

DOWNGRADE



n28'2019

```

      10      0      0      0
00001      11      0110001      10
001100101011001      10      1001111001      1      1110111110101
010110001001101101100      001      00      000011001      00      00      10010100010011100100
111100001000110000101000011      10      0100      11110000001      00      010100101110110001001111110
101011000111101010010101100100000      01011010011      1001111111001110101101000101101
10111111101001000100101001010001011100001110011011011110000000010111101011010
101000110101011001011111101001101110011110101001100111011011000101101111
10001001010001111001010101001111010100111001000010111001101110000
01000110011111100000111011010001100111001100110001110101000
0111110001000000111000100000010100001100
00010100001000101000111111100111011011110101000010001011000011101001010110100
001010101100001110011111011010100101001011100101100000100101011100111100010100101110
11000110100110111110110011100011101001101110000100100101010111110111100101000110000100
010111011100011011011111101110001000011      010      0010001100111101000010000111011100000011
0111111010101111110111110101100111111      011      100010001100100011001000000001111001
01101101010010001000000010      11      000      1      1010100111111000011000100001
010
010
110
111
111
001
010
010
101
100
001
100
111
110
1101
1111
0000
010
111
100
```

СОДЕРЖАНИЕ

● Обложка -----	1
● Содержание -----	2
● От редактора -----	3

ТЕОРИЯ DOWNGRADE

● Новости, события, комментарии (uav1606&eubpc)-----	4
● Интервью с Виктором Фигурновым (В.Э. Фигурнов, uav1606, В.Рытиков aka eubpc, Андрей Шаронов aka Andrei88)-----	6
● Киберсин. Взорванное будущее (Олег Павлов)-----	13

DOWNGRADE-ВИДЕО

● Downgrade-видео: разное. (А.Шаронов, А.Чуклинов) -----	17
--	----

КНИЖНАЯ ПОЛКА

● Рецензия на книгу: Тристан Донован. «Играй! История видеоигр» (Андрей Тумилович) -----	19
● Downgrade-машина в физическом кабинете (А. Шаронов)-----	23

DOWNGRADE-ЖЕЛЕЗО

● СМ18330 - ещё одна ISA-звуковуха в строю (А. Шаронов) ----	25
● ESS688 - снова бюджетный звук на 486-м компьютере (Андрей Шаронов aka Andrei88) -----	30

DOWNGRADE-СОФТ

● Цифровой звук на БК-0010 (А. Мачуговский aka Manwe)-----	34
● Доктор твоего модема (Андрей Шаронов aka Andrei88) -----	47

ПРОГРАММИРОВАНИЕ

● Разработка программ для текстового редактора NewsMaster («Журналист»). Ч.2 из 3: драйвер для принтера (SysTools)--	54
---	----

ИНТЕРНЕТ И СЕТИ

● Календарь по мотивам истории фидонет (В.Фёдоров) -----	70
--	----

СТАРЫЕ ИГРЫ

● Йааа король дороги!!! (Елена Шаронова) -----	78
● Галопом по Европам (Андрей Шаронов) -----	82

РАЗНЫЙ ЮМОР

● Просто разный юмор -----	88
● Над номером работали -----	90

От редактора

Приветствую вас на страницах №28 журнала Downgrade!

Честно говоря, я ожидал, что статей в этот раз пришлю мало – летом обычно так и бывает.

Но этот номер стал исключением – статей более чем достаточно. Есть и новые авторы – Александр Мачуговский (**Manwe**) с очень интересной статьёй о звуке на БК и Елена Шаронова со статьёй про игру «Дальнобойщики 2».

Также в этом номере редколлегия взяла интервью у В.Э. Фигурнова – думаю, многие читатели знают его серию книг «IBM PC для пользователя». Я и сам учился компьютерной грамотности по ним, так что был вдвойне рад поговорить с Виктором Эвальдовичем. Кстати, огромная ему благодарность за это интервью.

В общем и целом, я бы сказал, что номер вышел удачный – заполнены все рубрики, объём более чем солидный. Надеюсь, что и следующий не подкачает.

Как обычно, если у вас есть предложения, замечания, статьи, какое-то творчество на тему Downgrade – присылайте мне на e-mail [uav16060 \[Cerberus\] mail.ru](mailto:uav16060@Cerberus.ru)

uav1606

НОВОСТИ, СОБЫТИЯ, КОММЕНТАРИИ

50 лет Unix

В августе 1969 года Кен Томпсон и ещё несколько сотрудников Bell Labs создали прототип операционной системы Unix, предназначенный для работы на DEC PDP-7.

Первым названием системы было Unixplexed Information and Computing Service — UNICS, которое позже превратилось в Unix.

Первые версии ОС были написаны на ассемблере. Примерно тогда же Кен Томпсон при содействии Денниса Ритчи разработал язык высокого уровня Би (B), позже ставший основой всем известного Си.

Doom и Doom II на смартфонах

К 25-летию Doom Bethesda Softworks выпустила мобильные версии первой и второй части этой легендарной игры. Пользователям доступны версии для iOS и Android (скачать последние можно [тут](#) и [тут](#)).

К сожалению, игры не бесплатные, их цена составляет 379 рублей.

Doom уже выходила на мобильных платформах, но сейчас доступна расширенная версия с новым эпизодом Thy Flesh Consumed, включающим 9 уровней. Doom II также включает 20 созданных фанатами Master Levels.

К iPhone подключили мышь и клавиатуру от первых Mac'ов

Niles Mitchell заинтересовался новой функцией iOS 13 — поддержкой USB-мыши. Чтобы её проверить, он решил подключить к iPhone одну из ранних мышей от Apple — M0100, а также клавиатуру M0110. Эти устройства использовались в оригинальных «Макинтошах» 80-х годов.

Конечно, Niles подключал эти устройства не напрямую, а через переходник на USB.

Видеобзор от автора можно посмотреть здесь:

<https://youtube.com/watch?v=RL7f7dDBvXA>



Умер Рудольф Лов

11 сентября на 67-м году жизни умер Rudolph R. Loew (**RLoew**).

Downgrade-сообществу он в первую очередь известен своими патчами для Windows 9x — например, для снятия ограничения на объём RAM, решения проблем с большими дисками, для поддержки AHCI и т.д.

Сайт с его программами доступен через Web-Archive:

[http://web.archive.org/web/20190812081523/
http://rloew.x10host.com/](http://web.archive.org/web/20190812081523/http://rloew.x10host.com/)

Анонсирована System Shock 2: Enhanced Edition

К 20-летию игры разработчики студии Nightdive анонсировали выпуск улучшенной версии System Shock 2. Как обычно в таких случаях, планируется усовершенствовать графику, мультиплеер и т.п.

Точная дата выхода System Shock 2: Enhanced Edition пока неизвестна. Оригинал новости [здесь](#).



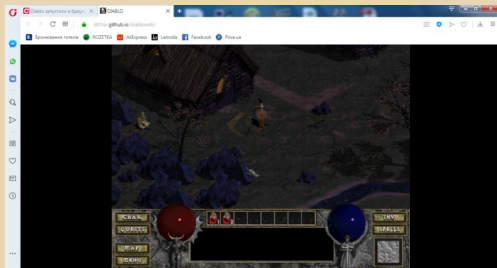
```
;-) :-| :-> :-) %-)
:*) B:-) B-) 8:-) :-{)
:-E :-* :-(- :-@ :-/
X-( :-[ :-] :-D :-P
```

День рождения смайлика

19 сентября 1982 года считается днём рождения классического смайлика. Именно тогда Скотт Фалман из Carnegie Mellon University предложил использовать двоеточие, дефис и закрывающую скобку для обозначения шуточных высказываний.

То есть в этом году смайлику исполняется 35 лет.

Несмотря на всеобщее засилье анимированных эмодзи, текстовые смайлики по-прежнему сохраняют своё очарование, при этом они требуют меньше ресурсов для своего отображения и могут использоваться в чисто текстовых сообщениях, где по тем или иным причинам нельзя использовать изображения.



Diablo в браузере

Благодаря компании Rivsoft теперь любой желающий может поиграть в оригинальную версию Diablo 1996 года прямо в браузере.

Онлайн-версия доступна здесь:

<https://diablo.rivsoft.net/>

Вы можете сыграть в shareware-версию с ограничениями (только один персонаж и два уровня) или загрузить свой файл **DIABDAT.MPQ** из полной версии.

К сожалению, Вам понадобится современный браузер – к примеру, на Mozilla Firefox 45 ESR игра работать отказалась.



Умер Юрий Живов

22 августа 2019 скончался Юрий Викторович Живов – переводчик-синхронист, занимавшийся в 80-е – 90-е годы авторским переводом и озвучиванием зарубежных фильмов. В его копилке более 1000 переводов.

Хоть этот человек и не имел прямого отношения к Downgrade, но он, можно сказать, символизирует ту эпоху – VHS, видеосалоны, кооперативы, приставки Dendy, первые IBM PC-совместимые компьютеры и многое другое...



Duke Nukem на движке Serious Sam 3

Один из фанатов Duke Nukem 3D с ником **Syndroid** создал модификацию своей любимой игры под относительно новый движок Serious Sam 3 (вышел в 2011-м году). Ремейк получил название Serious Duke 3D. На данный момент он включает обновлённый и довольно существенно переделанный первый эпизод оригинального Duke Nukem. Для ремейка необходим Serious Sam 3, установленный через Serious Sam Fusion.

Скачать модификацию и посмотреть обзор можно [здесь](#).

Обзор подготовили:
Вячеслав Рытиков (euбрс)
uav1606



ИНТЕРВЬЮ С ВИКТОРОМ ФИГУРНОВЫМ



Вашему вниманию предлагается интервью с Виктором Эвальдовичем Фигурновым. Многим читателям журнала он знаком как автор легендарных самоучителей по работе на ПК (серия «IBM PC для пользователя» и др.). Думаю, трудно найти человека, который бы не учился работать на компьютере по этим книгам, ну или хотя бы просто не видел их или не держал в руках. Виктор Эвальдович любезно согласился ответить на вопросы редакции и рассказать о своей жизни, книгах и других интересных вещах.

Расскажите немного о себе – сколько Вам лет, где родились, какое образование получили и т.д.

Родился в 1961 году в Москве, окончил физико-математическую школу № 2 (теперь это лицей «Вторая школа»), потом механико-математический факультет МГУ и его аспирантуру. В 2013 году получил второе высшее образование — окончил Финансовый университет при Правительстве РФ по специальности «Финансы и кредит».

В Википедии написано, что Вы гражданин Латвии, но живёте в Москве?

Всё верно.

Каким был Ваш первый компьютер? Можете рассказать историю знакомства с ним?

Первый компьютер, с которым мне пришлось иметь дело, это был компьютер «Мир 2». Мне тогда было 12 лет. Я тогда посещал компьютерный кружок во Дворце пионеров, на Ленинских (ныне Воробьёвых) горах в Москве. Участникам этого кружка раз в неделю разрешали работать на компьютере. Программы в компьютер «Мир-2» вводились с помощью консоли, это что-то вроде электрической пишущей машинки, подключённой к компьютеру. И с помощью перфолент. Но компьютер был очень интересный: его встроенный язык АНАЛИТИК сочетал возможности языка общего назначения и системы компьютерной алгебры. Он позволял выполнять алгебраические преобразования, вычислять интегралы и производные в аналитическом и в числовом виде, выполнять вычисления с произвольной точностью и др. То есть, он позволял решать очень разные задачи и был очень хорошей базой для знакомства с компьютерами.



С чего началась Ваша карьера как автора компьютерных учебников? Как появилась идея написать первую книгу? Или, быть может, сначала были статьи или что-то другое?



Идеи писать книги по работе с компьютерами у меня не было. Книга «IBM PC для пользователя» появилась практически сама собой. Это было в 1989 году. Тогда я работал в совместном предприятии «Интерквадро» старшим научным сотрудником в отделе математических разработок. Статьи у меня были, но по математике, а не по компьютерному делу. Но для нужд отдела нужно было вводить в компьютер много разных текстов. Программы для распознавания текстов, которые справлялись бы с русским текстом и математическими формулами, тогда не было. Поэтому в отдел приходили девушки и вводили эти тексты вручную. Компьютеры тогда были в диковинку, работать с ними почти никто не умел. Девушек надо было учить, и эта приятная обязанность была поручена мне. Я объяснял, как включать компьютер и как выключать, как запускать программы, как вводить тексты, как их оформлять, как копировать документы на дискеты и многое другое. Но девушки часто менялись и плохо запоминали. И я скоро понял, что мне проще написать инструкцию, чем объяснять одно и то же десятки раз. Скоро таких инструкций стало много, я объединил их, и получилась небольшая книжка, страниц на 150. Но на этом дело не закончилось. В соседних с нашим отделом помещениях находился учебный центр фирмы. Я показал эту книжку руководителю учебного центра. Он одобрил и сказал, что хорошо бы описать ещё несколько тем. Я описал, и в книжке стало примерно 450 страниц, а потом и больше. Из них потом и получилась книжка «IBM PC для пользователя», которая была издана издательством «Финансы и статистика».

Трудно было в то время найти издателя для «IBM PC для пользователя»?

А я не искал издателя и даже не думал об этом. У меня с детства было убеждение, что издать книгу очень трудно. Так было в среде моих родителей и их знакомых-экономистов. Об издании книги говорилось как об огромном успехе и достижении, для которого нужны

чрезвычайные усилия. Даже очень известные и авторитетные учёные дожидались издания своих книг по нескольку лет. А начинающим, никому не известным авторам, да ещё без покровителей, это было практически невозможно.

А получилось совсем иначе. Когда из нескольких инструкций я составил самоучитель по работе с компьютером, то многие друзья и знакомые стали просить его почитать. Учебников по работе с компьютерами типа IBM PC тогда не было, а научиться хотелось многим. Я распечатывал книжку на матричном принтере, переплетал на пластмассовую спираль и отдавал. И однажды один из моих хороших знакомых, математик Борис Васильевич Черкасский, показал эту книжку своему знакомому, заведующему редакцией издательства «Финансы и статистика» Константину Коробову. Тот связался со мной, познакомил меня с директором издательства Алевтиной Николаевной Звоновой, и моя книга была во внеочередном порядке включена в план и издана.

Какое программное обеспечение Вы использовали для создания Вашей первой книги? Вёрстку Вы делали сами? Если да, то какие программы для этого использовались? Чем делались скриншоты? Какие другие утилиты использовались?

Я пользовался редактором Microsoft Word для MS-DOS. И верстал в нём же. Скриншоты снимались клавишей **Print Screen**, специальные программы для этого мне тогда были неизвестны.

А какой тогда у Вас был компьютер? (Модель, характеристики)?

Компьютер при написании книги был стандартный для того времени: IBM PC AT/286, 16 MHz, 1 MB RAM, 40 MB HDD, дисководы для 3- и 5-дюймовых дискет, монитор VGA 14" с разрешением 640x480.



Расскажите о процессе создания книги – какие были трудности, много ли времени это занимало?

При написании книги особенных трудностей не было. Оно заняло примерно полгода в свободное от основной работы время. Сейчас я думаю, трудностей не возникло потому, что я неосознанно подражал стилю лучших учебников по современной математике. Особенно книжке Атьи и Макдональда «Введение в коммутативную алгебру». Я старался писать кратко и ясно, без воды и только по делу. Следил, чтобы все используемые понятия были заранее определены, а до них были даны необходимые мотивировки. Чтобы читатель сначала понимал ситуацию, овладевал основными понятиями, узнавал, зачем и что ему нужно делать, а потом получал чёткие и однозначные инструкции, как это делать. А также примеры, если они необходимы. Этот способ изложения оказался подходящим для материала книги и был одобрен большинством её читателей.

Трудность возникла потом. После того как рукопись была одобрена, меня попросили сделать вёрстку и напечатать оригинал-макет книги на лазерном принтере. Иначе, дескать, издание книги будет отложено не менее чем на полгода. И я смело пообещал, что сделаю это за две недели. Но на самом деле я не умел делать вёрстку, совершенно не представлял себе объём и сложность этой работы, и у меня не было лазерного принтера. К счастью, сотрудники редакции очень чётко и доходчиво объяснили мне, что им требуется, а лазерный принтер нашёлся у знакомых. И я смог справиться с этой задачей. После этого все свои книги я верстал сам.

Не знаете, каким суммарным тиражом разошлись Ваши книги? Переводились ли они на какие-либо иностранные языки?

Суммарный тираж примерно 3,5 миллиона экземпляров. Был перевод на литовский язык.

Во втором издании «IBM PC для пользователя» в конце книги упоминается некий расширенный комплект документации и программ «Работа пользователя с IBM PC». Что он собой представлял, что входило в состав комплекта?

Издательство «Финансы и статистика» установило объём моей книги 240 страниц в первом издании и 288 страниц во втором. А у меня было материала в 2,5 раза больше. Поэтому я и подготовил этот комплект документации и программ. Туда входили три книги о работе с компьютерами и дискета с программами и шрифтами. Комплект продавался организациям по заказам.

Сколько вообще вышло изданий книги «IBM PC для пользователя»? У Вас были ещё какие-то книги, статьи на компьютерную тематику (кроме упомянутой тут статьи в «Радио» о Conan)?

Изданий книги «IBM PC для пользователя» было 7, из них седьмое вышло в 2 вариантах: полном («IBM PC для пользователя») и кратком («IBM PC для пользователя. Краткий курс»).

Кроме них, были три маленькие книжки в 1990-е годы в издательстве «Наука» в серии «Библиотечка пользователя IBM PC».

И в 2005 году были две книги про Windows: «Windows для начинающих» и «Windows для начинающих и опытных».

Статей было, наверное, с десяток. Но я уже и забыл про них, где и про что. Недавно с удивлением нашёл [PDF-файл](#) номера «Компьютер-Пресс» за 1993 год, № 10 со своей статьёй «Размышления о компьютерной литературе» (с. 61-64). Самому было интересно прочесть и вспомнить.



Возможно, Вы участвовали в каких-то выставках, презентациях или конференциях – расскажите о наиболее запомнившихся Вам.

В 1990-е годы я участвовал во многих выставках: Softool, «Комтек» и др. Но больше всего мне запомнились не они. В 1994 году я написал по заказу фирмы «1С» документацию к «1С:Бухгалтерии», и меня пригласили на семинар партнёров «1С». И он мне запомнился больше всего — там собрались сотни людей, объединённых общим делом, там был заинтересованный обмен опытом, коллективная работа на общий результат.



Обложки книг В.Э. Фигурнова



Главный редактор издательства «Диалектика» (Киев) Виктор Штонда и Виктор Фигурнов на выставке Softool 1994



Андрей Борзенко, Виктор Фигурнов, Геннадий Петриковец и Виктор Штонда (издатели и журналисты) на выставке Softool 1994



Поступали ли в ваш адрес предложения включить за определённое вознаграждение в книгу рассказ о каком-нибудь конкретном ПО? (Скажем так – сделать «заказной» обзор?)

Таких предложений не было. Но в моих книгах можно было разместить рекламу на отдельных страницах.

Случались ли в вашей профессиональной деятельности какие-нибудь курьёзы, забавные случаи?

Когда я готовил к публикации 7-е издание своей книги «IBM PC для пользователя», в моём компьютере умер жёсткий диск. А резервные копии были очень старые. Издатель предложил отправить жёсткий диск в США, в самую лучшую лабораторию по восстановлению данных. Диск послали с помощью очень уважаемой курьерской службы. Но где-то по пути курьеры этой замечательной службы выронили этот диск, и по нему проехал автомобиль. Восстановить данные стало невозможно. Так что мне пришлось писать всё заново. Это очень меня расстроило, но текст книги от этого стал только лучше.

Через пару лет в том же издательстве я рассказал эту историю авторам, имевшим отношение к «компетентным органам». Они сказали: «что же вы к нам не обратились»...

Вы в то время общались с создателями отечественного софта или вообще с людьми, имеющими отношение к IT? Скажем, с Е. Веселовым, создателем «Лексикона», авторами отечественных антивирусов или с кем-то ещё?

До написания книги я был разработчиком программного обеспечения, и большинство моих коллег имело отношение к IT. Но с Евгением Веселовым, Евгением Касперским, Игорем Даниловым и Дмитрием Лозинским я не знаком.

Почему, на Ваш взгляд, для IBM PC появились такие отечественные программы, как «Лексикон», «Русское слово», «Лингво», программы распознавания текста, но при этом

ничего подобного не было для DEC-совместимых компьютеров – хоть для БК и УК НЦ, хоть для ДВК или «Электроники 85». Также как-то не было видно попыток создать подобный софт для «Радио 86PK» или «Ориона». Почему именно PC?

Вы называете программы очень разного класса. «Русское слово» – это была небольшая программа — русификатор для Microsoft Word. Русификаторы программ для DEC-совместимых компьютеров были, хотя названий я уже не вспомню. А вот «Лингво», FineReader и другие программы распознавания текста — это большие программы. Вряд ли их можно уместить в БК, ДВК или «Электронику 85». Создание таких программ требует работы большого коллектива программистов, исследователей, менеджеров и других специалистов в течение многих месяцев и лет. Это очень дорогое мероприятие. И естественно, что люди, вкладывающие в это деньги, думают о том, окупится ли это, и если да, то когда. Пользователей IBM PC-совместимых компьютеров в России, а тем более в мире, было в сотни или тысячи раз больше, чем пользователей БК или ДВК. Поэтому и предпринимателей, готовых тратить свои деньги и свои силы на проекты разработки программ для IBM PC-совместимых компьютеров было в сотни раз больше.

Судя по воспоминаниям живших тогда людей, в народе большой популярностью пользовалась операционная система OS/2, но среди российского коммерческого софта можно только назвать VentaFax, авторы которой упоминают возможность работы программы под «полуосью». Как так получилось? Народ пользовался OS/2, а корпоративный сектор предпочитал Windows? Или же популярность OS/2 тоже преувеличена?

OS/2 пользовалась большой популярностью в медиапространстве. Было много статей в журналах, обсуждений. Как сейчас говорят, было много хайпа. Но этот широкий интерес не привёл к массовому использованию. Дело в



том, что у OS/2, при несомненных технических достоинствах, были высокие требования к аппаратуре, малое количество поддерживаемых устройств, ограниченный набор написанных для неё программ и глючный режим совместимости со старыми программами. Поэтому для большинства пользователей и организаций переход на OS/2 был бы дорогостоящей авантюрой с неясными преимуществами. В России я видел OS/2 всего несколько раз, и почти во всех случаях это были серверы. Для серверов описанные недостатки OS/2 были не столь существенны, а были важны надёжность и способность выдерживать высокие нагрузки, что OS/2 довольно хорошо обеспечивала.

Операционная система Windows тех лет не отличалась техническими достоинствами OS/2. Она не обеспечивала настоящую изолированность выполняемых программ, частенько зависала, не могла выдерживать высокие нагрузки. Она была пригодна только для настольных компьютеров, но не для серверов. Но она предоставила массовым пользователям намного более важные для них качества: возможность использования многозадачности и графического интерфейса при невысоких требованиях к аппаратуре, огромный набор программ самого разного назначения и очень широкую поддержку устройств. И практически все старые программы тоже работали. То есть, пользователям предоставлялась масса дополнительных преимуществ, и от них не требовался отказ от того, что у них было ранее. И пользователи это сразу оценили. За первый месяц после выпуска было продано больше копий Windows 3.0, чем было продано копий OS/2 за три предшествующих года.

Вскоре фирма Microsoft договорилась с множеством производителей персональных компьютеров о предустановке Windows на новые компьютеры, с тысячами производителей компьютерных устройств о разработке драйверов для Windows. Десятки тысяч производителей программ стали писать версии своих программ для Windows. В итоге, фирма Microsoft ещё в начале 90-х годов выиграла борьбу за рынок операционных систем для IBM

PC-совместимых компьютеров персонального использования и захватила более 95% рынка.

За OS/2 остались ниши операционных систем для серверов, для высокопроизводительных рабочих станций, для встроенных устройств типа банкоматов и др. Но во второй половине 90-х годов её вытеснили и оттуда. С одной стороны Window NT и её потомки, а с другой – различные версии Linux, FreeBSD и других операционных систем с открытым кодом.

Поддерживали ли Вы отношения с компанией «МикроАРТ»? Просто в книжке, выпущенной товарищами из данной компании, описывается программа «Конан», о которой Вы писали в журнале «Радио» (№6 за 1993 год), а также упоминается Ваша книга.

Насколько я помню, с этой фирмой поддерживал отношения автор программы Conan Алексей Кулаичев. Эта программа позволяла анализировать в реальном масштабе времени электрофизиологические сигналы, а для их ввода в компьютер нужны были электронные платы аналого-цифровых преобразователей. Компания «МикроАРТ» производила такие платы, и программа Conan продавалась с ними в комплекте.

Каково вообще Ваше мнение о тогдашних разработках класса «Продвинутый Спектрум» – типа ATM Turbo или «Скорпиона»? Виделись ли они Вам изначально «мёртворождёнными» или же могли занять какую-то нишу?

Если бы эти компьютеры были созданы лет на 8 раньше, то они могли бы стать очень успешными проектами. А в 1991 году уже нет. Прототип этих компьютеров, ZX Spectrum, в 1992 году был снят с производства как устаревший.

Периодически в статьях и книгах упоминаются перепайки, сопряжения, переделки отечественного оборудования для работы с IBM PC-совместимыми компьютерами. Обычно это были принтеры, мониторы, клавиатура МС7004. Носило ли это явление какой-то массовый характер?



Трудно ли тогда было достать соответствующее «железо»?

В воспоминаниях также можно прочитать об удачных попытках разработки материнских плат для IBM PC/XT – «Винница» и XL7. Возможно, что и Вам известны какие-то подобные разработки, которые делались в Москве? Насколько были успешны и распространены подобные материнские платы и компьютеры?

Скорее всего, эти перепайки и переделки были единичными опытами. Про материнскую плату «Винница» я ничего не слышал, а XL7 – это плата сингапурской фирмы Micoms. Возможно, в России в начале 90-х гг. продавались её самодельные копии (<http://ebay.de/i/181633832534>).

Насколько я знаю, одно из последних изданий «IBM PC для пользователя» вышло в 2006 году? Не собираетесь в ближайшее время написать ещё какую-нибудь книгу? Какие Ваши творческие планы?

Пока планов писать книги у меня нет. Рынок компьютерной литературы сейчас очень изменился. Базовые навыки работы с компьютерами молодое поколение приобретает ещё в школе, а то и до неё. Востребованы в основном книги по отдельным современным компьютерным технологиям, они издаются очень небольшими тиражами.

Чем Вы занимаетесь сейчас, где работаете?

Работаю в фирме «1С», отвечаю за разработку документации для облачных сервисов.

Какое у Вас мнение о сфере IT сейчас? Вы стараетесь изучать и использовать всё то новое, что появляется в среде компьютерных технологий, или придерживаетесь более консервативного подхода?

Сфера IT сейчас является одной из наиболее важных областей деятельности в современном обществе. Она сильно изменила и меняет нашу повседневную жизнь. Мы узнаём новости

в интернете, общаемся в социальных сетях, ищем нужные сведения с помощью «Гугла» и «Яндекса», платим квартплату через Интернет, покупаем товары в интернет-магазинах, смотрим школьные отметки детей в электронном дневнике, вызываем такси через мобильное приложение и так далее. Сервисы типа Госуслуг кардинально меняют наши взаимоотношения с государственными структурами. Всё это делает нашу жизнь удобней и насыщенной, экономит наши время и силы, увеличивает наши возможности. Внедрение IT-технологий увеличивает производительность труда. И уровень жизни, который будет в нашей стране через 10-20 лет, сильно зависит от её конкурентоспособности в области IT-технологий.

Изучать всё новое в сфере IT вряд ли возможно, нового слишком много. Я стараюсь просматривать IT-новости, а детально изучаю лишь то, что меня заинтересовало, или что может быть полезно для проектов, которыми занимаюсь.

Каково Ваше отношение к движению Downgrade? Например, к проектам выставок старого «железа» или к архивным проектам, таким как Old-DOS.ru или Archive.org?

Я бы ещё добавил <http://www.computer-museum.ru/>. Отношение очень положительное, как и к другим историческим проектам. Образованный человек должен знать историю развития человечества, его культуры и его технологий, в том числе и компьютерных технологий.

Может быть, у Вас будут какие-то пожелания читателям журнала?

Желаю им, чтобы интерес к истории компьютерных технологий сочетался с интересом к современным компьютерным технологиям, умением их использовать и развивать!

В.Э. Фигурнов

Вопросы составили:

uav1606

Андрей Шаронов (Andrei88)

Вячеслав Рытиков (eubrc)



КИБЕРСИН. ВЗОРВАННОЕ БУДУЩЕЕ



Иногда, казалось, полузабытые, малоизвестные истории в дальних уголках Ойкумены рассказывают о том, каким должно было быть наше будущее.

Каким оно было, и каким, несомненно, станет.

Таким локусом будущего был проект Киберсин (Cybersyn), реализованный в Чили.

В 1970 году выборы в этой стране выиграл социалист Сальвадор Альенде.

Социально-экономически Чили немногим отличалось от большинства стран Латинской Америки. Если в других странах континента заправляли землевладельцы-латифундисты, то в Чили, крайне богатой минеральными ресурсами, — промышленники и военные.

Эта прослойка составляла (да и составляет ныне) не более 3% населения, и отличалась просто запредельным уровнем личного потребления.

Альенде как социалист попытался совместить государственное регулирование с народным самоуправлением. В отличие от частично реализованной ОГАС, Киберсин был реализован в железе и показал свою эффективность.

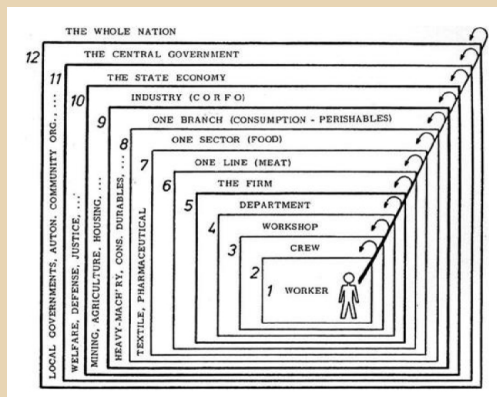
Немного о структуре Киберсин

Она базировалась на идеях Стаффорда Бира. Он рассматривал экономику в надсистеме любой страны, в системе фирмы, в подсистеме производства как аналог живого организма, с глубокими обратными связями.

Модель этой системы (Viable System Model — VSM) основывалась на принципах нервной системы человека, где мозг поддерживает общую работоспособность организма (гомеостаз).

То есть в организме не существует ни главных, ни второстепенных органов. Функции рук, ног, желудка так же важны, как мозга или сердца. Сальвадор Альенде, как врач по профессии, поддержал эту идею.

В своё время Бир написал книгу «Мозг фирмы», посвящённую этой теме.



Позднее Бир вспоминал: «Я впервые видел, чтобы человек на такой высокой должности понимал меня с полуслова».

Президент Чили с симпатией относился к преобразованиям в СССР, однако понимал недостатки жёсткого администрирования без обратной связи у нас.

Но одновременно ему не нравилась и запредельное потребление олигархии за счёт всего общества.

Структура Киберсина

С. Биром при помощи Альенде и поддерживаемой ими команды из чилийцев и англичан были реализованы следующие проекты.



Cyberstride — система, осуществляющая обработку реальных данных от фабрик и производств и логистики. Она управлялась двумя компьютерами IBM 360/50 и Burroughs B3500 и 200 терминалами.

Cybernet — сеть связи, которая соединяла правительство и основные узловые точки национализированных заводов и фабрик, путей сообщения и складов готовой продукции.

CHECO (Chilean Economic Simulator) — модель экономики Чили, где можно было проверить (хотя бы частично) правильность принятых решений.

Для оперативного управления происходящими процессами планировалось создать **Opsroom** — операционную комнату, откуда осуществлялось управление Cybersyn. (Была создана и позднее уничтожена Пиночетом.)

Cyberfolk — система обратной связи с населением (опробована в работе, но не введена из-за переворота).

Подробнее о каждом из проектов.

Cyberstride

Статистическая система Cyberstride должна была ежедневно обрабатывать данные и строить прогнозы состояния производств. Она базировалась на передаче данных от рабочих и мастеров, знавших реальное положение дел на производстве.

Программа на основе анализа предыдущих данных в реальном времени выдавала прогноз на будущее изменение (линейное, аномальное, нормальное).

Этот метод позволял по прошлым изменениям данных о производстве предсказать будущее изменение. ПО не просто записывало и обрабатывало статистические показатели, а

делало прогнозы, изменяя их «на лету» при поступлении новых данных. Программное обеспечение базировалось на уравнениях Харрисона-Стивенса, описывающих эти процессы. Статья, которая описывала эти уравнения, была издана в декабре 1971 года, а ПО заработало в январе 1972 года (!).

Были созданы модели 48 предприятий, 23 фабрик, создавались модели предприятий лёгкой промышленности и транспорта.

Вот это скорость внедрения!!!!

Cybernet

Система строилась на 2 (двух!) компьютерах и 200 терминалах, откуда шли данные в центр.

Велась работа по подключению Cybernet к Cyberstride для передачи данных из сети напрямую в ПО.

CHECO (Chilean Economic Simulator)

Запущен в марте 1972 года.

Первоначально обрабатывал статистические данные с производств, но к ноябрю 1972 года отслеживал также курсы валют, уровень инвестиций и инфляцию.

В общей сложности обрабатывалось 74 временных экономических показателя, и готовилось ПО ещё на 180 показателей.

Cyberfolk

С. Бир предложил систему обратной связи с населением. На ~~тёплых ламповых~~ аналоговых «альгедонических счётчиках» — шкалах, показывавших настроения людей. Название прибора происходило от греческого algos — «боль», и hedone — «удовольствие». Устройства служили для измерения эмоциональной реакции людей и последующей отправки этих данных в центр. Данные поступали на вход системы Cyberstride, отображаясь на одном из экранов. Результаты предполагалось сделать общедоступными для любого жителя страны.



Opsroom

Комната принятия решений. Отличалась революционным «космическим дизайном».



Экраны представляли собой плоские дисплеи, на которые с обратной стороны проецировались информационные слайды. Графики и схемы рисовались вручную, фотографировались и переводились в слайды «девочками-дизайнерами». ☺

Достаточно было нажатия кнопки, и на четыре экрана выводились картинки и диаграммы, предоставлявшие статистические данные по разным проблемам.

Управление проектами было встроено в ручку кресла, что ясно из прилагаемого фото.

Этот дизайн был сделан задолго до Win и Mac. И без копирайта.

По срочным проблемам был отдельный стенд, где красным цветом выделялись необходимые срочные вопросы.

Статистические экономические данные можно было распечатать или вывести на экран по первому запросу. Этому помогали 20 теле-тапов.

Руководил всем этим проектом с чилийской стороны Фернандо Флорес.

Здесь надо сказать о конфликте интересов Бира и Флореса.

Бир предлагал разработать новую программу Beat-o'clock, расширяющую функции Киберсина. Флорес выступил против. Более того, став

впоследствии министром финансов, он стал сознательно тормозить новые разработки.

Планировалось создавать оперативные комнаты Cybernet на низовых уровнях, где, после подтверждения их полномочий в Сантьяго, можно было распределять сырьё, продукцию, ремонты, сообщения о чрезвычайных ситуациях на местах. В определённом коридоре значений система регулировалась автоматически и не требовала вмешательства человека. Если же машина не справлялась – давался сигнал. Если на своём уровне и человек не справлялся с ситуацией – сигнал передавался на высший уровень.

Подобную систему в СССР предлагал В.М. Глушков. Тем самым мы получали элементы прямой демократии сверху донизу, а не её имитации.

Самое главное новшество системы в том, что она отрицала то, что называется волюнтаризмом. В любой момент, при своеобразном «интернете вещей», мы сможем увидеть, как идут финансовые и материальные потоки, где находится груз и т.д.

Впоследствии работы Бира использовались на низовом уровне многими консалтинговыми фирмами на Западе. Как консультанта его использовали правительства Канады, Мексики, Парагвая, Бразилии.

Был разработан Proyecto Urucib (Uruguay + Cybernetic) для Уругвая.

Используя работы Бира, американский профессор Терри Виноград и Фернандо Флорес написали в 1987 году книгу «Understanding Computers and Cognition», где пришли к пониманию ограничения кибернетики в творчестве. В ней впервые сформулированы общие принципы программ ИСТП (Интеллектуальных Программ Творческой Поддержки). Она до сих пор не переведена.

Интересный факт: учеником Винограда был Ларри Пейдж – один из соучредителей и гендиректор «Гугла».



Как был уничтожен Киберсин

«Чили управляется компьютером» — так называлась в 1973 года статья про Бира в британской газете The Observer.

«Компьютер управляет Кремлём» — статья «Вашингтон пост» о работе В.М. Глушкова.

Альенде покусился на прибыли мегакорпораций. Кроме того, в стране начались теракты ультраправых боевиков, взрывавших мосты, подстанции и линии электропередач.

В октябре 1972 года общенациональная забастовка дальнобойщиков парализовала подвоз продовольствия в ключевые точки страны. В последствии ЦРУ признало, что лидерам забастовщиков были выплачены немалые деньги за этот акт.

Благодаря Киберсину, имевшиеся у правительства 400 грузовиков предотвратили коллапс экономики.

Одновременно при давлении США происходили блокада, саботаж, диверсии и теракты.

Первоначально после захвата операционной комнаты один из офицеров порезал ножом слайды Киберсина.

Разработчики пытались во время допросов рассказать о структуре и программах Киберсина. Однако военные не поняли «яйцеголовых».

Позднее сама комната была взорвана.

После переворота С. Бир приложил много сил для освобождения тех, кто с ним работал.

Флорес был освобождён в 1976 году и стал позднее миллионером в Силиконовой долине.

Тим О'Райли (Tim O'Reilly), продолжая работы Бира, написал книгу об «алгоритмическом регулировании».

Брайан Ино, Дэвид Бирн и Дэвид Боуи (David Bowie) считают книгу С. Бира одной из своих настольных.

Оперативная комната Киберсина вдохновляла будущих дизайнеров «Эппл».

Программы, отслеживающие запросы в реальном времени, используют многие фирмы, в частности фирма такси Uber.

Вместо послесловия

Для того чтобы изменить что-то, не нужно ни суперкомпьютеров, ни сверхскоростного интернета. Нужно только желание и умение правильно анализировать происходящее вокруг себя. Однажды мы вернёмся. Вооруженные старыми компьютерами и новыми идеями... 640 килобайт хватит на всё! Но пасаран, камрадас! Венсеремос!

Олег Павлов



DOWNGRADE-ВИДЕО.



РАЗНОЕ



Представленная на суд общественности подборка основана на материалах, представленных Александром Чуклиновым. К сожалению, сам он пока что не высказывает желания оформлять результаты своих интернет-поисков в виде статей и заметок, но это не снижает ценности найденного. Итак:

Канал Сергея Фролова

Имя и фамилия Сергея Фролова известны читателям журнала. Тем более что в третьем номере журнала даже печаталось интервью с ним. Известен также читателям и сайт виртуального музея <http://www.leningrad.su/museum/main.php?lang=1> и блог <https://sfrolov.livejournal.com/>. Оказалось, что есть и YouTube-канал: <https://www.youtube.com/channel/UCWvURyJ3uiGkD-XvO3T2A>

Канал можно считать, думается, дополнением к записям в блоге. На канале также можно увидеть видео, посвящённые калькуляторам, электронным часам и другим коллекционным находкам Сергея, а также результатами его радиоловительского творчества, например, ролик о тестировании микросхем K565PY7, который можно увидеть сразу на главной страничке канала:

<https://youtube.com/watch?v=LWP9x5XBmlw>



Канал же был обнаружен благодаря ролику об электромеханической счётной машине (всё же ещё не калькулятору ☺) ВММ-2:

<https://youtube.com/watch?v=05ch2rAlaaw>



Думается, что канал станет прекрасным дополнением и к виртуальному музею, и к блогу. Хотя, почему станет? Уже стал. ☺

Канал Кирилла Лейфера

Ещё один канал, посвящённый downgrade-тематике. В этот раз не от такого именитого автора, но всё равно достаточно интересный. Располагается он по адресу <https://youtube.com/channel/UC4cKlgZhuJtmN66tNliM-1g>. Как оказалось, большей частью канал посвящён мобильным устройствам – особенно плейлист «Древние гаджеты». Представлены и ноутбуки, и карманные компьютеры, и даже Smart-Watch... под PalmOS. ☺ Канал был найден по ролику, посвящённому Epson PX-8:

<https://www.youtube.com/watch?v=M3jJazr8PX8>

Его собрату – «Таргумону» – была посвящена статья Юрия Литвиненко в Downgrade-журнале №18.



Однако не менее интересными оказались и другие видео. Например, Андрею Шаронову показались интересными ролики о коммуникаторах Nokia – [9110](#), [9210](#) и [7710](#).



Конечно, может быть, можно без проблем найти другие обзоры, но специально искать не особо хотелось. А тут попались на глаза обзоры, и с удовольствием посмотрел. Самое обидное, что в роликах достаточно мало внимания уделено софту, но всё же между роликом и статьёй есть разница, и то, что творится на экране, очень трудно отразить в статье. Всё же крутить перед камерой устройство и показывать манипуляции рук при использовании куда интересней, это очень трудно сделать в статейных иллюстрациях. Скриншоты же разного софта в статейных иллюстрациях куда более уместны. Так что, наличие обзора только разогло интерес к описанным коммуникаторам.

Канал MADDY Murk

Канал посвящён, в основном, сборке новых и относительно новых конфигураций, однако на канал привело совершенно другое видео:

<https://www.youtube.com/watch?v=XLG7jTnBYNU>

«Новый» комп на Windows 95 за 500 руб. Вам часто попадаются неиспользованные полнокомплектные компьютеры середины 90-х годов? Мне как-то ни разу не выпадало – максимум – запечатанные комплектующие – это было, но компьютера целиком – нет. А вот автору видео по сходной цене удалось купить практически новый компьютер. Конечно, компьютером пользовались, но пользовались, видимо, достаточно аккуратно.

Какая конфигурация попала автору видео?

- Pentium MMX 166 МГц на плате ASUS TXP4;
- 32 МБ оперативной памяти;
- Жёсткий диск Seagate 3.2 Гб;
- CD-ROM – вроде бы в видео промелькнуло, что NEC;
- Видеокарта Tseng ET-6000 2 МБ;
- Звуковая карта Creative Vibra-16.



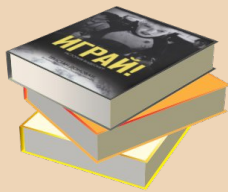
Компьютер, такое ощущение, какого-то отечественного бренда: мало того, что есть шильдик – много у кого есть, но ещё и установленные «ДИСКО Командир» и «Сократ», думается, скорее, поставлялись вместе с компьютером.

Кроме них на компьютере присутствует «Нортон» и ДОСовский «Лексикон», судя по ярлыкам, MS Office, MathCAD, а также несколько игрушек. Автор запустил автосимулятор, но не менее интересно было бы посмотреть на игрушку, крошущуюся за значком «Lines & Colors» – не знаю, мне почему-то не удалось найти её по названию.

Видео оставило приятное впечатление, но впечатлила скорее находка – обжитый фирменный российский компьютер, с которого предыдущие владельцы, к счастью, не удалили кучу интересного софта, а только её преумножили.

Андрей Шаронов (Andrei88)
Александр Чуклинов





РЕЦЕНЗИЯ НА КНИГУ: ТРИСТАН ДОНОВАН. «ИГРАЙ! ИСТОРИЯ ВИДЕОИГР»



Как и следует из названия, книга рассказывает о истории видеоигр, но автор сразу же оговаривается, что под «видеоиграми» он понимает не только игры для консолей, но и компьютерные игры. В отличие от аналогичных исследований, созданных в США и описывающих местную индустрию, автор, по его уверениям, попытался создать более общую картину, рассматривая течения, которые возникли в Японии, Европе и «прочих местах» (отдельная глава отведена Восточному блоку, Алексею Пажитнову и истории «тетриса»). Также, в Предисловии заявлена идея описать историю видеоигр не через

эволюцию «железа», а через эволюцию программ. Временной диапазон книги – от 1945 года (ядерная бомбардировка Японии) до 2010, нарастание популярности и влияния инди-игр¹.

Собственно история начинается с рассказа о том, как в начале пятидесятих годов двадцатого века вычислительные машины впервые были применены для простых в программировании игр (ним, крестики-нолики), что было лишь побочным продуктом, «пробой сил» в процессе серьезных научных работ. Практически одновременно были созданы первые самообучающиеся программы для игр в шашки и шахматы, и уже к шестидесятому году шашечная программа смогла победить американских чемпионов. Тогда же появились и предки того, что теперь называется игровой приставкой: контроллер, управление и телевизор в качестве монитора, однако расцвет идеи состоялся только в шестидесятих...

Дальше повествование переходит к появлению первых игровых автоматов, первым серийным игровым приставкам и расцвету индустрии развлечений. Читается довольно легко, в процессе подготовки книги было взято сто сорок интервью, фрагменты которых использованы в соответствующих главах. Иллюстраций, к сожалению, довольно мало, и книгу это не красит, поскольку по описанию в тексте не всегда возможно представить себе, как именно выглядела та или иная игра, тот или иной компьютер. В книге дан относительно хороший обзор развития ситуации, но за деталями нужно обращаться к первоисточникам.

К сожалению, у меня нет возможности ознакомиться с оригинальным текстом, что же касается русского перевода, то он оставляет

¹ Русское издание вышло в 2014 году: Москва, издательство «Белое Яблоко», ISBN 978-5-9903760-4-5. Дизайн обложки: Джей Прист и Том Хоумвуд, фотография на обложке: Блейк Селл © Bettmann/CORBIS.



желать лучшего. Открываю книгу наугад, глава 2 «Не упускайте мяч и ставьте рекорды»: все ли русскоязычные читатели смогут сходу перевести 120 градусов по Фаренгейту в градусы Цельсия? Поскольку человек, исследующий сеть подземных туннелей и коммуникаций, не называет себя диггером, то он так и остаётся с многословным определением, хотя понятие в русском языке давно закрепилось. Загадкой от лёгкой промышленности является «куртка с выпадающей подкладкой»² (или подкладка просто отстёгивалась?). В интервью не всегда правильно понято и, соответственно, передано интонирование, из-за чего смазанной оказывается мотивация, и так далее. Но если читатель не склонен анализировать текст и не знает контекста, погружаться в который переводчик не счёл нужным, то никаких «странностей» он и не заметит.

Что касается содержания – набор фактов сам по себе редко бывает увлекательным, чтобы их последовательность такой стала, нужны не только фрагменты интервью, но и оригинальная субъективная позиция автора. Именно автор интерпретирует факты и вдыхает жизнь в повествование. Здесь же мы имеем дело с работой иностранца, носителя западного менталитета, искренне уверенного в том, что видеоигры зародились из холодной войны и последовавшего противостояния – таким образом, изначально отмечается идея о том, что компьютер послужил новой средой реализации творческого потенциала. Сама по себе идея рассмотреть историю видеоигр через эволюцию программ очень интересна, но мы сразу приходим к необходимости определить, что конкретно будет рассматриваться в качестве эволюционирующего элемента. Автор, к сожалению, пошёл по достаточно шаблонному пути, и эволюция – в его версии событий – в основном сведена им к масштабам захвата рынков сбыта и развитию технической составляющей

(габариты, графика, звук и т.п.), но не собственно к содержательной стороне игры. Тема была заявлена, но рассмотрена крайне поверхностно. В том же Предисловии автор уверяет, что «хотел написать о видеоиграх как о виде искусства, а не как о коммерческом продукте», но опять же, нигде далее по тексту эта заявка ничем не подтверждена. В конце книги приводится раздел «Игрография», который сам по себе мог бы быть основой для действительно интересного исследования, но увы, увы... Также, хотелось бы, чтобы повествование было построено с учетом восприятия геймера, того человека, ради и для которого всё это и делается, но зато много места уделено «влиятельным людям из игровой индустрии».

Особенно сильно разница в менталитетах чувствуется в главе 16 «Самолет до Москвы», в которой затронута история Алексея Пажитнова: сказка «Конёк-горбунко» полагается сатирой на социальное устройство царской России, якобы из-за чего она и была так популярна в СССР³, и автором совершенно органично не принимается во внимание идея о том, что государственная идеология в СССР существовала в качестве некой второй реальности, а первой реальностью у людей были их собственные интересы, для них гораздо более актуальные. И сейчас очень интересно читать вот такие наблюдения: «...советские игровые автоматы были довольно дорогими — игровой сеанс стоил 15 копеек. На эту сумму можно было перекусить⁴. Коммунистическая идеология оказывала влияние на любой аспект жизни за железным занавесом, и советские игровые автоматы не являлись исключением. В руководствах к этим автоматам прямо говорилось, что они являются не просто развлечением, а инструментом для «активного досуга и развития глазомера». В отличие от американских, японских и западноевропейских игр в советских играх не было набора очков, — в основном потому, что такой подход

² Здесь и далее в рецензии в кавычках приведены фрагменты текста в русском переводе И.Воронина.

³ В царской России (до 1917 г.) сказка была переиздана 26 раз, первое издание — 1834 г. С 1843 — после выхода третьего издания — сказка была запрещена цензурой и 13 лет не переиздавалась. В 1856 и 1861 гг. выходят новые издания, без цензурных пропусков и с переработанным текстом. Это классика детской литературы, записанная автором практически так, как он услышал её от рассказчиков.

⁴ В советской столовой этого хватало на тарелку макарон с подливой, стакан чая и хлеб. (Проезд в городском автобусе стоил 5 копеек, билет в кино: 10 коп. на утренний детский сеанс (в 10 утра), от 20 до 30 коп. дневные и вечерние сеансы). Но и на Западе сеанс игры на первых игровых автоматах стоил как литр бензина, считалось, что это дорого. Были в СССР и автоматы для игры в кегли (боулинг), с шарами, дорожкой для броска и кеглями, электронные викторины для двух игроков на знание дорожных знаков и пр.



рассматривался как форма поощрения конкуренции, а не сотрудничества между игроками. Запрещены были любые игры, в которых нужно было стрелять в людей, а не в животных или по мишеням.

Когда перестройка набрала обороты, разработчики аркадных игр восточного блока стали яростно и быстро сокращать отставание»⁵.

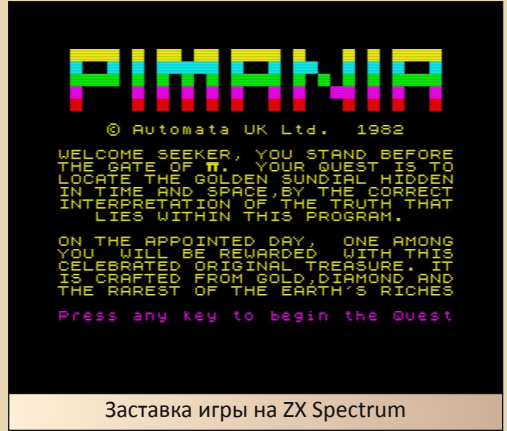
Очень жаль, что в этой же главе автор опускается и до ничем не подтверждаемого заявления о том, что «десятилетия коммунистического правления довели экономику стран Восточной Европы до ручки», якобы из-за чего они так тяжело входили в рыночную экономику на протяжении девяностых годов. Отмечая уникальность и самобытность игр, созданных в Восточной Европе, автор, тем не менее, выводит их уникальность и самобытность из опыта борьбы против коммунизма. И это на фоне того, что вполне доступна информация о состоянии промышленности, культуры и территориальной целостности стран Восточного блока в период СССР и сейчас, когда победа над коммунизмом уже давно позади.

Забавный факт: когда «Тетрис» впервые появился на компьютерах в офисе британской компании Mirrorsoft, работа над проектами на некоторое время оказалась парализованной, настолько сотрудники были увлечены новой игрой. Тем не менее, игре было отказано в потенциале, и решение о покупке на неё лицензии и выпуске игровых копий было принято только после получения восторженных отзывов от их американского подразделения, Spectrum Holobyte.

В отдельной главе – глава 9 «Дядюшка Клайв» – рассказана история любимого многими компьютера ZX Spectrum, намного короче, чем в известной книге Родни Дейла, и намного поверхностней информации сайта «Неизвестный Синклер» (zxnext.narod.ru).

Англия. Остров, одаривший мир великолепной музыкой «Битлз», но также мини-юбками, «Летающим цирком Монти Пайтона» и компьютером ZX Spectrum. Свойственные

островитянам эмоциональная зажатость и волевая направленность не подавляют полнотью живую человеческую природу, этот конфликт и оказывается хорошей почвой для «британского сюрреализма» и невероятных проявлений, оказавших влияние на весь остальной мир.



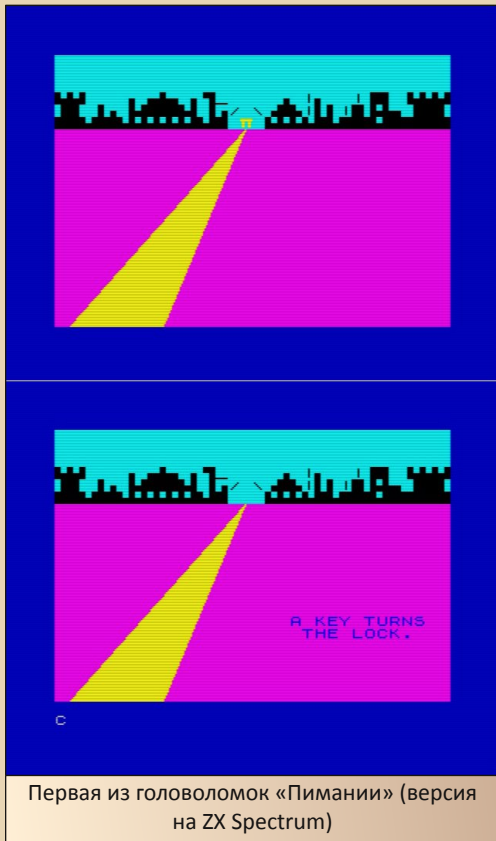
Заставка игры на ZX Spectrum

В 1982 году компания Automata, созданная Мелом Краучером, отцом британской индустрии видеоигр, сразу на четырех платформах (BBC Micro, Dragon 32, ZX Spectrum и Sinclair ZX81) выпускает игру «Пимания» (Pimania), которая по сути своей была антиигрой: игроку нужно было решить ряд головоломок, и в результате определить место в физическом мире, куда нужно было прибыть за получением ценного приза (золотых солнечных часов стоимостью шесть тысяч фунтов). Прибыть на место вручения нужно было вполне определённым образом и в определённое время. «Пимания» была первой игрой, в которой ставилась подобная – выходящая за пространство игры – задача, и первой игрой, за прохождение которой назначался ценный приз. По словам самого разработчика, он «пытался размыть грань между фантазией и реальностью, переделать все эти тоскливые подходы к играм для того, чтобы люди смеялись во время игры, а вместо этого они стали всерьёз выполнять все эти дурацкие квесты»⁶.

⁵ Глава 16.

⁶ Историю прохождения «Пимании» крайне интересно проследить по публикациям в тематических журналах, но это приведёт к неоправданному увеличению объёма рецензии, поэтому остаётся для самостоятельного рассмотрения





Первая из головоломок «Пимании» (версия на ZX Spectrum)

...приз был получен только спустя три года после выхода игры. В СССР из четырех заявленных платформ массово распространился только ZX Spectrum, однако в советском и последующем русскоязычном сообществе, насколько мне известно, игра вообще не прозвучала и даже не упоминается ни в одной из серий «Описания 500 игр». Она построена на ломке привычных игровых шаблонов, игроку как раз таки не нужно погружаться в игровой мир, а сохранять отстранённость – и тогда крайне нестандартные задачи решаются хоть и не с лету, но решаются. Представьте: на экране нечто вроде стены и ворот, потом эти «ворота» улетают в небо и от игрока требуется что-то ввести с клавиатуры. Чаще всего нажимался «reset». Так и

получились три года. И даже сейчас, при всей простоте технической реализации, игра интересна для прохождения.

THE GOLDEN SUNDIAL OF PI!



Finally! The Pi-Man has come out of the closet to present the Golden Sundial of Pi to a couple of female Pimaniacs who managed to unravel the mystery of Pimania! They solved the world's longest running computer puzzle by being at Hindover Hill in Sussex on July 22nd. Sue Cooper and Lizi Newman of Ilkley endured the frightening sight of the Pi-Man in broad daylight to collect the sought-after Golden Sundial! More news from Keith Campbell next ish!

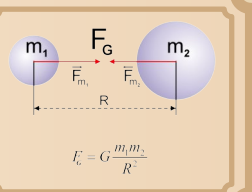
Заметка из журнала «Computer & Video Games» №9, 1985 год

Предыдущие три абзаца – это просто импровизация рецензента, вариант того, какой примерно могла бы быть рассказанная история, в книге ничего такого нет. Тем не менее, книгу стоит рекомендовать для прочтения и пополнения ею домашней библиотеки.

Андрей Тумилевич
(2:5020/570.10@fidonet)



DOWNGRADE-МАШИНА В ФИЗИЧЕСКОМ КАБИНЕТЕ



В своё время автор пару раз был достаточно активным подписчиком журнала «Радио». Хотя журнал с компьютерной тематикой напрямую не связан, статей, интересных фанату компьютеров, а уж тем более старых компьютеров, там было предостаточно. Особенно до 2005-го года (меня подписали родители в конце 2002-го года, и я получал журналы за 2003-й и 2004-й год). Потом, конечно, журнал начал отходить от таких приятных взору даунгрейдера тематик, как подключение советских и СЭВовских принтеров к PC-совместимым компьютерам, приставок с управляющими программами под DOS, программаторы, подключаемые к LPT- и COM-порту и т.д. Да, предполагалось, что радиолюбитель заинтересуется интерфейсом USB, а на компьютере у него уже установлена Windows XP, но статья, о которой пойдёт речь, появилась именно тогда – когда уже триумф USB, PCI и «экспишки», да даже и «семёрки» на радиолюбительских компьютерах, по мнению редакции, был безоговорочным.

Статья «Компьютер на уроке физики. Изучаем равноускоренное движение» В. Пискоржа (надеюсь, фамилия склоняется ☺) была опубликована в февральском номере журнала «Радио» за 2009-й год (стр. 45):

<http://radiolub.ru/uploads/jurnal/r-2009-02.djvu>

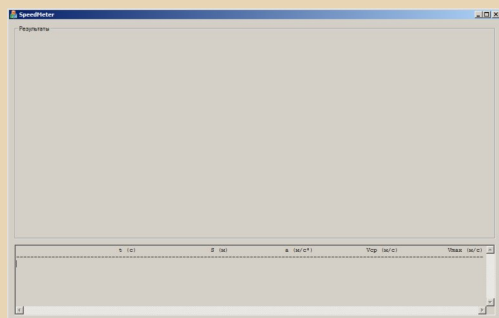
В статье описывается достаточно интересная электромеханическая установка для демонстрации опытов с равноускоренным движением тел. Как видно из рис. 1 статьи, установка представляет собой наклонную плоскость, по которой скатывается в нужный мо-

мент металлический шар. На наклонной плоскости установлено два регистратора прохождения шара – оптопары, подающие сигнал на электронный блок установки. Электронный же блок с помощью встроенного ждущего мультивибратора формирует импульсы, подаваемые на LPT-порт компьютера. Программа же компьютера измеряет время между импульсами, формируемыми при прохождении шаров первой и второй оптопары. Так как предполагается, что движению шара ничего не будет мешать, то программа по этим данным достаточно легко вычислит значение ускорения, да и, думается, скоростей при прохождении первой и второй оптопары тоже.

Программа, работающая на компьютере, написана, к сожалению, уже не под DOS:

<https://ftp.radio.ru/pub/2009/02/speedmeter.zip>

Ваш покорный слуга даже не гарантирует, запустится ли она на Windows 9x (сам проверил только в режиме совместимости с Windows 98 на семёрке – машина с Windows 98 сейчас находится очень далеко, ☹ но в режиме совместимости программа нормально запустилась)¹.



¹ К сожалению, программу под Windows 98 удалось запустить только с установленной KernelEx 4.5.2 – прим. ред.



Но даже если не запустится – сама идея применить Downgrade-машину в составе подобного наглядного пособия достаточно интересна. Ведь работать с портами и аппаратным обеспечением под DOS всё же проще, а заполняющая весь экран программа, выводящая на экран результаты измерений, не только закрывает отвлекающую кнопку «Пуск», но и, при красивом и стильном оформлении, сможет привлечь наблюдающих.

Другое дело, что такому стенду понадобится не только старый компьютер с написанной под DOS или Windows программой, но и электронное, а ещё куда главнее, механическое обрамление, но и эта задача посильная. Ведь даже редактор журнала в двадцатом номере в статье «Брейн-ринг в стиле Downgrade» написал, как он сумел сделать небольшую приставку к ноуту, позволившую применить старый компьютер в викторине.

Значит, в принципе, задача построения наглядных пособий вполне себе посильная. Сложная, но посильная. Так что, может быть, старый компьютер сослужит хорошую пользу не только для обучения программированию, обучения с помощью игр, интерфейсам периферийных устройств, но и там, где обходились без него? Да, обходились, но с ним может оказаться ещё интересней, зрелищней и наглядней.

Как знать, возможно, наблюдающий за старым компьютером школяр будет думать не «Отстой!», а то, что сей компьютер чем-то напоминает агрегаты из старых американских боевиков, или же заинтересуется, как же так сумели сопрячь комп и такую нехитрую установку? А, может быть, уже и хитрую – ведь не только скорость и ускорение шарика может измерить компьютер с подходящим обвесом. Возможно, что на каком-то стенде будет измеряться уже напряжение или давление, и задачу

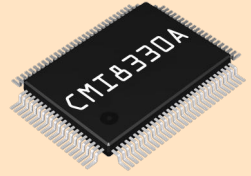
передачи данных в компьютер будет решать маленькая коробочка или плата.

Что в итоге? Возможно, мир не получит ещё одного даунгрейдера, но хотя бы получит ещё одного электронщика, что уже неплохо.

Андрей Шаронов (Andrei88)



СМ18330 – ЕЩЁ ОДНА ISA-ЗВУКОВУХА В СТРОЮ



Во время разгребания завалов старого железа и повального списания автор умудрился утянуть себе ещё одну звуковую карточку – СМ18330. Низкопрофильная карта, у которой GAME-порт был прикручен к крепёжной планке, а с платой соединялся шлейфом, показала себя бюджетной и близкой к ESS-кам (см. рис. 1). Поэтому ли прихватил её автор или же просто – до кучи – по принципу «такой у меня ещё не было», но карточка была отложена и пролежала несколько лет в ящике стола.



Рис. 1. Героиня обзора – низкий профиль, соединение игрового порта с платой шлейфом – чем-то напоминает и ESS186х, и низкопрофильные мультикарточки, где так же крепится COM-порт

Почему не побежал сразу подключать? Одним из препятствий стала болтающаяся крепёжная планка. Из двух креплений присутствовало только одно. Автору показалось тогда, что крепёжную планку на текстолите держат заклёпки. Поиск пары болт-гайка не очень ра-

довал и постоянно откладывался на потом. Но тут зимой была собрана ещё одна «четвёрка», и встал вопрос о звуке для неё. Зимой этот вопрос не решился, хотя на сайте «Авито» в Орске (где и находится новособранный компьютер) и было вывешено объявление о продаже такой же платы, но купить тогда так и не собрался.

Зато летом для экспериментов были привезены целых три карточки – две ESS-ки, о которых было рассказано в другой статье, и СМ18330. ESS-ки были потихоньку подключены, проверенны и описаны в статье :-). Эксперименты со звуком были ненадолго отложены. Казалось, что со звуковой уже определился, отдав предпочтение настраиваемой перемычками ESS688, решив потихоньку потом попытаться найти и CD-привод с проприетарным интерфейсом, но, походив немного, решил запустить и эту карточку. Покрутил её в руках, пригляделся к креплению. Разве заклёпка? Пригляделся – обычный болтик – пусть и не сильно длинный. Пригляделся и к отверстию для второго крепежа – тоже какое-то подобие резьбы. Может быть, тогда получится найти в хозяйстве ещё один такой болт и привести плату в более-менее нормальное состояние?

Первый болтик оказался немного больше-го диаметра – как его ни вкручивал – далеко он ввинтиться не мог. Ну, уж лучше так – хоть крепёжная планка не шатается. Но, может быть, посмотреть ещё? Абстрагироваться, так сказать, от похожей шляпки? Абстрагировались – и от длины, и от похожести, и что видим? Есть более тонкий болтик – шляпка другая, длиннее, но спокойно вкрутился до конца. Да, торчит с другой стороны немного, но держит крепёжную планку крепко, чего и добивались.



А раз так всё удачно получилось – переходим к установке платы в компьютер.

Установка драйвера

Перед поиском и скачиванием драйверов желательно посмотреть маркировку чипсета на плате или же название. Автор этого делать не стал – на old-dos.ru только один комплект драйверов – для CMI8738 – конечно же это наш случай! Странно, конечно, что драйвера для Windows 3.x нет, но ничего – настроим! Вот только почему такие проблемы с совместимостью в играх на эмуляции Sound Blaster? Баааа, так это же плата для PCI-шины! А у нас тогда что впяно? Автор взял снова в руки звуковуху и внимательно оглядел – да, не CMI8738 – CMI8330!

Что ж, давайте искать драйвера для неё. Нужный комплект драйверов нашёлся на сайте Vogons ([ссылка](#)) – включал в себя драйвера для Windows 3.x и DOS – одним комплектом, Win9x, WinNT 4.0 и комплект программ для Win9x.

Чудесно, качаем архив, перекидываем на старый компьютер и начинаем установку. Для других бюджетных карточек, например, ESS или Crystal CX4235-XQ3, инсталлятор запускается под Windows, хоть и при этом ставит DOS-утилиту для предварительной инициализации платы и её настройки. Здесь же установка идёт из-под DOS – см. рис. 2.

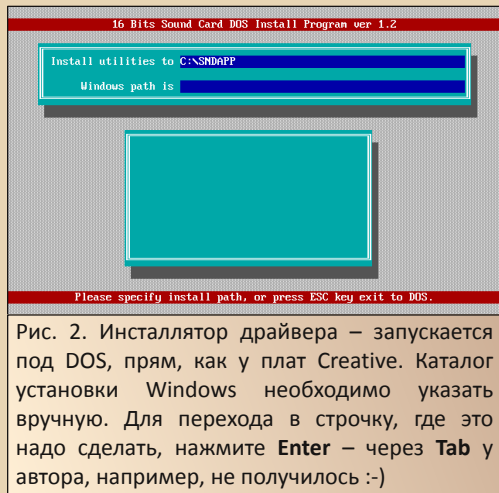


Рис. 2. Инсталлятор драйвера – запускается под DOS, прям, как у плат Creative. Каталог установки Windows необходимо указать вручную. Для перехода в строчку, где это надо сделать, нажмите **Enter** – через **Tab** у автора, например, не получилось :-)

Переходим в строчку, где необходимо прописать путь к каталогу Windows, прописываем, запускаем установку. После этого инсталлятор предложит проверить наличие платы. Автор случайно нажал **Esc**, чем отменил проверку. Инсталлятор самостоятельно запустил Windows, но не смог запустить программу **wsetup.exe** – винда не нашла нужный файл, о чём и известила. Однако сей файл находится в каталоге драйвера, и его можно запустить самостоятельно, что автор и сделал.

Инсталлятор поставил софт для винды, обслуживающий карточку, и предложил перезагрузиться.

Вот тут-то началось – сперва винда выдала синий экран с сообщением об ошибке одного из файлов, явно связанных с драйвером. Однако по нажатию **Enter**'а загрузка продолжилась. Но винда отрапортовала, что не нашла звуковую карту (см. рис. 3).

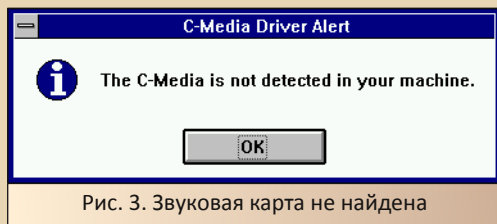


Рис. 3. Звуковая карта не найдена

Ладно, может быть, что-то не так в **autoexec.bat**? Открываем файл и видим:

```
C:\WINDOWS\SMARTDRV .EXE
@echo off
PATH=C:\WINDOWS;%PATH%
PATH C:\NU;%PATH%
SET SYMANTEC=C:\SYMANTEC
SET NU=C:\NU
SET TEMP=C:\WINDOWS\TEMP:C:\SNDAPP\CMINIT
/WA:530 /WI:11 /WM:0 /VA:220 /VI:5 /VL:1
/VH:5 /MD /MA:330 /MI:9
SET BLASTER=A220 I5 D1 H5 T4
C:\SNDAPP\CMMIX /MCC000 /FEE000 /WEE000
/LEE00E /EEE00E /AEE00E /D00100 /C70170
/300000 /O00000 /S00000
```



Инсталлятор затолкал запуск инициализационных программ в самый конец файла. Может быть, из-за этого проблемы? Правим файл – начальную настройку карты в самое начало файла:

```
C:\SNDAPP\CMINIT /WA:530 /WI:11 /WM:0
/VA:220 /VI:5 /VL:1 /VH:5 /MD /MA:330
/MI:9
SET BLASTER=A220 I5 D1 H5 T4
C:\SNDAPP\CMMIX /MCC000 /FEE000 /WEE000
/LEE0EE /EEE0EE /AEE0EE /D00100 /C70170
/300000 /000000 /S00000
C:\WINDOWS\SMARTDRV.EXE
@echo off
PATH=C:\WINDOWS;%PATH%
PATH C:\NU;%PATH%
SET SYMANTEC=C:\SYMANTEC
SET NU=C:\NU
SET TEMP=C:\WINDOWS\TEMP
```

Перезапускаем – тот же эффект – ошибка драйвера, плата не найдена. Хорошо, запускаем **CMTEST** (см. рис. 4) – как раз та самая программа, которая должна была найти и протестировать плату.

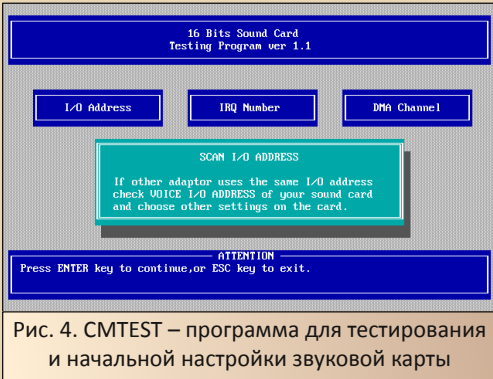


Рис. 4. CMTEST – программа для тестирования и начальной настройки звуковой карты

Запускаем тест и... плата не находится. Вот это здрасте... Ну, для найденной чёрт знает где звуковухи, в принципе, возможный исход, но очень автору не понравилось то, что карточка в

слоте еле держится. Компьютер был выключен, плата извлечена, крепёжная планка подогнута... В этот раз плата встала более уверенно. Запускаем компьютер и... никаких сообщений об ошибках. Подключаем наушники – звук есть, музыка тоже (да-да, игры Nikit'ы :-). Можно пользоваться. :-)

Эквалайзер и 3D-эффект

Когда автор вертел в руках плату, изучая надписи и переключки, в глаза бросилась надпись «**3D surround**». Объёмный звук? Интересно, конечно, но, скорее всего, эта функция реализована только для Windows 9x, так что прочитанное сразу было забыто.

Вспомнилась надпись только после окончания оживления платы – в группе программ, обслуживающих звуковую карту, оказался яркий **Surround sound demo**. То есть, как минимум, фирма предоставила программу, демонстрирующую объёмный звук? Запускаем (см. рис. 5).

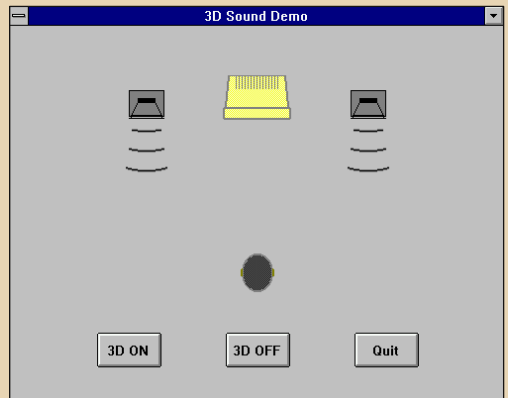


Рис. 5. Программа для демонстрации эффекта трёхмерного звучания (трёхмерный звук отключен)

Начинает воспроизводиться музыка. По умолчанию трёхмерный звук отключен. Для включения трёхмерного звучания нужно нажать кнопку **3D ON** (см. рис. 6).



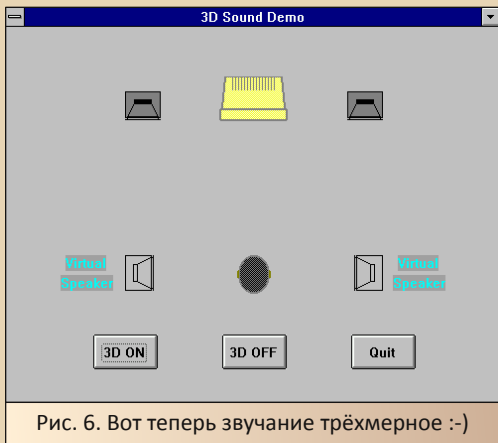


Рис. 6. Вот теперь звучание трёхмерное :-)

Удастся ли услышать трёхмерность? Не знаю. Автору показалось, что звук стал громче и сместился в левую сторону (слушалось в наушниках). После же отключения трёхмерности, звук сдвигался вправо и становился глуше, и создавалось впечатление, что звучит из головы. Возможно, на колонках будет какой-то другой эффект.

Но демонстрация демонстрацией. А можно ли включить эффект в играх? Оказалось, что и это возможно – открываем микшер – ярлык **Mixer** в группе программ, обслуживающих звуковую карту (см. рис. 7), и можем настраивать как уровни звука, так и включать и выключать эффект трёхмерности.

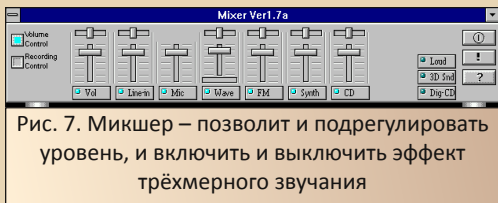


Рис. 7. Микшер – позволит и подрегулировать уровень, и включить и выключить эффект трёхмерного звучания

Опять же – вопрос трёхмерности достаточно спорный. Конечно, лучше такие вещи проверять на трёхмерных шутерах – там-то сразу должно быть понятно – есть эффект присутствия или нету. Но даже в никитовских играх

заметно, что при нажатии на кнопки **Loud** и **3D Snd** звук меняется.

А можно ли включить трёхмерное звучание изначально? При начальном запуске DOS было видно, что появляется табличка, где выводятся текущие настройки громкости, а также показано, что трёхмерное звучание выключено. Значит, можно попробовать включить трёхмерное звучание в ДОСовском микшере.

Настройку микшера можно запустить командой:

```
C:\SNDAPP\CMMIX
```

Почему-то у автора не получилось запустить программу из файлового менеджера (использовался описанный в 26-м номере журнала IBM HandShaker), так что, если у вас тоже не получится – попробуйте из командной строки.

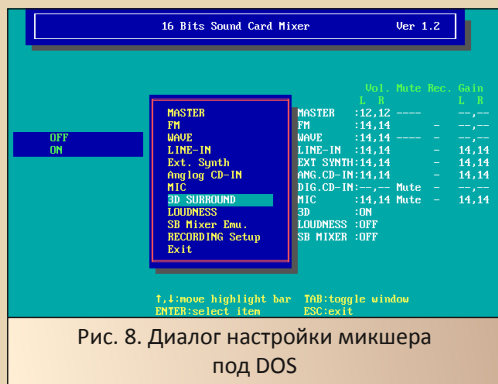


Рис. 8. Диалог настройки микшера под DOS

В открывшемся диалоге (см. рис. 8) можно настроить как громкость звучания, так и включить или выключить звуковые эффекты, в том числе и трёхмерный звук.

После выключения диалога настройки звука, изменения будут отражены в табличке (см. рис. 9) – такой же, как появляется при начальной загрузке DOS.



```

      Mute:  Mute:  Mute:  Mute:
MASTER  :06.06  ---  ---  ---  ---
FM       :14.14  ---  ---  ---  ---
MIDI     :14.14  ---  ---  ---  ---
LINE-IN  :14.14  ---  14.14  ---  ---
EXT SYNTH:14.14  ---  14.14  ---  ---
SPEAKER :14.14  ---  14.14  ---  ---
DIG. CD-IN: ---  Mute  ---  ---  ---
MIC      :14.14  Mute  ---  14.14  ---
3D       :ON
LDRPRESS :OFF
SB MIXER :OFF
C:\SNDAPP>
    
```

Рис. 9. Таблица текущих настроек микшера

Так что, предварительные настройки громкости и включение эффекта трёхмерного звука можно произвести и до загрузки Windows, что может оказаться полезным при использовании звуковой карты с ДОСовскими играми (всё же для DOS трёхмерных шутеров сделано куда больше :-).

Заключение

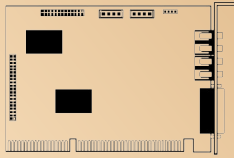
Что можно сказать напоследок? Описанная в статье звуковая карта, пожалуй, единственная из трёх, которая бралась с прицелом на статью. Предполагалось, что для неё подойдут драйвера с old-dos.ru. Почему-то тогда прочитал, что данные драйвера эмулируют не просто Sound Blaster, а Sound Blaster Pro, драйвера для которого шли в дополнительном драйвер-пакете для Windows 3.x. Предполагалось применить именно такую связку.

Увы, судьба распорядилась по-другому. Для звуковой карточки нашлись родные драйвера, а проблемы с запуском – в этот раз – не были связаны с какими-то программными изысками, а были прямым следствием невнимательности автора и его неаккуратности. Ну как ещё можно иначе охарактеризовать ситуацию, когда синий экран «лечился» подгибанием крепёжной планки, хотя, такое «лечение», думается, не редкость, как и починка завинчиванием болтиков. Получалось, что и писать-то не о чем? Задуманный хитрый программный

финт по аналогии с ESS1869 из статьи в двадцатом номере журнала не понадобился, карта запустилась практически из коробки – никаких нетривиальных действий не понадобилось. Фактически, всё спас трёхмерный эффект – пожалуй, не самая тривиальная фишка для бюджетной карточки? Каким бы он ни был, но карта действительно обладает функцией, изменяющей звучание. Трудно сказать, сделает ли она «скучный плоский звук» из китайских колонок «более объёмным, певучим, интересным, наполненным нюансами», но всё равно повозиться с карточкой было очень интересно.

Андрей Шаронов (Andreii88)





ESS688 – СНОВА БЮДЖЕТНЫЙ ЗВУК НА 486-М КОМПЬЮТЕРЕ

3 звуковой карте на чипсете от ESS автор уже посвящал одну статью (см. «ESS1869 под Windows 3.11 или бюджетный звук на ретрокомпьютере» в 20-м номере Downgrade-журнала). Карточка та или её собратья были установлены на компьютер товарища, родственников и, думается, много где ещё. Для автора в далёком 2005-2006-м году стать обладателем такой карточки было более чем реально, но судьба распорядилась иначе – тогда на 386-ой машине ваш покорный слуга плюхался с AWE32 (или AWE64, как её определяла ASTRA, но плата была с разъёмами под SIMM 30 pin ☺) – как показала дальнейшая практика – проблем было меньше. ESS1869 была же установлена на старый компьютер и настроена. Но снова автору всё не так – плата, хоть и ISA-шная, но относительно новая – всё же устанавливалась уже на 433-е «Селероны» (ну, у товарища, возможно, на 200-м пеньке такая же стояла, но сейчас и не сыскать следов того компьютера) – в общем, для «четвёрки» несколько не то. Вот здесь описывается достаточно бюджетный 486-й компьютер с ESS 488:

<http://oldcomputer.info/pc/skladaki/green486/>

Увы, такой красавицы на 8-разрядный слот у меня нет, но зато удалось, в своё время, унести две платы на ESS 688. Когда собирал компьютер на UMC U55, одну из звуковых карт не получилось запустить. Но, может быть, сейчас получится? Вторую же не пробовал вообще.

Для экспериментов был использован 486SX на 25 МГц на материнской плате с VLB-шиной и 8 МБ ОЗУ. На компьютере установлена MS-DOS 6.22 и русская версия Windows 3.11.

Плата первая – Plug'n'Play-ная

Когда-то автор во время списания кучи старого железа в институте заполучил парочку звуковых карт. Звука тогда было достаточно своего, но всё же две платы унёс. Одной из них оказалась ESS ES968F (по крайней мере, так её называли [здесь](#)). Вариант автора можно увидеть на рис. 1.



Рис. 1. ESS ES968F, имеющаяся у автора

Гугль находил ещё Miss Melody MF-688SP, похожую расположением БИС, но несколько отличающуюся расположением перемычек: [ссылка](#).

Оказалось же, гребёнка в центре платы (где установлены две перемычки) настраивает какие-то эффекты – скорее всего, работу с дочерней платой. А раз так, то можно попробовать оставить, как есть – на найденных фотографиях в интернете перемычки установлены также. Увы, найти документацию на карточку не получилось. На самой плате были подписаны только JP1 – режим P'n'P – если перемычка установлена, данный режим отключён, если отсутствует – то карта работает в режиме Plug'n'Play. В случае автора, на плате даже не был распаян разъём под перемычку.



Второй подписанный джампер – JP2 – включает или отключает наборный IDE-контроллер – переключатель установлена – IDE отключён, отсутствует – IDE включён. В нашем случае переключатель стоит.

В таком виде плата была установлена в системный блок. Для запуска была выкачана утилита автоматического определения платы **Ess pnp sound card Setup** с сайта пятидюймовки: <http://www.fdd5-25.net/sounddrv.htm> (тот же архив можно взять с Old-DOS: <http://old-dos.ru/dl.php?id=17131>) и драйвер-пак для звуковых карточек ESS – Драйвер **ess xx88** со странички <http://www.fdd5-25.net/sounddrv.htm> или с Old-DOS: <http://old-dos.ru/dl.php?id=17130>.

Первым делом была запущена утилита для обнаружения платы – плата обнаружилась – в прошлый раз были какие-то проблемы – или не находилась, или же не выводился звук – сейчас не вспомню. После этого автор запустил установку драйверов. Думалось, что, как и в случае ESS 1869, будет DOS-утилита, но инсталлятор требовал запуска именно под Windows. Хорошо, запускаем инсталлятор в винде. Нас просят ввести параметры платы: адрес, прерывание, канал DMA. По умолчанию инсталлятор предложил следующие параметры:

- Адрес: 220h;
- IRQ7;
- DMA1.

Автор поменял прерывание на пятое, и ему предельно перезапустил Windows. Как потом выяснилось, утилита обнаружения и начальной настройки карты была, но все изменения **autoexec.bat** сделал винدوزный инсталлятор. Кстати, так поступает и инсталлятор драйверов Crystal CX4235, которая была описана в статье «Crystal – дешевле некуда» в прошлом номере журнала. Там тоже есть досовская утилита для обнаружения платы, но её настраивает сам драйвер.

Увы, тогда проверить, выводится ли звук или нет, не было возможности – как назло, куда-то запропастились наушники. В поисках наушников прошло несколько дней – в конечном итоге были куплены новые.

Ну, ключ на старт – наушники воткнуты в соответствующий разъём, компьютер запущен, винда стартует – никаких сообщений об ошибках звуковой карты... но и звука нет. Что не так? Открываем панель управления, а тут беда – настройки звуковой карты неактивны (см. рис. 2), настройки системных звуков – не лучше (см. рис. 3). Что же не так?

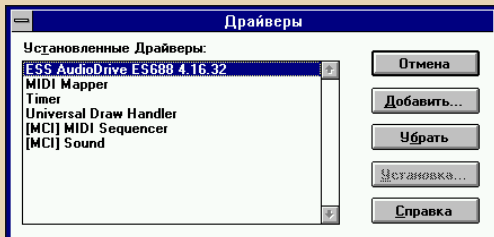


Рис. 2. Пропали настройки драйвера звуковой карты

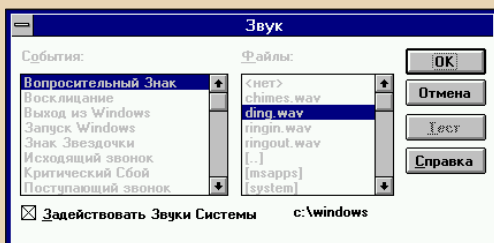


Рис. 3. И звуки системные неактивны

Выяснилось, что «не так» в утилите обнаружения платы. Её-то не запустили. Как только утилита обнаружения платы была запущена, в наушниках стала слышна музыка запуска Windows. Посему автор прописал её запуск в **autoexec.bat**.

Теперь стала активна и кнопка настройки параметров звуковой карты.

Но радость была недолгой – запуск одной из игр Nikit'ы показал, что музыки-то нет – MIDI не воспроизводится.

Возможно, виновата хитрая программа предварительной инициализации? Хорошо, убираем её, удаляем драйвер (это можно сделать с помощью всё того же инсталлятора – при запуске с уже установленным драйвером, но **autoexec.bat** чистить придётся самостоятельно), запускаем инсталляцию снова. Но в



таким случае установщик драйвера вообще не видит плату (см. рис. 4).



Рис. 4. Утилита автоматической настройки PnP-карты не запущена – установщик драйвера вообще ничего не нашёл

Что делать? Поступать, как в случае ESS1869 – драйвера для DOS, а там стандартный виндозный? Можно было бы, если б автор озаботился этим, когда подключал жёсткий диск к новому компьютеру – увы, сетка к старой машине не проведена. Но, может быть, получится что-то сделать имеющимися средствами? Может быть, присмотреться к хитрой утилите для обнаружения Plug'n'Play-ной карточки? Запускаем утилиту с параметром /? и дивимся – утилита-то позволяет указать и адрес, и прерывание, и канал DMA. Но ещё и настроить режим работы карточки – или ESS1688, или ESS688 – по умолчанию стоит первый вариант. Может быть, тут загвоздка (да, по умолчанию ставятся те же самые параметры, что и у драйвера – адрес 220h, седьмое прерывание и 1-й канал DMA)? Автор выставил с помощью параметров режим ESS688 и пятое прерывание. Строчка для запуска стала выглядеть так:

```
C:\init_ess.com /N /I:5
```

Что ж, теперь ставим снова драйвер. И да – музыка зазвучала. А ещё поставился отдельный драйвер MIDI-устройства (см. рис. 5).

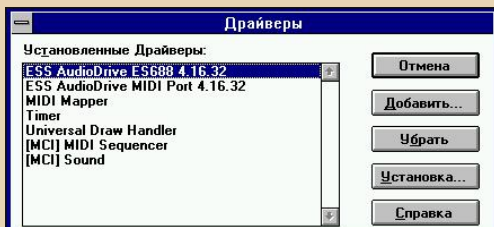


Рис. 5. После «перенастройки» установился ещё один драйвер – MIDI-устройства

Вот теперь можно считать, что карточка установлена полностью.

Плата вторая – перемычная

С появления первого старого компьютера автор фантазировал – какая она будет, ISA-шная звуковая карточка? Наверное, простенькая понапее, произведённая если не в известной стране, то в соседнем с ней Тайване или же Малайзии, которая сможет работать с драйверами от Sound Blaster в составе винды, о чём писал Роман Карпач в «Записках пользователя Windows 3.11» (сейчас под названием «Windows 3.11 для пользователя») эту статью можно найти здесь: <http://fdd5-25.net/publications/?p=108>). Оказалось несколько не так – первая карточка, как уже писал выше, была AWE32. Но мысли о карточке, настраиваемой перемычками, засели в голове. Какой будет перемычная карта? На этот раз казалось, что попадётся SB16 или SB32, либо вариант AWE32 с джамперами. В принципе, такая карта где-то лежит в закромах, но в этот раз речь пойдёт о плате на основе чипсета ESS688 – Miss Melody MF-1115 (см. рис. 6). Сразу скажу, что документацию я на неё не нашёл и настройкой работы набортного контроллера CD-ROM'ов не занимался. Надеюсь, что появится такой привод, и можно будет попытаться счастья с его настройкой и запуском на старой машине.



Рис. 6. Miss Melody MF-1115 – на этой карточке уже всё настраивается с помощью перемычек

На этот раз мы настраиваем только звук.

В отличие от перемычек, отвечающих за интерфейсы CD-ROM, назначение перемычек звуковой подписано на самой плате. Ниже в таблицах



продублированы настройки основных параметров звуковой карты.

Таблица 1. Настройка адреса платы

№	Адрес	JP4	JP13
1	220h	2-3	2-3
2	230h	2-3	1-2
3	240h	1-2	2-3
4	250h	1-2	1-2

Таблица 2. Настройка прерываний (JP2)

№	Прерывание	Замкнутые перемычки
1	IRQ5	3-4
2	IRQ7	1-2
3	IRQ9	Перемычки не устанавливаются
4	IRQ10	1-2 3-4

Таблица 3. Настройка канала DMA (JP12)

№	Канал DMA	Замкнутые перемычки
1	DMA не используется	Перемычки не устанавливаются
2	DMA0	3-4
3	DMA1	1-2
4	DMA3	1-2 3-4

В случае автора перемычки отсутствовали вообще. Руководствуясь вышеизложенным, плата была настроена по аналогии с предыдущей со следующими параметрами:

- Адрес 220h;
- IRQ5;
- DMA1.

Перед установкой платы в компьютер, драйвер был удалён, а **autoexec.bat** полностью вычищен.

Драйвер для Windows, использованный для предыдущей платы, нормально всё опознал и вывел также соответствующий диалог:

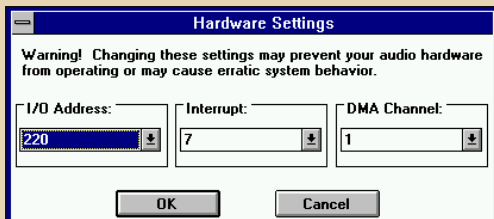


Рис. 7. Диалог настройки параметров звуковой карты при установке драйвера

Автор скорректировал прерывание – изменил с седьмого на пятое, после чего была перезагружена Windows, а потом и сам компьютер – решено установить регулятор громкости сзади карты на нейтральное положение, и лучше это сделать с выключенным компьютером ☺ (простите за фобию ☺). После этого плата нормально воспроизводила как звуки, так и MIDI-музыку в играх.

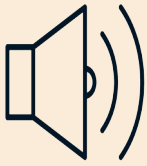
Конечно, запуск звука на плате, обладающей такими возможностями, – половина дела, но на этом пока что придётся остановиться. Возможно, со временем всё же получится приобрести CD-привод с проприетарным интерфейсом (ставить на звуковую карту IDE-сдиск как-то не хочется ☺) и попробовать подключить уже его.

Заключение

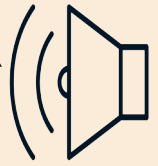
Большинство карт на чипсетах от ESS относятся к бюджетному сегменту (о как завернул ☺). Конечно, шипение и свист в наушниках при запуске не добавляют платам преимуществ, но что-то в них всё же есть. Хотя и приходится порой поплюхаться с настройкой и установкой драйверов – как видите, даже на ровном месте возникают плюхи, и не сразу их увидишь. Автор не сразу догадался, что после установки драйверов на P'n'P-карточку поддержка MIDI отсутствует. Но, действительно, в небольших бюджетных платах есть своя прелесть, и то, что поздние ISA-платы использовались на компьютерах с Windows 98, тоже придаёт им какое-то странное очарование. Вроде бы, шина ISA больше ассоциируется с DOS, но нет, есть платы для этой шины, которые работали под виндой и оставались в строю ещё в начале 2000-х. Что-то в этом всё же есть. А поставить более раннюю плату на 486-й компьютер оказалось тоже очень приятным делом.

Андрей Шаронов (Andrei88)

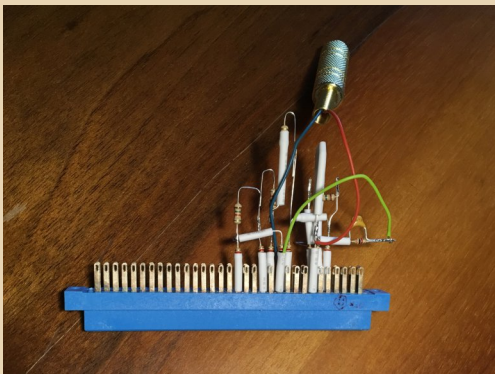




Цифровой звук на БК-0010



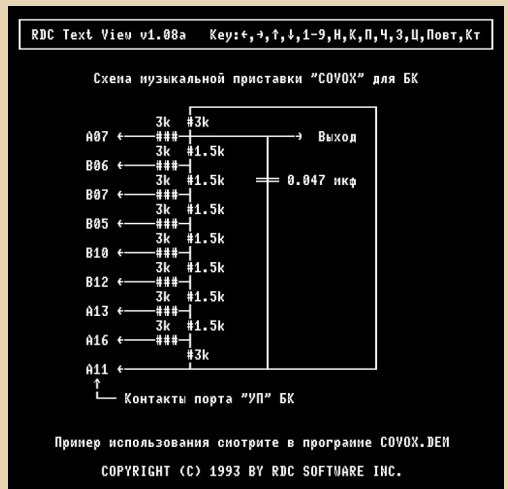
Всё, что происходит внутри компьютера, можно назвать цифровым – в том числе звук встроенного однобитного пьезодинамика, музыкального чипа AY, цифро-аналогового преобразователя и даже щелчки реле управления двигателем магнитофона. Но в этой статье «цифровым звуком» я буду называть только **оцифрованные сэмплы** – настоящие музыкальные инструменты, переведённые в цифровой вид. Речь пойдёт о популярном на старых платформах решении: 8-битном ЦАПе, собранном из горсти сопротивлений. Чаще всего его называют словом **«Covox»**. Припаянный к разъёму параллельного порта БК-0010, он выглядит так:



Свой первый Covox я спаял ещё в 1992-ом году, когда мне надоели поскрипывания встроенного динамика IBM PC. В то время я слушал цифровую музыку в проигрывателе STMOD и в демке Crystal Dream. Затем появился гораздо более ценный источник звука: музыкальный редактор **Scream Tracker 2**. Созданная в нём музыка называлась **трекерной**, потому что каждая нота записывалась в один из четырёх «треков»

(в данном случае трек – то же самое, что дорожка, канал, голос). Всё свободное время я просиживал за Scream Tracker 2, набивая по нотам песни Beatles и собственные произведения. Огорчало только одно: IBM PC стоял в учреждении, где я проходил школьную практику, а дома был лишь БК-0010.

В ноябре 1993-го года (здесь и далее хронология основана на сборниках программ московского Клуба БК) Денис Сотченко (RDC) опубликовал схему подключения Covox к БК и сопроводил её программой **COVOX.DEM**. Программа выводила простой текст и играла двухголосую мелодию парой инструментов, но зато это были живые инструменты!



Не помню, где впервые услышал COVOX.DEM – может быть, в Клубе БК на Свиблово, а может быть, дома у RDC. Конечно, я очень воодушевился и сразу набросился на автора с расспросами! Затем перепаял свой



Совох с PC на БК и начал эксперименты – благо, к тому времени у меня накопилась большая библиотека инструментов и был опыт создания цифровой музыки на PC. Через какое-то время я написал свою первую программу, воспроизводящую двухголосую мелодию.

MUSIC DEMO FOR COVOX
COPYRIGHT (C) RDC SOFTWARE

Опытом взлома чужих программ, как и инструментами для этого, я тогда не обладал: COVOX.DEM занимала всю память БК-0010, места под отладчик не оставалось, контроллера расширенной памяти у меня не было. Да и смотреть чужой алгоритм не особо тянуло – вроде, и так понятно. Зато теперь, благодаря [эмулятору](#) БК со встроенным отладчиком, я смог заглянуть в код и посмотреть, как он работает. И знаете, программа 26-летней давности удивила меня чрезвычайно! Вот код основного цикла:

```
MOV #177714,R5 ; Covox port address
MOV #5400,R4 ; Note duration
PLAY: ADD R0,#12345 ; R0: Frequency for sample 1
      ADC R2 ; R2: pointer to sample 1
      MOVB (R2),(R5) ; Put sample 1 data to Covox
      ADD R1,#12345 ; R1: Frequency for sample 2
      ADC R3 ; R3: pointer to sample 1
      MOVB (R3),(R5) ; Put sample 2 data to Covox
      SOB R4,PLAY ; Loop
```

Если вы не знакомы с ассемблером архитектуры DEC – не расстраивайтесь, я сейчас вкратце объясню, здесь всё просто. Помимо счётчика команд и указателя стека, есть шесть 16-битных регистров общего назначения, они нумеруются от **R0** до **R5**. Когда регистр указывают в скобках, имеют в виду, что он хранит не само число, а адрес числа (указатель на число), которое нужно использовать. Числа можно задавать и напрямую, перед ними ставят знак #. Метки обозначают латинскими буквами. Аргументы команд записывают слева направо: **источник, приёмник**.

Сами команды:

- **MOV** перемещает число из источника в приёмник.
- **ADD** прибавляет 16-битный источник к 16-битному приёмнику.
- **ADC** прибавляет «бит переполнения» к приёмнику (обычно этот бит равен 0, но он устанавливается в 1, если предыдущая команда выдала слишком большое число, то есть результат вышел за пределы 16 бит).
- **SOB** организует цикл (Subtract One and Branch) – вычитает единицу из источника и переходит на метку.

В процедуре проигрывания COVOX.DEM регистры использованы так:

- **R0** хранит некое число, соответствующее частоте **ноты** басового инструмента;
- **R1** хранит некое число, соответствующее частоте **ноты** сольного инструмента;
- **R2** указывает на хранящийся в памяти **сэмпл** басового инструмента;
- **R3** указывает на хранящийся в памяти **сэмпл** сольного инструмента;
- **R4** – счётчик цикла;
- **R5** – адрес порта, к которому подключён Covox.

Программа постоянно прибавляет **R0** к некому 16-битному числу-накопителю, это ни на что не влияет, пока вдруг не произойдёт переполнение. Тогда команда **ADC R2** увеличивает на 1 указатель басового сэмпла. 16 бит числа-накопителя не особо страдают от переполнения, в них остаётся некое значение, к которому можно снова и снова прибавлять. Таким образом, время от времени происходит переполнение и перемещение указателя **R2** вперёд по сэмплу. Скорость перемещения зависит от числа в **R0** – чем оно больше, тем чаще случаются переполнения. Команда **MOVB** (Move Byte) постоянно извлекает 8-битные данные из памяти, куда указывает **R2**, и записывает их в Covox. То же самое происходит с регистрами **R1** и **R3** для второго сэмпла. На Covox посылаются сначала значение басового сэмпла, затем, через пару



команд, значение сольного сэмпла, они чередуются в цикле. Когда счётчик цикла **R4** доходит до нуля, нужно прочитать значения следующей пары нот (этот код я не привожу). Если вы внимательно посмотрите программу, то заметите, что сольный инструмент держится в Covox'e немного дольше басового. Поэтому соло звучит чуть громче баса.

Что ж, причину моего удивления вы, возможно, уже поняли: зачем по очереди посылать на Covox то один сэмпл, то другой? **Сейчас так никто не делает!** Надо просто сложить значения, а сумму послать на Covox один раз. Очевидно же!..

Но вот в чём дело: прогресс работает не так. Я снял целый [выпуск](#) видеоподкаста SCENE, где на примере трекерной музыки для PC показал, что решения, теоретически воплотимые в начале 90-х, не были никем использованы аж до конца 2000-х, потому что в 90-е уровень идей и образ мышления ещё не развился до нужного уровня. В этом смысле экспериментальная археология (когда современные люди со всем багажом знаний XXI века пытаются из камней и палок собрать чудо техники, «доказывая», что древние могли сделать то же самое) теряет всякий смысл.

Так почему же в COVOX.DEM такой странный алгоритм? Избегая операции сложения, **RDC** добивался высокой частоты дискретизации звука (15,3 КГц на БК-0010). Тогда казалось, что самое главное – высокая частота. Этот подход издавна применяли в программировании музыки для однобитного пьезодинамика и, похоже, программистам трудно было с ходу перестроить мышление на новый лад.

Ещё раньше, в июне 1993-го года, вышла демка **RDC Amiga11M**, она воспроизводила двухголосую музыку на Covox, на музыкальный чип AY и на однобитный пьезодинамик. Во всех трёх режимах она работала одинаково – чередовала однобитные данные из двух заранее записанных музыкальных фрагментов. Никакой арифметики с нотами и скоростями проигрывания там не было. Программа требовала 128 килобайт памяти БК-0011M, не запускалась на

обычной БК-0010 и поэтому прошла мимо меня.



Самой известной и впечатляющей Covox-демкой **RDC** стала **STDEMO**, выпущенная в марте 1994-го года. Но ещё до этого, если верить статье из электронного журнала **SPCinfo 1995**, я написал свою двухголосую демку. Затрудняюсь вспомнить её название, возможно **BEATLES**. По крайней мере, на меня это похоже: два года спустя я написал 4-килобайтную демку с таким же названием. 17-й номер электронного журнала **КПСС** сообщал, что к марту 1994-го года на БК существовало как минимум четыре музыкальных демки для Covox: **AMIGA11M**, **BEATLES**, **COVOX.DEM** и **STDEMO**. Пользователи БК любили переименовывать файлы во что-нибудь типа «covox1», поэтому сейчас затруднительно найти в архивах программу с названием **BEATLES** (тем более что у неё есть тёзка).



Мне попала только одна программа, подходящая под описание из **SPCinfo**. Вот код её главного цикла:

```
PLAY: ADD -(R5),R1 ; Add frequency integer
      ADD -(R5),-(R5) ; Add frequency fract
      ADC R1 ; R1: pointer to sample 1
```



```

MOVБ (R1),R3      ; Get sample 1 data
ADD -(R5),R2      ; Add frequency integer
ADD -(R5),-(R5)   ; Add frequency fract
ADC R2            ; R2: pointer to sample 2
MOVБ (R2),-(SP)   ; Store sample 2 data
ADD (SP)+,R3      ; Add sample 2 data
MOVБ R3,@#177714; Put mixed data to Covox
ADD #14,R5        ; Reset pointer to variables
SOB R0,PLAY       ; Loop

```

Регистр **R4** не задействован, я использовал его в другом месте для ускорения чтения информации о нотах. Тогда казалось важным, чтобы между нотами не возникало пауз. В то время мы ещё нащупывали подходы, не знали толком, что имеет решающее значение, а чем можно пренебречь. Да что говорить, при том уровне аудиотехники мы даже толком не слышали, что наши сэмплы содержат щелчки, а алгоритмы портят звук.

Но вернёмся к программе.

- **R1** указывает на хранящийся в памяти сэмпл первого инструмента;
- **R2** указывает на хранящийся в памяти сэмпл второго инструмента;
- **R5** указывает на блок переменных.

Что это за переменные:

- 1) число-накопитель для первого сэмпла;
- 2) дробная часть ноты (скорости) первого сэмпла;
- 3) целая часть ноты (скорости) первого сэмпла;
- 4) число-накопитель для второго сэмпла;
- 5) дробная часть ноты (скорости) второго сэмпла;
- 6) целая часть ноты (скорости) второго сэмпла.

Регистр **R5** указывает на адрес, следующий за блоком переменных. Запись **-(R5)** означает, что нужно сперва уменьшить **R5** на два, а затем взять 16-разрядное число из адреса, на который указывает **R5**. Алгоритм считывает целую

часть скорости и прибавляет к **R2** (это указатель на сэмпл). Затем прибавляет дробную часть к накопителю, и если получилось переполнение, увеличивает указатель **R2** ещё на 1. Благодаря введению целой части, можно достичь сколь угодно высокой скорости перемещения по сэмплу. Идею использовать эмуляцию вещественной арифметики с фиксированной точкой я почерпнул из обсуждений с **RDC**. Но вместо переменной выдачи сэмплов в Covox я стал суммировать их в регистре **R3**. Сейчас сложно вспомнить, но, возможно, идея с суммированием также родилась во время общения с **RDC** — я полностью преклоняюсь перед его гением и не претендую на роль изобретателя.

Особое место в программе — запись числа в стек и последующее сложение с **R3**. Процессор БК не может прибавить байт к 16-битному слову. Да и сложить два байта не может: команда **ADDB** в процессоре отсутствует, складывать можно только 16-разрядные числа. Поэтому мне пришлось преобразовать байт в 16-разрядное слово. Для этого использован стек. В ассемблере БК (и вообще DEC PDP) регистр стека обозначают **R6** или **SP** (Stack Pointer). Оценивая этот алгоритм сейчас, я вижу, что автодекрементная адресация **-(SP)** была лишней, как и автоинкрементная **(SP)+**. В обоих случаях можно было обойтись обычной косвенной адресацией **(SP)**. Главное заранее подготовить ячейку памяти по адресу, куда указывает **SP** — записать ноль в её старший байт.

Ещё один важный момент: складывая байты, мы должны следить, чтобы результат не превысил 255, иначе при записи в 8-битный Covox часть данных потеряется. Проверять переполнение внутри основного цикла накладно, поэтому громкость всех сэмплов я заранее снизил, переведа их в 7 бит. Кто-то может предложить более радикальное решение: спаять 9-разрядный цифро-аналоговый преобразователь — и качество сэмплов не пострадает, и никакого переполнения! Но как быть, если мы решим микшировать не два, а больше каналов — припаивать ещё разряды? Стандартом был всё же 8-разрядный Covox.



Предпоследняя команда возвращает указатель **R5** в конец блока переменных. Всего было обработано 6 переменных, так что нужно увеличить **R5** на 12 (каждая переменная занимает два байта). В восьмеричной системе число 12 выглядит так: 14. Да, на БК все числа записывают в восьмеричной системе счисления.

Отмечу деталь, которая не попала в этот фрагмент кода: вместе с каждой парой нот хранилась их длительность, она считывалась в регистр **R0**.

На БК-0010 алгоритм работал с частотой микширования 8,4 КГц – это чуть больше исходной частоты сэмплирования инструментов.



STDEMO 1994-го года стала прорывом – она на лету синтезировала полифоническую 4-голосую музыку, смешивая в реальном времени сэмплы инструментов, да ещё показывала немножко анимации. Мелодию написал известный музыкант Дмитрий Комиссаров (**XPEH**) на компьютере Commodore Amiga, используя программу **Sound Tracker**. Но **RDC** решил назвать демо по имени более известного музыкального редактора для IBM PC.

Услышав **STDEMO** в гостях у **RDC** на его БК-0011M, я был потрясён – четыре голоса, как на PC! Сразу зашла речь о создании проигрывателя и музыкального редактора для БК. Но сам **RDC** за такую задачу браться не стал – у него было множество других не менее интересных проектов для разных платформ.

Код **STDEMO** выглядит так:

```
PLAY: ADD #100000,#12345 ; Channel 1
      ; frequency
ADC R1      ; R1: pointer to sample 1
MOVB (R1),R0 ; Get sample 1 data
ADD #100000,#12345 ; Channel 2
      ; frequency
ADC R2      ; R2: pointer to sample 2
MOVB (R2),R5 ; Get sample 2 data
ADD R5,R0   ; Mix samples 1 and 2
ADD #100000,#12345 ; Channel 3
      ; frequency
ADC R3      ; R3: pointer to sample 3
MOVB (R3),R5 ; Get sample 3 data
ADD R5,R0   ; Mix sample 3
ADD #100000,#12345 ; Channel 4
      ; frequency
ADC R4      ; R4: pointer to sample 4
MOVB (R4),R5 ; Get sample 4 data
ADD R5,R0   ; Mix sample 4
MOVB R0,@#177714 ; Put mixed data to
      ; Covox
DEC #4000   ; Note duration
BNE PLAY   ; Loop
```

Числа, отвечающие за частоту ноты (#100000) для каждого канала, записаны прямо в теле цикла. Так что по окончании цикла приходится модифицировать код, записывая новые значения частот. Цикл сделан не командой **SOB** (свободных регистров не осталось), а вычитанием единицы (**DEC**) из числа и переходом, если не ноль (**BNE**).

Сэмплы в **STDEMO** были даже не 8-битные, а 6-битные, чтобы избежать переполнения при сложении четырёх каналов. Частота микширования 7,6 КГц.

Как я уже говорил, **RDC** не взялся за написание музыкального редактора. Поэтому редактор написал я.

Но сначала произошли две вещи:

1. **RDC** познакомил меня с **XPEH**'ом, а **XPEH** – с **Tangerin**'ом – эти два музыканта сильно повлияли на моё участие в трекерной сцене и вообще в компьютерном андеграунде, в демосцене. Мы дружим до сих пор, и у нас даже есть [совместный музыкальный альбом](#).





Слева направо: Tangerine, XPEh, Manwe

2. Я разработал формат файлов для будущего трекера, набил для пробы три мелодии (прямо кодами в отладчике), написал проигрыватель (он запускался на обычной БК-0010) и отдал **Вадиму Новаку** из «АльтПро». Фирма «АльтПро» рассылала почтой дискеты с программами по всей стране, благодаря чему началось бурное распространение трекерной музыки среди владельцев БК.

В конце восьмидесятых — начале девяностых годов в 1155-ой школе московского района Строгино образовалась компания любителей и владельцев БК-0010. В старших классах четверо из моих друзей имели у себя дома БК-0010. Я был пятым. У остальных друзей компьютеров не было вообще. Этот феномен объясняется, должно быть, школьным классом информатики с БК-0010Ш: родители учеников гораздо охотней поддавались на уговоры купить БК, так как на этой «серьёзной» машине проходили школьные уроки и производственная практика. Имея такой компьютер дома, ученик мог повысить успеваемость по информатике и даже подготовиться к серьёзной профессии. Думаю, авторитет государственного образовательного учреждения сильно влиял на решение родителей: БК — это не какая-то игрушка типа «Спектрума»... Чуть позже к нам присоединились ребята из других школ. На моей улице жил и создатель операционной системы MKDOS Михаил Королёв (**МКТ**). В общем, строгинское БК-сообщество было весьма активным.

Так и появилась **Strogino Programming Company** (сокращённо **SPC**). Никнейм **Manwe** я тогда ещё не использовал, да и вообще предпочитал выступать не лично, а от имени коллектива.

Эксперименты быстро привели меня к решению отказаться от 4-голосой полифонии: обработка четырёх голосов отнимала слишком много процессорного времени, частота микширования получалась низкой, качество было просто ужасным. STDEMO, конечно, очень крутая демка, но на то она и демка, чтобы удивлять каким-то одним специально подобранным частным случаем. В общем случае произвольные инструменты на низкой частоте звучали очень плохо. Кроме того, как музыкант я знал, что в реальной музыке редко звучат все 4 канала одновременно. Хорошие трекерные музыканты могли уложиться даже в два канала. Ещё один аргумент: нотный текст для четырёх каналов занимает больше места, чем для трёх. Памяти у БК и так мало, лучше оставить её под сэмплы.

Но и на обработку трёх каналов не хватало регистров, а значит приходилось производить вычисления в ячейках памяти, что замедляло программу и снижало качество. На компромиссы я был не готов — к тому времени ещё один мой друг из Строгино, связанный с БК-0010 — **Роман Заернюк** — познакомил меня с самыми современными музыкальными картами для PC и с настоящей японской Hi-Fi техникой, мы слушали трекерную музыку с PC и Amiga на максимально качественной акустике, пути назад в мир Lo-Fi уже не было. И тут у меня случилось, буквально, озарение. Правда, оно вряд ли произошло без опыта работы в Scream Tracker на PC. В общем, **барабаны**. Барабаны всегда звучат с той же частотой дискретизации, на которой были оцифрованы, всегда на одной и той же ноте. И в 1994-м году вряд ли могли встретить компьютерную музыку без барабанов. Поэтому один из каналов я выделил полностью под барабаны и лишил этот канал возможности изменять частоту. Всё ускорило. Регистров хватило. Качество выросло. Волшебство!



Ну а если кто-то хотел играть барабаном разные ноты – что ж, для этого есть ещё два канала.



Новогодние каникулы с 1994-го на 1995-й год ушли на эксперименты с новым форматом, результатом стала демка **SSTdemo**. Как видите, слово «comprau» написано странно – у меня был чёрно-белый монитор и, рисуя буквы, я промахнулся на полпикселя (у БК в цветном режиме два бита на пиксель, а в чёрно-белом – один бит, но разрешение по горизонтали вдвое выше). Несмотря на то, что мелодию я набивал без редактора, прямо в кодах, результат получился симпатичным, и пару лет спустя я даже перенёс его на PC в Scream Tracker 3.

В отличие от BEATLES (или как там называлась моя первая программа для Covox), здесь во внутреннем цикле использовались все регистры. **R6** (он же **SP**) указывал на блок переменных. Частота ноты задавалась только дробной частью, целая часть была упразднена. Восстановление указателя на блок параметров зачем-то делалось через стек (это уменьшало внутренний цикл на 2 байта, но не ускоряло его). Поиск оптимальной процедуры проигрывания ещё не был завершён. Но формат данных уже почти устоялся.

```
PLAY: ADD -(R6),-(R6) ; Add fract
      ADC R1          ; R1: pointer to sample 1
      ADD -(R6),-(R6) ; Add fract
      ADC R2          ; R2: pointer to sample 2
      ADD -(R6),R     ; Reset pointer to variables
      MOVB (R1),R     ; Get sample 1 data
      MOVB (R2),R     ; Get sample 2 data
      ADD R4,R5       ; Mix samples 1 and 2
      MOVB (R3)+,R4   ; Get drum data
      ADD R4,R5       ; Mix drum
      MOVB R5,(R0)    ; Put mixed data to Covox
DEC #4000 ; Loop
BNE PLAY
```

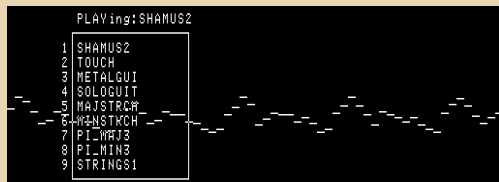
Интерес представляет регистр **R3**, который указывает на сэмпл барабана. Благодаря автоинкрементной адресации он всегда увеличивается на 1, скорость воспроизведения фиксирована. Но иногда в музыке встречались моменты, когда барабаны не нужны. В таких случаях код модифицировался: аргумент **(R3)+** заменялся на **(R3)**.

На БК-0010 частота микширования получилась 9,3 КГц, а на более быстрой БК-0011М – 10,4 КГц.

Программа SSTdemo хранила музыку в удобном для проигрывателя формате, но этот формат был неудобен для музыканта. Хотелось чего-то простого, наглядного, текстового. Я разработал такой формат и назвал его **SST**, от [Strogino Sound Tracker](#). Ноты и инструменты задавались уже не частотами и адресами, а порядковыми номерами. В этом формате я набил две мелодии: музыку из популярной тогда на БК игры «Солдат» и из ещё более популярной на PC игры **Prehistorik II** (в оригинале музыка была цифровая, но я не стал заимствовать сэмплы из самой игры, а подобрал похожие из своей коллекции). Обе мелодии я часто играл на фортепьяно в полифонической аранжировке, поэтому без труда набил их в отладчике по памяти.

Был написан и проигрыватель – **SSTplay**. При запуске он предлагал выбрать конфигурацию БК (модель, частоту), и в зависимости от выбора играл музыку двумя разными алгоритмами (на разогнанных машинах микширование получалось более качественным, так как сэмплы не обрезались до 6 бит и при сложении точность не терялась). Проигрыватель загружал в память файлы с нужными сэмплами, несколько секунд конвертировал нотный текст в частоты и адреса, после чего играл музыку.

Помимо этого я написал **WAVEplay** – программу для MKDOS с весьма нестандартным интерфейсом.



Она позволяла загружать с диска сэмплы и играть ими на клавиатуре БК как на фортепьяно. Это и стало первым релизом в июле 1995-го года: WAVEplay, SSTdemo, SSTplay, несколько мелодий и десятка два сэмплов.

Андрей Тимофеев (**Terra Software**) эмоционально писал в 1995-ом году: «Да RDC COVOX.DEM и STDEMO фигня по сравнению с SST'шными музонами. Я решил покопаться и разобраться с форматом нотного текста и самим проигрывателем. Формат текста описан мной полностью. Проигрыватель мне очень даже понравился: интересные идеи, нестандартные приёмы программирования». Что ж, музыка звучала действительно круто. Все, кто её слышал, торопили: ну когда же будет готов редактор?

БЛОК : 01		ТЕМП : 30
11		
12	04-3-DO	01
13		
14	04-3-DO	
15		
16	03-3-DO	02
17		
18	03-3-DO	02
19		
20	04-3-DO	02
21		
22	04-3-DO	02
23		
24	05-3-DO	02
25		
26	05-3-DO	02

1.	BUZZ - (02)	5. ERASURE4 - (01)
2.	HEATBEAT - (02)	6. SYNTHTOO - (01)
3.	ERASURE2 - (01)	7. ZSNARE - (01)
4.	ERASURE3 - (01)	8. LEED - (02)
ОКТАВА : 01		ИНСТРУМЕНТ : 01

Первая версия была написана ещё летом 1995-го года и позволяла только редактировать нотный текст, но не слушать музыку. Приходилось сперва сохранять работу на диск в формате SST, затем запускать SSTplay и ждать, пока всё сконвертируется. Тем, кто привык компилировать программы в объектные модули и затем собирать линковщиком, подобные мелодвижения не покажутся странным. Профессиональные композиторы тоже стерпят: они пишут в нотной тетради или «в голове», им не обязательно часто прослушивать результат. Но большинство пользователей БК, конечно, не такие. Меня самого жутко раздражали все эти лишние

действия. За неимением лучшего, я набрался терпения и набил в редакторе ещё пару мелодий, после чего принёс показать Вадиму Новаку в «АльтПро». Сделать это было просто – «АльтПро» размещалась на последнем этаже института, где я в то время уже учился. В «АльтПро» я был частым гостем.

Вадим забрал у меня редактор и сказал, что будет рассылать его почтой. Я протестовал как мог: сырая неудобная версия, такое нельзя никому показывать! Мелодии – пожалуйста, но не редактор. Вадим убеждал меня: мол, я даже не представляю, как люди ждут и на какие мучения готовы ради возможности писать цифровую музыку на БК.

Конечно, я раздал редактор друзьям на тестирование. Каково же было моё удивление, когда Саша Ильин (**Real**) принёс сделанную им мелодию. Она звучала очень круто! Вот уж не думал, что кто-то сможет преодолеть все неудобства интерфейса и создать действительно классное произведение.

А потом я в очередной раз зашёл в «АльтПро» с новой порцией мелодий.

– погоди, не уходи, – сказал Вадим под конец и протянул пачку денег, сравнимую с месячной стипендией.

- Это что? – удивился я.
- За Strogino Sound Tracker.
- Какой трекер? Он ещё не готов.
- За демо-версию. Люди активно покупают.

Я был шокирован. И тем, что Вадим всё-таки начал продавать редактор, и тем, как он оказался востребован, и тем, что у меня появилась новая статья дохода. Когда ко мне вернулась способность соображать, я осознал: надо срочно доводить редактор до ума.

Первым делом я прикрутил возможность проигрывать текущий паттерн нажатием одной клавиши, не выходя из редактора. Это был «костыль»: чтобы освободить память, редактор сохранял сам себя во временный файл, незаметно конвертировал SST во внутренний формат, запускал проигрыватель, а после проигрывания паттерна загружал себя обратно.



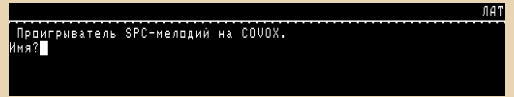


Вторым делом я нарисовал заставку по мотивам Scream Tracker 3.

Третьим делом... Третьим делом я решил, что нужно писать новый редактор. В формат SST были заложены неиспользованные пока возможности: глобальные громкости инструментов и sample offset (он давал возможность эмулировать громкость нот и оживлять скучные инструменты, придавая каждой ноте чуть разное звучание). Для всего этого требовался совершенно другой интерфейс. Перспектива проектировать интерфейс с нуля выглядела пугающей, поэтому я просто отнёс недоделанный редактор в «АльтПро» и вместо программирования с головой погрузился в сочинение музыки.

Музыки накопилось много. Я понял, что обычным слушателям удобней хранить её сразу во внутреннем формате проигрывателя (как было сделано в SSTdemo). Поэтому написал конвертер **SST2SPC** и проигрыватель **SPCplay** (от Strogino Programming Company). Файл с расширением **SPC** содержал частоты нот и адреса инструментов. Файл с расширением **INS** хранил используемые в мелодии инструменты. Объединять ноты с инструментами в один файл (как сделано на PC и Amiga) я не стал: удобней грузить их раздельно по фиксированным адресам памяти, это упрощало и укорачивало программу. **MKT** придумал сделать **SPCplay** частью своей дисковой операционной системы **MKDOS**, чтобы запускать **SPC**-файлы одним нажатием клавиши прямо из оболочки – всё подставлялось и загружалось автоматически, сразу начинала звучать музыка. Можно было и по старинке загружать мелодии с магнитофона, набивая имя файла вручную. **SPCplay** работал даже на БК-0010 минимальной конфигурации в любой операционной системе.

Думаю, это поспособствовало его распространению.



```

MOV #4000,R0 ; Note duration
PLAY: ADD #100000,#12345 ; Channel 1 speed
ADC R1 ; R1: pointer to sample 1
ADD #100000,#12345 ; Channel 2 speed
ADC R2 ; R2: pointer to sample 2
MOVB (R3)+,R5 ; R3: pointer to drum sample
MOVB (R2),R4 ; Get sample 2 data
ADD R4,R5 ; Mix samples 2 and drum
MOVB (R1),R4 ; Get sample 1 data
ADD R4,R5 ; Mix sample 1
MOVB R5,@#177714; Put mixed data to Covox
SOB R0,PLAY ; Loop
    
```

По сравнению с SSTplay и SSTdemo, этот алгоритм получился на 20% быстрее, он выдавал частоту микширования 12,5 КГц на БК-0011М. Немного изменился и формат данных. Считывая очередную порцию нот, SSTdemo смотрел, что записано в канале барабанов: если 1, то в код помещалась команда **MOV** с аргументом **(R3)+**, а если 0, то просто **(R3)**. В **SPCplay** я пошёл дальше: сам код команды **MOV** с нужным аргументом хранился в **SPC**-файле, откуда считывался наравне с частотами нот и адресами сэмплов. Сейчас я понимаю, что при желании можно было внедрить в **SPC**-файл эксплойт, заменив **MOV** на что-то другое, и выполнить произвольный код. Тогда о подобных вещах на БК не думали.

На дворе стоял 1995-й год, когда у нас с друзьями родилась идея провести первую в России демопати. Мы были заморожены демосценой на PC и Amiga, у нас дома появились PC-совместимые компьютеры, модемы и доступ к BBS, откуда можно было скачать свежие демки для PC, а также европейские электронные журналы. В общем, мы узнали о том, что такое демопати. Нам захотелось так же. Я пришёл к Вадиму Новаку с идеей бросить клич среди его покупателей: мол, вот вам редактор, пишите музыку и привозите на конкурс в



Москву – это будет замечательная пати с классной атмосферой и призами. А ещё пишете демки: в 4 килобайтах делаете что хотите, зрители оценят и выберут победителей. К тому времени на БК уже существовали демки, и люди в общем понимали, о чём идёт речь. Идея Вадиму понравилась, он сразу предложил призы от «АльтПро», но сказал, что для распространения информации потребуется гораздо больше времени, чем я могу себе представить. В 1995-м году мы не успевали. Решено было перенести на 1996-й год. Так «БК-мания» не стала первой демопати в России, уступив пальму первенства питерскому Enlight'у.

Осенью 1996-го года на демопати «БК-мания» съехались владельцы БК со всей страны. Такого количества демок и музыки я не ожидал. А главное, такого качества. Особенно меня поразили композиции Максима Полинцева (**Maxi G.C.**) из Самары. Победителем в конкурсе демо стал **Insult**, в создании которого **Maxi** также принимал участие. **Insult** играл музыку на AY и на Covox (стандартной подпрограммой SPCplay). Сам **Maxi** не приехал на демопати, но чуть позже был проездом в Москве, мы познакомились и быстро сдружились.



Ещё в середине 1996-го года я понял, что 8 инструментов – слишком мало, и в качестве эксперимента набил в отладчике [музыку](#) финского демосценера Purple Motion. Мелодия

называлась «Minimum Velocity» и использовала много инструментов. На «БК-мании 96» с её помощью тестировали звуковую аппаратуру. В музыкальном конкурсе «Minimum Velocity» не участвовала, поскольку туда допускали только работы собственного сочинения. Теперь эта программа входит в стандартный набор [эмулятора БК](#) и называется почему-то **covox.bin** (о уж эта любовь пользователей БК всё переименовывать). Эксперимент убедил меня в неизбежности создания нового формата.

После бесконечных обсуждений демосцены и трекерной музыки, мы с **Maxi** приняли следующий чёткий план действий: **Maxi** делает проигрыватель под CSI-DOS, я разрабатываю новый формат SST2, а дальше мы вместе как-нибудь что-нибудь придумаем. На первых двух этапах план сработал блестяще. Мы даже его перевыполнили: помимо проигрывателя, **Maxi** написал прототип редактора, а я – демку **SSTintro2**, которая обещала новую ускоренную процедуру проигрывания (но использовала старую).

SIREGINO

HIGH QUALITY 8 BIT COVOX
SUPPORT OLD SST FORMAT
SCREAM TRACKER SAMPLES
NEW FAST PLAYER DRIVER

Sound Tracker

Особенность процессора БК в том, что при загрузке байта из памяти в 16-разрядный регистр страдают старшие разряды этого регистра: они заполняются копиями седьмого бита (это называется «расширением знака»). SSTplay, если его запускали на турбированном компьютере, активировал особую процедуру микширования: считывал 8-битные данные



трёх каналов в регистры, на ходу сбрасывал им старшие биты, суммировал. Получившийся 10-разрядный результат делил на 4 двумя командами арифметического сдвига.

Идея новой процедуры (код которой был опубликован ещё в конце 1995-го года в электронном журнале [SPCinfo](#)) состояла в том, чтобы заранее преобразовывать 8-битные сэмплы в 16-битные. Это вдвое повышало расход памяти, но позволяло выполнять команду **ADD** прямо по указателю сэмпла, минуя переброску байта в промежуточный 16-разрядный регистр. Меньше команд – выше скорость и качество. Сэмплы переводились не только из 8 бит в 16, но также из беззнакового формата (диапазон 0..255) в знаковый (-128..127).

Полностью алгоритм можно посмотреть в [SPCinfo 95](#), а здесь я приведу только процедуру микширования.

```
MOV #177714,R5 ;Covox port address
PLAY: ADD #100000,#12345 ; Channel 1 speed
      SBC R1 ; R1: pointer to sample 1
      ADD #100000,#12345 ; Channel 2 speed
      SBC R2 ; R2: pointer to sample 2
      MOV (R3)+,R4 ; R3: pointer to drum sample
      ADD (R2)+,R4 ; Mix sample 2
      ADD (R1)+,R4 ; Mix sample 1
      MOVB TABL(R4),(R5) ; Convert 10 bit to 8 bit
      SOB R0,PLAY ; Loop
```

На смену команде **ADC** (прибавить бит переполнения) пришла команда **SBC** (вычесть бит переполнения), потому что автоинкрементная адресация приводит указатели **R1**, **R2** и **R3** в постоянное движение. Команда **SBC** затормаживает это движение. Максимум, на что она способна – замедлить проигрывание сэмпла в два раза. Получается, что диапазон нот – всего одна октава. Модифицируя код, можно заменить **SBC** на **ADC**, это раздвинет диапазон примерно на половину октавы вверх. Также можно расширить диапазон, храня варианты инструментов для разных октав (например, в комплекте с редактором шли инструменты **PIANO1**, **PIANO2**... **PIANO5**).

Предпоследняя команда использует табличный метод адресации. Заранее была

посчитана и сохранена в память таблица перевода 10-битных значений в 8-битные. **TABL** – это адрес середины таблицы, а **R4** – индекс в таблице (он мог принимать значения от -128*3 до 127*3). Хитрость в том, что данные в таблице исполняли роль компрессора динамического диапазона (максимайзера): усиливали тихие звуки и не давали громким звукам превышать допустимый уровень. Идею подал Александр Васильевский (**Tarh**) – мы с ним учились в МИЭМ в одной группе и сдружились на почве интереса к гитарам, ассемблеру, демкам и харкству. Описание табличного компрессора **Tarh** обнаружил в технической документации к **Scream Tracker 3**.

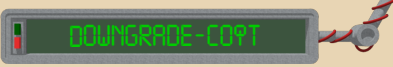
Замер производительности нового алгоритма на БК-0011М показал частоту микширования 13,9 КГц.

А что же с форматом SST2? У меня сохранилась распечатка следующего содержания:

```
Strogino Sound Tracker 2. File format
specification (c) Strogino Programming Company
1996.
07500 Song name
07600 Author
07640 Volumes (for 32 samples)
07700 Order list (56 positions)
07770 «SST2»
07774 Loop order (number in Order list for cycling
song)
07775 1 (byte=001)
07776 Temp (word in 01000..40000 range)
10000 Sample file name (8 symbols only) |
10010 Length (without header) |
10012 Loop start offset |32 times
10014 Loop end offset |
10016 Sample rate (must be 177777) |
11000 Pattern data
```

Также был придуман формат хранения инструментов на диске. Инструмент содержал не только голый сэмпл, но и длинное название (32 символа), информацию о точках цикла, громкости, частоте сэмплирования, зарезервированные поля. Сейчас мне кажется, что инструменты были лишней затеей. **INS**-файл





предполагалось оставить совместимым со старым форматом (просто набор голых сэмплов безо всяких опознавательных знаков).

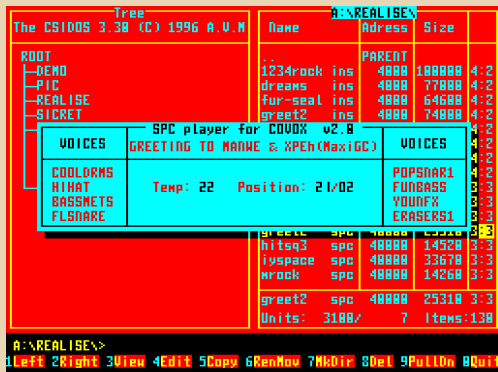
По счастью, после конвертации SST-файла сэмплы в INS-файле сохранялись в своём изначальном 8-разрядном виде (разве что обрезались покороче, соразмерно тому, какой длительности ноты использовались в музыке). SPCplay после загрузки INS-файла делил все данные на 4, чтобы при микшировании не возникло переполнения. В следующей версии **SPCplay 1.2** я догадался с помощью таблицы делить сэмплы на 3 – громкость музыки возросла, качество слегка повысилось. Поле **Volume** было заложено в формат SST на будущее, оно содержало глобальную громкость для каждого сэмпла. Предполагалось, что следующая версия SPCplay будет учитывать **Volume** и делить каждый инструмент на индивидуальное значение. Это немного усложняло проигрыватель, но некритично. Основная проблема была в интерфейсе музыкального редактора – как управлять глобальными громкостями.

Новый формат содержал и базовую частоту сэмпла – в теории это позволяло строить разные таблицы нот для сэмплов разной частоты. В этом случае нагрузка ложилась не на проигрыватель SPCplay, а на конвертер SST2SPC. Но главной проблемой снова оставался интерфейс.

SST2 подразумевал использование зацикленных сэмплов. Как реализовать зацикливание программно, я представлял лишь смутно. Это был задел на будущее.

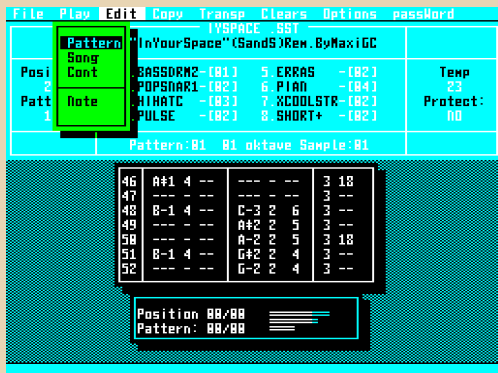
Тем временем, **Maxi G.C.** написал проигрыватель SPCplay под CSI-DOS. CSI-DOS была прогрессивной операционной системой для БК-0011М: как и MKDOS, она позволяла привязывать действия к расширениям файлов, SPC-музыку можно было прослушать по нажатию одной клавиши. Кроме того, CSI-DOS предлагала программисту API для создания оконного

интерфейса, поэтому написать симпатичный проигрыватель было относительно несложно.



В начале 1997-го года **Maxi G.C.** создал и [редактор музыки](#), совместимый с форматом SST. Он также назывался **SST**, но был написан с использованием API CSI-DOS и работал только в этой системе. В отличие от моего редактора, стеснённого ограничениями БК-0010, этот хранил сэмплы в дополнительных страницах памяти БК-0011М и мог проигрывать их в любой момент. Такая возможность изменяла процесс написания музыки кардинально!

Нажимая ноту, музыкант сразу слышал звук. По клавише **Tab** можно было прослушать ноты всех трёх каналов, расположенных на текущей строке. А ещё редактор поддерживал sample offset. Теперь на БК можно было создавать цифровую музыку так же просто, как на PC и Amiga!



Понятно, почему я был недоволен своим недоделанным Strogino Sound Tracker, но удивительно, что и **Maxi G.C.** стеснялся своего редактора: якобы, он сделан на скорую руку, кое-что из задуманного не работало. В любом случае, доводить до ума эту версию **Maxi** не собирался, так как мы ставили более амбициозную задачу: создать редактор под новый формат SST2 и под любые операционные системы. Ну а пока пользователи CSI-DOS довольствовались тем, что было: вполне хорошим, на мой взгляд, редактором. Музыканты написали на нём множество композиций. Когда на следующий конкурс «БК-мания 97» поступили работы из Самары, стало ясно, насколько удобный инструмент развязывает руки творцу и повышает качество музыки.

Подводя итог изысканиям 1993-1995 годов, можно составить хронологическую таблицу алгоритмов:

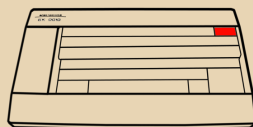
Про-грамма	Автор	Число голосов	Частота в Гц на БК 0011M	Особенность
COVOX.DEM	RDC	2	19734	Первая программа для Covox на БК
BEATLES	Manwe	2	10000	Название программы точно неизвестно
STDEMO	RDC	4	7576	Новаторский анимационный эффект
SSTDEMO	Manwe	3	10417	Впервые опробован формат SPC+INS
SSTplay	Manwe	3	10417	Дебют формата SST
SPCplay 1.1	Manwe	3	12500	Оптимальный алгоритм проигрывания
SPCinfo 95	Manwe	3	13889	Двойной расход памяти, максимайзер

Планам относительно Strogino Sound Tracker 2 не суждено было сбыться. Обычно в этом месте подобные истории печально заканчиваются: энтузиасты выросли и остепенелись, их засосала работа и семейная жизнь, советский компьютер БК был вытеснен с рынка китайскими клонами IBM PC, всё плохо, кладбище, гроб, могила... Но не в нашем случае! В следующий раз я расскажу про невероятно насыщенный период с 1997 по... 2019 год.

Прим. ред.: упомянутые в статье Covox-демки можно скачать здесь:

<https://www.pouet.net/prod.php?which=82746>

Александр Мачуговский (Manwe)





ДОКТОР ТВОЕГО МОДЕМА

К аких только воспоминаний, порой, не навеет тематический номер и просмотр Old-DOS'a. ☺ Так, например, когда искал тестовые программы для Windows 3.x, вспомнилась утилита, описанная когда-то в книжке Леонтьева – Modem Doctor. Леонтьев писал, что программа восходит ещё к эпохе DOS, что уже взволновало. А уж то, что программа и умелая, и счастливо дожила, как минимум, до 2004-2005-го года, когда была написана и выпущена книжка, внушало определённое уважение. А то, что программа имела версии и для DOS, и для Win32, наводило на мысли о существовании версии для Windows 3.1. Однако, с наскаку, найти такую версию для третьей винды не удалось, а статья с тестами и так получилась достаточно объёмистой. Посему, Modem Doctor'у было решено посвятить отдельную статью, где рассмотреть версию для DOS. Однако поиск всё же дал результат, и для Windows 3.x также была найдена версия – Modem Doctor for Windows 1.0. Поэтому в данной статье будут рассмотрены обе версии этой утилиты.

В качестве тестовой машины будет выступать материнская плата с 486DX2 66 МГц и 8 МБ ОЗУ. В качестве модема автор взял ISA-плату, которая верой и правдой служила ему 13 лет назад, снабжая ещё компьютер на 5x86

под управлением третьей винды интернетом – US Robotics Sportster 33.6k (0374C) – см. рис. 1.

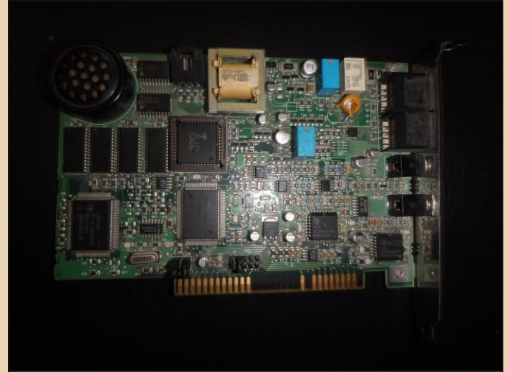


Рис. 1. US Robotics Sportster 33.6K 0374C – именно на этом красавце и будем испытывать обе версии Modem Doctor'a

К сожалению, подключиться к интернету через PPP сложновато, если возможно, но просто протестировать модем – почему бы и нет. У автора ещё и присутствует кабельный телефон, так что с линией тоже проблем не будет.

Ну что ж, начинаем.

Modem Doctor для DOS

В отличие от винدوزных версий, среди реализаций Modem Doctor'a для DOS можно немного повыбирать. На Old-DOS доступно (на сегодняшний день – лето 2019-го ☺) две версии программы – 7.0S 96-го года и 2.1S – 89-го. Автор решил поэкспериментировать с более свежей версией, то есть седьмой. Скачать её можно по ссылке:

<http://old-dos.ru/dl.php?id=14698>

Сразу после запуска программа опросит все доступные последовательные порты на наличие модема и выведет на экран соответствующую информацию кратким списком (см. рис. 2), где вы увидите наличествующие в вашей системе последовательные порты, настроенные на них прерывания, а также сообщение о наличии модема.



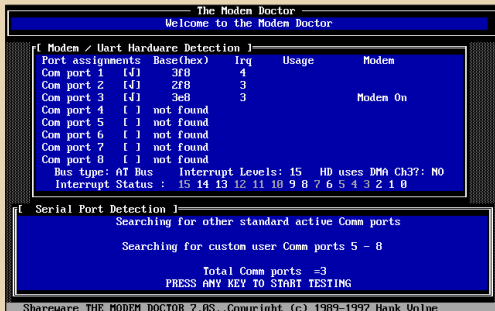


Рис. 2. Сразу после запуска программа протестирует все доступные последовательные порты на наличие модема и выведет соответствующую информацию. В нашем случае внутренний модем настроен на COM3, но занимает третье прерывание (опытным путём установлено, что, если поставить правильно – четвертое – при обращении к модему в винде будет виснуть мышь, так что такая настройка выбрана на перспективу)

После нажатия любой клавиши на клавиатуре программа спросит вас о том, какой порт будете тестировать, а также на какой скорости будет проверяться модем. К сожалению, присутствует только три варианта – 900, 1200 и 2400 бит/с (см. рис. 3). Правда, при этом в процессе начального тестирования модема всё же определяется поддержка протокола V.34+ и скорости 33600 бит/с (см. рис. 4).

Кроме того, программа покажет значения регистров модема (см. рис. 5)

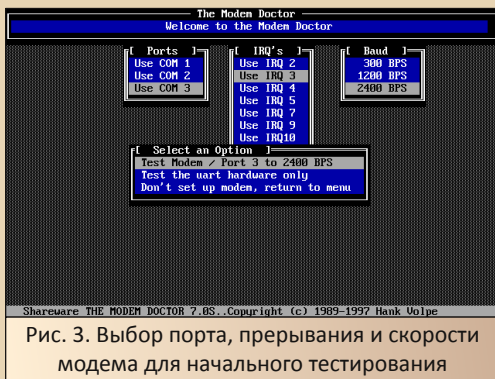


Рис. 3. Выбор порта, прерывания и скорости модема для начального тестирования

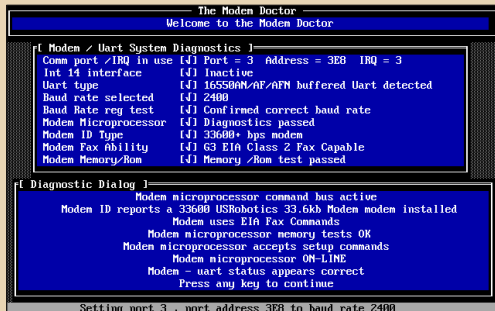


Рис. 4. Результаты общего тестирования модема – определена даже поддержка протокола V.34+ и скорости обмена 33600 бит/с

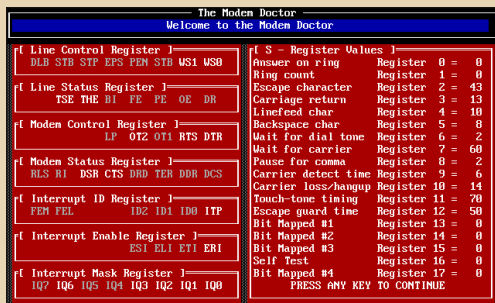


Рис. 5. Значения регистров модема – также выводятся программой автоматически

После нажатия любой клавиши все экраны с результатами тестов закрываются, и пользователь может самостоятельно средствами меню выбирать более подробные тесты. Собственно, именно этот экран и можно назвать главным «окном» программы (см. рис. 6).

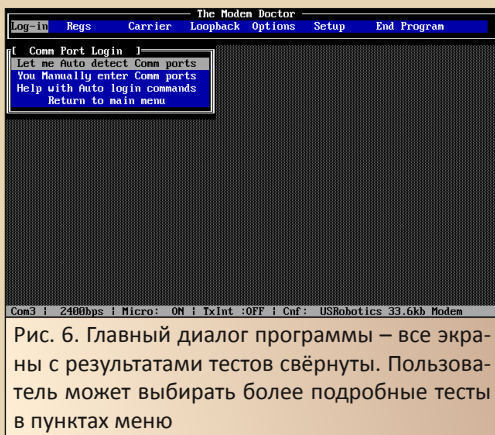


Рис. 6. Главный диалог программы – все экраны с результатами тестов свернуты. Пользователь может выбирать более подробные тесты в пунктах меню



В меню **Log-in** вы можете либо повторно запустить автоматическое определение портов, либо ввести порт самостоятельно с помощью диалогов программы.

В меню **Regs** пользователю доступны следующие тесты:

- **Quick UART register diagnostics** – выведет на экран ту же информацию, что и показана на рис. 5;
- **DTR/DSR RTS/CTS echo test** – видимо, тестируется связь линий DTR и DSR, а также RTS и CTS, используемых для аппаратного управления потоком данных;
- **DTR (RTS) continuous pulse test** – импульсы на линии DTR или RTS. Наверное, на внешнем модеме можно посмотреть мигание соответствующих светодиодов, увы, для внутреннего модема проверить это абсолютно невозможно;
- **Dynamic RTS/CTS (Xon-Xoff) handshake test** – видимо, должен также тестировать связь с модемом с аппаратным или программным управлением потоком данных, но в данной версии такая функция вообще недоступна – только для зарегистрированных пользователей;
- **Display any 10 S-registers** – выведет на экран значения любых десяти регистров модема. В диалоге пользователю необходимо только указать номер первого регистра (ну и, конечно же, знать, за что какой регистр отвечает – в диалоге будут выведены только числовые значения регистров – см. рис. 7).

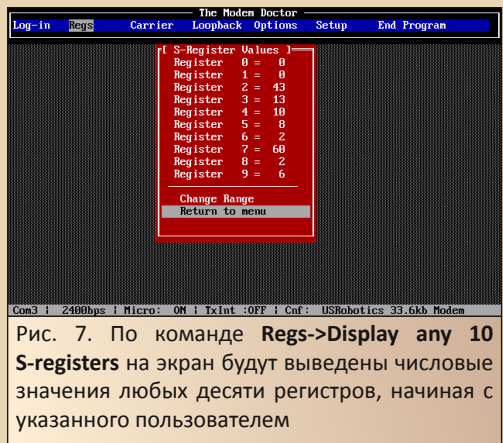


Рис. 7. По команде **Regs->Display any 10 S-registers** на экран будут выведены числовые значения любых десяти регистров, начиная с указанного пользователем

В разделе **Carrier**, такое ощущение, что собраны тесты для проверки аналоговой части модема. Модему отправляется пакет символов, и принимается эхо. По крайней мере, глядя на два окна терминалов, направивается именно такой вывод (см. рис. 8).

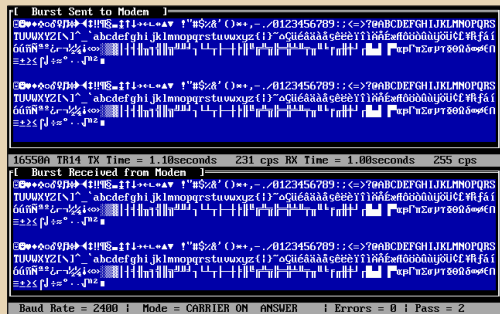


Рис. 8. Такое ощущение, что программа тестирует целостность передачи символов, а раз так была щепетильна к наличию сигнала от телефонной сети (с неподключённым телефонным проводом вообще работать отказалась), то скорее тестируется аналоговая часть модема

Кроме того, в разделе **Carrier** можно указать специальный драйвер модема.

В разделе **Loopback**, скорее всего, также собраны эхо-тесты, но они также не активны в данной версии.

В разделе **Options** собраны, можно сказать, прочие тесты и инструменты. Здесь присутствует терминал для работы с модемом (см. рис. 9), информация о факсимильных возможностях модема (см. рис. 10), статистика тестирования модема (см. рис. 11), а также инструменты сброса, сохранения и печати статистики.

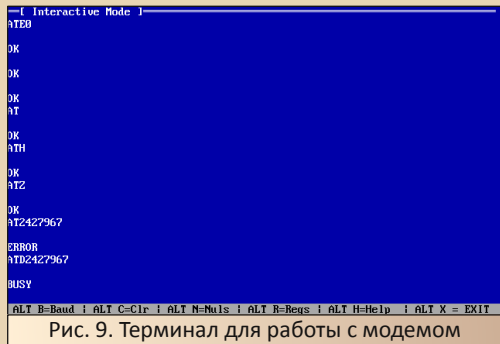


Рис. 9. Терминал для работы с модемом



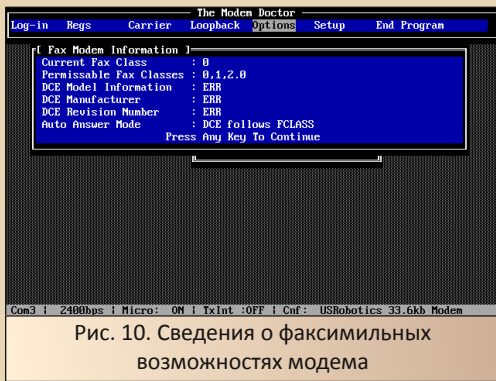


Рис. 10. Сведения о факсимильных возможностях модема

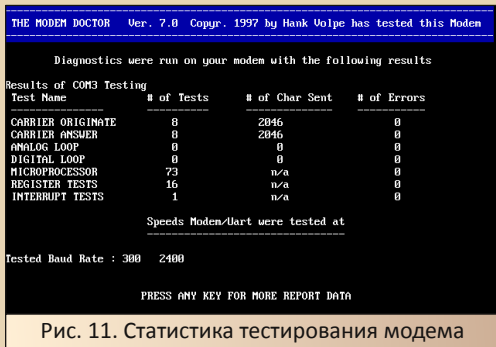


Рис. 11. Статистика тестирования модема

В меню **Setup** можно запустить либо выбор порта модема, как это делалось при запуске программы, либо настройку цветового оформления программы.

К сожалению, автору трудно оценить функциональность программы. С одной стороны, мы видим большое количество тестов, с помощью которых можно проверить и правильность распайки кабеля, и правильность реакции модема, а, куда важнее, определить на каком порту висит внутренний модем, и какое прерывание на него назначено, а заодно проверить работу модема терминалом. Но, когда автор активно пользовался телефонным модемом, как-то никогда не обращался к этой программе, как и к тонким настройкам модема. Может быть, и зря.

Modem Doctor для Windows 3.x

В «Новейшей энциклопедии программ» Леонтьева, к сожалению, описывалась не

Досовская, а винدوزная версия Modem Doctor'a, и описываемая в книжке версия предназначалась для запуска 32-разрядными версиями Windows, то бишь, начиная с Windows 95. Но была ли версия для 16-разрядных Windows или, хотя бы, для запуска под Win32s? «Скорее всего», – скажет читатель, посчитав, что в промежутке между версиями для DOS и Win32 должна вклиниться хоть одна версия для Windows 3.x, но, например, у Dr Web'a таких версий не было... Но веб-архив всё же помог распотать Modem Doctor for Windows 1.0, которая так запускается под третьей виндой. Скачать её также можно с Old-DOS:

<http://old-dos.ru/dl.php?id=20049>

Итак, скачиваем, распаковываем архив, устанавливаем программу и запускаем её значком в группе диспетчера программ. Как и версия программы для DOS, Modem doctor for Windows также выведет список с результатом сканирования последовательных портов (см. рис. 12). К сожалению, тут не получилось сразу увидеть в списке обнаруженный модем, зато программа указала, что COM1 занят мышью.

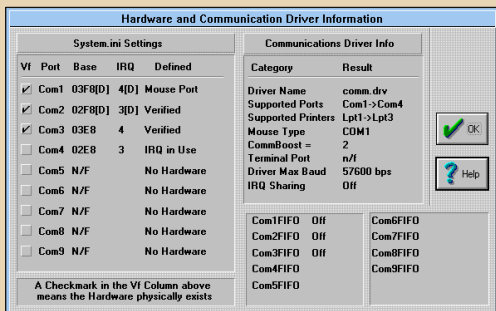


Рис. 12. Список с результатом сканирования последовательных портов – найдены порты COM1-COM3. COM4 – порта нет, но прерывание используется и, вместо модема, обнаружена мышь

После закрытия данного диалога нажать кнопки **OK**, откроется другой диалог – уже с настройками для тестирования модема. Тут программа скажет, что модем может оказаться подключённым к COM3 (см. рис. 13).



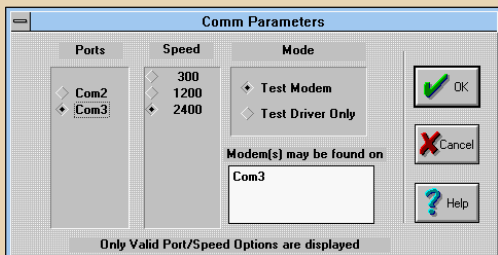


Рис. 13. А это уже явно настройки для тестирования модема

После выбора последовательного порта и скорости, на которой будет проводиться тестирование модема, открывается следующее окно, которое так же, как и в программе для DOS, проводит общее тестирование модема. И что же мы видим? Модем не отвечает, а мышь намертво зависла. Почему? Внимательный читатель, который внимательней автора, наверное, уже в настройках увидел, что порту COM3 программа присвоила прерывание IRQ4, не спросив пользователя. Что же делать?

Закрываем окно тестирования нажав кнопку **OK** – делаем сие с помощью клавиатуры, то же самое делаем и со следующим окном, а потом – также с помощью клавиатуры – командой меню **Login->Manual login** открываем меню ручной настройки последовательного порта (см. рис. 14).

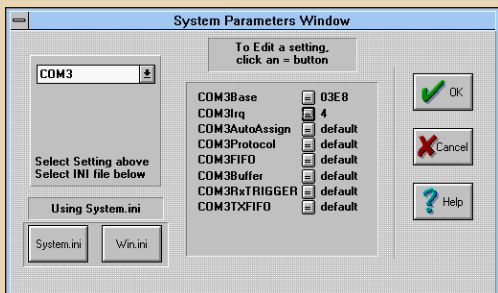


Рис. 14. Меню ручной настройки параметров последовательных портов

Здесь можно установить уже правильное значение прерывания. Выбираем нужный порт, выбираем необходимую кнопку, нажимаем

Enter и в открывшемся окне вводим новое значение прерывания (см. рис. 15).

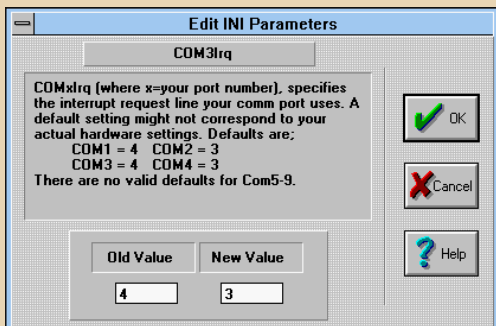


Рис. 15. Установка нового значения прерывания

После сохранения настройки, программа пропишет новое значение в системные файлы Windows. Увы, у автора мышь всё равно не отвисла, и пришлось перезапустить Windows. Зато после нехитрой процедуры и модем начал откликаться на команды, и мышь уже не зависает.

Вот такое окно с результатами предварительного тестирования выдаёт программа:

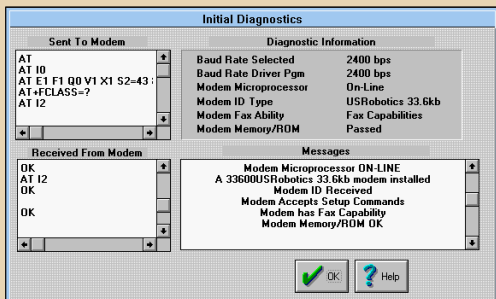


Рис. 16. Результаты предварительного тестирования. Как видно, в диалоге показаны не только результаты тестирования, но и команды, передаваемые модему и ответы на них

После нажатия кнопки **OK**, откроется диалог состояний линий последовательного порта (см. рис. 17).



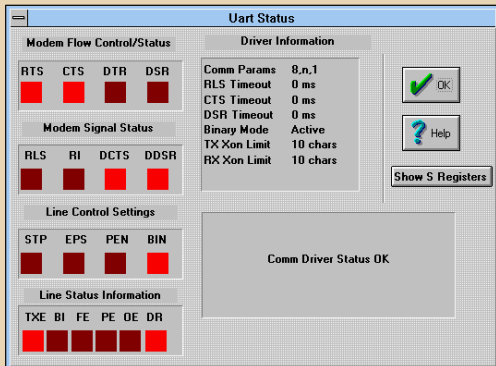


Рис. 17. Диалог состояний линий последовательного порта

После его закрытия, можно приступать к запуску тестов из меню.

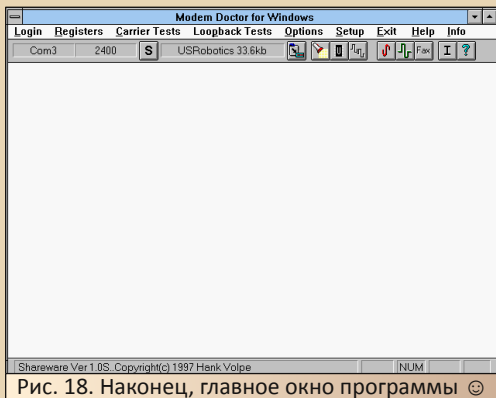


Рис. 18. Наконец, главное окно программы ☺

В целом, меню версии программы для Windows достаточно сильно напоминает меню ДОСовской версии. Разве что меню упрощены.

Меню **Login** не претерпело изменений. В меню **Registers** остались только пункты:

- **Quick UART registers tests** – выводит диалог состояния линий последовательного порта, как на рис. 17;
- **RTS/CTS DTR/DSR Echo;**
- **Dynamic RTS/CTS (Xon/Xoff) handshake.**

Возможно, программа стала более «интеллектуальной» и каким-то образом обнаружив, что имеет дело не с внешним модемом, убрала ненужные пункты, но верится в это с трудом. ☹

Вместо меню **Carrier** у программы для Windows присутствует меню **Carrier tests**. Там

все тесты собраны в один пункт **Carrier test**. А также есть пункт выбора драйверов модема. Что ж, выбираем пункт **Carrier test**. В открывшемся диалоге мы можем выбрать, какого типа данные будут пересылаться – символы или двоичные числа, после – по нажатию кнопки **OK** – откроется окно теста (см. рис. 19).

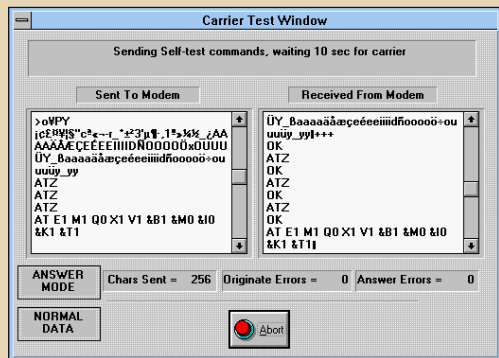


Рис. 19. Идёт тест, видимо, аналоговой части. В отличие от программы для DOS, динамик модема при тесте начинает орать при передаче данных. Возможно, кого-то раздражает, а автора, наоборот, порадовало ☺

Эхо-тесты из раздела **Loopback** также доступны только для зарегистрированных пользователей.

В разделе **Options** так и остались пункты вызова терминала (см. рис. 20), информации о факсимильных возможностях модема (см. рис. 21), статистика (см. рис. 22) и добавился отчёт об аппаратном обеспечении (см. рис. 23).

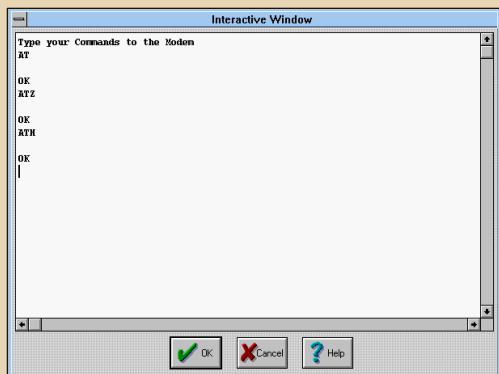


Рис. 20. Терминальное окно Modem Doctor for Windows



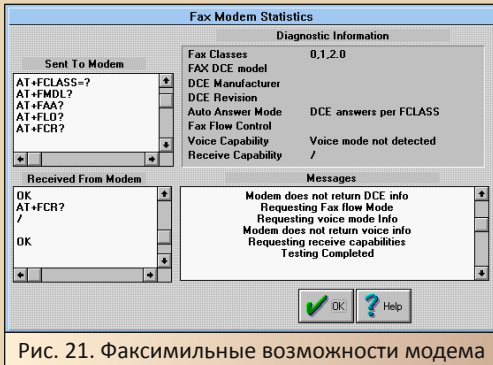


Рис. 21. Факсимильные возможности модема

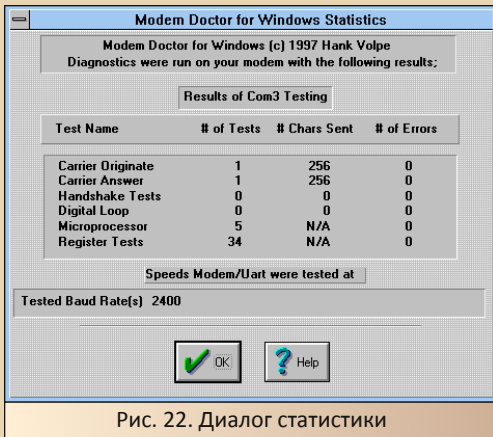


Рис. 22. Диалог статистики

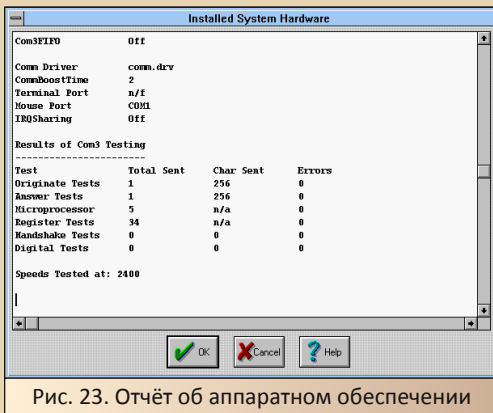


Рис. 23. Отчёт об аппаратном обеспечении

Кроме всего прочего, в главном окне программы ряд функций продублирован кнопками:

- настройка параметров модема (см. рис. 13);

- выбор драйвера модема;
- автоматическое определение аппаратных средств (см. рис. 12);
- отображение состояния линий последовательного порта (см. рис. 17);
- эхо-тестирование линий RTS/CTS и DTR/DSR (см. меню **Registers**);
- эхо-тестирование аналоговой части (**Carrier test**);
- тестирование кабеля (только для зарегистрированных пользователей);
- сведения о факсимильных возможностях модема (см. рис. 21);
- терминал (см. рис. 20);
- а также Вызов справки.

Как и с любой тестовой программой для Windows 3.x – трудно определить её полезность на фоне имеющихся тестов для DOS и Win32, но само наличие симпатичного теста – лично для автора – всегда радость.

Андрей Шаронов (Andrei88)



РАЗРАБОТКА ПРОГРАММ ДЛЯ ТЕКСТОВОГО РЕДАКТОРА NewsMaster («Журналист») Часть 2 из 3: ДРАЙВЕР ДЛЯ ПРИНТЕРА

В прошлой статье (см. Downgrade N27) мы разобрали создание драйвера для клавиатуры, чтобы позволить «Журналисту» корректно печатать русские буквы на QWERTY-раскладке. Теперь же давайте рассмотрим создание своего драйвера для принтера, чтобы иметь возможность печатать документы из программы в BMP-файлы на диске. Помимо теоретического, у такого драйвера есть и вполне практическое применение: официальная версия DOSBox не работает с принтером, да и не у каждого он есть, а посмотреть результат работы на широком экране, вместо мелкого режима предварительного просмотра в «Журналисте», тоже хочется. Это не говоря уже о том, что таким образом можно собрать в электронном виде набранную там аутентичную статью или даже целый номер журнала. Здесь же стоит заметить, что создание драйвера и печать на диск наименее трудозатратный способ из возможных — весь объём работ по формированию изображения на странице будет проделан самим «Журналистом», так что нам не придётся разбирать форматы файлов статей, шрифтов и т.д.



Специально для
Downgrade
N27 2019

<http://www.rusnet.ru/>

SysTools

<http://www.systools.com/>

Редактор NewsMaster, разработанный фирмой UNISON WORLD и выпущенный в 1986 году (первая версия 1.0), является тактовым процессором по типу *MicroSoft Word*. Программа не только позволяет использовать разнообразные шрифты, но и делать разметку страницы на колонки и столбцы, а также вставить рисунки с использованием гибких объектов текстов на поставляющемся набором библиотечного растрового клипарта. И, конечно же, проанализировать вышедший документ на печать — для этого через копипейст вставить драйвером была предусмотрена поддержка весьма солидного количества принтеров того времени. Сам же редактор работает в черно-белом графическом режиме *OBJ* с разрешением *640x200* пикселей.

В конце 80-х — начале 90-х редактор версии 1.0 попал в СССР и был переведен на русский язык отвлеченно 280 (валитация И Ю) — к сожалению, кроме этих скучных выводов с титульного экрана, про перевод или тех, кто над ним работал, большого ничего неизвестно. При переводе NewsMaster стал «Журналистом» — именно под этим именем он и стал известен широкому отечественному пользователю. Среди использованных приемов редактора на время его выхода стоит отметить копипейст, печать рисунков, но отсутствующий текст на документах форматированной — именно русский вариант «Журналиста» с помощью, но отсутствующий отключившийся русский шрифтами, но раз приколится на помощь и вырывает тотальная ситуация.

Посмотреть и загрузить программу можно с сайта *Old-Dos*:
<http://old-dos.ru/431.html>
Несмотря на свой солидный возраст, «Журналист» чувствует себя отлично не только под последними версиями DOS или DOSBox, но даже под XP/Vista и Windows XP. Однако именно из недостатков русского перевода было то, что русские шрифты были рассчитаны на раскладку *YUKEN*, а не *QWERTY*, в результате чего русские буквы не совпадали с теми, что нанесены на клавиши клавиатуры. Вероятно, и сами авторы делали перевод под *YUKEN*-клавиатуру.

Давайте попробуем эту самую задачу устранить. Замечание, текст ниже предполагает, что читатель на минимальном уровне знаком с основами программирования *Assemblers*, а также работой программы и сегментной организацией памяти в DOS, потому что эти вещи выйдут за рамки данной статьи. Более подробные сведения можно получить по ссылке, а также показано, как и почему для использования были выбраны именно они.

Пример печати в BMP-файл (начало прошлой статьи)

Начнём, опять же, с изучения входных данных и граничных условий.

Как уже было сказано в предыдущей статье, с программой поставлялся большой комплект драйверов для принтеров, пополнявшихся с каждой новой версией. При этом печать осуществлялась через работу с прерыванием **017h**, которое, как и в случае с клавиатурным драйвером, можно было бы перехватить, а уже в коде перехватчика решать, что делать с отправленными на печать данными. Однако такой подход создаёт целый ряд проблем, таких, например, как:

- код перехватчика придётся внедрять либо в уже существующий загрузчик для клавиатуры (неэффективно с точки зрения использования драйвера для других программ как универсального), либо запускать два загрузчика подряд (неэффективно по памяти), либо выходить из программы и запускать её снова через загрузчик для принтера, когда нужно что-то напечатать (просто неудобно);

- т.к. данные, отправляемые прерыванию **017h**, предваряются ESC/P-последовательностями (о которых мы поговорим чуть позже), да ещё и отсылаются по одному символу-байту, то в драйвере придётся обрабатывать текущее состояние работы всевозможных режимов (или сохранять всё в файл, затем писать ещё одну отдельную программу, которая будет разбираться, что со всем этим делать);

- одни и те же ESC/P-последовательности могут различаться у разных моделей принтеров как по формату (количеству пересылаемых байт в одной последовательности), так и по способу действия, а т.к. ничто не запрещает пользователю выбрать любой драйвер принтера из доступных в программе, то создание универсального перехватчика, рассчитанного на произвольную модель принтера, окончательно превращается в практически невыполнимую задачу.

Отсюда становится ясно, что будет лучше, а главное во много раз проще, создать свой



драйвер для принтера и сразу получать на входе ещё не обработанные данные для печати, работать с которыми нам будет гораздо удобнее и легче.

Нужно заметить, что не существует каких-то стандартов для написания драйверов к принтерам в программах под DOS – у каждой такой программы, использующей драйвера, будет свой формат, и NewsMaster здесь не исключение. Поэтому нам придётся дизассемблировать какой-то из уже существующих драйверов, разобраться с тем, как он работает, и на его основе сделать свой. В качестве такого драйвера удобнее всего будет взять файл **EPSON-FX.PRN** по целому ряду причин:

- поставляется с самой первой версии NewsMaster в неизменном виде (т.е. совместим со всеми версиями программы – 1.0, 1.5 и II);
- практически самый маленький из всех по размеру – всего 361 байт (чем меньше драйвер, тем он проще для разбора и изучения);
- этот драйвер для принтеров Epson, а фирма Epson для принтеров являлась в то время флагманом в технологии, на который все ровнялись.

Поэтому давайте дизассемблируем этот драйвер и посмотрим, как же он устроен.

В силу того, что драйвер достаточно простой, а его дизассемблирование выходит за рамки данной статьи, то код уже готового дизассемблированного драйвера будет добавлен в приложение к статье. Этот драйвер компилируется тоже в 361 байт и полностью повторяет на уровне команд работу оригинального, но будет различаться в 13 байтах из-за того, что Flat Assembler слегка иначе записывает некоторые команды, в отличие от того компилятора, который использовался разработчиками NewsMaster. Так, например, команда «xor al, al» в оригинальном драйвере записывалась как байты 032h 0C0h, в то время как Flat Assembler записывает как 030h 0C0h. На работу эти различия никак не влияют.

```
; FASM source code
; http://flatassembler.net/
; epson_fx.asm
```

```
; (c) SysTools 2019
; http://systools.losthost.org/
;
; NewsMaster / PrintMaster Epson *X
; compatible printer driver source codes.
; Source codes restored by disassembling
; original driver file called "EPSON-
; FX.PRN".
; This source code has no optimization of
; any kind because it sole purpose
; is to be instruction-exact to the
; original printer driver file.
; Not byte-exact since there are
; difference in 13 bytes with the
; original file
; due to fact that some assembler
; instruction can be written in a
; different ways,
; for example command "xor al, al" can be
; 32 C0 (original file) or 30 C0 (FASM).

; no "org 0100h" here - it's an overlay
; file
use16

; these jumps MUST be short jumps
; (2 bytes)
jmp @LnPitch8 ; 1
jmp @LnPitch6 ; 2
jmp @PrintRow ; 3
jmp @ResetAll ; 4
jmp @SkipPage ; 5
jmp @Skipline ; 6

@LnPitch8:
lea si, [_SeqSpacing8]
mov cx, 4d
nop
call @PrintSeq
retf

@LnPitch6:
lea si, [_SeqSpacing6]
mov cx, 4d
nop
call @PrintSeq
retf

; input:
; es:si - image buffer address
; cx - size for image buffer in bytes
; al - non-zero if printing 0x0A at the
; end required after 0x0D
@PrintRow:
mov [_BufSeg], es
mov [_BufOfs], si
mov [_BufLen], cx
mov [_PrnFlg], al
```



```

mov di, si
add di, cx
dec di
; al = 0
xor al, al
std
; find non-al byte at es:di
repe scasb
; if line empty - skip it
jz @f
; print only non-empty data
; (trim empty tail)
inc cx
mov word [_Seq960LCols + 2], cx
call @PrintHead
jnz @prnt_data_ret
mov si, [_BufOfs]
mov cx, word [_Seq960LCols + 2]
call @PrintBuf
jnz @prnt_data_ret
@@:
call @PrintLine
jnz @prnt_data_ret
call @CheckKey
or ax, ax
@prnt_data_ret:
retf

@ResetAll:
lea si, [_SeqPrnReset]
mov cx, 7d
nop
call @PrintSeq
retf

@SkipPage:
mov al, 00Ch
call @PrintChr
retf

@SkipLine:
mov al, 00Ah
call @PrintChr
retf

@PrintSeq:
cld
@@:
lodsb
call @PrintChr
loope @b
retn

@PrintBuf:
cld
@@:
lods byte [es:si]

call @PrintChr
loope @b
retn

@PrintChr:
push dx
xor dx, dx
xor ah, ah
int 017h
pop dx
test ah, 029h
retn

@PrintLine:
mov al, 00Dh
call @PrintChr
jnz @f
test [_PrnFlg], 0FFh
nop
jz @f
mov al, 00Ah
call @PrintChr
@@:
retn

@PrintHead:
lea si, [_Seq960LCols]
mov cx, 4
nop
call @PrintSeq
retn

@CheckKey:
push dx
mov ah, 001h
int 016h
jz @f
xor ah, ah
int 016h
cmp al, 01Bh
jz @CheckKey_ret
call @PrintPause
xor ah, ah
int 016h
push ax
call @PrintClear
pop ax
cmp al, 01Bh
jz @CheckKey_ret
@@:
xor ax, ax
pop dx
retn
@CheckKey_ret:
mov ax, 1
pop dx
retn

```




```
@PrintPause:
push bx
mov dx, 01814h
xor bh, bh
mov ah, 002h
int 010h
lea dx, [_StrWaitText]
mov ah, 009h
int 021h
pop bx
retn
```

```
@PrintClear:
push bx
mov dx, 01814h
xor bh, bh
mov ah, 002h
int 010h
lea dx, [_StrWaitFill]
mov ah, 009h
int 021h
pop bx
retn
```

```
_SeqSpacing8 db 01Bh, 041h, 008h, 00Dh
_SeqSpacing6 db 01Bh, 041h, 006h, 00Dh
_Seq960LCols db 01Bh, 04Ch, 0C0h, 003h
_SeqPrnReset db 01Bh, 040h, 01Bh, 039h,
01Bh, 04Fh, 00Dh
_StrWaitText db 'Pausing... Press a key
to continue.$'
_StrWaitFill db ' $'
_BufSeg dw 0
_BufOfs dw 0
_BufLen dw 0
_PrnFlg db 0
```

Просмотрев код драйвера, можно заметить, что начинается он с 6-ти коротких **jmp** (это важно, они должны быть именно короткими – по два байта) на следующие участки кода:

```
LnPitch8
LnPitch6
PrintRow
ResetAll
SkipPage
Skipline
```

Чтобы понять, за что же именно отвечает каждая из этих подпрограмм, нам придётся заглянуть в справочник фирмы Epson по ESC/P-последовательностям. И здесь хочется ещё раз напомнить о том, что попытка разобрать любой другой драйвер (а есть и короче, пусть и всего на пару байт) окажется неуспешной из-за того, что ESC/P-последовательности, использу-

ющиеся там, будут несовместимы со стандартом Epson, а документацию на малоизвестные принтеры того времени найти сейчас практически нереально.

Но сначала стоит немного рассказать о самих ESC/P-последовательностях.

ESC/P (Epson Standard Code for Printers) – язык команд, разработанный фирмой Epson, для управления принтерами. Он используется преимущественно в матричных и некоторых струйных принтерах. ESC/P является фактическим стандартом для матричных принтеров и используется другими производителями, иногда в несколько расширенном или, наоборот, сокращённом виде.

(c) Wikipedia <https://ru.wikipedia.org/wiki/ESC/P>

При этом английская Wikipedia справедливо добавляет, что ESC/P гораздо реже встречается на современных принтерах, потому что был разработан в первую очередь именно для матричных.

Итак, каждая последовательность в этом языке команд начинается с символа **ESC – 01Bh** (десятичное 27), затем следует код команды и параметры (если есть).

Вооружённые этим знанием, загружаем по ссылке из английской Wikipedia справочное руководство:

– за 1997 год: <https://files.support.epson.com/pdf/general/escp2ref.pdf>

– или за 2004: <http://www.epson.ru/upload/iblock/057/esc-p.pdf>

Вот теперь можно предметно рассмотреть оригинальный драйвер для принтера Epson от разработчиков NewsMaster.

Подпрограмма по адресу **LnPitch8** отправляет на печать последовательность **01Bh, 041h, 008h, 00Dh (ESC A)**. Ищем её по первым двум байтам в справочнике Epson и из описания узнаём, что эта команда устанавливает межстрочный интервал в n/60 от дюйма (1 дюйм = 2.54 см). Т.е. данная последовательность устанавливает этот интервал в 8/60 дюйма. Последний символ (**00Dh – возврат каретки**), вероятно, нужен на тот случай, если принтер сдвинул головку при обработке этой последо-



вательности, так что далее не будем обращать на него внимание.

Подпрограмма по адресу **LnPitch6** делает практически то же самое, но устанавливает интервал в 6/60 дюйма.

Подпрограмма **PrintRow** непосредственно выводит данные на печать. На вход она получает три параметра в регистрах:

ES:SI – указатель на начало буфера с данными для печати;

CX – количество байт в буфере;

AL – параметр, отвечающий за необходимость посылать (если **AL** не ноль) дополнительно символ **00Ah** (перевод строки) в дополнение к **00Dh** (возврат каретки) в конце напечатанной строки, если принтер не делает этого автоматически; этот параметр настраивается через текстовый конфигурационный файл **NEWS.CFG** (**LineFeed 0** или **LineFeed 1**) или программу выбора и настройки принтера **NMCONFIG.EXE** (последний вопрос: «**Does your printer automatically print a linefeed when it prints a carriage return?**»).

Does your printer automatically print a linefeed when it prints a carriage return?

1. Printer prints linefeed automatically
2. Printer does NOT print linefeed automatically

Press ESC to exit without changing the configuration_

Настройка принтера для NewsMaster 1.0 в **NMCONFIG.EXE**

Подпрограмма по адресу **ResetAll** посылает на печать целую пачку **ESC/P**-последовательностей (разбиты на отдельные строки для удобства):

01Bh, 040h – **ESC @** – сброс настроек принтера к значениям по умолчанию.

01Bh, 039h – **ESC 9** – включить датчик отсутствия бумаги.

01Bh, 04Fh – **ESC O** – отменяет настройки верхнего и нижнего полей.

Подпрограмма по адресу **SkipPage** отправляет на печать всего один символ – **00Ch** (конец страницы), который прогоняет бумагу до следующей страницы (это тоже, кстати, описано в документации Epson).

Наконец, подпрограмма по адресу **SkipLine** тоже отправляет на печать один символ – **00Ah** (перевод строки), который пропускает текущую и переходит на новую строку (также отражено в документации Epson).

Перед вызовом любой из этих подпрограмм регистр **DS** устанавливается равным значению сегмента, в который загружен драйвер, т.е. адрес **DS:0** будет указывать на первый из тех самых 6 коротких **jmp**, что очень удобно при работе с данными, т.к. регистр не придётся инициализировать вручную.

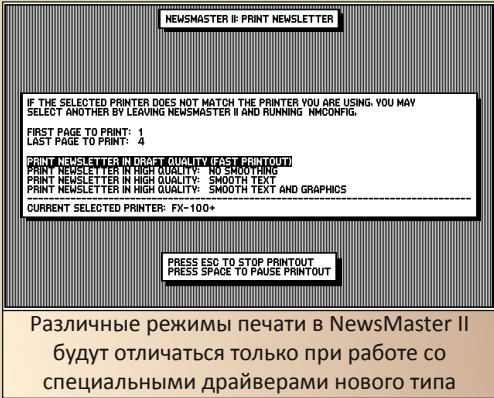
Выход из упомянутых подпрограмм происходит через дальний возврат – **retf**, при этом если **zero flag** установлен, то печать продолжается, а иначе прекращается (когда флаг сброшен, это сигнализирует об ошибке).

Все регистры можно изменять без опасений, как это делает оригинальный драйвер, потому что NewsMaster самостоятельно заботится об их сохранности при вызове подпрограмм.

Важный момент: файл драйвера будет загружен через функцию DOS прерывания **021h** под номером **04B03h** – **Load Overlay** – в заранее выделенный статический буфер в программе. Размер этого буфера очень небольшой, всего 1200 байт в NewsMaster версий 1.0 и 1.5, поэтому и драйвер не должен превышать этого размера, в противном случае последствия будут катастрофическими, вплоть до фатальных ошибок при запуске программы. В NewsMaster II этот буфер был увеличен примерно в 5 раз, но универсальный драйвер, как поддерживающий все версии упомянутый выше **EPSON-FX.PRN**, должен уместиться в минимальный размер. Для com-формата этот размер будет равен размеру файла на диске (под все статические буферы место обязательно должно быть распределено заранее, чтобы учитываться в размере файла). Но вот



exe-формат может занимать немного больше за счёт заголовка MZ и таблицы размещения, однако, загруженный в память, опять же, не должен превышать указанного размера. Здесь же стоит добавить, что функция Load Overlay не подготавливает PSP, не инициализирует регистры и не делает многое другое, что происходит при обычном запуске программы. Поэтому нельзя полагаться и рассчитывать на эти вещи ¹.



Теперь неплохо будет узнать, в каком порядке и как вызываются все описанные выше подпрограммы. Для этого сделаем коротенький отладочный драйвер, записывающий в файл имена функций, которые были вызваны, и параметры для **PrintRow**.

```
; FASM source code
; http://flatassembler.net/
;
; logdmpnr.asm
; (c) SysTools 2019
; http://systools.losthost.org/
;
; NewsMaster / PrintMaster virtual
; printer driver logs dumper
; This printer driver tested and
; compatible with NewsMaster version 1.0,
; 1.5 and II.
; Set up any printer driver in
; NMCONFIG.EXE and replace NEWS.PRN file
; in the program
; directory with compiled binary file of
; this virtual printer driver.
; LOGDMPRN.BIN => NEWS.PRN
```

```
; WARNING!
; Printer driver will be loaded via DOS
; Fn 04B03H (Load Overlay) to tiny static
; buffer.
; This buffer size only 1200 bytes long
; in NewsMaster 1.0.
; Because of that the driver must be as
; small as possible.
; For example:
; 1. Largest driver CGP220.PRN size is
; 1192 bytes and it's safe to use.
; 2. Driver in 1384 bytes long are
; already too big and will cause the
; program to crash.
; 3. Note that size of the driver
; IBMCMPTC.PRN is 1664 bytes (bigger than
; 1200) but it's an .EXE file and amount
; of bytes it takes in the memory will be
; less than the file size.
```

```
; no "org 0100h" here - it's an overlay
; file
use16
```

```
; these jumps MUST be short jumps (2 bytes)
jmp @LnPitch8 ; 1
jmp @LnPitch6 ; 2
jmp @PrintRow ; 3
jmp @ResetAll ; 4
jmp @SkipPage ; 5
jmp @Skipline ; 6
```

```
; call order for each page:
; [4] ResetAll
; [1] LnPitch8
; [3] PrintRow (for each row)
; [5] SkipPage
```

; and here long calls can be used

```
@LnPitch8:
mov dx, _LnPitch8
mov cx, 10d
call @DumpLogs
xor ax, ax
retf
```

```
@LnPitch6:
mov dx, _LnPitch6
mov cx, 10d
call @DumpLogs
xor ax, ax
retf
```

```
@PrintRow:
; save al for later usage
```

¹ См. подробности, например, в TECH Help! «DOS Fn 4b03H: Load Overlay».



```

push ax
; print cx (data size)
mov ax, cx
mov cx, 4d
mov di, _PrintRow + 9d
call @ToHex
; print al (line flag)
pop ax
mov cx, 2d
mov di, _PrintRow + 14d
call @ToHex
; print current row
inc [_rows]
mov ax, [_rows]
mov cx, 4d
mov di, _PrintRow + 18d
call @ToHex
; and now output
mov dx, _PrintRow
mov cx, 25d
call @DumpLogs
xor ax, ax
retf

```

```

@ResetAll:
mov dx, _ResetAll
mov cx, 10d
call @DumpLogs
xor ax, ax
; reset rows count
mov [_rows], ax
retf

```

```

@SkipPage:
mov dx, _SkipPage
mov cx, 10d
call @DumpLogs
xor ax, ax
retf

```

```

@SkipLine:
mov dx, _SkipLine
mov cx, 10d
call @DumpLogs
xor ax, ax
retf

```

; -----

```

@DumpLogs:
; save for later usage
push dx
push cx
; try to open file
mov dx, _name
mov al, 002h
mov ah, 03Dh
int 021h

```

```

jnc @f
; failed to open - try to create
mov dx, _name
xor cx, cx
mov ah, 03Ch
int 021h
; failed to create - exit
jc @DumpLogs_ret
@@:
; save file handle
push ax
; seek to file end
mov bx, ax
xor cx, cx
xor dx, dx
mov al, 002h ; from file end
mov ah, 042h
int 021h
; append data to file
pop bx
; saved before
pop cx
pop dx
push dx
push cx
push bx
mov ah, 040h
int 021h
; close file
pop bx
mov ah, 03Eh
int 021h
@DumpLogs_ret:
pop cx
pop dx
retn

```

; -----

```

@ToHex:
test cx, cx
jz @ToHex_ret
add di, cx
@conv:
dec di
mov dl, al
shr ax, 004h
and dl, 00Fh
cmp dl, 9d
jbe @f
add dl, 'A' - '0' - 10d
@@:
add dl, '0'
mov [di], dl
loop @conv
@ToHex_ret:
retn

```



```

; -----
; data area
_rows dw 0
_name db 'LOGSDUMP.LOG',0
_LnPitch8 db 'LnPitch8',13,10
_LnPitch6 db 'LnPitch6',13,10
_PrintRow db 'PrintRow ????'',13,10
_ResetAll db 'ResetAll',13,10
_SkipPage db 'SkipPage',13,10
_SkipLine db 'SkipLine',13,10

```

Установка этого драйвера, как и будущего драйвера для виртуальной печати, очень простая – достаточно заменить файл **NEWS.PRN** в корне каталога программы. На всякий случай, если программа ещё ни разу не настраивалась, то нужно вызвать **NMCONFIG.EXE**, выбрать там любой принтер, а уже затем заменить **NEWS.PRN** новым драйвером.

Итак, после компиляции отладочного драйвера, установки и печати в NewsMaster 1.0, получим файл **LOGSDUMP.LOG** примерно такого содержания:

```

ResetAll
LnPitch8
PrintRow 03C0 01 (0001)
PrintRow 03C0 01 (0002)
<пропущены строки с 0003 до 005E>
PrintRow 03C0 01 (005F)
SkipPage

```

Отсюда видно, что вызов происходит в следующем порядке:

```

ResetAll
LnPitch8
PrintRow (для каждой строки)
SkipPage

```

И всё это повторяется для каждой страницы в документе.

Интересно, что подпрограммы **LnPitch6** и **SkipLine** не вызываются никогда. Здесь нужно заметить, что рассматриваемый драйвер для Epson поставляется со всеми версиями News-

Master, которые удалось найти: 1.0, 1.5 и II, но, возможно, эти функции «остались в наследство» со времён раннего этапа разработки или используются в PrintMaster (схожая программа от тех же разработчиков).

Нам же для записи печати в файл необходимо:

- создать файл (открыть на запись);
- записывать в него отправляемые на печать данные;
- и закрыть по завершении печати.

Для этого как нельзя лучше подходят подпрограммы **ResetAll**, **PrintRow** и **SkipPage**, а подпрограмму **LnPitch8** лучше опустить как малозначимую. Потому что при подготовке к печати для инициализации принтера обязательно будет вызываться **ResetAll**, при отправке данных на печать **PrintRow**, а при завершении **SkipPage** (справочник Epson прямо рекомендует прогонять страницу после окончания печати). А все остальные подпрограммы могут оказаться опциональными, поэтому их лучше избегать.

Следующая важная вещь, на которую стоит обратить внимание при использовании отладочного драйвера, это то, что в NewsMaster II подпрограмма **PrintRow** вызывается не 95 (05Fh) раз, а уже 126 (07Eh) раз – т.е. высота страницы была увеличена, и в разных версиях программы она разная. Мы к этому вернёмся чуть позже, а пока что, давайте разберём подпрограмму **PrintRow** из оригинального драйвера более подробно.

Сперва с конца входного буфера ищется хотя бы один не нулевой байт². Если все байты нули, то строка считается пустой, никакой графики не печатается, а принтер всего лишь переходит на следующую строку. Это не только позволяет более бережно относиться к технике, а не гонять зазря головку принтера туда-сюда, но и увеличивает скорость печати. А вот

² Один из немногих моментов, когда заглядывать в справку TECH Help! вредно, потому что в описании команд языка Assembler для работы со строками, к сожалению, допущены ошибки даже в последней, шестой, версии справочника – в частности у **scasb** указаны регистры **DS:DI**, в то время как из кода драйвера видно, что работа идёт с **ES:DI**. Справочник хороший, но содержит ошибки – будьте осторожны.



полученное таким образом, но не нулевое, количество байт отправляется на печать, предрекаясь следующей ESC/P-последовательность:

01Bh, 04Ch, 0C0h, 003h – ESC L – печать строки графики.

Где последние два байта будут перезаписаны драйвером на необходимое количество вместо 960 (003C0h). Здесь также стоит обратить внимание, что и регистр **CX**, содержащий размер буфера, всегда равен 960 на входе в подпрограмму (что видно из логов работы отладочного драйвера).

Если сейчас начать писать данные в файл и попытаться просмотреть их в какой-нибудь программе (например, через [Hexapad](#) – **View->Sprite Search, width: 960, format: 1bpp**), то вместо изображения там будет маловразумительная каша.

Попробуем разобраться, почему же так происходит. Размер рисунка, действительно, 960 точек, но т.к. это чёрно-белый режим, следовательно, одна точка должна кодироваться 1 битом, а в байте их 8. Отсюда: $960/8 = 120$. Но размер буфера в байтах, как мы помним, в 8 раз больше этого – 960. Как же так? Дело в том, что в матричных принтерах иглолки для печати на головке принтера расположены не горизонтально, а вертикально. Поэтому один байт кодирует не 8 точек по горизонтали, как это обычно принято в графических форматах, а 8 точек по вертикали. Таким образом, головка принтера печатает не одну строку в 960 точек, а сразу целую полосу из 8 таких строк³. Поэтому посылаемые на печать данные придётся переставлять перед сохранением в другом порядке, чтобы получить осмысленное изображение для графического формата.

Обладая этими знаниями, мы можем легко подсчитать размер рисунка в пикселях – помните, чуть выше было показано, сколько раз вызывается подпрограмма **PrintRow**? Будет

960x760 (95*8) и 960x1008 (126*8) соответственно.

И давайте разберёмся, откуда взялась ширина в 960 точек. Странно, но этой величины в явном виде нет в справочниках Epson по ссылкам выше с официального сайта. В описании команды **ESC L** упоминается только про 120 на 60 dpi и всё. Так что здесь здорово спасает бумажная документация к принтерам – удивительно, но, например, руководство от фирмы HYUNDAI к матричным принтерам моделей HDP-910/920 (Epson-совместимые) за 1989 год описывает пусть и не все ESC/P-последовательности, но зато более детально. Так, про интересующую нас последовательность там указано, что в этом режиме можно напечатать 960 точек на 80-колоночных принтерах и 1872 на 156-колоночных. Если же взять русское руководство от самой Epson для LX-1050+ за 1994 год, то про **ESC L** там всё так же скудно, как и в PDF-документации, зато подробно описано вертикальное расположение иглолок и их программирование, в том числе всякие нюансы о том, что в этом режиме из 9 иглолок используются только верхние 8 (LX-1050+ – это 9-ти иглолочный принтер), а также зачем устанавливать межстрочный интервал в 8/60 дюйма перед выводом изображений (см. про **LnPitch8** выше) – оказывается, это убирает отступ между строками, чтобы печать графики шла неразрывно⁴.

Теоретически, подойти к решению этой проблемы можно было и с другой стороны: dpi – dots per inch, т.е. точек на дюйм. Бумага формата A4 согласно [Wikipedia](#) имеет ширину в 8.27 дюйма. Печать графики, как мы помним из документации Epson, идёт для ширины в 120 dpi. Отсюда $8 * 120 = 960$, хотя и не совсем очевидно, что дробную часть дюйма (под поля?) нужно было отбросить. Поэтому поиск точных числовых величин в документации – наиболее надёжный способ.

³ Что, вообще говоря, логично, т.к. увеличивает скорость печати (по сути 8 строк за один проход), но не совсем очевидно, особенно если никогда с матричными принтерами на низком уровне работать до этого не приходилось.

⁴ Т.е. установку интервала можно было бы поместить в **ResetAll**, но, видимо, в силу уже сложившейся архитектуры драйвера, всё оставили как есть.



Фото документации

Графика

Печатающая головка

Чтобы разобраться в точечной графике вам необходимо знать, как работает печатающая головка вашего принтера.

Печатающая головка принтера LX-1050+ имеет девять иглонок. При ее перемещении вдоль страницы электрические импульсы вызывают быстрое перемещение иглонок. Всякий раз, когда иглонка перемещается, она ударяет по ленте с краской и прижимает ее к бумаге, на которой остается маленькая точка. При движении головки вдоль бумаги иглонки раз за разом перемещаются, формируя буквы, цифры или символы.

Точечные рисунки

Печатающая головка принтера LX-1050+ может печатать графику и дополнение к текстам, потому что графическое изображение формируется принтером таким же способом, как печатаются иллюстрации в газетах и журналах. Если вы посмотрите внимательно на фотографию в газете, то вы увидите, что она состоит из множества маленьких точек. Принтер также формирует такие изображения рисунком из точек с плотностью 240 точек на дюйм по горизонтали и 72 точки по вертикали. Поэтому изображения, печатаемые принтером, могут иметь такие же мелкие детали, как и на рисунке на стр. 4-9.

В своем основном режиме графики принтер печатает одну колонку точек для получаемого кода и при этом используются только восемь первых иглонок из девяти. Поэтому ваша программа графики должна посылать коды для точечных рисунков по одному для каждой колонки в строке. В каждой из этих колонок печатающая головка печатает рисунок из точек, который вы определите.

Для печати рисунков выше, чем восемь точек, печатающая головка делает более одного прохода. Принтер печатает одну строку, затем перемещает бумагу и печатает следующую, также как он делает это с текстом.

Чтобы печатающая головка не делала пропусков между строками графики, как между строками текста, расстояние между строками должно изменяться до устранения пропуска между строками. При изменении расстояния между строками ваш принтер может печатать графические изображения с мелкими деталями так, что не будет видно, что они состоят из отдельных линий, каждая высотой не более 8/72 дюйма.

Графика

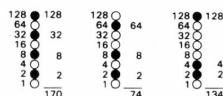
Маркировка иглонок

Чтобы сообщить принтеру, какие иглонки должны перемещаться в каждой колонке, вам необходимо сначала разделить каждую вертикальную колонку на три секции по восемь колонок в каждой и рассмотреть каждую секцию отдельно. Так как имеется 256 возможных комбинаций восьми иглонок, вам необходима система нумерации, которая показана ниже.



Для перемещения любой иглонки, вы посылаете ее номер. Для перемещения одновременно более одной иглонки, сложите номера иглонок и пошлите сумму принтеру. Поэтому при этой маркировке иглонок, вы перемещаете верхнюю, посылая число 128. Для перемещения нижней иглонки вы посылаете 1. Если вы хотите переместить только верхнюю и нижнюю иглонки, вы просто складываете 128 и 1 и посылаете 129.

Складывая соответствующие маркировочные номера вместе, вы можете перемещать любую комбинацию иглонок. Ниже показаны три примера вычисления чисел, которые будут создавать конкретный рисунок иглонок.



При этой системе нумерации любая комбинация сложения для восьми иглонок представляется без дублирования десятичным числом от 0 до 255. Однако, перед тем как вы заложите эти значения в графическую программу вам необходимо знать формат графических команд.

4-10 Программное обеспечение и графика

Программное обеспечение и графика 4-11

Bit image printing

ESC^L' n₁ n₂ Double-density Bit Image Mode

Function This code sequence is used to print user-designed graphics in the double-density bit image mode.

Code Hex. n₁<4C>n₂<n₁><n₂>
Dec. <1B>n₁<78>n₂<n₁><n₂> (0 ≤ n₁ ≤ 255, 0 ≤ n₂ ≤ 255)

BASIC format Hex. CHR\$(A+1B)";L";CHR\$(n₁);CHR\$(n₂);
Dec. CHR\$(127)";L";CHR\$(n₁);CHR\$(n₂);

Related data This code sequence switches the printer to the double-density 8-dot bit image printing mode. Following execution of this sequence, subsequent character codes are handled as bit image data for the number of bytes specified by n₁ and n₂. Here, n₁ and n₂ are the lower and upper bytes, respectively, of a 16-bit hexadecimal number (or its decimal equivalent) which indicates the number of character codes which are to be handled as bit image data. Both n₁ and n₂ must be in the range from <00> to <FF>.
With decimal numbers, n₁ and n₂ can be calculated for a given value n as follows, where n is the number of data.
n₁ = INT (n/256) (The integer portion of the quotient of n/256)
n₂ = INT (n/256) (The integer portion of the quotient of n/256)
In the double-density mode, the maximum number of dots of bit image data which can be printed on one line is 96(80-column model) and 187(156-column model), respectively. Adjacent data are horizontally spaced at intervals of 1/120 of an inch, and are vertically spaced at intervals of 1/72 of an inch.
However, if bit image printing reaches the right margin before the specified number of bytes of bit image data have been printed, data extending beyond the right margin will be ignored, and printing automatically returns to the text mode. Also, when the printer receives an LF code, the bit image mode is terminated and the text mode restored. It is recommended that bit image mode be set for every line if you are printing data on several lines.

See also ESC^K Single-density bit image mode p.122
ESC^Y Double-speed, double-density bit image mode p.126
ESC^Z Quick-double-density bit image mode p.127
ESC^* Select bit image mode p.129

Bit image printing

Example

```
[program]
100 'Double-density bit image mode ----- [ESC L...]
110 LPRINT CHR$(27)";m";
120 '
130 FOR N=1 TO 3
140 '----- set 60 dots/inch mode ----- ESC L ..
150 LPRINT CHR$(27)";L";CHR$(480 MOD 256);CHR$(INT(480/256));
160 GOSUB 220
170 LPRINT
180 NEXT N
190 LPRINT
200 END
210
220 FOR I=1 TO 60
230 LPRINT CHR$(81B);CHR$(800);CHR$(8H7E);CHR$(8H00);
240 LPRINT CHR$(8HFF);CHR$(8H00);CHR$(8H3C);CHR$(8H00);
250 NEXT I
260 RETURN
```



Теперь несколько слов и о формате BMP, выбранном для хранения изображений.

Вообще говоря, если писать аутентичный для того времени драйвер (следует напомнить, что NewsMaster 1.0 вышел в 1986 году), то сохранять изображения лучше всего, конечно, в какой-нибудь формат со сжатием, ибо диски тогда были достаточно скромного объёма, а без сжатия 3-4 сохранённых страницы-изображения будут занимать больше, чем весь NewsMaster версии 1.0.

Из существующих тогда форматов можно было бы взять PCX от фирмы ZSoft Corporation, благо, что и разработан он был за год до этого (в 1985). Да и сжатие RLE, которое используется в этом формате, конечно, не самое лучшее, но, во-первых, место экономит, а, во-вторых, достаточно простое в реализации.

Но всё же экономить мы не будем в силу следующих причин:

- чем драйвер проще, тем он легче для изучения и дальнейшего изменения;

- алгоритм RLE в PCX хоть и достаточно прост в реализации, однако нужно помнить про ограничение драйвера в 1200 байт – на добавление туда сжатия может просто не хватить места;

- к сожалению, былая слава и популярность формата PCX давно прошли, поэтому открыть и редактировать BMP на «голой системе» сейчас гораздо проще (например, работа с PCX не поддерживается в Windows штатными средствами).

Так что для сохранения изображений будем использовать формат BMP версии 3, потому что первая версия хоть и появилась в 1985, но именно третья версия получила самое широкое распространение и поддержку, хоть и вышла на 5 лет позже⁵.

Подробное описание формата BMP можно найти в Интернете, нам же, чтобы не загромождать статью, стоит уделить особое внимание трём важным вещам.

Во-первых, изображение в BMP файле хранится записанным снизу-вверх. Т.е. первая

прочитанная строка пикселей от начала смещения растровых данных – это не первая строка изображения, а последняя (нижняя), вторая прочитанная строка – предпоследняя снизу и т.д. Для представления же изображения в «нормальном» виде, т.е. сверху-вниз, высота должна быть указана отрицательной. Это отражено в документации на формат ещё с начала 90-х годов, но, к сожалению, даже сейчас изредка попадаются разработчики, не обращающие внимание на этот важный момент. Так что если сохранённый нами файл какая-то программа не сможет открыть, будет сообщать о неверном формате, упадёт с ошибкой или покажет вверх ногами – значит, её авторы невнимательно читали документацию. Поэтому в коде драйвера высота рисунка (как мы помним, она может быть разной) будет не увеличиваться, а уменьшаться на единицу с каждой записанной в файл строкой пикселей.

Во-вторых, каждая строка пикселей в BMP-формате, опять же, согласно документации, должна быть выровнена на 32-х битную границу. Это значит, что если число байт в строке пикселей не кратно 4 ($32/8 = 4$), то необходимо дописать от 1 до 3 байт. У нас же, как было упомянуто выше, $960/8 = 120$ байт, а 120 прекрасно делится на 4. Поэтому в данном (частном) случае выравнивание не потребуется, но упомянуть о нём всё же стоит.

Наконец, в-третьих, палитра по умолчанию в чёрно-белом BMP файле соответствует чёрно-белым режимам работы монитора, где 0 – чёрный, а 1 – белый пиксель. Но из разобранного выше кода драйвера видно, что в матричных принтерах наоборот: 0 – белый (пустое место – цвет бумажного листа), а 1 – чёрный (напечатать точку чернилами), поэтому придётся обращать (менять на противоположные) биты в байте, благо это не сложно и делается унарным оператором **not**⁶.

А вот теперь, зная, как устроена внутренняя архитектура драйверов для принтеров в NewsMaster и описание формата BMP, можно начать работу над своим драйвером.

⁵ Версия 1 появилась вместе с Windows 1.0 (1985), а версия 3 – вместе с Windows 3.0 (1990), согласно [этому сайту](#)

⁶ Можно поменять чёрный и белый цвета палитры местами в BMP-файле, но тогда некоторые программы при сохранении файла будут насильно менять цвета назад, и обращать биты в байтах (тот же Paint в Windows, впрочем, он и высоту делает положительной, но так хотя бы затронут будет только порядок строк). Здесь же, справедливости ради, стоит заметить, что в BMP тоже есть некое RLE-сжатие, однако его поддерживают ещё меньше программ, чем отрицательную высоту.




```

; FASM source code
; http://flatassembler.net/
; prntodsk.asm
; (c) SysTools 2019
; http://systools.losthost.org/
; NewsMaster / PrintMaster virtual
; printer driver
; This printer driver tested and
; compatible with NewsMaster version 1.0,
; 1.5 and II.
; Set up any printer driver in
; NMCONFIG.EXE and replace NEWS.PRN file
; in the program
; directory with compiled binary file of
; this virtual printer driver.
; PRNTODSK.BIN => NEWS.PRN
; WARNING!
; Printer driver will be loaded via DOS
; Fn 04B03h (Load Overlay) to tiny static
; buffer.
; This buffer size only 1200 bytes long
; in NewsMaster 1.0.
; Because of that the driver must be as
; small as possible.
; For example:
; 1. Largest driver CGP220.PRN size is
; 1192 bytes and it's safe to use.
; 2. Driver in 1384 bytes long are
; already too big and will cause the
; program to crash.
; 3. Note that size of the driver
; IBMCOMPCT.PRN is 1664 bytes (bigger than
; 1200) but it's an .EXE file and amount
; of bytes it takes in the memory will be
; less than the file size.

; no "org 0100h" here - it's an overlay
; file
use16

; these jumps MUST be short jumps (2 bytes)
jmp @LnPitch8 ; 1
jmp @LnPitch6 ; 2
jmp @PrintRow ; 3
jmp @ResetAll ; 4
jmp @SkipPage ; 5
jmp @SkipLine ; 6

; call order for each page:
; [4] ResetAll
; [1] LnPitch8
; [3] PrintRow (for each row)
; [5] SkipPage

; -----

; and here long calls can be used
; stubs since this driver don't need it
@LnPitch8:
@LnPitch6:

```

```

@SkipLine:
; set zero flag as no error
xor ax, ax
retf

@PrintRow:
call @DumpFile
retf

@ResetAll:
call @OpenFile
retf

@SkipPage:
call @FreeFile
retf

; -----

@OpenFile:
; init .BMP header
xor ax, ax
; file size = HDRLEN for now
mov [_bfSize_lo], HDRLEN
mov [_bfSize_hi], ax
; image height = 0
mov [_biHeight_lo], ax
mov [_biHeight_hi], ax
; image size = 0
mov [_biSizeImage_lo], ax
mov [_biSizeImage_hi], ax
; find first non-existent file
@OpenFile_find:
; next number
inc ax
; save it in the file handle variable
mov [_file], ax
; convert decimal number to text string
mov di, _name + 7
mov bx, 10d
mov cx, 4
@@:
xor dx, dx
div bx
add dl, '0'
mov [ds:di], dl
dec di
loop @b
; try to get file attribute
; it's a check if file or a directory
; with that name already exists
mov dx, _name
mov al, 000h
mov ah, 043h
int 021h
; error if carry flag is set (probably
; file not found)
jc @f
xor ax, ax
; make file handle invalid in case of

```



```

; overflow
xchg ax, [_file]
cmp ax, 9999d
; if 9999 overflow - do not print
; anything
jz @OpenFile_ret
; or else - continue checking
jmp @OpenFile_find
@@:
; clear file handle for error - will be
; used as invalid handle
; since file handle 0 already preserved
; as standard input handle
; also cx - file attributes
; create new file
xor cx, cx
mov [_file], cx
mov dx, _name
mov ah, 03Ch
int 021h
jnc @f
; zero flag will be non-empty for error
mov al, 001h
jmp @OpenFile_ret
@@:
; no error - save file handle for
; future use
mov [_file], ax
; write .BMP header
mov dx, _head
mov cx, HDRLEN
mov bx, ax
mov ah, 040h
int 021h
; no error
xor ax, ax
@OpenFile_ret:
or ax, ax
retn

; -----
@FreeFile:
; set as error
mov ax, 001h
; check for valid file handle
cmp [_file], 000h
jz @f
; seek to file start
xor cx, cx ; offset hi = 0
xor dx, dx ; offset lo = 0
mov bx, [_file] ; file handle
mov al, 000h ; seek from start
mov ah, 042h
int 021h
; write updated .BMP header
mov dx, _head
mov cx, HDRLEN
mov bx, [_file]
mov ah, 040h

```

```

int 021h
; close file
mov bx, [_file]
mov ah, 03Eh
int 021h
; no error
xor ax, ax
; file handle not valid anymore
mov [_file], ax
@@:
or ax, ax
retn

; -----
; input:
; es:si - image buffer address
; cx - size for image buffer in bytes
; al - non-zero if printing 0x0A at the
; end required after 0x0D
@DumpFile:
; set as error
mov ax, 001h
; check for valid file handle
cmp [_file], 000h
jz @DumpFile_ret
; revert image data
; note that each byte in B/W screen
; mode or a .BMP file represents
; horizontal line of 8 pixels
; but the data of a matrix printer is
; different: each byte represents a
; VERTICAL line of 8 dots
; because of that binary image data
; needs to be rearranged before it can
; be saved as bitmap
mov cl, 7
@for_j:
; y = 0..7 also used as shift index
; save data pointer original value
push si
mov di, _data
mov ch, 120d
@for_i: ; x = 0..119
xor al, al
mov bl, 8
@for_k: ; index in 8 byte block
mov ah, [es:si]
; since only cl register can be
; used as shift index value use
; it as for_j loop index too
shr ah, cl
and ah, 1
shl al, 1
or al, ah
inc si
dec bl
jnz @for_k
; note that in printer driver bit 1
; means print (black ink) and 0

```



```

; means skip (white paper)
; but in the .BMP file 0 are black
; and 1 are white, so bits in byte
; needs to be inverted
not al
mov [ds:di], al
inc di
dec ch
jnz @for_i
; save cx
push cx
mov cx, 120d
xor ax, ax
; increment height
; do a decrement instead because this
; must be a negative value
; (see explanation below in .BMP
; header)
sub [_biHeight_lo], 001h
sbb [_biHeight_hi], ax
; update .BMP file headers
add [_bfSize_lo], cx
adc [_bfSize_hi], ax
add [_biSizeImage_lo], cx
adc [_biSizeImage_hi], ax
; write row of pixels to the file
; also cx - number of bytes to write
mov dx, _data
mov bx, [_file]
mov ah, 040h
int 021h
; check for any key pressed
call @CheckKey
; restore cx
pop cx
; rewind input data pointer to start
pop si
; test zero flag from CheckKey
jnz @DumpFile_ret
; next row
dec cl
jns @for_j
@DumpFile_ret:
or ax, ax
retn

```

```
; -----
```

```
@CheckKey:
; check key in buffer
mov ax, 0001h
call @ReadKey
; no key in buffer
jz @f
; ESC pressed - exit
cmp al, 01Bh
jz @f
; show pause text
mov dx, _wait
call @ShowText

```

```

; wait for key
mov ax, 000h
call @ReadKey
push ax
; clear pause text
mov dx, _fill
call @ShowText
pop ax
; ESC pressed - exit
cmp al, 01Bh
jz @f
; no error - continue printing
xor ax, ax
@@:
; if ESC pressed - close file
cmp al, 01Bh
jnz @f
push ax
; close file handle since printing
; aborted (@SkipPage will be never
; called)
call @FreeFile
; and delete partially saved file
mov dx, _name
mov ah, 041h
int 021h
pop ax
@@:
or ax, ax
retn

```

```
; -----
```

```
@ReadKey:
; ax = 0 - read, do not check
test ax, ax
jz @f
; check for key in buffer
mov ah, 001h
int 016h
jnz @f
xor ax, ax
jmp @ReadKey_ret

```

```
@@:
; read key from buffer
mov ah, 000h
int 016h
@ReadKey_ret:
or ax, ax
retn

```

```
; -----
```

```
@ShowText:
push dx
; go to screen center
mov dx, 01814h
xor bh, bh
mov ah, 002h
int 010h

```



```

; output string text
pop dx
mov ah, 009h
int 021h
retn

; -----

; data area
_name db 'PAGE0000.BMP',0
_wait db 'Pausing... Press a key to
continue.$'
_fill db '                                     '$'
; make everything below word-aligned
align 2
; DOS file handle
_file dw 0
; BITMAPFILEHEADER
_head dw 04D42h ; _bfType
_bfSize_lo dw 0 ; whole file
; size
_bfSize_hi dw 0
_bfReserved1 dw 0
_bfReserved2 dw 0
_bfOffBits dd HDRLen ; offset to the
; bitmap data
; BITMAPINFOHEADER
_biSize dd 40d ; header size
_biWidth dd 960d ; page width
_biHeight_lo dw 0 ; 760 / 1008
; (NewsMaster 1.x / NewsMaster II)
_biHeight_hi dw 0 ; if biHeight
; is negative, the bitmap is a top-down DIB
_biPlanes dd 1
_biBitCount dw 1
_biCompression dd 0 ; 0 - BI_RGB -
; uncompressed
; whole bitmap
; image data size
_biSizeImage_lo dw 0
_biSizeImage_hi dw 0
_biXPelsPerMeter dd 0
_biYPelsPerMeter dd 0
_biClrUsed dd 0
_biClrImportant dd 0
; palette
_palette dd 00000000h, 00FFFFFFh
HDRLen = $ - _head
; buffer to hold one line of pixels
; (960 / 8 = 120)
_data db 120d dup(0)

```

Код драйвера достаточно простой, с подробными комментариями, но объёмный, так что ниже будут кратко описаны только важные моменты.

1) Чтобы в начальной таблице было 6 коротких **jmp** (по 2 байта) и они не превратились в длинные (по 3 байта), были сделаны

небольшие подпрограммы, находящиеся сразу за этой таблицей, а уже из них вызывается необходимый код.

2) Подпрограммы **LnPitch8**, **LnPitch6** и **Skipline** указывают на один и тот же код – заглушку, которая ничего не делает и лишь возвращает статус для **zero flag**, что всё в порядке, т.к. в данном драйвере они не используются.

3) Подпрограмма **ResetAll** вызывает **OpenFile**, где выполняется целая последовательность действий:

- инициализируется заголовок будущего BMP-файла;

- предпринимается попытка найти ещё несуществующий файл перебором по маске **PAGE####.BMP**, где **####** – порядковый номер от 0001 до 9999; делается это через получение атрибутов файла – если атрибуты удалось получить, то файл существует, поэтому ищем следующий; такой способ лучше, чем попытка открыть файл, потому что, во-первых, файл потом не нужно закрывать (код короче), а, во-вторых, это также позволяет отсечь ситуацию с каталогами вместо файлов (например, каталог с именем **PAGE1234.BMP** нельзя открыть как файл, но и файл такой нельзя будет потом создать, потому что имя элемента файловой системы уже занято каталогом); если все 9999 файлов существуют, то возвращается ошибка – печать невозможна;

- предпринимается попытка создать файл, а если не получилось, то точно так же возвращается ошибка и печать останавливается;

- в случае удачной попытки создания файла, его дескриптор (описатель / handle), сохраняется для дальнейшего использования, а также сразу пишется начальный вариант BMP-заголовка, т.к. потом его в начало не запишешь (мы обновим его актуальными данными позже – перед закрытием файла);

4) Подпрограмма **PrintRow** вызывает **DumpFile**, где происходит сохранение присланных графических данных – перестановка битов и построчная запись в файл. Отдельно стоит заметить, что сейчас драйвер занимает 660 байт в



скомпилированном виде. Для записи строки он использует статический буфер в 120 байт $(120*8 = 960 \text{ точек})^7$. Можно было бы, конечно, не писать по одной строке, а сразу сформировать в памяти все 8 и записать их разом, но $660+(120*7) = 1500$ – иными словами, это невозможно, ибо тогда код драйвера не уместился бы в отведённые под него 1200 байт. Динамическое же выделение памяти не рассматриваем, т.к. статический буфер удобен ещё и тем, что память для него есть всегда, если драйвер успешно загрузился. Также в этой подпрограмме происходит перерасчёт высоты рисунка (отрицательной) и полей BMP-заголовков под размеры файла и изображения. Ну и, конечно, как и в оригинальном драйвере, выполняется проверка на нажатие любой клавиши, кроме **Esc**, что приостанавливает печать до повторного нажатия, в то время как **Esc** отменяет печать совсем, при этом закрывается и удаляется текущий недописанный BMP-файл. Обратите внимание, что в случае отмены печати подпрограмма **SkipPage** не вызывается, поэтому закрывать и удалять недописанный файл нужно именно здесь, чтобы не произошла утечка файлового дескриптора.

5) Наконец, подпрограмма **SkipPage** вызывает **FreeFile**, где происходит перезапись начального BMP-заголовка на корректный обновлённый, а затем и закрытие файла – запись рисунка окончена.

P.S. NewsMaster позволяет вставлять текст из текстовых файлов (имена должны соответствовать маске «*.TXT») – они грузятся в буфер обмена, откуда их потом необходимо вставлять в текст. Но русские символы так вставить не получится, поэтому для загрузки именно русского текста в редактор «Журналист» нужно будет перед этим обработать входной файл, и произвести замену букв соответственно таблице символов (русские => английские), которую можно построить, например, на основе данных из прошлой статьи:

```
; было
db 'АБВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЬЪЫЭЮЯ'
```

```
db 'абвгдежзийклмнопрстуфхцчшщъыэюя'
db 'Èè'
; стало
db 'ABWGDEVZIJKLMNOPRSTUFHC^[]_YX\@Q'
db 'abwgdevzijklnoprstufhc~{}',07Fh,'yx|\`q'
db 'Ee'
```

Данные приведены больше для наглядности, и в реальной программе для конвертирования, конечно же, лучше сделать таблицу замены для всех 256 значений байта – такой способ будет и быстрее, и удобнее.

И пара слов об оригинальных драйверах для принтера. Как уже было сказано в прошлом постскриптуме, в NewsMaster II все они были упакованы в файл **PRINTERS.PRG**. Формат этого архива простой, без сжатия, так что драйвера оттуда можно легко достать при помощи вот такого скрипта:

```
Get Count Short
For I = 1 To Count
  GetDString Name 14
  Get Offs Long
  Get Size Long
  Log Name Offs Size
Next I
```

Этот код необходимо сохранить в текстовый файл с именем **UNPAKPRG.BMS**, поместить рядом упомянутый файл **PRINTERS.PRG**, а также программу **QuickBMS.exe**, затем вызвать:

```
quickbms.exe -d UNPAKPRG.BMS PRINTERS.PRG
```

Программу QuickBMS можно взять здесь: <http://quickbms.aluigi.org/>

Исходные коды из этой статьи вместе с готовыми скомпилированными программами доступны по ссылке:

http://systools.losthost.org/files/news_prg.zip

Специально для Downgrade N28'2019

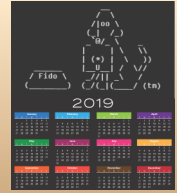
© SysTools 2019

<http://systools.losthost.org/>

⁷ Из-за того, что весь блок в 1200 байт используется только драйвером для принтера, то необходимости в статическом буфере на диске нет, и его можно заменить с «**data db 120d dup(0)**» на «**data rb 120d**», сократив размер файла. Однако когда все переменные объявлены явно в статическом виде, то это позволяет легко и наглядно контролировать ситуацию с выходом размера драйвера за отведённые ему границы.



КАЛЕНДАРЬ ПО МОТИВАМ ИСТОРИИ ФИДОНЕТ (ИЮЛЬ – СЕНТЯБРЬ)



ИЮЛЬ

1 июля

1985. Bob Hartman (Fido #10101, Нашуа, штат Нью-Гэмпшир) начал гейтовать две ньюс-группы USENET (net.micro.pc и net.lang.c) в Fidonet. В этой версии «UN*X Gateway» все письма из группы поступали одним большим файлом, как-то разделять их (и тем более отвечать) было нельзя.

1993. Andrew Milner, Mats Birch, Mats Wallin предлагают новый формат почтовых баз – JAM (The Joaquim-Andrew-Mats Message Base Proposal) и публикуют проект под названием JAM-001. Этот формат получил большое распространение в FTN-сетях.

1994. В нодлисте появилась сеть 2:5077 (Тюмень).

1994. В Цюрихе (Швейцария) началась конференция SwissCon/EuroCon'94.

1996. Возобновлён выпуск FidoNews после месячного перерыва, вызванного тем, что предыдущий редактор Donald Tees пропал из Фидо. Новым редактором стал Christopher Baker, 1:374/14 (Тайтусвилл, Флорида).

2012. В регионе 2:24 (Германия) начат процесс удаления «мёртвых» эх. Таких было около 200, «живых» же – 37. Процесс удаления занял около двух месяцев.

2018. Новым сборщиком поинтлиста N5020 (Москва) назначен Юрий Богоявленский, 2:5020/601.

2 июля

1993. В коммуне Ремих (Люксембург) началась европейская конференция Фидо «LuxCon/EuroCon'93».

1999. В нодлисте появилась сеть 2:457 (Витебск, Беларусь).

3 июля

1998. В нодлисте появилась сеть 2:5018 (Дедовск).

1998. В городе Кёнигсбрунн (Германия) началась очередная европейская конференция Фидонет – EUROCON'98. Для её проведения был целиком арендован отель Zeller.

4 июля

1997. В городе Лафайетт (штат Луизиана, США) заработала Positronium BBS (сисоп – Ben Ritchey). Она оказалась одной из самых долгоживущих BBS. Диалогная часть проработала 21 год, по телнету доступна до сих пор.

2008. Координатором Региона 50 (Россия) выбран Владимир Донской.

5 июля

1996. В нодлисте появилась сеть 2:5039 (Архангельск). С 1992 по 1995 год в Архангельске существовала сеть 2:5031, отданная затем Мурманску.

1996. В нодлисте появилась сеть 2:5089 (Чимкент, Казахстан).

2009. Составлена первая версия русскоязычного списка «Доступ к фидо-ресурсам из интернета». Список включает доступ к эхам через WebBBS, NNTP и Telnet, а также к файло-эхам через FTP и HTTP. Поддерживается до сих пор.

6 июля

1996. Christopher Baker (1:374/14), новый редактор бюллетеня Фидо, создал эконоференцию FIDONEWS.



2001. Координатором Региона 2:46 (Украина и Молдова) становится Pavel Gulchouck.

2018. Координатором первой зоны Фидо (Северная Америка) официально стал Nick Andre.

2018. Координатором Региона 40 (Израиль) стал Авшалом Донской.

8 июля

2005. Координатором Региона 50 (Россия) становится Алексей Баринов.

9 июля

1997. Игорь Милонов (2:5020/1132) пишет своё ставшее знаменитым «последнее письмо» с оскорблениями в адрес известных фидошников, рассылает его во все доступные ему эхи и получает закономерную экскоммуникацию.

1999. В нодлите появилась сеть 2:6053 (Балаково).

10 июля

1992. В нодлите появилась сеть 2:5060 (Таганрог).

1998. В нодлите появилась сеть 2:5091 (Джезказган, Казахстан).

2018. Хенри Лаан (2:46/49) анонсировал телеграм-бота, созданного Антоном Рыжковым (2:5030/1081). Бот умеет автоматически выдавать поинта и линка, управлять подпиской, смотреть различную информацию, получать уведомления о пришедшем нетмейле и новых письмах в карбонке.

11 июля

1997. В нодлите появилась сеть 2:455 (Могилёв, Беларусь).

12 июля

2002. Координатором Региона 2:45 (Беларусь) становится Sergey Mookhin.

13 июля

1990. Начало европейской конференции Eurocon / Techcon в Антверпене, Бельгия.

1998. Анонсирован сайт FidoNews в Эстонии (с переводом FidoNews на эстонский язык).

14 июля

1990. В Антверпене (Бельгия) начался Eurocon IV. Провести конференцию в Бельгии предложил Ward Dossche. В рамках Eurocon IV также проходила двухдневная конференция Techcon-I, целиком посвящённая фидософту.

2008. Поинткомплект fidoip стал доступен не только в версии для Linux, но и для FreeBSD и Windows.

2016. Дмитрий Каменский анонсировал запуск на wfido.ru интерфейса модератора.

15 июля

1991. Анонсирована программа MRCYPT 1.10 (автор – Eric Jacksch, из Непина – Онтарио, Канада) для PGP-шифрования нетмейла в FTN-сетях. Программа не трогала заголовков и клуджи, шифруя только текст.

1994. В нодлите появилась сеть 2:5078 (Нижний Тагил).

1999. Состоялся релиз редактора GoldEd+ 1.0.0.

2010. Иван Агарков собрал фидопакет для MacOS X с автоконфигуратором – macfidoip.

16 июля

2009. Юрий Рощупкин выпустил второй номер «русского Фидоньюса».

2016. В Москве прошёл первый и последний Fidonet Install Fest, организованный Сергеем Чумаковым.

17 июля

1989. Впервые в публикуемом в FidoNews списке программного обеспечения для Fidonet и BBS появились подразделы «Apple Macintosh» и «Commodore Amiga».

1992. В нодлите появился регион 2:44 (Грузия) и сеть 2:446 (Тбилиси)

19 июля

1996. В нодлите появился регион 2:54 (Саудовская Аравия) – первый регион на Ближнем Востоке.



20 июля

1987. Thom Henderson (Клифтон, штат Нью-Джерси), анонсировал новую систему конференц-связи – MGM Group Mail System. Она была полностью совместима со стандартом FidoNet echomail и отличалась модульной структурой. Однако широкого распространения не получила.

1992. Stephen M. Dorman (Бабилон, штат Нью-Йорк) в статье для FidoNews озаботился вопросами общения с сисопами из России, кодировки кириллицы и русскоязычных эхоконференций.

21 июля

1995. В нодлисте появилась сеть 2:5043 (Южно-Сахалинск).

22 июля

1994. Сеть 2:466, ранее бывшая белорусской, стала Николаевской (Украина).

24 июля

1989. В FidoNews сообщено о том, что David Dodell подал в отставку с поста Z1C.

25 июля

1988. Стал известен первый случай попытки вынести виртуальные разборы в зал суда. Студентка из Индианы – пользователь BBS – подала иск в Федеральный суд на местного сисопа Bob Predaina, утверждая, что тот намеренно раскрыл её частные сообщения другим пользователям без её разрешения. Чем всё закончилось, история умалчивает.

2005. Джейсон Скотт (Jason Scott) сообщил о завершении четырёхлетней работы над проектом «BBS: The Documentary». Три DVD-диска про историю и субкультуру BBS и Fido (на английском языке) можно до сих пор заказать на сайте <http://www.bbsdocumentary.com/> (или посмотреть на YouTube).

26 июля

1991. В Уотфорде (графство Хартфордшир, Англия) началась конференция сисопов Fido Worldcon'91, включавшая в себя Eurocon V.

Спонсорами и организаторами выступили «Xenon Management LTD» и «P&O European Ferries».

27 июля

1987. Lee Rothstein, Phil Zimmermann и Steve Welch предложили использовать шифрование и электронную подпись с открытым ключом в системах Fidonet и BBS.

28 июля

1986. Принято решение о создании Комитета стандартов Фидонет (FidoNet Standards Committee). Чуть позже в его название было добавлено слово «Technical» – FTSC.

29 июля

1985. Bob Hartman (Нашуа, штат Нью-Гэмпшир) сообщил о разработке фидо пакета ROVER и выложил бета-версию для тестирования. Скорее всего, это была первая альтернатива программе «Fido» Тома Дженнингса.

1994. В нодлисте появилась сеть 2:5054 (Пермь).

1994. В нодлисте появилась сеть 2:4643 (Кировоград, Украина).

1996. В FidoNews объявлено о создании Zone7 – проекта выделения стран ex-USSR в отдельную зону FidoNet. О проекте говорили очень долго, но он так и не был реализован.

1996. David Rye (Грандин, штат Флорида) предложил желающим присоединиться к разработке фидософта с графическим интерфейсом («GNU FidoNet»).

1997. Стал известен первый случай массового заспамливания эх во второй зоне. Он произошёл в R25 (Великобритания). Злоумышленник воспользовался уязвимостью почтовой программы «Blue Wave» и заспамил все доступные эхи в этом регионе. К сисопу узла никаких санкций применено не было.

2016. В Екатеринбурге торжественно открыта памятная табличка Евгению Зорину, одному из основателей компьютерной сети Fidonet на Урале.



30 июля

1993. В нодлите появилась сеть 2:452 (Гомель, Беларусь).

31 июля

2018. Анонсировано приложение «GoldED+ VK» для чтения фидошных эх ВКонтакте, стилизованное под редактор GoldEd. Автор – Антон Рыжков (Санкт-Петербург).

АВГУСТ**1 августа**

1990. В Линдхерсте, Нью-Джерси, началась конференция FidoCon'90 aka Conclave'90.

1997. Объявлено, что новым координатором третьей зоны стал Alwyn Smith (Австралия).

2 августа

1988. Том Дженнингс получает свидетельство N1498374 (серийный номер 73665097) на товарную марку «Fido».

3 августа

2009. Michiel van der Vlist (Дрибегрен, Нидерланды) впервые в FidoNews поднимает вопрос «Fidonet через IPv6».

4 августа

1990. На годовом общем собрании членов IFNA (International FidoNet Association, Inc.) было утверждено решение Совета директоров о роспуске IFNA.

2000. В нодлите появилась сеть 2:6028 (Дубна).

2011. Павел Гульчук поднял веб-интерфейс для домена binkp.net. Сисопам IP-узлов стало можно самостоятельно править свои записи в домене.

5 августа

1985. Первое издание National Fido User List (США). Его можно назвать расширенным нодлистом.

6 августа

1993. В нодлите появилась сеть 2:468 (Херсон, Украина).

2011. Сеть 5020 (Москва) перешла с двухуровневой системы нетмейл-хабов на одноуровневую полносвязку.

7 августа

1987. Вышел первый NodeDiff (219), собранный при помощи программы MakeNL от Ben Baker. Система diff'ов просуществовала до настоящего времени, однако сейчас практическая необходимость в ней отпала, и нод- и поинтлисты распространяются целиком.

1998. В нодлите появилась сеть 2:5099 (Жуковский-Раменское, Московская область).

9 августа

1996. В нодлите появилась сеть 2:5034 (Кострома).

12 августа

1996. Jim Barchuk (1:141/355) предлагает использовать в Fidonet гипертекст. Правда, первоначально его предложение касалось только бюллетеня FidoNews.

2016. Дмитрий Каменский запустил на движке vBulletin форум – гейт фидошных эхо-конференций forum.wfido.ru.

2018. Александр Сковпень сообщил, что начал работу над консольным юникодным редактором GoatEd:

<https://github.com/askovpen/gossiped>

13 августа

1990. Dennis McClain-Furmanski (Черчленд, штат Вирджиния) публикует в FidoNews большую статью «Networking The Soviet Union». В ней, в частности, он призывает присылать ему дискеты с программным обеспечением для Fido и BBS, чтобы курьеры могли переправить это в СССР. Также он предлагает не настаивать на оплате софта, который будет использоваться сисопами в Советском Союзе.

1992. В Луисвилле, Кентукки, начинается IBECC'92 – International BBSing and Electronic Communications Conference.



1992. Первая конференция BBSCON в Денвере, штат Колорадо.

2018. В R46 (Украина и Молдова) официально разрешены E-mail-only nodes (EON).

14 августа

1986. Начало Международной Конференции Фидонет (International FidoNet Conference) в Колорадо-Спрингс. Мероприятие продолжалось четыре дня. На нём присутствовало около 100 сисопов из США, Канады, Великобритании и Нидерландов. Сисопам был представлен первый состав FTSC во главе с Randy Bush. Том Дженнингс объявил о скором выходе Fido v.12 и введении в фидошную нумерацию зонального принципа. В рамках конференции прошла первая официальная встреча членов Международной ассоциации FidoNet (IFNA). В Совет директоров вошли Ken Kaplan, Ben Baker и Thom Henderson. К слову, членство в IFNA с правом голоса стоило \$25, что вызвало, мягко говоря, бурную полемику.

1997. Объявлено о создании в США сисопами BBS организации «The Council for Online Community Alternatives» (COCA). Она была сформирована для содействия всем желающим содержать BBS и WebBBS. Членство было бесплатным.

15 августа

1991. Начало пятой ежегодной конференции первой зоны (Северная Америка) – FidoCon'91 под кодовым названием «Новое начало» («A New Beginning»). Мероприятие также включало в себя Международную конференцию BBS («International BBSing Conference»).

2015. Дмитрий Каменский сообщил об открытии telnet-BBS [telnet://wfidonet.ru](http://wfidonet.ru).

16 августа

1996. В нодлисте появилась сеть 2:454 (Брест, Беларусь).

17 августа

1998. Редактором FidoNews становится Henk Wolsink (5:7104/2). Впервые выпуск был

подготовлен в пятой зоне (Порт-Элизабет, ЮАР).

18 августа

1985. Bob Hartman (Нашуа, штат Нью-Гэмпшир) выложил работающую версию фидо-пакета ROVER v.1.6. По-видимому, это была первая альтернатива программе «Fido» Тома Дженнингса.

1987. Том Дженнингс получает свидетельство N1452977 (серийный номер 73591035) на товарную марку «Fidonet».

19 августа

1993. На сисопке объявлено о создании треугольника между московскими узлами 5020/35-5020/68-5020/134. Он пришёл на смену полносвязке типа «блин».

20 августа

1985. Tom Jennings публикует вторую часть заметок, озаглавленных «FidoNet History».

1987. В Александрии, Вирджиния, началась Четвёртая Международная конференция FidoNet. Она продлилась до 23 августа. Tom Jennings представляет свою программу Fido v.12 и высказывает недовольство по поводу чрезмерной роли IFNA в управлении Fidonet. Официально избран первый Совет Директоров IFNA. Ken Kaplan председательствует, президентом стал Don Daniels, вице-президентом – Mark Grennan, секретарём – Tom Marshall. Представлен проект Policy v.4.

21 августа

1992. В пригороде Таллина (Эстония) началась конференция BBSSummer'92. В ней участвовало около 75 человек, в основном из Эстонии, а также три сисопа из R50, в том числе Майкл Браво, координатор питерской сети. Кроме того, присутствовали европейский координатор Ron Dwight и администратор интернет-гейта «Casino» Esa Laitinen.

1996. Опубликован проект Эхополиси второй зоны – General Echomail Policy 2. Для его обсуждения создана эхоконференция ECHO-POL2.



1998. В нодлисте появилась сеть 2:5098 (Кировск, Ленинградская область).

23 августа

1988. Bob Hartman (Нашуа, штат Нью-Гэмпшир) выпускает Conference Mail System Revision 4.00 – систему почтовой рассылки, предназначенную в первую очередь для связывания фидошных эх и групп USENET.

24 августа

1989. Начало Международной Конференции Fidonet «FidoCon'89» в Сан-Хосе, штат Калифорния. Она продлилась до 27 августа.

1992. Майкл Браво (координатор питерской сети) публикует в FidoNews инструкцию о том, что нужно учитывать при директных звонках модемом в Россию.

1992. В Z1 (Северная Америка) впервые опубликован список файлохоббов. Организовать «файлбон» предложил Erik VanRiper в 1991 году.

25 августа

1985. Первая ежегодная сисопка («Sysop's Picnic») в Южной Флориде, на территории Tropical Park.

1988. Начало Международной Конференции Fidonet «FidoCon'88» в Цинциннати. Она продлилась до 28 августа.

1997. Damian Walker (Кингстон-апон-Халл, Англия) анонсировал написанную им игру по переписке (PBM) «Astrowar», разработанную специально для Fidonet, а также программу «Astromail», позволяющую воспроизводить игру с помощью нетмейла. Программы были выпущены под лицензией GNU GPL, бинарники – под Linux и DOS (386 DPMI).

26 августа

1994. В нодлисте появилась сеть 2:5012 (Курган).

1994. В нодлисте появилась сеть 2:5087 (Душанбе, Таджикистан).

1996. У бюллетеня FidoNews появился собственный оф. сайт в интернете (ныне недоступен): <http://ddi.digital.net/~cbaker/fidonews.html>

27 августа

1993. На BBSCON'93 в Колорадо-Спрингс создана ассоциация «National Online Media Association» (NOMA).

28 августа

1989. Сообщено о том, что Совет координаторов зон единогласно выбрал нового Международного координатора. Им стал Matt Whelan из Сиднея, Австралия.

2009. В нодлисте появилась сеть 2:6078 (Калининград).

31 августа

2005. R50C сообщил о том, что Z2C авторизовал флаг INA, предназначенный для указания IP-адреса.

СЕНТЯБРЬ

1 сентября

1993. Создан треугольник между московскими узлами 5020/35-5020/68-5020/134. После того, как вершины треугольника стали «падать», он был назван Бермудским.

1994. George Pease вступил в должность Международного Координатора Фидонет (IC). На этом посту он заменил Matt Whelan. George Pease был выбран на Совете Координаторов Зон единогласным решением ZС – 6:0. Это первый случай, когда IC был выбран не из числа действующих ZС.

1995. В нодлисте появилась сеть 2:5095 (Подольск).

1995. Kevin Kuphal выпускает, судя по всему, первый 32-битный тоссер под Windows NT/Windows 95 – ALTAIR v1.00a. Он продавался в минимальной «комплектации» за \$5, и за \$50 – в максимальной.

2003. Алексей Сердюков (2:5020/1042.42) анонсирует в FidoNews новую эху RUSSIAN_TUTOR для желающих выучить русский язык.

2 сентября

1994. В нодлисте появилась сеть 2:5023 (Калуга).



4 сентября

1992. В Далласе, штат Техас, началась конференция сисопов Северной Америки Fido-Con'92 под кодовым названием «Gateway Networking».

2010. Юрий Лукьянец выпустил первую тестовую версию вКонтактовского приложения vFido.

7 сентября

1990. Началась конференция «FIDOCON 1990 – ZONE 5» (Африка).

9 сентября

1985. Rives Mc Ginley и Luck Hurder (1:101/105 – Орлеан, Массачусетс) создали бесплатный сервис Fidoграм для передачи личных сообщений, пришедших по Фидо, адресатам через радиоловительскую сеть голосом или при помощи азбуки Морзе (протокол AX.25).

10 сентября

1989. Появление первых узлов Fido на территории Африки (будущая пятая зона Фидонет).

11 сентября

2017. Европейский координатор Z2C Ward Dossche авторизует в нод- и поинтлистах новые юзерские флаги для мобильных девайсов, использующих Aftershock (AFS), HotdogEd (HDG) и другие похожие программы (MOB).

12 сентября

2003. В нодлисте появилась сеть 2:6080 (Новошиимский, Казахстан).

13 сентября

1988. Том Дженнингс получает свидетельство N1503643 (серийный номер 73665098) на товарный знак «Собака с дискетой».

1991. Иван Рябов получает узел 2:5020/30. Он стал известен, когда в дни августовского путча 1991 перетащил свою станцию в Белый дом и рассылал оттуда по Фидо сводки во все края.

16 сентября

1994. В нодлисте появилась сеть 2:5065 (Сочи).

18 сентября

2000. В FidoNews анонсирована книга Эмира Шабашвили «FidoNet. Руководство системного оператора: Профессиональная работа в непрофессиональной сети» и приведена её краткая аннотация.

19 сентября

1982. День рождения смайлика. Профессор Университета Карнеги-Меллона Скотт Фалман (Scott E. Fahlman) в письме на местную BBS впервые предложил использовать три символа, идущие подряд – двоеточие, дефис и закрывающую скобку, для обозначения «улыбающегося лица» в тексте, который набирается на компьютере.

21 сентября

1984. Ben Baker и Ken Kaplan публикуют первый нодлист – «St.Louis Nodelist». До этого нодлисты представляли собой листки бумаги, на которых Том Дженнингс писал изменения от руки, а затем приклеивал «нодлист» на стену.

1990. Появление ФИДО в СССР. Первые узлы сети Фидонет на территории России появились в Новосибирске, их системными операторами были Владимир Лебедев и Евгений Чуприянов, более известный как Eric Fletcher. 21 сентября 1990 года эти узлы появились в чехословацком сегменте нодлиста.

22 сентября

1997. Для бюллетеня FidoNews зарегистрировано доменное имя в интернете. Создан сайт <http://www.fidonews.org> (доступен до сих пор).

2018. Анонсирована эха RU.MODEM.NERDS, трафик которой планируется передавать только по модему.



23 сентября

1985. В Фидонет появляется первая не IBM-совместимая система – узел Zeta из Австралии работал на компьютере Tandy TRS-80.

1988. В Денвере начался Первый ежегодный TBBS User's Group Symposium.

24 сентября

1986. Joaquim Homrighausen объявляет о выходе созданной им и Peter Stewart программы FrontDoor v1.0, сочетающей в себе мейлер, редактор и некоторые утилиты. Официально релиз состоялся 12 октября 1986. Этот пакет был первым, получившим распространение в ex-USSR.

25 сентября

1997. Создана эхоконференция STARPER.UNLIMITED.

26 сентября

2003. В нодлисте появилась сеть 2:6001 (Брянск).

27 сентября

1993. Icon Systems Ltd выпускает программу для гейтования MailGate Gateway v.1.0.

1996. В нодлисте появилась сеть 2:4647 (Павлоград, Украина).

28 сентября

1985. Состоялась Южнокаалифорнийская пляжная Фидо-вечеринка (SoCal Fido beach party).

1990. Появление первой фидошной станции в Москве. Kremlin BBS (сисоп – поляк Тадеуш Радюш) де-факто становится узлом ФИДО и появляется в мировом нодлисте (правда, в польском сегменте).

1990. Регион 2:49, ранее числившийся за Польшей, а потом за Южной Африкой, передан Эстонии. Номер сети 2:490 стал принадлежать Таллину.

2009. Европейский координатор Z2C начинает выпускать ежедневный полный нодлист. До этого нодлист выходил еженедельно по пятницам.

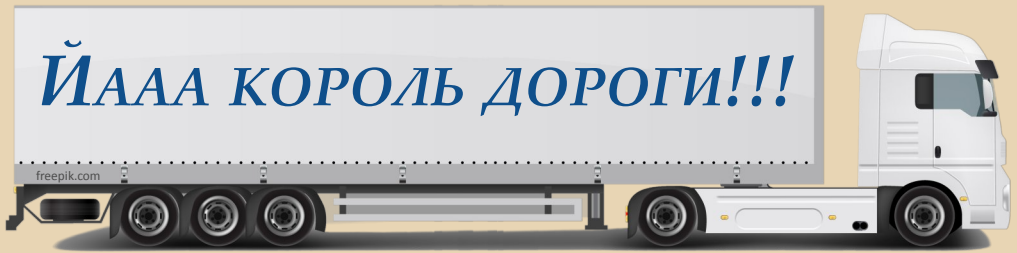
30 сентября

1985. Bob Hartman (1:132/101 – Нашуа, штат Нью-Гемпшир) доработал свой «UN*X Gateway» – гейт из Usenet в Fidonet. Теперь даже стало возможно писать в ньюс-группы из Фидо.

1995. Принят ныне действующий стандарт FTS-0001.016 «A Basic FidoNet(r) Technical Standard».

Владимир Фёдоров, 2:50/15





Когда я училась в школе, компьютер дома – это была редкость. Не знаю, как у одноклассников, но у меня компьютера не было. Но зато компьютер был у дедушки с бабушкой. По выходным я ходила к ним в гости и зависала за компьютером. На компьютере у дедушки было много разных игр. Начиная от простейших пасьянсов и заканчивая вторыми «Героями».

Больше всех мне нравилась игра «Дальнобойщики 2». Вы можете удивиться, чем девочку могла зацепить мальчишеская игра. Зацепила меня игра тем, что музыкальное сопровождение было в исполнении моей любимой группы – «Ария»! А ещё мне нравился тот драйв и адреналин, когда ты под группу «Ария» вжимаешь газ и мчишься вперёд, обгоняя своих соперников. Соперники дышат тебе в спину, но ты мчишься, минуя все посты полиции, и пытаешься удрать от бандитов. И как же ты бываешь доволен, когда доезжаешь до конечного пункта первым и получаешь кубок первенства.

Поэтому я решила рассказать вам именно об этой игре.

Установка игры на современный компьютер

Диск с «Дальнобойщиками» я пронесла через школьные годы, через молодость, которая ещё не закончилась. Этот диск занял почётное место среди остальных дисков, которые у меня скопились за годы пользования компьютером. Диск пережил дедовский старый компьютер и два моих – стационарник, купленный с рук в 2007-м году, и ноутбук, купленный

в 2011-м. Приобретая новый компьютер, решила установить дальнобойщиков и на него. Но, к сожалению, установить и запустить мне не удалось. Игра ставилась, но запускаться не хотела – вываливалась с ошибкой. Сказать, что я была в печали – не сказать ничего. Читайте, новый компьютер был куплен зря. В принципе, если бы не накрылся ноут, этого не случилось бы, но ноут приказал долго жить, а новый компьютер уже стоит и загружает седьмую винду.

На новый компьютер я загрузила много разных игрушек, но моё сердце было отдано именно «Дальнобойщикам». В скорби по любимой игре были куплены такие игрушки, как «Восемнадцать стальных колёс. Конвой» и «Восемнадцать стальных колёс. По дорогам Америки», но они не оставили такого яркого впечатления, как «Дальнобойщики». Поэтому я слёзно молила своего мужа, чтоб он закачал «Дальнобойщиков» из интернета. Но скачанный им вариант также вываливался с ошибкой.

Но тут помог вселенский разум, и нам удалось найти программу DgVooDoo2 (https://yadi.sk/d/e_SiYr20sC8Gq) и небольшую инструкцию по использованию ([ссылка](#)).

Итак, из архива программы необходимо взять файлы **DDraw.dll**, **dgVoodooSetup.exe**, **D3DImm.dll** и кинуть в каталог установки игры. На моём компьютере, например, это **C:\Program Files (x86)\King**. После этого запускаем **dgVoodooSetup.exe** (см. рис. 1), выбираем полный экран (**Full Screen**), жмём **Apply** и, вроде, этого хватило для запуска игрушки щелчком по ярлыку или файлу **King.exe**.



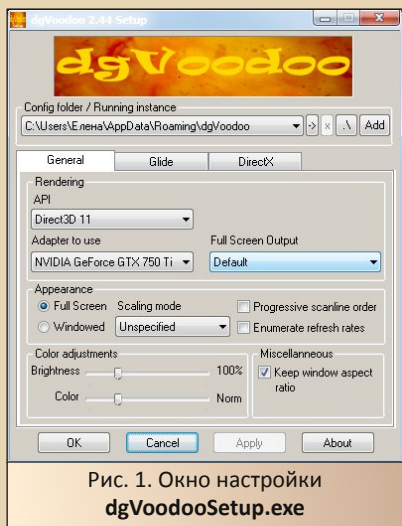


Рис. 1. Окно настройки dgVoodooSetup.exe

Инструкция просила нажать автопоиск видеокарты после запуска игры, но игра прекрасно запустилась и позволила начать гонку.

Теперь играем

Вот настал тот волнительный момент, когда можно запустить игру и окунуться в мир дорог и запаха бензина. С трепетом нажимаю на ярлычок игры. Начинается запуск, и вот уже из динамика доносится до боли знакомые звуки вступительного ролика игры. Ролик показывает нам, что нас примерно ждёт, и с чем мы можем столкнуться в ходе игры. Альтер-эго игрока голосует на трассе и останавливает фуру. Водитель фуры везёт игрока, а по дороге делится премудростями профессии – рассказывает, как уходить от полиции или бандитов, как брать грузы, затрагивает щекотливый момент контрабанды – палёных грузов – и рассказывает о тайной дороге на Алмазный, которая скрыта где-то в горах между Шахтами и Боровым. К этой дороге мы ещё вернёмся.

Прежде, чем приступить к игре, надо определиться с внешностью водителя. Игра позволяет выбрать водителя любого пола и любого цвета кожи. Так как я – девочка, я выбираю игрока женского пола. ☺ Есть там такая brutальная девчонка с короткими рыжими волосами и в кожаной косухе (см. рис. 2) – это я.



Рис. 2. Это я – в «Дальнбойщиках»

Также для комфортной игры можно выбрать погодные условия, интенсивность движения на дороге, продолжительность времени суток. Лично для себя я выбираю средние погодные условия (есть две крайности – только солнце и гроза с молнией – средние погодные условия предполагают солнечную погоду вперемешку с дождём и грозой). Так как у меня нету прав – в реале – для игры я выбираю интенсивность движения ниже среднего, чтобы нечаянно в кого-нибудь не врезаться. Длительность времени суток выбираю тоже среднюю. При таких настройках день длится примерно 15 минут, столько же длится ночь.

Итак, мы выбрали настройки и теперь нажимаем вкладку «Новая игра».

База «Южный» – отправная точка для всех начинающих дальнбойщиков. В начале игры мы окажемся на ней. В начале игры дальнбойщику предоставляется большой грузовик. Возможно, парням по душе такая машина, но я – девочка, и я выбираю для себя старую добрую «Газель» – маленькую, но проворную.

Вообще, автопарк игры представлен разнообразными видами грузовых автомобилей. У каждого свои достоинства. Я выбираю «Газель» из-за того, что на ней можно за достаточно короткий промежуток времени разогнаться до 130 км/ч даже при наличии груза. А так как она маленькая, на ней с лёгкостью можно маневрировать между грузовиками.

Необходимой мне «Газели» на парковке базы «Южное» нет. Поэтому перед тем, как начать возить грузы, я еду в Приозёрск и покупаю



там «Газель». И первый груз, который я везу, я буду брать в Приозёрске.



Рис. 3. Моя тачила. :-). Обычно езжу на ней, но на рабочем компьютере мужа, на котором делал скриншот, не получилось купить :-).

Ну а дальше начинаются рабочие будни обычного российского дальнобойщика. Идёт череда заказов в разные уголки игрового мира. Карту мира вы можете видеть в левой части экрана (см. рис. 4). Также в интернете удалось найти достаточно подробную карту <http://dalnoboichiki.narod.ru/images/map.jpg>, но на ней нет тайных дорог – указанной выше дороги в Алмазный и переезда на Ельнино.

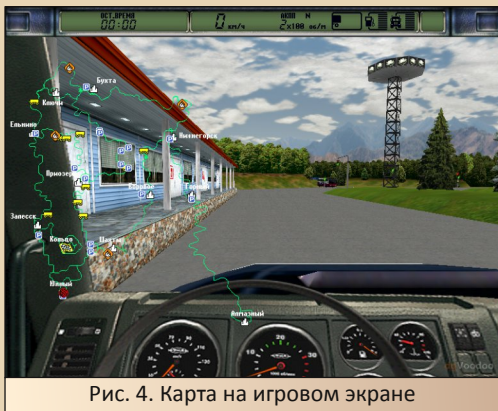


Рис. 4. Карта на игровом экране

Картой можно пользоваться в трёх режимах:

- кататься без карты по указателям;
- в развёрнутом режиме;
- в сокращённом режиме.

Развёрнутым режимом я пользуюсь тогда, когда беру груз на базе.

Итак, мы поехали. Какие неприятности нам могут встретиться по дороге? Конечно же, полиция, интересно, что, несмотря на российские дороги и не менее российские автомобили, полицейские больше напоминают американских – в тёмных очках, галстуках и подтяжках. К чему они могут прикопаться? К проезду на красный свет – плата 830 долларов, но когда ты очень опаздываешь, а полицейских рядом нет, можно проехать и на красный, не опасаясь штрафа. Если ты выезжаешь на встречную полосу, а в этот момент рядом находится полицейская машина – тоже можно попасть на бабки. Ну и, естественно, они тебя могут оштрафовать, если ты в кого-нибудь врезался. Там уже штраф от пяти тысяч долларов – за преступную халатность. Кроме того, в игре присутствуют полицейские вертолёты. Их можно встретить на двух дорогах – из Алмазного в Горный, так как там запретная зона, куда впускают и выпускают только по лицензии. Нет лицензии – никуда не поедешь. Второй вертолёт находится по дороге из Шахты в Боровое, но зачем он там курсирует – непонятно. Ещё вертолёт попадался на подъезде к Южному со стороны Залесска. Также, в отсутствие полицейских, на дорогах вас может встретить радар. Штрафы за превышение скорости зависят от скоростного режима на трассе. Самый маленький штраф 250 долларов. Самый же крупный штраф на моей памяти – 750 долларов. Но и даже радары можно обмануть – на станции техобслуживания можно установить на машине глушитель радаров.

Другая же беда – бандиты. Бандиты могут просто, даже, когда ты едешь пустой, предложить остановиться и поговорить. Этого делать ни в коем случае нельзя. Когда ты остановишься, они могут ограбить тебя на энную сумму – просто так. Также бандиты могут пристать к тебе и начать обстреливать, если ты везёшь запрещённые вещества. Ну и, конечно, когда вы пытаетесь уехать из Алмазного тайной дорогой, на этой тайной дороге от бандитов вы не уйдёте – придётся вам и отдать алмазы бандитам, и заказчику выплатить неустойку. Поэтому – по своему опыту – я рекомендую по тайной дороге ехать только в направлении



Алмазного. Обратно же возвращаться по обычной трассе, где курсирует полицейский вертолёт, а на дороге стоят стражи правопорядка.

И, кстати, про тайную дорогу. Тайная дорога на Алмазный находится в горах между базами Шахты и Боровое. Найти дорогу, в принципе, нетрудно – стоит только внимательно смотреть направо. Там можно увидеть между гор поворот. Вы едете по этому повороту и потом, въехав немного на гору, вы находите тоннель. Въезжая в тоннель, вы исследуете его и находите выход из тоннеля. Заблудиться там нельзя – где бы вы ни свернули, вы всегда куда-то да выедете. И, при любом раскладе, вы придёте в Алмазный быстрее, чем ваши конкуренты.

При каждой удачной доставке груза, если вы прибыли первым, вам дают лицензии на найм водителей. Лицензий может быть от одной до трёх штук. Но если у вас недостаточно денег для найма водителей, то лучше немного повременить и упорно зарабатывать деньги. Когда у меня появляется достаточная сумма денег, первым делом я еду на станцию техобслуживания и там прокачиваю машину по полной программе: устанавливаю всевозможные вспомогательные системы – например, антирадар, систему пожаротушения, систему сканирования груза и автомобилей. Также обязательно ставлю пуленепробиваемые стёкла, второй бензобак и шины-вездеходы – это основное, что я устанавливаю при первой возможности.

Также есть возможность заработать деньги не только перевозками грузов, но и участием в гонках на Кольце. Кольцо находится недалеко от базы Залесск. Вам на пейджер приходит сообщение о гонках и сведения о времени их проведения. В гонке принимают участие от трёх до пяти машин. Протяжённость гоночной трассы – три круга. Конечно, денег платят не особо много – всего пять тысяч долларов, но к пяти тысячам дают несколько лицензий на водителей. Совет будущим игрокам – перед тем, как ехать на гонки, убедитесь, что бензобак полон под завязку. ☺

Ещё один совет по игре – внимательно читайте все сообщения, которые приходят к вам на пейджер. В них указано, на каких дорогах производится ремонт.

Главная задача игры – на мой взгляд – создать свою компанию по перевозке грузов. Чем больше вы зарабатываете денег, тем у вас больше возможностей нанимать водителей. В игре существует лимит найма водителей. Ты можешь нанять всего пятнадцать водителей. Когда в твоей компании десять водителей, то существует большая вероятность, что кто-то из водителей станет твоим конкурентом по доставке грузов. В игре есть приятный бонус – когда ты нанимаешь своего пятнадцатого водителя, игра выплачивает тебе определённую сумму денег. Точную сумму сказать не могу из-за долгого перерыва в игре – сейчас приходится всё начинать сначала, и до пятнадцати человек мне не хватает восьми водителей.

Заключение

В заключение хочу сказать, что это был первый мой опыт в написании статьи. Мой муж долго просил меня написать. И вот, наконец, я исполнила его желание. Конечно, я не знаю, какой она получилась, но надеюсь, что для вас она будет интересна, а для кого-то даже и полезна, если вы соберётесь поиграть в мою любимую игру.

Если статью примут, и она понравится, то следующая моя статья будет про другую не менее увлекательную игру, а какая это будет игра – вы узнаете уже из следующей статьи. Всем лёгких дорог и счастливого пути!



Рис. 5. Я еду!





Наиболее известными разработками компании Nikita, безусловно, является комплект из пяти игр, описанный в статье «Четыре головоломки и аркада» в 15-м номере журнала Downgrade. Однако компанией были созданы и другие игры. В частности, гуляя по сайту Old-DOS.ru, автор откопал игрушку под названием «Путешествие по Европе» (скачать можно по адресу <http://old-dos.ru/dl.php?id=8302>).

Игрушка рассчитана на Windows 3.x, а так как сделана в России, то автор не смог пройти мимо. Собственно, этой игре и будет посвящена настоящая статья.

Одна из интересных особенностей игры в том, что при установке и начальной настройке с ней не возникло никаких проблем – скачиваем, распаковываем архив и усё – можно запускать игру. Единственное требование игры, которым не стоит пренебрегать – звуковая карта. Как и в рассмотренных ранее пяти играх, здесь музыка и звук являются достаточно важными элементами, например, когда вы закрываете окно игры, при подключённой звуковой карте вы услышите фразу «До свидания!» – мелочь, а приятно и интересно. :-)

Но прежде, чем мы будем закрывать окно игры, его надо хотя бы открыть. Что ж, делаем. При запуске на весь экран раскроется симпатичная заставка, а после этого откроется уже основное окно игры с картой Европы.

В первую очередь нас попросят ввести своё имя (см. рис. 1). Причём, попросят очень настойчиво – пока автор не представился, игра

даже не разрешала переключать окна (по крайней мере, по сочетанию клавиш **Alt-Tab**).



Рис. 1. Для начала игра попросит вас ввести своё имя

После ввода можно приступить к «работе» с программой. Почему не игре? Дело в том, что творение Nikit'ы ещё содержит в себе небольшой справочник по каждой европейской стране. Для вызова справки необходимо сперва изменить режим курсора – с лупы на указатель (кнопка с пальцем вверх окна), выбрать интересующую страну (выделится жёлтым цветом – см. рис. 2, а справа появится название, флаг, столица и связанная со страной картинка, кроме того, диктор произнесёт название страны – не зря же звуковую карточку требует :-), и нажать в правой нижней части окна кнопку с книжкой.



Рис. 2. Страна выделена





Рис. 3. Меню детальной информации о стране

После нажатия кнопки, откроется меню с более детальной информацией о стране (см. рис. 3), включающее в себя такие сведения, как:

- население;
- территория;
- форма государственного устройства;
- проживающие национальности;
- языки;
- исповедуемые религии;
- крупнейшие города страны и их население;
- климат;
- валюта.

При выборе нужного раздела нажатием соответствующей кнопки, в текстовом поле в правом нижнем углу окна будут выведены соответствующие сведения. Сведения даются актуальные, видимо, на 92-94-й год, когда и была разработана и выпущена игра.

При этом вы можете выбирать не только информационный раздел, но и страну, о которой желаете узнать информацию – выбранный вами пункт в меню информации сохраняется.

Чтобы закрыть режим информации, нажмите чёрную стрелку в левом верхнем углу окна – и снова перейдёте в основное меню.

Кстати, чтобы детально рассмотреть выбранную страну, не обязательно прибегать к лупе – пятая слева кнопка в верхней части окна

позволяет вывести на экран выбранную страну крупным планом с указанием городов (см. рис. 4).



Рис. 4. Германия крупным планом

Теперь же перейдём к играм. Что нам могут предложить?

- Найти страну по названию и флагу на карте.
- Найти ошибки на карте.
- Вопросы по странам.
- Добавить недостающую страну.
- Найти пары флагов стран.

Во всех играх против нас будет играть компьютер. Что ж, приступим.

Найди страну

Включить эту игру можно нажатием средней кнопки верхнего ряда – как раз справа от кнопки с книжкой. Игра предложит четыре режима:

- поиск страны по названию;
- поиск страны по столице;
- поиск страны по флагу;
- поиск названия страны в списке.

Выберем первый вариант – справа будет выведено название страны, а заодно и флаг – задача игрока найти страну на карте и ткнуть в неё. В случае попадания игрок получает очко. В случае ошибки – очко получает компьютер и страна окрашивается жёлтым уже автоматически.

При этом, как оказалось, играем с компьютером не на скорость, а на точность.





Рис. 5. Находим страну по названию

Со звуком игра и тут не подкачала – в случае нахождения страны игроком, вы услышите «Нашёл!», если нет – игра издаст звук нового сообщения в «аське» – да, именно то, что вы подумали – ICQ :-). Кстати, одной только фразой «Нашёл» игра не ограничивается – подбадривая игрока фразами «Хорошо», «Правильно» и т.д.

После окончания игры, когда все страны найдены, игрок услышит похвалу с диким акцентом «Харащо!» и вернётся в основное меню. Варианты игры с флагом и столицей не особо отличаются, разве что справа будут показывать или флаг, или столицу.

В случае же выбора названия страны, справа выведется список стран, где необходимо найти выделенную жёлтым страну (см. рис. 6) и выбрать её двойным щелчком.



Рис. 6. Теперь обратная задача – найти указанную страну в списке :-)

Как выяснилось, наибольшие затруднения автор испытал, выискивая на карте страну по флагу – увы, половину флагов даже и не знал. :-)

Поменять местами

Собственно, игра на исправление карты достаточно проста – на экран выводится фрагмент карты, где две страны поменяли названиями – необходимо эти две страны найти (см. рис. 7).



Рис. 7. Достаточно простая задача – ткнуть на страны с неправильными названиями

Вот только автор не понял, в каком же порядке выбирать страны – вроде бы показалось, что надо выбирать слева направо, но нет – практически каждый раз игра выдавала ошибку. Или я что-то не понял, или у игры в этом месте глюк. Увы, больше сказать нечего, так что, перейдём к следующей игре.

А знаете ли вы?

С викторинами уважаемый читатель, думается, сталкивался, и не раз – тысяча и один факт о ком угодно и чём угодно. Собственно, игра, посвящённая вопросам о стране, именно такая викторина – вопрос и три варианта ответа. Если что, на карте укажут страну, о которой спрашивают (см. рис. 8). В основном, вопросы касаются религии, населения, столицы и площади. Собственно, при определении на глазок



площади и населения – карта весьма неплохое подспорье.

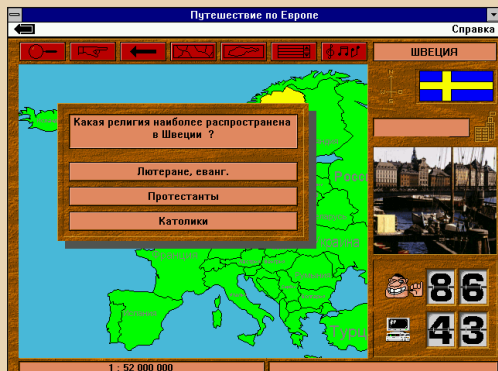


Рис. 8. Вопросы по стране – собственно, обычная викторина-тест – выбери правильный ответ. Если что – страна подсвечена на карте

В случае неправильного ответа игра сообщает вам не только об ошибке, но и правильный ответ (см. рис. 9).

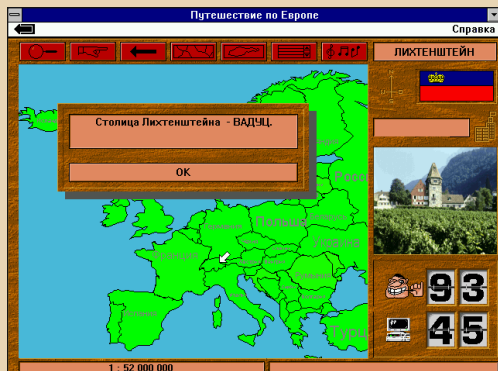


Рис. 9. Правильный ответ – не знал – теперь узнал :-)

Кстати, играя, увидел такой глюк – при достижении ста очков, игра пишет об ошибке отрисовки фона и счёт начинается с нуля (см. рис. 10).

В викторине же игра не спрашивает чего-то сверхъестественного – в принципе, все данные есть в справке. Конечно, трудно запомнить, но порой можно и угадать :-)) – ту же площадь или население. :-)

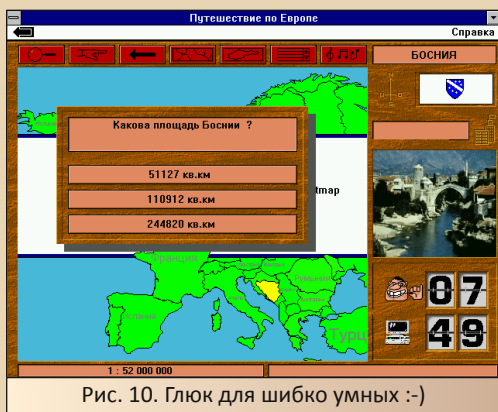


Рис. 10. Глюк для шибко умных :-)

Географический пазл

Игра с контурной картой – идёт следующей после викторины в нижнем ряду – представляет собой некое подобие мозаики или пазла. Вам дают белую карту с парочкой отмеченных стран и одну из стран, которую необходимо поместить на нужное место. Чтобы начать «устанавливать» страну, щёлкните по ней в квадратике справа – где, собственно, она находится до перемещения, и тащите на нужное место. Страна, которую установили вы, сохранит зелёную окраску, если установить у вас не получится с трёх попыток – страну установит компьютер, и она будет иметь жёлтый цвет (см. рис. 11).



Рис. 11. Процесс закрашивания карты Европы ☺



Собственно, процесс «закрашивания» практически не вызвал никаких проблем. За исключением незнания расположения некоторых стран, что можно увидеть на рисунке.

А так – немного плавал курсор мыши, но, так как игра не требовала установки страны один в один, то даже с шариковой мышью нормально всё удавалось выполнить.

Найди пару

Ну уж тут даже не дежа вю – просто классика жанра. Сколько было сделано игр такого типа? Кучи. И для Windows 3.11, думается, тоже. Вот и ещё одна – эта с европейскими флагами (см. рис. 12).

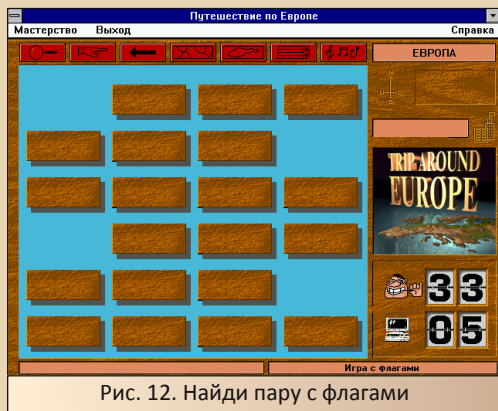


Рис. 12. Найди пару с флагами

Ходим по очереди – игрок и компьютер. Кто открыл два парных флага, ходит ещё раз. И так до победного конца.

Компьютер при этом также может открыть ошибочную пару, так что, тут более-менее честно, и даже такое ощущение, что игрок находится в более выгодном положении – он может видеть, какие пары открывал компьютер и запоминать. Использует ли машина результаты игрока – неизвестно, но, судя по ляпам – скорее нет.

Игры с отдельной страной

Выше автор писал, что игра даёт возможность открыть карту одной выбранной страны. Как понял читатель из названия, для этой карты есть отдельные игры.

Среди игр для одной страны доступны следующие:

- поиск города;
- исправление ошибок;
- викторина.

Игра по поиску городов имеет два режима – выбор города из списка – верхняя кнопка – и поиск города на карте – кнопка нижняя.

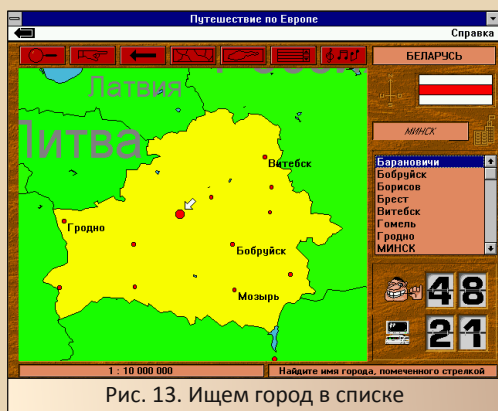


Рис. 13. Ищем город в списке

Кстати, в игре попался интересный глюк – если при угадывании города ткнуть мышью на стрелку, показывающую город на карте – игра сама выберет город в списке и игровую начислит очко.

Игра с перепутанными городами, как ни странно, в отличие от стран, не глючила, и действительно можно без проблем ткнуть мышью в два перепутанных города в любом порядке и, если игрок указал города правильно, он получает очко.





Рис. 14. Карта с «перепутанными» городами – в данном случае «перепутаны» Гомель и Борисов

Увы, отсутствие глюка в одном месте не исключает глюка в другом (см. рис. 15).



Рис. 15. Поиграли в перепутанные города :-)

Викторина же ничем не отличается от викторины по странам, разве что вопросы будут задаваться только по выбранной стране.

Нажатие на кнопки вызова игры с контурной картой или флагами ничем не отличаются от того, что описано выше, просто заполнение карты может начинаться с выбранной вами страны, а в наборе флагов окажется флаг, опять же, выбранной вами страны – вот и всё.

Заключение

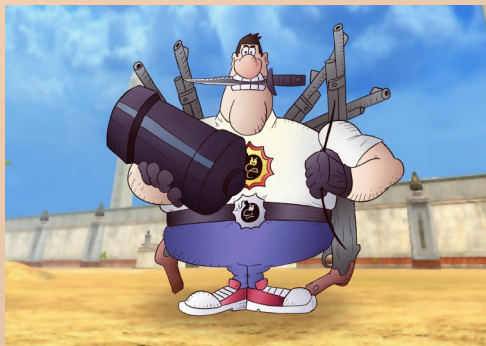
Что можно написать в заключении к такой – больше учебной, но всё же интересной игре? Собственно, игра, за исключением парочки указанных выше глюков, оставила очень приятное впечатление – особенно звуковое сопровождение. Уж не знаю, играл ли в игру человек, закачавший её на Old-DOS, но наличие звуковой карты действительно усиливает положительное впечатление от игры.

Что ещё добавить? Понятно, что игра – плод своего времени – у Беларуси бело-красно-белый флаг, а в Германии в ходу дойчмарка, но, с другой стороны – игра становится ещё и исторической свидетельницей и не теряет от этого ни в качестве, ни в интересности.

Андрей Шаронов (Andrei88)



Просто разный юмор



Картинки по мотивам компьютерных игр и мультика «Остров сокровищ». Автор – [T_T]. Его ВК: <https://vk.com/thef1amethr0wer>



ПОСТАВИЛ КАПЧУ ОТ БОТОВ

www.de-motiv.ru

off the mark .com

by Mark Parisi



© 2019 Mark Parisi Dist by Andrews McMeel Synd.

offthemark.com

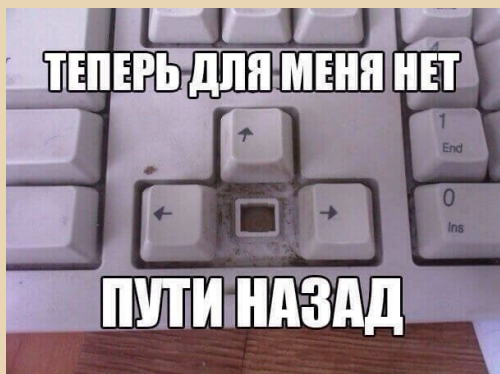
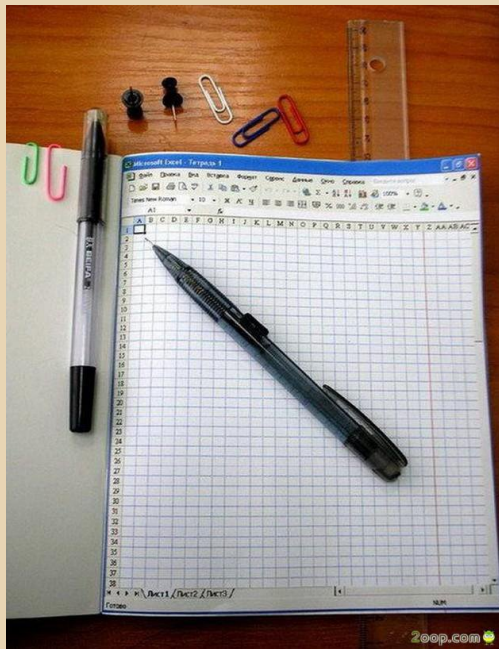
Сообщение GIMP

Изображение PNG Сообщение

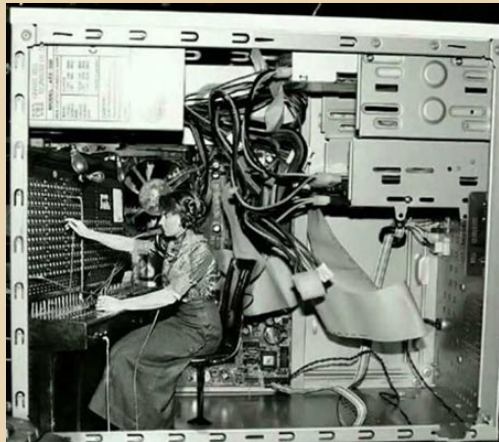
Невозможно сохранить прозрачность, вместо нее будет сохранена непрозрачность.

✓ OK





ДОСПЕХИ АНТИЧНОГО БОГА,
ДАРУЮЩИЕ БЕССМЕРТИЕ



Над номером работали

Дизайн/вёрстка/главный
редактор - uavль0bъ

Редактор - Вячеслав Рытиков (eуьрс)

Помощник редактора - Андрей Шаронов

Авторы:

Андрей Шаронов (Andrei0bъ)

Вячеслав Рытиков (eуьрс)

uavль0bъ

SysTools

Олег Павлов

Андрей Тумилович

Владимир Фёдоров

Александр Мачуговский (Manwe)

Александр Чуклинов

Елена Шаронова

Интервью:

В.Э. Фигурнов

Сайт журнала: <http://dgmag.in>

Раздел журнала на "Железных

Призраках Прошлого":

[http://www.phantom.sannata.ru/
articles/dgmag/](http://www.phantom.sannata.ru/articles/dgmag/)

Группа ВКонтакте: <http://vk.com/dgmag>

E-mail главного редактора:

uavль0bъ [sobaka] mail.ru