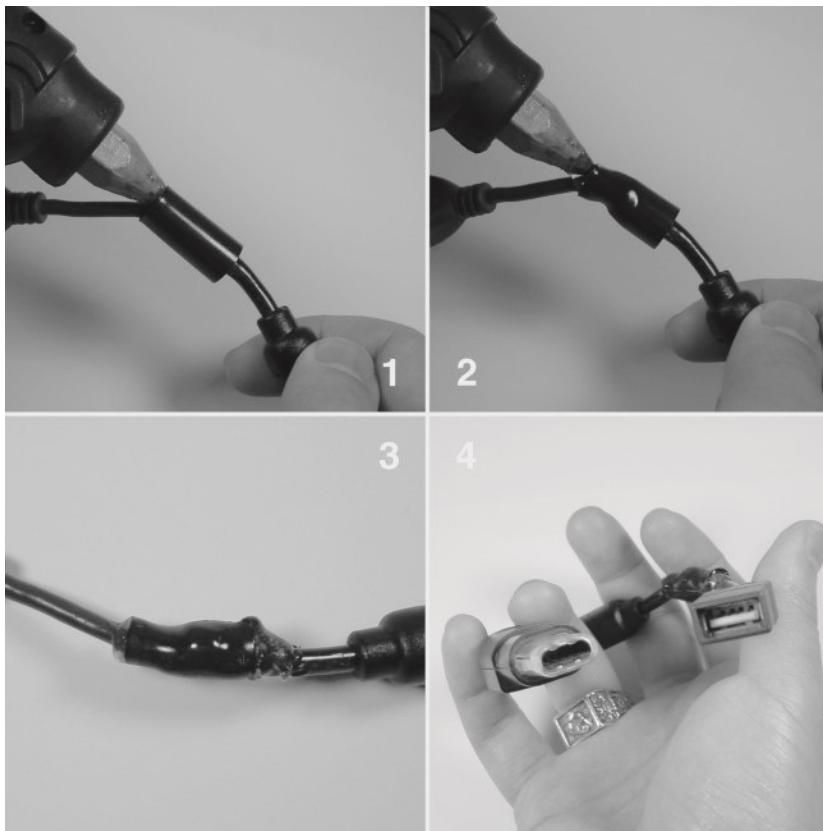
## 2 Взлом

### ЭХ: Введение в обратную разработку

Соединение. (Горячий клей также действует как средство разгрузки натяжения для соединения, поэтому кабель будет прочным в большинстве нормальных условий эксплуатации.)

Хотя кабель полностью работоспособен без обработки горячим kleem и термоусадочной трубкой, вы все равно можете улучшить стабильность соединения, если у вас нет этих предметов под рукой. Изоляционную ленту можно аккуратно обернуть вокруг отдельных соединений в качестве импровизированного механического усиления.

Теперь, когда вы закончили работу над адаптером USB-игрового порта Xbox, в главах 11 и 12 описываются некоторые шаги, необходимые для установки Linux на Xbox, чтобы вы могли использовать свой новый адаптер.



**Рисунок 4-5:** (1) Наденьте термоусадочную трубку на соединение и начните заполнять трубку горячим kleem. (2) Горячий клей приведет к смятию термоусадочной трубы. (3) Конечный продукт представляет собой постоянный конформный корпус, который выдержит большинство ненадлежащего обращения. (4) Фотография конечного продукта, на которой показаны концы кабелей Xbox и USB.

## CHAPTER 5 Замена сломанного блока питания

В случае неудачи (и на удивление частого), когда Xbox ломается после истечения трехмесячной гарантии, единственный официальный способ починить его — заплатить Microsoft за ремонт. Даже самые простые исправления могут обойтись более чем в сто долларов, или примерно в половину первоначальной цены покупки консоли. В результате я получил множество писем от людей, спрашивающих, как починить сломанные блоки питания и жесткие диски.

К сожалению, замена сломанного жесткого диска требует взлома системы безопасности Xbox, поскольку для блокировки материнских плат Xbox на определенном жестком диске используется уникальный ключ. Установка нового жесткого диска потребует модчипа, который может перепрограммировать или обойти блокировку жесткого диска. Кроме того, требуется копия заводского программного обеспечения Xbox, что является незаконным для распространения или копирования даже в целях ремонта. Поэтому тема замены жесткого диска Xbox слишком рискованна, чтобы обсуждать ее в этом тексте. Читателям рекомендуется поискать в Интернете любой из многочисленных часто задаваемых вопросов о замене жестких дисков.

С другой стороны, блок питания, используемый в Xbox, очень похож на тот, который используется в стандартном ПК. Существует множество веб-сайтов, где вы можете приобрести точную замену блоков питания Xbox, например, Llama.com ([www.llama.com/xbox/Repairs/repairs.htm](http://www.llama.com/xbox/Repairs/repairs.htm)), XboxRepair.com ([www.xboxrepair.com](http://www.xboxrepair.com)) и Firefly-HK ([www.firefly-hk.com](http://www.firefly-hk.com)), или вы можете попытаться собрать его самостоятельно из стандартного блока питания ПК!

Учитывая частоту сбоев питания, я покажу вам, как адаптировать стандартный блок питания ПК ATX для Xbox. Описанный в этом разделе подход не требует пайки, за счет необходимости переключать дополнительный переключатель для включения Xbox. Приложение C, «Введение в схему печатной платы», описывает простой проект, который вы можете реализовать, чтобы иметь удобство без дополнительных переключателей питания. Конечно, проще купить точный сменный блок питания для Xbox и использовать части процедуры, представленной в этой главе, чтобы помочь вам с установкой. Однако прямая замена имеет меньшую образовательную ценность, чем адаптация. Если вы решите адаптировать блок питания ATX для использования с Xbox, вы научитесь делать обжимные кабели, а также узнаете немного о теории электроники и о том, как работает Xbox.

Еще одна причина адаптации стандартного блока питания ПК ATX к Xbox — это обеспечение дополнительной мощности консоли. OEM (производитель оригинального оборудования) или «печатный» блок питания Xbox выдает едва достаточно мощности для удовлетворения потребностей Xbox. Подключение

## **4** Взлом

### **ЭХ: Введение в обратную разработку**

дополнительных приводов или вентиляторов к не модифицированному Xbox может перегрузить OEM блок питания Xbox и привести к его перегоранию.

Обратите внимание, что на момент написания этой статьи была выпущена новая аппаратная версия Xbox (известная как «v1.2» в сообществе хакеров Xbox), которая, по-видимому, имеет стандартный разъем питания ATX вместо фирменного разъема питания Xbox, описанного далее в этой главе. Убедитесь, что разъем питания вашего Xbox соответствует разъему, описанному в этой главе, прежде чем приступать к процедуре адаптации. Разъем питания Xbox, предполагаемый в этой главе, имеет двенадцать контактов в одном ряду, тогда как новый разъем питания Xbox имеет 20 контактов, расположенных в два ряда по десять. Кроме того, хотя последняя аппаратная версия Xbox имеет разъем типа ATX, он не обязательно электрически совместим со стандартным блоком питания ATX. Было бы разумно измерить напряжения на блоке питания разъем и сравните их со спецификацией ATX

([www.formfactors.org/developer/specs/atx/atx2\\_1.pdf](http://www.formfactors.org/developer/specs/atx/atx2_1.pdf)) перед попыткой подключить стандартный блок питания ATX к Xbox с новым разъемом типа ATX.

#### **Осторожность**



Замена блока питания может подвергнуть вас воздействию опасного напряжения. Перед извлечением блока питания всегда отключайте Xbox от розетки и ждите минуту, чтобы рассеялись накопленные заряды. Кроме того, неправильная установка сменного блока питания может привести к необратимому, даже взрывному повреждению консоли. Выполните эту процедуру только в том случае, если вы уверены, что блок питания разряжен и выключен, и если вы готовы пойти на риск дальнейшего повреждения консоли.

## **Диагностика неисправного блока питания**

Если у вашего Xbox возникли проблемы с включением питания, сначала необходимо диагностировать проблему и найти источник сбоя. Бесполезно заменять блок питания, если неисправность на самом деле находится в консоли или в розетке. Выполните эти диагностические шаги, чтобы убедиться, что именно блок питания Xbox является виновником, а не что-то другое.

1. Проверьте работоспособность розетки, подключив к ней лампу. Используйте лампу мощностью не менее 100 Вт, чтобы точно имитировать нагрузку Xbox.
2. Визуально проверьте шнур питания на предмет перегибов и порезов.
3. Убедитесь, что вилка шнура питания надежно вставлена в розетку питания Xbox и что Xbox по-прежнему не включается, несмотря на эти проверки.
4. Визуально осмотрите внутреннюю часть Xbox на предмет следов обугливания или разорванных конденсаторов. Если следы обугливания видны на материнской плате, вам, возможно, придется заменить материнскую плату Xbox (т. е. купить новый Xbox). Если следы обугливания есть на блоке питания, скорее всего, блок питания был поврежден, и вы можете приступить к его замене. (Имейте в виду, что отказ блока питания может также повредить материнскую плату, поэтому все еще есть вероятность, что Xbox не будет работать после замены блока питания.)
5. Убедитесь, что основной разъем питания и разъем платы передней панели надежно закреплены. (Расположение разъема платы передней панели показано на рисунке 3-3 главы 3 «Установка синего светодиода».) Выключатель питания Xbox подключается к материнской плате через разъем платы передней панели.
6. Пока Xbox выключен, но все еще подключен, используйте вольтметр, чтобы убедиться, что напряжение в режиме ожидания 3,3 В (3,3VSB) находится в пределах спецификации. Измерьте 3,3VSB, проверив шестой провод в разъеме блока питания с конца, ближайшего к передней панели, и любой из черных проводов на разъеме питания. Вы можете измерить напряжение питания, вставив кончики щупа вольтметра в свободное пространство между проводами питания и разъемом питания. (Вокруг проводов питания внутри корпуса разъема питания материнской платы есть металлический воротник.) Если значение 3,3VSB не находится в диапазоне от 3,14 до 3,47 вольт, вам может потребоваться заменить блок питания.
7. Нажмите кнопку питания на Xbox, чтобы включить его (предположительно, если блок питания сломан, Xbox не будет работать). Если блок питания шумит или дымит, отключите коробку и приступайте к замене блока питания. Если коробка кажется мертвый, измерьте каждое из основных напряжений, поступающих от блока питания. Желтый провод должен иметь напряжение от 11,4 до 12,6 вольт; красный провод должен иметь напряжение от 4,8 до 5,25 вольт; а оранжевый провод должен иметь напряжение от 3,14 до 3,47 вольт. (Все эти напряжения

---

#### Взлом Xbox: Введение в обратную разработку

указаны относительно черного провода.) Также проверьте, что напряжение на сигнале Power OK (расположенном на контакте 12, самом дальнем от передней панели) превышает 3,1 вольт. Если все в списке на предыдущей странице проверено, то маловероятно, что проблема в вашем блоке питания. Далее следует проверить электрическую и механическую целостность выключателя питания (см. Главу 3 о том, как снять плату с выключателем питания) и функциональность материнской платы. Однако если вы заметили признаки отказа блока питания, читайте дальше.

## Замена блока питания

Общая стратегия замены блока питания Xbox заключается в адаптации стандартного блока питания ПК ATX для использования в Xbox. Вот список необходимого вам оборудования:

- **Стандартный блок питания ATX.** Блок питания высотой 1U поместится в корпус Xbox, но, вероятно, будет немного высоковат, чтобы закрыть корпус.
- **(Необязательно) Удлинитель кабеля питания материнской платы ATX.** Удлинители кабеля питания можно приобрести у многочисленных поставщиков, включая PC Power and Cooling.  
[www.pcpowerandcooling.com](http://www.pcpowerandcooling.com)). Модификация удлинительного кабеля для Xbox вместо кабеля блока питания ATX позволяет повторно использовать блок питания в стандартном ПК, когда вы будете готовы выбросить Xbox.
- **Инструмент для обжима.** Универсальный обжимной инструмент Molex (номер детали Digi-Key WM9999-ND) настоятельно рекомендуется, но он немного дороже (около \$35). Более дешевый обжимной инструмент, такой как Jameco 159265, можно купить примерно за треть цены, но он более неудобен в использовании, и вам, возможно, придется использовать припой для обжима, чтобы достичь желаемой прочности соединения.
- **Один 12-позиционный корпус разъема с шагом 0,156 дюйма** (например, Digi-Key номер детали WM2313-ND) или два штабелируемых 6-позиционных корпуса разъема с шагом 0,156 дюйма (например, Jameco номер детали 104731). Этот корпус используется для замены разъема питания Xbox.
- Тринадцать обжимных клемм для

разъема питания с шагом 0,156 дюйма (например, Digi-Key номер детали WM2313-ND или Jameco номер детали 78318).

- **Два кремниевых выпрямительных диода 1N4001 или лучше в корпусе DO-41**(например, номер детали Digi-Key 1N4001DICT-ND или номер детали Jameco 35975).
- **Инструмент для зачистки проводов.**Подойдет любой инструмент для зачистки проводов, способный работать с проводами калибра 18.
- **Кусачки.**Подойдут любые диагональные кусачки.
- **Изолента.**

## Использование диодов для понижения напряжения

Xbox требует резервного напряжения питания +3,3 В, но блок питания ATX выдает только резервное напряжение +5 В. «Правильным» решением этой проблемы было бы использование регулятора напряжения, который точно преобразует +5 В в +3,3 В, но цель этого хака — заменить блок питания с минимальным количеством пайки.

Альтернативное решение — использовать два диода, чтобы снизить напряжение +5 В до «достаточно близкого» напряжения +3,6 В. Мы можем это сделать, поскольку напряжение на прямопроводящем диоде логарифмически пропорционально току через диод. Другими словами, для большинства токов напряжение на диоде почти постоянно. Оказывается, что кремниевые диоды почти равномерно имеют прямое падение напряжения около 0,7 В, поэтому два из них, соединенных последовательно, дадут 1,4 В.

Диоды, используемые в этом хаке, 1N4001, способны проводить только 1 ампер тока, поэтому не используйте этот трюк в других приложениях, требующих большого тока. К счастью, резервный источник питания для Xbox должен потреблять лишь небольшое количество тока, поэтому сгорание диодов не является проблемой.

В качестве последнего замечания, напряжение, падающее на диоде, немного колеблется в зависимости от величины тока через него, поэтому не используйте этот трюк в приложениях, требующих точной регулировки напряжения. В приложении Xbox мы немного повышаем напряжение, но, к счастью, цифровая логика, работающая от этого источника питания, может выдержать это состояние.

## Стратегия

Интерфейс стандартного блока питания ATX очень похож на интерфейс блока питания Xbox. Xbox требует +3,3 В, +5 В, +12 В, резервный источник питания +3,3 В, а также два управляющих сигнала: «power OK» и «power on». Сигнал power OK указывает на то, что выходная мощность блока питания стабильна и правильно регулируется, а сигнал Power On — это управляющий сигнал от Xbox, который включает и выключает блок питания. Типичный блок питания ATX имеет выходы +3,3 В, +5 В и +12 В с достаточным количеством энергии для работы Xbox, а также имеет сигнал power OK, совместимый с материнской платой Xbox. Однако блок питания ATX генерирует резервное напряжение +5 В вместо резервного напряжения +3,3 В, а сигнал Power On имеет инвертированную полярность от Xbox. Обе эти несовместимости можно устраниć способом, не требующим пайки. Два последовательно соединенных диода используются для снижения напряжения ожидания +5 В до напряжения чуть меньше +3,6 В. Сигнал включения питания блока питания ATX по умолчанию имеет значение «включено», поэтому он останется неподключенными, делая блок питания всегда включенным, даже если консоль выключена. Это не проблема для электроники консоли, но может быть эстетически неудобным. Приложение C, «Введение в компоновку печатной платы», описывает пример конструкции, которую вы можете реализовать, чтобы избежать этих несовместимостей более изящным образом. Однако конструкция, описанная в приложении, потребует от вас определенных усилий в виде пайки и проектирования платы.

## Процедура

Процедура замены блока питания Xbox состоит из двух частей:

1. Модификация стандартного кабеля питания ATX в кабель питания Xbox.
2. Снятие старого блока питания и установка нового.

## Сборка кабеля питания Xbox

Начните с отрезания разъема материнской платы существующего блока питания ATX, как показано на рисунке 5-1. Вы можете выбрать выполнение этой модификации с использованием удлинительного кабеля материнской платы ATX, чтобы сохранить разъем питания ATX на блоке питания для будущего использования. Процедура идентична для обоих вариантов, но фотографии в этой главе сделаны с

использованием удлинительного кабеля материнской платы ATX.

Теперь прикрепите обжимные клеммы к следующим проводам кабеля ATX, как показано на рисунке 5-2:

- Один желтый провод
- Три красных провода
- Один оранжевый провод
- Четыре черных провода
- Один серый провод

Если вы используете более дешевый обжимной инструмент, у вас могут возникнуть проблемы с созданием достаточно прочного обжимного соединения. В этом случае завершите соединение, припаяв обжимной наконечник к проводу. Используйте обильное количество тепла при пайке, иначе припой не полностью проникнет в провод и обжимной наконечник. Паяльник должен находиться в контакте с соединением около пяти секунд до и после нанесения припоя.

На фиолетовом проводе (проводе ожидания +5 В) подключите два диода последовательно между концом провода и обжимной клеммой. Процедура, показанная на рисунке 5-3, использует части обжимных клемм для соединения диодов, поэтому пайка не требуется. (Обратите внимание, что диоды являются поляризованными устройствами: они не будут проводить электричество, если установлены в обратном порядке. Диоды следует устанавливать катодами (концом с нарисованной на нем полосой) к материнской плате.)



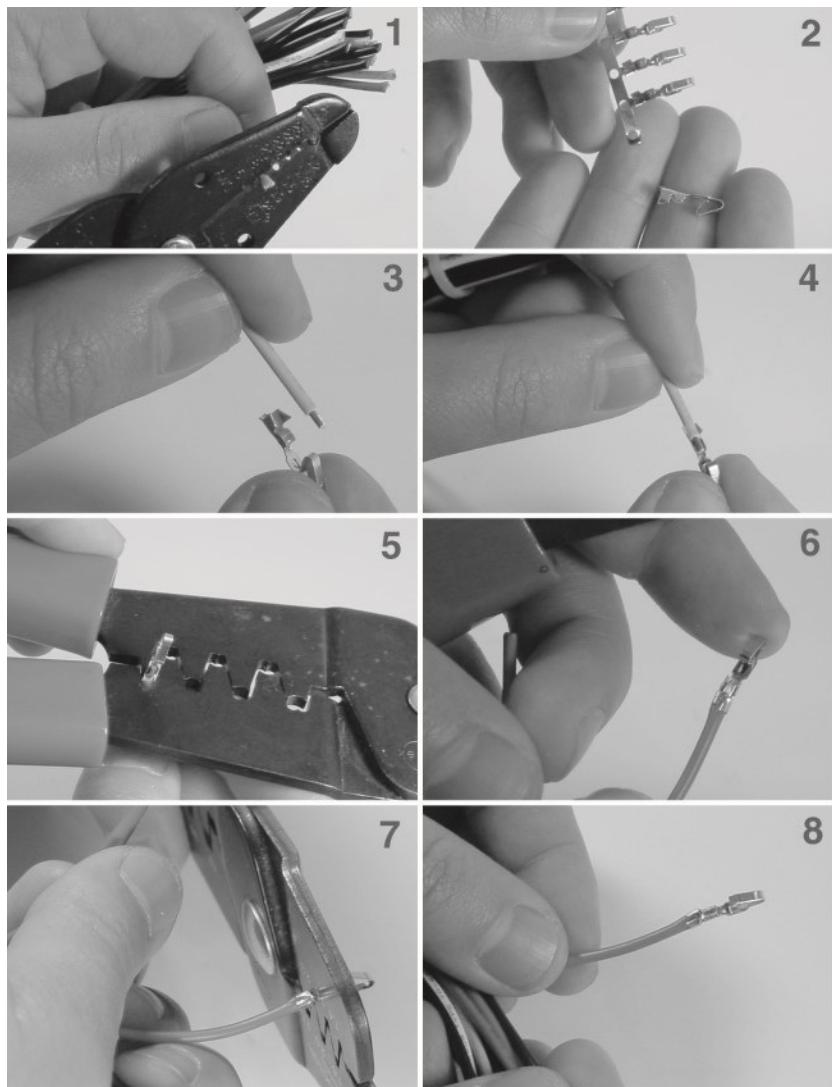
**Рисунок 5-1:** Отрежьте разъем от кабеля питания ATX.

Затем обмотайте все неиспользуемые провода кабеля ATX изолентой, чтобы предотвратить случайные замыкания, которые могут повредить блок питания и, возможно, Xbox. Не забудьте также обмотать диоды изолентой. (См. рисунок 5-4.)

Наконец, вставьте готовые обжимные клеммы в корпус разъема 0,156". Обжимные клеммы зафиксируются на месте внутри корпуса, когда они будут полностью и правильно вставлены. (См. рисунок 5-5.) Вставьте провода в порядке, указанном в таблице 5-1.

Некоторые поставщики не продают более крупные 12-позиционные корпуса разъемов. В этом случае используйте два 6-позиционных корпуса разъемов и уделите особое внимание порядку, который вы выбираете для укладки разъемов. Также обратите внимание на расположение контакта 1 относительно поляризационного выступа разъема. Контакт 1 находится в верхней части разъема, когда поляризационный выступ находится слева, и вы смотрите на разъем со стороны, с которой вставляются провода (вид сверху). Поскольку очень легко перевернуть разъем и каким-либо образом изменить его

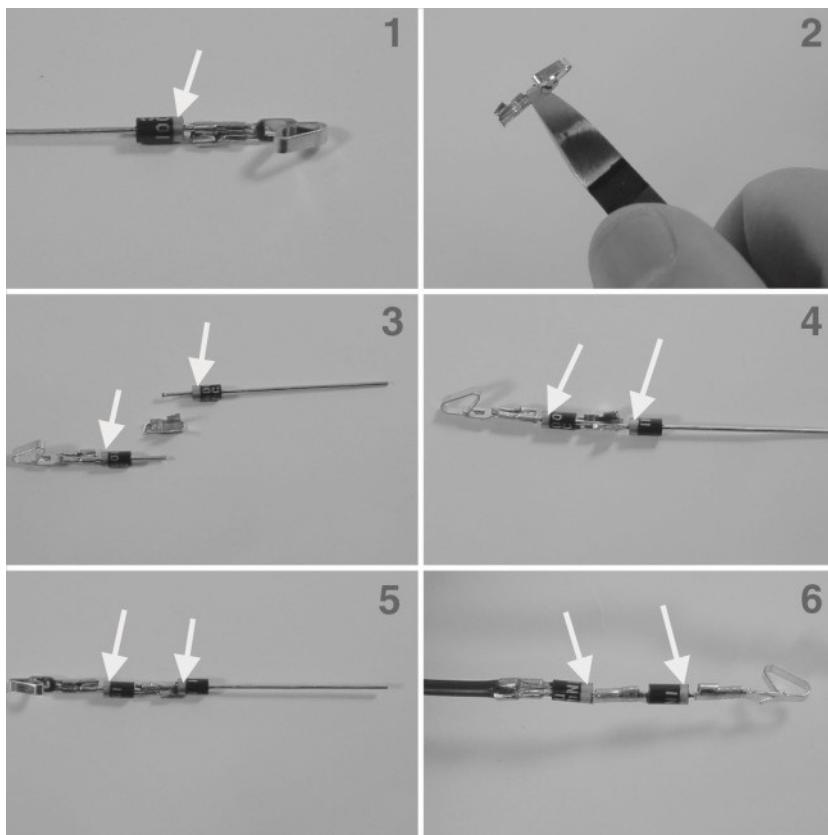
положение, используйте существующий разъем Xbox в качестве ориентира. Желтый, красный, оранжевый и черный провода должны быть выровнены при сравнении друг с другом.



**Рисунок 5-2:** Присоединение обжимной клеммы к концу провода. (1) Снимите около  $1/8"$  изоляции с конца провода. (2) При необходимости снимите чистую обжимную клемму с удерживающей полосы. (3 и 4) Вставьте провод в обжимную клемму так, чтобы  $1/16"$  изоляции находилось между более длинной парой обжимных пальцев. (5) Обожмите изоляционную часть (более длинную пару) обжимных пальцев. (6) Показан провод, у которого обжата только изоляционная часть. (7) Обожмите

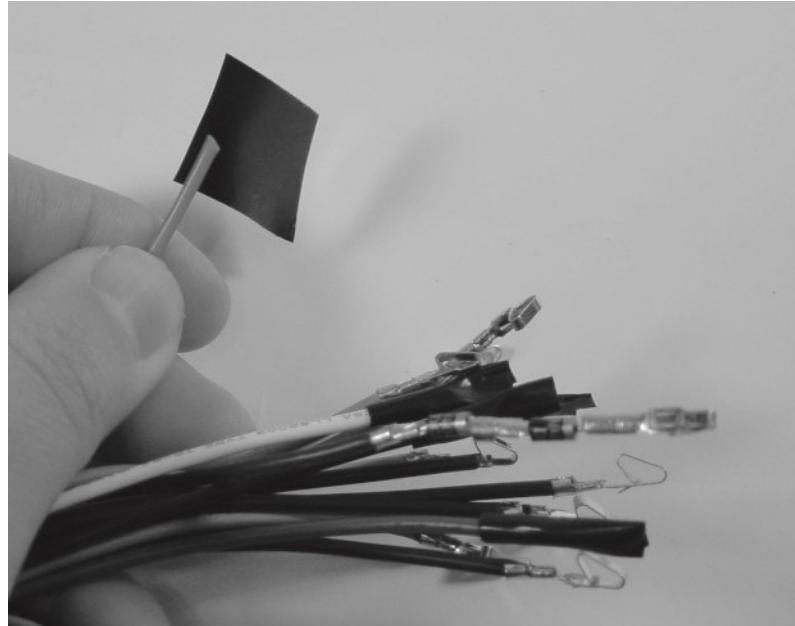
## Взлом Xbox: Введение в обратную разработку

проводящую часть обжимных пальцев. (8) Готовая обжимная клемма. Обжимные клеммы проводника должны быть плотно сложены на оголенном проводе для обеспечения хорошего контакта. Проверьте обжимное соединение, с усилием потянув за конец клеммы.



**Рисунок 5-3:** Присоедините два диода последовательно между концом фиолетового провода и обжимной клеммой, которая входит в разъем питания. Стрелки указывают правильную ориентацию поляризационной линии, нарисованной на конце диода. Эта процедура потребует в общей сложности три обжимных клеммы. (1) Присоедините один диод к обжимной клемме с поляризационной полосой рядом с обжимной клеммой. (2) Разрежьте обжимную клемму пополам, чтобы удалить листовой контакт. (3 и 4) Расположите диоды в обжимной части разрезанного разъема, отметив полярность диодов. (5) Диоды показаны после обжима. (6) Присоедините диоды к концу фиолетового провода, используя ту же процедуру со второй разрезанной обжимной клеммой.

**Дважды проверьте свою работу** после завершения сборки кабеля питания, так как любая ошибка может привести к постоянному, непоправимому повреждению консоли. На рисунке 5-6 показано, как должна выглядеть готовая сборка разъема. Вы можете использовать кабельную стяжку, если она есть, чтобы связать неиспользуемые провода так, чтобы они не мешали и не закорачивались или не повреждались.

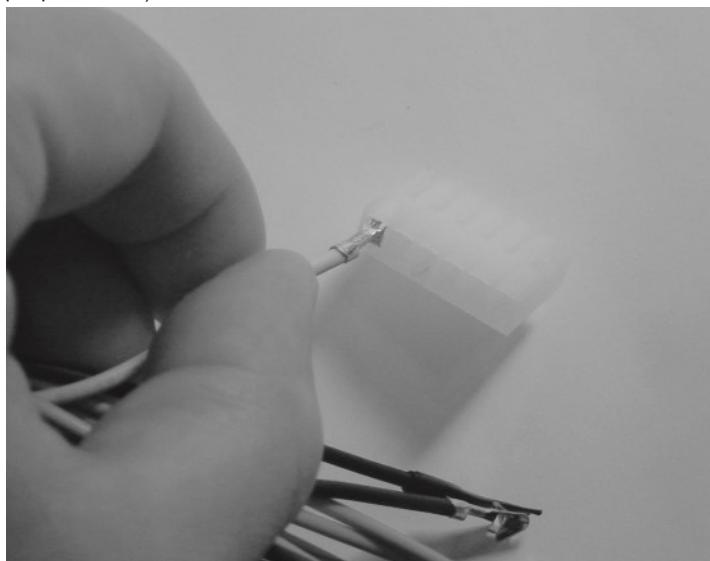


**Рисунок 5-4:** Оберните неиспользуемые провода и диоды изолентой, чтобы предотвратить случайное замыкание.

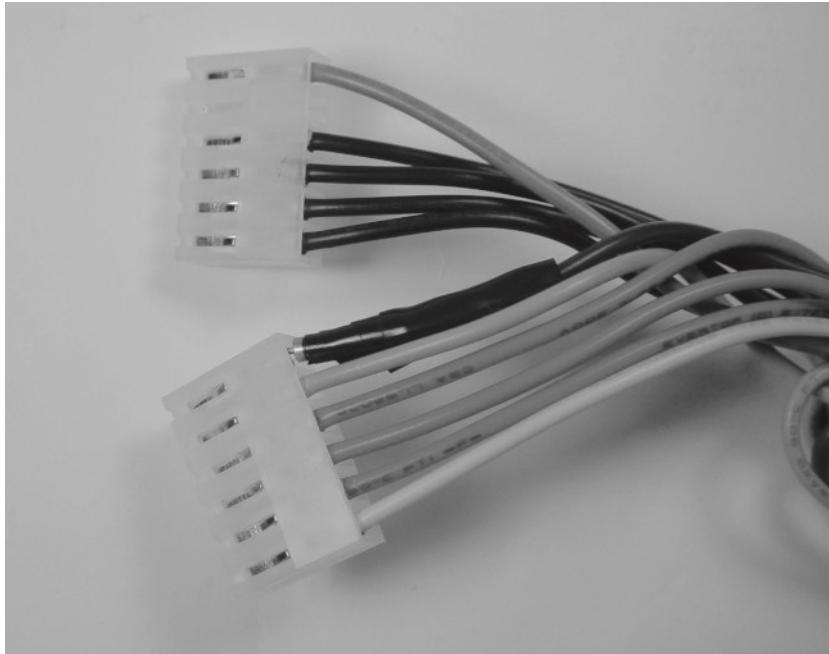
Приколоть Цвет

	1	Желтый
9	2	Красный
	3	Красный
	4	Красный
	5	Апельсин
	6	Диоды + Фиолетовый
	7	Черный
	8	Черный
	9	Черный
	10	Черный
	11	Пустой серый
	12	

**Таблица 5-1:** Таблица разводки для подключения кабеля питания ATX к разъему питания Xbox (вид со стороны ввода провода, с поляризационным ребром слева).



**Рисунок 5-5:** Вставка обжимного соединителя в разъем.



**Рисунок 5-6:** Окончательная сборка кабеля.



Рисунок 5-7: Расположение двух винтов крепления блока питания.

## Установка сменного блока питания

Теперь, когда мы подготовили сменный блок питания, нам нужно заменить наш старый, сломанный. Сначала снимите верхнюю часть корпуса Xbox, как описано в Главе 1, «Аннулирование гарантии», затем отсоедините разъем питания жесткого диска и выньте жесткий диск из корпуса. Вам не нужно отсоединять серый ленточный кабель IDE, подключенный к жесткому диску.

### Примечание



На этом этапе убедитесь, что Xbox отключен от сети и что у него была возможность постоять хотя бы минуту, чтобы рассеять накопленный заряд в блоке питания. Работать с Xbox, когда он подключен к сети или вскоре после того, как он был отключен, крайне опасно. Xbox вызовет неприятный, возможно, смертельный удар током, если вы прикоснетесь к любой части блока питания голыми руками до того, как он рассеет накопленный заряд.

1. Отсоедините разъем питания Xbox, взявшись за весь пучок проводов питания и крепко потянув за кабель, удерживая коробку другой рукой. При извлечении кабеля будьте осторожны с острыми металлическими краями корпуса и радиаторами.
2. Открутите два винта Torx T-10, которые удерживают блок питания Xbox на месте. (См. рис. 5-7.)
3. Извлеките блок питания из корпуса Xbox, сначала приподняв конец блока питания ближе к передней части Xbox.
4. Сравните созданный вами кабель с блоком питания Xbox. кабель, убедившись, что выровняли поляризационные ребра. Красный, желтый, оранжевый и черный провода должны быть выровнены между двумя разъемами. (Эта проверка поможет вам убедиться, что вы не повредите Xbox из-за ошибки в подключении кабеля.)
5. Подключите кабель питания ATX к разъему питания Xbox. Желтый провод должен совпадать с контактом, ближайшим к передней части Xbox. Помните, что ничего не мешает вам вставить кабель питания со смещением на один или два контакта. Если вы видите оголенный контакт на разъеме питания, вы вставили кабель со смещением на один контакт. Дважды проверьте это состояние, поскольку оно может привести к неизбежному повреждению оборудования Xbox и/или блока питания.
6. Подключите жесткий диск Xbox к одному из разъемов питания дисковода блока питания ATX. Жесткий диск использует разъем, идентичный стандартному разъему питания дисководов ПК, поэтому никаких модификаций не требуется.
7. Проверьте, нет ли закороченных проводов на корпусе или зажатых в лопастях вентиляторов охлаждения. Теперь вы готовы включить Xbox.

## Работа с запасным блоком питания

Большинство блоков питания ATX поставляются с выключателем питания. Установите этот переключатель в положение «выкл.», затем подключите блок питания ATX, а затем включите выключатель питания. В этот момент блок питания ATX будет подавать питание на Xbox, даже если системный контроллер Xbox считает, что приставка выключена. В результате вентиляторы охлаждения внутри Xbox

---

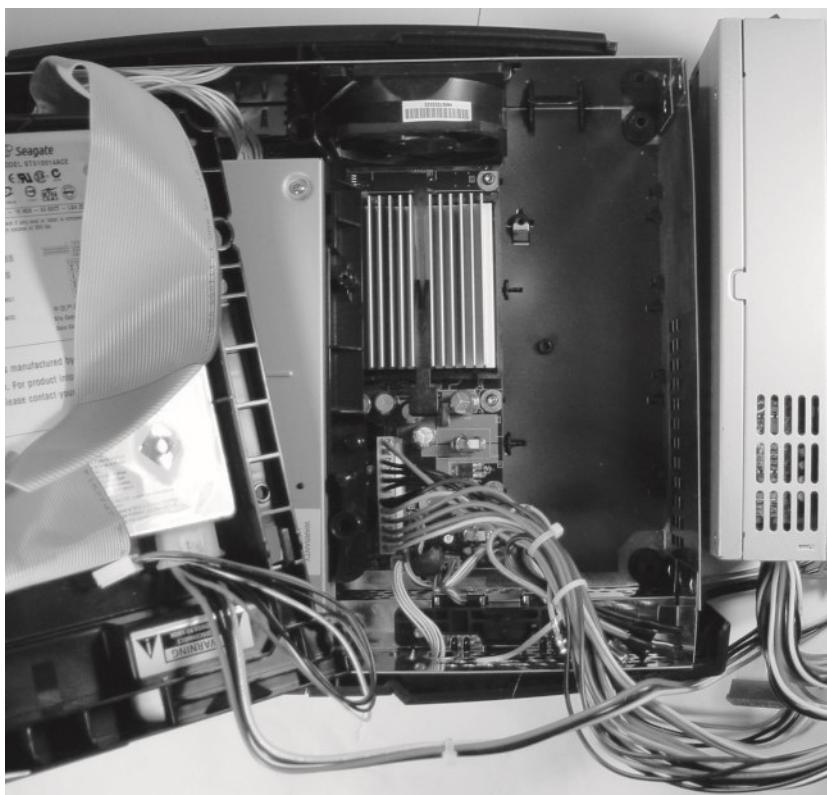
### Взлом Xbox: Введение в обратную разработку

должны начать работать. Теперь нажмите выключатель питания на передней панели Xbox. В этот момент Xbox должен нормально включиться. Если это так, поздравляем!

#### Осторожность



В некоторых версиях Xbox отсутствует вентилятор радиатора для графического процессора. Если на материнской плате Xbox нет вентилятора, установленного непосредственно на одном из чипов, то у вас именно такой Xbox. Xbox, у которых отсутствует вентилятор радиатора для графического процессора, склонны к перегреву при определенных условиях и должны работать с дисководами, установленными над материнской платой для надежной и долгосрочной работы. Нижняя часть дисководов образует систему воздуховодов, которая направляет воздух над радиатором графического процессора от основного вентилятора корпуса Xbox. (Ваш кабель питания ATX не позволит установить дисководы вровень с корпусом, но это не является серьезной причиной для беспокойства относительно воздуховодов.)



**Рисунок 5-8:** Блок питания ATX форм-фактора 1U slimline, подключенный к Xbox с помощью модифицированного удлинительного кабеля блока питания ATX.

Когда вы будете готовы выключить Xbox, вы можете просто переключить выключатель питания на блоке питания ATX. Или вы можете использовать выключатель питания на передней панели Xbox, чтобы сначала выключить Xbox, а затем выключить блок питания ATX.

## Советы по отладке

Если Xbox не включается должным образом после замены блока питания, проверьте кабель питания, используя контрольный список в разделе в начале этой главы под названием «Диагностика сломанного блока питания». Наиболее вероятная проблема, с которой вы столкнетесь, — это плохое обжимное соединение или плохо или неправильно прикрепленные диоды. Плохое обжимное соединение также может привести к прерывистой работе, когда Xbox включается, но часто выходит из строя.

Если Xbox включается, но по какой-то причине останавливается во время последовательности включения, см. Таблицу 3-1 в разделе «Отладка» в конце Главы 3 для списка возможных проблем и их причин. Приложение E, «Отладка: Советы и подсказки», содержит более подробное обсуждение методов и методологии отладки.

Если Xbox работает нормально, но иногда дает сбои, возможно, у вас Xbox, на котором нет вентилятора радиатора над графическим процессором. См. предупреждение на соседней странице, описывающее эту проблему. Чтобы устранить эту проблему, вам может потребоваться добавить дополнительный вентилятор или улучшить существующую систему воздуховодов с помощью листа бумаги и липкой ленты.



## CHAPTER 6 Лучшая игра Xbox: взлом безопасности

Следующий шаг после модификации и настройки оборудования Xbox — это получение контроля над оборудованием Xbox. К сожалению, получить контроль над оборудованием не так просто, как можно было бы подумать. Разработчики Xbox вложили много мыслей в защиту оборудования от сложных программных атак, а также от большинства простых аппаратных атак. Механизмы безопасности Xbox являются артефактом его архитектуры управления цифровыми правами.

### Примечание



В **принципе**, применение оборудования для целей «добросовестного использования», например, для запуска собственных программ-самоучок, не должно быть незаконным. Однако связь между добросовестным использованием, защищенным оборудованием и относительно новыми законами об обходе авторских прав все еще неясна. В главе 12 «Caveat Hacker» более подробно обсуждаются правовые вопросы взлома.

Существует множество способов обойти меры безопасности Xbox. В этой главе и в Главе 8 «Обратная разработка безопасности Xbox» я рассказываю историю своих приключений по составлению карты системы безопасности Xbox. Я пишу не только об успехах, но и о неудачах, с которыми я столкнулся, чтобы вы могли извлечь уроки из моего опыта. Глава 9 «Проникновение в бэкдор» объясняет некоторые подходы, которые используют другие, чтобы обойти меры безопасности Xbox. Глава 7 «Краткий вводный курс по безопасности» дает справочную информацию, необходимую для понимания Глав 8 и 9.

## Первые встречи с параноидальным дизайном

Когда Xbox был анонсирован весной 2000 года, волнение охватило сообщество энтузиастов оборудования. Причиной этого волнения был не только игровой потенциал Xbox, но и его потенциал для использования в качестве высокопроизводительного сетевого ПК с архитектурой x86 по доступной цене в 300 долларов. Снижение цен через несколько месяцев после его появления с тех пор снизило стоимость Xbox до менее 200 долларов. Сходство Xbox с ПК x86 означало, что

---

## Взлом Xbox: Введение в обратную разработку

огромная база существующих приложений и опыта могла быть, в теории, легко перенесена на консоль.

Впервые я заглянул внутрь Xbox в конце ноября 2001 года, когда моя девушка (теперь невеста) подарила мне его в качестве раннего рождественского подарка. Я немедленно приступил к делу. Чтобы взять под контроль аппаратное обеспечение Xbox, первой задачей является извлечение загрузочного ПЗУ и анализ его содержимого: Вспомним из обсуждения архитектуры Xbox в Главе 2, что загрузочное ПЗУ Xbox содержит весь код для установки операционной среды Xbox.

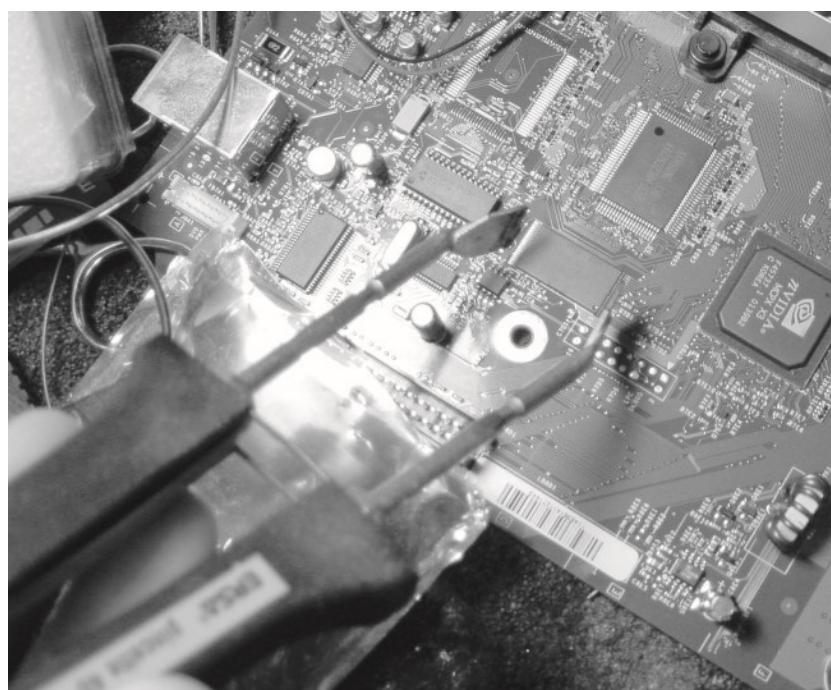
### Чтобы записать ПЗУ

Тип ПЗУ, используемый в Xbox, — это электрически стираемая и программируемая разновидность, известная как FLASH ROM. FLASH ROM обычно поставляется в одном из нескольких типов корпусов, и Xbox использует один из самых популярных корпусов, TSOP (Thin Small Outline Package). Он расположен в секторе U7 на верхней стороне материнской платы Xbox, а справочное обозначение для этой детали — U7D1. Корпус TSOP очень узнаваем, поскольку это один из немногих корпусов микросхем, который имеет прямоугольную форму и имеет контакты только на узких краях корпуса. Большинство других корпусов размещают контакты на длинном крае или на всех краях для максимизации подключения, но FLASH ROM имеет относительно низкие требования к вводу/выводу на единицу площади кремния. Быстрая проверка базового номера детали, 29F080, с помощью поисковой системы в Интернете подтверждает, что эта деталь действительно является 8-мегабитной ФЛЭШ-ПЗУ.

Есть несколько методов, которые можно использовать для считывания (snarf) содержимого FLASH ROM. Подход без пайки заключается в покупке тестового зажима, который защелкивается на FLASH ROM, и считывании его содержимого путем включения питания и управления ROM через тестовый зажим, в то время как остальная часть Xbox выключена. Подходящий для этой цели тестовый зажим можно приобрести в Emulation Technology, [www.emulationtechnology.com](http://www.emulationtechnology.com). (Подход с переопределением тестового зажима имеет несколько проблем, самая большая из которых — возможность необратимого повреждения микросхем, подключенных к FLASH ROM, которые не получают питание через тестовый зажим. Однако в случае Xbox это, похоже, не является проблемой, и те, кто пытался

использовать этот подход, добились успеха.<sup>9</sup>(Поначалу я не прибегал к такому подходу, поскольку не хотел рисковать повреждением материнской платы, а также потому, что не мог позволить себе тестовый зажим стоимостью 300 долларов, необходимый для этой работы.)

Другой гениальный подход заключается в пайке проводов к контрольным точкам вокруг FLASH ROM, чтобы подслушивать Xbox, пока он считывает содержимое ROM. Подслушивание может быть выполнено либо путем подключения проводов к специальной плате, которая может взаимодействовать с ROM, либо путем использования логического анализатора для захвата данных, когда к ним обращается процессор Xbox. Последний подход также был успешно использован, и фактически некоторые бэкдоры в последовательности загрузки Xbox были обнаружены в результате



<sup>9</sup>У Энди Грина есть прекрасная страница, на которой он документирует свой опыт применения метода тестовых клипов:  
<http://www.warmcat.com/milksop/milksop.html>  
{сайт недоступен, увы}

---

Взлом Xbox: Введение в обратную разработку  
**Рисунок 6-1:** Извлечение Xbox FLASH ROM с помощью паяльника-пинцета.

эта методология.<sup>10</sup> Я также решил не использовать этот подход, поскольку у меня не было логического анализатора, когда я получил свой первый Xbox, и поскольку пайка всех проводов может быть очень утомительной, сложной и подверженной ошибкам. Мой подход был более традиционным: просто извлеките FLASH ROM и вставьте его в считыватель ROM. Я также разместил гнездо на материнской плате, чтобы в будущем извлечение и программирование ROM было очень быстрым и надежным.

Извлечение FLASH ROM таким образом, чтобы сохранить целостность его мелкошаговых контактов, просто, если у вас есть правильные инструменты, и почти невозможно, если у вас неправильные инструменты. Главное — нагреть все контакты FLASH ROM одновременно; как только будет достигнут равномерный нагрев, FLASH ROM сразу же отвалится от материнской платы. Очевидно, что стандартный паяльник в виде карандаша не сможет нагреть все контакты одновременно. Правильным инструментом для этой работы является паяльник в виде «щипцов» или «пинцета», как показано на рисунке 6-1 ниже. Эти паяльники имеют два нагревательных элемента, поэтому они могут нагревать обе стороны чипа одновременно. Кроме того, паяльник должен иметь лопасть, которая достаточно широка, чтобы нагреть всю длину чипа одновременно.

Паяльник с такими функциями может стоить довольно дорого (сотни долларов), но это стоящая инвестиция, поскольку она пригодится в самых разных ситуациях. Я использую паяльник Ersa SMT Unit 60A, который я купил с хорошей скидкой на торговой выставке, и он быстро окупился за счет нескольких работ по сборке плат, которые я подбирал попутно, пока заканчивал обучение. Более доступный паяльник от Xytronics можно купить через Jameco (#168410) примерно за 70 долларов, но я им не пользовался, поэтому не могу ручаться за его качество. Другой бюджетный подход, который очень прост и понятен, — это использование сплава для распайки, как описано в Приложении B, «Методы пайки». (Обратите внимание, что подходящее гнездо для ПЗУ<sup>11</sup>(Она относительно дешева — менее 20 долларов США, — хотя для ее установки требуется твердая рука и какое-либо оптическое увеличительное устройство.)

---

<sup>10</sup>Visor описал свой опыт использования метода слежения с помощью логического анализатора на сайте <http://www.xboxhacker.net/visor/aXventure1.txt>

<sup>11</sup>Emulation Technologies (<http://www.emulation.com>) выпускает широкий ассортимент недорогих сокетов для таких целей. Конкретная модель для Xbox — S-TS-SM-040-A.

После того, как ПЗУ извлечено, а его штырьки очищены и осмотрены, его содержимое можно считывать в считывателе ПЗУ. Конечно, считыватели ПЗУ можно купить, но всегда полезно собрать свой собственный. Вы можете немного почитать о программаторах ПЗУ, которые я построил, на моем сайте, <http://www.xenatera.com/bunnie>. Мой оригинальный Flashburner<sup>12</sup>программатор — простое устройство, которое легче понять и построить, чем его вторая версия<sup>13</sup>, но он менее мощный.

Однако, если ваша цель — считывать ПЗУ как можно быстрее, просто купите считыватель ПЗУ. Хороший считыватель ПЗУ — это необходимый инструмент в наборе инструментов любого серьезного хакера оборудования. Needham's Electronics (<http://www.needhams.com>) выпускает отличную линейку программаторов/считывателей ПЗУ, которые подходят для широкого диапазона бюджетов.

## Встреча с Microsoft

После извлечения содержимого ROM следующим шагом будет предоставление его содержимого коллегам-хакерам для анализа. Или это так? В течение двенадцати часов после публикации содержимого ROM на моем сайте мне позвонил инженер из Microsoft и вежливо попросил удалить его запицценный авторским правом контент с моего сайта. Конечно, я немедленно удалил его контент со своего сайта; я должен был знать лучше, прежде чем публиковать его в первую очередь.

Это первое столкновение с Microsoft стало отрезвляющим предупреждением о том, что обратная разработка Xbox не будет похожа на любой другой проект по обратной разработке бытовой техники. Существуют законы, защищающие аспекты обратной разработки, и обширный корпус законов об авторском праве, защищающих владельца интеллектуальной собственности (ИС). Совместная обратная разработка Xbox при соблюдении прав Microsoft — это юридическое минное поле.

С одной стороны, Microsoft должна иметь возможность инвестировать в продукт и рисковать в надежде на прибыль. Однако прибыльность не гарантируется законом. Например, продажа консолей с огромным убытком, как это сделала Microsoft, в надежде продать программное обеспечение, чтобы компенсировать разницу (как «лидер убытков»), является рискованным предложением, и нет никаких гарантий по закону, что Microsoft в конечном итоге выйдет

<sup>12</sup><http://www.xenatera.com/bunnie/proj/flashburn/fb.html>

<sup>13</sup><http://www.xenatera.com/bunnie/proj/fb2/>

---

### Взлом Xbox: Введение в обратную разработку

победителем. С другой стороны, мы, как хакеры, имеем право возиться («честно использовать») с оборудованием, купленным на наши собственные с трудом заработанные деньги, и если Microsoft хочет по сути продавать нам ПК с огромной скидкой, это нормально. Купим ли мы достаточно игр (около десяти или больше), чтобы компенсировать убытки Microsoft на Xbox, полностью зависит от деловой и маркетинговой стратегии Microsoft.

На мой взгляд, большое соотношение убытков к доходам Microsoft является своего рода аномалией в этой отрасли. Sony и Nintendo примерно достигают безубыточности по стоимости своего консольного оборудования. Кроме того, операторы сотовой связи часто продают свои телефоны с убытком, сопоставимым с убытком Xbox, но требуют от абонента заключить договор, чтобы гарантировать возмещение стоимости телефона; разрыв договора подразумевает плату за расторжение. Возможно, это является отражением уверенности Microsoft в бизнес-модели Xbox Live.

Где-то посередине всего этого находится взаимодействие криптографических механизмов защиты авторских прав и права на добросовестное использование. Оказывается, Xbox широко использует криптографию для обеспечения защиты от копирования, а также политики использования консоли, что приводит нас к Закону об авторском праве в цифровую эпоху 1998 года (DMCA), относительно новому, непроверенному своду законов. С небольшим количеством устоявшихся судебных прецедентов и большим количеством серых зон между буквами закона вы как хакер должны оценить потенциальную ответственность, с которой вы можете столкнуться. Глава 12, «Caveat Hacker», более подробно рассматривает правовые вопросы взлома в новом тысячелетии.

## Анализ содержимого ПЗУ

Получив отказ от Microsoft, но сохранив содержимое ПЗУ, я приступил к анализу содержимого ПЗУ. Можно было бы ожидать, что загрузочное ПЗУ содержит процедуру инициализации оборудования, за которой следуют инструкции по загрузке операционной системы, а возможно, и сам код операционной системы. Но с чего начать?

Программу внутри ПЗУ можно представить как клубок пряжи: как только вы найдете начальную точку нити, это всего лишь вопрос времени и упорства, когда вы размотаете клубок пряжи до его сердцевины.

К счастью, начальная точка процессора Pentium Xbox очень хорошо документирована Intel. При включении питания процессор начинает

выполнять код в специальном жестко запрограммированном месте, называемом вектором сброса. Этот вектор сброса находится по адресу 0xFFFF.FFF0, около верхней части памяти. Давайте посмотрим на данные, содержащиеся в этом месте (в шестнадцатеричном формате):

```
0xFFFF.FFF0EBC6 8BFF 1800 D8FF FFFF 80C2 04B0 02EE
```

```
// процедура инициализации ключа unsigned char
K[256]; // 0xFFFFC80 во флэш-памяти unsigned
char S[256]; // 0x10000 в SDRAM

для( я = 0; я < 256; я++ ) {
    S[i] = i; } j = 0; for( i = 0; i < 256; i++ ) { // RC-4 сделает j = (j + K[i] + S[i]) % 256 j = (j +
    K[i] + S[j]) % 256; // поменять местами S[i],
    S[j] temp = S[i]; S[i] = S[j];
    S[j] = temp;
}

// процедура расшифровки unsigned char cipherText[16384];
// 0xFFFFFA000 во FLASH unsigned char plainText[16384]; //
0x400000 в SDRAM

for( index = 0x4000, i = 0, k = 0; index > 0; index-- ) { // версия xbox t = (S[i] ^ cipherText[k]) % 256; plainText[k]
= t;

// swap( S[i], S[t] ); temp
= S[i]; S[i] = S[t]; S[t] =
temp;

я = (я + 1) % 256; k++;
}
```

---

**Листинг 6-1:** Декомпиляция фиктивного шифра, найденного во FLASH ROM.

Первые два байта, EBC6, являются инструкцией перехода в местоположение 0xFFFF.FFB8. Первый байт, EB, является конкретным кодом операции для «перехода, короткого, относительного, смещения относительно следующей инструкции»; второй байт, C6, является 8-битным смещением со знаком перехода. Аругими словами, первое, что делает процессор, — это переход в другое местоположение — то, что делает каждая загрузочная программа, поскольку у вас есть только 16 байт взлетно-посадочной полосы в векторе сброса, прежде чем вы упадете с верхнего конца

---

### Взлом Xbox: Введение в обратную разработку

памяти. Поскольку этот код типичен для вектора сброса, можно перепечатать его здесь в образовательных целях.

Следующий фрагмент кода — это фрагмент, который инициализирует GDT процессора.

(Таблица глобальных дескрипторов) и состояние IDT (Таблица дескрипторов прерываний). GDT и IDT настраивают схему управления памятью процессора и схему обработки прерываний. Вам не нужно точно понимать, что делают эти регистры, но если вам интересно, в руководстве Intel «IA-32 Intel Architecture Software Developer's Manual, Volume 3: System Programming Guide» подробно объясняется функция этих регистров. Это руководство доступно на веб-сайте разработчиков Intel, <http://developer.intel.com>.

После настройки этих регистров процессор переходит в защищенный режим и переходит в 0xFFFF.FE00 — область ровно на 512 байт ниже верхней части памяти — и вот тут-то все и начинает становиться интересным. После короткого фрагмента кода, который настраивает регистры сегментов, выполняется программа, называемая интерпретатором таблиц Jam (также известная как интерпретатор X-Code в сообществе Xbox). Таблица Jam — это отраслевой жаргон для таблицы значений, которая содержит коды операций для чтения, записи и простых операций принятия решений, используемых в контексте инициализации оборудования. Для инициализации типичного ПК требуются сотни операций, и таблицы Jam помогают справиться с этой сложностью, не раздувая кодовую базу инициализации ядра. Использование таблиц Jam также помогает сделать инициализацию более гибкой и способной работать с настраиваемыми пользователем параметрами оборудования, такими как тип и объем установленной памяти. В случае Xbox интерпретатор таблиц Jam начинает извлекать коды операций таблицы Jam из местоположения около нижней части FLASH ROM. (Помните, что коды операций, реализованные интерпретатором Jam Table, весьма эффективны; с помощью кодов операций Jam Table можно записывать и считывать данные из любого места на Xbox.)

После того, как терминальный операционный код выполнен интерпретатором jam table, процессор очищает MTRR (регистры диапазона типов памяти, используемые для объявления кэшируемости различных областей памяти) и начинает расшифровывать область памяти размером 16 КБ, начиная с 0xFFFF.A000. Шифр, используемый для расшифровки этой области памяти, очень похож на RC-4, с некоторыми тонкими отличиями. Листинг 6-1 показывает шифр, реверсивно преобразованный в код C с помощью инструмента IDA Pro корпорации Data Rescue (подробнее об этом инструменте в следующих нескольких главах).

Данные, расшифрованные этим шифром, на самом деле представляют собой блок кода, который выполняется в конце процесса расшифровки, но здесь что-то идет не так.

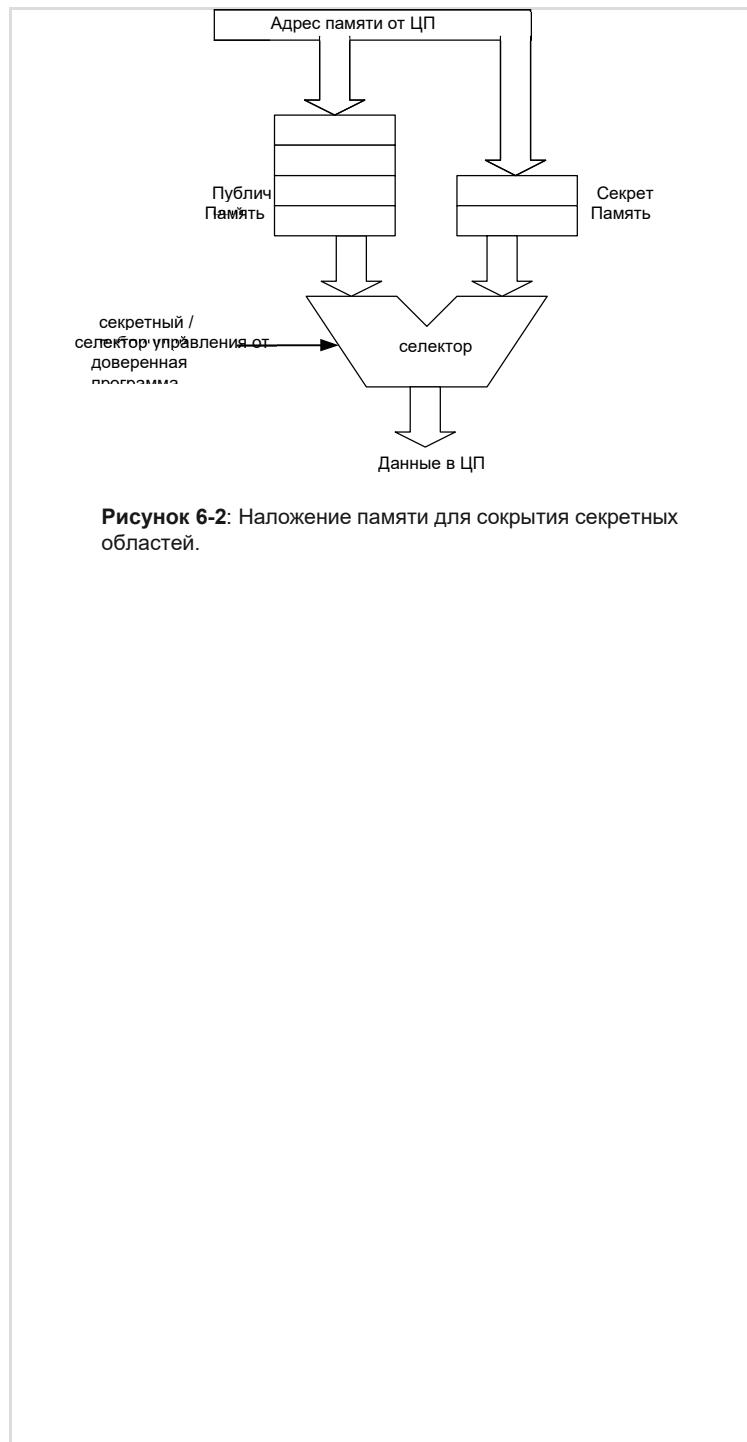
Расшифрованный код — мусор. Он не работает.

## Трюки декодирования адреса памяти

Существует ряд приемов, которые можно использовать для того, чтобы заставить области памяти выглядеть иначе, чем их физическое представление. Два приема, имеющие отношение к анализу последовательности загрузки Xbox, — это алиасинг и наложение.

Места памяти являются псевдонимами, когда два адреса ссылаются на одно и то же место памяти, что обычно достигается путем игнорирования нескольких бит адреса. Чтобы проиллюстрировать псевдонимизацию, рассмотрим систему, использующую 3-битный адрес. Существует только  $2^3 = 8$  уникальных адресов в 3-битной системе: 000, 001, 010, 011, 100, 101, 110 и 111. Теперь предположим, что у вас есть память с четырьмя адресами; для различия каждого из четырех адресов требуется всего два бита: 00, 01, 10 и 11. Если вы используете нашу 3-битную схему адресации для обращения к этой четырехадресной памяти, один из адресных битов должен быть проигнорирован. Если игнорируется старший бит, то адреса 000 и 100 будут оба отображаться в адрес 00 в памяти. Другими словами, адрес 00 сопоставляется с адресами 000 и 100.

Наложение памяти — это метод, при котором внеполосная информация используется для выбора между различными банками памяти. Предположим, что мы хотим иметь банк секретной памяти. Для этого мы вставляем селектор между нашими открытыми и секретными блоками памяти и ЦП. Этот селектор может выбирать, предоставлять ли ЦП данные либо из секретной памяти, либо из открытой памяти, как показано на рисунке 6-2. В результате программа, которая управляет селектором адреса, также контролирует, кто имеет доступ к секретному блоку. Если компьютер начинает выполнять код, расположенный в секретном банке памяти, программа в области секретного кода может использовать этот механизм, чтобы скрыть себя, установив селектор так, чтобы он указывал на открытую память, перед запуском программ, расположенных в открытой памяти.



**Рисунок 6-2:** Наложение памяти для скрытия секретных областей.

---

## Взлом Xbox: Введение в обратную разработку

Более того, коды операций `jam table`, похоже, повреждены. Это явление было подтверждено другими хакерами, работающими над проблемой, таким образом, исключая ошибку перевода кода. Очевидно, что в Xbox есть нечто большее, чем кажется на первый взгляд.

Теории и слухи начали появляться, чтобы объяснить это странное поведение. Некоторые из популярных теорий включали:

- **Шифрование адресов и/или линий данных.** Где-то адресные или информационные линии инвертировались или переставлялись с помощью некоторой функции отображения 1:1. Функция скремблирования могла быть запрограммирована в чипсете как часть процедуры инициализации, так что начальный загрузочный блок считывался как открытый текст, в то время как остальные данные были скремблированы.
- **Вторичный криптовпроцессор.** На самом деле инициализацией Xbox занимался другой процессор, а загрузочный код в ПЗУ оказался поддельным.
- **Загрузочный код, содержащийся в процессоре.** Процессор - это на самом деле инициализируется фрагментом кода, находящимся на кристалле процессора, а загрузочный код в ПЗУ является поддельным.
- **Загрузочный код, содержащийся в чипсете.** Процессор функционирует идентично стандартному Pentium, но чипсет содержит загрузочный код, который переопределяет фиктивный код внутри ПЗУ. Для почти всех этих теорий единственный способ доказать или опровергнуть их — провести эксперименты на оборудовании. Например, чтобы убедиться, что SMC (System Management Controller, 8-битный автономный процессор, который всегда включен, когда подключен Xbox) не играет никакой роли в безопасной последовательности загрузки машины, хакеры перехватили следы сигналов на всех контактах SMC и проанализировали их на предмет ожидаемой последовательности событий, если бы SMC играл решающую роль в инициализации машины.

Важнейшим наблюдением коллеги-хакера было то, что Xbox загружался идеально, даже когда был изменен код вектора сброса в `0xFFFF.FFF0`. Можно было бы ожидать, что если первая инструкция, выполняемая процессором в `0xFFFF.FFF0`, была повреждена, то машина зависла бы. Вместо этого машина работала безупречно. Это

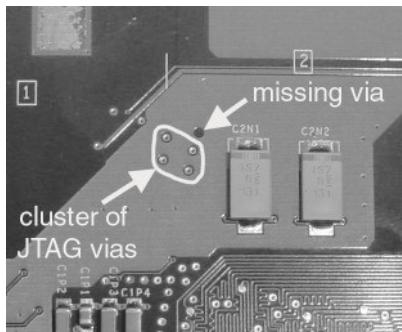
наблюдение было проверено серией экспериментов, в которых различные части FLASH ROM были намеренно повреждены. Результаты показали, что повреждение удивительно больших областей FLASH ROM не оказало никакого влияния на загрузку Xbox. В частности, всю последовательность инициализации загрузки от 0xFFFF.FE00 до 0xFFFF.FFFF можно было бы обнулить, и Xbox нормально загрузился бы.

Это открытие само по себе решительно подтверждало теорию фиктивного загрузочного блока во FLASH ROM. Однако оставался вопрос о том, где хранился настоящий загрузочный код. Было три варианта: во вторичном криптовпроцессоре, в процессоре и в чипсете. Теория вторичного криптовпроцессора была отвергнута на основании того, что на материнской плате не было чипов, которые были бы достаточно мощными или достаточно активными во время загрузки, чтобы играть роль криптовпроцессора. Хранение загрузочного блока в процессоре также считалось менее вероятным вариантом, чем хранение загрузочного блока в чипсете.

Обоснование этого анализа основано на экономике создания чипов. Процессор Pentium III очень сложен, содержит множество блоков ручной работы, и модификация кремния для включения безопасного загрузочного блока потребовала бы значительных инженерных ресурсов, а также первоначальных инвестиций в размере около четверти миллиона долларов только для масок, необходимых для производства специального кремния. Кроме того, ходили слухи, что Microsoft изначально выбрала процессор AMD для Xbox и в последнюю минуту переключилась на Intel. Если бы специальные блоки были интегрированы в ядро процессора, Microsoft не смогла бы так легко переключаться между поставщиками ЦП. С другой стороны, чипсеты nVidia разработаны модульно с использованием кремниевых компиляторов, поэтому технически проще добавлять бородавки, такие как безопасный загрузочный блок. Кроме того, чипсет в Xbox представляет собой индивидуальную сборку nForce, сделанную специально для Microsoft, специально адаптированную для системной шины Intel (FSB). В результате стоимость добавления безопасного загрузочного блока можно было бы включить в инженерные ресурсы и наборы масок, уже выделенные для такого проекта.

Действуя в соответствии с теорией, что настоящий загрузочный код находится в секретном ПЗУ в чипсете, оставшиеся проблемы заключались в том, чтобы определить, в каком чипе (северный мост или южный мост) хранится код, и как извлечь этот секретный ПЗУ. Было представлено несколько стратегий извлечения секретного ПЗУ:

- Используйте функцию JTAG «границочного сканирования» на Pentium, чтобы попытаться захватить начальный код загрузки. JTAG — это диагностическая шина, которая позволяет считывать и устанавливать состояние каждого вывода на чипе через специальный последовательный порт. Это очень мощный и универсальный инструмент отладки.
- Проверьте FSB (переднюю шину) процессора чтобы попытаться захватить загрузочный код при его поступлении в процессор.



**Рисунок 6-3:** Отсутствует JTAG via. Обратите внимание, что заполненная медная область (более светлая область) имеет отверстие там, где раньше было сквозное отверстие. Это результат изменения в последнюю минуту в макете платы без пересчета областей заполнения.

- Установить анализатор памяти попытаться захватить расшифрованный поток данных, записываемый в память.
- Использовать микроскопию для считывания содержимого защищенной загрузочной области с поверхности чипа.
- Проверьте автобус между Южным мостом и Северный мостом чипы, чтобы попытаться захватить код загрузки, отправляемый процессору чипсетом. Это сработает только в том случае, если данные загрузки хранятся где-то в чипе южного моста.

Ни одна из этих теорий не была тривиальной для проверки, поэтому попытки взлома Xbox постепенно сопали на нет, поскольку разочарованные хакеры отказались от попыток криптоанализа образа FLASH ROM. Я был бы одним из тех, кто бросил это дело (в конце концов, мне нужно было закончить и написать докторскую диссертацию всего за несколько месяцев), если бы не сообщество решительных хакеров, подбадривавших меня. Во время рождественских каникул в декабре 2001 года я поддерживал связь со

своими друзьями-хакерами через каналы IRC и веб-форумы. Хакеры со всего мира и из всех слоев общества заполонили канал взлома Xbox IRC, и мне нравилось учиться у них и общаться с ними об их разнообразном опыте, как техническом, так и личном.

Несмотря на то, что я был полон решимости посвятить весь январь написанию своей докторской диссертации<sup>14</sup> и избегая взлома Xbox, я все еще был привлечен интригующе сложной безопасностью, используемой Xbox. Со временем потребность в специалисте по оборудованию, который присоединится к небольшой группе хардкорных хакеров, тусующихся на канале IRC, становилась все более очевидной. К концу января отчеты, которые я слышал о схеме безопасности Xbox, стали слишком интересными, чтобы их игнорировать.

Я купил второй Xbox и начал снимать все его ключевые части с помощью термофена. Разборка Xbox преследовала множество целей. Во-первых, снятие чипов обнажило все дорожки и соединения на Xbox, так что я мог легко отслеживать соединения между чипами, используя режим проверки целостности цепи на моем мультиметре. Во-вторых, я смог поместить все интересные чипы в банну с горячей кислотой и снять их пластиковую оболочку для анализа под микроскопом. Наконец, покупка Xbox и полная его разборка дали мне своего рода душевное спокойствие, когда дело доходит до исследования и модификации работающего Xbox. (Обратная разработка похожа на садоводство. Посадка сада гораздо сложнее, если вы пытаетесь сохранить руки и колени чистыми, так что вы можете смириться с этим и начать кататься в грязи.)

Результаты разборки Xbox выявили некоторые меры, которые Microsoft предприняла для защиты устройства от аппаратных хакеров. Например, я сначала проверил соединения JTAG на процессоре Pentium. Все сигналы JTAG были удобно направлены на набор легко подключаемых резисторов рядом с процессором, за исключением одного, сигнала TRST#. TRST# играет важную роль в инициализации интерфейса JTAG. Интересно, что TRST# был привязан к внутренней заземляющей плоскости в труднодоступной области, навсегда деактивируя механизм JTAG. Дальнейший осмотр материнской платы Xbox выявил намеки на то, что сигнал TRST# был удален в последнюю минуту. (Самым большим намеком на отсутствие переходного отверстия является отверстие в дорожке питания,

---

<sup>14</sup>Если вас интересуют архитектура суперкомпьютеров, миграция данных и потоков, отказоустойчивость, высокоскоростные сети с малой задержкой или многопоточные машины, ознакомьтесь с моей диссертацией по адресу <http://www.xenatera.com/bunnie/phdthesis.pdf>.

---

### Взлом Xbox: Введение в обратную разработку

идеально подходящее по размеру для переходного отверстия рядом с группой переходных отверстий, предназначенных для сигналов JTAG, как показано на рисунке 6-3.)

Еще одним ударом по подходу JTAG для извлечения секретного ПЗУ является тот факт, что коды сканирования JTAG от Intel являются проприетарными. Обратное проектирование кодов до уровня, на котором я мог бы использовать их для извлечения секретных загрузочных данных, само по себе было крупным проектом.

Отказавшись от подхода JTAG, следующим методом извлечения секретного ПЗУ было снятие упаковки с ЦП, ГП и MCPX и осмотр голого кристалла с помощью микроскопа и поиск любых потенциальных структур ПЗУ. Удаление упаковки или «декапсуляция» достигалось путем погружения чипов в дымящуюся горячую серную кислоту. (Я не рекомендую пробовать этот подход дома; однажды я пролил токсичный, едкий раствор на себя, и, к счастью, вместо моей кожи сгорело мое защитное снаряжение. Дымящая серная кислота поглощает органические материалы быстрее, чем горячее пламя.) Дымящая азотная кислота, также очень токсичная и опасная, также может быть использована. Хотя я сам не пробовал, отчеты показывают, что дымящая азотная кислота более эффективна для удаления эпоксидной инкапсуляции, особенно в ситуациях, когда желательно выборочное удаление корпуса.

Ручной подход к проверке с использованием традиционного микроскопа видимого света давал некоторую надежду; однако, эта техника ограничена физикой света. Даже лучшая технология видимой микроскопии не может разрешить транзистор 150 нм, поскольку самая короткая длина волн света составляет 450 нм (соответствует синему цвету). Я надеялся, что секретный код будет храниться на чипах с использованием традиционной структуры массива ПЗУ, где металлические линии определяют 1 или 0, выгравленные на верхних металлических слоях, которые можно идентифицировать с помощью оптического микроскопа. Использование жестко защищенной структуры ПЗУ мотивировано стоимостью: FLASH-ПЗУ и PROM на основе предохранителей требуют дополнительных этапов обработки и производства, которые могут значительно увеличить стоимость системы, в то время как использование верхних металлических слоев мотивировано управлением рисками со стороны проектировщика. Верхние металлические слои являются самыми грубыми слоями (настолько грубыми, что оптический микроскоп может разрешить их), и, таким образом, являются самыми дешевыми слоями для замены в случае ошибки в коде ПЗУ. Кроме того, во время первоначального приведения в порядок верхний слой легче всего разрезать и перемкнуть с помощью машины для ремонта чипов, известной как

машиной FIB (фокусированный ионный луч). К сожалению, быстрый взгляд на чип под микроскопом не выявил таких структур.

На этом этапе единственным оставшимся вариантом извлечения секретного ПЗУ было зондирование работающего оборудования Xbox в попытке захватить код во время загрузки в процессор Xbox. Подслушивание кода выше по потоку от чипа южного моста и FLASH ROM означало зондирование либо шины Front Side, либо шины Northbridge-Southbridge, либо основной шины памяти. Мы обсудим компромиссы выполнения этих подходов зондирования в Главе 8 после краткого введения в основные концепции безопасности в следующей главе.

## CHAPTER7. Краткий курс по безопасности

Взлом Xbox требует взлома безопасности в дополнение к взлому оборудования и прошивки, как вы обнаружили в предыдущей главе. Мы рассмотрим некоторые возможные мотивы добавления сложной безопасности к чему-то столь обыденному, как игровая машина, а затем погрузимся в основные принципы и алгоритмы, необходимые для понимания и оценки механизмов безопасности Xbox.

### Кому вообще нужна безопасность?

Игровая приставка для большинства людей — это игрушка: это недорогая потребительская электроника. Почему Microsoft так старалась защитить свою систему? В игре взлома безопасности довольно часто понимание мотивов хакера помогает найти слабые места, которые можно использовать.

Криптография — это не безопасность. Криптография — это средство для достижения цели безопасности, но настоящая безопасность включает в себя всю архитектуру системы, включая конечных пользователей. Как сказал Кевин Митник в недавнем интервью Slashdot, «... безопасность — это не продукт, который можно купить в готовом виде, а состоит из политик, людей, процессов и технологий». <sup>1</sup> Я считаю, что безопасность — это фундаментально социальная концепция. На практике вы можете открыть окна и оставить входную дверь запертой, и люди не будут просто входить через окно или взламывать дверной замок, хотя и то, и другое — относительно простые задачи. Запертые двери и открытые окна работают, потому что запертая дверь — это в основном символическая мера; она заставляет злоумышленника совершить осознанный акт нарушения, чтобы войти в дом, и одного этого достаточно, чтобы отделить преступников от добродорядочных людей. Консоль Sony Playstation

---

<sup>1</sup> <http://interviews.slashdot.org/article.pl?sid=03/02/04/2233250&mode=nocomment&tid=103&tid=123&tid=172>

есть хороший пример безопасности замка входной двери. Механизм, используемый для защиты от копирования их игр, прост, не требует криптографии и легко обходит с помощью легко устанавливаемых недорогих аппаратных модификаций. Несмотря на это, цифры продаж

указывают на то, что покупка игр для Playstation не вышла из моды; замок входной двери работает.

Microsoft использует разновидность защиты типа замка на входной двери. Видеоигры для Xbox распространяются с использованием (пока) некопируемого формата DVD-9, одностороннего двухслойного формата носителя. С другой стороны, записываемые пользователем DVD-диски всегда имеют формат DVD-5, односторонний однослойный формат носителя. Пишущие устройства с поддержкой DVD-9 вряд ли появятся в ближайшее время из-за сложности создания записывающей системы, способной записывать один слой, не нарушая целостности другого слоя. Таким образом, распределая данные безопасности между двумя слоями диска DVD-9 и требуя, чтобы исполняемый файл игры был с носителя DVD-9, Microsoft имеет довольно эффективный замок на входной двери для своих видеоигр. Разумно требуя формат DVD-9, Microsoft фактически вынудила любого потенциального копировщика игр перейти в область, где необходима некоторая модификация оборудования.

Зачем же тогда Microsoft рискует инвестировать в такую сложную схему безопасности на Xbox? Действительно ли главная мотивация Microsoft — подавить пиратство? Вполне возможно, что на самом деле основная причина остальной части системы безопасности Xbox — запищенных загрузочных секторов, подписанных исполняемых файлов, доверительных отношений и зашифрованных/аутентифицированных сетевых протоколов — заключается не в мерах по борьбе с пиратством.

Одной из возможных причин для всей этой безопасности является предотвращение использования консоли Xbox для любых целей, кроме игр. Консоль Xbox находится в уникальном положении, поскольку является почти 100-процентным стандартным ПК. В отличие от Gamecube и Playstation2, существует огромное количество программного обеспечения, которое, кажется, должно работать только на Xbox, при условии правильного программирования BIOS. Что еще хуже, Microsoft теряет гораздо больше денег на своем консольном оборудовании, чем ее конкуренты. Некоторые подсчитывают, что ее потери могут достигать 200 долларов за консоль, если предположить, что последняя розничная цена составляет 199 долларов. Следовательно, в интересах Microsoft попытаться убедиться, что она не продает субсидированные коробки GNU/Linux. Однако даже это, вероятно, не является главной целью Microsoft. 64 МБ оперативной памяти Xbox, отсутствие клавиатуры или мыши из коробки и довольно медленный процессор по сегодняшним меркам делают его менее привлекательным, чем, например, Microtel PC за 200 долларов, доступный в Walmart в конце 2002 года. Кроме того, у Microsoft большие карманы: если Xbox наберет популярность на рынке и превзойдет Sony Playstation2, Microsoft придется понести только несколько миллиардов долларов первоначальных потерь — относительно немного по сравнению с примерно 40 миллиардами долларов наличных, на которых она сидит.

Таким образом, вполне возможно, что важнейшая миссия безопасности Xbox заключается не в предотвращении альтернативного использования консоли или сдерживания пиратства.

Возможно, настоящая причина сложной безопасности Xbox — обеспечение успеха Xbox Live, онлайн-игрового сервиса Microsoft. Маркетинговая шумиха и PR-заявления Microsoft указывают на то, что она делает ставку на успех

Xbox Live для стимулирования продаж оборудования. Кроме того, Xbox Live — это подписной сервис, и через год после его запуска пользователи должны будут платить ежемесячную плату. Если Microsoft сможет подсадить своих подписчиков на Xbox Live, то внезапно бизнес Xbox станет выглядеть весьма прибыльным, даже если существенная сумма денег будет потеряна авансом на оборудование. Конечно, весь фокус в том, чтобы подсадить пользователей Xbox на Xbox Live. Xbox Live, который называют «Диснейлендом онлайн-игр», призван обеспечить хорошо реализованный и честный игровой опыт. Центральным моментом ценностного предложения Xbox Live является отсутствие читеров. Чтобы гарантировать, что никто не мошенничает, пользователи должны быть вынуждены проходить аутентификацию в реестре, поддерживаемом Xbox Live, а их игровое состояние должно быть защищено и не поддаваться изменению. Кроме того, игровое программное обеспечение должно быть непатченным. Еще более важным является тот факт, что вам нужно всего несколько читеров, чтобы испортить игровой опыт целой пользовательской базы. Внезапно внешние средства безопасности, предлагаемые форматом DVD-9, кажутся недостаточными. Шансы против вас, если вы делаете ставку на успех бизнеса на мораль и честь пользователей из миллионов двадцатилетних хардкорных мужчин-геймеров с разумным количеством компьютерных навыков, распределенных по всей сети. Аппаратное обеспечение должно быть надежным, сетевые соединения безопасными, а исполняемые файлы подписанными и запечатанными.

Утверждение о том, что оборудование должно быть надежным, заслуживает повторения. Учитывая ненадежную базу пользователей, единственный способ установить доверительные отношения с клиентами — это если зерно доверия существует в каждой части оборудования. Следовательно, Microsoft должна включить в каждый клиент часть оборудования, защищенного от несанкционированного доступа, которая позволяет проводить некоторую аттестацию. Аттестация — это возможность доказать, что некоторая часть данных, например, личность игрока или состояние игры, на самом деле сгенерирована незапятнанным программным обеспечением и оборудованием. Защищенное от несанкционированного доступа оборудование не должно реализовывать функцию аттестации напрямую, но оно должно, по крайней мере, гарантировать, что система находится в надежном состоянии перед аттестацией.

Существует множество способов убедиться в надежности оборудования. Метод грубой силы заключается в том, чтобы сделать все оборудование физически защищенным. Банкоматы являются яркими примерами оборудования, которое физически защищено. Запечатанное в толстый лист металла и покрытое датчиками вторжения, трудно физически проникнуть и модифицировать оборудование банкомата. Тем не менее, хотя это эффективно, это непрактичное и дорогое решение для игровой консоли.

Более экономичным решением является использование небольшого доверенного защищенного от несанкционированного доступа оборудования, которое может выполнять «измерения» остальной части системы. Такого рода измерения обычно выполняются с использованием криптографической хэш-функции. Если все эти измерения доверия соответствуют ожидаемым значениям, то можно сделать вывод, что вся система заслуживает доверия.

Я говорю «может быть», потому что эта схема все еще уязвима для атак типа «человек посередине», когда хакер отправляет поддельные действительные данные в ответ на запрос измерения. Атаки типа «человек посередине» относятся к общему классу атак, когда злоумышленник может свободно изменять и контролировать информацию, передаваемую между двумя сторонами. Из-за слабости типа «человек посередине» не имеет смысла использовать чрезвычайно сложный защищенный от несанкционированного доступа модуль для проведения системных измерений. Одного кремниевого чипа в корпусе, вероятно, будет достаточно, поскольку обычно проще перехватить и подделать данные измерений, проходящие через печатную плату, чем проникнуть в эпоксидный корпус чипа и изменить его схему.

Система измерения доверия может быть реализована с использованием подхода «измерение один раз». Начиная с последовательности холодной загрузки процессора, каждый фрагмент кода измеряется на предмет доверия перед выполнением. Если процессор никогда не выполняет ненадежный код, то почему же тогда не доверять? Для этой схемы требуется очень простой защищенный от несанкционированного доступа аппаратный модуль — запечатанное от несанкционированного доступа ПЗУ, в котором хранится код холодной загрузки, «семя» доверия. Тип криптографии, используемый для процесса измерения и проверки, обычно представляет собой комбинацию хэшей и криптографии с открытым ключом. Криптография с открытым ключом предпочтительна для этого приложения, поскольку закрытый ключ, необходимый для генерации допустимого сегмента кода, является секретом, который хранится только поставщиком оборудования. Опять же, эта схема уязвима для многих видов атак типа «человек посередине», а также для чисто криптографических атак и атак на реализацию системы.

## Краткий курс по криптографии

**шифр(н):** 1 а: НОЛЬ б: тот, который не имеет веса, ценности или влияния: НЕСУЩНОСТЬ. 2 а: метод преобразования текста с целью скрытия его смысла — сравните с КОДОМ<sup>15</sup>

Шифры сами по себе не обеспечивают безопасности. Точнее, шифры обеспечивают безопасность только в том случае, если ключ безопасен, если алгоритм силен и если в системе нет бэкдоров. Если кто-то вречт вам CD-ROM, зашифрованный сильным шифром, и запрёт вас в мягкой комнате с суперкомпьютером, солнце, вероятно, станет сверхновой, прежде чем вы сможете расшифровать CD-ROM. С другой стороны, если бы вы могли наблюдать и исследовать машину, пока она работает над шифрованием CD-ROM, шифрование спорно. Вы могли бы получить ключ шифрования, подслушав клавиатуру. Или вы могли бы сбросить содержимое памяти компьютера и получить открытый текст, не зная ключа.

Ситуация с Xbox похожа на последнюю. В конечном счете, Xbox должен получить доступ и запустить программы, представленные ему на допустимых дисках. Более того, процессор Pentium, используемый в Xbox, не может отличить авторизованную инструкцию от неавторизованной. Наконец, пользователь имеет полный доступ к проверке и изменению оборудования Xbox. Таким образом, даже если Xbox использует сильные шифры, безопасность ключей остается под вопросом, и в систему могут быть лазейки.

В этом разделе будут кратко описаны типы криптографических алгоритмов, используемых в Xbox. Мы сосредоточимся на практических последствиях и вопросах реализации этих алгоритмов. Вам нужно будет понять эти алгоритмы, чтобы оценить доступные атаки на систему безопасности Xbox.

### Примечание



Я не претендую на рассмотрение теоретических аспектов криптографии; они находятся за пределами моих возможностей и выходят за рамки книги. Вместо этого я отсылаю заинтересованных читателей к превосходной книге Брюса Шнайера «Прикладная криптография» (John Wiley & Sons); большая часть моих знаний в области криптографии взята из этой книги. Читатели, которые уже знакомы с криптографией, могут пробежать или пропустить оставшуюся часть этой главы.

## Классы криптографических алгоритмов

<sup>15</sup>Словарь Merriam-Webster OnLine ([www.webster.com](http://www.webster.com)).

В Xbox используется несколько важных классов криптографических алгоритмов. Это:

- Хэши
- Симметричные шифры
- Шифры с открытым ключом

*Хэши*Существуют несколько разновидностей. Хеши криптографического типа используются для суммирования или «переваривания» большого количества данных. Сводка представляет собой число фиксированной длины, обычно около 100–200 бит, в то время как исходные данные могут быть практически любого размера. Самым важным свойством хеша является то, что это одностороннее вычисление. Другими словами, вычислить хеш легко, но очень сложно (см. врезку «Очень сложные проблемы», чтобы понять, что именно это означает) вывести последовательности данных, хеш-дайджесты которых идентичны, или определить что-либо об исходных данных из хеша.

Стойкость хеша к обнаружению двух последовательностей данных, которые хэшируются с одинаковым значением, называется его устойчивостью к коллизиям, и в целом, хороший  $n$ -битный хеш требует хэширования и сравнения около  $2n/2$  случайных последовательностей данных, чтобы вызвать коллизию. Поскольку хеши разработаны так, чтобы их было очень легко вычислять и они очень устойчивы к коллизиям, их часто используют для определения того, был ли изменен бит в больших областях защищенных данных. Для многих приложений достаточно включить только зашифрованный хеш сообщения вместо того, чтобы тратить вычислительные усилия на шифрование всего сообщения.

*Симметричные шифры*алгоритмы, которые имеют ключи шифрования и дешифрования, которые можно легко вывести друг из друга. В большинстве случаев ключи шифрования и дешифрования одинаковы. Симметричные шифры используют функцию смешивания для объединения ключевого расписания с данными, обработанными некоторой криптографической функцией. Это смешивание может повторяться несколько раз для одного блока данных, как в блочном шифре, или может происходить один раз, как в потоковом шифре. Все основные функции в симметричном шифре вычислительно просты, поэтому симметричные шифры являются предпочтительным методом для шифрования больших объемов данных.

Типичными примерами функций смешивания являются XOR, модульные сложения и модульные умножения. Простейшая функция XOR обладает тем свойством, что любое число XOR само по себе равно нулю. Операция XOR часто обозначается символом  $\oplus$ . Операция XOR также обладает всеми обычными свойствами арифметики (коммутативность,

ассоциативность, дистрибутивность и т. д.), поэтому  $(A \oplus B) \oplus B = A \oplus (B \oplus B) = A \oplus 0 = A$

Таким образом, если бы A было сообщением, а B — ключом, то  $(A \oplus B)$  было бы зашифрованным текстом, а открытый текст можно было бы восстановить, просто выполнив операцию XOR с B еще раз.

Расписание ключа — это алгоритм, который берет относительно короткий ключ и расширяет его информацию на длинную серию битов. Расписание ключа используется для того, чтобы помочь распределить ключевые данные по большему блоку данных, чтобы связь между шифртекстом и ключом была скрыта.

### Очень сложные проблемы

Криптографические функции все основаны на математических алгоритмах, результаты которых легко вычислить, если заданы все операнды, но операнды которых очень сложно вычислить, если задан только результат. Безопасность криптографической функции заключается именно в сложности вычисления этих операндов, если заданы только результаты. Давайте на минутку остановимся и рассмотрим, что значит быть очень сложным.

Рассмотрим симметричный шифр AES. Он использует 128-битный ключ, и на данный момент он устойчив ко всем известным аналитическим криптографическим атакам, таким как дифференциальный и линейный криптоанализ. Когда я говорю, что он устойчив к анализу X, я имею в виду, что для восстановления ключа или открытого текста с помощью перебора потребуется как минимум столько же операций, сколько и при использовании анализа X. Перебор — это когда я беру очень быстрый компьютер и пробую каждый из  $2^{128}$  возможные ключи для восстановления исходных данных. Большинство криптографических алгоритмов, которые широко используются сегодня, устойчивы ко всем известным методам криптоанализа, поэтому важно понимать силу атаки методом перебора.

Как оказалось, старые алгоритмы, такие как DES, 56-битный шифр, не являются очень сложной проблемой. Достаточно легко построить машину с использованием FPGA (Field Programmable Gate Arrays), которая может взламывать ключи с экономией около  $2^{22}$  ключей/секунда/доллар ( $2^{22}$  составляет около четырех миллионов). Обратите внимание, что это число увеличивается со временем со скоростью, эквивалентной закону Мура. Сегодня, если вы готовы ждать около недели для каждого ключа, вы можете восстановить их за

(продолжение)

Типичная криптографическая функция, используемая в симметричном блочном шифре, состоит из набора тщательно разработанных замен, перестановок, сжатий и расширений. Эти функции служат для запутывания и рассеивания открытого текста. Незначительные

изменения в любой части криптографической функции обычно оказывают губокое влияние на безопасность шифра.

Тот факт, что ключи шифрования и дешифрования тесно связаны в симметричном шифре, затрудняет их использование в определенных приложениях безопасности. Например, если я хочу распространить зашифрованный документ по списку рассылки, все в списке рассылки также должны знать мой ключ шифрования, если они могут прочитать документ. Кроме того, инициирование контакта с удаленной стороной затруднено, поскольку в какой-то момент мне придется передать им ключ. Кто-то, наблюдающий за средой передачи, может украсть ключ и прочитать, подделать и изменить все последующие сообщения.

*Шифры с открытым ключом* алгоритмы, которые используют разные ключи для шифрования и дешифрования. По этой причине их также называют асимметричными шифрами. Большим преимуществом шифров с открытым ключом является то, что один из ключей может храниться в секрете. Это позволяет обмениваться данными с ненадежными пользователями, не давая ненадежному пользователю возможности подделывать или читать другой запищенный контент. Недостатком алгоритмов с открытым ключом является то, что они

о цене хорошей машины. Будем надеяться, что банки не используют DES для шифрования данных своих счетов!

Преемник DES, AES, является шифром, который может использовать 128, 192 или 256-битные ключи. Эти ключи достаточно велики, чтобы считаться неуязвимыми для атак методом подбора (т.е. очень сложной проблемы). Согласно AES Q&A, опубликованному NIST (<http://csrc.nist.gov/encryption/aes/aesfact.html>), машина достаточно мощная, чтобы восстанавливать один ключ DES в секунду методом подбора (в среднем пробуя  $2^{55}$  ключей в секунду) все равно потребовалось бы 149 триллионов лет для восстановления 128-битного ключа AES. Мой любимый анализ прочности 256-битных ключей против атак методом перебора взят из книги Брюса Шнайера «Прикладная криптография». В своей книге он использует аргумент, основанный на количестве энергии, необходимом для взлома 256-битного ключа. Оказывается, даже термодинамически идеальному компьютеру потребовалось бы в 32 раза больше энергии, чем годовая выработка нашего Солнца, чтобы просто досчитать до  $2^{192}$ , не говоря уже о том, чтобы сделать что-то полезное с этим количеством. (Я должен подчеркнуть, что все это предполагает, что наиболее эффективная атака — это грубая сила. Кто знает, может быть, кто-то обнаружит уязвимость в алгоритме, которую можно будет использовать для проведения гораздо более эффективной атаки. Постоянно изобращаются новые методы анализа, которые медленно подрывают стойкость шифра.)

С другой стороны, шифры с открытым ключом основаны на широком спектре труднообратимых математических операций, таких как умножение простых чисел и модульное возведение в степень. В результате пространство ключей для многих шифров с открытым ключом разрежено, поэтому для эквивалентной безопасности симметричного шифра требуется больше битов ключа. Например, длина ключей в шифре с открытым ключом RSA обычно составляет несколько тысяч бит.

(продолжение)

обычно требуют более сложных вычислений и, таким образом, медленнее, чем симметричные шифры. Шифры с открытым ключом также, как правило, требуют более длинных ключей для эквивалентной безопасности. В результате, если необходимо обменяться большим объемом данных, шифры с открытым ключом часто используются для шифрования ключа для симметричного шифра, который используется для шифрования большей части данных. Этот ключ симметричного шифра может быть уникальным для каждой транзакции, и поэтому его часто называют «ключом сеанса».

## SHA-1 хэш

SHA-1 — это алгоритм безопасного хэширования, рекомендованный федеральным правительством в публикации FIPS 180-1 (<http://www.itl.nist.gov/fipspubs/fip180-1.htm>). Разработанный Агентством национальной безопасности и основанный на алгоритме дайджеста сообщений MD4 Рональда Л. Ривеста, SHA-1 работает с сообщениями

## Очень сложные проблемы (продолжение)

Точная корреляция между безопасностью длин открытых ключей RSA и длинами ключей симметричного шифрования неизвестна. Безопасность RSA, как полагают, заключается в сложности факторизации произведений больших простых чисел; однако, могут быть и другие атаки на алгоритм, которые еще предстоит обнаружить. Тем не менее, эффективная сложность факторизации произведений больших простых чисел снижается не только за счет достижений в области вычислительной техники (закон Мура), но и за счет достижений в теории чисел, таких как изобретение и усовершенствование квадратичного решета и общего решета числового поля.

В августе 1999 года группа исследователей использовала решето числового поля для разложения на множители 512-битного простого числа за 7,4 календарных месяца, включая время, необходимое для настройки процесса разложения на множители.<sup>1</sup>. Кроме того, новые технологии, такие как квантовые вычисления обещают сделать возможным факторизацию простых чисел за полиномиальное время. Однако я бы не стал затаивать дыхание; все еще ведутся споры о том, можно ли построить квантовый компьютер, достаточно большой, чтобы факторизовать интересное простое число.

На сегодняшний день компания RSA Security, Inc. рекомендует использовать ключи длиной 1024 бита для большинства корпоративных целей и 2048 бит для «чрезвычайно ценных ключей»<sup>2</sup>. Брюс Шнайер во втором издании «Прикладной криптографии» подсчитал, что длина открытого ключа 2304 бита обеспечивает эквивалентную безопасность 128-битного симметричного ключа, а длина открытого ключа 1792 бита соответствует примерно 112-битному симметричному ключу.

Читая о схеме безопасности Xbox, помните об этих основных рекомендациях о том, насколько сложно взломать эти схемы безопасности с помощью методов грубой силы. Время от времени на хакерских форумах и досках объявлений появляются сообщения с вопросом: «Почему бы нам не начать распределенный поиск ключей для этих ключей?» Теперь вы знаете ответ.

---

<sup>1</sup> <http://www.rsasecurity.com/rsalabs/challenges/factoring/rsa155.html>

<sup>2</sup> <http://www.rsasecurity.com/rsalabs/faq/3-1-5.html>

---

```
void encipher(unsigned long *const v,unsigned long *const w, const
unsigned long *const k) { register unsigned long y=v[0], z=v[1],
sum=0, delta=0x9E3779B9, a=k[0], b=k[1], c=k[2], d=k[3], n=32;

пока(n->0) { сумма += дельта; у += (z << 4)+a ^
z+сумма ^ (z >> 5)+b; z += (y << 4)+c ^ y+сумма
^ (y >> 5)+d; }
```

```
w[0]=y; w[1]=z; }
```

#### Листинг 7-1: Алгоритм TEA в ANSI C<sup>16</sup>

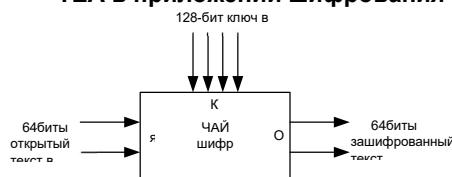
любая длина менее 264 бит, и он производит 160-битный вывод. Алгоритм хеширования SHA-1 начинается с детерминированного 160-битного начального состояния; это состояние смешивается с блоком из 512 бит данных сообщения в течение четырех раундов. Каждый раунд состоит из серии нелинейных функций, поворотов, сдвигов и операций XOR. Результат раунда используется для начального вычисления следующего раунда. В общем, необходимо сгенерировать 280 случайных сообщений, хешировать и «одновременно» сравнить, чтобы найти два сообщения с одинаковым значением хэша (т. е. коллизию хэшей). Нахождение двух случайных сообщений с одинаковым хэшем известно как «атака дня рождения», названная в честь вероятностного явления, называемого «парадоксом дня рождения»: вероятность того, что у двух человек в комнате из 23 человек одинаковый день рождения, превышает 50%. С другой стороны, необходимо сгенерировать, хешировать и сравнить 2160 случайных сообщений, чтобы найти сообщение, хеш которого будет иметь то же значение, что и определенное сообщение. Таким образом, сила хеш-функции сильно зависит от способа ее использования.

### ЧАЙ

TEA, или алгоритм крошечного шифрования, был разработан Дэвидом Уиллером и Роджером Ницхэмом в компьютерной лаборатории Кембриджского университета. (У разработчиков есть веб-страница для TEA по адресу <http://vader.brad.ac.uk/tea/tea.shtml>; большая часть представленного здесь материала взята с этой страницы.)

Как следует из названия, TEA — это компактный и быстрый алгоритм шифрования, подходящий

#### TEA в приложении шифрования



#### TEA используется как хэш-функция

<sup>16</sup>Код взят с <http://vader.brad.ac.uk/tea/source.shtml#ansi>

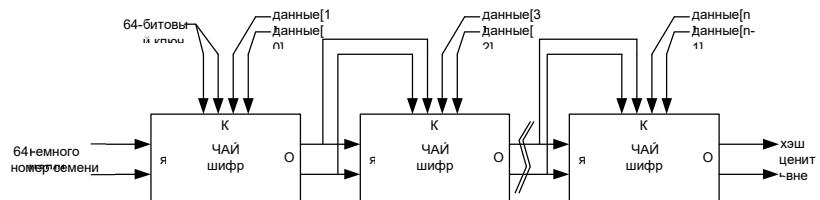


Рисунок 7-1: Сценарии использования чайного шифра.

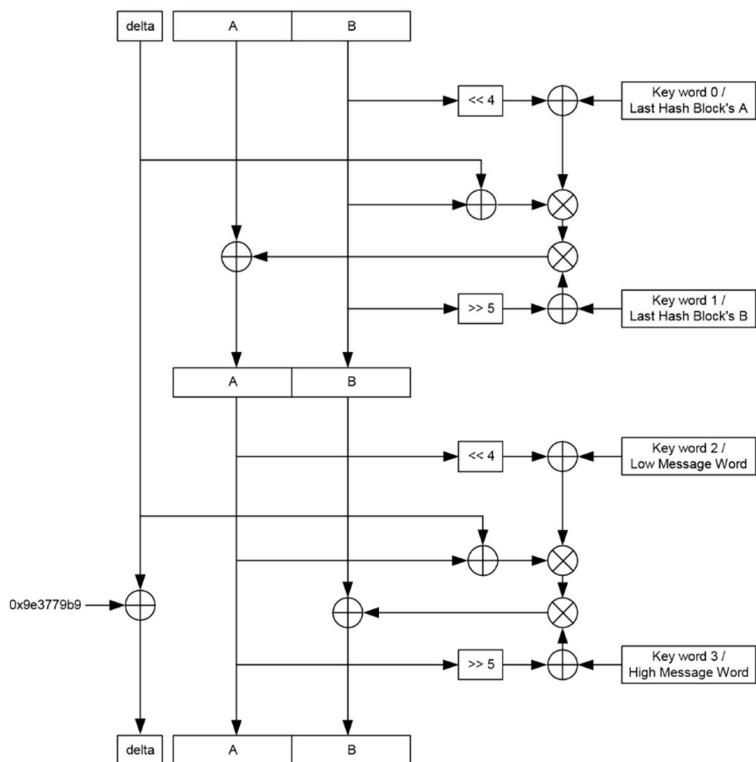


Рисунок 7-2: Внутренняя структура ТЕА. Эта диаграмма изображает один раунд ТЕА, который повторяется 32 раза для полного шифра. Ключевой график описан в полях справа для использования как шифра и как функции хеширования.

для шифрования потоков данных в реальном времени и встроенных приложений, где производительность процессора и дисковое пространство ограничены. ТЕА имеет 128-битный ключ и работает с 64 битами данных одновременно, и каждый из его 32 раундов использует только сдвиги, XOR и сложения. (Алгоритм, приведенный в листинге 7-1 и на рисунке 7-2, оптимизирован для реализации на 32-битных процессорах общего назначения.)

Алгоритм TEA от Bantam считается достаточно безопасным при использовании для шифрования и дешифрования данных. Однако TEA не используется для шифрования в Xbox; на самом деле он используется как хэш-функция, работая с шифром в модифицированном режиме Дэвиса-Майера. Область, подлежащая хэшированию, делится на 64-битные блоки. Эти исходные блоки данных используются как половина входных данных ключа для TEA. Другая половина входных данных ключа получается из результата предыдущей операции TEA, а первая операция TEA использует в качестве входных данных магическое число.

Результатом является 64-битная хэш-функция, как показано на рисунке 7-1. Этот хэш slab против атак по дням рождения, особенно с учетом вычислительной эффективности TEA, поскольку для обнаружения коллизии в среднем необходимо проверить только 232 пары сообщений. Несмотря на то, что атака по дням рождения не применяется в сценарии использования Xbox, Xbox запускает хэш дважды, каждый раз с другим начальным значением магического числа, и объединяет результаты для генерации одного 128-битного хэш-значения — вероятно, в попытке помешать атакам методом перебора.

К сожалению, у TEA есть слабость в его ключевом расписании: каждый ключ TEA имеет четыре связанных ключа. Другими словами, для каждого ключа вы можете сгенерировать три других ключа, которые дадут тот же результат шифротекста с теми же входными данными. Генерация связанных ключей так же проста, как дополнение пар ключевых битов (биты 31 и 63 — одна пара, биты 95 и 127 — другая пара). Это делает TEA непригодным для использования в качестве хэш-функции, и эта слабость хорошо документирована в статье «Криптоанализ ключевого расписания IDEA, G-DES, GOST, SAFER и тройного DES», написанной Джоном Келси, Брюсом

Шнайер и Дэвид Вагнер представили эту уязвимость много лет назад на конференции CRYPTO 1996. Позднее эта уязвимость была использована командой под руководством Энди Грина для взлома второй версии схемы безопасности Xbox.

#### PK-4

RC-4 (Ron's Code или Rivest Cipher 4) — это потоковый шифр с переменной длинной ключа, разработанный Роном Ривестом. Сердцем RC-4 является генератор потока ключей. Его можно рассматривать как криптографический генератор псевдослучайных чисел (CPRNG). Выходные данные CPRNG подвергаются операции XOR по одному байту за раз с потоком открытого текста для генерации зашифрованного текста. Расшифровка выполняется аналогичным образом. Грубо говоря, генератор «засевается» значением (ключом) длиной до 256 байт (2048 бит). Если ключ короче 256 байт, он повторяется для заполнения 256 байт перед использованием в качестве затравки; это позволяет использовать ключи

переменной длины. В Xbox длина ключа составляет 16 байт (128 бит), и поэтому шифр называется RC-4/128.

---

```

typedef struct rc4_key { unsigned
char state[256]; unsigned char x;
unsigned char y; } rc4_key;

void prepare_key(unsigned char *key_data_ptr, int key_data_len,
rc4_key *key) { unsigned char swapByte, index1, index2; unsigned
char* state; short counter;

state = &key->state[0]; for(counter = 0; counter < 256;
counter++) state[counter] = counter; key->x = 0; key->y = 0;
index1 = 0; index2 = 0; for(counter = 0; counter < 256;
counter++) { index2 = (key_data_ptr[index1] + state[counter]
+ index2) % 256; swap_byte(&state[counter], &state[index2]);
index1 = (index1 + 1) % key_data_len; } }

void rc4(беззнаковый символ *buffer_ptr, int buffer_len, rc4_key
*ключ) { unsigned char x, y, xorIndex;
unsigned char* состояніе; короткий
счетчик; x = ключ->x; y = ключ->y;

state = &ключ->state[0]; for(counter = 0; counter < buffer_len;
counter++) { x = (x + 1) % 256; y = (state[x] + y) % 256;
swap_byte(&state[x], &state[y]); xorIndex = state[x] +
(state[y]) % 256; buffer_ptr[counter] ^= state[xorIndex];
} ключ->x = x; ключ->y = y; }

```

---

**Листинг 7-2:** Код RC-4 на языке С, из оригинальной публикации Usenet.<sup>17</sup>

RC-4 считается сильным шифром, хотя в алгоритме планирования ключей есть несколько известных слабостей, которые можно использовать в плохо спроектированных криптосистемах, таких как WEP. Скотт Флурер, Ицик Мантин и Ади Шамир документируют эти слабости в статье под названием «Слабые стороны алгоритма планирования ключей RC4», представленной на Восьмом ежегодном семинаре по избранным областям криптографии (август 2001 г.). Ни одна из этих слабостей не может быть применена против реализации RC-4 в Xbox.

---

<sup>17</sup>Код из <http://www.cc.jyu.fi/~paasivir/crypt/rc4/ rc4article.txt>. Незначительные изменения пробелов, чтобы все уместилось на одной странице. Определение функции подкачки байтов также не включено, но вы можете догадаться, что она делает, по ее названию.

Однако существует потенциальная проблема в том, как RC-4 используется в первой версии безопасности Xbox. RC-4 используется на Xbox для шифрования потока кода x86, и не выполняется никакой существенной проверки расшифрованного кода для обеспечения целостности открытого текста. Это означает, что изменения в зашифрованном тексте приведут к изменениям в коде, который выполняет Xbox. Хитрость заключается в том, чтобы выяснить изменение в зашифрованном тексте, которое приведет к значимой модификации кода. Поскольку RC-4 шифрует по одному байту за раз, а коды операций x86 могут быть всего лишь одним байтом, требуется не более  $28 = 256$  итераций для «грубой силы» инструкции в одном известном месте путем мутации зашифрованного текста.

Определение местоположения для брутфорса может быть сложным, но я подозреваю, что много информации можно получить, манипулируя биты шифртекста и наблюдая за тем, что происходит с шаблоном выборки инструкций, даже при включенных кэшах. Цель будет заключаться в том, чтобы попытаться определить местоположение операндов кода операции перехода и изменить место назначения перехода таким образом, чтобы защищенная программа перешла в незащищенную область памяти.

Процесс будет похож на игру в классическую настольную игру «Морской бой». Имейте в виду, что атака настолько проста, что для угадывания килобайта кода требуется всего лишь максимум 218 итераций. Процесс угадывания можно автоматизировать, интегрировав логический анализатор с эмулятором ПЗУ через управляющий скрипт, запущенный на главном компьютере.

История RC-4 на самом деле довольно интересна. RC-4 был изобретен в 1987 году Роном Ривестом и хранился как коммерческая тайна RSA Security, Inc. до тех пор, пока не был раскрыт в 1994 году анонимным сообщением в списке рассылки шифронанков (см. Листинг 7-2). Благодаря таким достоинствам RC-4, как простота и надежность, он нашел свое применение во многих приложениях, включая

WEP, SSL, SQL и CDPD. Хотя исходный код RC-4 широко

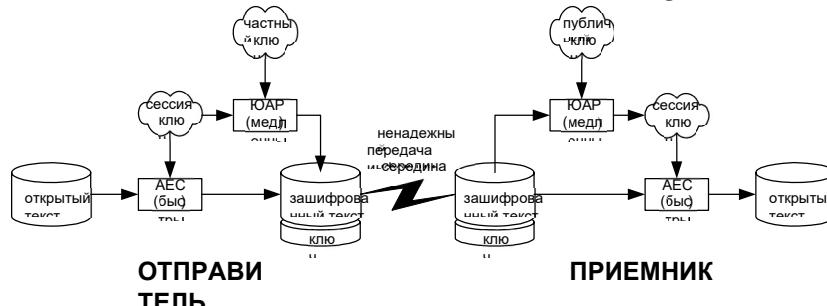


Рисунок 7-3: Использование RSA с сеансовыми ключами.

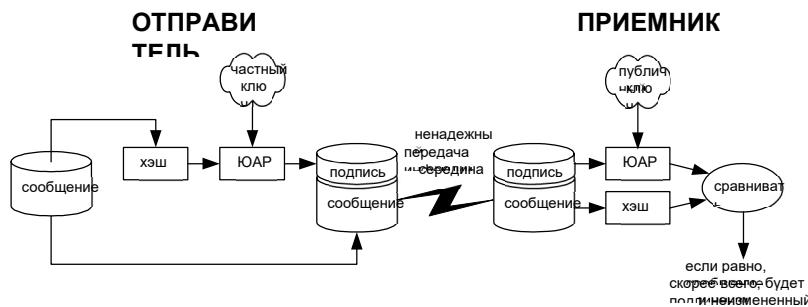


Рисунок 7-4: RSA используется для реализации цифровых подписей.

Распространенный и хорошо известный, этот шифр по-прежнему является интеллектуальной собственностью RSA Security. Я бы не рекомендовал интегрировать его в коммерческий продукт без предварительного получения лицензии от RSA Security.

## ЮАР

RSA — это алгоритм с открытым ключом, разработанный Роном Ривестом, Ади Шамиром и Леонардом Адлеманом в 1977 году. В алгоритме с открытым ключом используются два различных ключа: открытый и закрытый. Как следует из их названий, закрытый ключ должен храниться в секрете, в то время как открытый ключ может свободно распространяться. Математика, лежащая в основе RSA, кратко описана во врезке под названием «Алгоритм RSA». Вам не нужно понимать детали математики, лежащей в основе RSA, чтобы понять, как RSA используется в контексте Xbox.

В настоящее время атаки методом грубой силы считаются невозможными для RSA с длиной ключа более тысячи бит. Также следует отметить, что нельзя быть слишком беспечным в отношении того, как RSA интегрируется в криптосистему. Существуют некоторые атаки на протоколы, использующие RSA, например, обман держателя закрытого ключа, заставляющий его подписывать тщательно составленные сообщения, которые затем можно использовать для получения закрытого ключа подписчика.

Шифрование сообщения с помощью RSA так же просто, как и вызов RSA для сообщения. Однако шифрование RSA работает с блоками сообщений, которые слишком короткие, и процесс шифрования слишком медленный, чтобы быть практическим для большинства сообщений. Таким образом, RSA обычно используется для шифрования одноразового случайного ключа, называемого сеансовым ключом, для быстрого симметричного

шифра, такого как AES, который затем используется для шифрования массового сообщения. Этот процесс проиллюстрирован на рисунке 7-3.

В дополнение к шифрованию RSA позволяет использовать цифровые подписи. Цифровая подпись позволяет сторонам обмениваться сообщениями через незащищенный носитель, чтобы гарантировать, что сообщения не подделаны и не изменены. Сообщение не обязательно должно быть зашифровано. Типичный протокол цифровой подписи работает следующим образом: отправитель вычисляет хэш сообщения, которое должно быть отправлено. Затем этот хэш шифруется закрытым ключом отправителя и включается в открытый текст сообщения. Получатель расшифровывает зашифрованный хэш сообщения, используя открытый ключ отправителя, и сравнивает этот хэш с локально

## Алгоритм RSA

Алгоритм RSA был запатентован Массачусетским технологическим институтом и эксклюзивно лицензирован RSA Data Security, Inc в 1983 году. Патент на алгоритм RSA истек в сентябре 2000 года. Таким образом, сегодня RSA можно свободно использовать в любом приложении. В Интернете теперь можно найти множество отличных руководств и образовательных примеров с использованием RSA. Выполните поиск в Google, используя ключевые слова «алгоритм RSA», чтобы найти некоторые из этих примеров.

Алгоритм RSA выглядит следующим образом (адаптирован из <http://world.std.com/~franl/crypto/rsa-guts.html>):

1. Найдите два больших (длиной в тысячи бит) простых числа: «P» и «Q».
2. Выберите «E» так, чтобы  $E > 1$ ,  $E < PQ$  и E было взаимно простым числом к  $(P-1)(Q-1)$ . E не обязательно должно быть простым числом, но оно должно быть нечетным. Пара E и PQ является открытым ключом.
3. Вычислите «D» так, чтобы  $(DE - 1)$  делилось нацело на  $(P-1)(Q-1)$ . Это можно сделать, найдя целое число X, которое делает  $D = (X(P-1)(Q-1) + 1)/E$  целым числом. D — это закрытый ключ.
4. Открытый текст «T» шифруется с помощью функции  $C = (T^E) \bmod PQ$
5. Шифротекст «C» расшифровывается с помощью функции  $T = (C^D) \bmod PQ$

Обратите внимание, что  $T < PQ$ . Сообщения, размер которых превышает PQ, должны быть разбиты на последовательность сообщений меньшего размера, а очень короткие сообщения должны быть дополнены тщательно подобранными значениями, чтобы предотвратить атаки по словарю, среди прочего.

вычисленный хэш полученного сообщения. Если расшифрованный хэш, отправленный с сообщением, и локально вычисленный хэш совпадают, то получатель может сделать вывод, что сообщение является подлинным и неизмененным. Этот процесс показан на рисунке 7-4.

Если этот протокол кажется вам сложным, то так оно и есть. Есть много мест, где что-то может пойти не так. У получателя может быть поддельная копия открытого ключа отправителя. У отправителя мог быть скомпрометирован его закрытый ключ. У хэша могут быть слабые места. Использование цифровых подписей в состязательной среде требует внимания к деталям на всех уровнях проектирования системы.

В Xbox цифровые подписи используются для контроля распространения и продажи программ для консоли. Microsoft фактически контролирует как отправителя, так и получателя сообщений. Получатели — консоли Xbox — запрограммированы на запуск только тех программ, которые имеют цифровую подпись Microsoft. В идеальном мире это гарантирует, что Microsoft имеет последнее слово в вопросе о том, кто может или не может запускать программы на консоли, а хакеры не могут изменять игры, чтобы вставлять вирусы, троянских коней или бэкдоры. Сохраненные игры также запечатаны с помощью шифрования, и в результате номинально невозможно взломать игру и обмануть, исправив исполняемый файл или подняв статистику персонажа.

Очевидно, что ключевой проблемой взлома консоли Xbox является реализация системы цифровой подписи. Xbox использует хэш SHA-1 с 2048-битными ключами RSA, что делает вероятность успешной атаки методом подбора очень и очень малой. Конечно, вероятность равна нулю, если вы никогда не попробуете, но шансы против вас (см. врезку «Очень сложные проблемы»). Вам повезет больше, если вы попытаетесь выиграть в лотерею. Это не ошибка; обнаружение закрытого ключа сделало бы копирование игр тривиальным, и разработчикам не пришлось бы платить отчисления Microsoft (юридически они могут быть обязаны, но нет никаких технических причин, препятствующих им). Учитывая, что этот ключ, вероятно, стоит для Microsoft несколько миллиардов долларов, вполне вероятно, что ни один человек не знает полный ключ, поскольку методы криптоанализа с использованием резинового шланга (избиения) и зеленой бумаги (взятки) имеют тенденцию быть довольно эффективными для людей. (Не стоит исключать возможность применения «грубой силы», если вы пытаетесь защитить чрезвычайно ценный секрет!) Такие продукты, как система управления центрами сертификации SignAssure™ от BBN, обеспечивают физическую безопасность ценных ключей и реализуют схемы обмена секретами, требующие для активации машины участия нескольких доверенных пользователей.

Как упоминалось ранее, существует несколько известных жизнеспособных атак против RSA, но не все из них применимы в сценарии Xbox, поскольку они полагаются на группы пользователей или требуют выбранного шифротекста. Кроме того, список уязвимостей широко известен, и большинство реализаций цифровых подписей реализуют надлежащие контрмеры для защиты от таких атак.

## Остальная часть картины

Эффективная система безопасности требует хорошего управления ключами, надежных протоколов, а в случае Xbox — физической безопасности в дополнение к надежным шифрам и хэшам.

Управление ключами, пожалуй, одна из самых сложных задач по внедрению системы, с которой сталкивается любой архитектор безопасности. В конечном итоге ключи дешифрования должны попасть в руки пользователя. Пользовательский интерфейс должен быть спроектирован таким образом, чтобы среднестатистический пользователь с минимальной подготовкой случайно не раскрыл ключевую информацию. По мере того, как шифры становятся сильнее, самый простой путь атаки все чаще проходит через пользователя. Подслушивание с помощью видеонаблюдения, социальная инженерия или даже анализ шаблона звуков, издаваемых клавиатурой при вводе пароля, вероятно, дадут больше информации на единицу усилий о парольной фразе, чем криптоанализ. Криптография с открытым ключом частично помогает решить проблему распределения ключей, но отпечатки открытых ключей следует сравнивать лично, чтобы исключить возможность атак типа «человек посередине». Криптография с открытым ключом также не мешает кому-либо, имеющему физический доступ к клиентской машине, подслушивать расшифрованный вывод.

Кроме того, атаки на протоколы находят слабые места в способах манипулирования ключами и данными или в способах использования стойких шифров. Атака WEP на RC-4 и атака Майка Бонда и Росса Андерсона на криптопроцессор IBM 4758 являются примерами атак на протоколы. Красными флагами для потенциальных атак на протоколы являются системы, реализующие меры обратной совместимости, и системы, реализуемые инженерами, чья основная работа не связана с криптобезопасностью.

Наконец, в такой системе, как Xbox, где одной из целей является установление надежного клиента, бэкдоры и атаки с переполнением буфера также являются жизнеспособными атаками на состояние доверия машины. Ни один широко используемый коммерческий процессор не встраивает привилегии выполнения в потоки инструкций или теги данных. Процессоры слепо выполняют любой фрагмент кода, к которому им предписано перейти, независимо от того, был ли переход вызван временным аппаратным сбоем или вредоносным кодом. Периодические хэши состояния машины могут использоваться для устранения этого недостатка, но даже в этом случае проверки состояния могут быть подделаны.

Как обсуждалось в начале этой главы, установление состояния доверия клиента также требует части оборудования, устойчивой к взлому, чтобы

---

нести семя доверия. Уровень физической безопасности должен быть достаточным, чтобы сделать неэкономичным однократное нарушение безопасности, и достаточно надежным, чтобы один случай нарушения безопасности не позволил провести тривиальные атаки на остальные консоли. Некоторые компромиссы при проектировании физической безопасности, а также решения, принятые Microsoft с этой целью, обсуждаются в следующей главе.

Мораль этой главы в том, что безопасность требует хорошо спроектированной системы. Хотя шифры стали достаточно сильными, чтобы сделать атаки методом подбора бессмысленными, системы стали сложнее. Эта сложность увеличивает вероятность жизнеспособного протокола или атаки через бэкдор, но мало что делает для защиты пользователей от более традиционных атак с подслушиванием, резиновым шлангом и ошибками пользователя.



# CHAPTER 8. Обратный инжиниринг безопасности Xbox

В этой главе я опишу, как я победил первоначальную производственную версию системы безопасности Xbox, которая впервые была обнаружена в Главе 6. Система безопасности была обнаружена после анализа FLASH ROM и понимания того, что истинная инициализация оборудования и последовательность расшифровки загрузочного образа каким-то образом скрыты за пределами FLASH ROM. Глава 7 представила некоторые основные концепции криптографии, которые будут полезны для понимания содержания этой главы.

## Извлечение секретов из оборудования

Скрытый загрузочный код в Xbox, как указано в Главе 6, можно восстановить, перехватив одну из следующих шин: (1) FSB, (2) основную шину памяти или (3) соединение северный-южный мост.

Формат системной шины (FSB) процессора Pentium, используемого в Xbox, описан в технических описаниях процессоров Pentium III, доступных на веб-сайте разработчиков Intel. FSB — это двунаправленная 64-битная шина данных с примерно пятьюдесятью адресными и управляющими сигналами, все работающими на частоте 133 МГц. Шина использует соглашение о сигнализации, известное как AGTL+. Подслушивание этой шины — дорогостоящее и сложное занятие из-за большого количества сигналов и сложного физического форм-фактора. Возможные подходы включают: (a) подключение процессора к специальному разъему эмулятора, который стоит много тысяч долларов, или (b) обратная разработка значения каждой трассировки FSB на материнской плате Xbox и припаивание короткого зондирующего провода к каждому из почти сотни сигналов. Кроме того, требуется логический анализатор, поддерживающий сигнализацию AGTL+. Сочетание всех этих факторов заставило меня поискать отправную точку для подслушивания в другом месте.

Наш следующий кандидат на прослушивание, основная шина памяти, представляет собой 128-битную шину данных плюс адрес и управляющие сигналы, работающие на частоте 200 МГц с двойной скоростью передачи данных (DDR). Шина памяти использует соглашение о сигнализации, известное как SSTL-2. (Подробности этой шины можно узнать, прочитав техническое описание для части памяти Samsung K4D263238M, доступное на веб-сайте Samsung Electronics.) Несмотря на более высокую скорость,

прослушивание основной шины памяти, вероятно, проще, чем прослушивание FSB процессора, из-за пустых (запасных) отпечатков памяти, спроектированных на материнской плате Xbox.

Относительно недорогой стандартный 100-контактный адаптер TQFP (Thin Quad Flat Pack, прямоугольный корпус чипа со 100 контактами в форме крыла чайки) можно припаять к пустым местам памяти. Эти адаптеры обеспечивают удобные точки подключения логического анализатора. Проблема с этим подходом заключается в том, что вы можете захватывать только те данные, которые записаны в основную память. Ключи дешифрования обычно являются данными только для чтения, и только для чтения

## Подробнее о высокой скорости

### Передача информации

Подслушивание и изменение данных на компьютерных шинах — это мощная техника, которой трудно противостоять. Чтобы понять, как подслушивать, вам понадобится немного знаний о том, как цифровая информация передается внутри компьютера.

Существует две основные категории стандартов сигнализации: однопроводная и дифференциальная. Передача цифровой информации по проводу требует перевода в физические величины, такие как напряжение и ток. Классически сигналы определялись в терминах напряжений, измеренных относительно общего опорного потенциала, называемого «землей». Этот вид сигнализации известен как однопроводная или несбалансированная сигнализация. К сожалению, идея опорной точки заземления работает только тогда, когда сигналы изменяются медленно относительно времени их распространения. В действительности каждое изменение потенциала сопровождается потоком тока. Законы природы требуют, чтобы ток сохранялся, т. е. для каждого потока тока в одном направлении должен быть поток тока в обратном направлении. В однопроводной сигнализации обратный ток, также известный как обратный ток, должен найти свой путь обратно через «землю». На очень высоких скоростях обратные пути для тока не обязательно следуют тому же пути, что и ток сигнала. Этот дисбаланс приводит к искажению сигнала.

Дифференциальная сигнализация борется с этой проблемой, используя два провода для передачи сигнала, при этом один провод используется для тока сигнала, а другой — для явного пути обратного тока. Дифференциальный подход позволяет расположить пути сигнала и обратного тока так, чтобы они отслеживали друг друга, гарантируя сбалансированность потока тока. Результатом является более надежная система передачи сигнала за счет удвоенного количества проводов.

(продолжение)

Данные будут напрямую поступать из скрытого загрузочного ПЗУ в кэш процессора, не сохраняясь в основной памяти. Как только процессор

закончит работу со строкой кэша, содержащей ключ, он будет перезаписан, поэтому ключ никогда не должен покидать физический периметр процессора.

Третий потенциальный кандидат на подслушивание, соединение северного моста с южным мостом, представляет собой пару односторонних 8-битных дифференциальных шин, каждая из которых имеет только один управляющий сигнал и один тактовый сигнал. Шина использует соглашение о сигнализации HyperTransport и работает на частоте 200 МГц с тактированием DDR. Соглашение о сигнализации шины было выведено из общедоступной информации на веб-сайте nVidia о nForce, чипсете, тесно связанном с чипсетом Xbox. Несколько измерений с помощью осциллографа, перекрестно проверенных по открытой спецификации HyperTransport,

Конкретный стандарт для интерпретации напряжений как логических значений называется соглашением о сигнализации. Почтенные соглашения о сигнализации TTL и 3,3 В КМОП были изобретены в эпоху, когда транзисторы работали так плохо, что требовались большие отклонения сигнала. В последнее время набирает популярность множество новых и даже старых соглашений о сигнализации, таких как SSTL (логика с последовательным шлейфом), GTL (логика приемопередатчика gunning), LVDS (низковольтная дифференциальная сигнализация) и PECL (логика с псевдоэмиттерной связью). Эти высокоскоростные соглашения о сигнализации учитывают тот факт, что электрические волны распространяются медленно по сравнению со скоростью передачи данных. Они также учитывают тот факт, что электрические волны несут энергию, которая должна рассеиваться по завершении своего пути, в противном случае энергия будет отражаться и вызывать помехи для входящих волн.

В высокоскоростных приложениях провода часто называют «линиями передачи», чтобы подчеркнуть тот факт, что эти волны распространяются медленно по сравнению со временем перехода сигнала (временем, необходимым для перехода сигнала между состояниями «1» и «0»). (Обратите внимание, что сравнение скорости производится относительно времени перехода сигнала, а не его общей частоты сигнала.) Распространенной ошибкой является мнение, что эффекты линии передачи можно игнорировать, поскольку тактовая частота сигнала низкая. Даже если в год происходит только один переход, проблемы все равно могут возникнуть, если длительность этого перехода составляет всего пикосекунду (одну триллионную секунды).

Хорошей новостью для новичков является то, что новейшие ПЛИС от таких поставщиков, как Xilinx, поставляются со встроенной поддержкой практически всех широко распространенных стандартов сигнализации. Другая хорошая новость заключается в том, что стандарты сигнализации становятся все более хорошо документированными. Например, спецификации ПЛИС Xilinx иллюстрируют ожидаемое положение и значение резисторов согласования для каждого поддерживаемого стандарта сигнализации. Следуя рекомендуемым практикам в спецификации и примечаниях к применению, вы можете использовать ПЛИС для подслушивания широкого спектра сигналов. Просто помните, что

подслушивающие отводы должны быть как можно короче, и вы не ошибитесь.

Для проверки предположения о том, что действительно используется соглашение о сигнализации HyperTransport, использовались данные, доступные на веб-сайте консорциума HyperTransport.

Шина HyperTransport реализована на материнской плате Xbox со всеми параллельными и равномерно разнесенными сигналами, что, вероятно, обусловлено высокой рабочей скоростью шины. Это делает шину идеальной целью для подслушивания, если не считать того факта, что она работает на такой высокой скорости передачи данных. Подслушивание шины, работающей на такой скорости, требует особого внимания к длине шлейфа трасс подслушивания (чтобы сохранить целостность сигналов), а также требует довольно дорогого логического анализатора или специальной схемы анализатора.

В конечном итоге соединение Northbridge-Southbridge было выбрано в качестве первой шины для подслушивания, поскольку оно имеет гораздо меньшее проводов, и, следовательно, требует меньше всего пайки. Соединение Northbridge-Southbridge имеет всего десять уникальных сигналов, тогда как FSB и основная память имеют около сотни сигналов каждое. Пайка большого количества соединений не только занимает много времени, но и значительно увеличивает риск аппаратных сбоев из-за перемычек припоя или поврежденных дорожек. Таким образом, минимизация количества паяных соединений минимизирует риск сопутствующего повреждения материнской платы.

## Подслушивание скоростного автобуса

Я принял решение использовать метод подслушивания HyperTransport в конце января 2002 года. Существенными техническими проблемами этого подхода были:

- Подключения к высокоскоростной дифференциальной шине без нарушения целостности сигнала
- Поиск или создание инструмента ведения журнала, который мог бы поддерживать скорость передачи данных 400 МБ/с на шине HyperTransport
- Определение полярности и порядка бит дифференциальных дорожек шины HyperTransport на материнской плате

## Бюджетный автобус

Первые две проблемы тесно связаны. Высокоскоростные инструменты анализа и регистрации шин обычно имеют фирменные интерфейсы, которые требуют специального адаптера для материнской платы Xbox. Последняя проблема, определение полярности и порядка бит, просто требует большой постобработки и манипулирования данными после того, как регистратор данных подключен и работает.

HyperTransport — это открытый стандарт, получивший признание в отрасли, что означает, что для шины доступны готовые анализаторы протоколов и инструменты регистрации. Одним из таких примеров является анализатор протоколов HyperTransport от FuturePlus. К сожалению, на момент выполнения работы этот анализатор протоколов стоил более 25 000 долларов. Кроме того, анализатор протоколов требует, чтобы целевая плата была специально разработана для размещения модуля интерфейса шины анализатора протоколов.

Вместо того, чтобы покупать анализатор протоколов и тратить время и усилия на его адаптацию для использования с Xbox, я построил свой собственный упрощенный. Эта задача выполнима, поскольку протокол HyperTransport довольно прост. Реализация HyperTransport для Xbox использует две 8-битные односторонние шины, одну для передачи и одну для приема. Каждая шина имеет тактовый сигнал и линию строба, связанную с ней. Стандарт сигнализации требует, чтобы на каждом фронте тактового сигнала были представлены допустимые данные. Начало нового пакета обозначается выходом линий данных из состояния ожидания. Линия строба различает пакеты команд и данных. Все сигналы боковой полосы, типичные для других шин, такие как адрес, управление чтением/записью, выбор чипа и линии прерывания, обрабатываются в HyperTransport с использованием внутриполосных командных пакетов. Таким образом, для прослушивания шинам вам понадобится всего десять дифференциальных сигналов (двадцать проводов) — отличная новость для хакеров.

Протокол HyperTransport достаточно прост, но как насчет того, чтобы найти что-то, что может и физически взаимодействовать с шиной Xbox, и поддерживать скорость 400 МБ/с? Идеальным инструментом для создания этого ответвителя шины HyperTransport была бы ПЛИС. Однако в то время не было ПЛИС, которая могла бы поддерживать высокие скорости передачи данных, и, что еще важнее, не было ПЛИС, сертифицированной поставщиком для использования с HyperTransport. Теоретически, ПЛИС Xilinx Virtex-II могла бы работать для этого приложения, но продукт только что был выпущен, а устройства были чрезвычайно дорогими и их было трудно достать (сегодня вы можете купить ПЛИС Virtex-II начального уровня менее чем за сотню долларов). Лучшей ПЛИС, которая была у меня под рукой в то время, была ПЛИС Xilinx Virtex-E, которую я ранее спроектировал в прототип сетевого маршрутизатора суперкомпьютера в рамках своей

диссертации. Плата сетевого маршрутизатора использовала сигнализацию СТГ (Center Tap Terminated) для своих сетевых интерфейсов, а также имела на борту процессор Intel StrongArm для настройки, управления и отладки.

Таким образом, задача сводилась к тому, чтобы выяснить, как соединить сигналы HyperTransport с сигналами СТГ и как добиться производительности 400 МБ/с от ПЛИС, которая не была предназначена для работы на таких скоростях.

Оказывается, соглашение о сигнализации HyperTransport является близким родственником более распространенного соглашения LVDS (низковольтная дифференциальная сигнализация), указанного в стандарте TIA/EIA-644. Драйверы HyperTransport создают сигнал с дифференциальным размахом 600 мВ, как правило, сосредоточенным вокруг синфазного напряжения 600 мВ. С другой стороны, приемники LVDS могут извлекать смысл из данных, которые имеют дифференциальный размах более 100 мВ и синфазное напряжение в диапазоне от 50 мВ до 2,35 В. Таким образом, приемники LVDS напрямую совместимы с драйверами HyperTransport! (Хотя Virtex-E поддерживает прямой интерфейс с сигналами LVDS, я не мог этим воспользоваться, поскольку детали Virtex-E, которые у меня были, уже были спроектированы в систему, которая жестко подключена для сигналов СТГ.) Если вы разрабатываете собственную плату ответвления, лучшим подходом будет использование собственных возможностей LVDS ПЛИС вместо описанного хака

## А как насчет передачи сигналов на HyperTransport?

Для подслушивающего приложения, описанного в этой главе, требуется только приемник HyperTransport. Такие приложения, как атаки «man-in-the-middle», требуют устройства, которое может переопределять сигналы HyperTransport и вставлять один или два ложных бита. Такое устройство осуществимо, поскольку HyperTransport, как и LVDS, использует драйверы токового режима. Другими словами, драйверы предназначены для подачи только измеренного количества тока в провод, независимо от создаваемого им напряжения. В обычной ситуации это работает отлично, поскольку импеданс провода преобразует ток в напряжение в соответствии с законом Ома. Однако токи могут суммироваться и компенсировать друг друга. Антагонистический дифференциальный драйвер, который применяет ток перегрузки, который компенсирует предполагаемый сигнал, может быть подключен к линии HyperTransport. Этот вид перегрузки может быть достигнут с помощью гибкого программируемого ввода-вывода, предусмотренного в ПЛИС, таких как Xilinx Virtex-E и Virtex II.

Простейшим применением такого устройства переопределения шины будет изменение назначения вектора сброса по мере его передачи в ЦП, что позволит вам получить контроль над Xbox. Назначение вектора сброса кодируется в один байт, который следует за кодом операции «перехода», расположенным по адресу 0xFFFF.FFF0. Вектор сброса, вероятно, передается через определенное количество тактов с момента деактивации сброса, поэтому элемент синхронизации для этой атаки может состоять только из

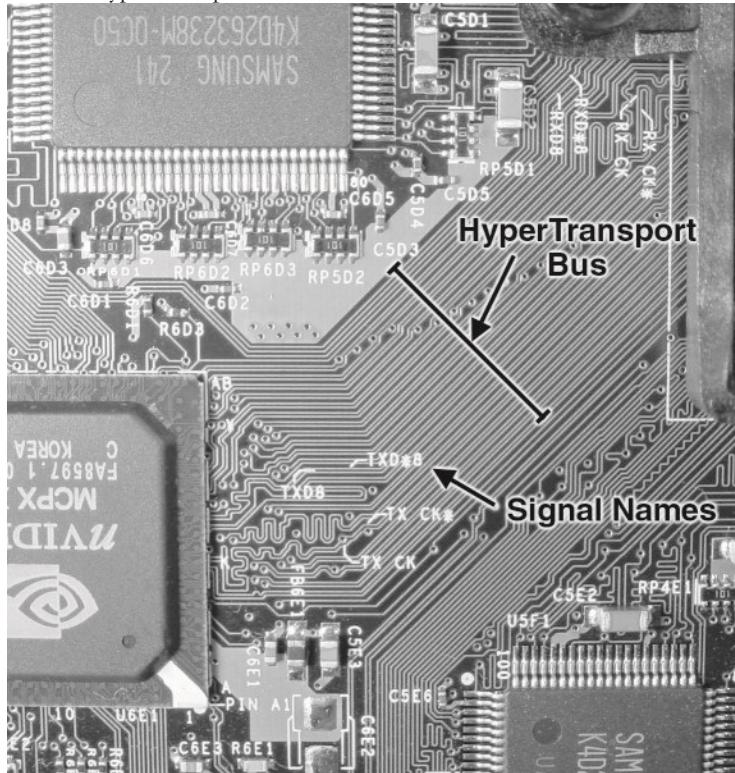
таймера, который тактируется тактовым сигналом шины HyperTransport и синхронизируется с сигналом сброса. Такая атака «человек посередине» разрушит даже криптографически защищенную реализацию загрузочного блока с открытым ключом.

здесь. Кроме того, приемник LVDS должен быть расположен очень близко к материнской плате Xbox, чтобы не искажать целевые сигналы. Длинный кабель будет рассеивать энергию из проводов и вносить шум и отражения, которые могут привести к прекращению работы системы.

Решением проблемы передачи сигналов HyperTransport на ПЛИС является использование микросхемы преобразования сигнала. LVDS — популярный стандарт для интерфейсов ЖК-панелей и объединительных плат, используемых в телекоммуникационных системах, поэтому доступны многочисленные недорогие преобразователи LVDS-CMOS. Конечно, желаемым соглашением о сигнализации является СТГ, но при более внимательном рассмотрении становится ясно, что сопряжение драйверов CMOS с приемниками СТГ на самом деле не является проблемой. СТГ — это соглашение о сигнализации в токовом режиме, которое подает +8 мА или -8 мА на линию передачи сопротивлением 50 Ом, нагруженную на 1,5 В. Приемник представляет собой дифференциальный усилитель, который сравнивает опорное напряжение окончания с напряжением линии передачи. В Virtex-E усилитель приемника СТГ рассчитан на работу до тех пор, пока принимаемое напряжение колеблется более чем на 200 мВ вверх или вниз от опорного напряжения. Большинство передатчиков CMOS, управляющих линией с оконечной нагрузкой СТГ, не будут иметь проблем с подачей или отводом 8 мА тока на нагрузку сопротивлением 50 Ом. Кроме того, передатчики CMOS не должны иметь проблем с управлением проводом, нагруженным на фиксированное напряжение. Таким образом, стандартный чип преобразователя LVDS в CMOS может использоваться для приема сигналов HyperTransport материнской платы Xbox и подачи их на плату, которую я ранее построил для своей диссертации. Я выбрал чип Texas Instruments SN65LVDS386, и вы можете найти технические описания для этого чипа на веб-сайте Texas Instruments.

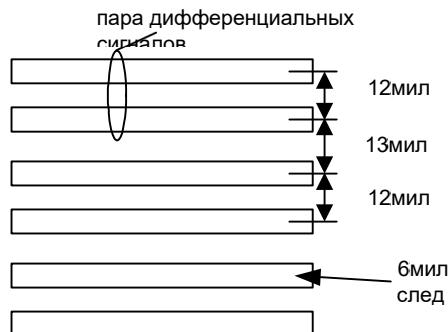
Присоединение микросхемы преобразователя LVDS-CMOS к плате стало восхитительно простым благодаря чистой компоновке, используемой для шины HyperTransport на материнской плате Xbox. На рисунке 8-1 показано, как выглядят трассы шины HyperTransport. Обратите внимание, как все провода идут параллельно и как они равномерно распределены. Некоторые из проводов, такие как тактовый (TX CK/TX CX\* и RX CK/RX CX\*) и линия стробоскопа (TXD8/TXD8\* и RXD8/RXD8\*), даже помечены для нас маркировкой полярности! Эта простая компоновка позволяет использовать простую в проектировании плату ответвления.

Плата ответвления содержит только микросхему преобразователя LVDS-CMOS, некоторые схемы кондиционирования питания и набор дорожек, проложенных прямо до края платы, которые идентичны шине HyperTransport на материнской плате Xbox. Для одинакового расстояния и простоты выравнивания и монтажа я измерил размеры этих дорожек с помощью цифрового штангенциркуля. На рисунке 8-2 показаны размеры дорожек шины HyperTransport.



**Рисунок 8-1:** Трассировки шины HyperTransport, расположенные на материнской плате Xbox.

Измерения были немного сложными. Мой подход заключался в том, чтобы измерить общую ширину шины и разделить ширину на количество дорожек и пробелов, чтобы получить среднее ожидаемое расстояние и ширину дорожки. Затем я разместил эти дорожки с помощью программы САПР печатных плат и распечатал макет на бумаге в масштабе 1:1. Я сравнил напечатанные дорожки с дорожками платы и внес несколько корректировок вручную. (Обратите внимание, что многие принтеры имеют небольшую погрешность масштабирования, поэтому, если вы пытаетесь это сделать, откалибруйте себя, распечатав несколько длинных линий известной длины и измерив их. Принтеры могут иметь разные погрешности масштабирования вдоль горизонтальной и вертикальной осей, поэтому обязательно печатайте линии в обоих направлениях.)



**Рисунок 8-2:** Размеры дорожек шины HyperTransport на материнской плате Xbox. «Мил» равен 1/1000 дюйма или 25,4 микрона.

Проектирование собственных плат довольно просто с правильным программным обеспечением. Вы можете узнать больше о том, как сделать собственные платы, прочитав Приложение C, «Начало разработки печатной платы».

После завершения процесса выбора компонентов, проектирование и компоновка платы HyperTransport и преобразования сигнала заняли всего несколько часов. Схема конструкции платы представлена на рисунке 8-6. Затем плата была изготовлена по заказу, размещенному через Интернет. Многие компании, изготавливающие платы, предлагают доступные, быстрые услуги по изготовлению плат, которые принимают проекты плат в формате Gerber по электронной почте или через FTP-загрузку. В этом случае мне было изготовлено две копии платы за пять дней по цене 33 доллара за плату (см. Приложение C, «Начало компоновки печатной платы», для получения дополнительной информации о том, как изготавливать собственные платы). Эта цена включает только стоимость резки платы на квадратный кусок. Однако мне нужно было, чтобы сторона платы с HyperTransport имела особую форму, которая облегчала бы монтаж платы, не мешая существующим компонентам на материнской плате Xbox. Мне также нужно было, чтобы сопрягаемый край платы был скопченным таким образом, чтобы плата устанавливалась под небольшим углом, чтобы упростить задачу пайки платы с отверстием к материнской плате. Я использовал ленточную шлифовальную машину, чтобы вручную придать кромке форму, показанную на рисунке 8-3. При лепке доска должна была быть ориентирована таким образом, чтобы абразивная лента ленточной шлифовальной машины сначала соприкасалась с направляющей стороной доски, чтобы предотвратить отрывание ленточной шлифовальной машиной медных дорожек от доски. Будьте осторожны при использовании ленточной шлифовальной машины для лепки небольших досок, таких как доска Tap Board, — ленточная шлифовальная машина может так же случайно поцарапать ваши пальцы.

## вид спереди

## вид сбоку



Рисунок 8-3: Формирование края платы ответвления HyperTransport.

После формирования сконченного края все детали были припаяны к плате. (См. Приложение B, «Техники пайки».)

Готовую плату отвода теперь нужно было прикрепить к материнской плате Xbox. Этот критический шаг был, пожалуй, самым сложным. Сначала материнская плата Xbox была подготовлена с помощью мелкозернистой наждачной бумаги, чтобы снять зеленую паяльную маску, обнажив яркую голую медь целевых дорожек. Затем эти дорожки были покрыты флюсом, и с помощью горячего паяльника был нанесен тонкий слой припоя.

Процедура, которую я использовал для крепления платы ответвления к материнской плате, показана на рисунке 8-4. Подготовленная плата ответвления была прикреплена к материнской плате в приблизительном месте и под углом с помощью тонкого (30 AWG) провода, припаянного между дорожкой на плате ответвления и материнской платой. Провод гвоздя служит только как временное средство для удержания

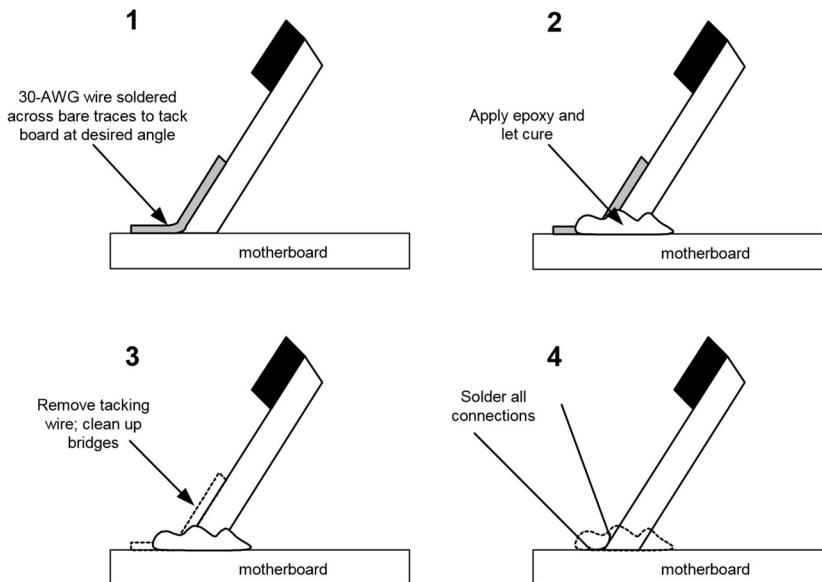


Рисунок 8-4: Процедура пайки платы ответвления.

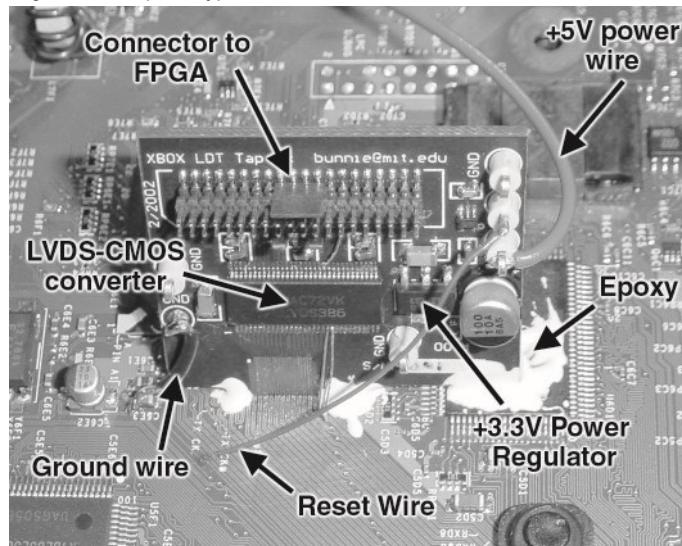


Рисунок 8-5: Плата HyperTransport, установленная на материнской плате Xbox.

плата на месте и будет удалена, поэтому не имеет значения, если провод перекрывает несколько дорожек. После того, как провод был прикреплен, я осторожно отрегулировал положение платы ответвления на материнской плате, нагревая провод, чтобы ослабить его связь, чтобы избежать подъема любых медных дорожек. (Я использовал микроскоп, чтобы помочь в определении оптимального выравнивания.) Как только я был удовлетворен положением платы, я нанес прочную эпоксидную смолу на соединение платы, чтобы закрепить все на месте. Эпоксидная смола должна затвердеть и образовать жесткое, жесткое соединение. (Обратите внимание, что некоторые эпоксидные смолы при неправильном нанесении затвердевают в гель; это неприемлемо, так как вся механическая целостность соединения должна исходить от эпоксидной смолы, а не от паяных соединений.) Я использовал формулу эпоксидной смолы MillerStephenson 907, и она застыла с достаточной прочностью, чтобы я мог поднять Xbox за плату ответвления и не повредить соединение ответвления.

После того, как эпоксидная смола затвердела, я удалил временную проволочную прихватку, которая использовалась для удержания платы ответвления на месте, и очистил голые сопряженные дорожки небольшим количеством припоя и флюса. Последний шаг пайки дорожек платы ответвления к голым дорожкам материнской платы теперь ничем не отличался от пайки любого компонента поверхностного монтажа на плату; большинство стандартных методов, описанных в Приложении В, применимы непосредственно к этой ситуации. На рисунке 8-5 показано, как выглядит готовая сборка.

## Создание регистратора данных

Вторая проблема прослушивания шины HyperTransport — это приобретение или создание регистрирующего устройства, способного поддерживать скорость передачи данных шины в 400 МБ/с. Учитывая мой бюджет, я решил, что единственным вариантом для меня будет создание регистратора, поскольку покупка любых инструментов с достаточной производительностью для этой работы была далеко за пределами моего бюджета.

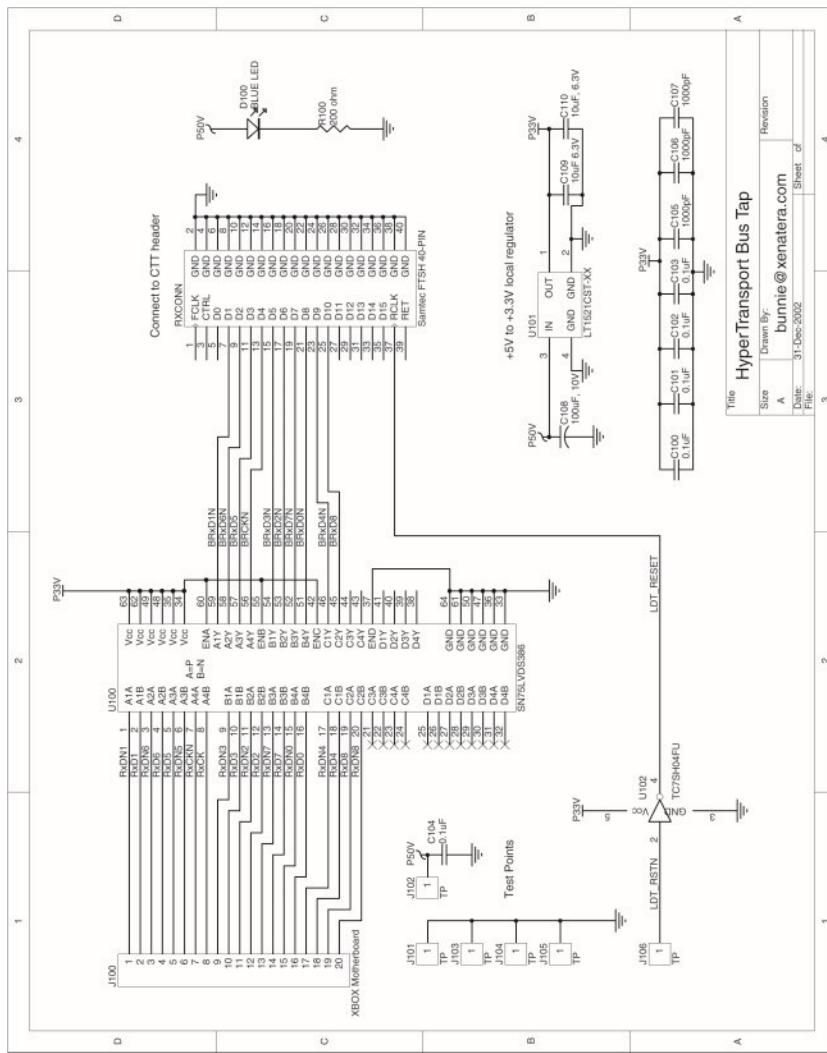
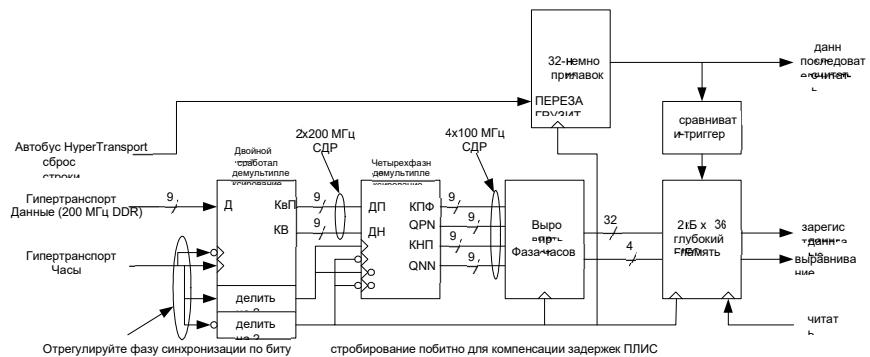


Рисунок 8. Схема платы ответвления HyperTransport.

При создании устройства регистрации я остановился на использовании Virtex-E FPGA, которая была интегрирована в плату, которую я построил ранее. Однако одна проблема с использованием Virtex-E FPGA заключается в том, что производительность FPGA (как указано в справочнике) недостаточна для того, чтобы идти в ногу сшиной HyperTransport. К счастью, FPGA хорошо разгоняются, потому что их производственный запас очень консервативен, и потому что производительность FPGA в значительной степени ограничена задержками распространения сигнала в настраиваемой структуре проводки. В результате некоторые ключевые пути, ограничивающие производительность, можно вручную определить и компенсировать с помощью линий мягкой задержки и выборочно инвертированных тактовых импульсов. Наиболее чувствительные к производительности блоки можно разместить вручную для оптимизации задержек, в то время как компилятор и



**Рисунок 8-7:** Блок-схема регистратора данных, встроенного в ПЛИС Xilinx Virtex-E.

автоматизированный инструмент размещения и маршрутизации обрабатывает некритические части схемы. На рисунке 8-7 показана общая конструкция, которая использовалась для сбора данных на шине HyperTransport.

Концепция конструкции довольно проста: взять высокоскоростные данные с шины HyperTransport и синхронизировать их в четыре фазы четвертьскоростного тактового сигнала, создав поток данных, который в четыре раза медленнее, но в четыре раза шире. Это ограничивает все ручное размещение и настройку только первыми несколькими входными триггерами. Затем перестроить данные с помощью набора задержек и ротаторов и сохранить данные по одному фрагменту за раз в памяти FIFO (первым пришел, первым вышел). Сигнал, который запускает начало захвата FIFO, генерируется таймером-компаратором, который начинает отсчет с первого сброса. Длинные окна данных можно захватывать путем объединения результатов нескольких запусков, каждый из которых имеет

точку запуска захвата, отложенную от предыдущего. Более поздняя оптимизация, применяемая к схеме запуска, — это функция «не хранить нули» (DNSZ). В режиме DNSZ данные, состоящие из одних нулей, не сохраняются в FIFO. Это полезно для отбраковки всех неиспользуемых данных нашине HyperTransport. Полученные трассировки данных представляют собой серию 32-битных слов с метками времени.

Самой сложной частью конструкции регистратора данных FPGA была калибровка задержек на входных путях. Калибровка задержки была выполнена с помощью осциллографа для зондирования небольшого окна данных нашине HyperTransport. Задержки проводов и вращения побайтам подстраивались до тех пор, пока зондируемые данные не совпадали с данными журнала. Этот процесс облегчался тем фактом, что во время простого общая последовательность команд повторялась нашине каждые несколько сотен микросекунд, что служило эталоном калибровки.

### Определение порядка и полярности шины

Последняя задача после регистрации данных — выяснить порядок сигналов нашине HyperTransport и их полярности. Обратите внимание, что хотя два самых важных сигнала шины HyperTransport на материнской плате Xbox помечены для нас, остальные восемь линий данных имеют неоднозначную полярность и порядок битов.

Правильная полярность восьми сигналов данных была определена путем наблюдения за шаблоном бит данных шины бездействия. Шина HyperTransport проводит большую часть времени в состоянии бездействия, поэтому это несложно. Если шаблон бездействия должен состоять из всех нулей, то любая позиция бита, которая отображается как 1, имеет инвертированную полярность. Это было исправлено аппаратно путем вставки инверсионного термина в ПЛИС на соответствующем проводе.

Однако определить правильный порядок битов гораздо сложнее. Действуя на основе предположения, что данные, поступающие через Шину HyperTransport в значительной степени должны исходить из FLASH ROM, подсчет единиц выполнялся побайтно. Теория заключается в том, что порядок шины представляет собой чистую перестановку, то есть число двоичных единиц в байте сохраняется между данными FLASH ROM и данными, захваченными регистратором. Шаблоны подсчетов единиц были выстроены друг против друга, чтобы определить возможные области соответствия между FLASH ROM и записанными данными. К счастью, первые несколько слов, которые встретились нашине HyperTransport, — это некоторые инициализации, специфичные для чипсета, которые расположены в нижней части FLASH-памяти, поэтому поиск набора шаблонов, которые выстроились правильно, не занял много времени. Набор байтов из каждого ROM и регистратора был сведен в таблицу, и с помощью короткой программы

на языке С столбцы битов были транспонированы до тех пор, пока не был найден порядок, который заставил все значения строк совпасть.

## Осмысление полученных данных

Теперь, когда были извлечены достоверные следы данных, проблема остается в расшифровке смысла всего этого. Прежде чем сделать это, давайте подведем итог тому, что мы знаем о данных, которые мы собрали до сих пор.

- **Временная корреляция.** Зарегистрированные данные в макроскопическом масштабе должны иметь сильную временную корреляцию с ожидаемой последовательностью событий инициализации: инициализация *jam table*, за которой следует шаг расшифровки, за которым следует выполнение из RAM. Области трассировок журнала, соответствующие каждому из этих событий, можно определить, просто наблюдая, когда происходят большие всплески активности, за которыми следуют области типины.
- **Продолжительность транзакций.** Поскольку процессор Pentium имеет как кэш данных, так и кэш инструкций, все выборки по шине HyperTransport во FLASH-ПЗУ или скрытое загрузочное ПЗУ должны осуществляться пакетами трафика одинаковой длины.
- **Гарантированный заказ.** Собранные данные имеют временную метку и хронологически верны, поэтому, если первую инструкцию, извлеченную векторе сброса, можно идентифицировать в журналах данных, можно вывести положение и структуру остальных инструкций.

Первоначально я не проверил макроскопическую организацию данных, проходящих через шину HyperTransport, и это вызвало у меня некоторые проблемы. Упрощенная блок-схема машины регистрации на рисунке 8-7 будет иметь журнал FIFO, сбрасываемый каждый раз при сбросе шины HyperTransport. Это кажется хорошей идеей, однако я изначально неверно предположил, что шина HyperTransport сбрасывается только один раз при подаче питания. В действительности шина HyperTransport сбрасывается второй раз после шага инициализации таблицы затворов. Таким образом, когда я впервые начал просматривать трассировки, все, что я увидел, были зашифрованные данные плюс немного кода, ни один из которых не мог быть выстроен в логическую схему с вектором загрузки.

Представьте, как это было разочаровывающе! Я сделал шаг назад и наблюдал события шины HyperTransport на осциллографе с временной шкалой, установленной на миллисекунды на деление. Я заметил, что был более ранний импульсброса, и после настройки механизма запуска на захват только первого импульса, инструкцию загрузки было легко идентифицировать. Шестнадцать байтов в 0xFFFFFFF0 в секретном ПЗУ оказались идентичны тем же шестнадцати байтам во FLASH-ПЗУ. С этого момента я отслеживал текущее значение счетчика программ, выполняя множество грубых трассировок и разборок с учетом, так что я мог поместить каждый блок инструкций в правильное место в памяти. Каждая выборка строки кэша состояла из 16 или 32 последовательных байтов памяти, что приводило к отличительному шаблону временной метки регистратора данных, который помог процессу обратного проектирования. После нескольких часов просеивания трассировок в поисках строк кэша, я

## Дополнительные инструменты для работы: инструменты анализа программного обеспечения

Неизбежно в какой-то момент вашего хакерского опыта вы столкнетесь с необходимостью дизассемблировать некоторый код на языке ассемблера. Я познакомился с прекрасным инструментом для этой работы от нескольких коллег-хакеров программного обеспечения в январе 2002 года, когда я занимался обратным проектированием безопасности Xbox. Инструмент называется «IDA Pro» от Ilfak Guilfanov, продается DataRescue Corporation (<http://www.datarescue.com/idabase/>). IDA Pro способен разбирать не только код x86, но и огромное количество кода встроенных процессоров. Качество вывода IDA Pro также очень высокое: сегменты кода автоматически аннотируются и организуются для удобства чтения. IDA Pro также имеет широкий спектр полезных и интересных инструментов. Некоторые из моих любимых включают возможность автоматического сопоставления сигнатур библиотеки кода с вызовами функций и возможность отслеживать переходы при нажатии клавиши.

Еще один инструмент, который оказался весьма полезным при анализе кода, — HackMan. HackMan — это бесплатная программа от Корпорация «ТехноЛогизмики» (<http://www.technologismiki.com/hackman/>). Номинально это «шестнадцатеричный редактор», т. е. редактор файлов, который позволяет вам напрямую манипулировать двоичными данными, но у него есть множество уникальных возможностей, которые выходят далеко за рамки простого редактирования. Например, в HackMan есть встроенный дизассемблер. Дизассемблер не такой мощный, как IDA Pro, но он интерактивен с шестнадцатеричным редактором. Это позволило мне быстро проверить строки кэша-кандидатов на допустимый код, просматривая журналы данных и собирая окончательный двоичный образ секретного ПЗУ.

собрал достаточно кода для загрузки в дизассемблер. (См. боковую панель об инструментах анализа программного обеспечения для получения дополнительной информации о дизассемблере, который я использовал.)

После небольшой обработки данных и значительной помощи от нескольких друзей-хакеров мы определили, что использовался шифр RC-4/128. RC-4 — это симметричный шифр, и ключ должен был храниться где-то в Xbox, но у меня возникли трудности с определением ключа в потоке данных. Ключ, казалось, охватывал выборки строк кэша, которые были общими с частями кода, которые в то время я не мог сопоставить с определенным местоположением.

Поскольку ночь была уже длинная, и я устал смотреть на шестнадцатеричные цифры, я решил попробовать то, что никогда не должно было сработать. Я адаптировал программу дешифрования RC-4 для дешифрования целевого изображения во FLASH ROM с помощью ключа, который был получен из скользящего окна в журнале данных. Это довольно грубый подход, поскольку он требует десятков тысяч дешифрований (по одному на каждый байт в журнале) для поиска по всему потоку данных. Я автоматизировал процесс, подав вывод дешифрования RC-4 в процедуру гистограммы. Если ключ не совпадал, вывод должен был быть статистически «белым». Другими словами, гистограмма вывода должна показывать, что все значения примерно равновероятны для несовпадающего ключа. Однако, если ключ был правильным, гистограмма должна быть смещенной, при этом некоторые значения были значительно популярнее всех остальных значений.

В конце концов я закончил программу trykeys, чтобы выполнить этот поиск методом перебора около 5 утра. С затуманенными глазами и усталый, я решил дать программе тестовый запуск, прежде чем объявить о завершении на ночь. Представьте себе опешомленное выражение моего лица, когда я наблюдал за выводом программы, пока она хрюстела потоком данных-кандидатов:

```
$ ./trykeys.exe ms4.bin binout.full
.....
..... найдена возможная комбинация клавиш: ср. 96, мин.
5, смещение 8745..... Образ
FLASH ROM имеет имя ms4.bin, а трассировка двоичного регистратора
данных называется binout.full. Программа trykeys
иентифицировала статистически отличную гистограмму (со
средним значением 96 и минимальной высотой контейнера 5) для
расшифровки образа ПЗУ, используя в качестве тестового ключа
данные, начинающиеся со смещения 8745. Затем я изолировал
потенциальный ключ от потока данных и проанализировал
расшифрованный вывод с использованием потенциального ключа.
Вывод выглядел как настоящий, действительный код. Я нашел
ключ в скрытом загрузочном секторе, сохраненном в чипе южного
моста! Несколько дней спустя, немного поспав и доделав
школьные задания, я закончил делать надлежащий анализ потока
данных и собрал изображение всего секретного загрузочного
сектора.
```

Имея на руках ключ RC-4 секретного загрузочного кода, я имел возможность генерировать образы FLASH ROM, которые могли быть приняты любой

Xbox в то время. Подразумевается, что весь механизм доверия Xbox может быть нарушен простой переопределением или заменой ROM на материнской плате Xbox. Это достигается с помощью тестовых структур, предоставленных Microsoft для переопределения FLASH ROM во время производства для целей тестирования и диагностики. Xbox должны сходить с конвейера со скоростью один каждые пару секунд, поэтому Microsoft разработала набор быстро подключаемых контрольных точек, которые позволяют переопределить FLASH ROM. Возможность загрузки на

### Правовые проблемы хакерства

Оглядываясь назад, можно сказать, что взлом Xbox был менее сложным технически, чем в социальном и юридическом плане. После извлечения секретного ключа из чипа Southbridge я встретился со своим научным руководителем, профессором Томом Найтом, в Лаборатории искусственного интеллекта MIT, чтобы обсудить свои результаты. Мой руководитель указал, что моя работа может нарушать DMCA, поэтому перед публикацией мы связались с юридическим отделом MIT для консультации. В конечном итоге юридический отдел MIT ответил, что DMCA делает дело слишком рискованным и что мне придется публиковать как индивидуальное лицо, несмотря на то, что моя работа проводилась в MIT как часть моего исследования в области компьютерной архитектуры. Я отчаялся, думая, что никогда не смогу позволить себе адвоката и что я никогда не смогу опубликовать свои результаты, но затем профессор Хэл Абельсон связал меня с Electronic Frontier Foundation (EFF). В результате Ли Тиен и Джо Лю из EFF и Бостонского колледжа были назначены, чтобы помочь мне опубликовать мою работу. Последовали месяцы размышлений и позиционирования. Это была битва на два фронта: нам нужно было убедить MIT принять работу, одновременно пытаясь у说服ить Microsoft. Спустя четыре месяца MIT капитулировал после обнадеживающего обзора моей работы Microsoft и подавляющей поддержки моих коллег по лаборатории и профессоров. MIT решил, что я могу опубликовать свою работу как студент MIT, а не как независимая организация. Результатом пяти месяцев юридического туника стал технический меморандум AI Laboratory, за которым последовала академическая презентация работы на конференции по криптографическому оборудованию во встроенных системах (CHES) в августе 2002 года.

Хотя финал этой истории может быть счастливым, все могло бы быть совсем иначе, если бы не поддержка моего руководителя, моей лаборатории и талантливых юристов EFF. DMCA проводит размытую границу между мошенником-хакером и законным исследователем; возможно, без одобрения MIT я бы не смог удовлетворить исследовательское исключение DMCA, и мое исследование никогда бы не было опубликовано, или оно могло бы быть опубликовано и оспорено Microsoft. Свобода слова распространяется на всех, а не только на тех, кому повезло сидеть в башнях из слоновой кости уважаемых академических учреждений. Есть бесчисленное множество других, которые также работали над Xbox с превосходными результатами, но их голоса навсегда останутся безмолвными за занавесом DMCA.

Альтернативный образ ПЗУ ценен для запуска программ производственного тестирования с использованием собственного

процессора Xbox. Физическая структура реализации интерфейса Xbox LPC позволяет пользователям, а также контрактному производителю Microsoft устанавливать надлежащим образом спроектированное устройство переопределения FLASH ROM без какой-либо пайки.

Очевидно, что возможность переопределять механизм доверия, используемый в Xbox, имеет липкие юридические последствия. В то время как моим намерением было в основном удовлетворить свое любопытство и, во-вторых, запустить свой собственный код на Xbox в соответствии с моими правами добросовестного использования, другие люди хотят копировать игры и изменять и распространять защищенный авторским правом код ядра Microsoft. Поскольку шифр слеп к своему применению, извлечение ключа RC-4 позволяет всем приложениям в равной степени. В результате я связался с Electronic Frontier Foundation (EFF), чтобы они помогли мне разобраться с юридическими проблемами. Юридический процесс медленный и громоздкий. Я извлек ключ в феврале 2002 года, и прошло почти до июня, прежде чем мне разрешили опубликовать результаты моего исследования на соответствующем академическом форуме.

Никогда я не испытывал столько шума из-за 128 бит. Закон об авторском праве в цифровую эпоху (DMCA) 1998 года навсегда изменил ландшафт взлома оборудования. Обратное проектирование раньше было защищенным актом, считавшимся частью того, что делает рынок здоровым и конкурентоспособным. Теперь вмешательство в криптографическую систему безопасности и ее обход для реализации ваших прав на добросовестное использование в уединении вашего собственного дома может обернуться для вас тысячами долларов штрафов и судебных исков. Я настоятельно рекомендую вам прочитать главу 12 «Осторожно, хакер», чтобы вы понимали свои законные права и обязанности.

## Безопасность через неизвестность

Техника, использованная Microsoft в первой версии безопасности Xbox, является прекрасным примером безопасности через неизвестность. Для шифрования образа ПЗУ использовался сильный шифр RC-4/128, чтобы помешать людям анализировать содержимое ПЗУ или создавать свои собственные ПЗУ. Однако RC-4/128 является симметричным шифром, что означает, что Xbox должен содержать ключ дешифрования, который также можно использовать в качестве ключа шифрования. Этот ключ дешифрования/шифрования является важной частью информации, спрятанной внутри секретного загрузочного ПЗУ. Скрытие этого ключа является безопасностью через неизвестность: как только ключ найден, шифр становится спорным, и вся безопасность теряется.

Настоящая безопасность потребовала бы, чтобы пользователь имел доступ к каждой отдельной части Xbox и все еще не мог зашифровать свой собственный действительный образ FLASH ROM. Это подразумевает, что некоторая тайна должна храниться за пределами Xbox. Криптография с открытым ключом была изобретена именно для этого сценария. Если бы

Microsoft использовала шифр с открытым ключом для шифрования или подписи загрузочного кода Xbox, то знание всего содержимого защищенного загрузочного ROM было бы бесполезным, поскольку главный секрет, закрытый ключ Microsoft, остается надежно вне нашей досягаемости в хранилище где-то в Редмонде, штат Вашингтон.

Однако есть и положительная сторона. В следующей главе представлены выводы моих коллег, многие из которых включают обнаружение бэкдоров в последовательности инициализации Xbox. Эти бэкдоры позволяют вам запускать свой собственный код на Xbox, не разрешая доступ к работам Microsoft, защищенным авторским правом, и не разрешая копирование игр. В следующей главе также будет представлена версия безопасности Xbox 1.1, которая была взломана всего за несколько дней Энди Грином в Великобритании.

## CHAPTER 9. Проникновение через заднюю дверь

Полный спектр возможных атак на Xbox слишком многочислен, чтобы описать его в этой книге. Xbox основан на архитектуре ПК, сложной, развитой архитектуре, изначально разработанной без учета безопасности. Многие из классических уязвимостей оборудования, используемых хакерами смарт-карт, такие как модуляция питания, атаки боковой полосы и сбой часов, даже не были затронуты в Xbox, насколько мне известно. (Более подробную информацию об этих уязвимостях безопасности можно найти в материалах конференции Cryptographic Hardware in Embedded Systems (CHES) в серии Lecture Notes in Computer Science (Springer-Verlag).

К сожалению, производители консолей и запущенных ПК не беспокоятся о слабостях аппаратной безопасности, поскольку атаки на оборудование «слишком сложны для выполнения среднестатистическим потребителем» и, следовательно, не представляют большой угрозы. Хотя верно, что для исследования атаки требуется опытный хакер с правильными инструментами, реализация атаки может быть очень демпевой и простой. Мне вспоминается притча, в которой механик, вызванный для ремонта важной части сломанной машины, проводит час, изучая ситуацию, и ремонтирует машину, постукивая по ней в нужном месте. Получив счет на 1000 долларов, владельцы машины требуют объяснить, почему кран стоит так много. Механик отвечает: «Кран стоит десять центов. Знание того, где нажать, стоит 999,90 долларов». Следствием этой притчи является то, что любой мог бы выполнить нажатие, чтобы починить машину, получив конкретные инструкции.

Атаки на безопасность часто одинаковы: их трудно вычислить, но легко распространить и реализовать. Производители запущенного оборудования также должны быть обеспокоены принятием в основном политики реагирования на хакерские вторжения. Многие хакеры работают тайно и держат свои методы и результаты втайне, чтобы поставщики не могли разработать надлежащие контрмеры. Эти хакеры также ведут библиотеку известных атак и бэкдоров, раскрывая только одну за раз, так что поставщики с политикой реагирования на оборудование всегда играют в догонялки.

## Комментарий к соглашениям об именовании

Хакерские сообщества часто изобретают собственную терминологию для важных концепций, которая может отличаться от сообщества к сообществу и от стандартной отраслевой терминологии. Ниже приведен список терминологии, принятой сообществом XboxLinux. Любые отклонения от терминологии, используемой мной в этой книге, отмечены.

- **X-код:** Коды операций таблицы Jam; коды операций, используемые секретным загрузочным ПЗУ южного моста (MCPX) для инициализации оборудования Xbox
- **2BL:** Второй загрузчик. Это код, который расшифровывается секретным загрузочным ПЗУ. Он называется вторым загрузчиком, потому что его основная обязанность — расшифровывать и распаковывать образ ядра.
- **Флэш-загрузчик:** В версии 1.1 безопасности это промежуточный загрузчик между секретным загрузочным ПЗУ и 2BL. FBL проверяется с помощью легковесного хеша по жестко закодированному значению в секретном загрузочном ПЗУ. В результате, FBL не может быть изменен без изменения кремния MCPX. FBL отвечает за проверку цифровой подписи на всех критических участках FLASH ROM.
- **Ядро:** Код ядра Xbox. Он хранится в сжатом и зашифрованном виде во FLASH ROM.
- **Безопасность версии 1.0:** Оригинальная система безопасности Xbox, использующая шифрование RC-4 на 2BL.
- **Безопасность версии 1.1:** Вторая система безопасности Xbox, использующая хеш TEA для проверки регионов FLASH ROM. Самая ранняя дата производства, указанная на коробках с версией безопасности 1.1, датируется августом 2002 года.

В предыдущей главе описывалась моя подслушивающая атака на механизм безопасности Xbox, которая в конечном итоге дала ключ RC-4, скрытый в блоке секретного кода. В этой главе описываются несколько других атак, доступных на Xbox, которые были разработаны моими коллегами, а также атака, которая была установлена на пересмотренной схеме безопасности Xbox, в дальнейшем именуемой версией безопасности 1.1.

## Задние двери и дыры в системе безопасности

Класс атак через бэкдор на Xbox использует фундаментальную уязвимость в способе инициализации оборудования с помощью секретного загрузочного кода.

Эта слабость обусловлена тем, что инициализация оборудования осуществляется с помощью мощного интерпретатора кодов операций Jam Table, который хранит свои команды в непроверенном открытом тексте.

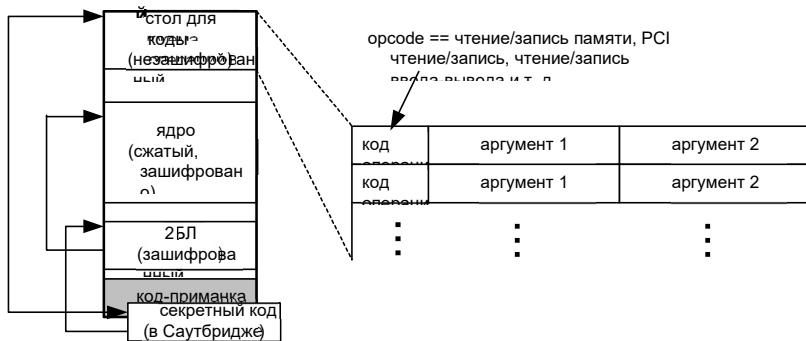
## Атаки на стол Visor Jam

Один из классов атак на Xbox включает в себя изменение последовательности инициализации оборудования. Напомним, что инициализация оборудования Xbox выполняется с помощью интерпретатора кодов операций, который извлекает свои команды из незашифрованной части FLASH ROM, называемой «таблицей jam». Связь записей таблицы jam с остальной частью FLASH ROM проиллюстрирована на рисунке 9-1. Записи таблицы jam хранятся как кортежи `<opcode, arg1, arg2>` рядом с самыми низкими адресами FLASH ROM.

Доступные коды операций включают функции чтения и записи памяти во все адресные пространства в архитектуре x86. Поскольку таблицы jam хранятся в незашифрованном виде и никогда не проверяются на предмет изменений, в таблицу jam могут быть вставлены коды операций, которые могут «заполнить» память Xbox вредоносными инструкциями или измененным состоянием оборудования перед расшифровкой RC-4 FLASH ROM 2BL.

Одним из применений модификации jam table является восстановление открытого текста ядра без знания ключа RC-4. Хакер, известный только как Visor, впервые описал мне этот подход. Вот краткое изложение подхода Visor:

1. Загрузите Xbox в обычном режиме. Часть обычного процесса загрузки поместит расшифрованный образ ядра в основную память.
2. Не отключая питание Xbox, переключите содержимое таблицы памяти FLASH ROM на альтернативную таблицу, которая копирует области основной памяти в легко контролируемое место, например, на шину FLASH ROM.
3. Выполните программный сброс процессора Xbox. Это заставит повторная инициализация оборудования без стирания основной памяти.

**ФЛЭШ-ПЗУ****содержани****(упрощены)**

**Рисунок 9-1:** Коды операций таблицы замятия по отношению к остальной части FLASH ROM.

4. Запишите содержимое основной памяти по мере выполнения измененной программы таблицы джемов.

Динамическое переключение содержимого FLASH ROM, необходимое на шаге 2, может быть выполнено с помощью эмулятора ROM или путем использования ROM увеличенного размера с избыточными адресными битами, подключенными к банку переключателей.

Visor также описал, как таблица jam может быть использована как часть сложного взлома для получения контроля над указателем инструкций Xbox (IP). Чтобы лучше понять этот взлом, мы более подробно рассмотрим, как секретный загрузочный код обрабатывает случай недействительного образа FLASH ROM.

После расшифровки 2BL в образе FLASH ROM секретный загрузочный код проверяет наличие магического числа в месте, близком к концу 2BL. Для недействительного образа FLASH ROM это число не совпадает и заставляет ЦП перейти к короткой последовательности инструкций, расположенных в 0xFFFF.FFFA. Этот набор инструкций продолжается до самого последнего адресуемого местоположения в физической памяти, местоположения 0xFFFF.FFFF. Как только ЦП выполняет эту последнюю инструкцию недействительного образа ROM, он должен аварийно завершить работу и остановить выполнение из-за ошибки границы сегмента кода, когда IP переходит с 0xFFFF.FFFF на 0x0000.0000. Однако этого не происходит; вместо этого ЦП с радостью пытается выполнить любую инструкцию, допустимую или недопустимую, расположенную в местоположении 0x0000.0000. Номинально эта инструкция недопустима, и ЦП в любом случае останавливается из-за ошибки инструкции. Однако допустимая инструкция может быть помещена туда во время инициализации таблицы Jam Xbox с помощью кода операции записи памяти Jam Table. Таким образом, повредив или удалив зашифрованный образ

FLASH ROM и изменив таблицы Jam для вставки инструкции перехода к вашему собственному незашифрованному коду во FLASH ROM, вы можете получить контроль над IP-адресом процессора Xbox, даже не касаясь шифра или любой подобной технологической меры, которая эффективно контролирует доступ к защищенной авторским правом работе. Следовательно, этот взлом может быть законным в соответствии с DMCA. Я говорю «может», потому что DMCA часто является неопределенным законодательным актом, и существует мало судебных precedентов для прояснения двусмысленностей. Аргумент в пользу законности этого подхода заключается в том, что никакой значительный код, защищенный авторским правом Microsoft, никогда не расшифровывается и не выполняется. Единственным исключением является часть секретного загрузочного ПЗУ, которая должна выполняться, поскольку она жестко запита в кремний южного моста. См. Главу 12 «Осторожно, хакер», для более глубокого обсуждения правовых вопросов, с которыми сегодня сталкивается сообщество хакеров.

## MIST преждевременная атака Unmap<sup>18</sup>

Чтобы защитить секретный загрузочный код в случае, если хакер получит контроль над Xbox, секретный загрузочный код в чипе Southbridge отменяет отображение себя незадолго до выхода. Другими словами, он навсегда скрывается от системы, когда завершает выполнение. Таким образом, пользовательская программа, пытающаяся получить доступ к любому из верхних 512 байтов в памяти, увидит блок-приманку во FLASH-памяти вместо секретного загрузочного кода. Майкл Стайл, руководитель проекта Xbox-Linux, обнаружил способ использовать эту функцию.

Процесс отмены сопоставления выполняется путем записи в 0x8000.8008, аппаратный регистр в пространстве конфигурации PCI. Основная стратегия заключается в том, чтобы включить код операции jam table, который записывает в 0x8000.8008 и отменяет сопоставление секретного загрузочного кода до завершения последовательности инициализации. Поскольку кэши в это время отключены, процессор начнет извлекать и выполнять инструкции из блока-приманки. К счастью, поскольку блок-приманку можно свободно изменять, поскольку он является частью FLASH ROM. Однако загвоздка в том, что интерпретатор jam table блокирует запись в расположение 0x8000.8008, поэтому это не должно работать. Однако ошибка в декодировании пространства конфигурации PCI в чипсете южного моста заставляет инструкцию отмены сопоставления реагировать на несколько псевдонимов адресов. В частности, битовое поле «function» игнорируется. Таким образом, запись в 0x8000.8X08, где X не равен 0, также делает свое дело, и эти записи не блокируются интерпретатором jam table. Поэтому, чтобы получить контроль над IP-адресом ЦП с помощью взлома MIST, вы должны изменить блок-приманку во FLASH, чтобы он содержал ваш код, а затем добавить соответствующий

<sup>18</sup>Из презентации Энди Грина на 19-м ежегодном конгрессе Chaos Communication, посвященной взлому безопасности Xbox.

код операции `jam table`, чтобы отменить сопоставление секретного загрузочного ПЗУ во время инициализации оборудования.

## Microsoft принимает ответные меры

Обнаружение уязвимостей в системе безопасности заставило многих предположить, что Microsoft быстро сменит схему безопасности. В августе 2002 года Xbox с новой материнской платой тихо начали появляться в Австралии. Первое официальное сообщение о новой системе безопасности пришло из неожиданного источника: от nVidia, производителя чипсетов, используемых в Xbox. После невыразительного второго квартала 2002 года представитель nVidia назвал это последней из нескольких причин, по которым квартал прошел плохо:

«Мы говорили о Xbox, что достигли рубежа скидок за объем, что еще больше снизило маржу. И что мы будем списывать инвентарь во втором квартале, связанный с количеством Xbox MCP, которые стали устаревшими, когда MSFT перешла на новый код безопасности (через хакера MIT), и излишками чипсетов nForce, которые мы создали в ожидании более высокого спроса на ПК на базе Athlon». — Дерек Перес, директор по связям с общественностью nVidia<sup>19</sup>

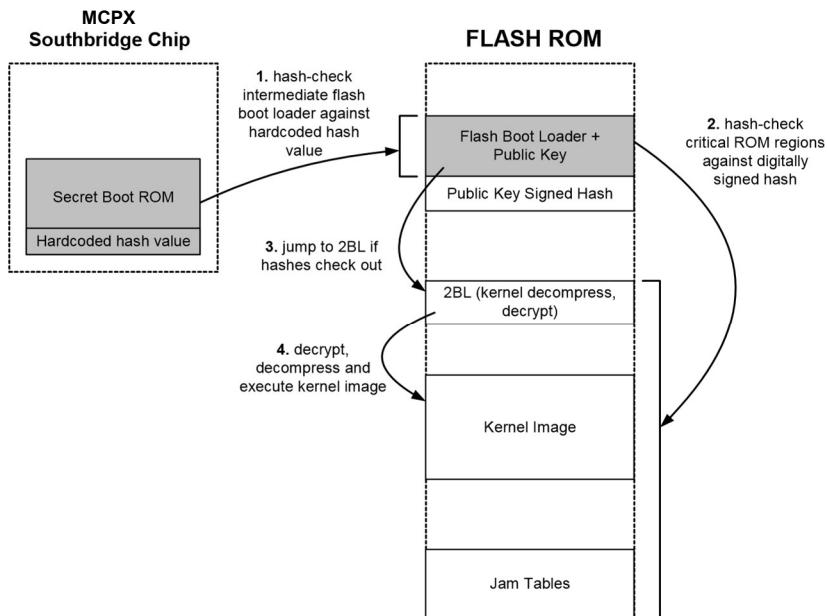
## Обратная разработка v1.1 Безопасность<sup>20</sup>

Специфика изменений кода безопасности не была раскрыта до октября 2002 года, когда хакер по имени Энди Грин начал исследовать первую версию Xbox 1.1, доступную в Соединенном Королевстве. Внешние физические различия между версиями Xbox 1.1 и 1.0 на материнской плате были едва заметны: графический процессор поменял свой вентилятор на более крупный радиатор, дочерняя плата USB была объединена с материнской платой, а чип синтезатора тактовой частоты PLL отсутствовал. Также отсутствовали конденсаторы фильтров здесь и там, но ничего существенного, похоже, не изменилось. Дальнейшее расследование показало, что дыра, использованная атакой MIST, была закрыта, но коды операций таблицы застревания остались неизменными. Шина LPC, ключевой вектор для получения доступа к Xbox, также присутствовала и не менялась.

<sup>19</sup>Из статьи Inquirer, <http://www.theinquirer.net/?статья=4735>

<sup>20</sup>Из презентации Энди Грина на 19-м ежегодном конгрессе Chaos Communication, посвященной взлому безопасности Xbox.

Энди смог извлечь MCPX ROM за один день, используя процедуру, придуманную его коллегой-хакером по имени Asterisk. Процедура использовала нераскрытою комбинацию ранее выявленных уязвимостей и бэкдоров. Первоначальный анализ содержимого ROM показал, что безопасность была реализована радикально иным образом. Беглый обзор версии 1.1 Xbox показал, что старая схема безопасности через неизвестность была выброшена и заменена схемой, которая номинально получает свою безопасность от прочности шифров с открытым ключом.



**Рисунок 9-2:**Xbox Security версии 1.1. Регионы, которые нельзя изменить без замены микросхемы MCPX, закрашены серым цветом.

Однако реализация новой схемы безопасности Microsoft была немного нелогичной. Поскольку секретное загрузочное ПЗУ в MCPX оставалось того же размера, 512 байт, они не могли вместить полный алгоритм цифровой подписи с открытым ключом в секретное загрузочное ПЗУ. Вместо этого они решили использовать облегченный хэш в секретном загрузочном ПЗУ для проверки области FLASH ROM, названной Flash Boot Loader (FBL). FBL содержит код (шифр RSA, хэш SHA-1, открытый ключ Microsoft и программу драйвера) для проверки цифровой подписи FLASH ROM. FBL выполняется только в том случае, если хэш FBL можно проверить с помощью константы, хранящейся в секретном загрузочном ПЗУ. Таким образом, FBL теоретически так же неизменяем, как и секретное загрузочное ПЗУ, даже если он хранится в изменяемом FLASH ROM.

Хотя эта схема звучала довольно надежно, хакерское сообщество не сдалось так легко. Они подробно изучили хэш секретного загрузочного ПЗУ и

обнаружили, что он основан на TEA, крошечном алгоритме шифрования Дэвида Уиллера и Роджера Нидахэма из компьютерной лаборатории Кембриджский университет. Франц Ленер, участник проекта, отправил запрос в новостную группу sci.crypt относительно уязвимостей в хэше TEA.

В пятницу днем, 11 октября, на их запрос был дан ответ. В статье, написанной Джоном Келси, Брюсом Шнайером и Дэвидом Вагнером, представленной в CRYPTO 1996, указывалось, что шифр TEA имеет уязвимость в своем планировщике ключей, где каждый ключ имеет три связанных ключа, которые могут быть сгенерированы путем инвертирования определенных пар битов (этот уязвимость, наряду с шифром TEA, более подробно обсуждается в Главе 7). В субботу Энди Грин опубликовал это на XboxHacker.net:

Ой, я простой смертный, мои ноги определенно сделаны из глины.

Хорошо, я не думаю, что это слишком много выдаст, если сказать первые 5 байт региона. fffffd400: E9 83 01 00 00 Это относительная длинная ветвь к 0xfffffd588

Если я переверну b31 этого как DWORD (и переверну его друга в DWORD адрес +1 таким же образом) Вместо этого я перехожу на 0x7fd588.... Хммм, это что, 8 Мб вверх, где...

где есть  
оперативная память

Xcodes.. метод Visor Ram Push... (смотрит на MCPX для RAM  
Напишите X-код)

X-Code код операции 3 ... неограниченный

Держитесь за шляпу, ребята! Настало время испытаний!

(мне кажется, это чудесное и своевременное откровение, я чувствую, как таинственные и невидимые силы помогают мне, за что я им благодарен!)<sup>21</sup>

Другими словами, слабость связанного ключа шифра TEA означает, что каждая смежная пара двойных слов в FBI может быть изменена одним битом, самым значимым битом, без каких-либо изменений в результирующем хэше. Эта слабость дала Энди и его команде достаточно места для изменения цели

---

<sup>21</sup>Публикация с www.xboxhacker.net в разделе Xbox Hacker BBS- > Взлом Xbox (ТЕХНИЧЕСКАЯ ИНФОРМАЦИЯ) -> BIOS/Flash ROM/Прошивка -> Новости от команды Xbox Linux, MS «устроили переполох», внутренности вылезли наружу.

одиночной инструкции перехода, чтобы указать на местоположение в основной памяти.

## Профиль: Энди Грин

**Можете ли вы рассказать нам немного больше о себе и о том, как вы начали заниматься хакерством?**

Мне 37 лет, я живу в Англии, недалеко от Кеттеринга в Восточном Мидленде, с женой, четырьмя детьми и двумя кошками.

Я интересуюсь компьютерами с 12 лет или около того, когда мой брат купил Commodore Pet. Этот 1 МГц 6502 занимал меня месяцами и месяцами, пытаясь написать сначала код BASIC, набранный из журналов, а затем игры для него; в конце концов я написал фантастическую штуку Space Invaders с символными ячейками в машинном коде. Машинный код — это то, где вы фактически программируете процессор непосредственно в шестнадцатеричном формате; я до сих пор помню общие коды операций 6502 в шестнадцатеричном формате. Это было настолько трудным усилием, что я решил, что моим следующим проектом будет ассемблер, написанный в машинном коде. 1978 год был до эпохи Интернета: я не мог позволить себе коммерческие ассемблеры, потому что я был всего лишь ребенком, и вокруг нас не было никого, у кого я мог бы сделать копию.

Это было довольно жалко, как это бывает с ассемблерами, но работало отлично. Из этого я узнал ценность наличия правильных инструментов, я мог писать гораздо быстрее на Ассемблере, и целые виды ошибок, например, неправильное вычисление относительных ветвей вручную, полностью исчезли. Затем у меня появился компьютер BBC Model B, и мне снова было интересно делать инструменты и игры. Мне предложили стипендию в государственной школе, но я отказался и вместо этого бросил школу в 16 лет, не получив дальнейшего образования. Я был вполне доволен тем, что мог самостоятельно научиться всему, что меня интересовало.

Я продал несколько игр для этой и другой платформы 6502 под названием Oric, и на эти деньги основал компанию, производящую ассемблеры и другие инструменты разработки. По пути я выучил С и C++, и каждый раз, когда я продвигался, целые горы ошибок и тратящих время мучений исчезали.

(продолжение)

Это единственное местоположение может быть предварительно загружено с последующей инструкцией перехода обратно в любой фрагмент пользовательского кода, используя ранее обсуждавшиеся коды таблицы джема. Сообщество хакеров Xbox объединилось в героическом усилии и взломало версию безопасности Xbox 1.1 за три дня. Отдельная попытка, не менее отважная, от Xecuter также взломала безопасность за тот же промежуток времени.

Первая мораль этой истории заключается в том, что безопасность настолько сильна, насколько сильна ее самая слабая связь. Хотя нет никаких сомнений в надежности шифра RSA и хэша SHA-1 для цифровой подписи, они были не

единственными элементами системы безопасности. Шифр TEA, используемый для расширения сферы доверия защищенного загрузочного ПЗУ в FLASH-ПЗУ, имел недостатки, которые позволяли хакерам обходить сильные алгоритмы цифровой подписи.

Это приводит нас ко второй морали: сложность порождает слабости. Сложные системы трудно проектировать, тестировать и анализировать. Версия безопасности 1.1 для Xbox, вероятно, была реализована на коротком запале, поэтому

Это похоже на ту картинку из «Восхождения человека» — от вычисления относительных ветвей Нетандерала до Homo Erectus с его виртуальными функциями.

Параллельно с этим я начал изучать проектирование цифрового оборудования, снова обучаясь на собственном опыте. Я обнаружил, что оборудование и программное обеспечение — это две стороны одной медали, хотя в образовании они рассматриваются совершенно отдельно. На самом деле это деталь реализации, решаете ли вы сделать свою логическую функцию в программном обеспечении или в оборудовании, или в некоторой смеси того и другого. Нахождение в обоих лагерях дает более глубокое понимание природы проектирования: например, можно сказать, что C++ заимствует многие концепции из электроники с точки зрения важности интерфейсов.

Незадолго до того, как я заинтересовался Xbox, я работал в американской компании с офисом в Оксфорде, выполняя много заданий, но последнее из них было связано с разработкой кремния для смарт-карт. Хотя дизайн был интересным, и там работали несколько замечательных людей, я все больше впадал в уныние из-за политики и проблем с руководством. Не помогло и то, что, несмотря на то, что я был занят на нескольких проектах, мне платили 2/3 зарплаты сотрудников в Сан-Хосе просто потому, что я работал в Великобритании. И не заставляйте меня начинать о патентах, которые они получили от меня без вознаграждения. В декабре 2001 года я обнаружил, что честность важнее денег, уволился и решил вернуться к работе на себя.

Я был довольно измотан некоторыми неприятными переживаниями, когда уходил из этой компании, пока переваривал их, я обнаружил, что меня зацепила огромная разница во взглядах между уродливыми, хватательными, контролирующими инстинктами вашей средней компании, занимающейся интеллектуальной собственностью, и природой проектов GPL и людьми, вовлечеными в поощрение снижения строгости патентных и авторских законов. Со временем я все больше стал видеть Microsoft и предыдущую компанию

(продолжение)

не было достаточно времени для анализа системы на предмет слабых мест. Либо это, либо Microsoft знала о слабости TEA и разработала этот черный ход в систему, чтобы снизить риск блокировки своего FBL в кремнии. Кажется довольно сомнительным, что Microsoft намеренно включила этот черный ход, поскольку модификация кремния MCPX — очень дорогое удовольствие (хотя расходы в конечном итоге легли на счета nVidia). С другой стороны, сложности трудно избежать. Мой научный руководитель в МТИ Том Найт однажды сказал мне: «В этом мире есть два вида конструкций: те, которые полезны, и те,

правильность которых вы можете формально доказать». В некоторой степени единственный способ обеспечить безопасность реальной системы — сделать ее детали открытыми (никакой безопасности через неизвестность!) и подвергнуть систему анализу со всех сторон. В некотором смысле, тщательный анализ безопасности Xbox проводится бесплатно для Microsoft, благодаря сообществу хакеров.

Профильт: Энди Грин (продолжение) Я  
работал в том же ключе.

Именно после этого я прочитал о взломе Банни на Slashdot. Я прочитал о методах Банни с некоторыми едкими эмоциями. Моими главными мыслями были: это то, что я мог бы сделать, поскольку я использовал ПЛИС, которые использовал Банни с 1989 года, восхищение кратостью атаки и смущение от себя, что я не делал что-то столь же крутое и интересное — и что соответствовало бы моим философским пристрастиям — в свое время. Вместо этого я сидел там, читал Slashdot, пил кофе, ничего не внося. (Кстати, я думаю, что это довольно распространенный опыт для многих читателей Slashdot — немного завидовать и испытывать сомнения, когда они читают о чьих-то крутых взломах. Я думаю, это объясняет постоянный фоновый шум насмешек и вопросов, почему кто-то хочет сделать такое.)

В течение следующих нескольких недель я собрал как можно больше информации о внутренних компонентах Xbox; Xboxhacker.net был для этого решающим. Там же я познакомился с Майклом Штайлом, когда проект Xbox Linux только начинался. Довольно скоро я смог определить интересные проекты, в которые я мог бы внести свой вклад, например, проект Milkosor. Опять же, благодаря этому, с помощью Surferdude, мне стало возможным собрать самый первый чистый ROM, который мог загружаться и поддерживать Xbox без сброса. Позже это стало основой сром 1MB Linux и cromwell, чистого ROM Xbox Linux. После первоначальных хаков и разработок я решил работать почти полностью над целью Xbox Linux.

#### Можете ли вы рассказать нам, зачем вы взламываете Xbox?

Почему? У всех разные причины, но для меня это было мое понимание возмутительного антимонопольного поведения Microsoft — все отрицать, все обжаловать, все откладывать, и в то же время создавать и выбрасывать (потому что они продаются по цене ниже себестоимости) на рынок миллионы персональных компьютеров только Microsoft — Xbox. Поскольку нашим представителям здесь, в Европе и США, похоже, все равно (возможно, как это было недавно в ЕС, потому что они планируют пойти работать в Microsoft

(продолжение)

Даже если бы Microsoft использовала более сильную хэш-функцию в секретном загрузочном ПЗУ, все еще есть ряд жизнеспособных атак на Xbox, которые еще предстоит опробовать. Можно провести атаку «человек посередине» на шину HyperTransport (см. Главу 8), перегружая сигналы тщательно рассчитанными импульсами. Этую атаку довольно просто реализовать, поскольку каждая трасса шины HyperTransport имеет

контрольную точку, видимую со стороны компонентов материнской платы. Полное аппаратное решение включало бы ПЛИС на плате с тестовыми разъемами типа «*romo pin*» в стиле «ложе из гвоздей». Эту плату можно напечатать на этих контрольных точках без какой-либо пайки. А другая атака, предложенная мне Ади Шамиром на конференции CHES, заключается в использовании синхронизированного сбоя в часах ЦП или блоке питания, чтобы нарушить расчет целевого адреса перехода. Этот вид атаки был успешно применен к процессорам в криптографических смарт-картах. Опять же, этот вид атаки можно реализовать довольно легко и дешево в виде устанавливаемого пользователем модуля. (Имейте в виду, что хакерам доступен гораздо более широкий спектр атак, если их целью является единовременное нарушение безопасности для восстановления, например, секретного ключа или блока критического кода.)

## Угроза лазеек

Как показала эта глава, поиск бэкдоров является практическим методом атаки на криптографически запищенное оборудование. Относительно высокий уровень успеха в поиске бэкдоров в Xbox частично объясняется тем, что

и забрать их серебряные монеты), было бы честью быть частью прокалывания этого злого плана раздутой монополии с помощью оружия GPL и Linux. Я знаю, что люди закрывают глаза и думают о своих опциях на акции, но должно быть трудно для порядочных людей — и, конечно, большинство людей, работающих там, именно такие — работать на такого монстра.

Мне повезло получить пару контрактов до 2002 года, что позволило мне провести вторую половину года, работая исключительно над получением первого ядра Linux в сгом и над тем, чтобы Cromwell мог управлять основными периферийными устройствами коробки и загружать Linux с HDD или CD. С тех пор моя доля призовых денег Project A (благодаря спонсору Майклу Робертсону) позволит мне продолжать работать полный рабочий день, по крайней мере, в течение следующих нескольких месяцев.

### Хотите ли вы поделиться советом?

Моя последняя мысль — поощрять людей, особенно молодых, прислушиваться к своему мозгу, когда дело касается вещей, которые их интересуют. Не бойтесь копаться и пытаться узнать о вещах, которые привлекают ваше внимание. Это чувство, которое вы испытываете, когда хотите что-то понять, своего рода тоска, — это способ вашего мозга сказать вам, что, по его мнению, эти знания могут пригодиться позже. Если вы будете слушать его достаточно часто, у вас будет хороший шанс узнать нужную вещь в нужное время, чтобы что-то изменить.

## Профиль: Франц Ленер

Франц Ленер, 29 лет, живет в Австрии со своей девушкой. Он изучал электротехнику 5 лет. Сейчас он программирует «автоматизированные решения», управляя интернет-провайдером. В свободное время он ищет проекты, которые одновременно и развлекательные, и познавательные.

После того, как он нашел документ по взлому Xbox от Bunnie, он встретил команду Xbox-Linux на sourceforge.net. Он присоединился к проекту XboxLinux, чтобы узнать о командном программировании, взломе и отладке ядра Linux и криптографических системах. Он также присоединился к проекту Xbox-Linux, чтобы лучше понять связанные системы, такие как Palladium.

Xbox представляет собой первую значимую попытку поставщика криптографически защищить ПК. Несмотря на уроки, извлеченные из опыта Xbox, будущие безопасные реализации ПК по-прежнему подвержены риску наличия уязвимостей аппаратной безопасности, поскольку наследием ПК является открытая и незащищенная аппаратная архитектура.

Аппаратное обеспечение ПК сложное, но хрупкое, и выстроить цепочку доверия из него сложно из-за этой хрупкости. По сути, каждый компонент ПК спроектирован так, чтобы «доверять» своей физической среде. В спецификациях любого коммерческого компонента интегральной схемы четко указано, что ИС гарантированно будет работать в ограниченном диапазоне температур, напряжений, частот и других условий. Если эти максимальные значения нарушаются, то поведение устройства «не определено», и все ставки отменяются. Большинство инженеров-разработчиков микросхем даже не рассматривают возможность заставить свои схемы изящно восстанавливаться после состояния, выходящего за пределы диапазона, поскольку и так достаточно сложно заставить микросхему работать в указанных рабочих условиях. Кроме того, большинство потребительских приложений очень чувствительны к стоимости, а накладные расходы на создание надежных мер отказоустойчивости приводят к тому, что продукт не является конкурентоспособным по цене.

Таким образом, чипы обычно реализуются без внутренней проверки ошибок. Если по какой-то причине арифметико-логическое устройство (АЛУ, вычислительный «мозг» ЦП) неправильно складывает два числа, проблема проявится только симптоматически; вы можете наблюдать только последствия такой ошибки, иногда спустя долгое время после события, вызвавшего ошибку. Можно представить атаки, которые используют ошибки, вызванные условиями выхода за пределы диапазона, как аналогию переполнения буфера в мире программного обеспечения.

Другая проблема архитектуры ПК заключается в том, что процессор слишком доверяет своей кодовой среде. Архитектура процессора Pentium не имеет аппаратного обеспечения для различения небезопасного и безопасного кода.

Если указатель инструкций случайно попадет в небезопасный сегмент кода из-за ошибки или вызванного сбоя, процессор с радостью выполнит этот код.

### Примечание



**Разделение кода на основе уровней аппаратной безопасности** — это техника, отличная от песочницы. Песочница не обеспечивает адекватного решения для ситуаций, когда пользовательская программа требует направления от секретного или защищенного кода или данных или взаимодействия с ними. В последнее время были предложены новые архитектуры процессоров, которые могут решить эту проблему с помощью тегов данных, которые встраивают своего рода журнал аудита безопасности.<sup>22</sup>

Другим источником бэкдоров являются ошибки проектирования, которые существуют в каждом сложном чипе. Обычной практикой является поставка чипов с большим количеством известных ошибок, также известных как ошибки. Например, процессор Intel i860 XP (впервые выпущенный в 1991 году, не путать с недавно выпущенным чипсетом i860 для процессора Pentium4) поставлялся с книгой ошибок, которая была сопоставима по размеру с техническим описанием процессора. Другой пример, более близкий к дому, — ошибка в декодере адресного пространства nVidia MCPX, которая сделала возможной атаку MIST Premature Untrap. Большинство этих ошибок имеют простые обходные пути или оказывают незначительное влияние на функциональность чипа в номинальных условиях. Однако некоторые ошибки, например, те, которые связаны с когерентностью кеша, декодированием адресов и управлением памятью, могут привести к серьезным уязвимостям программного обеспечения.

В случае Xbox влияние на бизнес аппаратного бэкдора, вероятно, невелико. Возможно, Microsoft теряет небольшую часть дохода от продаж игр, но потери из-за пиратства не идут ни в какое сравнение с потерями, которые Microsoft несет от продажи оборудования. Кроме того, Xbox — это всего лишь игровая консоль — из-за уязвимостей безопасности Xbox не выкачивают деньги со счета бабушки или не крадут номера кредитных карт. Однако с доверенным ПК под угрозой окажется не только доход от игр. Если архитектура доверенного ПК не станет фундаментальным изменением по сравнению с устаревшими ПК, люди будут слепо доверять финансовые секреты и безопасность личных данных ненадежному оборудованию.

Как и во многих вещах в жизни, первым шагом является образование. Чем больше мы узнаем о безопасности оборудования, даже если это подразумевает

---

<sup>22</sup><http://www.ai.mit.edu/projects/aries/Documents/Memos/ARIES-15.pdf>.

«Минимальная доверенная вычислительная база для динамического обеспечения безопасного потока информации», Том Найт и Джереми Браун.

ковыряние в игровой консоли, тем лучше будут наши системы безопасности завтра. А теперь продолжаем урок...



# Больше проектов по оборудованию

Сходство архитектуры Xbox с архитектурой ПК позволяет хакерам заимствовать технологии и опыт из мира ПК при создании аппаратных проектов. В результате, аппаратное обеспечение ПК, мониторы, кабели и периферийные устройства были адаптированы для работы с Xbox. В этой главе представлены некоторые из этих аппаратных проектов, обнаруженных, задокументированных и реализованных хакерами по всему миру.

## Интерфейс LPC

Версия 1.0 интерфейса LPC (Low Pin Count) была определена Intel в 1997 году. Интерфейс LPC — это бесплатная шина, которая предназначена для включения систем без явных возможностей ISA или X-bus (ISA-подобная шина расширения для памяти или универсальных устройств ввода-вывода). Потребность в интерфейсе LPC обусловлена большим количеством устройств с низкой скоростью передачи данных и большим количеством выводов, а также шин с несовместимыми интерфейсами, которые можно найти в стандартном ПК, например, интерфейсы дисководов, клавиатуры, мыши, последовательного порта, IrDA, параллельного порта, ISA и загрузочного ПЗУ. Совокупная пропускная способность, потребляемая всеми этими устройствами, невелика, но количество сигналов, необходимых для поддержки всех из них, легко превышает количество сигналов, требуемое шинами с более высокой пропускной способностью, такими как шина PCI или AGP. Что еще хуже, не все конфигурации компьютеров требуют всех этих устаревших устройств ввода-вывода, а неиспользуемые выводы и функции просто съедают прибыль. Стоимость вывода на корпусе чипа высока по сравнению со стоимостью кремния, необходимого для поддержки этих простых интерфейсов. (Практическое правило заключается в том, что один вывод корпуса стоит копейку, в то

время как в кремнии с размерами 0,13 мкм около десяти тысяч вентиляй — достаточно логики для реализации небольшого процессора — стоят копейки на площасти кремния, если предположить, что конструкция не ограничена контактными площадками<sup>1</sup>.)

Интерфейс LPC решает эту проблему с помощью одной шины с малым количеством выводов (требуется семь выводов по сравнению с 36 выводами, необходимыми для шины ISA), которая работает на высокой скорости. Все устаревшие функции ввода-вывода и расширения отображаются в этой шине с высокой пропускной способностью, что позволяет разработчикам систем создавать так называемые чипы «Super I/O», которые, в свою очередь, позволяют использовать чипы Southbridge с гораздо меньшим количеством выводов. Кроме того, разделение функций между чипами Super I/O и чипами Southbridge позволяет разработчикам выбирать комбинации чипов Super I/O и Southbridge, которые обеспечивают оптимальный набор функций для данного приложения.

Физический интерфейс LPC довольно прост. Интерфейс представляет собой 4-битную двунаправленную шину, работающую на тактовой частоте 33 МГц. Интерфейс также имеет два «боковых» сигнала: один сигнал кадрирования, который указывает на начало и конец циклов шины LPC, и один сигнал сброса, который заставляет все периферийные устройства LPC перейти в известное состояние для целей инициализации. Кроме того, есть несколько дополнительных сигналов для интерфейса LPC, которые обеспечивают возможности DMA и прерываний, а также управление питанием для более сложных устройств ввода-вывода. (Более подробную информацию о шине LPC и ее протоколе можно найти в спецификации интерфейса Intel Low Pin Count (LPC), версия 1.1.

Спецификацию можно найти на корпоративном сайте Intel по адресу <http://www.intel.com/design/chipsets/industry/lpc.htm>.)

## Интерфейс LPC на Xbox

Xbox включает в себя интерфейс LPC на материнской плате. Интерфейс LPC в этом случае используется для реализации отладочной и тестовой шины. Через этот интерфейс LPC можно подключить клавиатуру и мышь, а также альтернативное загрузочное ПЗУ для диагностических целей. Интерфейс LPC активируется для загрузки альтернативного загрузочного кода, когда FLASH ROM на Xbox недоступен. Отсутствие устройства FLASH ROM можно смоделировать, принудительно установив самый низкий бит данных (D0) шины данных FLASH ROM на уровень ноль вольт.

Многие предполагают, что интерфейс LPC является неотъемлемой частью производственной линии Xbox из-за возможности альтернативной загрузки ROM, предоставляемой интерфейсом LPC. Полностью собранные Xbox могут

быть сконфигурированы с помощью комплексной программы самотестирования через интерфейс LPC. Применение ЦП в качестве быстрого тестового контроллера позволяет быстро и эффективно изолировать дефектные блоки на заводе без затрат на дорогостоящие испытательные машины.

Для хакеров альтернативная загрузочная ПЗУ, предоставляемая интерфейсом LPC, является идеальным механизмом для ввода кода в Xbox. Действительные загрузочные образы ПЗУ LPC для Xbox могут быть созданы кем угодно, поскольку криптографически защищенная процедура загрузки Xbox теперь полностью понятна. Фактически, некоторые поставщики альтернативных загрузочных ПЗУ для Xbox использовали регулярность геометрии выводов интерфейса LPC на Xbox

<sup>1</sup>Схемы на чипе обычно окружены квадратами металла («контактными площадками»), которые соединены проводами с контактами на корпусе чипа. Чип называется ограниченным контактными площадками, когда площадь, необходимая для кольца контактных площадок, превышает площадь, требуемую схемой внутри чипа. Стоимость избыточных контактов становится еще выше в случае, если чип ограничен контактными площадками.

материнская плата для создания устройств ПЗУ, которые устанавливаются без пайки. Эти устройства используют набор подпружиненных «пружинных штифтов», похожих на те, которые использовались во время производства для тестирования Xbox, для контакта с интерфейсом LPC с помощью простого прижима. (Распиновку шины LPC, реализованную на Xbox, можно найти в Приложении F, «Справочник по оборудованию Xbox».)

## Использование интерфейса LPC

Тот факт, что интерфейс LPC является отраслевым стандартом, весьма удобен для хакеров оборудования Xbox. Во-первых, существует множество совместимых с LPC интерфейсных устройств, от микросхем Super-I/O до встроенных ПЗУ со встроенными интерфейсами LPC. Во-вторых, широкое признание интерфейса LPC в качестве диагностической и удобной шины для обычных ПК помогает снизить юридический риск использования интерфейса LPC и продажи устройств интерфейса LPC. Встроенное ПЗУ для интерфейса LPC может продаваться без какого-либо специфического для Xbox содержимого, поскольку конечные пользователи могут легко перепрограммировать свои устройства шины LPC с помощью простого и дешевого адаптера для своего ПК. Еще одним подспорьем в легальности устройств прошивки LPC является то, что распиновка разъема LPC Xbox почти идентична распиновке, рекомендованной Intel для обычных ПК. В результате устройство прошивки LPC, продаваемое для Xbox, очень похоже на устройство прошивки LPC, продаваемое для стандартного ПК.

Одно из первых устройств LPC boot ROM было разработано Энди Грином. Проект называется «Cheapmod» и представляет собой устройство SST 49LF020

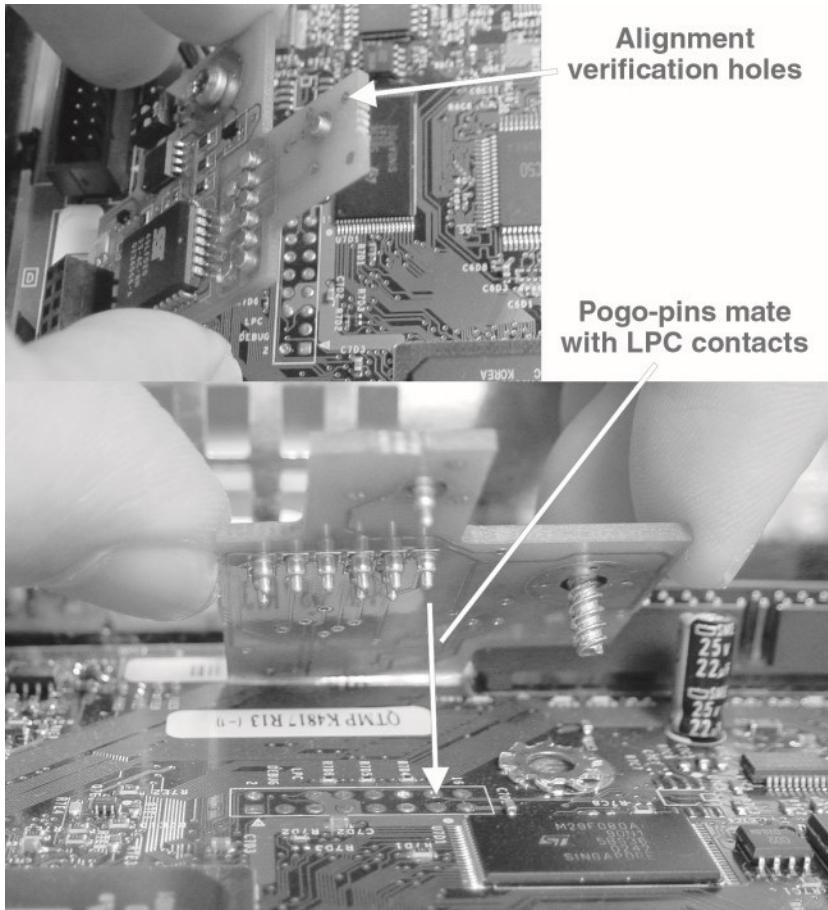
(256 кбайт FLASH ROM с интегрированным интерфейсом LPC) в разъеме, подключенным к LPC-совместимому разъему. Согласно веб-странице Энди Cheapmod,

«<http://warmcat.com/milksop/cheapmod.html>», «Если вы можете получить Имея SST 49LF020 за 2,50 доллара, вы можете собрать альтернативный BIOS за 4 доллара». Это устройство можно запрограммировать с помощью его программатора «CheapLPC».

(<http://warmcat.com/milksop/cheapLPC.html>), восхитительно простое устройство на базе параллельного порта ПК, которое может (медленно) общаться с устройством LPC и перепрограммировать его. Многие коммерчески доступные альтернативные устройства прошивки были основаны на его дизайне или вдохновлены им, включая дизайн Xodus/Matrix. Xodus/Matrix — особенно интересный вариант оригинального дизайна Энди, поскольку это было первое устройство альтернативной прошивки Xbox, в котором была реализована полностью беспаечная процедура установки. Это открыло мир взлома Xbox для хакеров, ориентированных на программное обеспечение, которые не были склонны припаивать провода к своим Xbox. (Фотография Xodus/Matrix представлена на рисунке 10-1.) Устройство Xodus/Matrix поставляется без какого-либо запрограммированного в нем кода; пользователь должен предоставить альтернативный образ прошивки.

При выборе следует учитывать некоторые важные функциональные соображения.

Микросхема FLASH ROM с интерфейсом LPC для использования с Xbox. Наиболее существенным является то, что собственная архитектура Xbox выделяет область размером 16 МБ для загрузочного ПЗУ. Если физическое загрузочное ПЗУ меньше 16 МБ, содержимое загрузочного ПЗУ алиасом заполняет все пространство в 16 МБ. Это дает разработчикам Xbox больше гибкости в выборе размера микросхемы ПЗУ, не вызывая проблем с процедурами, которые используют как нижнюю, так и верхнюю относительную адресацию.



**Рисунок 10-1:** Альтернативное устройство прошивки Xodus/Matrix без пайки, демонстрирующее подпружиненные контакты «родо-pin», которые обеспечивают беспаечное подключение к разъему LPC на материнской плате Xbox.

Давайте конкретизируем концепцию адресации снизу и сверху на примере. Адреса для области загрузочного ПЗУ размером 16 МБ в Xbox находятся в диапазоне от 0xFF00.0000 до 0xFFFF.FFFF. Программы на Xbox, использующие адресацию снизу, будут вычислять адреса, используя  $0xFF00.0000 + \text{смещение}$  (нижний адрес плюс смещение), в то время как программы, использующие адресацию сверху, будут использовать  $0xFFFF.FFFF - \text{смещение}$  (верхний адрес минус смещение). Предположим, что в Xbox установлено загрузочное ПЗУ объемом 1 МБ. Это означает, что процессор увидит 16 идентичных копий этого ПЗУ объемом 1 МБ, равномерно распределенных по адресному пространству ПЗУ объемом 16 МБ. Другими словами, содержимое загрузочной памяти будет выглядеть идентичным для

каждого адреса  $A + 0xFF00.0000 + n * 0x0010.0000$ ,  $n =$  от 0 до 15,  $A =$  от 0 до  $0x000F.FFFF$ . В результате программисты могут упаковать данные в меньшее загрузочное ПЗУ объемом 1 МБ, используя как адресацию сверху, так и адресацию снизу, не изменения при этом свой код: Действительная копия образа ПЗУ

## Альтернативные устройства прошивки против модчипов

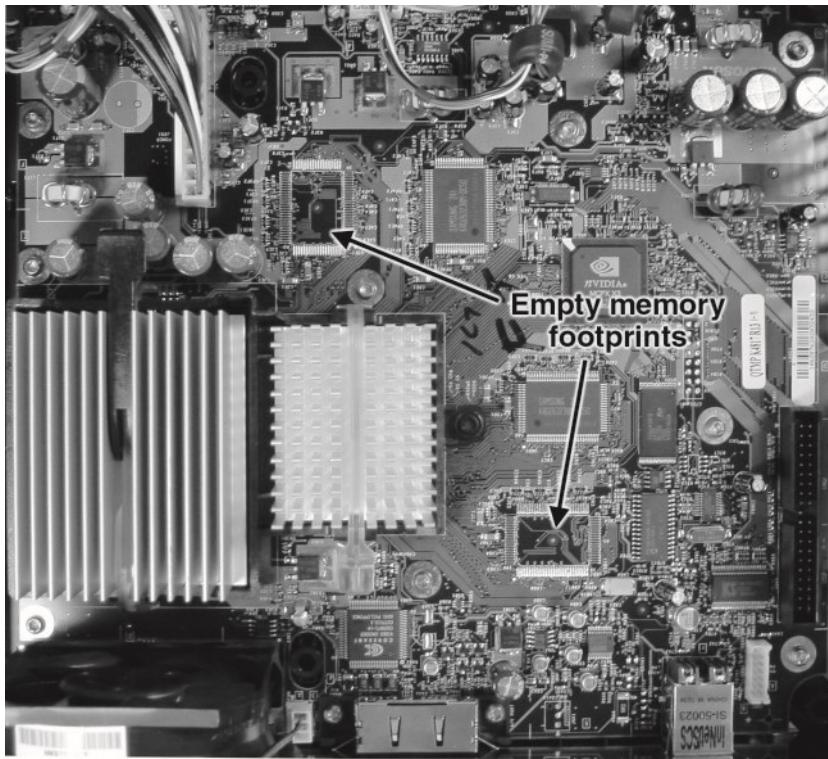
Альтернативное устройство прошивки — это аппаратный модуль, который предоставляет метод для запуска указанной пользователем прошивки на оборудовании Xbox. Альтернативные устройства прошивки отличаются от так называемого «модчипа» тем, что альтернативное устройство прошивки поставляется как пустое устройство и не имеет встроенной возможности обходить механизмы контроля авторских прав. Например, пустое устройство ПЗУ с LPC-интерфейсом — это альтернативное устройство прошивки: вы можете записать на него копию Билля о правах США, если захотите. Любая установленная пользователем FLASH ROM, которая поставляется пустой, также является альтернативным устройством прошивки. С другой стороны, модчип в разговорной речи подразумевает устройство, которое создано для воспроизведения резервных копий игр и иного изменения или удаления ограничений политики DRM (управление цифровыми правами). Таким образом, термин «модчип» охватывает определенные загрузочные устройства ПЗУ, которые были запрограммированы с помощью кода, который позволяет изменять политику DRM, а также такие устройства, как «патчеры», которые не содержат ПЗУ и работают путем динамического исправления нескольких ключевых мест прошивки Xbox по мере загрузки прошивки для выполнения.

появляется около обоих верхних и нижних базовых адресов. Теперь предположим, что Microsoft решила сэкономить на стоимости и уменьшить свой загрузочный ПЗУ размером 1 МБ до загрузочного ПЗУ размером 256 КБ. Теперь процессор видит 64 идентичных копии этого загрузочного ПЗУ размером 256 КБ, распределенных по адресному пространству ПЗУ размером 16 МБ, и весь старый код, использующий адресацию снизу и сверху, все еще работает. Примечательно, что ЦП в Xbox жестко запрограммирован на запуск выполнения кода при включении питания с адреса, расположенного в 16 байтах от верхней части памяти (его «вектор сброса»), в то время как процедуры инициализации оборудования, защищенные в чипсеты Xbox, используют ячейки ПЗУ, расположенные около нижней части пространства FLASH ROM размером 16 МБ. В результате, для оборудования Xbox требуется реализация LPC ROM размером либо 16 МБ, либо псевдоним меньшего содержимого ПЗУ по всему адресному пространству FLASH ROM. (SST 49LF020 — одна из немногих микросхем LPC FLASH ROM, которая размещает содержимое ROM по всему адресному пространству. Можно утверждать, что эта функция на самом деле является ошибкой: игнорируя верхние биты адреса и размещая содержимое ROM по всему адресному пространству, эта микросхема занимает пространство, которое могло бы быть выделено для

других функций. В результате SST выпустила обновленную версию «A-step» этой детали, названную 49LF020A, которая не размещает содержимое ROM по памяти. Аналогично, микросхема A-step не будет работать в качестве альтернативного устройства прошивки для Xbox.)

## Остальные 64 МБ SDRAM

Внимательный наблюдатель заметит, что на верхней стороне материнской платы Xbox отсутствуют два чипа, и что эти места с отсутствующими чипами выглядят



**Рисунок 10-2:** Незаполненные области памяти на материнской плате Xbox.

## Доверенные лица

Посмотрите на незанятое место памяти на материнской плате Xbox. Серебряная точка, окруженная темным кольцом внутри этих незанятых отпечатков микросхем, называется фидуциаром. Фидуциарные шаблоны используются машинами по сборке печатных плат в качестве контрольных точек для выравнивания больших микросхем с большим количеством выводов. Они разработаны для легкого распознавания системами машинного зрения, используемыми в машинах по сборке печатных плат. Фидуциары специальной формы также могут использоваться для автоматической идентификации ориентации и типа печатной платы.

подозрительно похожи на места, которые сейчас заняты чипами памяти. Переверните плату, и там есть еще два незанятых места для чипов. Эти пустые места на самом деле предназначены для чипов памяти. Расположение этих пустых мест показано на рисунке 10-2.

Следующий логичный вопрос, конечно, «Можно ли удвоить объем памяти Xbox до 128 МБ, припаяв подходящие чипы памяти в свободные слоты на материнской плате Xbox?» Ответ на самом деле да, но код инициализации для Xbox необходимо изменить, чтобы чипсет распознал и использовал дополнительную память. Кроме того, дополнительная память не улучшает графику или игровую производительность. Игры Xbox не предназначены для использования дополнительной памяти, поэтому дополнительная память обычно остается неиспользованной. Дополнительные ячейки памяти предоставляются в первую очередь для производства специальных консолей для разработчиков игр. Разработчики игр могут использовать дополнительную память для облегчения перехода игр на относительно ограниченный объем памяти Xbox, а также для хранения отладочных, мониторинговых и тестовых утилит в памяти, которые не являются частью образа игры. Обратите внимание, что дополнительная память может быть использована самодельным программным обеспечением, но сложность получения и установки чипов памяти делает расширение памяти Xbox скорее интересным упражнением по пайке, чем практической модификацией.

## Xbox VGA

Существует небольшая путаница относительно того, что делает адаптер Xbox VGA. Многие адаптеры Xbox VGA на самом деле являются преобразователями TV-to-VGA. Другими словами, они берут выходной сигнал TV с низким разрешением от Xbox и пропускают его через усилитель линий, чтобы получить низкокачественный VGA-дисплей. Настоящий адаптер Xbox VGA на самом деле настраивает Xbox на вывод видеосигнала с гораздо более высоким разрешением, обеспечивая на VGA-мониторе изображение лучшего качества, чем у телевизора.

Адаптер VGA настраивает графический режим Xbox с помощью назначенных контактов в разъеме AVIP (Audio Video I/O Port). Основная

проблема с этим подходом заключается в том, что игра должна быть специально написана для поддержки этого режима более высокого разрешения. В результате некоторые игры не будут работать с настоящим адаптером Xbox VGA, но, к счастью, вернуться к разрешению TV так же просто, как подключить стандартный кабель адаптера TV.

Оригинальный адаптер Xbox-VGA был разработан Кеном Гаспером. Он продаёт

версию на его сайте <http://xboxvga.xemulation.com>. В настоящее время он предлагает адаптер Xbox-VGA в виде «голой платы», а также в полностью собранном виде. Если вы ищете интересный проект по взлому оборудования для Xbox, который будет и полезным, и отточит ваши навыки сборки схем, возможно, стоит приобрести одну из его голых плат и попытаться собрать адаптер самостоятельно.

Приложение F содержит схему контактов Xbox AVIP.

## Замена запоминающего устройства большой емкости

Xbox содержит привод DVD-ROM и жесткий диск, оба из которых используют стандартный интерфейс IDE ПК для связи с материнской платой Xbox. Привод DVD-ROM также имеет фирменный разъем питания и состояния лотка DVD. Популярная и иногда необходимая хакерская деятельность для Xbox — замена этих приводов.

Пользователи заменяют или настраивают DVD-ROM, поскольку собственный привод Xbox DVD-ROM не может читать CD-R и многие типы носителей CD-RW. Это может быть особенно раздражающим для тех, кто пытается установить Xbox-Linux в первый раз, или для пользователей, которые пытаются скопировать музыку из своей коллекции CD-R на жесткий диск Xbox.

Существует множество методов замены и настройки привода Xbox DVD-ROM. Некоторые модели приводов Xbox DVD-ROM могут иметь регулировку интенсивности лазера для улучшения их способности читать носители CD-R и CD-RW. Это потенциально рискованная операция, так как вы можете навсегда повредить привод DVD-ROM, неправильно настроив выходную мощность лазера, но многие хакеры сообщают, что правильно выполненная процедура приводит к лучшей совместимости носителей. Я предлагаю поиск в Интернете последних новостей и методов, поскольку стиль и модель привода DVD-ROM, используемого в Xbox, часто различаются. Кроме того, привод Xbox DVD-ROM можно полностью заменить на стандартный DVD-ROM для ПК. Проблема с этим методом двоякая. Во-первых, обычный привод DVD-ROM для ПК не может читать оригинальные игровые диски Xbox из-за физических мер

безопасности, встроенных в игровой диск Xbox. Во-вторых, привод DVD-ROM для ПК необходимо адаптировать к пользовательскому разъему питания DVD и лотка на материнской плате Xbox.

Самый простой, но и самый некрасивый способ — установить стандартный привод DVD-ROM ПК, но оставить привод DVD-ROM Xbox подключенным через его фирменный кабель. В этом методе серый кабель IDE подключается к стандартному приводу DVD-ROM ПК (установленному в подчиненный режим с помощью конфигурации перемычек на приводе), а питание отбирается от разъема питания жесткого диска с помощью стандартного кабеля-разветвителя питания. Привод DVD-ROM Xbox остается на месте, но его разъем IDE пуст, а фирменный желтый кабель питания и состояния лотка установлен. Цель привода DVD-ROM Xbox — служить фиктивным приводом, который используется для ручной передачи состояния лотка DVD-привода на Xbox. Другими словами, пользователю необходимо вручную воспроизвести состояние лотка стандартного привода DVD-ROM ПК с помощью лотка DVD-ROM Xbox во время смены носителя.

Точная процедура работы Xbox в этой конфигурации зависит от конкретной модели привода DVD-ROM ПК и нюансов конфигурации оборудования Xbox, поэтому я снова предлагаю поискать в Интернете самую свежую информацию. Также есть несколько веб-сайтов, которые описывают, как адаптировать некоторые модели приводов DVD-ROM ПК для работы с фирменным состоянием лотка и разъемом питания Xbox. Такой проект является хорошим проектом среднего уровня для хакеров, которые в основном умеют обращаться с пайкой и отвертками. Изменения, выполненные на стандартном приводе DVD-ROM, позволяют точно передавать состояние лотка DVD стандартного привода на Xbox. Однако они не позволяют вам играть в оригинальные игры, если только Xbox не был модифицирован дополнительным оборудованием, которое обходит проверки безопасности привода DVD-ROM. Даже без возможности играть в игры, это все еще полезный метод для выявления проблем установки XboxLinux и для улучшения способности Xbox копировать вашу коллекцию CD или смотреть DVD. (Обратите внимание, что возвращение разъема IDE обратно в привод DVD-ROM Xbox восстановит исходную игровую функциональность Xbox.)

Жесткие диски Xbox также время от времени нуждаются в замене. Серьезные разработчики программного обеспечения для Xbox считают выгодным устанавливать жесткий диск большей емкости в Xbox, а пользователи со сломанными жесткими дисками также хотят заменить свои жесткие диски. К сожалению, жесткий диск OEM Xbox содержит защищенные авторским правом программы Microsoft. Жесткие диски Xbox также защищены блокировкой прошивки, что делает установку нового жесткого диска с оригинальной игровой функциональностью довольно сложной, особенно с точки зрения юридических вопросов. Блокировка прошивки также уникальна для каждого жесткого диска, что не позволяет вам заменить жесткий диск на использованный жесткий диск

Xbox. Однако, если вы хотите только запускать Xbox-Linux или другие программы homebrew и не хотите играть в игры, установка нового жесткого диска в Xbox так же проста и законна, как установка жесткого диска в любой ПК.



# Разработка программного обеспечения для Xbox

Хотя эта книга посвящена обучению читателей способам взлома оборудования и безопасности, одной из конечных целей взлома Xbox является запуск домашнего программного обеспечения. Эта глава посвящена описанию некоторых проектов домашнего программного обеспечения, которые велись для Xbox на момент написания этой статьи.

## Xbox-Linux

Целью проекта Xbox-Linux является создание удобного и легального порта GNU/Linux и приложений GNU/Linux на аппаратную платформу Xbox. Благодаря самоотверженности и вкладу хакеров по всему миру проект Xbox-Linux добился большого успеха в достижении своих целей. Изображение основной команды проекта Xbox-Linux можно увидеть на рисунке 11-1, а боковые панели в этой главе и в главе 9 содержат интервью с членами команды проекта Xbox-Linux. (Домашняя страница для Project Xbox-Linux — <http://xbox-linux.sourceforge.net>.)

Примечательно, что проект Xbox-Linux и его главные хакеры не являются противниками Microsoft. Они выступают за «свободу возиться», а не за ребяческих ненавистников Microsoft; их повестка дня касается сохранения тех самых свобод мысли и слова, которые привели технологию к тому, чем она является сегодня.

Xbox-Linux не является окончательным программным проектом для Xbox; напротив, это только начало хакерства программного обеспечения Xbox. Перенос знакомой среды разработки GNU/Linux на Xbox позволяет более широкой базе хакеров программного обеспечения присоединиться к проекту хакерства Xbox. С GNU/Linux Xbox может запускать широкий спектр прикладного программного обеспечения, от бесплатных видеоигр с открытым исходным кодом до приложений для обработки текстов и кластерного программного обеспечения для создания компьютерных кластеров в стиле Beowulf.

## Установка Xbox-Linux

В настоящее время для запуска Xbox-Linux необходимо установить загрузочное ПЗУ GNU/Linux с помощью альтернативного устройства прошивки. Для этого необходимо открыть Xbox. В Главе 10 описываются методы сборки и установки альтернативного устройства прошивки для Xbox через интерфейс LPC. Несколько поставщиков теперь предлагают простые в установке альтернативные устройства прошивки с интерфейсом LPC. В частности, устройство Xodus/Matrix является первым альтернативным устройством прошивки на рынке с полностью беспаечной процедурой установки. Все инструменты, необходимые для установки устройства Xodus/Matrix, описаны в Главе 1 «Аннулирование гарантии», а само устройство Xodus/Matrix поставляется с некоторыми простыми в установке

## Профиль: Майкл Стейл

**Майкл, можете ли вы рассказать нам немного больше о себе?**

Родился в 1979 году в Эрдинге/Германия, я студент факультета компьютерных наук в Мюнхенском техническом университете. Я преподаю ассемблер студентам в первом семестре и планирую получить степень магистра в следующем году. Я работаю с компьютерами с десяти лет; моим первым компьютером был Commodore 64, за которым вскоре последовал 386 PC. Моими основными интересами всегда были аппаратное обеспечение и операционные системы, и меня особенно увлекало разнообразие аппаратных архитектур (Commodore, PC, Amiga, Macintosh, . . .), а также популярные встроенные системы, такие как игровые консоли. (Знаете ли вы, что у «SEGA CD» три процессора, один Z80 и два M68000?). Вот почему я купил много игровых приставок для экспериментов, таких как Nintendo SNES, SEGA Genesis и Nintendo Game Boy. Я также посмотрел Linux для SEGA Dreamcast, но я никогда не видел Linux для Sony Playstation 2, так как весь комплект был для меня слишком дорогим, как для экспериментов, так и для реального использования.

**Как вы пришли в сферу взлома Xbox и, в частности, в проект Xbox-Linux?**

30 апреля 2002 года я купил Xbox, убежденный, что это будет отличная игрушка для взлома, и хорошо подходит для Linux. Посмотрев на системное программное обеспечение в течение часа или двух (я не купил игру), я разобрал Xbox. Поиск информации о взломе коробки сначала разочаровал меня: я не нашел ничего, кроме того, как подключить жесткий диск к ПК, и сайта о Xbox Linux, на котором практически не было информации. Поэтому я решил создать свой собственный сайт по взлому Xbox и разместить на нем информацию, которую я нашел, подключив жесткий диск к ПК.

(продолжение)

следуйте инструкциям по программированию и использованию альтернативного устройства с прошивкой.

### Примечание



Microsoft может и будет пересматривать свою материнскую плату и систему безопасности, поэтому перед покупкой уточните у поставщика вашего устройства совместимость с вашим конкретным системным оборудованием. Вам также понадобится кабель-конвертер Xbox gameport to USB, если вы хотите использовать стандартную клавиатуру и мышь с Xbox, которые можно приобрести у розничных продавцов, таких как Lik-Sang (<http://www.lik-sang.com>), или вы можете собрать их самостоятельно, следуя пошаговому руководству в Главе 4.

Перед установкой альтернативной прошивки устройства вам необходимо запрограммировать его с помощью образа ПЗУ, который загружает ядро

GNU/Linux. «Cromwell» — это загрузочное ПЗУ с открытым исходным кодом, «чистой комнаты» (т. е. не содержит кода Microsoft) для Xbox, которое способно загружать GNU/Linux. Важно отметить, что информация, содержащаяся в исходном коде и двоичном образе Cromwell, не может быть

Xboxhacker.net и оригинальный список рассылки Xbox Linux оказали большую помощь; они оба привлекли отличных хакеров и опубликовали ценную информацию. Недовольный изначальной инфраструктурой проекта Xbox Linux, я решил перейти на Sourceforge 23 мая. Теперь каждый участник мог добавлять что угодно на сайт, не обращаясь к сопровождающему. Но в то время все было еще довольно теоретически: без появления модчипов мы не могли сделать ничего, кроме как написать код, который «теоретически должен работать». Filtror Энди Грина ускорил все: этот мод позволил закончить загрузчик и, с помощью Милоша Мериака, адаптировать ядро Linux в очень короткие сроки.

«Анонимный донор», обратившийся ко мне в июне, не только привёл к дополнительной огласке проекта и, следовательно, к ещё большему количеству участников, но и к моей личной дружбе: Уолтер Мейер, создатель модчипа BioXX (OpenXbox), живёт всего в 20 километрах от меня. Помимо прочего, он очень помог мне с моддингом моих коробок, так как я не особо люблю паяльники.

Поскольку Linux уже работал на Xbox, в декабре 2002 года основная команда Xbox Linux (Энди Грин, Милош Мериак, Франц Ленер и я; Эдгар Хучек, к сожалению, не смог приехать) впервые встретилась лично на конгрессе Chaos Computer Club в Берлине.

Моя изначальная мотивация ко всему была просто в том, что это весело, и я мог многому научиться, делая это. Я не начал это, потому что хотел навредить Microsoft — тем не менее, я согласен, что Microsoft вредит своим клиентам, не позволяя им использовать программное обеспечение, которое они хотят использовать на купленном ими оборудовании, и именно поэтому проект Xbox Linux особенно важен.

(продолжение)

(Профиль: Майкл Стейл, продолжение)

[Мы] не «анти-рассеянный склероз» и не «ненавистники рассеянного склероза». Нам не нравится их рыночная стратегия, поэтому у нас есть рациональная причина работать против них.

**Хотите ли вы что-нибудь еще сказать о призе в 200 тысяч долларов для Xbox-Linux?**

Я думаю, что награда не привлекла людей, которые хотели увидеть деньги: Сейчас, спустя месяц после крайнего срока, деньги все еще не распределены, и ни один человек не прислал мне ни одного вопроса о том, когда он получит деньги. Награда привлекла прессу; мы получили больше рекламы, и таким образом у нас появилось больше хакеров. Но никто не делал этого из-за денег. Поэтому мы не хотим, чтобы нас считали получающими плату за работу от Майкла Робертсона. Хорошим доказательством является то, что мы все еще активны после крайнего срока.

**Можете ли вы рассказать нам больше о вашем «взломе MIST X-Code»?**

Через некоторое время после первоначального взлома Банни Энди полностью извлек MCPX ROM, и Стив, Пол и я начали анализировать код, и я провел обратную разработку интерпретатора X-Code, содержащегося в нем. При поиске ошибок, которые можно было бы использовать для выхода из цикла интерпретации X-Code, я обнаружил, что часть кода уже была написана с учетом наших атак. Вот мой первоначальный дизассемблированный код:

```
cmp ebx, 80000880 ; Мост ISA, отключение MCPX? jnz  
short not_mcpk_disable ; ОШИБКА: слишком специфичны:  
биты с 24 по 30 ; не определены и игнорируются  
оборудованием PCI! и есх, а не 2 ; очистить бит 1  
(ПЗУ MCPX будет ; отключено установкой бита 1)  
not_mcpk_disable: mov eax, ebx mov dx, 0CF8h out dx,  
eax ; Адрес конфигурации PCI add dl, 4 mov eax, есх  
out dx, eax ; Данные конфигурации PCI jmp short  
next_instruction
```

Я уже работал с «конфигурацией PCI», поэтому знал, что тест на атаку слишком специфичен: Похожие коды будут делать то же самое, но они проходят тест. Так что у разработчиков MS была хорошая идея, но реализация была неправильной, таким образом, говоря нам об их идеи таким образом!

Я отправил свою идею Энди, Стиву и Полу, и через некоторое время они подтвердили, что 0x88000880 работает так же хорошо, как 0x80000880 для отключения MCPX ROM и выхода из интерпретатора путем отображения кода интерпретатора из памяти!

используется для обхода любого из встроенных в Xbox механизмов контроля авторских прав. Другими словами, трудно утверждать, что Cromwell является каким-либо инструментом обхода контроля авторских прав. (Cromwell можно загрузить с веб-сайта Xbox-Linux на сервере Sourceforge.net по адресу <http://xbox-linux.sourceforge.net>.)

После записи Cromwell ROM на альтернативное устройство прошивки и установки устройства в Xbox вам нужно будет записать на CD/RW носитель установочный образ GNU/Linux, который вы можете загрузить с веб-сайта XboxLinux (опять же, <http://xbox-linux.sourceforge.net>). Этот установочный образ поставляется как довольно большой (100+ МБ) образ ISO, сжатый с помощью bzip2, и он содержит все программное обеспечение, интерфейсы и инструменты, необходимые для получения удобного для пользователя дистрибутива GNU/Linux и его запуска на Xbox. При записи этого образа ISO необходимо использовать опцию записи образа в программном обеспечении для записи компакт-дисков. Не копируйте образ ISO на CD как один большой файл. (Образы ISO представляют собой буквальные битовые шаблоны для компакт-диска, поэтому образ ISO уже содержит полное описание файловой системы. Запись образа ISO как обычного файла, а не как образа, инкапсулирует образ ISO в новую файловую систему, поэтому ISO выглядит просто как «мешок битов», а не как файловая система с файлами.)

Вам также может понадобиться второй диск, записанный с загрузочной программой для XboxLinux. Эта загрузочная программа поставляется в виде меньшего образа ISO, который должен быть доступен там же, где вы загрузили основной установочный образ GNU/Linux. Этот загрузочный образ позволяет вам загрузить установку Linux, просто перетащив ее в Xbox, как при запуске игры. (Вы также можете скопировать содержимое этого диска на жесткий диск с помощью сторонней панели управления и загрузить Xbox-Linux непосредственно с жесткого диска, если вы предпочитаете не иметь дело с отдельным загрузочным диском.)

Запись хорошего образа CD/RW, возможно, является одной из самых сложных частей установки Xbox-Linux. Лазер, используемый внутри привода DVD-ROM Xbox, не очень хорошо подходит для чтения записываемых носителей CD, поэтому Xbox очень привередлив в отношении типа носителя и типа записывающего устройства, а также настроек записывающего устройства, используемых для создания образа CD. Более того, точные детали того, как ухудшается лазер, различаются от Xbox к Xbox и зависят от модели установленного привода. Пользователи обнаружили, что лишь немногие Xbox могут надежно читать носители CD-R, поэтому необходимо использовать носители CD/RW. Кроме того, полезно записывать носители CD/RW на самых медленных настройках записи, используя либо чистый, чистый CD/RW, либо полностью стертый CD/RW (в отличие от быстрого стирания, которое просто сбрасывает файловую систему и фактически не уничтожает ранее записанные данные).

Прежде чем остановиться на определенном типе носителя CD/RW, попробуйте использовать обычные инструменты копирования WMA Xbox Dashboard, чтобы скопировать содержимое CD/RW, записанного вами с музыкой, на жесткий диск. Если это работает надежно и без ошибок, вы, вероятно, можете использовать этот тип носителя CD/RW для установки Linux. (Многие проблемы установки Xbox-Linux были связаны с проблемами чтения данных с привода CD/RW.) На момент написания этой статьи не было дистрибутивов в формате CD-ROM-носители. В сообществе Xbox-Linux ведутся разговоры о заказе набора пользовательских образов CD-ROM, поскольку это решило бы многие проблемы с CD/RW, с которыми сталкиваются пользователи. (Также обратите внимание, что в Xbox можно установить сторонний привод DVD-ROM, который лучше совместим с записываемыми форматами CD, как обсуждалось в предыдущей главе.)

### Примечание



Помните, что Xbox-Linux — это активный проект, который постоянно развивается. Самые последние инструкции по установке GNU/Linux на Xbox можно найти на сайте Sourceforge Xbox-Linux, и на момент написания этой статьи эти инструкции были переведены как минимум на поддюйнины языков. Если вы заинтересованы в том, чтобы внести свой вклад в проект Xbox-Linux, на сайте Sourceforge Xbox-Linux есть список проектов, которые нужно сделать, а также некоторые инструкции о том, как присоединиться к списку рассылки разработчиков.

## «Проект Б»

Ведется работа, которую разработчики Xbox-Linux называют «Проектом В», чтобы найти способ установки и загрузки Xbox-Linux без каких-либо аппаратных модификаций. Название «Проект В» происходит от критериев, определенных для присуждения приза в размере 200 000 долларов, предложенного Майклом Робертсоном, генеральным директором Lindows. Приз «Проекта А» составил 100 000 долларов, и он был присужден первой группе, которая запустит Linux на Xbox с аппаратными модификациями. Оставшиеся 100 000 долларов будут присуждены человеку или группе, которые завершат Проект В. Асимметричное распределение призовых денег намекает на сложность завершения Проекта В. (Более подробную информацию о Проекте В можно найти на веб-сайте Sourceforge Xbox-Linux по адресу <http://xbox-linux.sourceforge.net/articles.php?aid=2002354043211>.) Существует ряд стратегий Проекта В, реализуемых различными группами. Наиболее концептуально простой подход заключается в факторизации 2048-битного ключа RSA, используемого для подписи игровых дисков Xbox. Этот подход реализуется OperationProjectX (<http://sourceforge.net/projects/opx>) с использованием подхода распределенных вычислений. Проще говоря, если 2048-битный ключ RSA факторизуется для раскрытия закрытого

ключа Microsoft, любой может подделать цифровую подпись Microsoft и создать загрузочные игровые диски для Xbox, учитывая, что Microsoft никогда не удаляет из ядра Xbox возможность загрузки программ с обычных носителей CD или CD/RW. Примечательно, что Microsoft поставляет свои игры на двухслойных дисках формата DVD-9 со специальными структурами безопасности. Прошивка Xbox может быть настроена Microsoft для загрузки только с дисков, имеющих эту конкретную структуру, независимо от проверки цифровой подписи. Поскольку в настоящее время невозможно записывать двухслойные DVD-диски с помощью обычного привода для записи DVD, требование защищенного носителя DVD-9 в качестве единственного источника исполняемых файлов будет представлять собой препятствие для распространения Xbox-Linux посредством бесплатных загрузок из Интернета. Другая проблема с этим подходом заключается в том, что вероятность успешного факторинга закрытого ключа Xbox с помощью поиска методом перебора очень и очень мала. (Глава 7, «Краткий вводный курс по безопасности», содержит врезку «Очень сложные проблемы», в которой делается попытка сообщить о вычислительной сложности этой задачи.) Если закрытый ключ будет успешно восстановлен в течение разумного периода времени с помощью этого подхода, это значительно снизит доверие людей к алгоритму RSA. (С другой стороны, вы никогда не выиграете в лотерею, если не купите билет, а запуск бесплатного клиента распределенного факторинга с использованием свободных циклов вашего ЦП намного дешевле, чем билет Powerball.)

Другой подход, связанный со взломом ключа RSA-2048 бит, заключается в изменении существующего подписанного исполняемого файла Xbox полезным способом без изменения его криптографического хэш-значения. Такое конструктивное столкновение хэшей сделает измененный исполняемый файл идентичным оригиналу с точки зрения проверки цифровой подписи. Хэш, используемый в алгоритме цифровой подписи Xbox, — это SHA-1. SHA-1 — это 160-битный хэш без каких-либо публично известных алгоритмических слабостей; поскольку источник хеша фиксирован, для обнаружения коллизии нужно будет перебрать около 2160 случайных вариаций. В качестве побочного замечания, вы не можете использовать атаку дня рождения, чтобы снизить сложность атаки до 280 случайных вариаций, потому что мы не пытаемся найти два сообщения, которые хэшируются до одного и того же произвольного значения. Цель состоит в том, чтобы генерировать определенный целевой хэш или, возможно, один из очень ограниченного набора целевых хэшей, собранных из набора всех опубликованных игр Xbox. Следовательно, этот подход также попадает в категорию «Очень сложный». Проблемы».



**Рисунок 11-1:** Основная команда Xbox-Linux в 19-м Ежегодной конференции Chaos Computer Conference, проходящая в Берлине, Германия. На заднем плане Михаэль Штайль; на переднем плане слева направо: Энди Грин, Милош Мериак и Франц Ленер. (Фотография предоставлена Герхардом Фарфеледером.)

Альтернативный подход к проекту В заключается в поиске уязвимостей в программном обеспечении Xbox и использовании их для захвата контроля над указателем инструкций ЦП. Чтобы увидеть, насколько это полезно, рассмотрим следующий пример: предположим, что в игре был обнаружен сетевой экспloit переполнения буфера, который может привести к выполнению произвольного кода. Программа, работающая на ПК, подключенная к Xbox через сеть, может затем использовать этот экспloit для отправки пакетов на Xbox, которые установят простой загрузчик для Xbox-Linux. Этот загрузчик может быть чем-то таким же простым, как программа, которая запускает код в указанном месте на жестком диске Xbox или на DVD-приводе. Любой порт, через который Xbox может принимать данные, является вектором для такого рода атак, включая USB и сетевой порт, а также жесткий диск и привод DVD-ROM.) Поврежденные сохраненные игры или файловые структуры могут быть скопированы на жесткий диск или привод DVD-ROM, что заставит Xbox запустить код, разработанный пользователем. К чести Microsoft, все сетевые взаимодействия и протоколы сохраненных игр используют довольно надежные и хорошо проверенные методы безопасности. Кроме того, на презентации Xbox от Microsoft в МИТ я слышал, что весь игровой код проверяется средством проверки переполнения буфера и что у Microsoft есть договорные средства правовой защиты против разработчиков игр, которые признаны виновными в намеренном размещении бэкдоров в коде

своих игр. Это указывает на то, что кодовая база Xbox более безопасна, чем типичный продукт Microsoft, что делает ее еще более интересной проблемой для хакеров. (Если вы заинтересованы в участии во взломе Xbox в рамках «Проекта В», я рекомендую вам сначала проверить

## Профиль: Милош Мериак

### Можете ли вы рассказать нам немного о себе?

Моя общая история довольно проста. Я родился в 1976 году в Чехословакии. Мои родители (моя мать — учительница, мой отец — инженер-строитель) бежали во время холодной войны в Западную Германию из-за репрессий коммунистического режима. Мне было около трех лет, когда мы приехали в Германию. В немецком детском саду я сразу же выучил немецкий язык. С этого момента все было действительно просто — в десять лет я получил свой первый компьютер после нескольких месяцев нытья. Дело пошло.

После выпускных экзаменов в школе и странного интермеццо в немецкой федеральной армии я начал изучать кибернетику и информатику, но через три года я решил уйти из университета и сосредоточиться в качестве долгосрочной цели на собственной компании. Во время учебы я установил несколько ценных деловых связей, поэтому было легко работать фрилансером в различных компаниях в Германии. Я сделал несколько проектов по обратному проектированию, разработал встраиваемые системы Linux в реальном времени с небольшим объемом памяти, сделал несколько низкоуровневых программ, таких как расширения в реальном времени для систем Windows, и разработал программную защиту жесткого диска для известной немецкой компании. Сейчас я живу со своей девушкой в Берлине, и мы отлично проводим там время.

(продолжение)

на веб-странице правил конкурса Project B Prize по адресу  
(<http://xbox-linux.sourceforge.net/articles.php?aid=20030023081956.>)

Недавно был обнаружен экспloit переполнения буфера в способе обработки сохраненных игр в игре Electronic Arts «007: Agent Under Fire». Экспloit был впервые раскрыт хакером, известным просто как «habibi\_xbox», 29 марта 2003 года в сообщении на BBS XboxHacker.net. Примечательно, что экспloit был обнаружен в нераскрытом количестве игр, но «007: Agent Under Fire» была единственной игрой, явно названной в сообщении. Экспloit использует непроверенную строку для запуска короткого сегмента (несколько сотен байт) кода, который вставляет ряд исправлений ядра. В конструкцию взлома были включены различные меры, чтобы сделать очень сложным изменение взлома для выполнения чего-либо, кроме запуска предполагаемой цели Xbox-Linux. Например, взлом исправляет исходный открытый ключ Xbox RSA, используемый для

проверки цифровых подписей, новым открытым ключом, оставляя алгоритм проверки цифровой подписи неисправлением. Только загрузчик Xbox-Linux, предоставленный как часть взлома, надлежащим образом подписан соответствующим новым закрытым ключом. Другим хакерам пришлось бы учитывать новый открытый ключ, чтобы использовать этот взлом для запуска других исполняемых файлов. Кроме того, сама игра «007: Agent Under Fire» выполняет независимую проверку цифровой подписи во всех сохраненных играх, поэтому изменение кода эксплойта во взломанном файле сохранения не является тривиальным. Включение таких мер безопасности во взлом является похвальным решением на

#### **Зачем вы взламываете?**

Приобретя больше опыта в программировании, я начал понимать, что прекрасная и яркая сущность компьютерного мира на самом деле представляет собой хрупкое лоскутное одеяло.

В начале хакерство было для меня как игра. Вы могли бродить по своей компьютерной системе, каждый день открывая миры нового кода и возможностей. Иногда можно было вызвать авторов приложений на дуэль, пытаясь проанализировать и обойти их защиту от копирования. Иногда это было похоже на игру в шахматы; иногда это было похоже на смертельный бой.

С одной стороны, я был рад видеть, как растут мои знания, а с другой стороны, это, естественно, было большим стимулом для самолюбия 14-летнего ребенка, чтобы обойти системы безопасности переплачиваемых богоподобных хардкорных программистов. Во время моего обучения в старшей школе я пересмотрел эту точку зрения — во время программирования инструментов и приложений для некоторых местных компаний во время школьных каникул я встретил несколько настоящих программистов — и был разочарован: они не были ни богами, ни богоподобными.

Через некоторое время я понял, что написание крутой демки, взлом приложения X или поиск крутого хака для Y не меняют мир больше, чем мешок риса, упавший где-то в Китае. Поэтому я начал выбирать сферы своей деятельности более разумно — технологии повседневной жизни, такие как телефоны, компьютеры,

(продолжение)

(Профиль: Милош Мериак, продолжение)

сети и спутники. Я узнал, что можно изменить положение вещей, объясняя технологии обычным пользователям или помогая компаниям защищать свои продукты.

Сегодня я осознаю свою силу как белого хакера. Каждый человек в сегодняшней жизни подвержен влиянию информационных технологий: методы наблюдения, добыча данных, информационная война, Закон об авторском праве в цифровую эпоху, ТСРП, управление цифровыми правами, новые интерпретации авторского права и патентного права растут как грибы после муссонного дождя. Как и в прошлом, я жажду заглянуть за эти прекрасные и яркие сущности и, надеюсь, найти ошибки и ловушки прежде, чем они найдут нас.

**Можете ли вы рассказать нам о своем опыте работы с проектом Xbox-Linux?**

Я присоединился к проекту Xbox Linux и помог запустить ядро, что было непросто, поскольку архитектура Xbox имеет некоторые ловушки и отличия по сравнению с персональным компьютером. Я создал ранние дистрибутивы Linux для Xbox от Microsoft. Это было важно, поскольку у нас был только 1 МБ флэш-памяти для хранения полного дистрибутива и ядра, а жесткий диск еще не был разблокирован. Я также предоставил драйвер консоли для устройства filtror Энди Грина, поэтому мы могли видеть сообщения загрузки ядра и получать консоль Linux, используя его устройство в качестве своего рода удаленного интерфейса. Этот дистрибутив уже включал сетевые драйверы, драйверы звуковой карты, поддержку mp3, сервер telnet, веб-сервер, поддержку NFS и широкий спектр стандартных инструментов Linux. Это позволило нам избавиться от нашего собственного оборудования и позволило сотням людей присоединиться к проекту, либо в качестве авторов кода, либо в качестве тестировщиков. У нас еще не было вывода на экран, поэтому я добавил интерфейс кадрового буфера в ядро Xbox Linux и внес много других вкладов.

Число разработчиков, вносящих вклад, начало расти колоссально. Мы получаем огромную помощь со всего мира, чтобы сделать Xbox Linux возможным. Некоторые остаются скрытыми, потому что боятся юридических неопределенностей, таких как DMCA в Соединенных Штатах, в то время как другие могут вносить свой вклад свободно.

**Хотите ли вы поделиться какими-либо другими комментариями?**

Некоторые могут спросить, почему взрослые люди вроде меня возятся с этой игрушкой Xbox. У каждого человека, конечно, есть свои причины; моя причина в том, чтобы улучшить свои навыки и узнать больше о последних технологиях. Например, Microsoft Xbox является предшественником защищенного компьютера TCPA/Palladium со всеми техническими и социальными последствиями. Это прекрасная игровая площадка для моих исследований более безопасных компьютерных систем без давления на пользователей.

Одна из главных причин — наше сообщество. Очень весело и приятно работать вместе с этими яркими гиками — онлайн и особенно офлайн в пабе с кружками хорошего пива. Я каждый день поражаюсь растущей силе нашего сообщества. Спасибо всем за то, что сделали это возможным!

часть реализатора взлома, поскольку это помогает гарантировать, что взлом не будет напрямую полезен для таких приложений, как пиратство. Реализация мер безопасности, которые защищают интересы Microsoft, может помочь спасти проект Xbox-Linux от гнева Microsoft и Министерства юстиции США.

Заглядывая вперед, успех Project B может означать либо новую эру взлома Xbox, либо упадок взлома Xbox. Несмотря на то, что хакеры Project B продемонстрировали общественную совесть и добрую волю, пытаясь защитить интересы Microsoft, невозможно помешать менее щепетильным хакерам провести обратную разработку взлома и в конечном итоге выяснить, как воспроизвести технику в какой-то менее дружественной Microsoft форме. Конечным результатом может стать либо жесткое подавление Microsoft всей хакерской деятельности, либо полный выход Microsoft из видеонгрового бизнеса, поскольку их поток доходов будет отрезан, как у Sega в пиратском крахе Dreamcast. Или Microsoft может просто решить вложить больше денег в бизнес и выпустить переработанную консоль, которая включает исправления и контрмеры для известных ляпов в безопасности. Результат будет во многом зависеть от того, как будут развиваться события в ближайшие несколько месяцев. Однако, учитывая, что на горизонте маячат серьезные снижения цен на Xbox и циркулирующие слухи о полностью переработанной «урезанной» версии консоли, похоже, что краткосрочная стратегия Microsoft заключается в том, чтобы сосредоточить свои усилия на штурме рынка, а не на пресечении добросовестного использования или пиратства. В конце концов, каждая проданная Playstation2 или Gamecube, вероятно, оказывает худшее влияние на бизнес Microsoft, чем каждая Xbox, преобразованная для работы с GNU/Linux, или даже Xbox, преобразованная для работы с пиратскими играми.

## OpenXDK

Множество интересных и полезных проектов для Xbox, таких как XboxMediaPlayer и MAME-X (Multiple Arcade Machine Emulator for the Xbox) были разработаны для собственной игровой платформы Xbox. К сожалению, эти программы были разработаны с использованием неавторизованных версий Microsoft Xbox SDK (Software Development Kit). Предполагается, что Xbox SDK от Microsoft доступен только одобренным лицензированным разработчикам. Однако SDK просочился еще до запуска консоли, и с тех пор многие использовали просочившийся Xbox SDK для создания собственных программ для Xbox. Хотя фирменный Xbox SDK удобен и прост в использовании, его использование технически незаконно. Отсутствие легального SDK для собственной

платформы Xbox затрудняет привлечение большой базы разработчиков с открытым исходным кодом.

Проект OpenXDK был создан для решения проблемы необходимости легальной альтернативы Xbox SDK. Заявленная цель OpenXDK — создание легального комплекта разработки для создания исполняемых файлов Xbox (XBE). OpenXDK позволит пользователям создавать собственные файлы XBE, которые при подписании соответствующей цифровой подписью могут запускаться на обычной Xbox. Поскольку эта соответствующая цифровая подпись пока неизвестна, эта работа ведется в ожидании легальной технологии, которая обеспечивает взаимодействие с программами, разработанными с использованием OpenXDK.

## 172

Несмотря на свою полезность, проект OpenXDK все еще находится в зачаточном состоянии и ищет разработчиков. Подробнее о проекте OpenXDK можно узнать здесь на <http://openxdk.sourceforge.net>. Руководителями проекта OpenXDK являются Дэн Джонсон (также известный как SiliconIce, создатель XboxHacker BBS) и Аарон Робинсон (также известный как caustik; caustik также руководит проектами по перелинковке исполняемых файлов CXBX и эмулятора Xbox CXBE).

# Будьте хакеры! бдительны,

Обратное проектирование и право интеллектуальной собственности имеют некоторые сложные правовые взаимодействия. С одной стороны, инновации заслуживают своего справедливого вознаграждения. Право изобретателей или авторов на эксклюзивное производство или продажу плодов своего труда должно быть защищено. С другой стороны, для сохранения инноваций и обеспечения справедливых рынков также требуется свободный и конкурентный рынок. Изучение принципов проектирования, воплощенных в существующих продуктах, и способность производить улучшенные производные продукты являются важной частью конкурентного рынка.

В этой главе представлен обзор права интеллектуальной собственности и некоторые важные моменты, которые вам нужно знать как хакеру. Незнание не является оправданием, и существуют некоторые суровые наказания, предусмотренные законом для тех, кто игнорирует законы, регулирующие обратную разработку и права интеллектуальной собственности. Некоторые акты нарушения прав интеллектуальной собственности караются как тяжкие преступления и крупные штрафы.

Большая часть этой главы была написана Ли Тьеном, старшим юристом Electronic Frontier Foundation. Ли (и Джозеф Лю) были моими консультантами в тот период, когда я пытался опубликовать свои выводы о системе безопасности Xbox. Глава 8 имеет врезку под названием «Юридические проблемы взлома», в которой описывается моя борьба с МПГ за публикацию моей статьи.

Содержание этой главы представлено с целью предоставления информационного ресурса для хакеров. Если вы считаете, что можете оказаться в юридически компрометирующем ситуации, нет ничего лучше, чем обратиться к адвокату и получить надлежащую юридическую консультацию по вашей конкретной ситуации.

## Профиль: Ли Тъен

Ли Тъен — старший юрист в Electronic Frontier Foundation, специализирующийся на праве свободы слова, включая его взаимосвязь с правом интеллектуальной собственности и правом на неприкосновенность частной жизни. До прихода в EFF Ли был единственным практикующим специалистом, специализирующимся на судебных разбирательствах по Закону о свободе информации (FOIA). Г-н Тъен опубликовал статьи о детской сексуальности и информационных технологиях, анонимности, наблюдении и статусе публикации компьютерного программного обеспечения в соответствии с Первой поправкой. Ли получил степень бакалавра по психологии в Стэнфордском университете, где он активно занимался журналистикой в Stanford Daily. После работы репортером новостей в Tacoma News Tribune в течение года, Ли учился на юридическом факультете в Boalt Hall, Калифорнийский университет в Беркли. Ли также работал над выпускной работой по программе юриспруденции и социальной политики в Калифорнийском университете в Беркли.<sup>1</sup>

## Фонд Электронного Фронтира

Electronic Frontier Foundation (EFF) предоставил мне юридическую консультацию в период, когда я пытался опубликовать свою работу о системе безопасности Xbox. В следующих параграфах рассказывается о том, чем занимается EFF и кто они такие.

Представьте себе мир, в котором технологии позволяют нам всем делиться знаниями, идеями, мыслями, юмором, музыкой, словами и искусством с друзьями, незнакомцами и будущими поколениями.

Этот мир существует здесь и сейчас, и он стал возможным благодаря электронной сети — Интернету — с силой, которая может объединить нас всех. И будущие разработки в области технологий позволят нам получать доступ к информации и общаться с другими еще более эффективными способами.

Но правительства и корпоративные интересы по всему миру пытаются помешать нам свободно общаться посредством новых технологий, так же, как когда те, кто находился у власти, контролировали производство и распространение — или даже сжигали — книги, которые они не хотели, чтобы люди читали в Средние века. Но только борясь за наши права свободно говорить, независимо от носителя — будь то книги, телефоны или компьютеры — мы можем защитить и улучшить человеческое состояние.

Фонд Electronic Frontier Foundation (EFF) был создан для защиты наших прав думать, говорить и делиться своими идеями, мыслями и потребностями с помощью новых технологий, таких как Интернет и Всемирная паутина. EFF является первым, кто выявил угрозы нашим основным правам в Интернете и выступил в защиту свободы слова в цифровую эпоху.

EFF — это организация, базирующаяся в Сан-Франциско и финансируемая донорами. Ее деятельность направлена на защиту наших основных прав независимо от технологий; на информирование прессы, политиков и широкой общественности о вопросах гражданских свобод, связанных с

(продолжение)



технологии; и выступать в качестве защитника этих свобод. Среди наших различных видов деятельности EFF выступает против ошибочного законодательства, инициирует и защищает судебные дела, защищая права личности, запускает глобальные общественные кампании, представляет передовые предложения и документы, проводит частые образовательные мероприятия, регулярно взаимодействует с прессой и публикует полный архив цифровой информации о гражданских свободах на одном из самых популярных веб-сайтов в мире: <http://www.eff.org>.<sup>2</sup>

<sup>1</sup> С сайта EFF, [http://www.eff.org/homes/lee\\_tien.html](http://www.eff.org/homes/lee_tien.html)

<sup>2</sup> С сайта EFF, <http://www.eff.org/abouteff.html>

## **Caveat Hacker: Учебник по интеллектуальной собственности, Ли Тъен**

Обратное проектирование — это процесс извлечения ноу-хау или знаний из артефакта; на рынке это называется «проверенной временем техникой выяснения того, что заставляет работать продукт конкурента». 1 Но любой, кто изучает продукты массового рынка сегодня, должен знать о юридическом минном поле, окружающем обратное проектирование. Положения об обходе Закона об авторском праве в цифровую эпоху (DMCA), 2 договорные условия, запрещающие обратное проектирование, и Закон об экономическом шпионаже 3 — вот лишь некоторые из опасных правовых областей, о которых следует знать технологам. В этой главе мы кратко рассмотрим эти области, чтобы дать хакерам общее представление о проблемах.

Здесь есть два общих вопроса. Во-первых, является ли обратная разработка законной? Во-вторых, даже если вы можете провести обратную разработку продукта, можете ли вы опубликовать то, что вы узнали из обратной разработки?

### **Классическое право интеллектуальной собственности: обзор**

Закон об интеллектуальной собственности традиционно означал авторские права и патенты. Оба они созданы и ограничены федеральными законами, основанными на пункте Конституции об

интеллектуальной собственности: «Конгресс имеет право...  
содействовать прогрессу науки и полезных искусств, обеспечивая  
ограниченное  
Времена Авторам и Изобретателям исключительное Право на их  
соответствующие

---

<sup>1</sup> Джоэл Миллер, Обратное проектирование: честная игра или мошенничество?, IEEE Spectrum, апрель 1993 г., стр. 64, 64.

<sup>2</sup> 17 Свод законов США § 1201–1204.

<sup>3</sup> 18 Свод законов США § 1831–39.

Писания и открытия».<sup>23</sup> Компьютерные программы обычно охраняются авторским правом как «литературные произведения», но их также можно запатентовать.<sup>24</sup>

Недавно люди стали думать о коммерческой тайне как о еще одном виде интеллектуальной собственности. Изначально коммерческая тайна защищалась судами в рамках прецедентного права, но теперь она также является предметом как государственных, так и федеральных законов. В отличие от авторских прав и патентов, закон о коммерческой тайне исторически основан на принципах недобросовестной конкуренции.

В Соединенных Штатах авторы и изобретатели не имеют «естественных прав».<sup>25</sup> Вместо этого их права основаны на понятии общественного благосостояния. Общество выигрывает, если авторы и изобретатели получат некоторую защиту, потому что у них не будет достаточных стимулов для творчества, если другие смогут свободно использовать их работу. Но эта защита ограничена, чтобы гарантировать, что в конечном итоге выиграет общественность.<sup>26</sup> Например, авторские и патентные права действуют только «ограниченное время»; в конечном итоге запищенные произведения должны стать общественным достоянием.<sup>27</sup><sup>28</sup> Короче говоря,

---

<sup>23</sup> Конституция США, ст. I, §8, п. 8. Когда писалась Конституция, слово «наука» часто использовалось как синоним слова «знания».

<sup>24</sup> См. Diamond против Diehr, 450 US 175 (1981); In re Alappat, 33 F.3d 1526 (Fed. Cir. 1994).

<sup>25</sup> В Европе авторское право традиционно рассматривалось как защита неотъемлемого личного права создателя произведения.

<sup>26</sup> «Экономическая философия, лежащая в основе положения, уполномочивающего Конгресс выдавать патенты и авторские права, заключается в убеждении, что поощрение индивидуальных усилий путем личной выгоды является наилучшим способом повышения общественного благосостояния посредством талантов авторов и изобретателей в «Науке и полезных искусствах». Мейзер против Стайна, 347 US 201, 219 (1954).

<sup>27</sup> Feist Publications, Inc. против Rural Tel. Serv. Co., 499 US 340, 348-

<sup>28</sup> (1991) («Этот результат не является ни несправедливым, ни неудачным. Это средство, с помощью которого авторское право способствует прогрессу науки и искусства».)

закон об интеллектуальной собственности устанавливает условия «сделки» между общественностью и авторами или изобретателями.

## Авторские права

Закон об авторском праве защищает оригинальные произведения, которые «зафиксированы» на материальном носителе, и предоставляет автору (или правопреемнику) исключительные права на воспроизведение, распространение, адаптацию, публичный показ и публичное исполнение произведения. Он не защищает от независимого создания.

Произведения не то же самое, что копии или фонозаписи (копии звукозаписей). Когда вы покупаете книгу, вы владеете копией, но владелец авторских прав сохраняет права на само произведение. Обратите внимание, кстати, что доктрина «первой продажи» позволяет законным владельцам копий продавать или передавать эти законно принадлежащие копии,<sup>9</sup> за некоторыми исключениями.<sup>10</sup>

Существует множество различных типов произведений, и для каждого типа существует множество различных правил. Поэтому закон об авторском праве довольно сложен, а технологии не упростили ситуацию. Рассмотрим защищенную авторским правом песню. Песня или музыкальная композиция (МС) защищены авторским правом, которое обычно принадлежит автору песни. Чтобы записать песню, необходимо разрешение от владельца авторских прав на МС.<sup>11</sup> После записи на звукозапись (SR) действует независимое авторское право, которое защищает фактически записанные звуки, включая интерпретацию певцом базовой песни, а также усилия продюсера и звукорежиссеров. Звукозаписывающие компании обычно владеют авторскими правами на SR. В результате, если вы хотите использовать защищенную авторским правом звукозапись песни в телевизионной рекламе, вам необходимо разрешение как владельца авторских прав на МС, так и владельца авторских прав на SR.

Большинство прав владельца авторских прав достаточно очевидны, но некоторые из них не очевидны — особенно когда речь идет о компьютерах. Например, компьютеры загружают программы в оперативную память, создавая копию в целях защиты авторских прав. Закон об авторских правах содержит конкретное исключение, которое позволяет владельцу копии компьютерной программы копировать программу в память компьютера.<sup>12</sup> Это иллюстрирует общую строгость закона об авторских правах: то, что нельзя использовать защищенную авторским правом работу по назначению, не сделав копию, не означает, что создание копии не является нарушением авторских прав. Последствия этой строгости для Интернета серьезны, поскольку распространение в Интернете обычно подразумевает создание копий.

Право на адаптацию также может сбивать с толку. Адаптации или «производные работы» — это работы, основанные на работе, защищенной авторским правом: переводы на иностранные языки, фильмы, основанные на книгах и т. д. В одном из часто критикуемых дел суд постановил, что вырезание изображений из законно принадлежащих копий и размещение изображений на керамической плитке создали производные работы, нарушающие авторские права.<sup>13</sup> Большинство судов не согласны с этим результатом.<sup>14</sup>

---

<sup>9</sup> См. 17 USC Sec. 109. «Весь смысл доктрины первой продажи заключается в том, что как только владелец авторских прав помещает защищенный авторским правом объект в коммерческий оборот путем его продажи, он исчерпывает свое исключительное законное право контролировать его распространение». *Quality King против L'Anza Research Int'l*, 523 US 135, \_\_\_ (1998).

<sup>10</sup> Например, фонограммы и отдельные компьютерные программы рассматриваются иначе, чем книги в соответствии с разделом 109.

<sup>11</sup> В соответствии с действующим законодательством об авторском праве, право на воспроизведение MC контролируется положениями об обязательной лицензии, что означает, что вы автоматически получаете разрешение, заплатив установленную законом пошлину.

<sup>12</sup> 17 USC § 117 (разрешение создания копии или адаптации копии или адаптации «как существенного шага в использовании компьютерной программы совместно с машиной»).

<sup>13</sup> *Mirage Editions, Inc.* против *Albuquerque ART Co.*, 856 F.2d 1341, 1344 (9th Cir. 1988), сертификат отклонен, 489 US 1018 (1989).

<sup>14</sup> См., например, *Lee v. Deck The Walls, Inc.*, 925 F. Supp. 576 (ND Ill. 1996), aff'd sub nom. *Lee v. ART Co.*, 125 F.3d 580 (7th Cir. 1997) (отклонение доводов *Mirage Editions*); *Precious Moments, Inc. v. La Infantil, Inc.*, 971 F. Supp. 66, 68-69 (DPR 1997) (отклонение иска против того, кто купил ткань, а затем включил ее в постельное белье); *Paramount Pictures Corp. v. Video Broadcasting Sys., Inc.*, 724 F. Supp. 808 (D. Kan. 1989) (иск о распространении отклонен доктриной первой продажи, отличающей *Mirage Editions*).

Зашита авторских прав начинается автоматически с момента создания произведения и обычно длится в течение жизни автора плюс 70 лет.<sup>29</sup> По истечении срока действия авторских прав произведения становятся общедоступными для использования, т.е. переходят в общественное достояние.

Существует множество исключений из авторского права. Правило, согласно которому авторское право защищает выражение, означает, что оно не запрещает никому использовать идеи или факты, раскрытие которых в произведении. «Идеи» включают в себя сюжеты историй. В более общем

---

<sup>29</sup>Согласно первому закону об авторском праве, срок защиты составлял всего 14 лет.

смысле авторское право не защищает утилитарные аспекты произведения, поэтому вы можете написать компьютерную программу, которая делает то же самое, что и другая программа, пока вы не копируете ее выражение.

Факты считаются «вне» авторского права, потому что они обнаружены, а не созданы. Это включает, например, открытие новых простых чисел. Но вы можете иметь авторское право на выбор, последовательность или расположение фактов или что-либо еще, что само по себе не подлежит авторскому праву. Классический пример — антология поэзии, являющейся общественным достоянием. Вы можете иметь авторское право на сборник, даже если отдельные части не защищены, если выбор, последовательность или расположение достаточно оригинальны. Алфавитное расположение фактов в типичном телефонном справочнике «белых страниц» не соответствует конституционному требованию оригинальности. Вы не получаете никакой защиты просто потому, что вы вложили деньги, время или усилия в сбор телефонных номеров.

Авторское право не распространяется на многие «обычные» способы использования произведения. Само по себе чтение книги не является объектом авторского права, поскольку не нарушает никаких прав владельца авторских прав. Пение песни в душе — это представление, но у владельца авторских прав есть право только на публичные выступления. Однако и здесь Интернет все изменил. Когда вы читаете документ в своем веб-браузере, ваш компьютер, вероятно, сделал копию документа. Таким образом, многие ранее обычные способы использования теперь влекут за собой создание копии, что поднимает вопросы авторских прав.

Сегодня существует много споров о «добросовестном использовании». Добросовестное использование — это защита от нарушения авторских прав, которая была призвана позволить людям несанкционированно использовать защищенные авторским правом произведения. Добросовестное использование позволяет рецензентам цитировать книги. Это очень сложная область права; является ли использование «добросовестным», зависит от таких факторов, как цель, характер, объем и экономический эффект использования.<sup>30</sup>

## Патент

Патентное право защищает изобретения и дает изобретателю (или его правопреемнику) право исключать других из производства, продажи или использования изобретения в течение 20 лет с даты подачи патента. В отличие от авторского права, патентное право защищает от независимого изобретения другим лицом.

Сделка здесь заключается в том, что в обмен на патент изобретатель должен предоставить достаточно информации в патентной заявке, чтобы

---

<sup>30</sup>17 Свод законов США § 107.

позволить «специалисту в этой области» создать изобретение без особых экспериментов. После выдачи патента заявка становится публичной. Делая информацию публичной, патентообладатель вносит вклад в хранилище знаний общества.

Однако патент не предоставляет никаких утвердительных прав; если вы запатентуете улучшение чьего-либо изобретения, вы не сможете использовать это улучшение, не нарушая базовый патент. Если вы изобретаете и патентуете новый препарат, вам все равно может потребоваться одобрение регулирующих органов, прежде чем вы сможете продавать препарат.

Чтобы быть патентоспособным, изобретение должно быть полезным, новым и «неочевидным» для «специалиста в данной области». Требования новизны и неочевидности означают, что изобретение должно быть достаточным развитием технологии, прежде чем будет предоставлено право исключения. Разработки, не соответствующие этим высоким стандартам, лишаются защиты.

### **Коммерческие секреты**

Третья область права — коммерческая тайна — также считается частью права интеллектуальной собственности, хотя на самом деле это не собственность. Коммерческая тайна — это коммерчески ценная деловая или иная информация, известная пользователю, но не конкурентам. Секретность, хотя и не абсолютная секретность, является сутью коммерческой тайны; необходимо принимать разумные меры предосторожности для защиты коммерческой тайны от раскрытия.

Существует очевидная связь между патентами и коммерческими секретами, поскольку оба защищают полезную информацию. Если полезная информация вообще не патентоспособна, выбора нет. Но кто-то может не захотеть патентовать патентоспособное изобретение по нескольким причинам. Вы можете не захотеть раскрывать информацию в патентной заявке. Кроме того, если вы не ожидаете, что технология будет ценной очень долго, возможно, не стоит получать патент, который будет действовать 20 лет.

Главный недостаток коммерческой тайны в том, что она не обеспечивает защиты от независимого изобретения или обратного проектирования. Поэтому коммерческая тайна неразумна, если секрет можно вычислить из продукта. Если же, с другой стороны, изобретение представляет собой процесс, используемый при изготовлении продукта, его может быть трудно перепроектировать. Несмотря на то, что Coca-Cola присутствует на рынке уже много лет, по-видимому, никто не придумал, как его скопировать.

## Конституционная сделка об авторском праве

Права интеллектуальной собственности являются средством достижения цели — содействия прогрессу знаний и технологий. Как однажды заявил Верховный суд, «монопольные привилегии, которые Конгресс может разрешить, не являются ни неограниченными, ни изначально предназначены для предоставления особой частной выгоды».<sup>31</sup>

Вышеприведенный отрывок показывает, что право интеллектуальной собственности уже давно обеспокоено ограничением потенциальной монопольной власти, предоставляемой авторским правом и патентным правом. Например, доктрина первой продажи не позволяет владельцам патентов и авторских прав контролировать рынок после продажи запатентованных продуктов или копий защищенных авторским правом работ.

Кроме того, закон об авторском праве долгое время толковался судами и создавался Конгрессом для сохранения баланса со свободой слова. Такие доктрины, как дилемма идея/выражение, доктрина добросовестного использования и ограниченный срок действия авторского права, обычно рассматриваются как уменьшающие потенциальный конфликт между авторским правом и свободой выражения.<sup>32</sup>

Интересно, что беспокойство по поводу монополий исторически связано с беспокойством о свободе слова. Английское авторское право долгое время функционировало как своего рода спонсируемый государством картель; в обмен на частную монополию на произведения издатели соглашались выступать в качестве полицейских прессы на службе правительственной цензуры — в частности, Библии и других религиозных произведений.<sup>19</sup>

Аналогичным образом, дилемма идей и ее выражения в законе об авторском праве гарантирует, что не подлежащие защите авторским правом факты и идеи, а также непатентуемые функциональные принципы остаются в общественном достоянии, чтобы будущие создатели могли на них опираться.

---

<sup>31</sup>Sony против Universal City Studios 464 US 417, 429 и 432 (1984).

<sup>32</sup>См. в целом Нил Вайнсток Нетанел, Распознавание авторских прав в рамках Первой поправки, 54 Stan. L. Rev. 1 (2001). 19 См. в целом Л. Рэй Паттерсон, Свобода слова, авторские права и добросовестное использование, 40 Vand. L. Rev. 1 (1987).

## Традиционный взгляд на обратную разработку

Исторически обратная разработка всегда была законным способом получения информации, содержащейся в продуктах массового рынка. Для многих технологических компаний обратная разработка продуктов конкурентов для изучения их инноваций является стандартной практикой. Действительно, суды США также рассматривали обратную разработку как важный фактор поддержания баланса в законодательстве об интеллектуальной собственности, а Верховный суд назвал обратную разработку «неотъемлемой частью инноваций».

Закон признает три основные цели законного обратного инжиниринга. Конкурентный обратный инжиниринг направлен на создание прямой замены. Обратный инжиниринг совместимости или взаимодействия направлен на выяснение того, как сделать продукт, который будет работать с реверсивно спроектированным продуктом. И, конечно, исследователи часто реверсируют продукты, чтобы получить знания без коммерческой цели.

### Коммерческая тайна и «неправомерные средства»

В целом, коммерческая тайна считается незаконно присвоенной только в том случае, если лицо или фирма неправомерно использует или раскрывает секрет в нарушение соглашения или конфиденциальных отношений, совершает иные противоправные действия (например, взяточничество, принуждение, нарушение права собственности) с целью получения секрета или приобретает секрет у лица, которое знает или имеет основания знать, что информация является незаконно присвоенной коммерческой тайной.

Большинство штатов, например Калифорния, прямо предусматривают, что обратная разработка является законным способом получения коммерческой тайны. Несколько причин поддерживают обратную разработку как надежный принцип закона о коммерческой тайне.<sup>33</sup>Покупка продукта на открытом рынке обычно дает покупателю личные права собственности на продукт, которые включают право разбирать продукт, измерять его, подвергать его испытаниям и т. п. Закон также рассматривает продажу продукта на открытом рынке как публикацию инноваций, которые он воплощает, и передачу их в общественное достояние, если только создатель не получил на них патентную защиту.

---

<sup>33</sup>См. в целом Памелу Самуэльсон и Сюзанну Скотчмер, Право и экономика обратного проектирования, 111 Yale LJ 1575 (2002).

Уязвимость коммерческих секретов к обратному проектированию является частью общей конституционной схемы. В деле Bonito Boats против Thunder Craft Boats Верховный суд отменил закон Флориды, запрещавший производителям лодок использовать существующие части лодок в качестве «заглушек» для процесса прямого формования, который давал конкурирующие продукты, поскольку закон «запрещал всей общественности заниматься формой обратного проектирования продукта, находящегося в общественном достоянии».<sup>34</sup> Суд пояснил, что обратная разработка является «неотъемлемой частью инноваций», которая, вероятно, приведет к появлению изменений в продукте, которые «могут привести к значительному прогрессу в технологии». Действительно, «конкурентная реальность обратной разработки может подтолкнуть изобретателя» к разработке дополнительных патентоспособных идей.

В таких случаях, как Bonito Boats, вопрос заключается в том, «вытесняется» ли закон штата федеральным законом. Когда федеральный закон и закон штата конфликтуют, либо напрямую, либо в рамках целей федеральной политики, закон штата проигрывает в соответствии с доктриной «конфликтного» вытеснения. Это вытекает из пункта о верховенстве Конституции, согласно которому федеральный закон обычно превалирует над законом штата.<sup>22</sup> Закон об авторском праве также содержит конкретный пункт о вытеснении, который обсуждается ниже.

### **Закон об авторском праве и проблема Промежуточное копирование**

До недавнего времени закон об авторском праве не беспокоился об обратном проектировании, поскольку не было особых причин для обратного проектирования книг, произведений искусства или музыки. Теперь, когда компьютерные программы стали «литературными произведениями», все стало совсем иначе. Поскольку многие компьютерные программы распространяются только в объектном коде, процесс обратного проектирования обычно требует первоначальной декомпиляции исходный код, что влечет за собой создание копии.

Суды США установили, что закон об авторском праве не обязательно запрещает обратную разработку, поскольку копирование, связанное с обратной разработкой, может быть «добросовестным использованием»: «Закон об авторском праве позволяет лицу, lawfully владеющему копией произведения, предпринимать необходимые усилия для понимания идей,

---

<sup>34</sup>Суд продолжил, заявив, что «в случае, если находящийся в общем обороте предмет не защищен патентом, «воспроизведение функционального атрибута является законной конкурентной деятельностью»» (22 Конституция США, ст. VI, п. 2).

процессов и методов работы произведения».<sup>35</sup> Это может быть правдой, даже если конечная цель обратного проектирования — коммерческая. Суды обычно полагаются на конституционную цель защиты авторских прав: «содействие „Прогрессу науки...“».<sup>36</sup> Доктрина добросовестного использования продвигает эту конституционную цель, «побуждая других свободно развивать идеи и информацию, передаваемые произведением».<sup>37</sup><sup>38</sup>

Ключевым делом здесь было дело Sega Enterprises Ltd. против Accolade, Inc.<sup>26</sup> Accolade дизассемблировала игровые программы Sega, чтобы получить информацию, необходимую для того, чтобы сделать свои игры совместимыми с игровой консолью Sega Genesis. Затем Accolade продавала свои собственные игры, конкурируя с играми, созданными Sega и ее лицензованными разработчиками. Accolade выдвинула защиту добросовестного использования в ответ на заявления Sega о том, что дизассемблированные копии нарушают авторские права. Суд принял защиту Accolade по причинам, описанным выше. Он также отметил, что если Accolade не сможет дизассемблировать код Sega, Sega получит «фактическую монополию на [незапищенные] идеи и функциональные концепции [в программе]», что доступно только в соответствии с патентным правом.<sup>39</sup>

Однако решение суда ограничивалось обратным проектированием, осуществляемым по «законной причине», например, для получения доступа к функциональным спецификациям, необходимым для создания совместимой программы, и только в том случае, если это «предоставляет единственный способ доступа к тем элементам кода, которые не запищены авторским правом».<sup>40</sup>

### Патентное право

В патентном праве нет общей защиты добросовестного использования или исключения обратного проектирования. Теоретически вам не нужно проводить обратное проектирование запатентованного продукта, поскольку патентная спецификация должна информировать

---

<sup>35</sup>Atari Games Corp. против Nintendo, 975 F.2d 832, 842 (Федеральный округ, 1992 г.); см. Sony Computer Ent. Corp. против Connectix Corp., 203 F.3d 596 (9-й округ, 2000 г.).

<sup>36</sup>Там же, цитата из Конституции США, статья I, §8, п. 8.

<sup>37</sup>Feist Publications, Inc., против Rural Telephone Serv. Co., Inc., 499 US 340, 350 (1991).

<sup>38</sup>F.2d 1510 (9-й округ 1992 г.).

<sup>39</sup>Там же, 1526-1527.

<sup>40</sup>Там же, 1518.

соответствующее техническое сообщество о наилучшем способе создания изобретения.

Некоторые виды деятельности по обратному проектированию не нарушают патент. Например, покупатель машины, воплощающей запатентованное изобретение, обычно может разобрать ее, чтобы изучить, как она работает, в соответствии с принципом первой продажи патентного права. Покупка продукта означает, что у вас есть право использовать его, и простое его изучение не нарушает исключительных прав владельца патента на создание или продажу изобретения. Тем не менее, суды иногда применяют договорные ограничения на обратное проектирование.<sup>29</sup>

Также тот, кто пытается запатентовать изобретение, чтобы удовлетворить научные<sup>41</sup> любопытство может иметь защиту «экспериментального использования». Согласно законодательству США, эта защита узкая и, вероятно, не включает исследовательское использование, которое может привести к разработке патентоспособного изобретения или коммерческого продукта.<sup>42</sup>

Столкновение между этими тремя областями можно увидеть, если снова взглянуть на ситуацию с Sega. Предположим, что у Sega был патент на алгоритм, используемый во всех ее игровых программах. Разбирая программы Sega, Accolade, возможно, «создает» или «использует» запатентованный алгоритм, даже если он сделал это непреднамеренно. Короче говоря, проблема промежуточного копирования вновь возникает в патентном контексте.

## **Новые задачи для специалистов по обратному проектированию**

Важность обратного проектирования только возросла с ростом коммерческой криптографии в продуктах массового рынка, поскольку невозможно сделать системы более безопасными, не пытаясь их взломать. По иронии судьбы, растущее использование шифрования способствовало

---

<sup>41</sup>См. Pioneer Hi-Bred Int'l, Inc. против DeKalb Genetics Corp., 51 USPQ2d (BNA) 1797 (SD Iowa 1999) (введение «бирки на пакете», запрещающей покупателям защищенных PVPA семян кукурузы использовать эти семена в селекционных или исследовательских целях).

<sup>42</sup>См. Roche Prod. v. Bolar Pharmaceutical Co., 733 F.2d 858, 858-63 (Fed. Cir. 1984) (защита не разрешает «нелицензированные эксперименты, проводимые с целью адаптации запатентованного изобретения к бизнесу экспериментатора», в отличие от экспериментов, проводимых «для развлечения, удовлетворения праздного любопытства или для строго философского исследования»); Ребекка С. Айзенберг, Патенты и прогресс науки: исключительные права и экспериментальное использование, 56 U. Chi. L. Rev. 1017, 1023 (1989).

принятию законов против обратного проектирования. Например, индустрия развлечений теперь полагается на шифрование и другие технологии для защиты цифровой информации, такой как музыка на CD и фильмы на DVD, от несанкционированного копирования. Неудивительно, что были приняты новые законы, чтобы помешать людям «обходить» шифрование и другие формы безопасности.

Правовые посягательства на обратную разработку не ограничивались шифрованием. В 1970-х и 1980-х годах некоторые штаты запретили использование процесса прямого формования для обратной разработки корпусов лодок.<sup>43</sup> В конце 1970-х и начале 1980-х годов полупроводниковая промышленность добивалась и добилась принятия законодательства, защищающего топологии микросхем от обратного проектирования с целью изготовления клонированных микросхем.<sup>44</sup> В крупном международном соглашении о правах интеллектуальной собственности ничего не говорится об обратном проектировании.<sup>45</sup>

## Закон об авторском праве в цифровую эпоху и Проблема несанкционированного доступа

DMCA — один из важнейших законов, которые сейчас регулируют обратную разработку. Одна часть DMCA — его положения о «противодействии обходу» — предоставляет правовую защиту техническим мерам, которые эффективно контролируют доступ к или предотвращают копирование защищенных авторским правом работ. К сожалению, DMCA чрезвычайно сложен; например, DMCA делает незаконным обход «эффективных технических мер защиты» без четкого указания того, что означает этот термин.

---

<sup>43</sup>Эти законы штата были отменены Верховным судом в деле Bonito Boats.

<sup>44</sup>Закон о защите полупроводниковых чипов, Pub. L. No. 98-620, 98 Stat.

3347 (1984) (кодифицирован в 17 USC §§ 901-914 (1994)). Мы не будем обсуждать этот закон, за исключением того, что отметим, что он содержит особыю привилегию обратного проектирования, которая позволяет копировать защищенные проекты чипов с целью изучения компоновки схем, а также включение ноу-хау, полученных в результате обратного проектирования, в новый чип. Интересно, что обратные инженеры должны заниматься достаточным «прямым проектированием», чтобы разработать оригинальный проект чипа, который сам по себе подпадает под защиту SCPA.

<sup>45</sup>Соглашение по торговым аспектам прав интеллектуальной собственности (ТРИПС), 15 апреля 1994 г., Марракешское соглашение об учреждении Всемирной торговой организации, Приложение 1C, Юридические инструменты — Результаты Уругвайского раунда, том 31, 33 ILM 81 (1994). Положение о коммерческой тайне Соглашения ТРИПС — статья 39, 33 ILM на стр. 98.

## Несанкционированный доступ

DMCA по сути создает новое право «доступа» для владельцев авторских прав. Представители индустрии авторских прав сравнивают акт обхода технической системы защиты со «взломом и проникновением» в дом.

Одним из простых примеров являются программы-цензоры, используемые школами и библиотеками для предотвращения просмотра детьми неподходящих изображений. Эти программы часто содержат зашифрованные «черные списки» цензурированных веб-сайтов, которые поставщики обычно рассматривают как коммерческую тайну. Предположим, исследователь обнаруживает, что определенная программа блокирует сайты, которые полностью подходят для детей, и хочет прочитать черный список, чтобы выяснить, сколько подходящих веб-сайтов ошибочно блокируется.<sup>46</sup> Поскольку поставщик зашифровал черный список, чтобы помешать другим лицам получить доступ к его контенту, а список, по всей видимости, представляет собой сборник фактов, защищенных авторским правом, шифрование является технической мерой защиты, применяемой к работе, защищенной авторским правом, и несанкционированная расшифровка будет незаконным актом обхода — за исключением того, что в настоящее время в DMCA предусмотрено временное исключение для расшифровки черных списков цензурного ПО.

Другой пример: киноиндустрия использует схему шифрования, называемую Content Scrambling System (CSS), для защиты фильмов на DVD. В случае 2600,<sup>47</sup><sup>48</sup> CSS был признан технической мерой, которая «эффективно» контролирует доступ к фильмам. Обход CSS без разрешения владельца авторских прав является незаконным «актом обхода» в соответствии с DMCA. Обратите внимание, что суды не установили, что доктрина добросовестного использования применима к DMCA (в отличие от закона об авторском праве). Таким образом, если использование CSS не позволяет вам быстро перематывать рекламу в фильме на DVD, «обходить» это ограничение все равно незаконно.

Обратите внимание, что понятие «эффективности» здесь не связано с криптографической эффективностью. Даже слабое шифрование является «эффективным» согласно DMCA, поскольку обычный человек не может его преодолеть.

## Технологии обхода

DMCA запирает технические меры вторым способом: его положения, направленные против устройств, запрещают производство и

<sup>46</sup>См., например, <http://www.sethf.com>.

<sup>47</sup>Universal City Studios против Реймердеса, 111 F.Supp. 294 (SDNY

<sup>48</sup>), утверждено 273 F.3d 429 (2d Cir. 2001).

распространение технологий, позволяющих обходить защиту.<sup>49</sup> Продолжая метафору «взлома и проникновения», представители индустрии авторских прав сравнивают технологии обхода с «инструментами взломщиков», которые являются незаконными во многих штатах.

Раздел 1201 DMCA гласит, что «никто не должен производить, импортировать, предлагать публике, предоставлять или иным образом торговать любой технологией, продуктом, услугой, устройством, компонентом или частью этого», если они обладают одной или несколькими из следующих трех характеристик: (1) если они «в первую очередь разработаны или произведены с целью обхода [технической] защиты», (2) если они имеют «только ограниченную коммерчески значимую цель или использование, отличное от обхода [технической] защиты», или (3) если они «продаются этим лицом или другим лицом, действующим от его имени, с ведома этого лица для использования в обходе технической защиты».

Обратите внимание, что эти положения применяются не только к новому праву «контроля доступа», но и к правам владельцев авторских прав в целом. Таким образом, технологии, которые обходят меры защиты от копирования для компакт-дисков, могут быть незаконными в соответствии с этими положениями.

Вспомним два приведенных примера. В деле 2600 речь шла о программе DeCSS, которая позволяет людям расшифровывать DVD-фильмы, защищенные CSS. DeCSS была признана запрещенной технологией обхода. В примере с цензорским ПО исключение DMCA разрешает акт расшифровки, но ничего не говорит о том, может ли исследователь цензорского ПО предоставить доступ к компьютерной программе, используемой для расшифровки зашифрованного черного списка, или даже к деталям метода расшифровки.

## Обзор исключений DMCA

Так же, как вы не можете провести обратную разработку объектного кода без его декомпиляции или дизассемблирования, вы не можете провести обратную разработку технической меры защиты без ее обхода. Более того, для фактического выполнения обратной разработки часто требуется технологическое устройство или инструмент, поэтому запрет на технологию обхода также ограничивает обратную разработку.

---

<sup>49</sup>DMCA охватывает два различных вида технологий в зависимости от того, что они защищают: технологии, которые «эффективно контролируют доступ к [охраняемым авторским правом] работам», раздел 1201(a)(2), и технологии, которые «эффективно защищают[] право владельца авторских прав... на работу или ее часть». Раздел 1201(b)(1).

В совокупности эти положения DMCA создают серьезные препятствия для криптографов и исследователей безопасности, которые хотят анализировать меры безопасности, используемые в реальных продуктах массового рынка. Коммерческий обратный инженер, обнаруживший проблему с технической мерой другой фирмы и предложивший предложения по ее улучшению, рискует быть обвиненным в совершении уголовного преступления DMCA.

Даже академический специалист по обратному проектированию рискует быть привлеченным к ответственности за публикацию статьи о недостатках мер безопасности компании, поскольку такая статья может быть названа «инструментом обхода».<sup>50</sup> Одним из примеров является профессор Принстона Эдвара Фелтен, который собрал и ввел команду ученых в музыкальной индустрии «SDMI Challenge», конкурс по взлому цифровых водяных знаков и других технологий, рассматриваемых Secure Digital Music Initiative для защиты цифровой музыки. Фелтен и его команда приняли участие в конкурсе с намерением использовать SDMI Challenge в качестве реального исследования безопасности, и в конечном итоге они написали рецензируемую научную работу, которая должна была быть представлена на конференции. Перед тем, как работа была фактически представлена, Ассоциация звукозаписывающей индустрии Америки (RIAA) направила Фелтену и организаторам конференции письмо с предупреждением о том, что публикация работы нарушит законы об интеллектуальной собственности, включая DMCA.

DMCA также содержит несколько исключений, относящихся к обратному проектированию: обход технической системы защиты, когда это необходимо для достижения взаимодействия между компьютерными программами; обходы, осуществляемые в ходе законных исследований шифрования; и обход в целях тестирования компьютерной безопасности. К сожалению, каждое из этих исключений является и сложным, и узким. Даже когда акт обратного проектирования разрешен, DMCA строго регламентирует, что можно делать с полученной информацией.

**1201(f): обратная разработка для обеспечения совместимости** Это исключение позволяет обходить технические меры защиты для обратного проектирования взаимодействия. Оно также позволяет, в очень ограниченной степени, распространение информации, полученной в результате обратного проектирования. Обратите внимание, что 1201(f) не исключил бы атаку Фелтена на водяные знаки SDMI, поскольку она не имела никакого отношения к взаимодействию.

Дело 2600, упомянутое ранее, касается публикации компьютерной программы, известной как «DeCSS», на веб-сайте журнала 2600 Magazine.

---

<sup>50</sup>Хотя это и кажется странным, учтите, что многие научные работы по ценным бумагам включают в себя код компьютерной программы.

DeCSS может использоваться для обхода CSS, технической меры защиты, используемой для контроля доступа к фильмам на DVD. EFF, представлявшая журнал 2600 Magazine, утверждала, что DeCSS соответствует привилегии взаимодействия 1201(f). Мы утверждали, что DeCSS была разработана для того, чтобы позволить людям создавать программное обеспечение, которое позволило бы им воспроизводить законно купленные фильмы на DVD на их любимой платформе, а именно на компьютерных системах Linux.

Суды отклонили этот аргумент, заявив, что 1201(f) разрешает обход только в целях достижения взаимодействия программ, тогда как DeCSS допускает взаимодействие программ и данных, которое 1201(f) не охватывает. Это решение сомнительно, поскольку на DVD-дисках с фильмами есть как компьютерные программы, так и данные.

Хотя 1201(f) следует примеру Sega, разрешая обратную разработку для обеспечения совместимости, она является более ограничительной в нескольких отношениях: совместимость является единственной законной целью, для которой может быть выполнена обратная разработка; допускается только межпрограммная совместимость, хотя для достижения совместимости оборудования и программ или совместимости программ и данных может потребоваться обход; и информация, полученная в результате обратной разработки, не может быть свободно опубликована.

### **1201(g): исследование шифрования**

DMCA также содержит прямое исключение для исследований шифрования. К сожалению, оно также очень узкое. Во-первых, это исключение применяется только в том случае, если криптограф запросил (даже если он или она не получили) разрешение у владельца авторских прав на совершение действия по обходу до того, как обход будет завершен. Во-вторых, в законе подчеркивается необходимость для криптографа быть экспертом, чтобы иметь право на это исключение, хотя некоторые из самых блестящих умов в области криптологии не имеют формального образования. В-третьих, закон разрешает криptoаналитику создавать инструменты для обхода контроля доступа, но умалчивает о том, допустимы ли инструменты для обхода контроля использования или копирования (то есть он содержит исключение из одного, но не обоих правил против устройств). В-четвертых, он регулирует способность криптолога распространять результаты расшифровки.

Рассмотрим еще раз исследование SDMI профессора Фелтена: оно не подпадает под действие статьи 1201(g), поскольку цифровые водяные знаки не являются шифрованием.

### **1201(j): исследование безопасности**

Освобождение от исследования безопасности DMCA имеет похожую структуру: оно применяется только в том случае, если тестировщик

заранее просит об этом, и также позволяет создавать инструменты только для обхода контроля доступа, а не для копирования или использования контроля. Как и 1201(g), оно также регулирует распространение тестирующим результатах тестирования.

Даже в этой узкой форме неясно, будет ли исследование Фелтена охвачено. Раздел 1201(j) разрешает только создание инструмента для обхода контроля доступа. Является ли цифровой водяной знак контролем доступа или контролем копирования? Ответ на этот вопрос во многом зависит от того, как используется водяной знак. EFF утверждала, что, как предполагала RIAA, технологии водяных знаков SDMI являются как технологиями контроля доступа, так и технологиями контроля копирования.

### **Лицензионные соглашения с конечным пользователем и Договорные запреты на Обратный инжиниринг**

Интеллектуальная собственность — не единственное препятствие для обратного проектирования. Лицензии на программное обеспечение часто запрещают обратное проектирование. Типичное положение лицензии может гласить: «Вы не можете и не можете разрешать другим делать следующее: (a) разбирать, декомпилировать или иным образом извлекать исходный код из Программного обеспечения, (b) осуществлять обратное проектирование Программного обеспечения, (c) изменять или создавать производные работы Программного обеспечения, (d) копировать Программное обеспечение, за исключением случаев, прямо разрешенных в настоящем Соглашении, (e) сдавать Программное обеспечение в аренду или лизинг или (f) использовать Программное обеспечение любым способом, который нарушает права интеллектуальной собственности или другие права Лицензиара или другой стороны».

Компании утверждают, что такие положения юридически обязывают покупателей не заниматься обратной разработкой своего программного обеспечения. Если они все равно это сделают, они нарушают контракт и могут быть привлечены к ответственности за ущерб. Проблема, конечно, в том, что положение о запрете обратной разработки дает владельцу авторских прав права, выходящие за рамки тех, которые он имел бы, скажем, по решению Sega.

Является ли этот вид договорного запрета исполнимым — это горячо обсуждаемый вопрос. Иногда суды отклоняли защиту обратного проектирования в делах о коммерческой тайне, поскольку эта деятельность выходила за рамки лицензированного использования программного обеспечения.<sup>38</sup> Иногда суды отказывались применять ограничения лицензии на программное обеспечение в отношении

обратного проектирования из-за конфликта между пунктом и федеральной политикой в области интеллектуальной собственности. В деле *Vault Corp.* против *Quaid Software Ltd.*<sup>39</sup> производитель программного обеспечения для защиты от копирования пытался применить пункт о запрете обратного проектирования в соответствии с законодательством Луизианы против фирмы, которая выполнила обратное проектирование своей схемы защиты от копирования. Суд постановил, что федеральный закон имеет преимущественную силу над договорным положением в качестве вопроса федеральной политики, тот же аргумент использовался в деле *Bonito Boats* для отмены закона о корпусе лодки во Флориде.

Кроме того, раздел 301 Закона об авторском праве отменяет права, созданные или поддерживаемые государством, «которые эквивалентны любым исключительным правам в рамках общей сферы действия авторского права...». Как и следовало ожидать, ведутся споры о том, что означает «эквивалентный». Суды заявили, что положения контракта, подлежащие принудительному исполнению в соответствии с законодательством штата, «эквивалентны» федеральному авторскому праву, когда условия нарушения одинаковы. Но если нарушение права, созданного государством, требует «дополнительного элемента», оно не является «эквивалентным».

---

<sup>38</sup> Например, *Technicon Data Sys. Corp.* против *Curtis 1000, Inc.*, 224 USPQ (BNA) 286 (Del. Ch. 1984) (постановил, что консультант больницы использовал ненадлежащие средства для получения информации об интерфейсе, составляющей коммерческую тайну, путем прослушивания лицензированной системы программного обеспечения больницы с целью изучения способа, которым серверное программное обеспечение обменивалось данными с клиентским программным обеспечением, поскольку такое использование не было разрешено больницей; далее заявил, что даже если бы использование было разрешено, действие нарушило бы ограничительные условия лицензии); см. также *DSC Communications Corp.* против *Pulse Communications, Inc.*, 170 F.3d 1354 (Fed. Cir. 1999) (постановил, что имел место спорный вопрос факта относительно того, привело ли использование Pulsecom «доски слежения» в телефонной компании для получения доступа к информации об интерфейсе программного обеспечения DSC к незаконному присвоению коммерческой тайны ввиду ограничений в лицензии телефонной компании на использование программного обеспечения DSC).

<sup>39</sup> 847 F.2d 255 (5th Cir. 1988).

Недавно такой пункт договора был признан имеющим юридическую силу.<sup>40</sup> *Boyer* против *Baystate Technologies, Inc.*,<sup>41</sup> изобретатель продал

---

<sup>40</sup> F.3d 1334 (Федеральный циркуляр 2002 г.).

запатентованное программное обеспечение автоматизированного проектирования (САПР) «инструментарий» с оговоркой о лицензии против обратного проектирования. Baystate, конкурент, провел обратный инжиниринг программного обеспечения Bowers, а затем продал конкурирующий САПР-инструментарий. После некоторых сложных судебных разбирательств суд в конечном итоге постановил, среди прочего, что Baystate нарушила свой контракт с Bowers.

Суд постановил, что лицензия не была перенята, поскольку в договоре есть «дополнительный элемент» — стороны должны прийти к соглашению.<sup>52</sup> Из этого следует, что федеральный закон об авторском праве никогда не может преобладать над договорным запретом. Проблема с решением Боуэрса заключается в том, что оно фокусируется только на конкретном пункте о преобладании Закона об авторском праве и полностью игнорирует конституционное «конфликтное» преобладание.<sup>53</sup>

Единый закон о транзакциях с компьютерной информацией (UCITA) представляет собой законодательную попытку решить эти проблемы, но он также вызывает споры.

## **Коммерческие тайны и Экономика Закон о шпионаже**

Закон об экономическом шпионаже (ЕЭШ)<sup>54</sup><sup>55</sup>создало первый федеральный иск о незаконном присвоении коммерческой тайны. Но в нем нет защиты обратной инженерии. Это вызывает беспокойство, поскольку права, предоставленные в рамках ЕЭЗ, возможно, подразумевают определенные действия по обратной инженерии, которые ранее считались законными. В частности, неясно, могут ли декомпиляция и дизассемблирование компьютерных программ нарушать правила ЕЭЗ, запрещающие копирование коммерческой тайны.

---

<sup>52</sup>Суд сослался на более раннее дело ProCD, Inc. против Зейденberга, 86 F.3d 1447, 1454. (7th Cir. 1996) («Авторское право — это право против всего мира. Контракты, напротив, обычно затрагивают только их стороны; посторонние лица могут поступать так, как им заблагорассудится, поэтому контракты не создают «исключительных прав»»).

<sup>53</sup>EFF представила заключение в поддержку ходатайства Baystate о повторном слушании дела в полном составе. [добавить цитату]

<sup>54</sup>Закон об экономическом шпионаже 1996 года, Публичный закон № 104-294, 110 Стат.

<sup>55</sup>(кодифицировано в 18 USC § 1831-1839 (Supp. V 1999)).

## Ответственный хакер: невежество не является защитой

В целом, существует два способа нарушения законов об интеллектуальной собственности. Прямое нарушение означает, что вы фактически нарушили. Косвенное нарушение означает, что вы способствовали фактическому нарушению кем-то другим. Например, в деле Betamax вопрос заключался в том, может ли Sony, продавая видеомагнитофоны, быть признана ответственной за нарушение авторских прав своих клиентов.

### Гражданские и уголовные правонарушения и наказания

Юридические теории, о которых мы говорили, несут широкий спектр потенциальных штрафов. Главной проблемой является гражданская ответственность, либо экономический ущерб, либо запрет на деятельность, либо и то, и другое. Ущерб обычно связан с размером вреда, причиненного нарушением.

В патентном праве, например, обычной основой для возмещения ущерба является «разумный роялти». Суд рассчитает, сколько вы должны были заплатить владельцу патента в виде роялти, если бы заключили договор на лицензию. Ущерб также может быть основан на прибыли нарушителя или упущенном выигоде владельца патента.

«Умышленное» нарушение рассматривается более жестко. Патентный закон позволяет суду по своему усмотрению увеличить размер ущерба до трехкратного размера базового ущерба (а также оплатить гонорары адвоката владельца патента), если нарушитель знал о патенте и не консультировался с компетентным патентным юристом.

Текущая тенденция в законодательстве об интеллектуальной собственности направлена на большее внимание к уголовным наказаниям. Согласно первому федеральному закону об авторском праве 1790 года, нарушение авторских прав было чисто гражданским делом. Только в 1897 году Конгресс добавил уголовные наказания в закон об авторском праве, и уголовное нарушение авторских прав было классифицировано как проступок.<sup>56</sup> Более того, уголовное преследование за нарушение авторских прав применялось редко.

Сегодня риск уголовного преследования кажется значительно выше, а уголовные наказания намного серьезнее. Например, поправки к закону об

---

<sup>56</sup> См. в целом Лидия Лорен, Оцифровка, Коммодификация, Криминализация: эволюция уголовного нарушения авторских прав и важность требования преднамеренности, 77 Вашингтон. ULQ 835, 840 (1999).

авторском праве 1982 и 1992 годов классифицировали некоторые виды нарушений как тяжкие преступления. Однако даже тогда уголовное нарушение должно было быть совершено преднамеренно и ради коммерческой выгоды или личной финансовой выгоды.

Закон 1997 года No Electronic Theft Act (NET Act) криминализировал воспроизведение или распространение одной или нескольких копий защищенных авторским правом произведений, общая розничная стоимость которых превышает 1000 долларов США в течение любого 180-дневного периода, независимо от того, как эти копии были созданы или распространены. Он сохранил требование преднамеренности, но устранил требование о том, чтобы нарушение ответчика было мотивировано прибылью или коммерческой выгодой.

DMCA также содержит положения об уголовной ответственности, которые были использованы в судебном преследовании Дмитрия Склярова и компании ElcomSoft, в которой он работал. Elcomsoft производила и распространяла программное обеспечение, которое может использоваться для преобразования цифровых книг из формата электронных книг Adobe в формат PDF Adobe. В ходе преобразования формата ограничения на использование, налагаемые форматом электронных книг, снимаются. Неоспоримым было то, что программное обеспечение Elcomsoft может использоваться для содействия ненарушающему использованию электронных книг (например, добросовестное использование выдержек или содействие автоматическому переводу на шрифт Брайля для слепых читателей). Сам Скляров никогда не обвинялся в нарушении авторских прав или содействии в нарушающей деятельности какой-либо третьей стороны. Тем не менее, для

За его участие в разработке программного обеспечения ФБР арестовало его и держало под стражей в течение 3 недель.<sup>57</sup><sup>58</sup> Ему и Elcomsoft было предъявлено обвинение большинством жюри присяжных; согласно обвинительному заключению, Склярову грозило максимальное наказание в виде 25 лет лишения свободы и штраф, который мог превышать 2 миллиона долларов.<sup>59</sup> ElcomSoft и Скляров в конечном итоге были признаны невиновными в нарушении DMCA.

---

<sup>57</sup> См. профессор Ларри Лессиг, «Тюремное заключение в цифровую эпоху», Нью-Йорк.

Времена (30 июля 2001 г.) (доступно на <<http://www.nytimes.com/08/07/30/opinion/30LESS.html>>); Деклан МакКаллах, «Арест хакера вызвал протест», Wired News (19 июля 2001 г.) (доступно по адресу <<http://www.wired.com/news/politics/0,1283,45342,00.html>>); Дженинифер 8 Ли, «США арестовали российского криптографа как нарушителя авторских прав», NY Times, 18 июля 2001 г.

<sup>59</sup> См. Брэд Кинг и Мишель Делио, «Скляров, босс не признал себя виновным», Wired News (30 августа 2001 г.) (доступно по

## Обратный инжиниринг как «свобода для экспериментов» и другие правовые вопросы

Эдвард Фелтен, профессор компьютерных наук в Принстонском университете, рассматривает обратную разработку как часть «свободы возиться», которая должна включать свободу «разбирать их, обсуждать их, исследовать, как они работают, модифицировать их, делать их лучше». Фелтен утверждает, что «поскольку все больше и больше нашего мира воспринимается через электронные устройства, а коммуникации и культура все больше и больше опосредуются этими устройствами, становится все более важным, чтобы мы могли возиться с ними, чтобы иметь возможность понимать эту часть нашего мира».

Свобода чинить должна также включать право говорить о чинке. Но, как мы видели, многие из новых правил интеллектуальной собственности ограничивают право обратных инженеров делиться тем, что они узнают из чинки. Эти ограничения не только поднимают серьезные вопросы свободы слова в соответствии с Первой поправкой, они затрагивают суть конституционной основы авторского права и патентного права: прогресс в искусстве и науке. Одним из основных вопросов, поднятых DMCA, является его сдерживающий эффект на ученых.<sup>60</sup>

---

адресу  
<<http://www.wired.com/news/politics/0,1283,46396,00.html>>).

<sup>60</sup>См. в целом Electronic Frontier Foundation, Непреднамеренные последствия: четыре года в соответствии с DMCA (2003) [ссылка на веб-сайт EFF]



## CHAPTER13

# Вперед!

Уровень техники взлома Xbox постоянно совершенствуется. Тысячи хакеров постоянно исследуют, изобретают, открывают и делятся новыми методами и приемами, чтобы сделать Xbox более полезным и ценным устройством для конечных пользователей. Быть в курсе последних разработок в области взлома может быть ошеломляющим. Надеюсь, чтение этой книги дало вам возможность понять последние сообщения и новости на различных веб-сайтах и веб-форумах, посвященных взлому Xbox. В этой главе обсуждается, куда можно обратиться, чтобы узнать больше о последних взломах, куда обратиться за помощью и как вы можете внести свой вклад в сообщество, используя свои уникальные способности и точку зрения. В этой главе также обсуждаются некоторые из более серьезных проблем, с которыми хакеры сталкинутся в будущем, а именно инициативы по созданию доверенных ПК.

## Хакерское сообщество

Хакеры Xbox — это анархическое сообщество, которое в основном работает подпольно, поддерживая связь и обмениваясь информацией через различные интернет-форумы.

(*fora* — множественное число от *forum*, так как *data* — множественное число от *datum*). Большая часть сообщества хакеров Xbox не выходит на публику, и хакеры часто используют псевдонимы, чтобы защитить свою личность. Причины использования псевдонимов различаются, но в целом анонимность дает преимущество большей свободы действий. Хакеры более склонны делиться своими результатами и открытиями, если знают, что могут отступить невредимыми, если дела пойдут плохо. Использование псевдонимов также уравнивает шансы. Хакеры судят друг друга в первую очередь по качеству и частоте их вкладов и мало по чему другому. Тот факт, что вы можете быть молоды, не умаляет вашего первого впечатления или уличного авторитета, как это могло бы быть в других ситуациях. Аналогично, многие хакеры не стесняются быть резкими, когда вы совершили ошибку, и они еще меньше терпят глупость,

представленную как эрудиция или грубые утверждения. С другой стороны, многие хакеры с радостью протянут руку помощи тем, кто честно потрудился прочитать раздел часто задаваемых вопросов, поискать информацию в Интернете и вообще сделать все возможное, чтобы проверить и убедиться, что на вопрос еще не был дан ответ.

## Хакерские форумы

Сообщество хакеров Xbox имеет множество гражданских форумов для обмена результатами и выражения своих опасений. Наиболее популярными форумами являются веб-сайты BBS, такие как

как [www.XboxHacker.net](http://www.XboxHacker.net) и [www.xbox-scene.com](http://www.xbox-scene.com), а также каналы IRC, такие как #xboxhacker. Веб-сайты BBS обычно содержат журналы новостей, часто задаваемые вопросы и полезные ссылки на информацию. Что еще более важно, BBS включают форумы, где люди могут делиться информацией и задавать вопросы. С помощью этих форумов вы можете получить доступ к колективным знаниям всех хакеров, которые часто посещают эти BBS. Зарегистрированная история этих форумов также содержит массу информации о взломе Xbox (и дезинформации). Я призываю читателей, у которых есть вопросы из этой книги, на которые нет ответов, ознакомиться с этими форумами для получения ответов.

Одним из первых форумов по взлому Xbox был XboxHacker BBS. ([www.xboxhacker.net](http://www.xboxhacker.net)). Многие из лучших и самых ярких хакеров Xbox внесли свой вклад в форумы. Например, одна из тем форума документирует в режиме реального времени приключения Энди Грина (известного как numbnut на XboxHacker BBS), когда он взломал схему безопасности версии 1.1 Xbox. Я многому научился, читая сообщения на форуме XboxHacker BBS. Я также встретил некоторых из самых интересных людей через форумы BBS.

Основатель

XboxHacker BBS, Дэн Джонсон (также известный как SiliconIce) рассказывает свою историю в боковой панели «Профиль: Дэн Джонсон».

Еще одним ресурсом для поиска дополнительной информации об Xbox в целом является поисковая система Google ([www.google.com](http://www.google.com)). По мере того, как все больше хакеров вовлекаются в Xbox, Google становится все более важным инструментом для забрасывания широкой сети и обнаружения новейших инструментов и методов. Например, на момент написания статьи Google начал индексировать ряд ресурсов для замены компонентов Xbox. Это может быть полезно для тех, кто не хочет тратить время на адаптацию блока питания ATX для работы с Xbox.

Попробуйте использовать максимально конкретные ключевые слова при поиске в Google. Например, если ввести Xbox hacking в Google, вы получите большое количество ссылок, связанных с общей темой взлома Xbox, но мало конкретики. Например, сегодняшним самым популярным в Google по запросу «Xbox hacking» является статья LWN.net под названием «LWN: Windows CEO

финансирует конкурс по взлому Xbox (News.com). Это кажется довольно далеким от информации о том, как установить новый жесткий диск или подробностей о системе безопасности Xbox. При сужении поиска попытайтесь выяснить, каков фактический жаргон и написание для вашей концепции. Предположим, вы ищете информацию о новой системе безопасности Xbox. Если вы ищете по new Xbox, вы вряд ли получите какую-либо техническую информацию. Однако, если вы ищете по xbox v1.1, поиск вернет гораздо больше полезных технических результатов. Один из лучших способов собрать текущий жаргон и аббревиатуры — это просмотреть хакерские BBS.

## Внесение вклада

Если вы ищете способ внести свой вклад в сообщество хакеров Xbox, помните, что большинство хакеров обладают уникальным навыком или силой, которая обычно соответствует его или ее области наибольшего интереса, и что большинство хакеров взламывают ради развлечения. Например, мне очень нравится оборудование, особенно когда оно требует создания чего-то. Кроме того, хотя я и умею писать код, мне это не особенно нравится. Таким образом, мой вклад в сообщество хакеров в основном заключается в проектах по оборудованию. Аналогично, мне трудно мотивировать себя заниматься программными проектами. Поэтому большую часть времени я просто сижу и наслаждаюсь тем, что другие люди делают на Xbox. Это и познавательно, и развлекательно.

Пытаясь подумать о том, что вы можете сделать для сообщества хакеров Xbox, не беспокойтесь о том, чтобы быстро вскочить и заняться проектом, который вам не нравится или который вам не по душе.

## Профиль: Дэн Джонсон (он же SiliconIce)

Можете ли вы немного рассказать о себе и о том, как вы пришли в хакерство?

Некоторое время я интересовался хакерством электроники, хотя по большей части мой интерес ограничивался только чтением о таких подвигах. Взломанные устройства, такие как Sega's Dreamcast и Netpliance I-Opener, время от времени привлекали мое внимание. Однако мой первый настоящий опыт хакерства электроники произошел с «ePods», снятым с производства интернет-устройством/веб-планшетом. Я прочитал об этих интересных устройствах в Интернете и нашел свой путь к I-Appliance BBS Кена Сеглера (<http://www.linuxhacker.net>), родина знаменитых хаков I-Opener. Прочитав о классных вещах, которые люди делали с этими планшетами, я был заинтригован и потратил большую часть своих накопленных в то время денег (\$200) на один из них. Получив свою новую игрушку, я провел много времени, выполняя многие из задокументированных хаков, в основном на основе программного обеспечения. Это должно было стать моим первым взглядом в мир электронного хака. После ePods я перешел к другому интернет-устройству, Gateway Connected Touchpad. Изящный 10-дюймовый сенсорный экран с 400-мегагерцевым процессором Crusoe и 96 МБ оперативной памяти, размещенный за ним; это выглядело как выгодная сделка и идеально подходило для забавного проекта. Устройство работало под управлением специальной сборки Linux для AOL с карты CompactFlash объемом 32 МБ. Я заменил его на Microdrive, и после долгого обмена этим диском между ноутбуками, USB-ридерами и Gateway, устройство загрузило урезанную версию Windows с помощью «98Lite». Устройствоказалось идеальным для использования в качестве сетевого mp3-музыкального автомата с управлением пальцем (помимо прочего), поэтому я начал работать над специальным плеером, который бы

(продолжение)

Профиль: Дэн Джонсон (продолжение)

позволяют легко управлять с помощью сенсорной панели. Весь опыт Gateway стал прекрасным летним проектом между моими младшими и старшими классами средней школы.

#### Как появилась XboxHacker BBS?

Именно в этот период времени у меня возникла идея XboxHacker.Net. Мне Xbox показался потенциально идеальным вычислительным устройством для телевизора... как только его можно будет запрограммировать. Аппаратное обеспечение было самым впечатляющим для того времени и более чем достаточным для того, чего я надеялся добиться на устройстве. В отличие от других консолей, Xbox также должен был иметь встроенный жесткий диск и порт Ethernet. После взлома Xbox можно было использовать для эмуляции старых консолей, таких как NES, SNES, N64 и даже PSX, для удобного воспроизведения любого типа медиафайлов, таких как mp3 или DivX, на вашей домашней развлекательной системе, для воспроизведения потокового мультимедиа из домашней сети или даже в качестве базового ПК. Некоторые утверждали, что стоимость устройства в 299 долларов делала его нецелесообразным для взлома, поскольку ПК с аналогичными характеристиками можно было собрать без необходимости взлома за намного большую стоимость. Однако Xbox имел преимущество в том, что он также мог запускать игры Xbox и был изначально разработан для работы с вашим телевизором. Хаки были бы просто добавленной стоимостью. После моего короткого набега в мир взлома электроники, я подумал, что взлом Xbox может быть интересным проектом, который поможет организовать, так как меня заинтриговала возможность иметь удобную коробку для моего телевизора, чтобы выполнять задачи, упомянутые выше. Только много месяцев спустя я действительно приобрел домены XboxHacker.Net.

#### Каков был ваш опыт развития и управления XboxHacker BBS?

С самого начала XboxHacker.Net был сосредоточен на нескольких основных целях: предоставление и распространение технической информации об Xbox и предоставление места для коллег-хакеров для обсуждения технической информации, связанной со взломом Xbox. Хотя из-за моих ограниченных технических знаний и опыта я мог сделать немногое в плане реального взлома, чтобы внести свой вклад в усилия, я знал достаточно много, чтобы помочь облегчить усилия, собирая и распространяя соответствующую информацию и модерируя доску обсуждений.

Вскоре после запуска сайта нам посчастливилось получить несколько ссылок с таких известных сайтов, как The Inquirer Майка Маги и Van's Hardware Ван Сmita. XboxHacker.Net BBS быстро стала одним из основных мест в Интернете для обсуждения любых материалов, связанных со взломом Xbox, а новостная страница XboxHacker.Net содержала самые свежие новости о состоянии Xbox. Через несколько недель после запуска сайта он был упомянут в статье на CNET, и с тех пор уровень активности неуклонно рос. Вскоре XboxHacker.Net перерос небольшой общий сервер, на котором мы находились, поэтому сайт переехал на гораздо более

крупный аккаунт, который также перерос за считанные недели.

Трафик

(продолжение на стр. 198)

способный выполнять. Вы поймете, когда придет ваше время: проект просто выкрикнет ваше имя, и вы, естественно, будете вынуждены его взломать.

## Надежные вычисления

Доверие — краеугольный камень безопасности, и чтобы быть уверенным в своей системе безопасности, вам нужно быть уверенным в используемом вами оборудовании и программном обеспечении. Доверенный компьютер — это машина, спроектированная так, чтобы быть устойчивой к атакам, которые могут поставить под угрозу надежность машины. Существует множество подходов к созданию доверенного компьютера, от модели банкомата с физической безопасностью и устойчивостью к взлому до менее аппаратных решений, таких как те, которые используются в Xbox. Xbox вписывается в более широкую картину доверенных вычислений, поскольку это одна из первых громких, широко распространенных реализаций доверенных ПК. В некотором смысле Xbox дает нам намек на то, чего мы можем ожидать в будущем от доверенных вычислений,

Доверенные вычисления — это потенциально разрушительная новая технология. Рост числа доверенных клиентов в такой сети, как Интернет, обещает безопасные и конфиденциальные финансовые транзакции в режиме онлайн, сокращение или исключение возникновения компьютерных вирусов, а также сокращение или исключение спама в электронной почте. Доверенные ПК также могут использоваться для безопасного хранения конфиденциальных данных, таких как ваши медицинские и финансовые записи, ваши непристойные или смущающие секреты. Еще одно применение доверенных ПК — надежное обеспечение прав доступа к цифровому контенту и политик управления для пользователей. Аспект управления цифровыми правами (DRM) доверенных вычислений может кардинально изменить то, как мы используем компьютеры сегодня; многие из нас пользуются преимуществами псевдосвободного контента и гибкой реализации авторских прав. Основная проблема заключается не в существовании политик управления контентом — на самом деле, политика управления контентом может быть выгодна для потребителей. Реальная проблема заключается в том, что политики, которые регулируют ваши права, могут быть установлены поставщиками контента в одностороннем порядке. В текущих предложениях о доверенных ПК пользователю не доверяют контроль над определенными ключевыми секретами внутри его машин. Вместо этого информация об аттестации (информация, необходимая для установления надежности) машины пользователя частично поддерживается третьей стороной. Можем ли мы полагаться на невыборную, нерегулируемую третью сторону с деловыми интересами,

чтобы определить, кто может вести бизнес, отправлять электронную почту и иным образом быть признанным как заслуживающий доверия субъект?

Доверенные вычисления подобны оружию. Здорово иметь его, пока вы контролируете курок. К сожалению, многие противники доверенных ПК опасаются, что на практике системы будут развертываться с предустановленными правами, политиками и сторонними ресурсами подтверждения доверия, направленными в неправильном направлении для потребителей. Компания в хорошие времена может устанавливать политику конфиденциальности и безопасности в пользу пользователей, чтобы привлечь большую клиентскую базу, но как только стучится Глава 11 или компания продается или иным образом меняет владельца, эти политики могут и будут меняться. Что мешает бизнесменам продвигать доверенные вычисления изначально с помощью удобных для пользователя политик, а затем внезапно переходить в режим DRM, выжимающий деньги из кошелька? Этика?

Профиль: Дэн Джонсон (продолжение)

на форумы продолжало быстро расти, поэтому я принял решение перенести сайт на отдельный выделенный сервер, где у нас было бы место для роста и нам не приходилось беспокоиться о каждом из нескольких мегабайтов используемой пропускной способности или о том, сколько пользователей форума одновременно находились в сети.

Прошло совсем немного времени, прежде чем деятельность XboxHacker.Net привлекла внимание Microsoft. В начале истории сайта со мной пару раз связывались с просьбами удалить материалы. В первый раз я получил снимок экрана инструмента разработчика без особых объяснений, на котором были видны «сектора безопасности» на дисках. Во второй раз поступила просьба удалить ссылку, которую кто-то разместил на форумах на образ BIOS Xbox. За исключением этих нескольких незначительных инцидентов, контакты между Microsoft и XboxHacker.Net практически отсутствовали. С самого начала была принята политика избегать вопросов сомнительной законности, таких как обсуждение резервного копирования игр, ссылок на материалы, защищенные авторским правом, или размещение кода Microsoft из BIOS.

В начале, хотя интерес был высок, прогресс, казалось, был относительно медленным... с Xbox можно было сделать не так уж много, пока не была взломана система безопасности. Было много людей, которые внесли заметный вклад на раннем этапе. Одними из первых участников мира XboxHacking были Энди и Люк, чьи веб-страницы содержали множество информации о форматах файлов и другие ценные сведения об Xbox. Стив «SurferDude» Гельбах внес вклад в разработку схемы преобразователя VGA для Xbox, а Кен Гаспер усовершенствовал ее, чтобы создать настоящий адаптер VGA для Xbox. Ранний анализ Банни и страница информации об оборудовании вызвали интерес к XboxHacking после ее появления на Slashdot. Действительно, есть много других, которые также оставили свой след. Имея привилегию иметь таких опытных хакеров среди участников форумов, XboxHacker.Net превратился в центр технической информации, обсуждений и новостей Xbox.

Жемчужиной XboxHacker.Net, без сомнения, всегда была XboxHacker BBS. Сайт был сосредоточен вокруг форумов и деятельности наших участников. Часто последние новости XboxHacking были просто ссылками на темы форума и клипы из сообщений, сделанных нашими участниками. Целью форумов было дать возможность разнообразной, многонациональной группе людей, заинтересованных во взломе Xbox, работать вместе для достижения этой общей цели. Форумы стали отличным местом для обмена информацией и обсуждения идей с другими хакерами, а также объединили многих талантливых хакеров, которые в противном случае не имели бы возможности сотрудничать. Было здорово наблюдать за прогрессом, которого добились форумы. Одна конкретная дискуссия, которая приходит на ум, происходила в течение нескольких дней. Несколько участников обсуждали новую систему безопасности во втором поколении Xbox, искали недостатки и способы их обхода. Было захватывающе наблюдать за

тем, как разворачивались события, по мере того как хакеры приближались к решению и в конечном итоге взламывали безопасность на обновленном Xbox. Сотни наших участников форума внесли большой вклад в успех

(продолжение на стр. 200)

## Сделав шаг назад

Проблема с фразой «доверенные вычисления»: она стала синонимом криптографически защищенных доверенных компьютеров. Давайте сделаем шаг назад и просто поговорим об альтернативных подходах к построению доверенных компьютеров.

Надежность всегда была важна для компьютеров. Однако на заре вычислительной техники машины были настолько дорогими, что необходимое для обеспечения строгой политики доверия оборудование было недоступно потребителям. Например, многие ранние машины поставлялись с разъемом для микросхемы аппаратного блока управления памятью (MMU). MMU был одним из первых шагов к надежным моделям аппаратной памяти; частью работы MMU является обеспечение защиты доступа к памяти на уровне страниц. MMU продавались как опция, поскольку в то время они были довольно дорогими. К сожалению, движение к надежному оборудованию остановилось на MMU, отчасти потому, что компьютерные сети не существовали в какой-либо крупной форме до относительно недавнего времени. В сетевом мире данные требовалось защищать только от ошибок программистов и от доступа нескольких избранных пользователей с физическим доступом к машине. Сегодня компьютерам нужно что-то более сильное, чем просто MMU, что-то, что может обеспечить доверие перед лицом вирусов и удаленных злоумышленников, пытающихся использовать тонкие уязвимости программного обеспечения для запуска вредоносного кода.

Естественным расширением аппаратно-усиленной модели страничной виртуальной памяти MMU могут быть возможности адресации с тегированной моделью памяти. Тег памяти — это набор битов, которые регистрируют тип данных или кода, хранящихся в ячейке памяти. В тегированной модели памяти каждая ячейка памяти имеет набор битов тега, примерно так же, как каждая ячейка памяти в обычной реализации памяти с исправлением ошибок связана с некоторыми битами ECC. Биты тега помогают оборудованию применять политики управления типами данных; например, ячейка памяти, помеченная как часть данных, никогда не может быть случайно или намеренно выполнена как код. Возможность — это указатель, предоставленный доверенным ядром, который нельзя подделать. Свойство неподдельности предпочтительно реализуется аппаратно с помощью битов тега. Многие архитектуры также включают возможность применять границы доступа как часть аппаратной возможности.<sup>61</sup> Возможности и теги памяти — не новые идеи; в 1961 году

<sup>61</sup> Эффективную, высокопроизводительную аппаратную реализацию точных границ объектов с использованием тегированных возможностей можно найти в технической заметке под названием «Представление возможностей со встроенным адресом и почти точными границами объектов» Джереми Брауна, Дж. П. Гроссмана, Эндрю Хуанга и Тома Найта.

<http://www.ai.mit.edu/projects/aries/Documents/Memos/ODEH-05.pdf>

Burroughs B5000 использовал возможности (тогда называвшиеся дескрипторами) и тегированную память для защиты на аппаратном уровне от атак переполнения буфера и для изоляции кода от данных.<sup>62</sup>Операционные системы MIT PDP-1, Intel i432, IBM System/38, Mach и Amoeba также реализовали некоторые возможности

---

<sup>62</sup>«Архитектура Burroughs B5000 — 20 лет спустя и все еще впереди времени?» Аластер Дж. В. Майер.  
<http://www.ajwm.net/amayer/papers/B5000.html>

Профиль: Дэн Джонсон (продолжение)

XboxHacker.Net и прогресса мира XboxHacking в целом.

Помимо активности на XboxHacker.Net BBS, несомненно, было много других групп, тайно работавших над взломом Xbox. Одна такая группа, о которой я знал на IRC, включала в себя участников форума, а также других из подполья электронного хакерства. Имена здесь не будут упомянуты. Многие крупицы информации от независимых групп или отдельных лиц были переданы мне для отчетов с течением времени. Однако, несмотря на растущий интерес к XboxHacking и количество людей, вовлеченных в эту работу, должно было пройти некоторое время, прежде чем какие-либо крупные прорывы стали достоянием общественности.

Сцена XboxHacking довольно быстро набрала обороты после того, как модчики стали общедоступными летом 2002 года. К этому времени система безопасности Xbox была взломана несколькими группами, и это был лишь вопрос времени, когда модчики станут общедоступными. В этот момент работа по большей части перешла от взлома оборудования к разработке программного обеспечения. Дочерний сайт XboxHacker.Net, XboxDeveloper (<http://www.xboxdeveloper.net>) был запущен, чтобы помочь каталогизировать программное обеспечение, которое стало доступно для Xbox, хотя он так и не достиг уровня XboxHacker.Net. Используя просочившиеся копии Microsoft Xbox SDK, программисты начали писать и портировать различные приложения для Xbox, включая медиаплееры и эмуляторы, такие как MAME.

Проект Xbox-Linux (<http://xbox-linux.sourceforge.net/>) под руководством Майкла Штейла успешно запустили операционную систему Linux на Xbox. Я настоял на запуске проекта OpenXDK (<http://sourceforge.net/projects/openxdk/>), комплект разработчика с открытым исходным кодом, который позволил бы разрабатывать программное обеспечение для Xbox без каких-либо юридических проблем, хотя этот проект имел лишь переменный успех. Сейчас он находится под руководством «Caustik» (Аарон Робинсон), студента факультета компьютерных наук в Университете Кейс Вестерн. Из-за более доступной природы разработки программного обеспечения по сравнению с аппаратным взломом количество участников, вносящих вклад в общие усилия XboxHacking, быстро возросло. Количество «домашнего» программного обеспечения, доступного для Xbox сегодня, поражает и включает в себя все: от медиаплееров, эмуляторов консолей и утилит до изначально написанных игр.

форме, и это ни в коем случае не полный список систем, которые использовали возможности или маркированную память. Свойства безопасности возможностей также были продемонстрированы во многих академических исследованиях, таких как EROS (Extremely Reliable Operating System).<sup>63</sup> К сожалению, возможности и маркированная память так и не нашли своего места

<sup>63</sup>Первоначально задумано в Университете Пенсильвании Джонатан Шапиро (<http://www.eros-os.org/>)

в сердце архитектуры мейнстрима ПК. Безопасность и надежность всегда уступали место стоимости, обратной совместимости и производительности. Этот краткий урок истории показывает, что доверенные вычисления не требуют криптографического подхода, который сегодня предлагается Palladium и Trusted Computing Platform Alliance (TCPA). Фактически, криптография сама по себе не обеспечивает никакой безопасности. Безопасное управление ключами — это то, что действительно обеспечивает всю безопасность в Palladium/TCPA. Криптографические алгоритмы просто переносят безопасность ключа в домен пользователя.

При этом можно провести аналогию между возможностью и криптографическим ключом. Для обоих требуется доверенная ОС для управления их созданием, распространением и уничтожением. Оба одинаково слабы, если система не может защитить от поддельных ключей или возможностей. Главное отличие в том, что в случае утечки секретного ключа вся безопасность теряется навсегда. С другой стороны, возможности создаются и уничтожаются динамически, поэтому утечка возможности может привести к нарушению безопасности, но масштаб и продолжительность нарушения ограничены. В этом смысле возможности обеспечивают более надежное решение для компьютерной безопасности.

Обратите внимание, что полагаясь исключительно на криптографические методы для аппаратной безопасности, машины по-прежнему остаются открытыми для классических атак типа переполнения буфера и уязвимостей безопасности из-за ошибок программирования. «Измерения» состояния программного обеспечения помогают смягчить эту уязвимость, обнаруживая изменения кода до выполнения критически важных для безопасности операций, но измерения не являются идеальным решением. С другой стороны, атаки переполнения буфера невозможны в системах, использующих аппаратные возможности с проверкой границ.

Теги памяти также могут использоваться для реализации функций безопасности, которые невозможны при использовании чисто криптографического подхода к доверенным вычислениям. Одним из примеров является доверенная параллельная обработка секционированных секретов.<sup>64</sup> В этом примере несколько потоков с различными уровнями допуска к безопасности работают на одном процессоре. Аппаратное обеспечение применяет политику, в которой все потоки накладывают свой уровень безопасности на данные, к которым они получают доступ. Другими словами, каждое вычисление одновременно работает в двух доменах: домене обычной арифметики и домене безопасности.

---

<sup>64</sup>Более подробную информацию о подобных системах безопасности можно найти в технической заметке «Минимальная доверенная вычислительная база для динамического обеспечения безопасного потока информации» Джереми Брауна и Тома

Заметки/ОВЕН-15.pdf Рыцарь. <http://www.ai.mit.edu/projects/aries/Documents/>

Предположим, что неклассифицированный поток складывает два неклассифицированных числа и создает фрагмент данных с именем foo. Тег безопасности foo также вычисляется параллельно с арифметической операцией сложения. В этом случае результат тега безопасности — «неклассифицированный». Теперь предположим, что совершенно секретный поток касается foo: тег безопасности foo теперь меняется на «совершенно секретный». Неклассифицированные потоки больше не могут читать foo, даже если у неклассифицированного потока есть действительный указатель на foo; foo должен быть явно переклассифицирован, прежде чем его снова смогут прочитать неклассифицированные потоки.

Такая строго разделенная система безопасности может использоваться, например, для обеспечения того, чтобы никакие внутренние структуры ядра никогда не были доступны пользовательским процессам, даже при наличии ошибок и утечек памяти (включая сценарии, когда память ядра освобождается и переназначается пользовательскому процессу без шага очистки памяти). Эта схема также может использоваться для создания журналов аудита безопасности, которые полезны для помощи программистам в отслеживании первопричины нарушения безопасности, а также в качестве мер по контролю ущерба в случае нарушения безопасности.

Справедливости ради, одно из преимуществ криптографического подхода к доверенным вычислениям заключается в том, что если мошенник действительно завладеет данными с помощью некоторых аппаратных средств — подслушивая оборудование или крадя жесткий диск — данные не могут быть расшифрованы. Однако пользователи могут выбрать использование криптографии для защиты данных в любой компьютерной реализации, включая те, которые используют аппаратные возможности и маркированную память. Проблема безопасного управления ключами по-прежнему остается сложной проблемой, но, возможно, ее можно частично решить путем интеграции криптографических считывателей смарт-карт в ПК.

Нет существенной причины, по которой доверенная реализация ПК должна использовать криптографические методы, предложенные в Palladium и TCRA. Методы, рассмотренные в этом разделе, а именно возможности и маркированная память, могут быть использованы для реализации безопасного, доверенного ПК таким образом, который не влечет за собой риск потери пользователями возможности устанавливать собственные политики доступа. Пользователи будут полностью контролировать свою машину и свои секреты в любое время.

## Палладий против TCRA

В настоящее время существует много путаницы в отношении текущих предложений Trusted PC, а именно Palladium от Microsoft и Trusted Computing Platform Alliance (TCRA). Между этими двумя предложениями достаточно много сходств, поэтому многие люди думают, что это одно и то же, но цели каждой инициативы различны.

TCRA — это альянс нескольких корпораций для создания компьютеров с некоторым номинальным уровнем доверия. Примечательно, что многие из его спецификаций могут применяться и к платформам, отличным от ПК. В TCRA доверие заложено в защищенном аппаратном модуле, называемом TPM (Trusted Platform Module). TPM содержит функции, гарантирующие, что секреты, содержащиеся в модуле, никогда не будут раскрыты посредством программных атак. Он также содержит функции, такие как защищенные системные «измерения», которые пытаются передать доверие, содержащееся в TPM, на хост-машину.

Palladium, с другой стороны, представляет собой концепцию безопасности ПК-центрической системы, созданную только Microsoft. Одним из ее компонентов является модуль безопасности типа TPM, но Palladium также требует радикальных изменений в аппаратном чипсете и способе реализации портов ввода-вывода. Чипсет требуется для обеспечения политик безопасности памяти из всех потенциальных источников DMA, таких как графическая карта.

Palladium также требует шифрования ввода-вывода для подсистем клавиатуры и видео.

Требование Palladium о сотрудничестве между поставщиками чипсетов, OEM-производителями и Microsoft является потенциально крупным недостатком. Сегодня в отрасли производства оборудования для ПК недостаточно маржи, чтобы поддерживать накладные расходы на масштабную перестройку криптографической безопасности. Кроме того, многие поставщики чипсетов не имеют опыта внедрения защищенных систем. Помимо языкового барьера, с которым сталкиваются многие зарубежные поставщики чипсетов, чипсеты обычно разрабатываются в спешке и с пристальным вниманием к кошельку. Можно ли ожидать, что чипсеты, разработанные в таких условиях, будут запинять конфиденциальные секреты?

Xbox — пример того, что может пойти не так, когда политики безопасности определяются одним органом и реализуются другой, совершенно другой организацией. Microsoft написала спецификацию для надежного оборудования, а именно Xbox. В Xbox широко используются сильные криптографические алгоритмы, а главный ключ для системы заперт глубоко внутри сложного куска кремния. Однако опыт показал, что систему безопасности Xbox можно обойти, используя комбинацию незащищенного отладочного порта, недостатка в схеме инициализации оборудования и ошибки в обработке граничного случая указателя инструкций в ЦП. Эти три незначительных упущения, допущенные тремя независимыми сторонами (подрядчиком по сборке, разработчиком прошивки Xbox и Intel), в совокупности обеспечивают удобный метод обхода безопасности Xbox.

Каждый из этих недочетов сам по себе не представляет собой значительной проблемы безопасности. Это приводит к обескураживающему вопросу о том, сколько нарушений безопасности в конкретной реализации Palladium будет вызвано наложением нескольких безобидных недостатков. Каждая сложная система потребительской электроники имеет незначительные ошибки или

упущения в конструкции, особенно когда системы состоят из компонентов, созданных несколькими независимыми организациями, чей основной интерес заключается в получении прибыли.

В индустрии потребительской электроники можно либо поставлять идеальный продукт, либо зарабатывать деньги. Продукты, которые не приносят денег, быстро отменяются. Таким образом, очень редко можно найти потребительский продукт, который был бы технически совершенным во всех отношениях. В результате, единственный практический способ гарантировать безопасность такой сложной системы потребительской электроники, как Palladium, — это выбросить ее на волю и позволить хакерам делать с ней все, что они захотят, пока все большие дыры в безопасности не будут обнаружены и закрыты.

С другой стороны, TPM TCPA — это устройство, созданное для решения определенного набора проблем, который по масштабу меньше, чем у Palladium. Таким образом, TPM не так интересен с точки зрения рынка, но может быть более практичным и пригодным для использования по своему прямому назначению. И TPM, и Palladium уязвимы для аппаратных атак, но TPM не пытается распространять требования безопасности так далеко на территорию проектирования сторонних систем. TPM — это в первую очередь защищенный модуль управления ключами, который может обнаружить большинство изменений и вторжений в хост-систему. Программные слои, построенные поверх этого субстрата, выполняют остальную тяжелую работу, *caveat emptor*.

## Взлом доверенного ПК

Текущие предложения по доверенному ПК уязвимы для некоторых довольно простых аппаратных атак, даже при отсутствии какого-либо контроля интеграции или бэкдоров, связанных с ошибками.

Первая атака — это та, которую я называю атакой «*Surreptitious BIOS*» или SPIOS (произносится как «*Spy OS*»). SPIOS можно использовать для обхода политик DRM, которые полагаются на криптографически запечатанную функцию хранения доверенного ПК для предотвращения несанкционированного доступа пользователя к данным. Основная идея заключается в загрузке ПК с немодифицированным BIOS в доверенном режиме и извлечении всех нужных данных в системную оперативную память, а затем в выполнении горячего сброса системы с заменой образа BIOS.

Измененный образ BIOS может быть использован для считывания нужных данных из системной оперативной памяти. Нужные данные могут быть сеансовым ключом, сохраненным в памяти, или фактическими расшифрованными данными, в зависимости от того, как программа структурирует и кэширует свои данные в памяти. Поскольку текущие спецификации доверенных ПК требуют BIOS на основе шины LPC, для выполнения этой атаки можно использовать недорогие альтернативные

устройства прошивки (похожие на те, что используются на Xbox). Существуют методы, которые программисты приложений могут использовать для усложнения этой атаки, например, расшифровывать только один блок данных каждый раз в системную память, но многие из этих методов серьезно ухудшают производительность системы. Ухудшение производительности системы может быть особенно выраженным, если отключены кэширование файлов и предварительная выборка.

А другая атака — это атака, которую я называю «*Surreptitious RAM*» или SPAM-атакой. Цель этой атаки — подменить доверенные процедуры, отвечающие за измерение пригодности состояния системы. Устройство, такое как FPGA или ASIC, устанавливается на подключаемых картах памяти между чипами DRAM и разъемом памяти. Это устройство отслеживает шаблон проходящих адресов или может иметь дополнительный разъем, который отслеживает состояние проводов, подключенных к контактам ввода-вывода криптомуодуля, отвечающего за аутентификацию системы.

В любом случае, когда выполняется измерение системы, устройство SPAM представляет образ памяти, соответствующий неизмененному, доверенному состоянию системы. Однако во всех других режимах работы устройство SPAM представляет образ памяти, который изменяется для выполнения любых действий, которые пожелает пользователь. Эта модификация может быть очень тонкой: достаточно всего лишь пары битов, перевернутых в нужных местах, чтобы изменить ключевые инструкции ветвления в ядре безопасности.

Это устройство мощнее SPIOS, поскольку оно работает на системе, которая включена и предположительно заслуживает доверия. Его можно применять для эффективного обхода более широкого спектра схем DRM, а также некоторых аутентифицированных транзакций между локальной машиной и сервером. Однако спам сам по себе не может быть использован для ложной идентификации системы как другой зарегистрированной доверенной системы, поскольку спам не имеет секрета, общего между защищенным криптомуодулем с защитой от несанкционированного доступа на локальной машине и сервером аутентификации. Ложная идентификация системы потребует либо извлечения ключа из защищенного криптомуодуля с защитой от несанкционированного доступа (возможно, но не trivialно и, скорее всего, разрушительно для модуля), либо каким-то образом обмануть защищенный криптомуодуль с другой зарегистрированной доверенной машины, чтобы предоставить фальсифицированную идентификацию.

Устройство SPAM может быть изготовлено за относительно небольшую сумму (сегодня высокопроизводительные ПЛИС могут стоить всего 50 долларов в единичных партиях) и может быть очень простым в установке. SPAM может быть либо встроен непосредственно в модуль памяти (в этом случае он функционирует как устройство нарушения доверия и как устройство расширения памяти), либо может быть предоставлен как устройство, установленное в стековой конфигурации между слотом памяти материнской платы и существующим устройством памяти. В некоторых конфигурациях карт

памяти, особенно тех, которые используют тепловые экраны, может быть возможно скрыть устройство SPAM и выдать модуль за обычное устройство расширения памяти. Несмотря на сложность, это может быть стоящей атакой на крупную корпорацию или банк, которые хранят высокоценные секреты на доверенном сервере на базе ПК.

## С нетерпением жду

При рассмотрении перспективы доверенных вычислений нам необходимо сначала рассмотреть, предложат ли предлагаемые в настоящее время схемы все преимущества, которые они обещают, а затем сопоставить их с потенциальным вредом для прав потребителей и потенциальными выгодами для преступников (улучшенная конфиденциальность может использоваться как во благо, так и во зло). Если доверенные вычисления могут обеспечить идеальную безопасность для онлайн-бизнеса, то это может стоить потенциальных рисков. Однако сценарии, описанные в этой главе, указывают на то, что безопасность доверенного ПК может быть неидеальной.

Рассмотрим Xbox. Xbox — это надежная реализация ПК, которую можно взломать всего лишь с помощью беспаечного модуля за 50 долларов. Это накладывает довольно жесткие ограничения на ценность секретов, которые можно доверить Xbox. Аппаратные модчики настолько недороги, что они окупаются стоимостью скопированной игры или двух игр, если вы решите заплатить кому-то за установку чипа.

Конечно, всегда есть моральные и социальные последствия кражи контента, а также новое законодательство, такое как DMCA, которое отчасти направлено на то, чтобы сделать такие действия преступлением. К сожалению, текущие предложения по доверенным ПК на столе также слабы перед лицом столь же недорогих атак на оборудование. Таким образом, маловероятно, что они обеспечат уровень безопасности, необходимый для ценных или очень постыдных секретов.

Факт в том, что хакерская технология будет развиваться независимо от того, является ли она незаконной или нет, и является ли намерение добрым или злым. Таким образом, в интересах потребителей и компаний информировать население о хакерстве и чтобы каждый понимал ограничения своего «доверенного ПК». Худшим сценарием было бы, если бы миллиарды долларов были инвестированы в доверенные вычисления, давая только чистый результат, не повышающий безопасность или конфиденциальность для потребителей, при этом серьезно ограничивая права потребителей из-за плохой реализации политики в отношении контента.

Хорошей новостью для сторонников надежных ПК является то, что уменьшение размеров элементов в интегральных схемах приводит к большей интеграции во всем ПК. В течение десятилетия современный ПК будет помещаться на одном куске кремния. Как только ОЗУ и BIOS

будут полностью интегрированы в один кусок кремния, взлом системы станет намного сложнее — но не невозможным. Машины Focused Ion Beam (FIB), инструмент, используемый разработчиками микросхем и лабораториями анализа отказов, могут вырезать и перемыкать наномасштабные элементы. Еще одним преимуществом для разработчиков надежных машин является то, что процессоры с открытым ключом могут стать настолько маленькими и дешевыми для интеграции в чип, что отдельные чипы, особенно чипы памяти, смогут начать использовать надежную криптографию для аутентификации и шифрования своего ввода-вывода.

А другая технология, которая могла бы помочь внедрению доверенных ПК, — это интегрированные функции защиты от несанкционированного доступа или обнаружения несанкционированного доступа. Например, рефлектометр временной области (TDR) может быть встроен в ячейки ввода-вывода чипа. TDR может обнаружить присутствие подслушивающего устройства на проводе, распознавая определенные изменения в электрических свойствах провода. Помимо возможности обнаруживать подслушивающих устройств, интегрированные устройства TDR желательны для высокопроизводительного ввода-вывода, поскольку их можно использовать для калибровки импеданса привода и фильтров выравнивания/предыскания, необходимых для многогигабитной скорости связи.

## Заключительные мысли

Эта книга провела вас через краткий обзор взлома оборудования Xbox, от основ пайки и разборки до новейших проектов и методов. Она также познакомила вас с социальными аспектами взлома Xbox: людьми, которые занимаются взломом, и взаимодействием между обществом, законом и взломом.

Хотя подробности установки синего светодиода в Xbox могут стать неактуальными через несколько лет, навыки, которые вы приобретете, выполняя установку, останутся с вами на всю жизнь. Более того, социальные и правовые проблемы, с которыми сталкиваются хакеры и потребители, выйдут за рамки Xbox и затронут каждую часть нашего нового информационно-ориентированного образа жизни.

Материал в этой книге — всего лишь отправная точка; существует целый мир оборудования, ожидающий своего исследования. Я надеюсь, что эта книга дала начинающим читателям прочную отправную точку для того, чтобы стать исследователем, фиксатором и новатором в мире, который все больше заполняется электроникой, контролируется ею и зависит от нее.



## АПРИЛОЖЕНИЕ А

# Где взять хакерское снаряжение

Значительным психологическим барьером для начала работы по взлому оборудования является воспринимаемая недоступность инструментов взлома. Хотя Radio Shack имеет небольшой запас компонентов и инструментов, эти компоненты дороги, и большинство инструментов трудно использовать с современными миниатюрными компонентами. В этом приложении перечислены несколько поставщиков компонентов и инструментов, которые имеют солидный выбор по разумным ценам. В этом приложении также указаны номера заказов Jameco для основных инструментов, необходимых для выполнения проектов в этой книге. Вы можете ввести эти номера заказов непосредственно на защищенном веб-сайте Jameco для заказов, чтобы помочь вам быстро начать работу.

## Поставщики для любителей

Существует множество дистрибуторов компонентов и оборудования, от тех, которые обслуживают в первую очередь крупные предприятия, до тех, которые обслуживают частных лиц, любителей и ремонтников. Вы можете испытать некоторое разочарование, имея дело с дистрибутором, который обслуживает крупные предприятия. Эти дистрибуторы обслуживают крупные объемы заказов, поэтому заказы обслуживаются назначенным торговым представителем. В результате может быть сложно получить полный каталог деталей или информацию о ценах и инвентаре для небольших заказов. Поставщики, перечисленные в этом разделе, дружелюбны к индивидуальным заказам, и их коллективный инвентарь будет иметь в наличии большинство деталей, которые вам когда-либо понадобятся.

Имя поставщика	Контактная информация поставщика	Особенности и примечания
----------------	----------------------------------	--------------------------

---

## Взлом

### едение в обратную разработку

Цифровой ключ	<a href="http://www.digikey.com">www.digikey.com</a> 1-800-344-4539 (телефон) 1-218-681-3380 (факс)	Широкий выбор оригинальных компонентов. Дружелюбное к любителям быстрое обслуживание. Сбор за обработку \$5 для всех заказов стоимостью менее \$25. Превосходный веб-сайт с информацией о наличии товара в режиме реального времени и проверкой цен, техническими паспортами и инструментами параметрического поиска.
Хамеко	<a href="http://www.jameco.com">www.jameco.com</a> 1-800-831-4242 (телефон) 1-800-237-6948 (факс)	Экономичный поставщик, дружелюбный к любителям. Компоненты представляют собой смесь оригинальных и излишков. Хороший выбор инструментов по разумной цене. Комиссия за обработку \$5 для всех заказов менее \$20. Доступен широкий выбор наборов для самостоятельного изготовления.
МСМ Электроника	<a href="http://www.mcmelectronics.com">www.mcmelectronics.com</a> 1-800-543-4330 (телефон) 1-800-765-6960 (факс)	Специализированные инструменты и детали для обслуживания/ремонта бытовой электроники, а также обычные инструменты и компоненты. Продает труднодоступные биты безопасности и сменные детали. Фиксированная плата за обработку \$2,95, минимальный заказ для покупок через Интернет не требуется.
Мышелов	<a href="http://www.mouser.com">www.mouser.com</a> 1-800-346-6873 (телефон) 1-817-804-3899 (факс) orders@mouser.com	Поставщик инструментов и компонентов, который имеет в наличии хороший выбор оригинальных компонентов. Подходит для любителей, нет минимального заказа, нет платы за обработку.
Ньюарк	<a href="http://www.newark.com">www.newark.com</a> 1-800-463-9275 (телефон)	Более крупный традиционный дистрибутор компонентов. Широкий выбор компонентов, но не все компоненты есть на складе; возможно, придется подождать несколько недель, пока будут выполнены некоторые заказы. Хорошее место для покупки компонентов, которые вы не найдете у других дистрибуторов, дружественных любителям. Комиссия за обработку \$5 для всех заказов менее \$25.
Фрай Электроника	<a href="http://www.frys.com">www.frys.com</a>	Розничный магазин электроники, в котором имеется разумный выбор компонентов и инструментов. Отличное место, чтобы зайти и просмотреть, если вам не нравится шопинг по каталогам. Встречается только в Калифорнии, Техасе, Орегоне и Аризоне.
Макмастер Кэрр	<a href="http://www.mcmastercarr.com">www.mcmastercarr.com</a>	Поставщик механического оборудования, металлопроката и станков. Место, где можно получить необработанный листовой металл и крепеж для завершения проекта. Веб-сайт всеобъемлющий и полезный.
FindChips	<a href="http://www.findchips.com">www.findchips.com</a>	Автоматизированная поисковая система запчастей. Поиск деталей среди десятков дистрибуторов. Большинство поставщиков,

		дружественных любителям, перечислены в этой поисковой системе.
--	--	--

Таблица A-1: Таблица поставщиков комплектующих и оборудования.

Приложение А. Где взять хакерское снаряжение **209**

## Подготовленные формы заказа оборудования

В этом разделе содержится избранный список инструментов, продаваемых Jameco, которые вы можете заказать, чтобы подготовиться к взлому. «Базовая» форма заказа содержит все биты и драйверы, необходимые для открытия Xbox, а также некоторые основные инструменты для пайки. «Расширенная» форма заказа содержит набор инструментов и утилит, которые полезно иметь под рукой, но они достаточно дороги, поэтому те, у кого ограниченный бюджет, могут рассмотреть возможность заказа их только по мере необходимости.

Помимо инструментов, указанных в этих формах заказа, вам понадобятся несколько небольших деталей для проектов, описанных в Части I. Пожалуйста, обратитесь к введению каждой соответствующей главы, чтобы ознакомиться со списком деталей, которые вам понадобятся.

## Базовая форма заказа

В таблице ниже перечислены основные инструменты, которые вам понадобятся для выполнения проектов, описанных в этой книге. Вы можете заказать эти детали, перейдя на сайт Jameco, [www.jameco.com](http://www.jameco.com), и нажав на желтую кнопку «Быстрый заказ» с левой стороны. Введите номера деталей Jameco, указанные здесь, и начните взлом! (Цены могут отличаться.)

Описание	Хамеко Часть Число	Цена
Паяльник с переменной мощностью от 16 до 30 Вт	116572	18,95\$
Конический наконечник для паяльника 1/32"	35326	4,49\$
Небольшая упаковка флюсового сердечника припоя Kester 60/40	73576	2,59\$
2 унции флюса из канифольной пасты Kester	73584	1,59\$
Фитиль для распайки	175986	2,49\$
Очиститель и кондиционер для жала паяльника	132986	4,95\$
Набор инструментов SAE Hobby из 26 предметов	170069	14,95 \$
Инструмент для зачистки проводов, калибр 22-30	127870	4,95\$
Общая стоимость		54,96 долл. США

Набор инструментов SAE Hobby из 26 предметов включает все биты, драйверы и отвертки, необходимые для разборки Xbox, включая биты Tорx T-10, T-15 и T-20. Я включил конический наконечник 1/32" в форму заказа выше, потому что паяльник поставляется с ужасно большим

наконечник 1/8". Наконечник 1/8" полезен для спаивания небольших стран и создания коротких припоев по всем вашим мелкошаговым компонентам. Конические скосы и наконечники долота имеют сплющенную область, которая обеспечивает быструю передачу тепла, и поэтому их легче использовать, чем конические острые наконечники. Я также включил немного очистителя для жала паяльника и немного флюса из канифольной пасты. Они немного дорогие, но они сделают вашу жизнь намного проще, поверьте мне.

## Форма предварительного заказа

В таблице ниже перечислены несколько инструментов, которые было бы неплохо иметь под рукой, но они достаточно дорогие, поэтому приобретать их следует только в том случае, если вы планируете много заниматься хакерством.

- Подставка для паяльника и губка просто необходимы, если вы планируете заниматься даже умеренной пайкой. Подставка надежно убирает горячий наконечник паяльника, снижая риск ожогов и возгораний.\*Попробуйте паять с полузубчатым наконечником 1/32". Возможно, он вам понравится больше, чем конический скосенный наконечник.
- Антистатический браслет всегда полезно иметь под рукой, особенно если вы носите одежду и обувь, генерирующие статическое электричество.\*Комплект для снятия SMT-компонентов пригодится при попытке снятия небольших SMT-компонентов.
- Мультиметр — это отличный многоцелевой инструмент для тестирования, измерения и диагностики, который всегда под рукой, и его можно использовать везде, где угодно, дома или в общежитии. Доступно много моделей мультиметров; перечисленный здесь предлагает больше всего функций по самой низкой цене. Я его опробовал, и он хорошо работает.
- Инструмент для зачистки проводов и обжимной инструмент, перечисленные здесь, очень полезны для проекта по замене блока питания. Обжимной инструмент продается Digi-Key, а не Jameco. Обжимной инструмент немного дороже, но в отличие от инструментов стоимостью менее десяти долларов он оснащен формованными обжимными матрицами для более качественного обжима без пайки.

Описание	Номер детали	Цена
Подставка для паяльника	36329	4,25\$
Губка для подставки под паяльник	134631	2,99\$
Полузубчатый наконечник 1/32"	35078	4,49\$
Регулируемый антистатический ремешок	159257	\$7.95
Комплект для снятия SMT-компонентов	141305	21,95\$

---

Взлом

едение в обратную разработку

Мультиметр (вольты, омы, амперы, фарады, температура, частота, диод/транзистор)	177480	49,95 \$
Инструмент для зачистки проводов, калибр 16-26	127861	4,96\$
Обжимной инструмент (заказ в Digi-Key)	WM9999-ND	36,19 \$

## АПРИЛОЖЕНИЕ Б

# Методы пайки

В этом приложении объясняются основы пайки, а также некоторые более продвинутые методы поверхностного монтажа. Вы получите максимальную пользу от этого приложения, если будете экспериментировать с методами пайки по мере чтения. Jameco Electronics ([www.jameco.com](http://www.jameco.com)) продает широкий ассортимент проектных наборов, требующих сборки в сквозные отверстия, которые вы можете использовать для базовой практики пайки. Такие компании, как TopLine ([www.topline.tv](http://www.topline.tv)), предлагают экономичные практические наборы с использованием заготовок компонентов, если вы хотите попрактиковаться в технике сборки SMT, а MCM Electronics ([www.mcmelectronics.com](http://www.mcmelectronics.com)) предлагает практичный набор для практики поверхностного монтажа. (Я настоятельно рекомендую вам попрактиковаться в пайке SMT, прежде чем пытаться прикрепить компонент SMT, который вам нужен.)

## Введение в пайку

Базовая техника пайки довольно проста: вставьте кончик паяльника в пространство между выводом компонента и контактной площадкой печатной платы, чтобы нагреть обе части. Как только они достаточно нагреются, постепенно вводите немного припоя в соединение, пока не образуется гладкая, ровная галтель. На самом деле, пайка требует некоторой практики и опыта, прежде чем вы сможете спаять типичную плату с более чем тысячей соединений без каких-либо плохих паяных соединений.

Для создания хорошего паяного соединения горячий жидкий припой должен «смочить» детали, чтобы соединение было выполнено. Таким образом, настоящее искусство ручной пайки заключается в понимании того, как гарантировать смачивание припоеем.

---

## Взлом

### едение в обратную разработку

Вы можете определить, когда жидкий припой смачивает кусок металла, посмотрев на него. Смоченное соединение выглядит так, как будто расплавленный припой потерял все поверхностное натяжение; жидкий припой блестит и плавно течет по рабочей области. В противоположной ситуации припой имеет тусклый блеск и имеет тенденцию собираться в шарик вокруг наконечника паяльника вместо того, чтобы вытекать наружу.

## Использовать флюс

Припой не смачивает металл, потому что кислород в воздухе или грязь и жир вступили в реакцию с металлами. В этом случае вы можете нанести флюс на заготовку, чтобы разрушить эти инородные соединения. Слово «флюс» происходит от латинского *fluxus*, что означает текучий. Большинство припоев имеют встроенный сердечник из флюса для улучшения паяемости. Если вы внимательно посмотрите на отрезанный кусок припоя, вы увидите сердечник из флюса, окруженный сплавом припоя. Всегда используйте припой с сердечником из флюса, иначе вы будете в мире мучений, пытаясь смочить припой. Почти все припои, предназначенные специально для электроники, имеют сердечник из флюса, но есть припои, которые вы можете случайно купить в хозяйственном магазине, в которых нет флюса, и которые предназначены, например, для соединения труб. Когда вы нагреваете припой с сердечником из флюса, поднимется небольшое облако испаренного дыма флюса. Небольшой вентилятор, размещенный рядом с рабочей зоной, выдует пары и предотвратит вдыхание.

Распространенная ошибка новичков — слишком большая вера в припой с флюсовым сердечником. Часто флюса, содержащегося в припое с флюсовым сердечником, недостаточно, чтобы смочить припой. В этом случае вам нужно будет нанести дополнительный флюс. Сырой флюс обычно поставляется в виде жидкости или пасты, поэтому наносить его легко. В жидкой форме одну каплю флюса можно нанести с помощью половины зубочистки. Разломайте зубочистку пополам, оставив излом слегка зазубренным. Окуните сломанный конец зубочистки во флюс, и небольшая капля прилипнет к концу. Наносить жидкий флюс на большую площадь можно с помощью тонкой художественной кисти, но обязательно очистите кисть, когда закончите, иначе она станет липкой и непригодной для использования через пару дней. Пипетка для флюса также удобна, но дорога. Пипетка для флюса — это флакон с тонкой капиллярной иглой сверху; когда вы переворачиваете флакон, флюс медленно капает из капилляра. В виде пасты флюс можно наносить, окуная любой кусочек, например зубочистку или кусок твердой проволоки, в пасту флюса. Наконец, флюсовые ручки удобны для новичков, поскольку они объединяют хранение и дозирование флюса в удобной и недорогой упаковке. Флюсовые ручки не обладают точностью или качеством других методов нанесения флюса, но они удобны и хороши для периодического использования.

Многие флюсы требуют очистки после использования. Флюсы могут со временем затвердевать, что затрудняет ремонт в будущем, они могут медленно

разъедать плату, и они могут впитывать воду и становиться проводящими. Традиционный паяльный флюс — это флюс на основе смолы. Флюсы на основе смолы требуют использования сильных растворителей, которые легко воспламеняются и токсичны. В результате я склонен рекомендовать водорастворимые флюсы или флюсы без отмычки. Водорастворимые флюсы можно удалить, просто промыв плату водой. Лучше использовать дистиллированную депонизированную воду, но я обнаружил, что большая часть теплой водопроводной воды также довольно эффективна. Когда мне нужно очистить большую партию плат, я бросаю их в посудомоечную машину (с очищенной ловушкой для еды и без моющего средства!). После того, как платы вымыты, положите их

---

## Взлом Xbox: Введение в обратную разработку

на токопроводящем чистом противнике или алюминиевой фольге и поставьте их в духовку на слабый огонь (около 200°F) и выпекайте около часа или двух, или пока вся вода не испарится. Не включайте духовку слишком сильно, иначе вода, попавшая в поры компонентов, превратится в пар и приведет к их растрескиванию или взрыву. Большинство деталей спроектированы так, чтобы быть «герметичными в процессе», поэтому их можно мыть водой. Однако будьте осторожны с разъемами и переключателями; вам может потребоваться заклеить их куском каптоновой ленты, чтобы предотвратить попадание воды.

### Предупреждение



Не используйте кислотные флюсы на печатных платах. Они будут воздействовать на плату и компоненты и со временем приведут к отказам. Кислотные флюсы часто досаждают новичкам, которые используют припой и флюсы, предназначенные для пайки труб.

## Советы для начинающих

Компоненты с большим количеством выводов необходимо выровнять и прикрепить на месте перед пайкой. Если компонент поверхностного монтажа, его можно закрепить на месте, припаяв два штырька на противоположных углах устройства. После присоединения компонента на месте еще раз проверьте выравнивание на случай, если компонент сместился во время присоединения. Если компонент сквозного типа, вам понадобится липкая лента, чтобы удерживать его на месте, пока вы переворачиваете плату. Прикрепите угловые выводы компонента на месте и проверьте, что он выровнен по отношению к плате, прежде чем припаивать все выводы. (Лента для маскировки имеет небольшой люфт, и часто вам придется нагревать один из приклеенных углов, одновременно нажимая на компонент, чтобы выровнять компонент по отношению к плате.)

Каптоновая лента — довольно удобная вещь для хранения на рабочем столе. Изготовленный компанией DuPont, каптон выдерживает температуру до 500°F, что значительно выше точки плавления припоя. Он удобен для маскировки близлежащих областей, где вы не хотите использовать припой. Однако каптоновая лента стоит дорого, поэтому используйте ее только в ситуациях, когда она будет контактировать с горячим припоеем.

Припой не смачивает слишком холодный металл. Это распространенная проблема при пайке больших соединений или при пайке соединений, прикрепленных к большим листам меди. В этих случаях соединенный металл отводит достаточно тепла, так что

соединение никогда не достигает точки плавления припоя. Решением этой проблемы является либо использование более мощного паяльника (но будьте осторожны — тепло очень больших паяльников также может привести к отпадению дорожек платы), либо оставление паяльника в контакте с соединением на более длительный период времени перед нанесением припоя. Один из приемов для более быстрого нагрева рабочей области — подать лишь немного припоя на наконечник паяльника, где он нагревает детали. Даже если припой не смачивает плату, жидкий припой на наконечнике паяльника увеличивает эффективную площадь контакта между паяльником и платой, и тепло будет передаваться в плату быстрее. После нанесения припоя на наконечник паяльника вам иногда придется вращать паяльник, поддерживая контакт в стыке, чтобы расплавленный припой соприкоснулся с платой.

Вы можете определить, когда вывод компонента или контактная площадка платы достаточно горячие, внимательно наблюдая за тем, как свет отражается от них. Выводы компонентов и контактные площадки на печатной плате обычно имеют на себе какое-то покрытие, обычно сделанное из припоя. Это припойное покрытие имеет слегка матовый блеск при нормальных условиях. Однако, когда оно достаточно горячее, блеск меняется от матового до почти идеально отражающего. Чтобы почувствовать, как это выглядит, попробуйте нагреть большую квадратную контактную площадку паяльником, используя описанную выше технику. Обычно вы можете наблюдать, как фронт плавления распространяется по контактной площадке, пока паяльник нагревает плату.

#### **Предупреждение**



Если вы используете паяльник без контроля температуры, используйте паяльник с минимальной мощностью, которую вы можете использовать для выполнения работы. Это поможет предотвратить повреждение платы, так как избыточное тепло может привести к отслоению медных дорожек от платы.

## **Пайка поверхностного монтажа**

Освоение навыков поверхностной пайки требует немного терпения, практики и хороших инструментов. Основные инструменты, которые требуются помимо основного набора для пайки, — это пинцет и увеличительная линза.

---

## Взлом Xbox: Введение в обратную разработку

Хорошая пара тонких пинцетов является необходимым аксессуаром для пайки компонентов поверхностного монтажа. Пинцеты необходимы для безопасного обращения с компонентами поверхностного монтажа во время пайки, так как небольшие компоненты быстро нагреваются и становятся достаточно горячими, чтобы обжечь палец. Пинцеты также необходимы для удержания небольших компонентов на месте, предотвращая их перемещение поверхностным натяжением жидкого припоя по мере его плавления и охлаждения. Кончик пинцета должен быть достаточно маленьким, чтобы поместиться между штырьками самой тонкой детали поверхностного монтажа, с которой вы собираетесь работать. Таким образом, вы можете использовать пинцет для манипулирования отдельными штырьками во время пайки и проверки.

Существует множество классов пинцетов. Классификация основана на остроте, качестве и долговечности наконечников, выравнивании наконечников и пружинном действии пинцета. Высококачественные пинцеты немного дорогие, но они стоят своих денег, если вы собираетесь много паять поверхностного монтажа. Дистрибуторы, которые сосредоточены на поставках для производства, такие как Future-Active

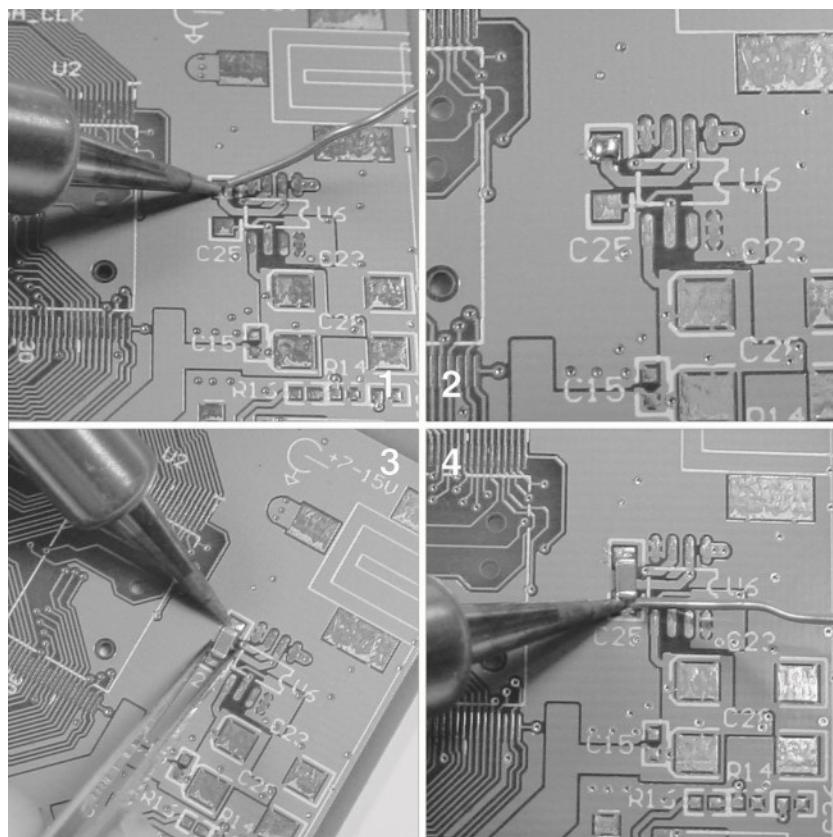
Электроника ([www.future-active.com](http://www.future-active.com)), продает разумный выбор хороших пинцетов.

Самая большая проблема в поверхностной пайке — это возможность видеть то, над чем вы работаете. Ваши руки способны легко и многократно манипулировать объектами, меньшими, чем может увидеть невооруженный глаз. Идеальным увеличительным решением для пайки является оптический стереоскоп, вроде тех, что используются для осмотра биологических образцов. К сожалению, эти микроскопы очень дороги, лучшие модели продаются примерно по цене подержанного автомобиля.

Более экономичным решением является использование настольной увеличительной линзы. Многие магазины чертежных/художественных принадлежностей продают такие линзы, а большинство магазинов канцелярских товаров продают как минимум одну модель лампы со встроенной увеличительной линзой. Эти увеличительные линзы помогут при сборке, но им не хватает мощности для тщательного осмотра вашей паяльной работы. Для осмотра готовых паяных соединений следует использовать относительно недорогую мощную портативную лупу, например ювелирную лупу или проверочную лупу.

## Методика для простых компонентов

Простые компоненты поверхностного монтажа, такие как резисторы, конденсаторы, индукторы и небольшие полупроводниковые приборы, такие как транзисторы и диоды, легко монтируются на печатной плате. Давайте рассмотрим технику монтажа этих компонентов.



**Рисунок Б-1:** (1) Нанесите каплю припоя на одну из контактных площадок целевого компонента. В данном случае целевым компонентом является С25. (2) Фотография капли припоя после нанесения. (3) Используйте пинцет для выравнивания целевого компонента, а затем нагревайте паяльником, пока начальная капля припоя не растечется по выводам компонента. (4) Припаяйте оставшиеся контактные площадки компонента.

Сначала поместите каплю припоя на одну из контактных площадок компонента, затем поместите и выровняйте компонент по его контактным площадкам с помощью пинцета. Как только вы почувствуете себя комфортно с размещением компонента, нагрейте

---

### Взлом Xbox: Введение в обратную разработку

каплю припоя, пока она не расплывется и компонент не встанет на место на своих контактных площадках.

Продолжайте подавать тепло, одновременно регулируя выравнивание компонента.

Уберите нагрев и подождите, пока припой остынет и затвердеет. Дважды проверьте выравнивание компонента, а затем припаяйте остальные контакты компонента к плате. Если первое паяное соединение выглядит тусклым или слабым, повторно нагрейте его с небольшим количеством припоя после того, как все остальные контакты будут припаяны (см. рисунок В-1). Этот метод работает для компонентов с двумя или тремя контактами, а также для многоконтактных компонентов с широким шагом, таких как 50-миллиметровые микросхемы в корпусе.

Самая сложная ситуация, с которой вы можете столкнуться при пайке этих компонентов, возникает, когда один или несколько выводов компонента прикреплены к большой площади меди, например, к силовой плоскости. В этом случае необходимо приложить большое количество тепла к контактной площадке печатной платы, прежде чем припой смочит и прилипнет к печатной плате. Будьте осторожны, когда это происходит, так как припой окисляется и скатывается, и создает плохой электрический контакт с платой. Добавьте немного флюса для пайки, если возникла такая ситуация, чтобы улучшить смачивающее действие расплавленного припоя.

#### Совет



«Сухое» жало паяльника будет испытывать трудности с нагревом больших участков меди или контактных площадок, которые подключены к силовым плоскостям.

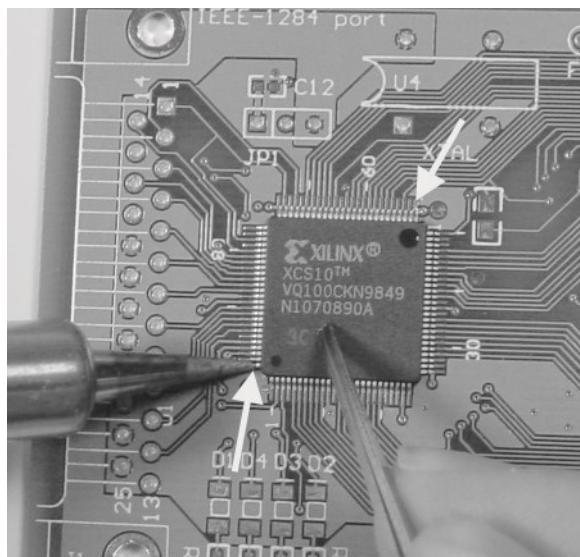
Один из способов увеличить скорость передачи тепла от паяльника — начать паять с каплей расплавленного припоя на жале и коснуться паяльником платы через каплю расплавленного припоя. Эта капля распространится и нагреет плату локально, а также установит хорошее тепловое соединение, которое более эффективно передает тепло. Это в конечном итоге приведет к лучшему паяному соединению после добавления большего количества припоя.

## Методика для сложных компонентов

Большинство интегрированных полупроводниковых компонентов сегодня доступны в каком-либо мелкотиражном корпусе для поверхностного монтажа. Такой тип корпуса может сначала пугать, но немного практики с несколькими простыми приемами и советами — это все, что нужно, чтобы прикрепить эти детали с высокой степенью уверенности. Процесс присоединения мелкотиражного компонента для поверхностного монтажа очень похож на процесс

присоединения простого компонента для поверхностного монтажа, и он должен занять не более нескольких минут для типичного корпуса для поверхностного монтажа среднего размера.

Первый шаг — прикрепить компонент к печатной плате, припаяв только два угловых штырька на чипе. Если выравнивание не получилось правильным с первой попытки, это легко исправить, просто нагрев один из двух углов, одновременно надавливая на чип в нужном направлении для выравнивания (см. рисунок В-2). Будьте очень внимательны к точности этого выравнивания: невыровненный чип приведет к бесконечному разочарованию при попытке припаять чип.



**Рисунок Б-2:** Сначала припаяйте по одному штырьку на каждом из двух противоположных углов чипа. Стрелки на этой схеме указывают на углы, используемые в этом примере.

Следующий шаг — нанести тонкую пленку припоя вокруг чипа. Это усилит смачивающее действие припоя на штырьки чипа и заставит припой растекаться по контактным площадкам компонентов, а не между выводами, что может вызвать короткое замыкание (см. рисунок В-3).

#### Совет



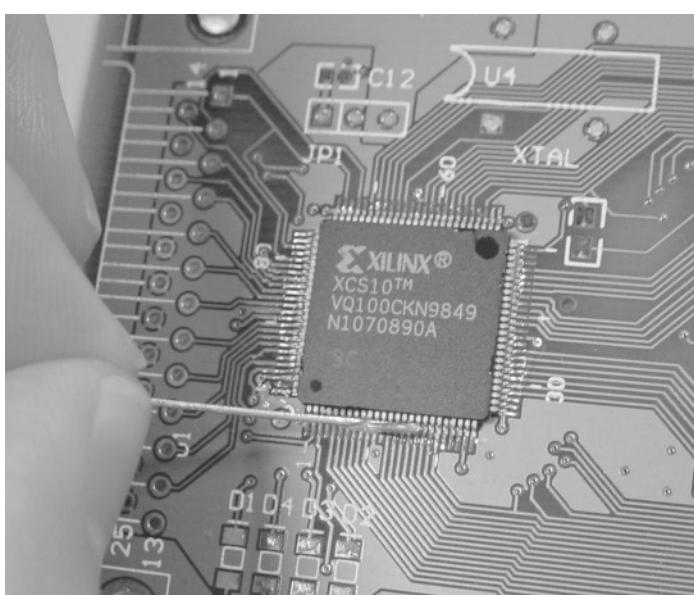
При компоновке платы используйте сверхдлинные площадки для мелкошаговых компонентов поверхностного монтажа. Дополнительная длина облегчит ручную пайку, хотя также немножко затруднит трассировку

---

### Взлом Xbox: Введение в обратную разработку

**и немножко увеличит размер платы. Дополнительная длина площадки отводит излишки припоя с выводов чипа, тем самым уменьшая вероятность образования перемычек припоя.**

После того, как все выводы будут равномерно покрыты флюсом для пайки, наденьте на кончик паяльника маленький шарик припоя и прижмите этот шарик к нераспаянным выводам. Шарик припоя затечет в пространство под выводами компонентов и вокруг них. Повторяйте этот процесс, пока все выводы не будут покрыты припоеем. Не беспокойтесь на этом этапе, если излишки припоя перекроют несколько контактов. После того, как вы закончите пайку всех контактов, используйте медную оплетку для распайки (припой-фитиль), чтобы удалить все припойные перемычки, как показано на рисунке B-4.



**Рисунок Б-3:** Нанесите тонкую пленку флюса для пайки на контакты вокруг чипа. На этой иллюстрации флюсовая паста наносится с помощью обрезка проволоки, который часто окунают в контейнер с флюсом.

Вы почти закончили. Наконец, убедитесь, что все штифты надежно припаяны. Трудно провести эту проверку визуально без микроскопа. Вместо этого потяните штифты иглой или кончиками пинцета, протаскивая пинцет вдоль штифтов, как показано на рисунке B-5, используя твердое, равномерное движение. Штифты, которые не припаяны на место, будут слегка двигаться. Если это так,

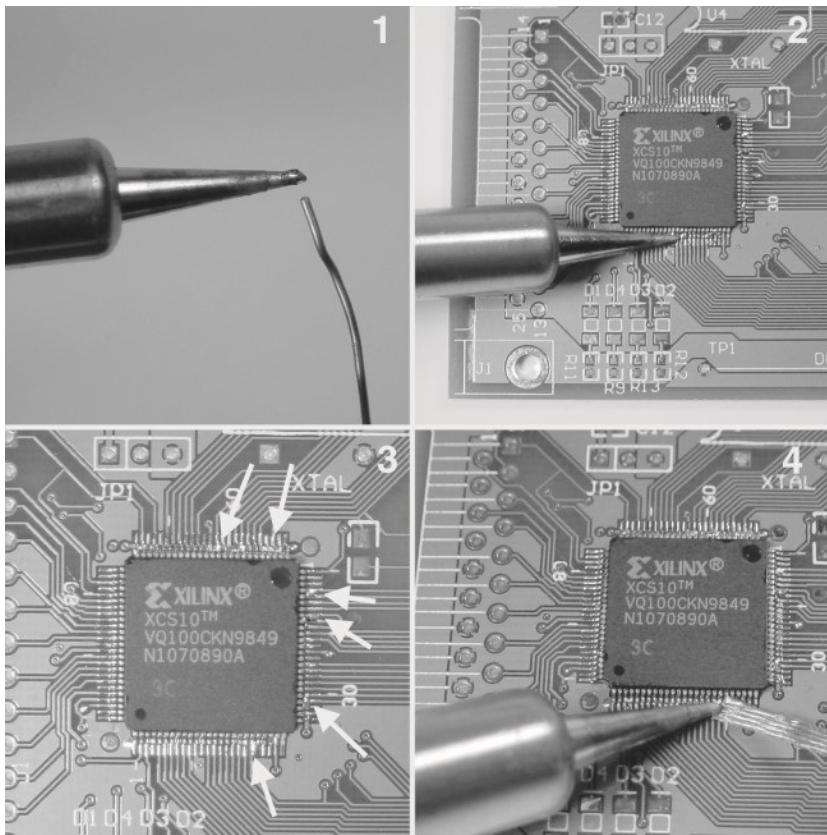
отремонтируйте их, вставив обратно на место и нанеся на штифт немного припоя.

### Осторожность



Всегда проверяйте наличие коротких замыканий после пайки печатной платы. Некоторые припойные мостики могут быть микроскопическими дендритами, а некоторые припойные мостики образуются за штырьками под чипом. Проверьте наличие коротких замыканий с помощью мультиметра, настроенного на измерение сопротивления. Не используйте режим звуковой непрерывности, так как большие высокоскоростные компьютерные платы обычно имеют низкое (несколько Ом) сопротивление между шинами питания, что достаточно низко для регистрации короткого замыкания на многих измерителях непрерывности. Кроме того, при измерении сопротивления между линиями питания подождите секунду или две, чтобы измерение стабилизировалось. Начальное сопротивление между шинами питания будет очень низким, пока большие конденсаторы, которые находятся на линиях питания, заряжаются.

После того, как ваш компонент будет припаян, очистите излишки флюса с помощью мягкого растворителя, например изопропилового спирта, и ватного тамponsа, как показано на рисунке В-6. Очистка важна, поскольку она делает визуальный



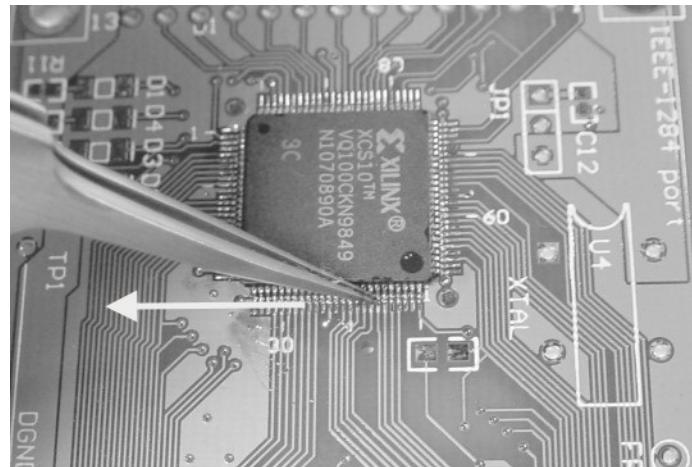
**Рисунок Б-4:** (1) расплавьте небольшое количество припоя на кончике горячего паяльника.  
 (2) Перенесите этот припой на контакты компонента. Повторяйте шаги (1) и (2), пока все контакты не будут спаяны. (3) Перемычки припоя, указанные стрелками, сформируются между многими контактами. (4) Удалите перемычки припоя с помощью куска медной оплетки для распайки.

осмотр становится проще, а также облегчается зондирование выводов чипа во время отладки. Это также важно, поскольку многие флюсы имеют тенденцию покрываться коркой и захватывать загрязняющие вещества со временем, что затрудняет будущий ремонт.

## Техника снятия компонентов

Существует множество методов удаления компонентов поверхностного монтажа, и многие из них требуют специальных инструментов, таких как паяльники-щипцы или термофены.

Паяльники-щипцы хороши для удаления небольших компонентов поверхностного монтажа, особенно на платах, которые плотно упакованы. Они довольно быстры и эффективны, и правильный паяльник-щипцы оставит контакты компонента в относительно хорошем состоянии. Однако они также довольно дороги, и требуется немного практики, чтобы выяснить правильное время и давление, которое следует использовать при удалении чипа.



**Рисунок Б-5:** Проведите штырьками вдоль стороны чипа с помощью пинцета или иглы, прилагая твердое, равномерное усилие. Штырьки, которые плохо припаяны, слегка согнутся. Стрелка указывает направление движения.



**Рисунок Б-6:** Удалите остатки флюса ватным тампоном, смоченным в слабом растворителе.

---

## Взлом Xbox: Введение в обратную разработку

Более простым и дешевым решением является использование термофена, например, тех, что продаются в хозяйственных магазинах для удаления линолеумных плиток. Термофен нагреет всю область платы, и компоненты оторвутся от своих площадок с минимальным повреждением выводов. Недостатком этого метода является то, что он не очень точен, поэтому это не идеальное решение для удаления чипов с плат, которые будут использоваться снова. В результате термофены наиболее эффективны для спасения хороших чипов со сломанных плат. Другая проблема с использованием термофенов заключается в том, что тепло может деформировать платы или вызывать разрушительные отказы чипов, которые впитали влагу в свои корпуса. Этот режим отказа, называемый «попкорн», происходит, когда влага, попавшая внутрь чипа, кипит, но не может выйти, вызывая нарастание давления, которое завершается разрушительным событием выброса. Чипы и платы, работающие во влажной среде или промывавшиеся в воде, следует запекать в духовке при температуре около 200–250 °F в течение нескольких часов перед отпайкой термофоном.

Другой вариант, который особенно привлекателен для любителей, — повторное легирование паяного соединения, чтобы оно плавилось при очень низкой температуре, ниже 300°F. Компания Chip Quik ([www.chipquikinc.com](http://www.chipquikinc.com)) владеет патентом на эту технологию, а наборы для удаления можно приобрести у многих поставщиков, включая Jameco (номер заказа 141305). Эта технология привлекательна тем, что не требует специального оборудования, проста и довольно безопасна, как показано на рисунке B-7.

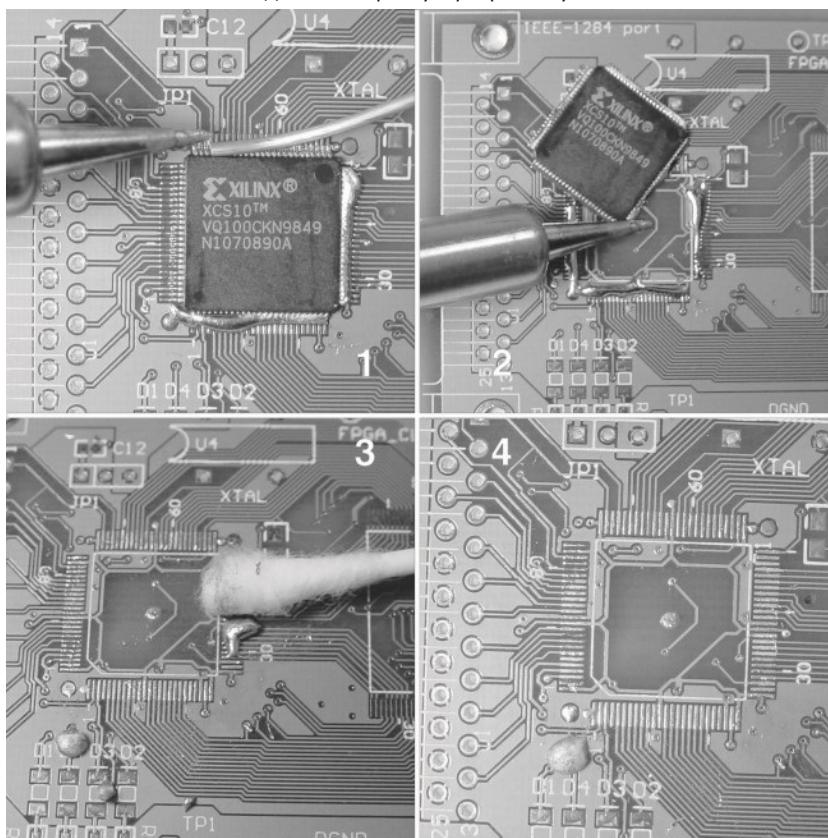
Для этого сначала покройте выводы чипа паяльным флюсом, затем расплавьте сплав для удаления чипа на выводах компонента, который нужно удалить. Это создаст большой валик сплава с низкой температурой плавления по всему чипу. Затем нагрейте весь валик, проведя кончиком паяльника через валик. Сохраненное в чипе и сплаве тепло будет поддерживать его расплавленным достаточно долго, чтобы весь чип легко соскользнул с контактных площадок. Наконец, очистите сплав с низкой температурой плавления, нагрев его паяльником, а затем вытерев его ватным тампоном. Сплав довольно легко сотрется как с платы, так и с чипа.

Этот метод повторного сплавления сохраняет целостность выводов удаленного чипа, а также контактных площадок на печатной плате. Недостатком этого метода является то, что очистка сплава может быть немного грязной, и крошечные частицы сплава могут попасть между выводами соседних чипов. (Осторожность при очистке предотвратит эту проблему.) Другим недостатком является то, что удаление чипов расходует удаляющий сплав, поэтому удаление большого количества чипов с помощью этого сплава может стать дорогим в долгосрочной

перспективе. К счастью, удаление чипов — довольно редкое явление для большинства любителей.

Корпус «BGA» (Ball Grid Array) — это тип корпуса для поверхностного монтажа, который становится все более популярным из-за его высоких электрических характеристик и плотности. Эти корпуса крепятся к плате с помощью большого массива шариков припоя под корпусом. Очевидно, что такой корпус нельзя паять с помощью обычного паяльника. Кроме того, корпуса BGA очень трудно осматривать, поскольку большинство паяных соединений находятся глубоко под корпусом. Единственный приемлемый метод крепления этих компонентов — использование печи, которая нагревает всю плату и компонент до точки, в которой все шарики припоя расплавляются. Обычно выравнивание этих корпусов BGA выполняется с помощью какой-либо машины, а осмотр этих корпусов BGA выполняется либо с помощью рентгеновских лучей, либо с помощью ультразвуковой визуализации.

Оборудование, необходимое для установки таких корпусов, очень дорогое, поэтому для любителя лучший способ установить устройство в корпусе BGA — это заплатить профессиональной сборочной фирме, чтобы она сделала это. Такие компании, как Naprotek ([www.naprotek.com](http://www.naprotek.com)), которые специализируются на быстром прототипировании в малых объемах, установят устройство в корпусе BGA за довольно разумную плату. При установке BGA всегда просите копию фотографии осмотра детали; дешевое спокойствие, которое приносит фотография, стоит дополнительных затрат. Детали BGA также можно снять и переделать, хотя этот процесс может быть намного дороже, чем процесс присоединения.



**Рисунок Б-7:** Удаление чипа путем повторного сплавления паяного соединения. (Перед началом этого процесса подготовьте выводы чипа с помощью паяльного флюса.) (1) Расплавьте повторно легирующий состав на всех выводах чипа. Повторно легирующий состав поставляется в форме проволоки. (2) Нагрейте капли металла вокруг чипа, проведя паяльником по металлу. Весь повторно легированный припой расплавится, и в этот момент чип можно будет сдвинуть с его контактных площадок. (3) Очистите повторно легирующий состав с платы и чипа, нагрев его и вытерев ватным тампоном. (4) Результат — чистое удаление чипа.

## АПРИЛОЖЕНИЕ С

# Знакомство с разводкой печатной платы

Это приложение познакомит вас с процессом проектирования и компоновки печатной платы, а также даст обзор доступных инструментов, которые могут вам понадобиться для достижения ваших целей в области проектирования, представив продукты и методы для тех, у кого ограниченный бюджет. В заключение я представлю несколько простых проектов, которые помогут вам начать свои приключения с оборудованием.

## Философия и дизайн-поток

Печатные платы — это холст хакера оборудования, а инструменты САПР — это кисти. Как и в любой инженерной или художественной дисциплине, для создания мастерства в проектировании и компоновке печатных плат требуется практика. К счастью, появление недорогих услуг по прототипированию печатных плат и бесплатных (или почти бесплатных) инструментов САПР сделало проектирование и компоновку печатных плат недорогим и доступным хобби.

Проектирование и компоновка печатных плат — это две тесно связанные задачи. Компромиссы идут на компромиссы на протяжении всех этапов проектирования и компоновки. Иногда компонент может не подойти или часть может быть недоступна, и вам придется изменить схему, чтобы учесть этот недостаток. В других случаях вы можете внести высокоуровневое изменение в проект или поймать ошибку, и печатную плату придется обновить, чтобы отразить эти изменения. По моему

опыту, ключ к быстрому созданию успешного проекта печатной платы — быть гибким на всех фронтах проектирования.

## Усовершенствование вашей идеи

Процесс проектирования печатной платы всегда начинается с вашей идеи. Первое, что вам нужно сделать, это получить очень четкое представление о том, что вы пытаетесь построить. Чем больше деталей у вас будет на старте, тем проще вам будет процесс проектирования. Вы должны иметь представление о том, насколько большой должна быть конечная плата, сколько она должна стоить и, конечно, что она должна делать. Я всегда нахожу полезным рисовать эскизы и, в случае крупных проектов, писать проектную документацию, которая помогает мне организовывать и записывать мои мысли.

Для первых нескольких проектов одним из самых сложных шагов будет закрепление идеи, потому что вы не будете знать, какие типы компонентов доступны для реализации вашей идеи, и какие типы ограничений реального мира вам придется проектировать. Лучший способ начать — найти существующую идею, которая очень похожа на вашу, и смоделировать свою идею по ней. (Многие производители микросхем предлагают бесплатные заметки по применению и образцы дизайна, которые образуют прекрасную отправную точку.) Другой способ определить свои идеи — учиться на существующих продуктах: если вы хотите сделать будильник, разберите существующие часы, чтобы посмотреть, как они были сделаны.

## Схематический захват

Как только у вас появится идея, вам нужно создать принципиальную схему. Схема — это символическое представление вашей идеи, выраженное в виде набора символов деталей и виртуальных проводов.

Большинство программ для захвата схем поставляются с библиотекой деталей, что помогает ускорить процесс захвата схем. Однако, если вы не найдете нужную деталь в библиотеке, вам придется создать свой собственный символ схемы. Все символы схемы связаны с посадочным местом компонента печатной платы, которое представляет собой рисунок меди на печатной плате, который сопрягается с компонентом.

Один из распространенных источников ошибок возникает из-за того, что не проверяется связь между символом и посадочным местом: 16-контактный DIP-отпечаток не подходит к 16-контактному разъему для поверхностного монтажа, и большинство инструментов проектирования не могут отличить их друг от друга. Поэтому убедитесь, что все назначения посадочных мест верны, и дважды и трижды проверьте схематический символ. В частности, всегда проверяйте и перепроверяйте контакты питания, так как они могут вызывать самые сложные и

разрушительные виды ошибок. Подумайте о том, чтобы попросить друга дважды проверить символы, чтобы избежать повторных ошибок или позволить ошибкам пройти неосознанно. (Такое количество избыточности может показаться глупым для простых деталей, но оно становится абсолютно необходимым для деталей с сотнями и тысячами контактов, где ваш мозг превращается в каплю на полпути процесса проверки.)

Внимание к деталям с самого начала является самым важным навыком для захвата схемы и может избавить вас от головной боли от раздражающих ошибок в дальнейшем. Каждый вывод на каждом компоненте там по какой-то причине, и если какой-либо вывод остается неподключенным, вы должны понимать, почему это нормально или нет. Для этого прочтите листы технических данных продукта, включая каждую страницу и сноску. Не игнорируйте мелкий шрифт, который требует подтягивающего резистора для установки условий запуска или конденсатора для фильтрации шума или стабилизации системы, иначе вы, скорее всего, получите еще больше раздражающих ошибок.

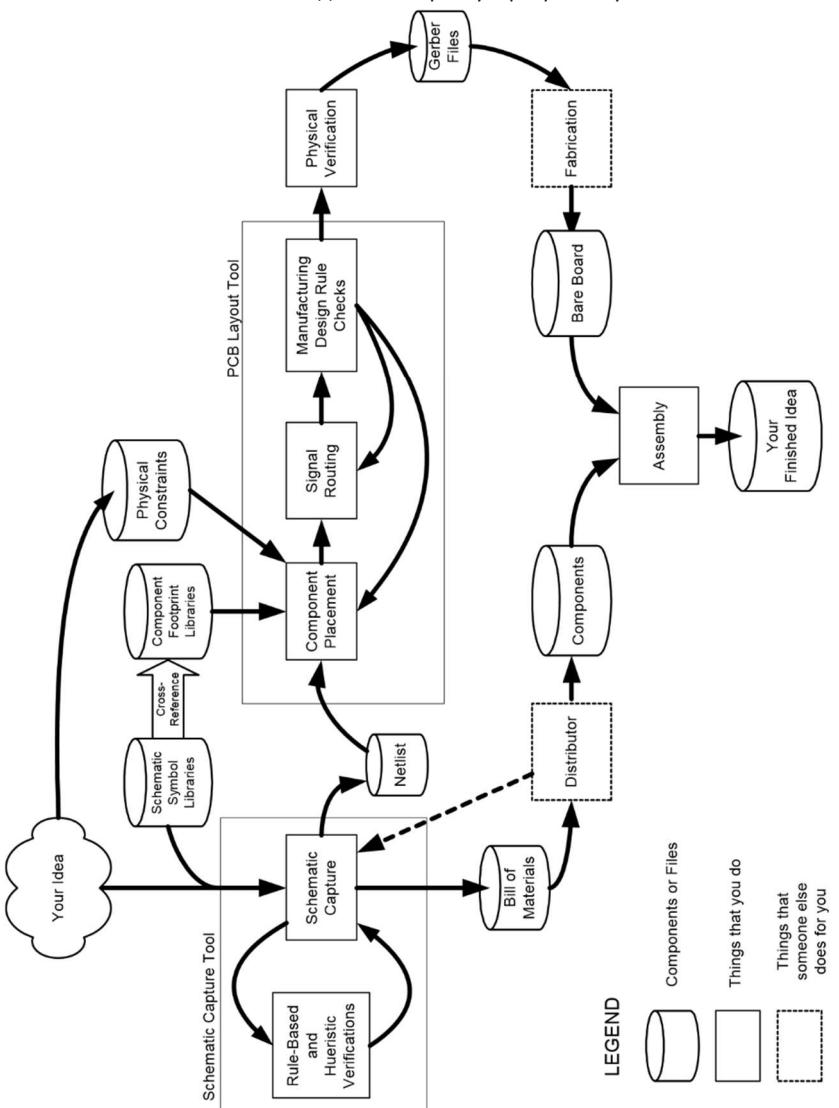


Figure C-1: The board design process, from idea to finished product.

Проверки правил проектирования полезны для поиска некоторых ошибок в схемах, но проверки, которые они выполняют, обычно довольно элементарные, и поэтому обнаруживают только самые грубые ошибки. Типичный набор проверок может включать в себя обнаружение дублирующихся обозначений деталей, оборванных сетей и плавающих входов.

Помимо обычных средств проверки правил проектирования вы также можете разработать свои собственные простые проверки. Например, многие инструменты проектирования не выполняют «проверки четности списка соединений», которые можно довольно легко генерировать с помощью скрипта Perl или программы электронных таблиц. Проверки четности списка

соединений — это эвристические проверки, которые подсчитывают количество компонентов, подключенных к каждому списку соединений, а затем сортируют подсчет по количеству соединений. «Одноконтактные сети» будут в верхней части отсортированного списка. Одноконтактная сеть почти всегда является признаком типографской ошибки (неправильное написание имени списка соединений) во время захвата схемы. Может быть полезно кратко просмотреть весь отсортированный список, чтобы увидеть, что все группы сигналов имеют одинаковое количество соединений. Большинство шин сигналов имеют одинаковое количество соединений с каждым сигналом вшине. Вы можете создать инструмент, который позволяет проверять подсчет соединений списка соединений для вашей программы захвата схемы, за один день, и вы, несомненно, сэкономите себе бесчисленные часы усилий и денег, обнаружив ошибки, которые он выявит.

Перед экспортом схемы в инструмент компоновки платы следует заказать все детали в спецификации материалов схемы. Часто критические детали будут в дефиците, и схему придется перепроектировать, чтобы учесть нехватку. Часто изменение назначения посадочного места компонента является единственным необходимым изменением. Изменение схемы для учета нехватки у производителя позволяет избежать сложной задачи внесения изменений в готовую компоновку платы.

## Макет платы

Входными данными для программы компоновки платы является netlist, который аннотирован информацией о посадочных местах компонентов. Netlist — это промежуточное представление каждого компонента и вывода, а также их соединения. Извлечение netlist — это хорошо автоматизированный процесс, но корреляция схемных символов с посадочными местами на плате ПК не всегда хорошо управляется.

Сложность корреляции символа с посадочным местом возникает из-за наличия нескольких вариантов упаковки для одной детали. Например, символ транзистора может в равной степени подразумевать крошечное устройство в корпусе SOT-23 или огромное устройство в корпусе TO-3, и вам нужно убедиться, что правильный корпус выбран во время извлечения списка соединений. Всегда полезно проверить подразумеваемый посадочный место, когда компонент размещается в схеме, а не проверять все сразу во время трансляции списка соединений или, что еще хуже, во время размещения или окончательного обзора проекта.

Как только у вас будет готовый список соединений, вы готовы заняться компоновкой платы.

Другим внешним входом в программу компоновки платы являются правила проектирования. Правила проектирования устанавливаются компанией, изготавливающей плату, и включают спецификации для минимальной

ширины дорожек и минимального расстояния между дорожками, минимального размера отверстий, минимального колыца сквозных отверстий и количества слоев питания и маршрутизации. Точные правила проектирования зависят от выбранного вами процесса, который, в свою очередь, определяется тем, что вы можете себе позволить. Лучшие процессы предлагают дорожки толщиной до 2 мил (мил равен 1/1000 дюйма или 25,4 микрона) и просверленные лазером глухие/скрытые переходные отверстия примерно такого же диаметра, но цена изготовления выходит далеко за рамки бюджета типичного любителя, составляющего менее ста долларов. Более типичный процесс любителя включает правила проектирования дорожек/пространств 6 мил с минимальными размерами готовых отверстий 15 мил, с двумя или четырьмя слоями меди. (Вы найдете список компаний, изготавливающих платы, в конце этого приложения.)

Компоновка платы состоит из двух этапов: размещение и трассировка. Интеллектуальное размещение деталей значительно упростит задачу трассировки. В целом, цель состоит в том, чтобы разместить все детали так, чтобы соединения были как можно короче, с как можно меньшим количеством переходных отверстий, чтобы свести к минимуму шум, задержку и потери сигнала. Размещение некоторых деталей, таких как разъемы, переключатели и силовые компоненты, хорошо ограничено, что оставляет вам мало выбора. Что касается остальных деталей, понимание конструкции поможет вам определить, какие детали следует разместить наилучшим образом.

После завершения размещения распечатайте проект в масштабе 1:1 и проверьте, что компоненты помещаются в соответствующие им посадочные места, заполнив распечатанный макет фактическими компонентами. Если вы собираетесь использовать гнездо с компонентом, обязательно используйте гнездо для проверки чертежа 1:1, так как гнезда требуют большие места, чем сам компонент. Эта проверка гарантирует, что у вас есть все компоненты в правильном типе корпуса, что все контуры ваших компонентов верны и что между каждым компонентом есть достаточный зазор для облегчения сборки. Еще одна важная вещь, которую следует проверить на чертеже 1:1, — это ориентация и расположение выводов всех разъемов, так как очень легко перевернуть разъем или использовать неправильный посадочный размер на печатной плате. Будьте осторожны при обращении с микросхемами, особенно с теми, у которых есть мелкотабковые выводы для поверхностного монтажа. Убедитесь, что вы не сгибаете выводы, и соблюдайте надлежащий протокол контроля статического электричества.

## Общие рекомендации по размещению и маршрутизации

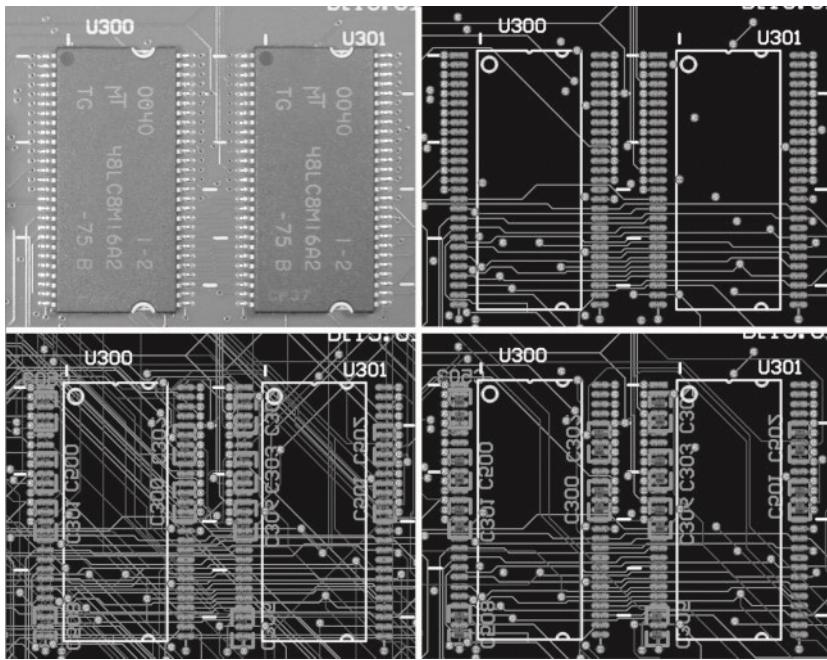
Вот краткий список некоторых рекомендаций по размещению и маршрутизации. Помните, что это всего лишь общие рекомендации, и, несомненно, будут ситуации, когда они не применимы.

## **Оставьте место для разветвлений переходов на поверхности Монтировать устройства**

Устройства поверхностного монтажа обеспечивают значительное преимущество в плотности по сравнению со старыми компонентами сквозного монтажа, которые были фактическим стандартом. Однако компоненты поверхностного монтажа по-прежнему требуют сквозных переходов для трассируемости, особенно в сложных и/или автоматически трассируемых конструкциях. Эти маршрутные переходы называются «разветвленными» переходами для контактных площадок SMD. На рисунках С-2 и С-3 показано использование разветвленных переходов на детали поверхностного монтажа.

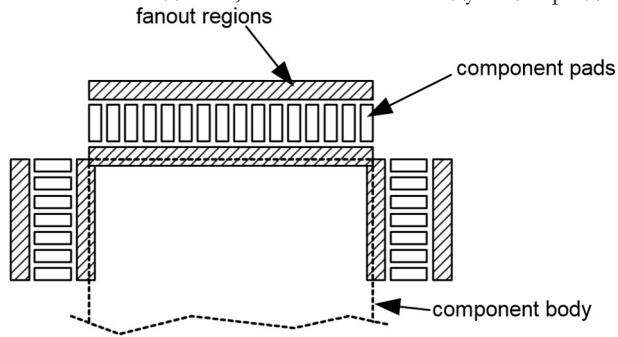
## **Разделительные конденсаторы прекрасно подходят под SMD-площадки**

Наиболее распространенным пассивным компонентом в типичном цифровом проекте является развязывающий конденсатор. Эти крошечные конденсаторы есть везде, и они могут занимать ценнное пространство маршрутизации и разветвления, если они неправильно размещены. Если вы хотите создать двухстороннюю плату поверхностного монтажа, развязывающие конденсаторы можно разместить на стороне платы, противоположной площадкам целевого компонента. Размещая эти компоненты под пространством площадки компонента, вы не занимаете площадь разветвления. Фактически, правильно размещенный



**Рисунок С-2:** Четыре вида компоновки печатной платы. Сверху слева, по часовой стрелке: изготовленная печатная плата с компонентами; вид верхнего слоя печатной платы в программе компоновки печатных плат; вид всех слоев печатной платы в программе компоновки печатных плат; вид только верхнего и нижнего слоев, демонстрирующий двухстороннюю компоновку SMT.

Конденсатор развязки может делить питание через используемые выводы питания компонента. Вид в нижнем левом углу рисунка С-2 наглядно иллюстрирует эту технику. (Есть некоторые особые случаи, когда вы можете не захотеть этого делать, как отмечено в следующем разделе.)



**Рисунок С-3:** Зоны разветвления вокруг посадочного места SMD-компонента.

## Знай свои особые следы

Хорошая новость о разводке цифровых плат заключается в том, что большинство дорожек требуют небольшого размышления, в отличие от типичной аналоговой платы. Плохая новость заключается в том, что если вы не сделаете остальные дорожки правильно, ваша плата будет демонстрировать странное и раздражающее поведение, которое будет трудно отладить. В результате разводка этих специальных дорожек немного похожа на черное искусство. В этом разделе дается всего несколько рекомендаций по работе с этими специальными дорожками, но я призываю заинтересованных читателей найти текст, посвященный разводке плат, чтобы действительно изучить и оценить эти методы. Два текста, которые я рекомендую, это «Digital Systems Engineering» Уильяма Дж. Далли и Джона В. Поултона (Cambridge University Press) и «High Speed Digital Design: A Handbook of Black Magic» Говарда В. Джонсона и Мартина Грэма (Prentice-Hall PTR).

Как правило, при разводке печатной платы особого внимания требуют следующие типы дорожек:

- Следы мощности
- Трассировки опорных сигналов синхронизации (тактов)
- Высокоскоростные трассы
- Аналоговые/смешанные трассировки сигнала

Как правило, дорожки питания должны быть толще, чем ваши средние сигнальные дорожки, особенно если вы используете один из высокопроизводительных производственных процессов, которые предлагают узкую (~ 5 мил) ширину дорожек. Дорожки питания должны быть толще, чтобы противостоять как резистивному нагреву, так и паразитной индуктивности. Узкие дорожки питания, особенно вблизи ключевых точек распределения питания, будут действовать как резисторы и нагреваться, падая напряжение питания до уровня, который вызывает неопределенные неисправности в ваших схемах.

Правильный размер дорожки питания зависит от толщины меди. Типичные платы используют «1-унциевую медь» толщиной 1,35 мил (один квадратный фут медной фольги толщиной 1,35 мил весит одну унцию). Внешняя дорожка шириной 12 мил из 1-унциевой меди требуется для пропускания 1 ампера тока с повышением температуры на 10 градусов Цельсия. Более толстые дорожки требуются для скрытых слоев для аналогичной пропускной способности тока.

При разводке трасс питания между слоями помните, что переходные отверстия также имеют сопротивление. Одного переходного отверстия недостаточно для соединения критических трасс питания между слоями. Критические трассы питания должны иметь несколько переходных отверстий, соединяющих их между слоями, чтобы снизить паразитные

сопротивления и индуктивности. Распределенные плоскости питания на нескольких слоях также должны иметь щедро распределенные переходные отверстия, чтобы гарантировать сохранение общего потенциала.

### Примечание



**В высокопроизводительных или малошумящих приложениях размещение сквозного отверстия между развязывающим конденсатором и силовым выводом может иметь слишком высокую цену электрической целостности для удобства трассировки. Сквозные отверстия нарушают распространение высокоскоростных (сотни мегагерц) электрических волн. Таким образом, оптимальное расположение развязывающего конденсатора в этих приложениях — между компонентным выводом и силовым выводом.**

Сигналы опорного времени включают часы и стробы. Многие устройства памяти требуют асинхронных стробов управления, которые имеют чувствительные требования к синхронизации. Эти сигналы должны быть правильно завершены и направлены в соответствии со стратегией завершения, обычно по маршруту «цепочка». Маршруты цепочки не имеют ответвлений, поэтому для прохождения волнового фронта сигнала существует только один путь.

Электрические сигналы распространяются со скоростью около 1/4 скорости света на печатной плате, или около трех дюймов за наносекунду. Таким образом, высокоскоростные трассы должны иметь согласованные длины, иначе сигналы могут приходить со значительным сдвигом по фазе относительно опорного сигнала синхронизации. Длины трасс согласуются путем удлинения более коротких трасс до длины самой длинной трассы. Удлинение длины трассы достигается с помощью змеевидных трасс, которые изгибаются и увеличивают эффективную длину трассы без изменения размещения конечных точек трассы.

Аналоговая и смешанная маршрутизация сигнала выходит далеко за рамки этого приложения. В среднестатистическом любительском цифровом проекте большая часть аналоговых схем будет изолирована от источников питания. Любые особые требования к компоновке для конкретного компонента источника питания обычно подробно документируются в техническом описании компонента.

Помните, что электрические сигналы ленивы и беспорядочны: ток сигнала всегда будет следовать по пути наименьшего сопротивления, а сигналы будут взаимодействовать с соседними дорожками. Кроме того, ток должен сохраняться, поэтому каждый путь тока сигнала должен иметь путь обратного тока, явный он или нет. Помните об этих простых правилах, когда вы размещаете любые аналоговые секции на вашей печатной плате.

## Печатные платы делают прекрасные радиаторы

При трассировке мощных компонентов, таких как регуляторы мощности и высокопроизводительные микропроцессоры, помните, что медь в печатной плате является отличным проводником тепла. Вы можете сэкономить на радиаторе при определенных условиях, просто разместив большую область меди, подключенную к теплопроводу или заземляющим контактам целевой детали. Если вы используете многослойную конструкцию платы с силовыми плоскостями, используйте несколько переходных отверстий, чтобы помочь привести тепло во внутренние слои.

Теплоотводящие возможности печатной платы также могут быть помехой при ручной сборке. Хорошая теплопроводность меди затрудняет нагрев вывода компонента, который также подключен к большой области меди. При подключении маломощных компонентов к плоскостям питания рассмотрите возможность использования переходных отверстий с термическими рельефами. Термический рельеф — это набор небольших зазоров в соединении переходного отверстия с плоскостью питания, который снижает теплопроводность без существенного влияния на электрические характеристики соединения. (Обратите внимание, что большая группа плотно упакованных термических рельефных переходных отверстий питания вокруг области меди может привести к неподключенным или плохо подключенными островам меди.)

## Установите предпочтительные направления маршрутизации для каждого слоя

Установление доминирующего направления маршрутизации для каждого слоя может упростить маршрутизацию плотных плат. Например, сделайте верхний слой горизонтальным слоем маршрутизации, а нижний — вертикальным слоем маршрутизации. Если вам нужно привести сигнал между двумя компонентами, расположенными по диагонали через плату, сначала проложите горизонтальную трассу сверху, а затем вертикальную трассу снизу, чтобы соединить два компонента. Альтернативная стратегия простого проведения трассы по диагонали через верхний слой платы, например, снижает общую маршрутизуемость между двумя половинами платы наполовину: единственный способ попасть из одной половины в другую теперь — пройти по нижней.

Исключения из этого правила допустимы, особенно если вам приходится идти на компромисс между целостностью сигнала и маршрутизуемостью.

## Сложите доску с ортогональными слоями

После установки предпочтительных направлений маршрутизации для каждого слоя, сложите слои так, чтобы никакие два слоя не имели параллельных предпочтительных направлений маршрутизации. Эта ортогональность помогает свести помехи сигналов между слоями к минимуму. Если у вас есть

слои питания, попробуйте сложить их между слоями, чтобы помочь экранировать помехи между сигнальными слоями.

## На двухслойных платах используйте пальцы для подачи питания

На двухслойной плате часто возникает соблазн просто проложить питание и заземление в виде кольца вокруг внешней стороны платы. Это не идеальная ситуация, поскольку кольцо истощает сердце платы, а также увеличивает вероятность возникновения больших паразитных токовых петель, которые ухудшают производительность схемы. Вместо этого используйте встречно-штыревые и/или сложенные пальцы питания. Эти пальцы устанавливаются доминирующее направление маршрутизации каждого слоя и должны быть проложены до маршрутизации любых сигналов.

## Советы по использованию автоматического маршрутизатора

Автотрассировщики — это сменное благо: они могут сэкономить часы времени на маршрутизацию, но они также могут стать причиной часов раздражающих проблем. Первое правило использования автотрассировщика — никогда не позволять ему работать с вашей единственной копией файла проекта печатной платы. Вместо этого создайте копию вашего проекта и позвольте автотрассировщику творить чудеса на копии.

Второе правило — изучить ошибки автотрассировщика с помощью простых тестовых проектов, прежде чем применять их в окончательном проекте. Автотрассировщики часто имеют критические ошибки или ограничения, которые необходимо понять перед использованием инструмента. При изучении ошибок автотрассировщика обратите особое внимание на то, как он обрабатывает заблокированные трассы, залитые полигоны и неудобные размеры трасс. Некоторые автотрассировщики фактически удаляют заблокированные трассы (дорожки, проложенные вручную и помеченные как неподвижные), в то время как другие игнорируют их или не работают в их присутствии. Это может быть особенно неприятно, если вы потратили часы на прокладку критических цепей питания и синхронизации перед включением автотрассировщика.

Наконец, не рассчитывайте на то, что автотрассировщик полностью разведет сложную плату. Автотрассировщики отлично подходят для быстрой разводки первых 90 процентов платы, но они действительно замедляются по мере того, как плата становится более загруженной. Обратите внимание, что незначительные изменения в размещении компонентов могут создать или сломать автотрассировщик. Многие автотрассировщики не распознают шины или прямые соединения без специальной аннотации или идеального размещения компонентов.

## Инструменты САПР

Инструменты для проектирования плат значительно упали в цене за последние несколько лет. Инструмент, который я чаще всего использую для проектирования плат, — это Protel 99SE. (Я еще не купил более новую версию, Protel DXP.) Protel — это высоконтегрированный инструмент, включающий в себя захват схем, моделирование, управление библиотеками и компоновку платы с проверкой правил проектирования и автоматической трассировкой, все это объединено в одном инструменте. (Похоже, что с каждым выпуском программного обеспечения в среду проектирования интегрируется какая-то новая функция, к лучшему или к худшему.)

Вы можете загрузить 30-дневную полнофункциональную демоверсию программного обеспечения Protel с их веб-сайта [www.protel.com](http://www.protel.com). Хотя полноценная лицензия на продукт стоит тысячи долларов, она все равно выгодно отличается от многих других программных пакетов, которые предлагают ту же глубину функциональности и количество функций. Другие поставщики высококлассных САПР печатных плат включают Mentor (PADS), Cadence (OrCAD) и Altium (P-CAD). Интересно, что Altium также владеет программным пакетом Protel.

Если вы только начинаете и хотите сделать простую разводку платы, некоторые компании по изготовлению плат предлагают полнофункциональные инструменты проектирования бесплатно. ExpressPCB ([www.expresspcb.com](http://www.expresspcb.com)) предлагает бесплатный инструмент для захвата схем и разводки печатных плат для клиентов, которые пользуются их услугами по изготовлению. Их инструмент функционален, но немногого ограничен в плане проверки правил проектирования и практической сложности, для которой вы можете его реально использовать. Однако ExpressPCB является отличным стартовым инструментом для новичков и способен реализовать практически любой проект по оборудованию на выходные.

Перед отправкой готового проекта на производство просмотрите экспортные файлы с помощью стороннего просмотрика файлов, чтобы защитить ваши инвестиции в производство от ошибок в инструментах проектирования. Наиболее распространенным форматом файлов, используемых для изготовления плат, является формат файла «Gerber». Хороший бесплатный просмотрик Gerber, которому я доверяю, создан Graphicode (<http://www.graphicode.com/>).

## Компании по изготовлению досок

Компании по изготовлению плат имеют такой же широкий спектр возможностей, как и инструменты САПР. Некоторые компании

выполняют только крупные производственные заказы, в то время как другие зарабатывают себе на хлеб с маслом, обслуживая рынок быстрых прототипов и любителей. Вот несколько моих любимых компаний по изготовлению плат, а также краткое описание их основных предложений.

## Сьерра Прото Экспресс

Расположенная в Сан-Хосе, Калифорния, компания Sierra Proto Express предлагает одни из самых конкурентоспособных цен на быстрое изготовление прототипов. На момент написания этой статьи компания Sierra Proto Express предлагает линейку процессов «No Touch Product». Эти процессы изготовления имеют строгие требования к правилам проектирования, но они очень доступны по цене. Например, вы можете изготовить двухслойную печатную плату за четыре дня по цене \$34 за плату (минимальный заказ из двух плат) или четырехслойную печатную плату за четыре дня по цене \$51 за плату (минимальный заказ из двух плат). Технология, предлагаемая по этим ценам, представляет собой правило проектирования трасс/пространства 6 миц с размером готовых отверстий 15 миц. Sierra Proto Express также предлагает более быстрые процессы с шириной трасс до 5 миц и размером готовых отверстий 10 миц. Для получения дополнительной информации посетите сайт [www.sierrprotoexpress.com](http://www.sierrprotoexpress.com).

## Системы передачи данных

Компания Data Circuit Systems, также расположенная в Сан-Хосе, Калифорния, является моим поставщиком по выбору для проектов, требующих агрессивных правил проектирования или специальных вариантов обработки, которые не вписываются в предложения более дешевых быстрых компаний. Их всеобъемлющий «Обзор возможностей процесса» (доступный для загрузки на их веб-сайте) является всеобъемлющим и четко написанным, поэтому при интерпретации правил проектирования не нужно много догадываться. Они также проводят довольно строгий набор заводских проверок вашего представленного проекта, которые часто выявляют незначительные ошибки компоновки, которые могут вызвать проблемы в дальнейшем. Я обнаружил, что их персонал компетентен и дружелюбен, и, хотя их цены немного выше, чем у большинства производителей инженерных прототипов, их хорошо документированные проверки процессов и правил проектирования помогают снизить риск агрессивных проектов, и в конечном итоге дополнительные расходы, вероятно, того стоят. Посетите [www.datacircuits.com](http://www.datacircuits.com).

## Расширенные схемы

Advanced Circuits of Aurora, Colorado ([www.4pcb.com](http://www.4pcb.com)) предлагает функцию мгновенной котировки на своем веб-сайте. Эта функция сама по себе делает их хорошим выбором для плат средней сложности, которые не соответствуют ни одному из правил скидок и быстрого выполнения процесса. Вы можете использовать функцию мгновенной котировки, чтобы оптимизировать свой выбор технологии внедрения по цене. Кроме того, они часто предлагают скидки и специальные предложения с быстрым выполнением.

## Печатные схемы Альберты

Компания Alberta Printed Circuits (AP Circuits), расположенная в Альберте, Канада, является одним из первых домов по изготовлению прототипов печатных плат быстрого изготовления с быстрым обслуживанием.

Предлагаемый ими процесс P1 является базовым, без паяльной маски или шелкографии. В результате сложно выполнять мелкошаговые конструкции поверхностного монтажа, поскольку припой имеет тенденцию попадать везде на этапе сборки. Тем не менее, они изготавливают и отправляют вашу плату по процессу P1 всего за один день по неслыханной цене, без требований к минимальному заказу. Базовая плата за производственный цикл составляет около 45 долларов США на момент написания статьи, с приблизительной платой в 0,65 доллара США за квадратный дюйм сверх базовой платы. Технология представляет собой дорожку/пространство 8 миль с минимальным размером сверления 20 миль (28 миль, если вы хотите придерживаться самого дешевого варианта процесса). AP Circuits отлично подходит для плат, которые нужно сделать в сжатые сроки и с ограниченным бюджетом, особенно если вы используете компоненты для сквозного монтажа или грубые компоненты SMT, которые легко собрать без паяльной маски. Посетите [www.apcircuits.com](http://www.apcircuits.com).

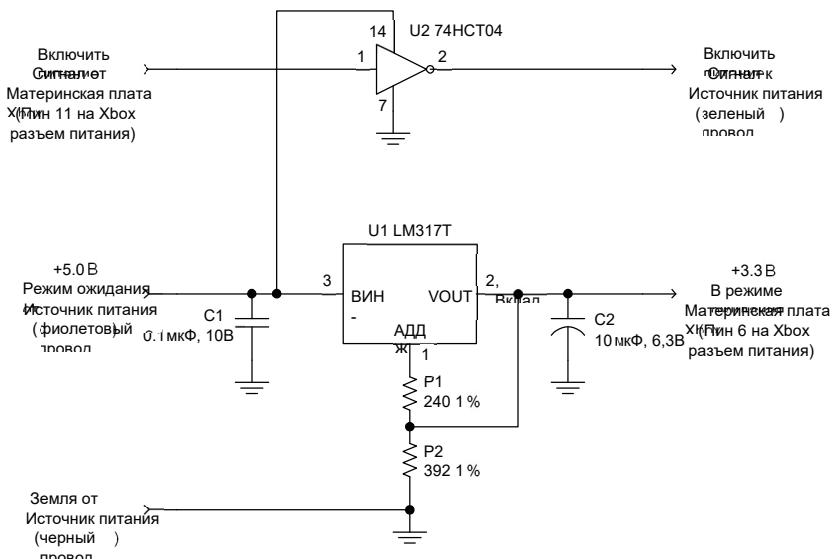
## Стартовые проекты

В главе 5 «Замена сломанного блока питания» вам объяснят, как заменить блок питания Xbox на стандартный блок питания ATX. Единственная проблема заключается в том, что полярность сигнала включения питания инвертирована между Xbox и стандартным блоком питания ATX. Решение, предложенное в главе, заключается в том, чтобы всегда оставлять блок питания включенным, а вместо этого включать и выключать Xbox, сначала включая блок питания, а затем нажимая кнопку питания Xbox.

Довольно легко спроектировать и спроектировать плату, которая позволяет вам инвертировать полярность сигнала питания, чтобы вы могли управлять состоянием питания Xbox с передней панели Xbox. Вы также можете правильно регулировать резервное питание, вместо того чтобы использовать два диода. Такая плата будет состоять из инверторной микросхемы, такой как 74HCT04, и регулятора, такого как LM317K. LM317K — это регулируемый регулятор, который можно настроить на снижение резервного напряжения +5 В, обеспечиваемого источником питания ATX, до резервного напряжения +3,3 В, требуемого Xbox. Пример принципиальной схемы этой платы показан на рисунке C-4.

Выбор разъемов для этой платы остается за вами. Самым простым решением будет просто использовать отверстия и припаять провода через отверстия. На этой плате всего пять соединений. Три из них идут к блоку питания: провод +5VSB (фиолетовый), провод заземления (черный) и провод выхода питания (зеленый). Оставшиеся два, +3.3VSB (контакт 6 на разъеме питания) и вход питания (контакт 11 на разъеме питания), идут к Xbox.

Обязательно проверьте выходное напряжение регулятора перед установкой готовой платы. Довольно легко ошибиться с номиналом резистора или перепутать штырь, и оба эти условия могут привести к опасно высокому напряжению, поступающему в Xbox. Кроме того, при постоянной установке платы обязательно изолируйте нижнюю и верхнюю части платы от случайного контакта с корпусом Xbox или другими компонентами Xbox.



**Рисунок С-4:** Пример принципиальной схемы платы адаптера для замены блока питания ATX. Резисторы R1 и R2 программируют выходное напряжение регулятора напряжения U1 на +3,3 В.



## АПРИЛОЖЕНИЕ Д

# Начало работы с ПЛИС

Интеграция — это бич хакеров, занимающихся оборудованием. Нам нравится разбирать вещи, изменять их и улучшать, но тенденция заключается в том, чтобы втиснуть все в одну или две ASIC (интегральная схема специального назначения). Такой тип интеграции недоступен простым смертным, поскольку стоимость набора масок, используемых для определения характеристик на чипах, стремительно приближается к миллиону долларов. Это один миллион долларов за уникальную ревизию чипа. Если будет допущена ошибка, требующая нового набора масок, вам придется потратить еще миллион долларов, чтобы ее исправить.

К счастью, миллион долларов авансом за чип — это слишком много даже для многих корпораций, и это создало рынок ПЛИС — универсальных программируемых («перенастраиваемых») аппаратных устройств, которые можно использовать вместо ASIC во многих приложениях.

## Что такое ПЛИС?

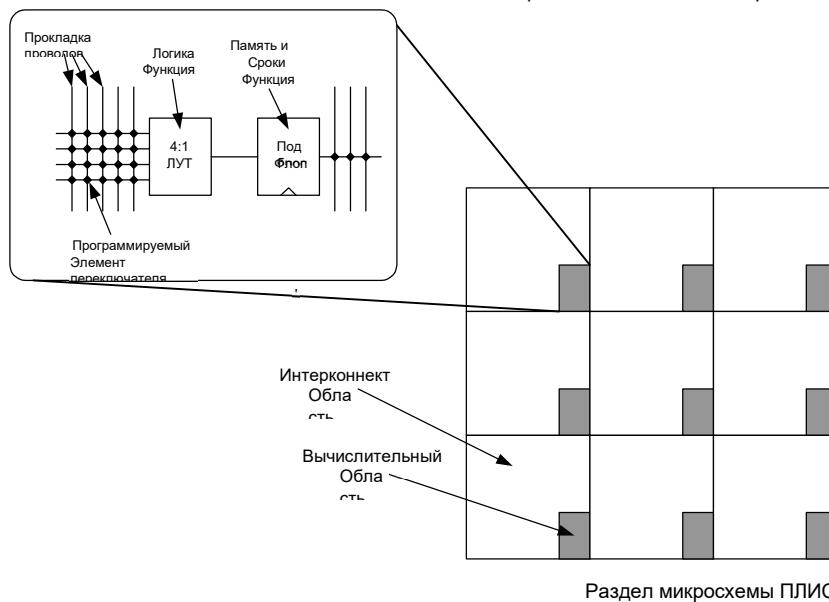
FPGA означает field programmable gate array (программируемая вентильная матрица). Другими словами, это массив вентилей, которые могут программироваться в полевых условиях конечными пользователями. Вы можете думать о FPGA как о специальном кремнии, которое вы можете построить в комфорте собственного дома, хотя тенденция к частичной реконфигуруемости и контекстно-зависимой реконфигурации добавляет измерение к FPGA, которого нет в ASIC. Хотя ASIC дешевле в расчете на единицу объема и могут иметь гораздо более высокую производительность тактовой частоты, FPGA зарекомендовали себя как инструмент

выбора для приложений с малым и средним объемом и для прототипирования.

Базовая архитектура FPGA представляет собой массив аппаратных примитивов, встроенных в гибкую сеть маршрутизации. Мощь FPGA заключается в том, что сложные вычисления можно разбить на последовательность более простых логических функций. Каждая из этих более простых функций может быть разбита по очереди, пока все вычисление не будет описано не более чем последовательностью базовых логических операций, которые можно отобразить в аппаратных примитивах FPGA. Таким образом, одну и ту же FPGA можно использовать для реализации микропроцессора, видеоконтроллера или игры в крестики-нолики, просто изменив конфигурацию аппаратных примитивов и сети маршрутизации.

Виды аппаратных примитивов, реализованных архитектурой FPGA, сильно влияют на эффективность реализации FPGA для заданного целевого приложения. Современные FPGA предоставляют разработчикам в основном примитивы шириной в один бит: таблица поиска с 4 или 5 входами для 1-битного выхода и один бит синхронизированного по времени хранилища, известного как триггер. Таблицы поиска используются в качестве логического примитива, поскольку их можно запрограммировать на выполнение любой логической операции с таким количеством терминов, сколько входов в таблице поиска. Затем эти примитивы подключаются в обширную программируемую сеть проводов; типичная высокопроизводительная FPGA может иметь многие десятки тысяч таких примитивных элементов.

Оказывается, что хотя структуры шириной в один бит являются очень общими, они могут быть очень неэффективными с точки зрения ресурсов в приложениях, где естественная ширина данных велика. В частности, область, выделенная для фактических логических примитивов, во многих случаях составляет около 1 процента, а остальное — это конфигурационная память



Раздел микросхемы ПЛИС

**Рисунок D-1:** Блок-схема типичной структуры ПЛИС, иллюстрирующая несоответствие между количеством проводов в ПЛИС и объемом вычислительной логики.

Типичная современная ПЛИС будет содержать несколько десятков тысяч таких базовых ячеек.

и межсоединение. Все эти провода необходимы для обработки многочисленных перестановок маршрутизации, которые могут потребоваться для однобитовых приложений.

Для повышения эффективности использования площади многие ПЛИС также включают несколько примитивов грубой обработки, таких как фрагменты ОЗУ или блок умножителя. ПЛИС Virtex II-Pro от Xilinx даже включают несколько ядер PowerPC на кристалле. Хотя это звучит впечатляюще, фактическая площадь, потребляемая таким ядром, на удивление мала: процессор PowerPC, вероятно, потребляет немного больше 1 мм<sup>2</sup> кремниевой площади, тогда как площадь ПЛИС составляет сотни квадратных миллиметров.

Самые последние ПЛИС на рынке имеют очень гибкие вводы/выводы в дополнение к очень гибкому вычислительному оборудованию. Типичная ПЛИС может взаимодействовать со всеми наиболее популярными высокоскоростными стандартами сигнализации, включая PCI, AGP, LVDS, HSTL, SSTL и GTL. Кроме того, большинство ПЛИС также могут обрабатывать сигналы с тактовой частотой DDR. Если эти аббревиатуры вам ничего не говорят, основная идея заключается в том, что ПЛИС можно

использовать для взаимодействия практически с любым оборудованием, которое вы можете найти на типичной материнской плате ПК, например, Xbox. Это чрезвычайно хорошая новость для хакеров оборудования, поскольку это означает, что ПЛИС можно использовать для эмуляции или мониторинга практически любого чипа, обнаруженного в ПК. (Конечно, ПК может быть понижен в тех случаях, когда ПЛИС не может справиться со скоростью ПК.)

## Проектирование для ПЛИС

У вас есть несколько вариантов ввода проекта на выбор для типичного потока проектирования ПЛИС. Если вы предпочитаете мыслить графически, большинство потоков проектирования поддерживают инструмент захвата схемы. Хотя захват схемы часто более интуитивен для аппаратных проектов, их может быть сложнее поддерживать и изменять. Например, изменение всех экземпляров имени сети может быть утомительным, если вам нужно щелкнуть по каждому проводу и ввести новое имя. Кроме того, размер любого отдельного уровня иерархии проекта ограничен размером листа схемы, поэтому сложный проект потребует значительного планирования и предусмотрительности только для захвата схемы.

В результате языки описания оборудования (HDL) являются предпочтительным инструментом для реализации сложных проектов. На первый взгляд HDL очень похожи на обычные языки программирования. Например, синтаксис Verilog очень похож на C или Java. Однако семантика языка может оказаться немного сложной для понимания.

Аппаратное обеспечение имеет присущий ему параллелизм, который процедурные языки, такие как C, не могут выразить. Если задуматься, каждый вентиль и каждый триггер на FPGA могут вычисляться параллельно, тогда как в программе на C номинально предполагается один поток выполнения. В результате HDL представляют аппаратное обеспечение как набор процессов, которые работают параллельно; задача кодера — сгруппировать все функции в правильные процессы, чтобы компилятор мог понять, как превратить процесс в вентили.

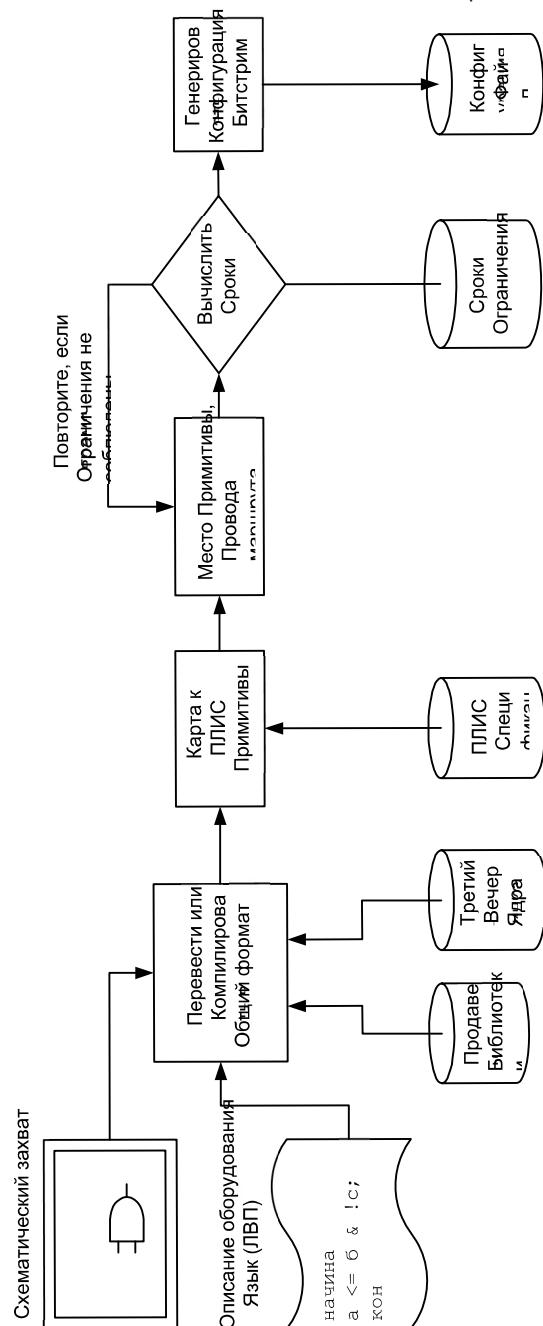


Рисунок D.1 Базовый процесс проектирования  
ПЛИС

Например, один тактируемый элемент хранения (триггер) в Verilog представляет собой «процесс», который обычно имеет структуру, подобную этой:

---

```
input inData; // объявляем ваши входы и выходы input clock;
reg bitOfStorage; // объявляем бит хранения как тип reg

всегда @(posedge clock) начало
bitOfStorage <= inData; конец
```

Этот код принимает значение на входном порту inData и на каждом переднем фронте тактового сигнала сохраняет inData в триггере, выход которого называется bitOfStorage. Несколько процессов, разделенных синтаксисом always @(... ) begin ... end, могут существовать в одной конструкции, и все процессы выполняются параллельно.

Комбинационная логика также может быть выражена как процесс. Например, следующий код Verilog реализует двухвыходовой мультиплексор, не имеющий часов:

```
вход a; вход b; вход select; выход
out; reg c; всегда @(a или b или
select) begin if( select == 1'b1 )
begin c <= a; end else begin c <= b;
end end

assign out = c; // операторы assign могут также содержать
логические // функции
```

В этом примере содержимое заключенного в скобки блока, следующего за ключевым словом always, содержит список чувствительности, включающий все входы, которые могут повлиять на выход. Исключение параметра из списка чувствительности означает, что выход не изменится, даже если этот параметр изменится. Например, если вы исключили a и b из списка чувствительности, то выход изменится только при изменении select: вы бы создали западку, которая хранит либо a, либо b в зависимости от состояния select. Однако желаемая операция мультиплексора — передавать изменения либо a, либо b на выход в любое время, даже когда select не переходит, поэтому a и b должны быть частью списка чувствительности.

При изучении HDL есть ряд тонкостей, которые выходят за рамки этой книги, но два фрагмента кода выше должны дать вам представление о том, чего ожидать. Опытный программист может столкнуться с большими трудностями при адаптации к HDL, чем новичок, поскольку многие программные трюки, которые считаются само собой разумеющимися, очень плохо переносятся в прямую аппаратную реализацию. Массивы, структуры, примитивы умножения и деления считаются само собой разумеющимися в мире программного обеспечения, но каждая из этих конструкций переводится в потенциально большие и неэффективные блоки оборудования. Более того, в аппаратной реализации все возможные случаи в операторе case существуют независимо от того, намеревались ли вы это делать; пренебрежение полным указанием оператора case с

помощью случая по умолчанию часто означает, что для обработки неявных случаев будет синтезировано дополнительное оборудование. Многочисленные учебные пособия и справочные руководства по синтаксису для Verilog проиндексированы в Google;

## Разгон ПЛИС-проектов

Стоит отметить, что модели синхронизации, используемые для FPGA, довольно консервативны. Это означает, что вполне вероятно, что FPGA будет работать правильно на частотах, намного превышающих допустимые временными анализатором. Фактически, тщательная ручная компоновка логики FPGA может значительно расширить производительность FPGA по сравнению с заявленными характеристиками.

Например, ПЛИС (Xilinx Virtex-E), используемая для реализации ответителя шины Xbox Hypertransport, рассчитана только на скорость передачи данных около 200 Мбит/с/контакт, но приложение требовало 400 Мбит/с/контакт. Причина, по которой мне удалось это осуществить, заключается в том, что реальные логические элементы и элементы хранения могут работать очень быстро, но большая часть производительности сжигается в проводах и повторителях, которые передают сигналы между логическими элементами. В частности, некоторые провода будут иметь такую большую задержку при 400 Мбит/с, что они фактически хранят данные в течение одного тактового цикла.

Я определил, какие провода были медленнее остальных, захватив последовательность данных и сравнив ее с шаблоном, который я ранее обнаружил с помощью осциллографа. После того, как медленные пути были идентифицированы, я инвертировал часы и/или вставил триггеры в каналы, которые имели слишком малую задержку. Конечным результатом был набор сигналов, которые были скорректированы по временному сдвигу. Затем эти сигналы можно было trivialно демультиплексировать до более низкой тактовой частоты, где можно было использовать традиционно скомпилированные методы проектирования HDL.

Хотя этот метод очень мощный, он не применяется повсеместно, поскольку величина задержки, вызванной проводом, варьируется от чипа к чипу и может зависеть от таких параметров, как температура окружающей среды и качество напряжения питания. Однако для одного конкретного чипа при контролируемых обстоятельствах мне удалось получить производительность в 2 раза выше номинальной. Еще одно важное отличие этого приложения от более общего заключается в том, что частота ошибок по битам порядка 1 ошибка на несколько тысяч была приемлемой, поскольку я мог просто взять три трассы и выполнить над ними операцию XOR, чтобы восстановить любую информацию, потерянную из-за случайных источников шума. Однако частота ошибок по битам 1 на 10 000 неприемлема для обычных приложений; более типичны неисправимые частоты ошибок лучше 1 на 10 000 000 000. Все это возвращает нас к моей поговорке: «Легко сделать что-то один раз, но сделать что-то идеально миллион раз — сложно».

Синтаксис verilog и учебник verilog — хорошие наборы ключевых слов для начала поиска справок по синтаксису или учебников. На сайте Xilinx

также есть хороший справочник по Verilog для разработчиков ПЛИС, а у Sutherland HDL, Inc. есть бесплатное краткое справочное руководство по Verilog по адресу [http://www.sutherland-hdl.com/online\\_ref\\_guide/vlog\\_ref\\_body.html](http://www.sutherland-hdl.com/online_ref_guide/vlog_ref_body.html). Еще одним преимуществом подхода к началу проектирования HDL является наличие бесплатных и платных «softcores». Такие веб-сайты, как [www.opencores.org](http://www.opencores.org), предлагают общедоступные лицензионные ядра HDL для таких функций, как интерфейсы USB, криптодвижки DES и AES и различные микропроцессоры. Кроме того, почти каждая стандартная функция предлагается сторонними поставщиками, которые продадут вам ядра за плату.

После ввода проекта я настоятельно рекомендовал вам смоделировать свой проект перед его компиляцией в оборудование. Попытка отследить ошибки, крутя код, отправляя его на оборудование и проверяя изменения, очень неэффективна. Моделирование позволяет вам проверить любой узел схемы нажатием кнопки. Кроме того, усилия, необходимые для моделирования изменения кода, очень малы, особенно по сравнению с усилиями по отправке изменения на весь путь через оборудование.

После ввода и моделирования проекта его необходимо скомпилировать или перевести в общий формат списка соединений. Этот формат списка соединений подается в программу, которая сопоставляет примитивы списка соединений с целевыми аппаратными примитивами ПЛИС, после чего сопоставленные примитивы размещаются и маршрутизируются. Полученный проект анализируется на соответствие набору ограничений, указанных проектировщиком. Если проект не соответствует спецификациям проектировщика, он итеративно дорабатывается с помощью последовательных проходов размещения и маршрутизации. После того, как проект проходит свои ограничения проектирования, он переходит к генератору битового потока конфигурации, где внутреннее представление ПЛИС преобразуется в двоичный файл, который ПЛИС может использовать для своей настройки. (Все эти шаги происходят довольно гладко одним нажатием кнопки в более поздних версиях инструментов проектирования ПЛИС.)

## Идеи проекта

Теперь, когда вы немного знаете о том, что такое ПЛИС и как их программировать, что вы можете с ними делать?

Как оказалось, в наши дни ПЛИС обладают достаточной логической емкостью и производительностью для выполнения весьма впечатляющего спектра задач. Очевидное промышленное применение ПЛИС — это эмуляция проектов, предназначенных для жестко смонтированного кремния. Стоимость создания заказного чипа стремительно растет, и скоро будет так, что исправление одной критической ошибки может обойтись в сотни тысяч долларов, если не в миллионы.

С другой стороны, исправление ошибки, допущенной в описании FPGA HDL, в основном стоит только времени и усилий по проектированию; вы не выбрасываете никаких деталей, и вам не нужно покупать никаких новых деталей. Таким образом, многие компании приняли стратегию полной имитации макета дизайна в FPGA перед тем, как выпустить окончательный кремний. Побочным преимуществом этого подхода является то, что команды по программному обеспечению и оборудованию, которые являются пользователями пользовательского кремния, могут начать проверку своих проектов с использованием макета FPGA, пока изготавливается пользовательский кремний; процесс, который иногда может занять пару месяцев.

Для хакеров FPGA являются своего рода панацеей для всех видов сложных проектов. Они являются отличным выбором для реализации криптографических функций, если вы заинтересованы в выполнении перебора ключей или быстрым шифрованием больших объемов данных. Они также очень полезны для реализации функций обработки сигналов, особенно с учетом существования свободных ядер умножителей и цифровых фильтров. FPGA могут достигать более высокой производительности при меньшем энергопотреблении, чем DSP, и, таким образом, они занимают уникальную нишу в таких приложениях, как робототехника с питанием от батареи. FPGA также полезны для приложений встроенных контроллеров: небольшое ядро микропроцессора, эквивалентное или лучшее, чем PIC, может легко поместиться в FPGA сегодня. Добавьте все ваши пользовательские периферийные устройства, такие как последовательный порт и генераторы синхронизации ШИМ, и вы в деле.

FPGA также полезны в ситуациях, когда фокус не сосредоточен на обработке больших чисел. FPGA отлично подходит в качестве связующего звена в труднодоступном месте, а правильно размещенная FPGA может избавить вас от необходимости добавлять перемычки для исправления платы из-за ошибки проектирования логики. FPGA также являются дешевой альтернативой логическому анализатору для тех из нас, кто не может позволить себе майнфрейм Tek TLA за 10 000 долларов. Высокоскоростные возможности ввода-вывода новейших FPGA в сочетании с большой автоматически генерируемой встроенной памятью FIFO позволяют быстро разработать систему захвата и анализа сигнала.

Наконец, FPGA имеют приложения в ситуациях со смешанными сигналами, которые не сразу очевидны. Наиболее распространенным приложением со смешанными сигналами, вероятно, является использование FPGA для управления аналоговыми сигналами монитора VGA. Пара резистивных делителей или хорошо выбранный тип выходного драйвера — это все, что вам нужно, а вся синхронизация и логика, необходимые для генерации цветных изображений, могут быть обработаны логикой внутри FPGA. FPGA также могут быть тривиально

использованы в качестве ШИМ-ЦАП или даже как часть сигма-дельта-ЦАП или АЦП.

## Где купить

Вы, вероятно, думаете, что любой универсальный и мощный инструмент должен стоить целое состояние. Хотя это было правдой около десяти лет назад, сегодня вы можете купить 100 000 вентильных ПЛИС менее чем за 50 долларов, а инструменты проектирования часто бесплатны для пользователей, обучающихся в сфере образования, и/или любителей.

Конечно, ПЛИС сама по себе не так уж и полезна; для использования ее необходимо установить на плату с соответствующими соединениями. Для этого компания XESS ([www.xess.com](http://www.xess.com)) выпускает линейку довольно доступных стартовых наборов ПЛИС. Их линейка продукции меняется по мере появления новых ПЛИС, но текущей платой ПЛИС начального уровня является плата XSA-50, которая поставляется с ПЛИС на 50 000 вентиляй примерно за 150 долларов. Плата также включает несколько мегабайт оперативной памяти, параллельный порт, порт VGA, порт клавиатуры PS/2 и несколько других важных элементов.

Другой вариант — собрать свою собственную плату с нуля, если вы чувствуете себя смелым. Другие приложения в этой книге описывают, как приступить к компоновке и изготовлению платы и как прикрепить мелкошаговые устройства FPGA к вашим платам. На самом деле, довольно полезно попытаться собрать свои собственные платы, и я рекомендую попробовать; стоимость изготовления платы в наши дни значительно ниже 100 долларов, так что вы не потеряете слишком много, даже если ваша плата в конечном итоге не будет работать.

Если вы делаете свою собственную плату, вам нужно будет купить ПЛИС у дистрибутора Xilinx. На веб-странице Xilinx ([www.xilinx.com](http://www.xilinx.com)) есть самые свежие ссылки на дистрибуторов. На момент написания этой статьи одним из наиболее удобных дистрибуторов является NuHorizons ([www.nuhorizons.com](http://www.nuhorizons.com)), поскольку они предлагают информацию о наличии продукции и ценах на своей веб-странице без необходимости регистрации или создания специальной учетной записи клиента.

Программное обеспечение для разработки FPGA обычно можно приобрести по низкой цене или бесплатно. Например, Xilinx предлагает бесплатную среду разработки для своих линеек деталей Virtex-II (до 300 тыс. вентиляй), Spartan II-E и CoolRunner. Среда разработки называется Xilinx ISE WebPACK и доступна для загрузки после регистрации на сайте [www.xilinx.com](http://www.xilinx.com). Эта бесплатная среда обладает впечатляющим списком функций, включая ввод схем и HDL, синтез HDL, flooplanner, управляемое временем место и маршрут, временной анализ и инструменты анализа мощности.

Xilinx также предлагает версию своего программного обеспечения под названием «Xilinx Student Edition» через Prentice-Hall. Это программное обеспечение поставляется в комплекте с рядом учебных пособий и документации, которые могут помочь вам в проектировании ПЛИС. Вы найдете широкий спектр полезных учебных пособий и лекций на веб-сайте Xilinx на вкладке «Образование».



## АПРИЛОЖЕНИЕ Э

# Отладка: советы и подсказки

## Не паникуйте!

Развитие навыков отладки так же важно, если не важнее, чем развитие навыков дизайна. Самый важный совет — никогда не паниковать: беспорядочные настройки и изменения привнесут больше неопределенности и ошибок, чем исправят.

Отладка проста, когда есть полная видимость системы и полное знание ожидаемого состояния правильно работающей системы. Простое сравнение наблюдаемого состояния с ожидаемым состоянием прояснит, что пошло не так. К сожалению, мир редко работает таким образом. Чипы — это черные ящики, и единственная видимость внутреннего состояния чипа — через его контакты. Многие сигналы также слишком сложно напрямую измерить или записать. Кроме того, спецификации, предоставляемые производителями, часто расплывчаты или трудны для интерпретации. Таким образом, настоящеe искусство отладки заключается в отслеживании набора симптомов до первопричины, несмотря на отсутствие видимости и полного знания системы.

## Понять систему

Попытка отладки системы без предварительного понимания того, что вы пытаетесь отладить, похожа на попытку прочитать японский комикс без знания японского языка. Вы можете понять на поверхностном уровне, кто плохой парень, а кто хороший, но вы действительно теряетесь в том, какое отношение ко всему этому имеет плавающий кот. Чтобы полностью

понять сюжет, вам понадобится японский словарь и много времени и терпения. Аналогично, основные принципы электроники и интуиция приведут вас к точке, где вы примерно будете знать, чего ожидать, но просветление наступит только после того, как вы прочитаете спецификации компонентов. Чем больше вы понимаете о системе, тем легче будет выяснить, почему что-тошло не так. Делайте заметки, когда вы больше читаете о системе, и думайте про себя о том, как проблемы могут проявить себя, если что-то пойдет не так. Также полезно увидеть другие системы, похожие на ту, которую вы пытаетесь исправить, и это помогает понять теорию работы.

## Наблюдайте симптомы

Ошибки проявляются через симптомы, и вам предстоит определить первопричину, наблюдая за несколькими симптомами и выявляя виновника. Пустой экран на телевизоре, на котором должен отображаться видеовыход вашей консоли, является примером симптома. Существует множество причин, по которым экран вашего телевизора может быть пустым, например, сломанный видеокабель, сломанный телевизор, сломанный видеоразъем, сломанный источник видео, пустой носитель в источнике видео или даже отсутствие питания системы. Как правило, вы должны заметить по крайней мере два, а лучше три симптома, которые соответствуют причине, прежде чем сделать вывод, что вы нашли первопричину. Имейте в виду, что наиболее показательные симптомы часто не очевидны внешне, и для их обнаружения потребуется измерение или эксперимент. В примере с нашим пустым экраном телевизора наши измерения так же просты, как проверка того, загорается ли индикатор питания на телевизоре или выходит ли звук из телевизора без видео.

Основная стратегия отладки — начать с очевидного симптома и изолировать различные части системы, чтобы определить, какая часть является непосредственной причиной симптома. Непосредственная причина определяется как то, что напрямую влияет на наблюдаемый симптом. Непосредственными причинами сбоя видео на телевизоре являются отсутствие сигнала на телевизоре, сломанный телевизор или отсутствие питания; ненепосредственными причинами будут аппаратный сбой в вашем видеоисточнике или фаза луны. Другими словами, при наличии симптома A подумайте обо всех возможных непосредственных причинах X, Y и Z, а затем проверьте каждую из них, чтобы определить, какая из них является фактической причиной. После того как вы изолировали проблему, подумайте о том, что могло вызвать ее сбой, и повторяйте процесс, пока не обнаружите основную причину.

Выделение причины ошибок может быть облегчено использованием заведомо хороших эталонов. В нашем примере вы можете исключить телевизор как источник сбоя, подав на него сигнал с заведомо хорошего DVD-плеера. Для того чтобы эксперимент с заведомо хорошим эталоном был действительным, вы

должны поддерживать все постоянным, за исключением той части, которую вы заменяете эталоном. Подключение хорошего DVD-плеера к другому входу, нежели консоль на телевизоре, скажет нам только о том, что дисплейная часть телевизора работает. Путь от входа консоли к телевизору не тестируется. Правильное выполнение эксперимента подразумевает подключение DVD-плеера к видеовходу, используемому консолью.

Такая паранойя или врожденное недоверие к системе становится очень важным при отслеживании тонких аппаратных ошибок. Не принимайте как должное ни один фактор, который может повлиять на систему, которую вы наблюдаете, и никогда, никогда не игнорируйте необъяснимое или непоследовательное поведение, даже если оно прерывистое. Например, иногда система будет работать правильно или сломается, если вы коснетесь определенного места на плате или помашете рукой около определенной области; иногда система будет демонстрировать другое поведение в течение короткого момента после включения питания. Возникает соблазн списать такие наблюдения как аномалии или тривиальные происшествия, но факт в том, что они произошли, и должно быть объяснение. Одним из конкретных примеров является прикосновение к плате и наблюдение изменения состояния системы. Где вы коснулись? Как вы коснулись ее? Ваши руки потные или сухие? Когда вы касаетесь платы, ваше тело действует как небольшая емкость и большое сопротивление. Это может немного замедлить сигналы или разрядить узлы с высоким импедансом, такие как неподключенный цифровой вход. Если вы сильно надавите на плату, вы можете согнуть ее таким образом, что изменятся электрические свойства треснувшей дорожки или плохого паяного соединения.

Есть некоторые симптомы, которые часто неправильно интерпретируются как причины. Сгоревшая дорожка или поврежденный компонент обычно являются симптомом, а не причиной проблемы. Другими словами, неисправность в другом месте цепи обычно является причиной отказа компонента. Спонтанный отказ компонента является относительно редким явлением. Предположим, вы отлаживаете сломанную стереосистему. Вы чувствуете запах горелого, исходящий от стереосистемы, и видите большой резистор, почерневший от перегрева. Есть вероятность, что если вы просто замените этот резистор, новый просто снова сгорит. Настоящей причиной может быть закороченный транзистор или поврежденная цепь питания, но они не проявляются так явно, как сгоревший резистор.

Другой эффективный метод наблюдения — сравнение с заведомо исправной системой. Если вы пытаетесь отладить сломанное устройство, найдите работающее и сравните напряжения и другие рабочие характеристики между ними. Если вы пытаетесь отладить свою собственную самодельную систему, постройте симуляцию схемы, если это возможно, или найдите схему с похожей конструкцией. Вы можете использовать эти заведомо исправные системы для быстрой изоляции аномального поведения. Кроме того, вы можете контролируемым образом вызывать сбои в заведомо исправном образце, чтобы проверить,

действительно ли вы нашли первопричину проблемы. Этот метод особенно применим к моделируемым системам.

## Распространенные ошибки

Наиболее распространенным источником аппаратных ошибок в проектах домашнего пивоварения являются плохие паяные соединения и неправильно установленные поляризованные компоненты, такие как конденсаторы, диоды, микросхемы и разъемы. Кроме того, разъемы являются особенно печально известными источниками отказов, поскольку они подвергаются наибольшему физическому воздействию, и обычно трудно определить, находится ли разъем в хорошем состоянии, только с помощью визуального осмотра. Ниже приведен список распространенных ошибок, ранжированных в порядке популярности.

1. **Плохое паяное соединение.** Сюда входят холодные паяные соединения, перемычки и забытые соединения. Тщательный визуальный осмотр может выявить множество случаев плохих паяных соединений. Припой между всеми соединениями должен выглядеть гладким и блестящим, а припой должен иметь влажный вид мениска на контактных площадках печатной платы и выводах компонентов. Фотографии хороших и плохих паяных соединений можно найти в Приложении В: Методы пайки. Плохие паяные соединения также можно быстро определить на многих корпусах поверхностного монтажа, осторожно проведя жесткой проволокой, например кончиком пинцета или скрепкой, по контактам по всей длине корпуса. Плохо соединенные контакты будут слегка изгибаться. Сгибание платы также может помочь выявить плохие паяные соединения. В других случаях вам может потребоваться использовать омметр для проверки качества паяного соединения. (Если у вас был грязный опыт пайки компонентов, очистите плату мягким растворителем, например, изопропиловым спиртом, используя ватный тампон перед проверкой.) Наконец, помните, что лучше один раз увидеть, чем сто раз поверить: используйте увеличительную линзу, чтобы облегчить осмотр. Предпочтительнее использовать микроскоп средней мощности, но любая установленная увеличительная линза (например, те, что используются в чертежных лампах) или колышевая лупа, вроде той, что используют ювелиры, окажут огромную помощь.
2. **Неправильные значения компонентов.** Неправильное значение компонента может возникнуть, когда на печатной плате случайно установлен похожий на вид, но другой по номиналу компонент. Это особенно проблематично для пассивных компонентов поверхностного монтажа, которые часто не имеют маркировки или имеют нечеткую маркировку. Помните, что единственный способ правильно проверить значение компонента — это снять его с платы и затем проверить. Заполнения плат неправильными компонентами можно избежать, если

очень внимательно и методично хранить компоненты в четко маркированных пакетах или коробках во время сборки.

3. **Плохие разъемы.** Сюда входят разъемы, которые были установлены наоборот или, что еще хуже, спроектированы с неправильным назначением контактов. Обратите внимание на то, где находится контакт 1, и на систему нумерации, используемую разъемом. Некоторые разъемы используют зигзагообразную систему нумерации контактов, в то время как другие используют круговую систему нумерации контактов. Разъемы «провод-плата» также трудно изготовить вручную. Осмотрите все точки, где провода соприкасаются с контактами разъема, на предмет плохого обжима, избыточной изоляции или плохих паяных соединений. В худшем случае используйте вольтметр, чтобы проверить непрерывность разъема.
4. **Ошибки в конфигурации из-за невнимательного прочтения паспорта.** Сложные микросхемы часто поддерживают несколько режимов работы, которые выбираются путем привязки набора выводов к высоким или низким логическим уровням. Микросхемам также часто требуются внешние резисторы для загрузки или смешения вывода для правильной работы. Иногда для микросхем требуется сети конденсаторов, резисторов и индукторов, а также для стабилизации внутренних функций. Имейте в виду, что неиспользуемые входы часто требуют терминации на фиксированном напряжение для правильной работы, поэтому не игнорируйте части спецификации только потому, что вы не используете определенные функции.
5. **Проблема проектирования или проблема реализации.** Иногда ошибка вызвана прямой ошибкой проектирования или проблемой перевода между правильной схемой и макетом платы. Проблемы перевода часто возникают из-за опечаток при указании имен схемных цепей или неявных имен питания на символах схемы. Неявные имена питания часто используются на цифровых компонентах для удобства, но могут вызывать значительные проблемы в проектах, использующих несколько напряжений питания. Эти виды проблем можно обнаружить до перехода к макету с помощью программы проверки эвристического списка соединений, как описано в Приложении С. Нарушения правил проектирования на высокой скорости представляют собой другой тип проблем реализации. Схемы, которые работают на высоких частотах (25+ МГц) или имеют высокие скорости фронтов (< 5 нс), требуют особого внимания к электрическому импедансу и согласованию линии передачи.
6. **Блок питания не соответствует спецификации.** Проверяйте напряжение питания как можно ближе к точке использования, так как провода могут снизить фактическое подаваемое напряжение. В некоторых случаях в схеме все в порядке, а блок питания просто не способен обеспечить достаточное количество энергии для работы вашего проекта. Также проверьте, не меняется ли напряжение питания со временем. Избыточный шум в блоке питания может вызвать проблемы, а системы,

использующие большое количество высокоскоростной логики КМОП, могут иметь очень требовательные сдвиги в потреблении тока, что может привести к коротким провалам и скачкам напряжения питания.

7. **Сломанные или поврежденные дорожки печатной платы.** Это может быть проблемой, если вы вручную собирали плату и у вас возникли проблемы с присоединением компонента. Избыточное тепло во время сборки может привести к отрыву дорожек от печатной платы. Кроме того, узнайте своего поставщика платы. Некоторые поставщики плат (особенно поставщики прототипов со скидкой на быстрые поставки) не будут проводить полный электрический тест списка соединений вашей печатной платы. Ищите перетравленные дорожки, которые истончились за пределы допуска, а также проверяйте, чтобы каждое сквозное отверстие имело серебристое кольцо вокруг отверстия. Иногда сверла смешены или наклонены во время сверления платы, и неправильно просверленное отверстие в конечном итоге нарушит электрические соединения.
8. **Проблема с защелкиванием или последовательностью включения питания.** Защелкивание — потенциально катастрофическое явление, при котором между питанием и землей внутри подложки микросхемы возникает паразитное короткое замыкание. Защелкивание запускается путем подачи тока в подложку. Это может произойти в системах со смешанным напряжением, где входные напряжения выше напряжения питания чипа. Во многих случаях защелкивание сопровождается перегревом чипа, что может привести к его необратимому повреждению. При первом включении системы рекомендуется использовать амперметр для контроля тока, потребляемого системой, и прикоснуться ко всем компонентам, чтобы проверить, не перегреваются ли какие-либо из них. Если компонент перешел в защелкивание, вы, как правило, будете наблюдать избыточное потребление тока порядка сотен миллиампер.
9. **Тепловая проблема.** Это проблема, в первую очередь, линейных регуляторов напряжения и мощные цифровые схемы. Убедитесь, что все мощные компоненты надлежащим образом отведены от радиаторов, а радиаторы надлежащим образом изолированы, когда они контактируют с электрически активной частью корпуса чипа.
10. **Непреднамеренное замыкание на оголенную медь.** Это проблема с разъемами и чипами, имеющими открытые области металла на нижней стороне, которые могут закоротить открытые области платы, такие как переходные отверстия. Это также проблема вокруг областей, где винты используются для удержания платы на месте. Головка металлического винта может непреднамеренно соприкоснуться с переходным отверстием, которое было размещено слишком близко к отверстию для винта.
11. **Загрязнение платы.** Эта проблема вызвана остатками флюса для пайки или другими остатками процесса на плате, что приводит к появлению путей утечки низкого тока. Некоторые остатки флюса имеют

существенное (менее одного мегаома) сопротивление, и это может вызвать проблемы с высокоомными цепями, такими как RC-цепи с медленной постоянной времени.

12. **Неисправное испытательное оборудование.** Это особенно проблема, если вы используете бывшее в употреблении или старое испытательное оборудование. Со временем испытательные щупы перегибаются и калибруются с ошибками, поэтому иногда плохой сигнал, который вы видите на осциллографе, на самом деле является результатом плохого испытательного щупа или плохого выбора заземления щупа. Откалибруйте свое испытательное оборудование на заведомо хороший сигнал, чтобы исключить проблемы с испытательным оборудованием.
13. Наименее вероятная проблема — это плохой чип или неисправный компонент. Производители компонентов прилагают большие усилия, чтобы гарантировать, что отправленные вам детали являются функциональными. Типичные показатели отказов измеряются в однозначных числах частей на миллион для простых и умеренно сложных деталей. Часто мы представляем, что причиной нашей проблемы является плохой чип от производителя, но это почти никогда не так. Обычно, если обнаруживается плохая деталь, она была повреждена либо из-за проблемы обработки (грубое обращение или проблемы сборки), либо из-за проблемы проектирования в другом месте схемы, которая вызывает наблюдаемый отказ.

## Восстановление после поднятия следа или подушечки

Поднятие или разрыв медных дорожек на печатной плате — распространенная проблема, с которой сталкиваются люди, пытающиеся установить модификации стороннего производителя с помощью летающих проводов. Это расслоение медной фольги обычно вызвано чрезмерным нагревом от паяльника. Другая распространенная причина — натягивание прикрепленного провода модификации, как это можно сделать при снятии изоляции с конца провода после того, как он был припаян к печатной плате. К счастью, обычно эту проблему довольно легко устранить. Совет



**Лучшее решение — профилактика.** Не используйте слишком мощный паяльник для работы с печатными платами. Предпочтительнее паяльник с регулируемой температурой, но подойдет и недорогой паяльник малой мощности (15 Вт). Кроме того, если припой не прилипает к плате, прекратите подачу тепла. Вместо этого нанесите немного флюса на плату и провод и очистите жало паяльника кондиционером для жал или губкой, смоченной дистиллированной водой (водопроводная вода содержит химикаты, которые могут ухудшить качество жала паяльника). Это улучшит паяемость, поэтому вам не придется прилагать слишком много тепла или силы для соединения.

Первое, что нужно сделать, когда вы видите, что дорожка или контактная площадка отслаиваются от платы, — ОСТАНОВИТЬСЯ! Не усугубляйте проблему еще больше; худшее, что вы можете сделать, — это заставить всю дорожку отслоиться, продолжая тянуть

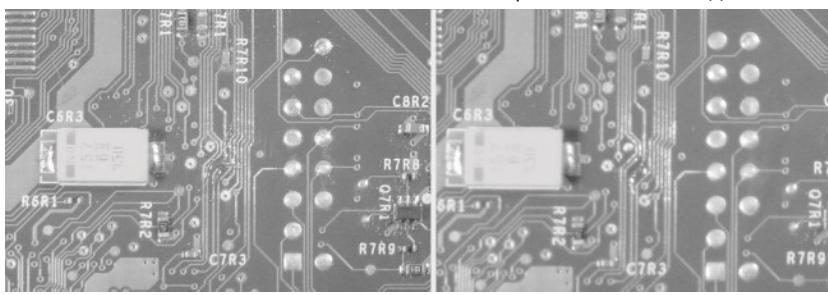


**Рисунок E-1:** Слева, стрелка указывает на исходную площадку, которая паяется. Справа, площадка была оторвана из-за избыточного тепла и силы.

провод. Удалите провод, если он все еще подключен, слегка коснувшись паяльником места соединения и дав проводу упасть. Рисунок E-1 иллюстрирует такую сцену катастрофы.

Стратегия восстановления поврежденной дорожки заключается в удалении паяльной маски, фиксации дорожки с помощью перемычки и поиске альтернативной точки пайки, следя по дорожке до ближайшего компонента или переходного отверстия.

Удаление паяльной маски обнажает нижележащие медные дорожки. К этим оголенным дорожкам можно припаять короткую перемычку, чтобы устраниить разрыв, вызванный разорванной дорожкой. Оголенная область также служит удобной отправной точкой для использования измерителя целостности, чтобы найти альтернативную точку для крепления перемычки. Удалите паяльную маску с помощью мелкозернистой (200 или мельче) наждачной бумаги или скребая поверхность острым ножом для хобби. При удалении паяльной маски будьте осторожны, чтобы не зацепить части сломанной дорожки и не разорвать дорожку платы еще больше. После удаления паяльной маски очистите область мягким растворителем, например спиртом, с помощью ватного тампона. Затем нанесите очень тонкий слой паяльного флюса на область и протрите чистым наконечником паяльника открытые дорожки. Небольшое количество припоя, прилипшее к наконечнику паяльника, попадет на печатную плату и покроет дорожки, предотвращая окисление открытой меди. Если наконечник паяльника слишком чистый, нанесите на него каплю припоя и слегка протрите наконечник влажной губкой и попробуйте еще раз. Не пытайтесь залудить открытые дорожки шариком расплавленного припоя на наконечнике. Избыток припоя будет осаждаться, что может привести к



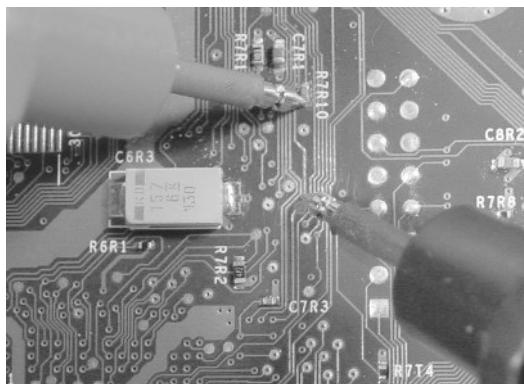
**Рисунок Е-2:** Слева, область после удаления паяльной маски мелкозернистой наждачной бумагой. Справа, область после лужения (восстановления для пайки).

короткие замыкания. (Обратите внимание, что флюс для пайки необходим для получения равномерного тонкого покрытия припоям на дорожках. Не пропускайте нанесение флюса для пайки.) На рисунке Е-2 показано, как будут выглядеть дорожки до и после процесса лужения.

На этом этапе вы можете захотеть использовать измеритель непрерывности, чтобы определить альтернативную точку для присоединения вашего модификационного провода. Большинство вольтметров оснащены звуковой функцией измерителя непрерывности. При выборе этой функции вольтметр издает тон, когда сопротивление между зондами становится очень низким.

Хорошими альтернативными точками крепления являются как переходные отверстия, так и выводы компонентов.

Если вы решили использовать переходное отверстие, вам необходимо соскоблить паяльную маску и



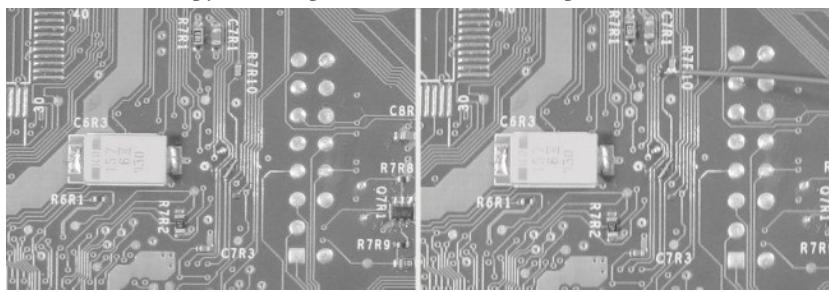
**Рисунок Е-3:** Использование измерителя непрерывности для поиска альтернативной точки присоединения. В этом случае R7R10 оказывается хорошей альтернативой.

подготовьте переходное отверстие перед присоединением провода модификации. На рисунке Е-3 показано использование измерителя целостности для поиска альтернативной точки пайки. Помните, что иногда

вам придется проследить через несколько переходных отверстий, чтобы найти лучшую альтернативную точку крепления.

Следующий шаг — прикрепить короткую перемычку через оборванную дорожку. Нанесите немного больше паяльного флюса на область оборванной дорожки. Отрежьте кусок тонкой проволоки (около 30 калибра), который примерно соответствует длине рассматриваемого зазора. Поместите провод над зазором, используя липкость паяльного флюса для облегчения процесса размещения. Удерживайте провод на месте с помощью пинцета и нагревайте паяльником, пока обе стороны не приклеятся к краям оборванной дорожки. Убедитесь, что провод на месте, осторожно нажав на него пинцетом; провод не должен двигаться. Также проверьте наличие коротких замыканий с соседними дорожками с помощью измерителя целостности цепи. Если обнаружено короткое замыкание, просто нагрейте перемычку, пока она не отвалится от платы, и попробуйте снова. На рисунке E-4 показано, как выглядит восстановленная дорожка.

Наконец, подсоедините модификационный провод к альтернативной точке пайки, обнаруженной ранее с помощью измерителя целостности цепи.



**Рисунок E-4:** Слева, перемычка установлена на поврежденную дорожку. Справа, модификационный провод успешно присоединен к альтернативной точке пайки.



## АПРИЛОЖЕНИЕ Ф

# Справочник по оборудованию Xbox

В этом приложении приведена схема расположения контактов основных разъемов, используемых в оборудовании Xbox.

## Распиновка блока питания

Блок питания, используемый в Xbox, представляет собой коммутатор с максимальным номиналом 96 Вт, с пиковой импульсной мощностью 160 Вт в течение менее 10 секунд. Microsoft покупает этот блок питания у нескольких поставщиков, включая Delta Electronics, Inc. ([www.deltaww.com](http://www.deltaww.com)). Delta используется в консолях Xbox в США, и вы можете найти техническое описание этой детали на их веб-сайте или с помощью поиска в Интернете.

Приколовы	Описание	Цвет провода
1	+12V	Желтый
2	+5V	Красный
3	+5V	Красный
4	+5V	Красный
5	+3,3V	Апельсин
6	+3,3 В в режиме ожидания	Коричневый
7	Земля	Черный
8	Земля	Черный

9	Земля	Черный
10	Земля	Черный
11	Включить питание	Белый
12	Питание в порядке	Синий

**Таблица F-1:** Распиновка основного разъема питания. Цвета проводов могут немного отличаться в зависимости от конкретной модели блока питания, используемого в вашем Xbox. Эта таблица применима к блоку питания Delta DPSN-96AP версии A.

Описание	Цвет провода
+12V	Желтый
Земля	Черный
Земля	Черный
+5V	Красный

**Таблица F-2:** Распиновка разъема питания жесткого диска.

## Распиновка видеоразъема

Распиновка видеоразъема немного загадочна, поскольку некоторые из его сигналов не показывают очевидных или узнаваемых шаблонов сигнала при зондировании, а также поскольку несколько режимов отображения поддерживаются одним разъемом. Есть несколько веб-сайтов, которые публикуют распиновку видеоразъема, но перекрестная проверка опубликованной информации с измерениями выявляет некоторые несоответствия. Я сделал все возможное, чтобы собрать воедино и согласовать два отдельных сообщения с XboxHacker BBS и веб-страницы ucon64 на Sourceforge.net. Оригинальные сообщения можно найти по адресу <http://www.xboxhacker.net/index.php?do=article&id=10&page=1> и на <http://ucon64.sourceforge.net/ucon64misc/conn.html>. Определение всех восьми видеорежимов, выбираемых сигналами MODE1-3, приведено на веб-странице XboxHacker BBS. Мои измерения показывают, что все сопоставления композитных видео- и аудиосигналов верны, но мне не удалось проверить сопоставления SDTV, HDTV и RGB, как указано в сообщении Xboxhacker BBS. Заранее извиняюсь, если какой-либо из этих сигналов неверен.

Обратите внимание, что контакты 12 и 24 имеют более длинные контакты на разъеме Xbox, что указывает на то, что они используются для подачи питания на периферийные устройства, подключенные к видеоразъему во время событий горячей вставки. Более длинные контакты позволяют схеме периферийного устройства включаться до получения сигналов в случае, если

периферийное устройство подключено, когда Xbox включен. Это помогает предотвратить потенциально разрушительную ситуацию внутри микросхем периферийного устройства, называемую защелкиванием.

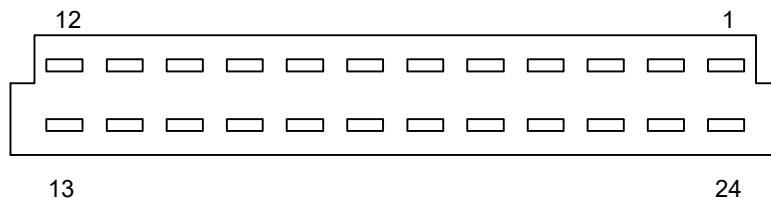


Рисунок F-1: Аудио-видеоразъем Xbox, вид на заднюю панель Xbox снаружи.

Приколоть	Сигнал Имя	Ввод/вывод	Комментарий
1	Правильный звук	Вне	Аудиовыход, правый канал
2	Земля	Власть	
3	SPDIF	Вне	Аудиовыход Sony/Philips Digital Interface (S/PDIF)
4	ВСИНХРОНИЗАЦИЯ	Вне	Вертикальная синхронизация (режим выхода VGA)
5	Земля	Власть	
6	Земля	Власть	
7	Земля	Власть	
8	Земля	Власть	
9	Свинцово-кислотный	Вне	Pb для режима HDTV, Blue для режима RGB
10	Земля	Власть	
11	Ж/Г	Вне	Y в режимах SDTV и HDTV, зеленый в режиме RGB
12	Земля	Власть	Имеет более длинные штырьки для горячей вставки
13	СВИДЕО	Вне	Композитный видеовыход.
14	Земля	Власть	
15	К / Пр / Р	Вне	C в SDTV, Pr в HDTV, Red в режиме RGB
16	Земля	Власть	
17	СТАТУС	Вне	PIN-код статуса SCART (Syndicat des Constructeurs)

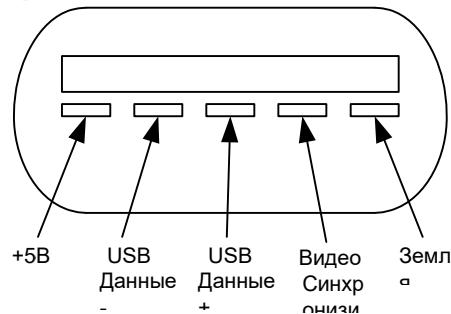
			d'Appareils Radio Récepteurs et Télésieurs)
18	РЕЖИМ3	В	Выбор режима видеовыхода, контакт 3
19	РЕЖИМ2	В	Выбор режима видеовыхода, контакт 2
20	РЕЖИМ1	В	Выбор режима видеовыхода, контакт 1
21	H SYNC	Вне	Горизонтальная синхронизация (режим выхода VGA)
22	Земля	Власть	Экран аудиокабеля левого канала
23	Левый аудио	Вне	Аудиовыход, левый канал
24	+5V	Власть	Питание +5 В, имеет более длинные контакты для «горячей» вставки

Таблица F-3: Распиновка видеоразъема.

Нумерация контактов, используемая в Таблице F-3 для видеоразъема Xbox, показана на Рисунке F-1. Как схема нумерации, полученная от Xboxhacker, так и схема нумерации, полученная от uscon64, расходятся, поэтому я выбрал схему нумерации, которая находится где-то посередине между ними. Интересно, что контакт 24 на Рисунке F-1 сопоставлен с квадратной площадкой на видеоразъеме Xbox, что указывает на то, что схема нумерации, которую я выбрал для этого разъема, не соответствует схеме нумерации производителя (квадратные площадки обычно обозначают контакт 1, а круглые — все остальные контакты). Это не должно повлиять на правильность таблицы, поскольку схемы нумерации контактов произвольны и должны соответствовать только таблице определения контактов.

## Распиновка разъема USB

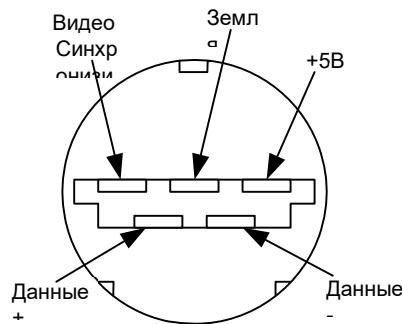
Xbox использует производную USB для портов игрового контроллера. На передней панели Xbox есть четыре порта игрового контроллера, и все они имеют одинаковую распиновку, как показано на рисунке F-2.



**Рисунок F-2:** Распиновка игрового контроллера, вид снаружи корпуса Xbox, если смотреть на разъем.

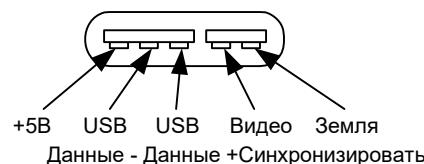
Сигнал «видеосинхронизации» — это 3,3-вольтовый КМОП- или TTL-совместимый сигнал. Это базовая последовательность импульсов положительной полярности частотой 15,734 кГц, синхронизированная с горизонтальным временем строки композитного видеовыхода, с одним более длинным импульсом в начале каждого видеополя. Этот сигнал позволяет периферийным устройствам, направленным на экран телевизора, таким как световое перо или световой пистолет для игр-стрелялок, получать информацию о местоположении.

Игровые контроллеры подключаются к Xbox через промежуточный разрывной разъем. Цель этого разрывного разъема — предотвратить повреждение консоли (в частности, повреждение жесткого диска) при перетаскивании или рывке в случае, если шнур запутается вокруг ноги пользователя. Распиновка этого разрывного разъема показана на рисунке F-3.



**Рисунок F-3:** Распиновка отсоединяемого игрового контроллера, вид спереди на разъем ближе к Xbox.

Игровой контроллер Xbox оснащен двумя слотами расширения для карт памяти, микрофонов и других периферийных устройств. Эти слоты также обеспечивают интерфейс, совместимый с USB. Игровой контроллер содержит USB-концентратор (chip концентратора Atmel AT43USB401), который повторяет входящий USB-сигнал в слоты расширения. Распиновка разъема расширения показана на рисунке F-4.



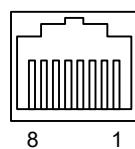
**Рисунок F-4:** Распиновка слота расширения игрового контроллера, если смотреть на слот игрового контроллера кнопками вверх.

## Распиновка разъема Ethernet

Порт Ethernet на Xbox — это стандартный разъем RJ-45 для витой пары 10/100 base-TX. Распиновка и цвета в Таблице F-4 соответствуют стандарту EIA/TIA 568B. На Рисунке F-5 показана нумерация контактов разъема.

Приколоть	Описание	Цвет провода
1	Передача +	Оранжевая полоса
2	Передача -	Апельсин
3	Получить +	Зеленая полоса
4	Не подключен	Синий
5	Не подключен	Синяя полоска
6	Получать -	Зеленый
7	Не подключен	Коричневая полоска
8	Не подключен	Коричневый

**Таблица F-4:** Распиновка Ethernet 10/100 RJ-45.



**Рисунок F-5:** Распиновка разъема Ethernet Xbox, вид снаружи на заднюю панель Xbox.

## Распиновка разъема ATA

Xbox использует стандартную шину Advanced Technology Attachment (ATA) для связи с жестким диском и DVD-приводом. Шину ATA обычно (но технически неправильно) называют шиной IDE (Integrated Drive Electronics). Большинство современных приводов квалифицируются как приводы IDE; например, приводы SCSI также оснащены встроенной электроникой привода. Однако годы

(неправильного) использования сделали термин IDE синонимом шины ATA.

Таблица F-5 показывает расположение выводов разъема ATA, как он виден на материнской плате Xbox, если смотреть на разъем сверху, а задняя часть Xbox должна быть обращена к наблюдателю (разъем должен быть справа). Обратите внимание на то, как нумерация выводов зигзагообразна: все нечетные выводы находятся на одной стороне, а четные — на другой.

Приколо ть	Имя	Комментарий	Приколо ть	Имя	Коммента рий
1	Перезагруз ить		2	Земля	
3	Данные 7		4	Данны е 8	
5	Данные 6		6	Данны е 9	
7	Данные 5		8	Данны е 10	
9	Данные 4		10	Данны е 11	
11	Данные 3		12	Данны е 12	
13	Данные 2		14	Данны е 13	
15	Данные 1		16	Данны е 14	
17	Данные 0		18	Данны е 15	
19	Земля		20	Ключ	Пустой штифт для поляризаци и
21	DMARQ	Запрос DMA	22	Земля	
23	DIOW-	Запись ввода- вывода	24	Земля	
25	DIOR-	Ввод/вывод Чтение	26	Земля	
27	ИОРДИ	Готовность к вводу/выводу	28	CSEL	Выбор кабеля
29	DMACK-	Подтверждение DMA	30	Земля	
31	ИНТРКВ	Запрос на прерывание	32	IOCS1 6-	16-битный ввод-вывод
33	ДА1	Адрес устройства, бит 1	34	PDIAG -	Пройдена диагностика

35	ДА0	Адрес устройства, бит 0	36	ДА2	Адрес устройства, бит 2
37	CS0-	Выбор чипа 0	38	CS1-	Выбор чипа 1
39	DASP-	Дев. Активный/Подчинен ный присутствует	40	Земля	

Таблица F-5: Распиновка разъема ATA.

## Разъем питания DVD-ROM

Xbox использует фирменный разъем питания DVD-ROM. Этот разъем не только обеспечивает питание, но и передает несколько сигналов управления и состояния. Эти сигналы передают информацию о состоянии привода и лотка привода. Распиновка, приведенная здесь, взята с Xboxhacker BBS, а оригинальный пост Кена Гаспера, на котором основана эта распиновка, можно найти по адресу <http://www.xboxhacker.net/forums/index.php?act=ST&f=5&t=1025&s=0755f2b600975b776552f93d0730e4b1>

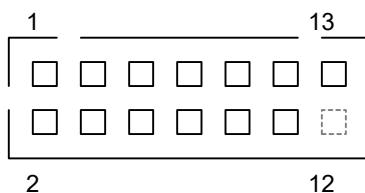


Рисунок F-6: Нумерация контактов разъема питания DVD-ROM, если смотреть сверху на материнскую плату Xbox.

Нумерацию контактов разъема, если смотреть сверху на материнскую плату Xbox, можно найти на рисунке F-6, а схему расположения контактов — в таблице F-6.

Приколо ть	Имя	Комментар ий	Приколо ть	Имя	Комментар ий
1	12 В постоянно го тока	+12 Вольт питания	2	5 В постоянного тока	+5 Вольт питания
3	Земля	Текущая доходность, ссылка	4	EJECT-	Активный выброс нижнего лотка
5	TC0	Состояние Traystate 0	6	TC1	Состояние Traystate 1

7	TC2	Состояние Traystate 2	8	АКТИВНОСТЬ-	Поиск на диске/передача данных
9	12 В постоянно го тока	+12 Вольт питания	10	5 В постоянного тока	+5 Вольт питания
11	Земля	Текущая доходность, ссылка	12	Земля	Текущая доходность, ссылка
13	Ключ	Не подключен			Заготовка для поляризации

Таблица F-6: Распиновка разъема питания DVD (на материнской плате).

## LPC-разъем

Xbox имеет порт отладки и тестирования на основе шины LPC (Low Pin Count). Эта шина была изначально определена Intel для использования с чипсетами Southbridge, чтобы уменьшить количество контактов, тем самым экономя на стоимости, сохраняя поддержку устаревших функций ввода-вывода ПК. Эти устаревшие функции ввода-вывода раньше располагались на почти исчезнувшейшине ISA, и они включают клавиатура, мышь, последовательный порт, параллельный порт и загрузочное ПЗУ. Спецификацию Intel для шины LPC можно найти по адресу <http://www.intel.com/design/chipsets/industry/25128901.pdf>

{не дает скачать этот файл}

Отладочный разъем LPC особенно важен, поскольку его можно использовать для подачи альтернативного образа ПЗУ на Xbox в случае, если встроенный ПЗУ отсутствует или поврежден таким образом, что ПЗУ кажется отсутствующим или пустым. Эта функция может быть использована и использовалась для создания легко устанавливаемого альтернативного загрузочного ПЗУ для Xbox.

Распиновка отладочного разъема Xbox LPC, по-видимому, основана на руководстве по проектированию устанавливаемого отладочного модуля LPC от Intel, [http://www.intel.com/technology/easeofuse/LPC\\_mod\\_spec72.pdf](http://www.intel.com/technology/easeofuse/LPC_mod_spec72.pdf),

{нет уже такого файла}

с некоторыми незначительными изменениями, как указано в Таблице F-7. В частности, функция контакта 16 неясна, поскольку его сопутствующий контакт 15 был переназначен на контакт питания на материнской плате Xbox. Назначение контакта 15 как контакта питания выводится по толстой дорожке и расположенному рядом развязывающему конденсатору, выделенному для контакта. Если бы контакт 15 предназначался для использования в качестве постоянно

высокого сигнала SPDA1, то использовался бы более узкий контакт без нормализации питания.

<b>Приколовть</b>	<b>Имя</b>	<b>Комментарий</b>	<b>Приколовть</b>	<b>Имя</b>	<b>Комментарий</b>
1	ЛКЛК	Тактовая частота 33 МГц	2	VCC	Текущая доходность
3	LFRAME#	Начало, конец транзакций LPC	4	ШПОНКА	Заготовка для поляризации
5	LRST#	Сброс LPC	6	VCC5	+5В питание
7	ЛАД3#	Мультиплексированный адрес/данные	8	ЛАД2#	Мультиплексированный адрес/данные
9	VCC3	+3,3В питание	10	ЛАД1#	Мультиплексированный адрес/данные
11	ЛАД0#	Мультиплексированный адрес/данные	12	VCC	Текущая доходность
13	СКЛ	Последовательные часы I2C	14	ПДД	Последовательные данные I2C
15	VCC3	+3,3В питание (было SPDA1 в спецификации Intel.)	16	SPDA0	Выбор адреса для последовательного устройства EEPROM (?)

Таблица F-7: Распиновка разъема LPC (на материнской плате).

## Разъем вентилятора

Разъем для вентилятора в Xbox представляет собой трехконтактный разъем, где контакты 1 и 3 подключены к терморегулирующему контроллеру скорости вращения вентилятора с широтно-импульсной модуляцией (ШИМ), а контакт 2 подключен к источнику питания +12 В.

Разъем передней панели. Функции передней панели Xbox, а именно мигающий светодиод, выключатель питания и выключатель выброса, подключены к материнской плате Xbox через разъем передней панели. Распиновка этого разъема приведена в Таблице F-8. Распиновка отражает нумерацию контактов разъема на материнской плате Xbox, как видно при взгляде на разъем сверху.

<b>Приколовть</b>	<b>Комментарий</b>	<b>Приколовть</b>	<b>Комментарий</b>
1	Земля	2	Выключатель питания
3	Земля	4	Переключатель выброса
5	Зеленый светодиод	6	Красный светодиод
7	Красный светодиод	8	Зеленый светодиод

9	Не подключен, но проводной	10	Нет штифта (поляризация)
---	-------------------------------	----	-----------------------------

**Таблица F-8:** Разъем передней панели (если смотреть на материнскую плату).



