

**PENTAGON 2.666**

история задумки, реализации,  
воплощения в жизнь и планы  
на будущее.



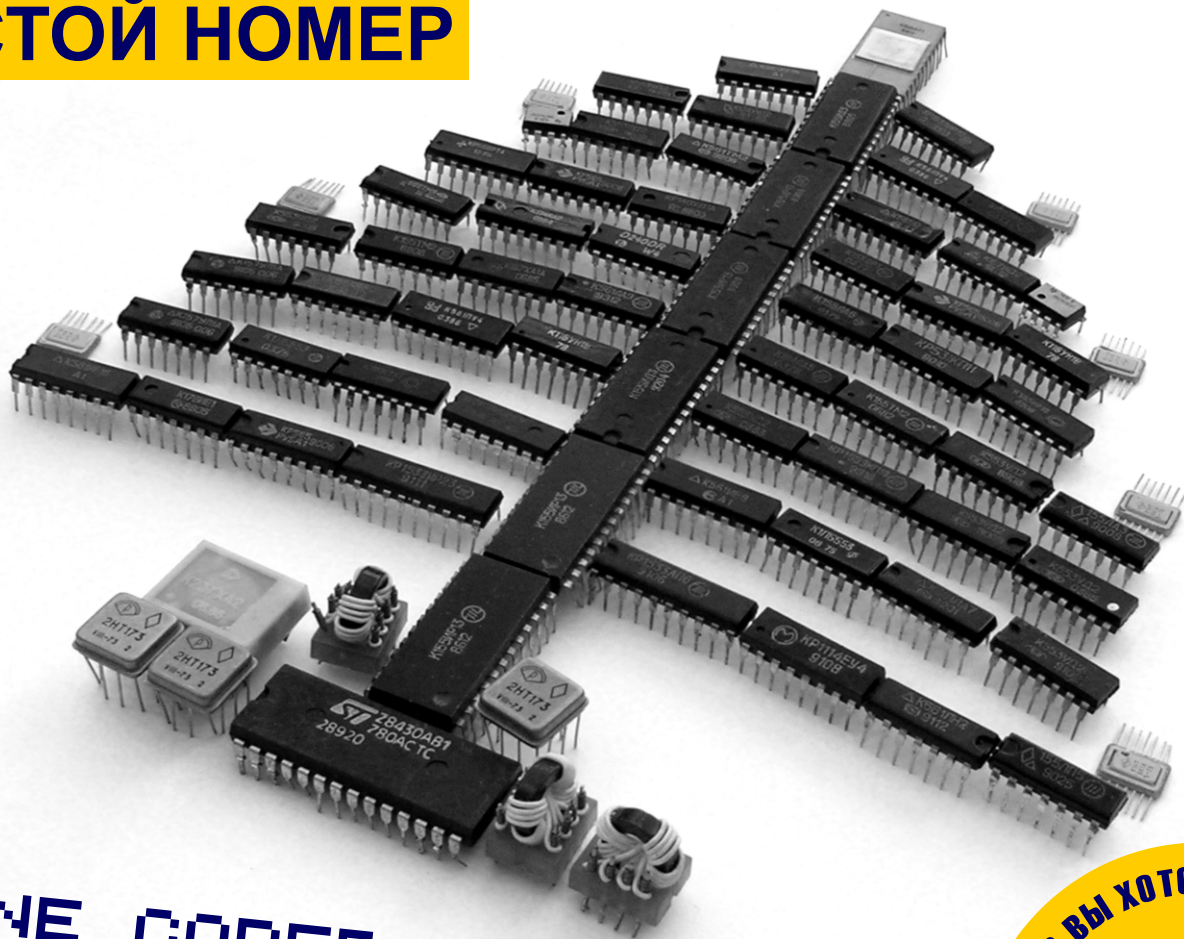
# The magazine

# NedopPC

NedoPC is not PC

Декабрь 2011

**ШЕСТОЙ НОМЕР**



## ALONE CODER

Использование NeoGS в качестве акселератора

...спросить программиста, чего больше всего не хватает Спектруму, он скорее всего ответит: скорости. Да, мы уже привыкли к чуть менее чем 1 MIPS и проблеме заполнить весь экран за фрейм.

**Если...**

### Время собирать камни

Первый умерший на руках компьютер... Казалось бы — хорошего мало, однако именно тогда во мне проснулся интерес к процессорам, толкнувший меня на путь их коллекционирования.

### USA — рай для электронщика

В любом маломальском городе США есть магазин RadioShack. Там наряду с обычной бытовой электроникой и компьютерной техникой продаются детские электронные игрушки и конструкторы, несколько вариантов паяльников...



 **WWW.NEDOPC.COM**  
больше информации  
на нашем сайте

## Содержание номера

|  |    |
|--|----|
| <b>Колонка редактора</b><br>CHRV ( <a href="mailto:chunin@mail.ru">chunin@mail.ru</a> )  | 3  |
| <b>Время собирать камни</b><br>Александр ( <a href="mailto:info@silirium.ru">info@silirium.ru</a> )  | 4  |
| <b>USA — рай для электронщика</b><br>Shaos ( <a href="mailto:shaos@mail.ru">shaos@mail.ru</a> )  | 8  |
| <b>Использование NeoGS в качестве акселератора</b><br>Alone Coder ( <a href="mailto:dmitry.alonecoder@gmail.com">dmitry.alonecoder@gmail.com</a> )                               | 13 |
| <b>FAT на Спектруме. Теория.</b><br>Savelij ( <a href="mailto:asavelij13@yandex.ru">asavelij13@yandex.ru</a> )   | 15 |
| <b>“PENTAGON” 2.666 / 2.666le: история задумки, реализации, воплощения в жизнь и планы на будущее.</b><br>KOE ( <a href="mailto:KingOfEvil@yandex.ru">KingOfEvil@yandex.ru</a> ) | 22 |
| <b>Почему “ZX Evolution” ?</b><br>CHRV ( <a href="mailto:chunin@mail.ru">chunin@mail.ru</a> )  | 26 |

Номер подготовлен московским отделением NedoPC ([www.nedopc.com](http://www.nedopc.com)) в декабре 2011.  
Вопросы и отзывы по номеру 6 журнала пишите на адрес: [nedopc@mail.ru](mailto:nedopc@mail.ru) .

Верстка журнала осуществлялась в операционной системе Windows с помощью системы подготовки текстов OpenOffice.Org и графического редактора Paint.Net.  
Первая страница подготовлена Breeze в программе CorelDraw.

Журнал распространяется бесплатно в бумажном виде (A5) и электронном виде (pdf).

Вопросы по журналам и их распространению пишите по следующим адресам:  
E-Mail: [journal@nedopc.org](mailto:journal@nedopc.org), [nedopc@mail.ru](mailto:nedopc@mail.ru) Сайт: <http://journal.nedopc.org>

## ***Колонка редактора***



Очередной номер не привязан к каким-то событиям. Просто накопилась “критическая масса” материала и пора ее оформить в журнальный вид.

Как и в прошлый раз, номер выпускается силами Московского отделения NedoPC. К сожалению, с каждым разом, все сложнее найти свободное время для творчества.

Статьи из этого номера написаны в разное время и не объединены какой-то общей тематикой. Радует, что у нас появились новые авторы. Они познакомят нас со своими не менее захватывающими увлечениями и интересами.

Надо отметить, что тематика ретрокомпьютеров и событий вокруг них охватывает все большую аудиторию. Участники нашей команды активно сотрудничают со СМИ. Видеорепортажи о фестивале ArtField были показаны в том числе и по государственным каналам. Статья о нашей группе размещена на сайте

онлайн-журнала «Компьютерра».

Приятно, что наш проект «ZX-Evolution» положительно принят ZX сообществом и в кратчайшие сроки стал очень популярным. К сожалению, ресурсы группы не позволяли без задержек поставлять платы всем желающим. Я еще раз хочу от себя лично принести извинения за то, что многим приходилось ждать свой экземпляр платы по несколько месяцев. Сейчас производится новая ревизия платы, меньшего размера (miniITX) под современные корпуса.

Также на стадии обдумывания находится еще один интереснейший проект. Но пока я воздержусь от его анонсирования. Как говорится — чтобы не сглазить.

Наша группа по прежнему старается поддерживать фестивали и демопати. К сожалению, конкурс «Твоя игра» скорее всего прекратил свое существование в старом формате. Надеюсь, что все таки найдется организатор, который сможет потянуть подобный конкурс, ну а мы ему, конечно же, поможем.

Успехов и до будущих встреч.

***Редактор номера CHR.V***

## Время собирать камни



Автор: Александр

Email: [info@silirium.ru](mailto:info@silirium.ru)

### Как всё начиналось.

Первый умерший на руках компьютер... Казалось бы — хорошего мало, однако именно тогда во мне проснулся интерес к процессорам, толкнувший меня на путь их коллекционирования. Когда после очередной бесплодной попытки воскресить машину я держал в руках Интеловский 486-ой — чем-то он меня заворожил. Это и был первый процессор в коллекции - Intel A80486DX2-66 SX911, вроде бы малоинтересная и распространённая «четвёрка», однако для меня он, наверное, один из самых ценных.

На тот момент желание ещё не оформилось в энтузиазм, потому коллекционирование было вялотекущим — несколько экземпляров наиболее распространённой архитектуры x86, вся коллекция составляла десяток-другой чипов, бессистемно сваленный в кучу. В то же время я всегда испытывал интерес к процессорным архитектурам, отличным от x86, и по мере сил изучал доступные материалы, тайком мечтая заполучить что-нибудь экзотическое в своё собрание.



Так продолжалось где-то до конца 2006-го года, когда интерес вспыхнул с новой силой,

да и возможности стали позволять серьёзней заниматься коллекцией. Стоило взяться за дело более осмысленно — стали появляться интересные экземпляры, вроде ныне покойного семейства RISC-процессоров Intel 960 и их конкурента AMD Am29000, первого Intel Itanium и т.п. Хотя, конечно, закономерно, что большая часть поступлений всё же была x86.

С ростом собрания встал вопрос учёта и описания хранимых богатств. Поначалу хватало простенькой локальной системы учёта, однако со временем это становилось всё более неудобно. Да и с объёмом коллекции около 150 экземпляров (на август-сентябрь 2007-го года) уже было с чем показаться на люди. К тому времени я уже просмотрел множество зарубежных сайтов, посвящённых той же тематике, однако российских среди них не было. Было решено делать свой сайт.

### Выход в свет.

Так получилось, что знакомые мне на тот момент российские единомышленники (которых оказалось не так уж много) обращались в основном в среде любителей старых компьютеров (к которым в какой-то мере отношусь и я). Мне показалось важным, чтобы по возможности сайт был простым и максимально лёгким, и при его просмотре на устаревшем оборудовании/ПО было бы как можно меньше затруднений. Однако полностью текстовый ресурс вряд ли радовал бы глаз, да и протокол gopher давно почил, поэтому выбор был прост — практически чистый HTML. Ну, конечно, ещё немного CSS и SSI. С нуля было писать лень, благо в Сети предостаточно бесплатных шаблонов, одним из которых я и воспользовался. Шаблон пришлось изрядно перекроить, но результатом я остался доволен.

Конечно, использовать какую-нибудь готовую CMS (Content Management System) было бы удобнее. Однако в ручном наполнении такого вот «олдскульного» сайта

есть что-то медитативное: вёрстка - текстовый редактор, отладка - персоналка с Линуксом (для лучшей совместимости, т.к. хостинг линуксовый), загрузка на сайт - по FTP. Ну и есть ещё пара банальных доводов в пользу такого решения - дешевизна (хостинг с PHP/SQL дороже) и безопасность (взлом статического HTML-сайта - дело, как правило, заведомо бестолковое).

С названием пришлось крепко подумать, совсем уж банальности не хотелось, диких аббревиатур тоже, в зоне .ru домен Silicium (кремний) оказался занят сквоттерами, и как то так родилось созвучное и загадочное имя сайта **Silirium.ru**, на котором и решено было остановиться. И так, 16 августа 2007-го года и появился на свет мой виртуальный музей.

Думаю, техническую сторону развития сайта описывать смысла не имеет, концептуально сайт сильно не менялся — мелкие оптимизации, новые разделы и т.п.

Но всё же непосредственно с самим сайтом связана одна забавная история. Дело было так.

Иногда попадаются чипы, по которым информации крайне мало. Обычно если сразу не удалось найти какую-то информацию по новому чипу — я его добавляю на сайт с тем минимумом, что всё-таки есть. Но периодически проглядываю Интернет, вдруг удастся обновить информацию. И вот как-то в один из таких «рейдов» наткнулась на цельнотянутую копию своего сайта! Причём, сайт перекурочен чуть менее, чем полностью - половина копирайтов моя, половина какого-то товарища, фотки тянутые все целиком и полностью. Судя по информации о домене сайт оказался где-то в США. Пришлось вспоминать английский и писать гневное письмо. Ответ пришёл на удивление быстро, в нём оказались извинения, объяснение, что это был только «эксперимент» и обещание сделать сайту полный «take down».

К его чести обещание он сдержал - «take down» произошёл очень шустро. Что интересно, этот случай произошёл меньше чем через 4 месяца после публикации сайта — видимо, ресурс стал широко известен в

узких кругах (по крайней мере, приятно так думать). Ну а я после этого стал вешать копирайт на фотографии.

### А зачем?

Теперь, если я ещё не утомил читателя, хотел бы остановиться подробнее на предмете коллекционирования. Все-таки, несмотря на возрастающую у нас популярность этого увлечения, для большинства оно не совсем понятно.

Многие люди, когда слышат о коллекции процессоров, сразу с хитрым прищуром интересуются: «А зачем тебе вообще всё это? Ладно, там монеты, банкноты, значки какие-нибудь. Небось планируешь лет через 30 продать как антиквариат?» Ответ тут простой и сложный одновременно. Что будет через N лет - неизвестно, а сейчас это, как и любое другое хобби — для души. Считаю, что это ничем не хуже, например, собирания марок, монет и т.д. Правда, моё личное мнение тут ещё в том, что в отличие от монет и т.п. каждый процессор, как устройство крайне сложное и высокотехнологичное, является как бы таким маленьким памятником человеческому таланту, его породившему.

К тому же, я позиционирую своё собрание не только как коллекцию (которые зачастую имеют тематическую направленность), но и как виртуальный музей, т.е. я стараюсь охватить всё многообразие типов микропроцессорной техники и архитектур. Да и некоторые крайне интересные вещи порой малоизвестны у нас, потому приятно осознавать, что хотя бы так сохраняется и становится более доступной немаловажная часть компьютерной истории.

Поэтому в моём собрании можно встретить как процессоры общего назначения, так и специализированные — от математических сопроцессоров, микроконтроллеров и однокристалльных ЭВМ до DSP (процессоров цифровой обработки сигналов) и микропроцессорных наборов, физически состоящих из нескольких микросхем.

Правда, МК или ОЭВМ стараюсь не очень увлекаться — их выпущена огромная масса, что сводит на нет прелесть

коллекционирования. Поэтому основной упор делается всё же на процессоры общего назначения и интересные экземпляры. Исключение делаю иногда для отечественной техники, всё-таки советское наследие считаю необходимым сохранить в как можно большей мере.

Особый интерес представляют вещи не массовые (как впрочем в любой коллекции). В данном разрезе времени это, например, практически всё, что построено по архитектуре, отличной от Intel x86, или какие-то тупиковые ветви развития, как например Intel Overdrive, или вообще что-то экзотическое. Кстати, следует разделять действительно редкие экземпляры (инженерные образцы, Pentium II Overdrive, DEC PDP-11 и т.п.) и экземпляры труднодоступные в силу своей цены (например, относительно свежие Sun UltraSparc, DEC/Compaq/HP Alpha, Intel Itanium и т.д.).

В качестве иллюстрации и чтобы было понятнее, в чём тут может быть интерес, приведу несколько наиболее любопытных примеров из своей коллекции.

Многие ли слышали, например, про микропроцессор Ferranti F100L, производимый в Великобритании с 1976-го года? А между тем, по некоторым источникам информации это один из первых (если не первый) 16-битный процессор (к тому же разработанный и произведённый в Европе). Основными его областями применения были военная и аэрокосмическая.

Или так называемые «транспьютеры», разработанные в 1984-ом году компанией Inmos и казавшиеся перспективным направлением в своё время. Основная их идея заключалась в параллельной обработке данных большим количеством специализированных вычислительных элементов — собственно транспьютеров (которые, кстати говоря, оснащались наборной RAM и в последних поколениях ещё и FPU), соединенных между собой межпроцессорными соединениями (до 4-х

связей с другими параллельно работающими транспьютерами).



Крайне интересны также разработки DEC (Digital Equipment Corporation) - легендарные PDP-11, VAX и Alpha. В моём собрании, например, есть несколько процессоров CVAX и сопроцессоров CFPA. Особо примечательно то, что CVAX был первым в индустрии микропроцессором, использующим интегрированный кэш первого уровня для инструкций и данных (on-chip L1 cache). Кстати, пара CVAX/CFPA попались мне в составе работающей (!) машины MicroVAX 3100, которую удалось совершенно случайно приобрести за смешные деньги.



Любопытные разработки были у компании Weitek, специализировавшейся на разработке и выпуске математических сопроцессоров. Причём, сопроцессоры выпускались не только для ЦП x86, но и для, например, Sun Sparc. Как пример - в их сопроцессоре 3167 (Abacus) для i386 применялась собственная система команд (не x87!) и устанавливался он в отдельный сокет на материнской плате — т.н. Enhanced Math Coprocessor (EMC) socket.



Иногда и известные, казалось бы, компании преподносят сюрпризы. Взять ту же AMD. Так, например, удалось заполучить весьма любопытный AMZ8002DC - нечасто встречающийся клон Zilog Z8000 производства AMD (пользуясь случаем — ещё раз спасибо, Роман!). Или микропроцессорный набор серии Am29300 — набор микросхем, предназначенный для построения специализированных 32-битных компьютеров, состоящий из ALU, FPU, микропрограммного секвенсора и другой обвязки. Что любопытно — в Советском Союзе был аналог (или клон, теперь уже трудно разобрать) — серия 1843. Ну и более известные RISC-процессоры семейства Am29000, идеи которых позднее перекочевали в AMD K5.

Intel тоже порой выпускала интересные устройства. Помимо достаточно известных не-x86 (i960 и IA-64 (Itanium)) и до-x86 (4004/4040) есть диковины и среди x86-процессоров. Например, одна из жемчужин коллекции — i487SX, который был не сопроцессором, как можно было бы подумать, а фактически полноценным i486DX с FPU, предназначенным для замены i486SX. Стоит упомянуть и последний из Овердрайвов — Pentium II Overdrive, представляющий собой Pentium II под Socket 8 для апгрейда систем на Pentium Pro.

Да и среди советских изделий порой попадаются очень интересные вещи. Так, например, в коллекции есть уникальный Л1839ВМ1 (наш аналог VAX-11), исполненный в розовом керамическом корпусе (обычное исполнение — белая керамика). Насколько я знаю, этот экземпляр — пока единственный известный.

В общем, разнообразие впечатляет. Мое

технарское «Я» говорит, что это даже интереснее всех привычных объектов коллекционирования.



#### Заключение.

Обычно в заключительной части говорят о планах. Но определённый план тут вряд ли может быть, всё-таки это занятие для души. А вот чего пока не планируется — так это останавливаться на достигнутом. И быть может, когда нибудь мой виртуальный музей превратится в реальный.

Надеюсь, что из этой статьи читатель увидел, насколько увлекательным и познавательным может быть подобное хобби. К тому же, за то время, что я всерьёз этим занимаюсь, я успел познакомиться со многими замечательными людьми, которые зачастую безвозмездно помогали или помогают мне в пополнении собрания, за что им особая благодарность. Буду признателен, если и читатели окажут посильную помощь.

В заключение - хочу поблагодарить читателей за внимание и редакцию за публикацию материала. Успехов!

## USA — рай для электронщика



Автор: Shaos

Email: [shaos@mail.ru](mailto:shaos@mail.ru)

В этой статье я постараюсь осветить вопросы пребывания российского радиолюбителя (не профессионала) на территории США, а также плюсы и минусы с этим связанные. В этой статье НЕ будут затрагиваться вопросы попадания на территорию США, способы остаться (легальные или нелегальные) и, в частности, история того, как лично я туда попал и остался - это банально и неинтересно.

### Комплектующие.



Рисунок 1: "Бредборд"

Начнём с обычных магазинов. В любом маломальском городе США есть магазин RadioShack (см. <http://www.radioshack.com><sup>1</sup>). Там наряду с обычной бытовой электроникой и компьютерной техникой продаются детские электронные игрушки и конструкторы, несколько вариантов паяльников (от самых простых до некоторого подобия паяльной станции), припой (обычный и серебрянный), канифоль (твёрдая), разнообразные макетные платы (для пайки или "бредборды"), корпуса для самодельных устройств, блоки питания и кое-какие электронные компоненты в розницу (большие и маленькие трансформаторы, резисторы, конденсаторы, диоды, транзисторы, светодиоды, микросхемы типа регуляторов напряжения, операционников, компараторов). Плюс этих магазинов в том, что они всегда где-то рядом.

<sup>1</sup> <http://www.masterkit.ru> - российский магазин электронных конструкторов, игрушек и т.п.

Минус - там всё стоит относительно дорого. Например, пользуясь только деталями, продающимися в RadioShack, можно собрать двухполярный блок питания +5/-5 вольт (см. [br.jpg](#)), но на это придётся истратить порядка полсотни долларов. И ещё все местные магазины берут налог с продаж - почти 8% от суммы покупки (процент зависит от штата и города).

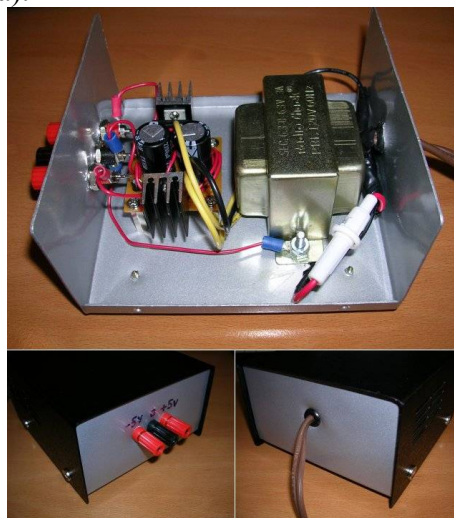


Рисунок 2: Блок питания

Если хочется иметь большой выбор и дешёвую цену, то надо идти в интернет. В случае онлайн покупок как правило не берётся налог с продаж (если продавец физически находится в другом штате), но оплачивается пересылка, которая может быть очень дешёвая (иногда даже бесплатная - т.е. за счёт продавца). Книгу или диск обычной наземной почтой (идёт около недели) можно отправить в любую точку США менее чем за 3 доллара. Пухлый конверт или небольшая посылка потребуют порядка 10 долларов. Посылка побольше типа системного блока или монитора потянет уже на 30-40 долларов. Есть варианты ускоренной доставки - на второй день или на следующий день (не считая выходных) - этим уже курьерские службы занимаются (как правило UPS или FedEx). Варианты оплаты интернет покупок (в порядке убывания популярности) - кредитные карты (MasterCard, Visa, American



Express), служба онлайн-платежей PayPal, почтовый перевод (так называемый Money Order - можно приобрести в любом почтовом отделении или банке), чек (самый долгий способ оплаты - до недели на пересылку плюс неделя на обработку и перевод средств с банковского счёта). Как правило предпочитают "мгновенные" способы оплаты - кредитки или PayPal (через который также и кредитками можно платить).

Крупных онлайн-магазинов электронных компонентов, которые с радостью работают с частниками, на территории США не так уж и много - я систематически пользуюсь [jameco.com](http://jameco.com), [digikey.com](http://digikey.com) и [mouser.com](http://mouser.com)<sup>2</sup>. Кроме этого существуют масса мелких онлайн-магазинов, многие из которых имеют своё представительство в системе онлайн-аукционов [ebay.com](http://ebay.com). Как правило у серьёзных продавцов имеется бесплатный телефон вида 1-800-xxx-xxxx, по которому всегда можно позвонить при возникновении проблем или при желании вернуть товар (что как правило делается очень безболезненно и без лишних вопросов).

Компания "Jameco Electronics" (см. <http://www.jameco.com>) территориально расположена в штате Калифорния. Семь лет назад этот магазин был одним из немногих онлайн-продавцов электронных компонентов в розницу, которые могли заранее посчитать стоимость пересылки при выбранном способе доставки. Позже он стал поступать как все - высчитывать стоимость пересылки только по факту пересылки, т.к. цена может сильно варьироваться в зависимости от веса посылки и её размеров и точная цифра будет известно только в момент отправки. [Jameco.com](http://jameco.com) торгует практически всем диапазоном электронных компонентов, который можно себе представить - причём цены в случае покупки единичных экземпляров самые выгодные (чем больше единиц покупать, тем будет дешевле). При

поиске в таблице результатов кроме наименования показаны изображения компонентов, что сильно упрощает задачу. Почти все компоненты имеют даташиты в формате PDF, которые можно скачать прямо с сайта. Также у них есть раздел "clearance" где за несколько центов можно приобрести вполне рабочие микросхемы мелкой логики и много чего другого. Ещё у них есть готовые наборы (резисторы, конденсаторы, микросхемы) и экзотические вещи типа мешочка с сюрпризом за \$17.95, в который они складывают более сотни разнообразных старых микросхем (в основном мелкая логика TTL), а также разъёмы, индикаторы, светодиоды, конденсаторы, подстроечники и т.д. Причём, покупатель узнаёт что же он купил только в момент распаковки посылки! Интересно то, что они до сих пор торгуют всяким старьём типа 6502 и 68000. Также они бесплатно рассылают свой каталог, состоящий из 200-300 страниц.



Рисунок 3: Содержимое "мешочка с сюрпризом"

Компания "Digi-Key Corporation" (см. <http://www.digikey.com>) территориально расположена в штате Миннесота. По большей части они ориентируются на профессионалов, однако с частниками тоже работают. Поиск позволяет выбирать характеристики для фильтрации по нескольким полям - очень удобно. Однако результаты поиска показаны только в текстовом виде - чтобы увидеть изображение компонента (между прочим в высоком разрешении), надо зайти в страницу описания компонента и кликнуть на соответствующий линк. Даташиты также доступны для скачивания почти по каждой позиции. Номенклатура продаваемого у них

<sup>2</sup> <http://www.voltmaster.ru>, <http://www.chipdip.ru> – российские поставщики, работающие в розницу и через почту РФ.

намного шире, чем у Jameco, однако цена как правило выше (если покупать небольшое количество). Кроме электронных компонентов они продают демоплаты и киты для разработки практически для всех микроконтроллеров, присутствующих на рынке. Они также рассылают свой бесплатный каталог, имеющий более 2000 страниц!

Компания "Mouser Electronics" (см. <http://www.mouser.com>) территориально расположена в штате Техас. По номенклатуре и цене они приближаются к DigiKey, однако цены, как правило, чуть ниже. Поиск не очень удобный - в описании компонентов встречается некорректная информация, зато в результатах поиска есть картинки (правда мелкие). По объёму их бесплатный каталог также пытается догнать DigiKey - чуть меньше 2000 страниц. Большим плюсом сервиса считается возможность создавать "проекты" - пользовательские списки компонентов, которые можно открывать в публичный доступ с сайта mouser.com и делиться с другими.

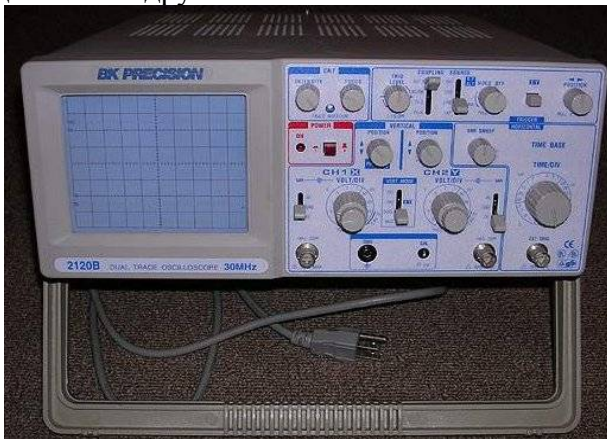


Рисунок 4: Осциллограф 30МГц

Радиодетали также можно покупать на <http://www.ebay.com><sup>3</sup>, однако велика вероятность нарваться на жуликов либо на заокеанских продавцов - пересылка может быть долгой, дорогой и ненадёжной. А ещё на ebay.com можно покупать всякое старьё, причём дёшево и рабочее! Например в 2004 году я купил там подержанный (но вполне рабочий) двухлучевой аналоговый

осциллограф 30 МГц и это обошлось мне (включая доставку) в 93 доллара. А в 2007 году мне удалось купить отлично работающий хоть и старый (1997 года) ноутбук Toshiba Tecra (Pentium MMX 150MHz, 32MB RAM, 1.5GB HDD, CD-ROM) всего за 49 долларов (включая доставку) и следом внешний дисковод к нему всего за 14 долларов. Поставив туда FreeDOS я стал его использовать как интерфейс к программатору, подключаемому через LPT и имеющему программу управления для DOS (программатор кстати говоря я приобрёл в Jameco).

### Печатные платы

Теперь перейдём к разработке печатных плат. Самый популярный среди местных любителей электроники редактор печатных плат это Eagle (см. <http://cadsoftusa.com>). Eagle может работать в Windows, Linux (сборка для x86) и MacOS X (если кто не знает, то это операционная система для компьютеров Apple, которые достаточно популярны в США). Бесплатная версия Eagle позволяет создавать двухслойные платы размером до 100x80мм с одним листом схемы. При доплате (сейчас это 249 долларов за редактор плат и столько же за редактор схем) можно получить "стандартную" версию, которая умеет до 6 слоёв размером до 100x160мм с 99 листами схем (на каждую разрабатываемую плату). Профессиональная версия без ограничений уже будет стоить около полутора тысяч долларов. Для производства плат используется формат Gerber 274-X (плюс файл для сверления отверстий Excellon). Некоторые производители предоставляют свои файлы настроек для Eagle, чтобы герберы получались именно такими, которые они принимают (в смысле имён файлов, зеркалирования, совмещения слоёв и т.д.). Также случается, что производитель может принимать в производство непосредственно исходный файл платы в формате Eagle (но это скорее редкость). В последнее время стала расти популярность бесплатного редактора плат с открытыми исходниками PCB,

3 <http://molotok.ru> — российский интернет-аукцион.

который также умеет генерировать герберы и является частью пакета gEDA - я его смотрел, но платы полученные в нём пока заказывать не пробовал.

Теперь собственно о производстве. За 7 лет я попробовал трёх производителей плат, а именно: [expresspcb.com](http://expresspcb.com), [pcbexpress.com](http://pcbexpress.com) и [batchpcb.com](http://batchpcb.com)<sup>4</sup>. Основная их особенность - они не берут деньги за фотошаблоны, подготовку и т.д. (что по английски называется "tooling") и имеют простую и понятную систему ценообразования - что очень походит для людей, у которых электроника это только хобби.

Первый из вышеприведённого списка (см. <http://expresspcb.com>) размещает свою рекламу во всех радиолобительских журналах Америки каждый месяц - вот уже как минимум 7 лет он предлагает за 51 доллар (плюс доставка) сделать 3 платы размером 96x63мм - правда развести их надо в их собственном редакторе плат, который бесплатно скачивается с их сайта. Если у вас Windows и вам надо быстро сделать простую платку указанного размера, то это вполне подходящий вариант.

Далее рассмотрим <http://pcbexpress.com> - это сервис для быстрого производства (1-2 дня) небольших объёмов плат - до 100 штук, причём на входе они берут герберы, но также могут принять и исходник Eagle. Таблица цен у них идёт с шагом в 10 квадратных дюймов и количеством 2, 4, 6, 8, 10, 15, 20, 25, 30, 50, 75, 100. Они могут делать двухслойные платы без зелёнки и шелкографии (самая дешёвая партия - 95 долларов за 2 платы размером до 10 кв.дюймов), двухслойные платы с зелёнкой и шелкографией (от 121 доллара за 2 платы размером до 10 кв.дюймов), четырёхслойные платы с зелёнкой и шелкографией (от 169 долларов за 2 платы размером до 10 кв.дюймов) и шестислойные (уже от 284). С некоторых пор они предлагают бесплатную опцию серебрянного лужения. Стоит отметить, что у этого сайта

<sup>4</sup> <http://www.resonit.ru> – российский изготовитель плат, в том числе и минимальными тиражами. Возможно express изготовление плат.

есть большой брат [pcbpro.com](http://pcbpro.com), где можно заказывать большие объёмы, однако за более продолжительное время (до 5 недель) и ценообразование там более запутанное (или точнее сказать - традиционное), ну и там есть возможность изготовления разноцветных плат и плат с золотыми контактами. За обоими сайтами стоит компания Sunstone Circuits (см. <http://sunstone.com>), которая работает с профессионалами (путём общения с живым человеком, выставлением счетов и т.д.).

Третий в списке - это <http://batchpcb.com>, который, наверное, является самым дешёвым вариантом для прототипирования (но не быстрым). Расчёт стоимости у них ведётся очень просто - заказываемая плата (в формате гербер) укладывается в наименьший квадрат (размер стороны квадрата от 12.7 мм) и берётся его площадь в квадратных дюймах (число может быть дробным), далее эта площадь умножается на 2.5 доллара (для двухслойных либо 8 для четырёхслойных плат), умножается на количество плат плюс 10 долларов на обработку заказа. Далее они собирают заказы от разных людей (около недели для двухслойных и несколько дольше для четырёхслойных), чтобы сформировать большой лист, который затем отправляется на производство в Китай - там оно делается ещё недельку-другую, затем прибывает обратно в США, разрезается и рассылается заказчиком. Плюсы - очень дёшево для небольшого количества плат, возможность выложить в публичный доступ свой дизайн и даже заработать на нём деньги (правда заработать могут только те, кто проживает на территории США). Минус - непрогнозируемо долго, возможны непредвиденные замены спецификации как например внезапная смена обычного лужения на органическое покрытие (разрушается при пайке либо через полгода лежания) и обратно.

### Радиолобительство.

Пришло время поговорить о радиолобительстве. Как и во многих других странах мира в США любят и лелеют своих доморощенных радиолобителей (тут

любительская радиосвязь называется "HAM radio", а сами радиолюбители - "hams"). Чтобы стать американским радиолюбителем (хамом), нужно сдать экзамен, при этом гражданства США или наличия гринкарты не требуется (правда надо иметь SSN - номер социального страхования, что как правило означает, что человек легально работает на территории США и платит налоги), также отсутствуют возрастные ограничения (требуется лишь уметь прочитать вопросы и закрасить правильный вариант ответа) - известны случаи успешной сдачи экзаменов дошкольниками. Экзамены периодически принимаются на базе разнообразных клубов, организаций и учебных заведений специально лицензированными радиолюбителями достаточно часто и почти в любом городе - при этом существует три уровня: Technician, General и Extra (раньше ещё существовали Advanced и Novice, но их уже давно устранили). У каждого уровня - свой набор тем и вопросов, причём знания морзянки уже не требуют. Стоимость одной попытки - 15 долларов. Чтобы получить самый высокий уровень, надо успешно сдать экзамены на все предыдущие уровни (говорят, что это можно сделать в один день). Лично я пока сдал на Technician и готовлюсь на General<sup>5</sup>.

Для подготовки к экзаменам существуют книжки разной степени толстоты, аудиокурсы и программное обеспечение (небесплатное). Книжки как правило можно взять в любой городской библиотеке. Программное обеспечение обычно имитирует экзамен с ограничением по времени и вычислением факта прошёл-непрошёл по количеству правильных ответов. Все вопросы и ответы к ним опубликованы и известны заранее, неизвестен лишь порядок вопросов на экзамене, а также ответы к вопросам могут быть переставлены местами.

После успешной сдачи FCC (федеральная комиссия по коммуникациям) выдаёт

человеку позывной (callsign) либо увеличивает уровень человека, оставляя ему предыдущий позывной (хотя для высоких уровней есть возможность заказать себе более короткий позывной) и после этого человек уже может выходить в эфир с новыми правами. Technician даёт возможность выходить в эфир голосом на радиолюбительских УКВ диапазонах и морзянкой на некоторых участках радиолюбительских КВ диапазонов, General получает возможность голоса на КВ и более широкие участки радиолюбительских диапазонов, Extra допускается во все радиолюбительские диапазоны (см. <http://www.arrl.org/files/file/Regulatory/Band%20Chart%20Color%20May%202008.pdf>).

Стоит отметить тот факт, что радиолюбительство - очень дорогое хобби. Начиная с того, что радиопередающее оборудование, как правило, дорогое (возможно потому, что "Made in USA" в отличие от всего остального китайского ширпотреба), и заканчивая тем, что ошибки тут очень дорого стоят - штрафы за нарушение правил работы в эфире исчисляются тысячами долларов.

### **Заключение.**

Для поддержки сообщества любителей электроники в США существует несколько журналов и много веб-сайтов с вмняемыми коммьюнити. Периодически разными организациями проводятся конкурсы разработок, в которых может принять участие любой желающий, причём даже можно получить бесплатную плату для разработки в рамках этого конкурса (если успеть подать заявку). Из всего вышеизложенного можно заключить, что США если и не рай для электронщика, то очень близко расположенное к этому понятию место. Очень хотелось бы надеяться, что и Россия рано или поздно вернётся к воспитанию собственных Кулибиных и Поповых, способных удивить мир...

---

5 На текущий момент Shaos уже сдал на General, с чем мы поздравляем Александра от лица участников группы NedoPC.

## Использование NeoGS в качестве акселератора



*Автор: Alone Coder*

*Email:*

[dmitry.alonecoder@gmail.com](mailto:dmitry.alonecoder@gmail.com)

Если спросить программиста, чего больше всего не хватает Спектруму, он скорее всего ответит: скорости. Да, мы уже привыкли к чуть менее чем 1 MIPS и проблеме заполнить весь экран за фрейм. Но 64К интро Pimp My Spectrum от Ate Bit<sup>6</sup> очень хорошо показывает, как изменяется восприятие графики, когда она работает действительно быстро!

И вот, у нас есть Z80, который работает на 24 МГц. Его в наше время можно купить на любом углу. Но в ZX его не поставишь - мешает видеоконтроллер. Зато его можно поставить в звуковую карту, что ребята из NedoPC с успехом и сделали - и в результате мы имеем NeoGS<sup>7</sup>.

Принципиальное отличие NeoGS от классического General Sound - наличие режима DMA для передачи данных между ZX и NeoGS. Если на классическом GS требовалось крутить цикл поллинга портов с обеих сторон, то на NeoGS достаточно установить начальный адрес, а потом читать или писать на ZX через нижнее окно памяти. NeoGS при этом может заниматься чем угодно и практически не замедляется. ZX не замедляется тем более. Это значит, что в турбо-режиме (много ли осталось живых Спектрумов с ZX-BUS, но без турбо-режима?) можно обновлять стандартный экран целиком за фрейм, и останется достаточно времени на музыку, обработку клавиатуры и чего бы там ни было ещё. Всё абсолютно плавно, и получается практически эффект игровой приставки.

Важно заметить, что адрес ячейки, которую читает ZX, не важен (NeoGS про неё даже не знает). Можно читать любую ячейку в области #0000..#3fff. NeoGS всё равно выдаст содержимое текущего адреса и

автоматически передвинет указатель вперёд. Поэтому типичный цикл переброски экрана выглядит так:

```
ld sp,#db00
ld h,0
dup 6912/2
ld d,(h1)
ld e,(h1)
push de
edup
(86417 тактов).
```

Разумеется, для этого экран, заготовленный в NeoGS, должен лежать там задом наперёд.

Поскольку остаются свободные такты на ZX, можно заготовить экран в каком-то другом формате, чтобы NeoGS было удобнее его просчитывать. Например, по столбцам (192 байта данных, 64 байта пропустить - и так 32 раза):

```
ld sp,0
ld l,#1f
ld bc,#f920
col0
_=#c0
if0 _&7
ld h,_
endif
dup 24
dup 4
pop de
ld (h1),e
inc h
ld (h1),d
inc h
edup
org $-1
add h1,bc
_+=1
edup
dup 32
pop de
edup
dec l
jp p,col0
(115040 тактов, не считая атрибутов).
```

Тут уже зависит от скорости конкретного ZX: Scorpion и Pentagon 1024SL v2.2 не вытянут такую скорость, PentEvo сможет, а KAY под вопросом. Впрочем, можно уменьшить количество столбцов или строк.

Обратите внимание, NeoGS, в принципе, не обязан использовать какие-либо прерывания,

<sup>6</sup> <http://www.youtube.com/watch?v=7KegY8YIzQ4>

<sup>7</sup> <http://www.nedopc.com/gs/ngs.php>

ведь синхронизацию нужно производить только после построения кадра — чтением признака от ZX. А это значит, что допустимо любое, даже самое безумное использование стека!

В предыдущем примере можно было бы пропускать байты не пустым чтением, а перестановкой указателя, но выигрыш сомнителен, так как для этого надо организовать обработчик NMI в NeoGS. Тогда пришлось бы отказаться в NeoGS от некоторых типов стековых процедур.

Понятно, что нужно обязательно использовать двойную буферизацию передаваемых данных. Программа в NeoGS строит один кадр, а в это время ZX через DMA читает предыдущий. После построения кадра NeoGS ждёт бит в порту, а ZX его установит по прерыванию. Потом можно обменяться какими-то данными (например, состоянием клавиатуры) и начать всё сначала.

Двойную буферизацию на NeoGS делать довольно просто благодаря наличию двух окон переключения памяти. В одном окне можно держать экран, а в другом графику. Можно оба экрана хранить в нижней памяти. В принципе, под NeoGS можно адаптировать и спектрумовские программы, если организовать вышеописанный цикл переброски экрана - синхронизации - чтения клавиатуры.

Не обязательно ограничиваться стандартным спектрумовским экраном. Например, первый эффект в демо The Link<sup>8</sup> использует мультиколор 4x2. Исходники The Link лежат на самом диске с демой - можете посмотреть как это работает. К сожалению, в мультиколоре доступно всего 8 цветов (15 дадут клешинг), но есть выход: можно пускать на чётных строках раstra мультиколор (экран 0), а на нечётных - чанки (экран 1). Если ограничиться только чёрными, белыми и серыми чанками, то и тогда цветов будет 24. Правда, придётся передавать в два раза больше данных, но никто не мешает включать турбо-режим в нижнем бордере! По моим расчётам, тактов

на Pentagon 1024SL v2.2 вполне хватает.

На NeoGS можно реализовать даже фреймовый 2D движок под 16col в стиле Ball Quest - скорости впритык, но хватает. Правда, с появлением спрайтов в PentEvo (<http://forum.nedopc.com/viewtopic.php?f=30&t=622>) это уже не так нужно. Подождём их эмуляции.

Если не заморачиваться насчёт фреймовости, то можно передавать 16col экран целиком. Это займёт 4 фрейма на Pentagon 1024SL v2.2, что оставляет достаточно времени NeoGS на всякие трёхмерные вычисления.

Можно играть и 16col видео с SD-карточки, установленной в NeoGS, пользуясь встроенным mp3-декодером для звука - собственно, такую программу у нас уже написали<sup>9</sup>. Так что проблема видеовставок в играх проблемой не является.

Поскольку в NeoGS есть ПЛИС, которая заполнена лишь частично, то можно ускорять через неё ресурсоёмкие операции типа умножений или c2p (ACNews #56<sup>10</sup>). Прошивку можно менять на лету, это может делать каждая программа. Пока что таких прошивок нет, но исходники стандартной в свободном доступе - так что всё возможно. Тут опять встаёт проблема эмуляции, которую можно решить разработкой одной раз и навсегда стандартной прошивки с наворотами. В случае чего, её можно обновлять одновременно с эмулятором, но при условии совместимости со старым софтом.

Стандартный NeoGS, во всяком случае, прекрасно эмулируется.

---

8 <http://www.youtube.com/watch?v=yNL2NCOGN5Q>

9 <http://www.youtube.com/watch?v=nuBTQKdnNow>

10 <http://alonecoder.nedopc.com/zx/books/ACN56.rar>

## FAT на Спектруме. Теория.



Автор: Savelij  
Email: [savelij13@yandex.ru](mailto:savelij13@yandex.ru)

Статья основана на документе «FAT32 File System Specification» (fatgen), в которой описаны все разновидности FAT (целых 3 штуки-12, 16 и 32 битные), а не только FAT32 как гласит название. На основе этой спецификации был написан драйвер по работе с FAT всех трех разновидностей, определение разрядности делается при инициализации переменных драйвера.

В статье описано только то, что реально используется в драйвере, за более полной информацией обращаться к вышеуказанному документу или к другим соответствующим документам.

### Некоторое отступление от темы или с чего начинается файловая система.

Любая файловая система может располагаться на дискетах, винчестерах, SD картах и других устройствах хранения информации. При размещении на носителях для каждой файловой системы выделяется место, называемое разделом (partition). Разделы подразделяются на основные (primary partition) и расширенные (extended partition). Расширенный раздел может быть только один. Внутри расширенного раздела можно создать любое количество логических дисков.

Для размещения информации о начале, типе и размерах разделов в самом первом секторе носителя располагается MBR (Master Boot Record-главная загрузочная запись). MBR состоит из предзагрузчика и 4-х 16-ти байтных описателей разделов. В драйвере используются только описатели разделов, если MBR обнаружен.

### Структура MBR:

| Смещение в секторе | Размер   | Описание               |
|--------------------|----------|------------------------|
| 446 (#1BE)         | 16 (#10) | Описатель 1-го раздела |
| 462 (#1CE)         | 16 (#10) | Описатель 2-го раздела |
| 478 (#1DE)         | 16 (#10) | Описатель 3-го раздела |
| 494 (#1EE)         | 16 (#10) | Описатель 4-го раздела |

### Структура описателя раздела:

| Смещение | Размер | Описание                   |
|----------|--------|----------------------------|
| 0        | 1      | Признак активности раздела |
| 4        | 1      | Код типа раздела           |
| 8        | 4      | Смещение первого сектора   |

Признак активности раздела — может принимать два значения:

- #00-раздел не активен;
- #80-раздел активен.

Тип раздела — соответствует типу файловой системы (всего типов файловых систем очень много). Для FAT может принимать значения:

- #01 – FAT12;
- #06, #0E – FAT16;
- #0B, #0C – FAT32.

Смещение первого сектора — смещение раздела от первого сектора носителя в секторах. При работе в LBA режиме на винчестере и при работе с SD картами достаточно знать смещение раздела.

Основных разделов можно создать только четыре. Возможно создать расширенный раздел (чтобы создать число разделов больше 4-х), под который может отводиться все оставшееся место на устройстве. Внутри расширенного раздела можно создать любое количество дополнительных разделов (secondary partition). Расширенный раздел может быть только один.

При создании на устройстве хранения данных одного раздела, MBR может отсутствовать и в самом первом секторе может располагаться непосредственно описатель раздела, который может занимать все пространство носителя. При отсутствии MBR имеет место быть некоторая проблема в

правильном определении разрядности FAT.

### FAT — File Allocation Table (таблица размещения файлов).

Раздел FAT подразделяется на следующие области:

- 1) загрузочный сектор – загрузчик раздела и BPB;
- 2) таблица размещения файлов - собственно FAT (традиционно в двух экземплярах, но вообще-то количество копий указано в загрузочном секторе). Содержит список всех кластеров;
- 3) корневой каталог;
- 4) файлы.

### BPB (BIOS Parameter Block).

BPB (BIOS Parameter Block) – описатель FAT раздела. Описатель раздела — первый сектор раздела, на начало которого указывает 16-ти байтный описатель в MBR или при отсутствии MBR сектор находящийся в самом первом секторе носителя. Кроме описателя раздела в этом же секторе может находиться загрузчик операционной системы. В связи с тем, что BPB не является целостной структурой и изменялся по мере повышения разрядности FAT некоторые значения BPB перемешаны со структурой самого загрузочного сектора.

Приведу описания некоторых значений, достаточных для работы с FAT любой разрядности (12, 16 или 32 битной). Полное описание можно найти в соответствующей документации. В дальнейшем описании, все поля, начинающиеся с BPB\_, являются частью BPB. Все поля, начинающиеся с BS\_, являются частью загрузочного сектора (к BPB не принадлежат). Смещение указано в байтах от начала сектора, размер указан также в байтах.

BPB структура:

| Имя            | Смещение | Размер | Описание   |
|----------------|----------|--------|--|
| BPB_BytsPerSec | 11 (#0B) | 2      | Количество байтов в секторе.   |
| BPB_SecPerClus | 13 (#0D) | 1      | Количество секторов в кластере. Значение должно быть числом в степени 2, и больше 0. Разрешённые |

|                |          |   |   |
|----------------|----------|---|---|
|                |          |   | значения: 1, 2, 4, 8, 16, 32, 64 и 128.   |
| BPB_RsvdSecCnt | 14 (#0E) | 2 | Количество секторов в Reserved region (начинается с первого сектора диска). Должно быть больше 0. Для FAT12 и FAT16 дисков, это значение должно быть только 1.  |
| BPB_NumFATs    | 16 (#10) | 1 | Для FAT12 и FAT16 дисков, это поле содержит число 32-байтных элементов корневой директории. Для FAT32 дисков, это поле должно быть 0. Для FAT12 и FAT16 дисков значение этого поля, умноженное на 32 должно быть кратно BPB_BytsPerSec. Для максимальной совместимости FAT16 диски должны содержать значение 512.   |
| BPB_RootEntCnt | 17 (#11) | 2 | Для FAT12/16 дисков это поле содержит число 32-байтных элементов корневой директории. Для FAT32 дисков это поле должно быть 0. Для FAT12/16 дисков значение этого поля, умноженное на 32 должно быть кратно BPB_BytsPerSec. Для максимальной совместимости FAT16 диски должны содержать значение 512.   |
| BPB_TotSec16   | 19 (#13) | 2 | Старое 16-битное поле: общее количество секторов на диске. Это количество включает в себя все четыре региона диска. Значение не 0, но если равно 0, то BPB_TotSec32 должно быть не 0. Для FAT32 дисков значение всегда 0. Для FAT12 и FAT16 дисков это поле содержит количество секторов, а BPB_TotSec32 равно 0, если значение «умещается» (меньше 0x10000). |
| BPB_FATSz16    | 22 (#16) | 2 | Для FAT12/FAT16 это количество секторов одной FAT. Для FAT32 это значение равно 0, а количество секторов одной  |



|              |          |   |  |
|--------------|----------|---|--|
|              |          |   | FAT содержится в BPB_FATSz32.  |
| BPB_TotSec32 | 32 (#20) | 4 | Новое 32-битное поле: общее количество секторов на диске. Это количество включает в себя все четыре региона диска. Может быть 0, если 0 то BPB_TotSec16 должно быть не 0. Для FAT32 дисков значение всегда не 0. Для FAT12/FAT16 дисков поле содержит количество секторов, когда BPB_TotSec16 равно 0 (количество равно или больше 0x10000). |

Начиная со смещения +36 могут находиться один из двух описателей: для FAT12/16 или для FAT32 соответственно. Структура соответствующая FAT12/16 по этому смещению мной игнорируется, так как для работы с FAT12/16 достаточно предыдущих значений.

#### Структура FAT32:

| Имя          | Смещение | Размер | Описание  |
|--------------|----------|--------|---|
| BPB_FATSz32  | 36 (#24) | 4      | Это поле есть только в FAT32 и отсутствует в FAT12/16. Это 32-битное поле FAT32 содержит количество секторов одной FAT. При этом BPB_FATSz16 должно быть 0. |
| BPB_RootClus | 44 (#2C) | 4      | Это поле есть только в FAT32 и отсутствует в FAT12/16. Номер первого кластера корневой директории. Обычно 2, но может быть и другим.                        |
| BPB_FSInfo   | 48 (#30) | 2      | Это поле есть только в FAT32 и отсутствует в FAT12/16. Номер сектора со структурой FSINFO в зарезервированной части FAT32. Обычно 1.                        |

#### FSInfo.

На разделе FAT32 таблица FAT может иметь значительный размер, в отличие от FAT16 ограниченного размером 128Кб и FAT12 ограниченного размером 6Кб. Поэтому “последнее известное” количество свободных кластеров хранится в специальном секторе, чтобы не вычислять его каждый раз когда

нужно узнать размер свободного места на диске. Смещение сектора FSInfo от начала раздела хранится в поле BPB\_FSInfo.

Структура сектора FSInfo:

| Имя            | Смещение   | Размер | Описание  |
|----------------|------------|--------|---|
| FSI_Free_Count | 488 (#1E8) | 4      | Хранит “последнее известное” количество свободных кластеров диска. Если равно #FFFFFFFF, то количество неизвестно и должно быть вычислено. Может быть любое другое значение, при этом не обязательно корректное. Оно должно проверяться на выполнение условия <= количества кластеров на диске.   |
| FSI_Nxt_Free   | 492 (#1EC) | 4      | Содержит номер кластера, начиная с которого надо искать свободный кластер. Поскольку FAT велика, могут происходить значительные затраты времени, когда в начале FAT много занятых кластеров, а драйвер ищет свободный, начиная с номера 2. Обычно здесь номер последнего выделенного кластера. Если здесь значение #FFFFFFFF, то нет вспомогательного значения и поиск должен производиться с номера 2. Любое другое значение может использоваться, но должно предварительно проверяться на условие <= количества кластеров на диске. |

При совместном использовании носителя с FATом и на Спектруме и под Windows имеет место небольшая проблема. При записи под Windows старшие 16 бит значения FSI\_Free\_Count чаще всего обнуляются и содержимое FSI\_Nxt\_Free содержит неправильное значение. И поиск следующего свободного кластера на Спектруме может занимать много времени, тем большее чем больше размер носителя.

## Файлы и директории.

Файл (англ. file — папка, скоросшиватель) — концепция в вычислительной технике: сущность, позволяющая получить доступ к какому-либо ресурсу вычислительной системы и обладающая рядом признаков... (википедия). В общем некоторое количество байтов занимающих место на носителе ;-).

Директория FAT - ни что иное как “файл”, состоящий из списка 32-байтных структур. Лишь одна специальная директория, которая всегда должна присутствовать это корневая директория. На дисках FAT12 и FAT16 корневая директория расположена в фиксированном месте - непосредственно после последней таблицы FAT и состоит из фиксированного количества секторов, вычисляемого из BPB\_RootEntCnt. Чаще всего размер директории имеет размер, рассчитанный на 512 32-х байтных записей, а при использовании длинных имен это число еще меньше. Для дисков FAT12 и FAT16 номер первого сектора корневой директории зависит от номера первого сектора таблицы FAT:

$$\text{FirstRootDirSecNum} = \text{BPB\_ResvdSecCnt} + (\text{BPB\_NumFATs} * \text{BPB\_FATsSz16})$$

Для FAT32 корневая директория может быть произвольного размера из последовательности кластеров, так же как любая другая директория. Номер первого кластера корневой директории FAT32 хранится в BPB\_RootClus. В отличие от других директорий корневая директория любой FAT не имеет штампа даты и времени, не имеет имени файла и не содержит “.” и “..” в первых двух записях. Ещё один аспект - только в корневой директории может содержаться файл у которого установлен единственный атрибут ATTR\_VOLUME\_ID — фактическое имя раздела.

Расположение корневой директории зависит от разрядности FAT: для FAT12/16 под корневую директорию отведено несколько секторов между таблицей кластеров и началом области, где непосредственно находятся файлы и другие директории. На FAT32 корневая директория находится там

же, где и остальные файлы/директории.

Все вновь создаваемые директории всегда начинаются с двух стандартных записей с именами «.» (точка) и «..» (две точки). У первой записи устанавливаются DIR\_FstClusLO и DIR\_FstClusHI такие же, как у этой новой директории (номер кластера, который содержит эти записи точка и две точки), то есть указывает сама на себя. У второй записи DIR\_FstClusLO и DIR\_FstClusHI устанавливаются в номер первого кластера родительской директории или в 0, если она корневая, даже для FAT32.

Структура 32-х байтного описателя в директории:

| Имя           | Смещение | Размер | Описание   |
|---------------|----------|--------|--|
| DIR_Name      | 0        | 11     | Короткое имя.  |
| DIR_Attr      | 11 (#0B) | 1      | Атрибуты файла (по битам):<br>7-всегда 0<br>6-всегда 0<br>5-ATTR_ARCHIVE<br>4-ATTR_DIRECTORY<br>3-ATTR_VOLUME_ID<br>2-ATTR_SYSTEM<br>1-ATTR_HIDDEN<br>0-ATTR_READ_ONLY |
| DIR_FstClusHI | 20 (#14) | 2      | Старшее слово номера первого кластера (всегда 0 для FAT12/16).   |
| DIR_FstClusLO | 26 (#1A) | 2      | Младшее слово номера первого кластера.   |
| DIR_FileSize  | 28 (#1C) | 4      | 32-битный размер файла в байтах.   |

DIR\_Name — состоит из двух частей, 8-символьная основная часть имени, и 3-символьное расширение (сокращенно называемое 8.3 или короткое имя). Свободные концы обеих частей заполнены байтами #20. Первый байт имени не может содержать байт #20 (пробел) и может содержать значение:

- #00 - первая свободная запись в директории, после этой записи ничего нет;
- #05 - настоящий байт имени равен #E5 - соответствует старшему байту японской кодировки KANJI;
- #E5 - запись в директории свободна, так как файл/директория были удалены.

Следующие символы не разрешены в имени файла/директории: #22, #2A, #2B, #2C, #2E, #2F, #3A, #3B, #3C, #3D, #3E, #3F, #5B, #5C, #5D, #7C.

DIR\_Attr — сочетание атрибутов ATTR\_READ\_ONLY, ATTR\_HIDDEN, ATTR\_SYSTEM и ATTR\_VOLUME\_ID установленных в 1 указывает, что описатель файла принадлежит к части длинного имени.

DIR\_FstClusHI и DIR\_FstClusLO — номер кластера с которого начинается цепочка кластеров занятых файлом/директорией.

DIR\_FileSize — собственно размер файла в байтах.

### Длинные имена.

Файл/директория кроме короткого имени может иметь так называемое длинное имя. Длинное имя может иметь длину 255 символов. Каждый символ имени занимает 2 байта в кодировке Unicode. Для нас интересны только диапазон от #0020 до #007F содержащий ASCII символы и следующие последовательности #0401, #0410-#044F и #0451. Из этих кодов в драйвере соответственно получаются английские и русские имена, циферки и прочие печатные символы. Преобразование происходит в так называемую альтернативную кодировку.

Набор записей длинных имён всегда ассоциирован записи короткого имени, которому они предшествуют. Записи длинных имён не могут легально существовать сами по себе. Записи длинного имени всегда непосредственно предшествуют и физически прилегают к записи короткого имени. Последовательность записей длинного имени:

| Запись  | Номер                      |
|---|----------------------------|
| N (последняя) запись длинного имени                       | LAST_LONG_ENTRY (0x40)   N |
| ... дополнительные записи длинного имени                  | ...                        |
| 1 (первая) запись длинного имени                          | 1                          |
| Запись короткого имени, которому предшествует длинное имя | нет                        |

### Структура длинного имени:

| Поле           | Смещение | Размер | Описание  |
|----------------|----------|--------|---|
| LDIR_Ord       | 0        | 1      | Порядковый номер этой записи в последовательности записей длинных имён, ассоциированных с записью короткого имени в конце списка. Если соответствует бит 6 включен, то запись является последней в списке записей длинных имён. |
| LDIR_Name1     | 1        | 10     | Символы 1-5 длинного имени в данном компоненте.   |
| LDIR_Attr      | 11       | 1      | Атрибуты – содержит ATTR_LONG_NAME  |
| LDIR_Type      | 12       | 1      | Если 0, то запись является компонентом длинного имени.  |
| LDIR_Chksum    | 13       | 1      | Контрольная сумма короткого имени (которое располагается в конце списка).   |
| LDIR_Name2     | 14       | 12     | Символы 6-11 длинного имени в данном компоненте.  |
| LDIR_FstClusLO | 26       | 2      | Должно быть 0. Для длинного имени значение не используется.   |
| LDIR_Name3     | 28       | 4      | Символы 12-13 длинного имени в данном компоненте.   |

Являясь парой, запись короткого имени хранит многие важные поля: дата последнего обращения, время создания, дата создания, номер первого кластера, размер. В записи длинного имени содержится лишь длинное имя файла.

### Таблица кластеров.

Все пространство FAT раздела разделено на кластера. Кластер — это непрерывное пространство состоящее, из одного или нескольких секторов. Количество секторов в кластере может быть от 1 ( $2^0$ ) до 128 ( $2^7$ ).

Каждый файл занимает один или несколько кластеров, занятые файлом кластера образуют цепочку. Цепочка кластеров необратима. Можно прочитать номер следующего кластера, но нет никакого

способа узнать номер предыдущего кластера. Начало цепочки хранится в описателе файла и указывает на таблицу кластеров находящуюся на некотором расстоянии от начала раздела. На сколько секторов от начала раздела находится таблица кластеров зависит от разрядности FAT. Отсчет номеров кластеров начинается с номера 2. Кластера 0 и 1 зарезервированы. Кластер 0 содержит значение BPB\_Media в младших 8 битах, а все остальные биты содержат 1.

Данные файла ассоциируются с файлом следующим образом. В директории (32-х байтный описатель) содержится номер первого кластера, в котором располагаются данные файла. По номеру этого кластера вычисляется смещение в таблице кластеров из которого берется номер следующего кластера файла. Кластер, расположенный в FAT таблице может содержать значение EOC (End Of Clusterchain – конец цепочки кластеров) или номер следующего кластера файла. Значение EOC равно одному из следующих значений: #0FFF, #FFFF или #0FFFFFFF для FAT12, FAT16 или FAT32 соответственно и указывает на окончание цепочки кластеров. Все свободные кластеры имеют значение 0.

Сектор нахождения начала кластера вычисляется по нескольким формулам. Сначала вычисляется размер в секторах корневой директории:

$$\text{RootDirSectors} = ((\text{BPB\_RootEntCnt} * 32) + (\text{BPB\_BytsPerSec} - 1)) / \text{BPB\_BytsPerSec}$$

Для FAT32 поле BPB\_RootEntCnt всегда содержит 0, поэтому для FAT32 RootDirSectors всегда 0. Здесь число 32 это размер одного элемента директории FAT (в байтах). Далее вычисляется начало региона данных:

$$\text{FirstDataSector} = \text{BPB\_ResvdSecCnt} + (\text{BPB\_NumFATs} * \text{FATSz}) + \text{RootDirSectors}$$

FATSz - это значение взятое из BPB\_FATSz16 (для FAT12/16) или BPB\_FATSz32 (для FAT32). Этот первый сектор начала региона данных считается не от начала носителя, а от первого сектора BPB. То есть сектор BPB равен 0, но BPB может находиться и в нулевом секторе

носителя и вначале раздела. Имея номер кластера N номер первого сектора этого кластера (от начала носителя) вычисляется так:

$$\text{FirstSectorofCluster} = ((N - 2) * \text{BPB\_SecPerClus}) + \text{FirstDataSector}$$

В драйвере первые две формулы рассчитываются на этапе инициализации переменных и при вычислении реального сектора начала кластера все сводится к последней формуле.

### Проблемы определения разрядности FAT.

Перед определением какие разделы и сколько их находится на носителе, нужно определить присутствует ли на носителе MBR.

Как один из вариантов определения мной использован следующий метод:

в каждой 16-ти байтовой записи проверяется -активность раздела. Возможные значения #00 или #80.

-тип раздела. Возможные значения: #01, #06, #0E, #0B, #0C - числа соответствующие одному из типов FAT. При других значениях тип раздела считается не поддерживаемым и пропускается.

При неудаче делается попытка определить первый сектор как описатель раздела. Соответственно, проверяются несколько контрольных точек этого сектора, относящихся к BPB. Проверяются следующие значения (по смещению от начала сектора):

- +13 (байт) - число секторов в кластере. Являясь числом 2 в степени от 0 до 7, соответственно должен содержать только один установленный бит. Нестандартный размер кластера (произвольное число секторов в кластере) игнорируется.
- +14 (слово) - количество секторов в Reserved region (начинается с первого сектора диска). Должно быть больше 0. Для FAT12 и FAT16 дисков, это значение должно быть только 1. Для FAT32 дисков обычное значение 32.
- +19 (слово) и +32 (двойное слово) - общее количество секторов на разделе.

Одно из этих значений должно быть равно ноль, в то же время другое значение не должно быть равно ноль.

- +21 (байт) - маркерный байт совпадающий с байтом, записанным в нулевой байте таблицы кластеров. Может принимать значения: #F0, #F8, #F9, #FA, #FB, #FC, #FD, #FE и #FF. Проверяется на равенство #F0, игнорируя младшие 4 бита.

Определение типа раздела и разрядности FAT делается легко и просто при наличии MBR. Сложность появляется, если на устройстве один раздел и отсутствует MBR, а в нулевом секторе устройства хранения находится непосредственно описатель раздела. Дело в том, что в описателе раздела FAT отсутствует конкретное указание разрядности FAT. Текстовая строка в нулевом секторе по смещению от начала сектора +54 (для FAT12 и FAT16) или +82 (для FAT32) ни на что не указывает и может отсутствовать или содержать что угодно. И определять по этой текстовой строке разрядность FAT не самое умное решение.

Поиски способа правильного определения ни к чему не привели. Способ, описанный в спецификации на FAT имеет недостатки и дает ложные срабатывания, которые приводят к проблемам в работе драйвера. Способ заключается в проверке количества кластеров на разделе и принятии решения о разрядности: для FAT12 количество кластеров должно быть 4084 и менее, для FAT16 не менее 4085 и не более 65524, для FAT32 более 65525 кластеров. Проблема в том что в некоторых случаях носитель может быть отформатирован в FAT (под Windows), когда количество кластеров не соответствует этому правилу. Простой пример: SD карта объемом 512 мегабайт, отформатированная как FAT32, хотя по количеству кластеров это 100% должна быть FAT16.

Мной был «изобретен» способ правильного определения разрядности FAT при отсутствии MBR. Процедура инициализации FAT драйвера фактически состоит из 3 этапов:

- на первом этапе определяется наличие MBR выше описанным способом;
- на втором этапе инициализируются переменные драйвера. На этом этапе разрядность FATа не важна, некоторые переменные переносятся из ВРВ, а некоторые вычисляются простой математикой;
- на последнем этапе для правильного определения используются переменные содержащие: количество кластеров на разделе (CLS\_DSC) и размер FAT таблицы в секторах (SEC\_FAT). Проверка производится по следующему алгоритму: это FAT16, если  $(SEC\_FAT - ((CLS\_DSC * 2) / 512)) < 16$  или это FAT32, если  $(SEC\_FAT - ((CLS\_DSC * 4) / 512)) < 16$  иначе это FAT12.

Число 2 (4) в формулах это соответствующее число байт зависящее от разрядности FAT, то есть сколько байт занимает номер кластера. Проверка на число меньше 16 связано с тем, что таблица кластеров может занимать на несколько секторов больше реально необходимого количества.

Метод показал свою эффективность, ложных срабатываний не обнаружено.

### **Немного о грустном.**

Что до сих пор не сделано: нет удаления (Delete), нет перемещения (Move), нет возможности переименовывать файлы/директории с длинными именами (только чтение), не написаны утилитки по разметке (fdisk) и форматированию (format).

Пока не делалось никакой поддержки расширенных разделов и, соответственно, логических дисков, но это можно добавить и позже.

## “PENTAGON” 2.666 / 2.666le: история задумки, реализации, воплощения в жизнь и планы на будущее.



Автор: KOE

Email: [KingOfEvil@yandex.ru](mailto:KingOfEvil@yandex.ru)

Проект “PENTAGON” 2.666 задумывался как дальнейшее развитие и модификация версии 2.2. После выемки ряда черновиков из архива удалось выяснить, что точная дата начала работ над проектом 27 июня 2007 года. На начальном этапе работы преследовались две основных цели: во-первых – в максимальной степени исключить устаревшую и снятую с производства элементную базу, во-вторых – сделать архитектуру новой платы максимально гибкой. При этом важными задачами являлись как сохранение аппаратно-программной совместимости с платами версии 2.2 “снизу-вверх”, так и полная совместимость с классическим российским клоном ZX-Spectrum – Pentagon-128k.

Выбор элементной базы для нового проекта производился по критериям использования наиболее распространенных и одновременно наиболее современных элементов. Важным этапом стал выбор ПЛИС – уже на начальном этапе планировалось использовать помимо реального процессора Z80 его логический аналог на vhdl. Была выбрана ПЛИС фирмы Altera EP2C8Q208 (Cyclone-II, на тот момент одна из наиболее “продвинутых” серий). Её логическая ёмкость позволяла разместить не только vhdl-код Z80 и всю сопутствующую логику, но и код музыкального процессора и впоследствии эмулятора Спектрумовского контроллера дисковода KP1818BG93. Кроме того, было решено все же оставить реальный Z80 для успокоения “фанатиков”, однако впоследствии стало ясно, что необходимость в нем отсутствует практически полностью, а трудности с закупками устаревших компонентов с течением времени только увеличиваются.

Для работы с периферией был выбран

микроконтроллер семейства ARM7 производства фирмы NXP (тогда еще Philips). В качестве видеоконтрольного устройства уже на первом этапе планировалось использование только VGA-монитора, без поддержки обычных телевизоров. Хотя изначально решение казалось смелым и не всеми было однозначно воспринято, а для его реализации в технике пришлось приложить некоторое количество усилий, по прошествии времени стало ясно, что решение было исключительно правильным.

На этапе воплощения проекта в жизнь выяснилось, что конструкцию в задуманном виде невозможно реализовать на привычных для спектрумостроения двухслойных печатных платах, в первую очередь из-за сложной топологии соединений шин реального Z80 (с уровнями 5-вольтовой КМОП-логики), ПЛИС (с 3.3-вольтовой КМОП-логикой) и шины ZX-BUS. Опыты показали, что исключение хотя бы одного элемента (т.е. реального Z80 или шины) позволили бы реализовать топологию на 2-х слоях. Однако желание сделать полноценную архитектуру пересилило и первый вариант конструкции, получивший номер версии 2.666, делался на 4-х слойной плате. Кстати, изначально номер версии был 2.665, т.к. я предполагал, что реализовать конструкцию задуманной сложности с первого раза не получится, а номер версии 2.666 хотелось зарезервировать для окончательного варианта. Однако, к моему великому изумлению, первый вариант конструкции и стал окончательным, без единого (!!!) исправления схемы и без единой порезанной дорожки (!!!) на плате (мастерство, все-таки, не пропьешь, как гласит народная мудрость, кроме того, многократно оплачивать из своего кармана производство 4-слойных плат на тот момент (разводка платы была закончена в начале 2008 г.) являлось дорогим удовольствием (тогда на производствах многослойных плат не делали дешевые прототипы, а стоимость

любого заказа автоматически равнялась стоимости технологической заготовки, \$200 ... 250 в зависимости от сложности), что дополнительно стимулировало делать все предельно внимательно). Для справки, все предыдущие версии (1.4, 2.2) приобретали окончательный вид лишь со второго раза.

Процесс разводки печатной платы был весьма интересным. Несмотря на то, что платы такой сложности не всегда успешно получается делать с помощью программ автоматической трассировки, я решил попробовать в порядке эксперимента. Автоматическому трассировщику (использовался Spectra autorouter фирмы Cadence) было доверено делать трассировку основных сигнальных шин на внешних слоях по заданным правилам и алгоритмам трассировки. Наиболее ответственные узлы и шины питания на внутренних слоях были разведены вручную.

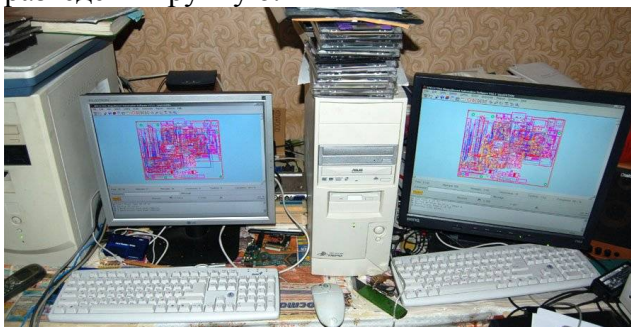


Рисунок 1: . Процесс автоматической трассировки соединений на плате "Pentagon" ver.2.666

Сам процесс трассировки занимал от нескольких десятков минут до нескольких часов, а результаты сильно зависели от задания начальных условий и правил трассировки. Было сделано более десятка промежуточных вариантов топологии, прежде чем, фактически методом проб и ошибок, было найдено правильное сочетание правил трассировки и получена окончательная топология соединений. Поскольку процесс оказался весьма долгим, трассировка делалась параллельно на двух одновременно работающих компьютерах с разными начальными условиями. Можно часами наблюдать, как горит огонь, как

льется вода и как кто-то другой работает, в моем случае к известной фразе следует добавить "и как разводится плата очередного Спектрум-клона". Наблюдать за процессом было действительно интересно, я даже сделал фото на память (рис.1). После получения окончательной топологии с помощью автотрассировщика, рисунок платы дополнительно вычищался вручную, т.к. автотрассировщик все-таки обладает рядом недостатков.

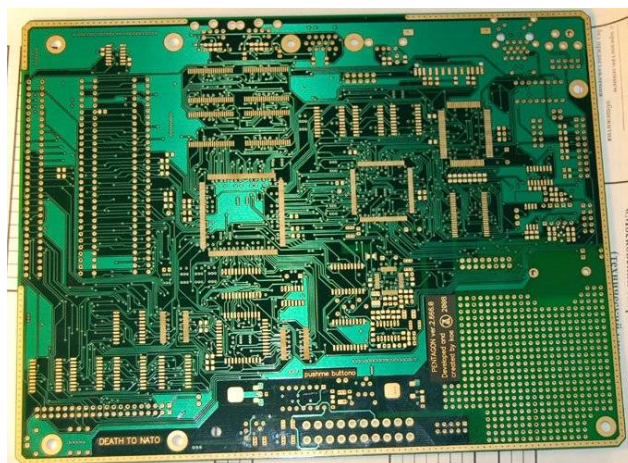


Рисунок 2: Печатная плата "Pentagon" ver.2.666

После окончания разработки конструкции на производстве было заказано изготовление комплекта плат, объемом сразу в две технологические заготовки и с финишным покрытием контактных площадок иммерсионным золотом (рискованный шаг, т.к. жалко было бы выбрасывать сразу весь комплект из-за возможных ошибок, но я все же надеялся на положительный результат, в итоге надежды оправдались). Образец вышедшей из производства платы версии 2.666 показан на рис.2, а на рис.3 показан вид собранной платы.

В рамках этого проекта впервые в истории отечественного спектрумостроения на одной плате был собран полный комплект современных контроллеров периферии, от PS/2 клавиатуры и мыши до IDE и SD-карты. Также была заложена аппаратная возможность использования Ethernet, USB, однако их программная поддержка пока отсутствует.



Рисунок 3: Собранная плата “Pentagon”  
ver.2.666 (2008 г.)

Работа над проектом не остановилась в 2008 году, практически сразу же после появления первых работоспособных версий встроенного программного обеспечения (прошивок ПЛИС и FPGA) появилась идея создания облегченной версии конструкции на двухслойной печатной плате. В первую очередь это было связано с дороговизной четырехслойных плат, а также к тому моменту я на практике убедился в отсутствии необходимости в именно реальном процессоре Z80 (практика показала, что те, кому он жизненно необходим, все равно ничего кроме “расово правильных и православных” Пентагонов образца 1991 года на рассыпной логике не приемлют принципиально). Новая плата появилась в 2009 году и получила версию 2.666le. В отличие от предыдущей, вся топология прорисовывалась вручную. Как ни странно, вновь конструкция получила окончательный вариант с первого раза, без переделок и исправлений. Внешний вид платы показан на рис.4.

В целом схемотехника и архитектура платы 2.666le практически повторяет версию 2.666, за исключением отсутствия реального процессора Z80, изменений видео-ЦАП и некоторой периферии.

Процесс разработки встроенного программного обеспечения (прошивок ПЛИС и контроллера периферии) довольно бурно начался еще в 2008 году. Полностью с нуля был написан основной vhdl-код, описывающий логическую часть архитектуры системы и С-код контроллера

периферии. Была реализована часть функций, кроме расширенных видеорежимов и эмулятора КР1818ВГ93. Однако в 2010 – первой половине 2011 года из-за сильной занятости на работе и параллельной учебе в аспирантуре МЭИ работу над проектом пришлось сначала замедлить, потом и вовсе заморозить. Возобновление работ над проектом произошло лишь в конце лета 2011 года, после окончания аспирантуры и перехода на более спокойный режим работы. За последнее время были реализованы два дополнительных видеорежима (16 цветов на точку и “прозрачный текстовый режим”, позволяющий выводить текстовую информацию на экран в формате до 32 строк по 80 символов прямо поверх Спектрум-экрана, несколько снизив его яркость). Кроме того, обычная реализация звукового чипа YM2149 заменена на Turbo Sound, переработана реализация CMOS-памяти с целью увеличения программной совместимости, заработал двухканальный АЦП. Новые версии прошивок на данный момент (ноябрь 2011) не зарелижены, но в скором времени ожидается серьезное обновление. Таким образом, на данный момент первостепенной нерешенной задачей и планом на будущее развитие проекта является реализация аппаратного (или аппаратно-программного) эмулятора КР1818ВГ93.



Рисунок 4: Печатная плата “Pentagon”  
ver.2.666le (2009 г.)



В заключении приведу основные технические характеристики плат версии 2.6661e:

|  |  |
|--|--|
| Платформа                                  | ZX-Spectrum  |
| Аппаратно-программная совместимость        | Pentagon   |
| Архитектура                                | Открытая, системная шина ZX-BUS  |
| Тип микропроцессора                        | T80 vhdl core (совместим с Zilog Z80)  |
| Тактовая частота процессора                | 3.5 МГц (Spectrum-режим) / 28 МГц (турбо-режим)  |
| Объем ОЗУ                                  | 2048 к (в текущей прошивке программно поддержано 1024к)  |
| Тип используемой ПЛИС                      | EP2C8Q208C8N, ф. Altera  |
| Тип встроенного периферийного контроллера  | LPC2368FBD100, ф. NXP  |
| Тактовая частота периферийного контроллера | 72 МГц   |
| Объем встроенного flash-ПЗУ                | 512к   |
| Тип музыкального процессора                | 2 x YM2149 (Turbo Sound), vhdl core  |
| Интерфейсы ввода                           | PS2 keyboard/mouse   |
| Интерфейсы ввода/вывода                    | SD-CARD (стандарт ZC), IDE (стандарт Nemo), UART (в текущей прошивке не поддержан), USB (на данный момент программной поддержки нет) |
| Экран                                      | SVGA, 5 бит на каждый цвет   |
| Питание                                    | +5 В, 1.5 Вт (без периферии), поддерживаются ATX блоки питания   |
| Форм-фактор                                | Micro-ATX  |

## Почему “ZX Evolution” ?



*Автор: CHR V*

*Email: [chunin@mail.ru](mailto:chunin@mail.ru)*

В этой статье попробую ответить почему ZX Evolution получился именно таким.

### **Начало.**

Самый первый чертеж, точнее даже блок-схема прототипа очень сильно напоминала Pentagon 2666. Было большое обсуждение, в основном, между LVD, КОЕ и мной, но дальше блок-схемы проект не продвинулся. Были закуплены какие-то детали, но тема всеравно медленно заглохла. А тем временем КОЕ продемонстрировал новый Pentagon 2666.

Понимание новой концепции компьютера рождалось в длительных дискуссиях с LVD. Главной идеей являлось сохранение основных «органов» спектрума (процессор, звуковой сопроцессор, контроллер дисководов) и использование современной периферии, носителей информации и корпусов. То есть хотелось иметь такой компьютер, чтобы он сохранил в себе веяние 90ых и был легким для сборки и запуска современным не очень продвинутым пользователем.

На этапе подбора элементной базы исходили из того, чтобы получить не самый навороченный компьютер, но иметь достаточный запас для развития архитектуры. Поэтому была выбрана загружаемая архитектура — то есть схему можно было исправлять и совершенствовать без использования паяльника. Это возможно благодаря программируемой логической матрице, являющейся основой компьютера (Altera EP1K50). Более того, удалось сделать универсальную платформу на которой можно реализовать практически любые отечественные клоны (но об этом позже).

Потихоньку рисовалась схема rev.A и неожиданно через Москву проездом ехал DDP и было некоторое время чтобы пересечься. Так как я был на работе, мне

ничего было показать и я решил распечатать схему и показать ее DDP. Собственно так в проект влился еще один активнейший участник, программист и железячник. После чего дело пошло веселее. Достаточно вспомнить горячие споры о глубине цвета и о использовании ног микросхемы Altera.

Мне, как «майнтейнеру-главному» проекта, пришлось продавливать некоторые решения с которыми другие участники проекта не были согласны. Но для себя я давно решил, что проект не должен выходить за пределы возможностей обычного спектрума и не гнаться за какими-то сомнительными «наворотами». Большинству пользователей нужна стабильная машинка, позволяющая удобно запускать уже имеющиеся программы, а демомейкерам иметь платформу наиболее приближенную к отечественному «стандарту» Pentagon 128 (меня, как конструктора железа, выбор такого стандарта всегда мало радовал).

### **Revision A.**

Как говорится, «первый блин комом» и первая ревизия была как и со смешными так и с неожиданными ошибками. Но обо всем по порядку. Изготовлено было всего пять плат. Первую плату собрал LVD и на его долю выпала почетная честь «оживления» ZX Evolution. Затем «подтянулись» и я с DDP. Первым сюрпризом был «отзеркаленный» разъем клавиатуры-мыши, затем проявилось непредусмотренное поведение контроллера дисковода, но надо отметить что разработчики с честью справились со всеми испытаниями. Когда ZX Evolution уверенно «задышал», было принято решение отдать одну из плат в эксплуатацию моему соседу по району — ААА. Наверно с этого момента можно отсчитывать начало большой жизни ZX Evolution.

### **Revision B.**

Первая партия массово производящихся ZX Evolution-ов расходилась достаточно

медленно. Покупали в основном «старые бойцы», кто хорошо знаком с продукцией NedoPC. Я, как изготовитель, набивал руку на пайке. Savelij активно начал адаптировать и устранять глюки у «Gluk reset service». DDP написал первую версию бутлоадера, который с трудом «понимал» SD карты. LVD боролся с глюками железа, а также с нашими новшествами и исправлениями. Но тут грянул феерический спрос на платы (не сомневаюсь - AAA приложил к этому не только руки, но и ролики в ютубе). Очередь выстроилась аж на три-четыре месяца вперед. Часть сборки пришлось отдать специализированной фирме, что немного подняло стоимость платы, но не уменьшило спрос. Спустя полгода удалось насытить «голодающих» спектрумистов и я засел за новую версию платы.

#### **Revision C.**

Вдохновили меня на новую версию, замечательные корпуса формата miniITX, появившиеся в продаже, и прекращение производства держателей для SD карт, применяемых на плате предыдущей версии. Уменьшить размер платы безболезненно не удавалось, несмотря на использование процессора в маленьком корпусе и отказа от «лишних» разъемов — пришлось отказаться от PAL-кодера, но в качестве бонуса был добавлен USB-RS232 переходник. Дизайн получился компактным, но все равно хотелось добавить какую-то изюминку. Так как этот год был знаменателен исторической датой — 50летия запуска человека в космос, то я решил сделать плату цветом флага своей Родины (большой страны под названием СССР). Несомненно этот ход дал еще большую привлекательность плате. Опять выстроились очереди, но пока удается сдерживать их в рамках двух месяцев.

#### **Open source & open hardware.**

Это первый коллективный железный проект NedoPC. Раньше мы делали проекты коллективом не более двух человек, а тут сразу четыре человека. Пришлось перейти на современные «нано-рельсы», то есть создать полноценный репозиторий для исходных

кодов, документации и прочего. Также размещено зеркало репозитория в интернете. От себя скажу, что это повышает ответственность и теперь не отделаешься «говнокодом» на скорую руку — «заклюют» свои же товарищи, да и сообщество добавит «пинков».

Но главное не это, открытость кода позволяет сторонним разработчикам, не согласным с общей концепцией проекта, создавать и развивать свои конфигурации. И мы не грязнем в вечных религиозных спорах о спектре, а делаем то что хочется. Уже созданы конфигурации Scorpion от Evgeny7, а также конфигурация с новыми графическими возможностями от TSLabs. Более того, появились пользователи именно этих альтернативных конфигураций. Я как разработчик всячески приветствую это начинание, перерастающее в самостоятельное движение. Значит «линия партии» была выбрана правильно и за opensource, несомненно, большое будущее. Призываю хобби-проекты делать именно opensource – это не только дисциплинирует, но и позволяет учиться у других и конечно делиться своими идеями.

Не секрет, успешность альтернативных конфигураций, добавляя «огня» разработчикам и вдохновляет на исправление имеющихся ошибок, в чем тоже их большая польза.

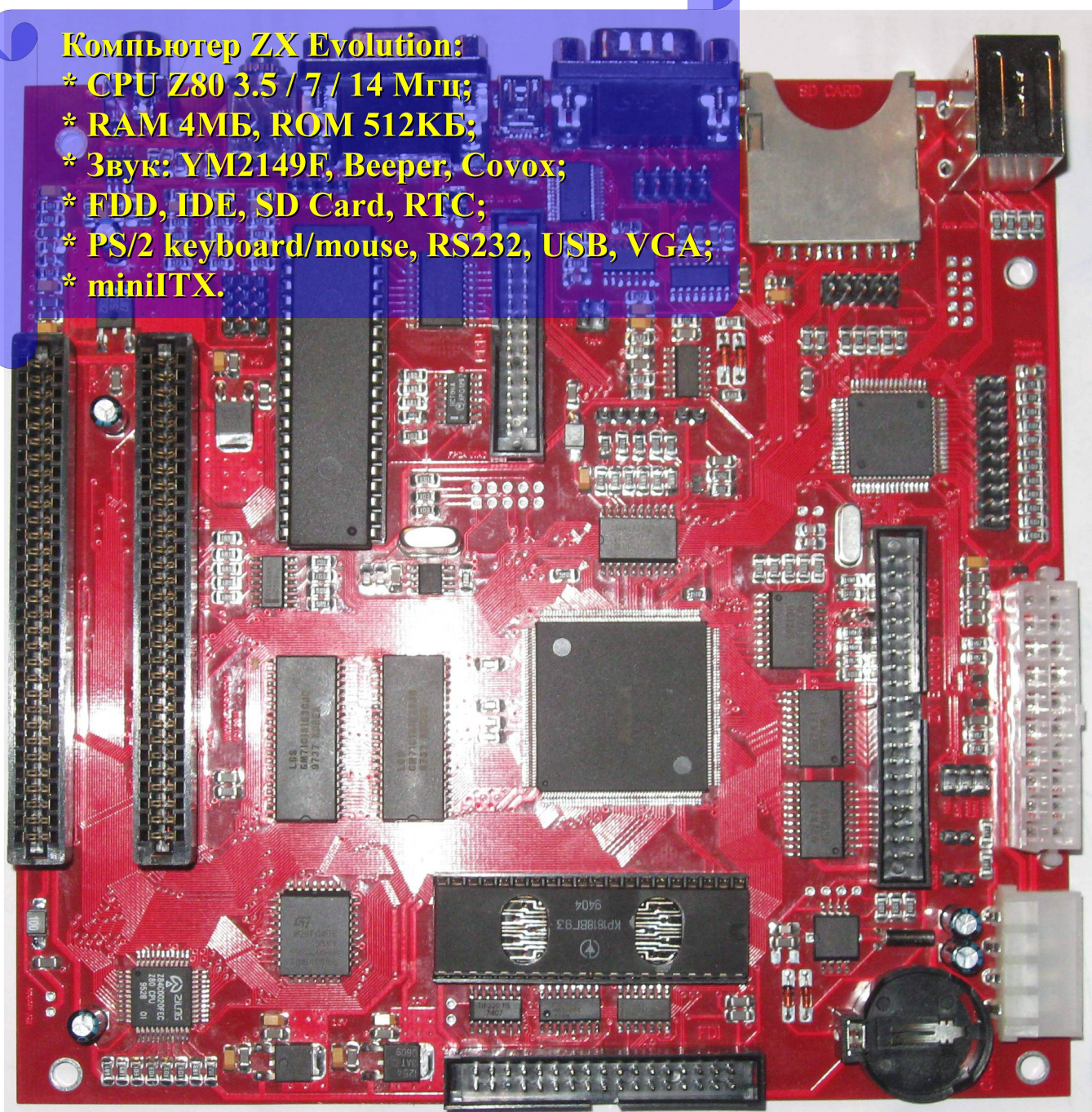
#### **Сообщество.**

Скажу честно — удивлен тем, что интерес к Spessу растет и ширится само сообщество. По государственным каналам показывают сюжеты о выставках и мероприятиях. Онлайн-журнал Computerra и некоторые другие издания регулярно публикуют статьи и обзоры. ZX Evolution уверено шагает за рубеж — если наличие пользователей из бывшего соцлагеря не удивляет, но то что пользователи с родины Spessу признали наш проект — радует.

Спасибо Вам за то, что Вы не даете угаснуть этому, теперь уже большому, увлечению!

**Компьютер ZX Evolution:**

- \* CPU Z80 3.5 / 7 / 14 МГц;
- \* RAM 4МБ, ROM 512КБ;
- \* Звук: YM2149F, Beeper, Covox;
- \* FDD, IDE, SD Card, RTC;
- \* PS/2 keyboard/mouse, RS232, USB, VGA;
- \* miniITX.



**WWW.NEDOPC.COM**  
**WWW.NEDOPC.ORG**  
**WWW.TERNARY.INFO**  
**WWW.VNTB.RU**