

DOWNGRADE



N412023



СОДЕРЖАНИЕ

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

ТЕОРИЯ DOWNGRADE

● Новости, события, комментарии (uav1606, В. Рытиков)-----	4
● Дорога из СССР в США: интервью с Ю.А. Стариковым (Ю. Стариков, uav1606, А. Шаронов)-----	5
● Во власти искусственного интеллекта (М. Бабичев)-----	18
● «Обучает» ретрософт (Андрей Шаронов)-----	24

DOWNGRADE-ЖЕЛЕЗО

● Эволюция Магистра (Валерий Сурженко)-----	31
● Электроника МК-85: 16 бит в советском кармане (Bs0Dd)-----	39
● Скромное начало легенды (Павел Ижевский aka paHan)-----	64
● Многофункциональный контроллер на VLB-шину (А. Шаронов)---	69

DOWNGRADE-СОФТ

● Захватил, увидел, перевёл! (Валерий Сурженко)-----	72
● Picture Man 3.0 для Windows 3.1 (Андрей Шаронов)-----	81
● Naiku OS: на первый взгляд (uav1606)-----	86

ПРОГРАММИРОВАНИЕ

● Расширенный экран Profi: что это такое и как с ним работать. Практикум 002. Вертикальный скроллинг (Тарасов А.Е. aka ТАЕ, Вадим Чертков)-----	91
--	----

ИНТЕРНЕТ И СЕТИ

● Web off-line (Андрей Шаронов)-----	104
● BBS в 2023 году (longhorn_gnu aka Захар)-----	111

СТАРЫЕ ИГРЫ

● Магия крови, или Что делать, когда обычный Diablo уже надоел? (Валерий Сурженко)-----	114
---	-----

ЮМОР И ПРОЧЕЕ

● Просто разный юмор -----	122
● Над номером работали -----	123

От редактора

Приветствую, уважаемые читатели.

Наконец-то удалось доделать этот номер. Материалов в этот раз получилось очень много (тем более, что тема свободная).

Итак, что у нас тут имеется?

Например, очень интересная статья-исследование приставок серии «Репетитор» от **Hippiman**. А ещё очень большая и подробная статья про микрокалькулятор (или даже микрокомпьютер) «Электроника МК-85» от **BsODd**.

Вообще, в этот раз больше всего статей прислали в «Железо» – в кои-то веки.

Кроме того, в номере есть, к примеру, статья **Антиквара** с размышлениями о развитии ИИ. И воспоминания Андрея Шаронова о студенческих компьютерах. Ещё интервью с человеком, который занимался русификацией DOS – Юрием Стариковым.

И ещё много чего – аж глаза разбегаются. :-)

Ещё и два новых автора.

Жаль, что номер задержался, со следующим будет трудновато – нужно до Нового года успеть.

Но где наше не пропадало...

Спасибо всем авторам и вообще людям, помогающим с журналом.

Как всегда, если у вас есть какие-то предложения, пожелания, статьи – присылайте на мой e-mail: uav16060 [круглый такой значок] mail.ru

uav1606

НОВОСТИ, СОБЫТИЯ, КОММЕНТАРИИ

Скончался Джон Уорнок

19 августа в возрасте 82 лет ушёл из жизни один из основателей Adobe Джон Уорнок.

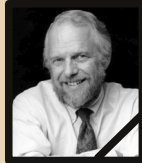
Он очень многое сделал для развития компьютерных наук — например, в 1969-м году разработал алгоритм удаления невидимых поверхностей в компьютерной графике (так и называется — Warnock algorithm).

В 1982 году вместе с Чарльзом Гешке Уорнок основал компанию Adobe, где они совместно разработали технологию PostScript.

В 1991-м году Джон приступил к работе над системой Camelot, позже ставшей форматом PDF.

Лауреат многих IT-премий.

В 2000-е годы много занимался благотворительностью.



слышали его голос — ведь у него на счету более 5000 картин.

Наверное, деятельность Володарского не имеет прямого отношения к тематике журнала, но в каком-то смысле его и других авторов односторонних переводов/озвучек можно считать символом той эпохи — 90-е, видеосалоны, VHS, Dendy, «Спектрум»...

Кстати, Леонид Вениаминович в какой-то степени имеет отношение и к компьютерам — именно он озвучил русскую версию игры 2004 года The Suffering.



30 лет Myst

24 сентября 1993 года был выпущен квест Myst. Его создателями стали братья Робин и Рэдэм Миллеры.



Myst быстро завоевала популярность, было продано несколько миллионов экземпляров, а игра долгое время удерживала статус самой продаваемой (лишь в 2002 году её сменила The Sims).

Квест получил несколько продолжений (последнее вышло в 2010 году), и даже было выпущено несколько романов и комиксов по мотивам игры.

Закрылась студия Volition

В результате реструктуризации Embracer Group закрыла игровую студию Volition. Эта студия — наследник Parallax Software (разработчики Descent), она образовалась в 1996 году в результате разделения Parallax. Volition известна серией космических симуляторов FreeSpace. Одной из последних игр компании стала Saints Row.

Фестиваль «Демодуляция 2023»

12-13 августа в Москве под эгидой «Яндекса» прошёл очередной (уже 5-й по счёту) ретрокомпьютерный фестиваль «Демодуляция».

В его рамках читались доклады (например, об истории платформы MSX или ПЭВМ «Ириша» и др.), проводились конкурсы, была живая музыка, выставка старых компьютеров и многое другое.

Подробнее на официальной странице:

<https://yandex.ru/museum/yaretrocomp>

Видео:

<https://youtube.com/playlist?list=PL6Wui14DvQPzXjSO7U-Ezpv1H3htlxb6E>

<https://youtube.com/watch?v=cTQSy76EPSM>

Умер Леонид Володарский

На 74-м году жизни скончался легендарный переводчик фильмов Леонид Вениаминович Володарский. Если вы смотрели телевизор или видеокассеты в 90-е, то почти наверняка

Обзор составил [uav1606](#)

Новости присылал Вячеслав Рытиков ([eubrc](#))



ДОРОГА ИЗ СССР В США: ИНТЕРВЬЮ С Ю.А. СТАРИКОВЫМ



Вашему вниманию предлагается интервью с Юрием Стáриковым – человеком, много лет проработавшим в сфере ИТ, в частности в компании Microsoft. Он наиболее известен русификацией MS-DOS 4.01 и PC Works. Однако в этом интервью мы постарались более подробно коснуться и других проектов Юрия, так как история русификации DOS уже была подробно описана на сайте <http://rdos401.org>

Расскажите немного о себе – что угодно: как Вас зовут, сколько Вам лет, где родились, какое у Вас образование и так далее. Любую общую информацию, которой хотели бы поделиться с читателями.

Юрий Стариков, родился в Сибири: г. Прокопьевск Кемеровской области. Закончил ММФ НГУ.

Расскажите о том, как Вы познакомились с компьютерами. Когда это произошло, что это была за модель?

У нас в Новосибирском университете у каждого на первом курсе была практика по программированию, где все должны были написать и отладить работающую программу на М-222. Что я, в принципе, и сделал... То есть ты сдаёшь написанную от руки программу, тебе делают распечатку того, что написано, ты смотришь, что там нет ошибок, и прогон – первый прогон, там перфокарты набивают. И получаешь пачку

перфокарт, и получаешь пачку синтаксических ошибок. У меня, например, была одна синтаксическая ошибка. Возможно, это набили так неправильно перфокарту – не я её набивал (я горжусь, что за свою жизнь не набил ни одной перфокарты).

Затем сдавал обратно, те опять «перекомпилировали» и опять её прогоняли. Как сейчас помню, мне надо было найти все корни гамма-функции с точностью до какой-то там одной тысячной.

Программа писалась обычно в то время на Фортране или Алголе.

А саму эту машину Вы видели? Т.е. что она собой представляла? Большая? Сколько места занимала?

Был вычислительный центр Новосибирского университета, это был огромный машинный зал, где стояла эта штука, на которой всё это работало. Я на ней сам ничего не набивал. Я прямо к компьютеру, к терминалу, сел только на 3-м курсе, когда у нас уже базовые институты пошли. У Новосибирского университета была очень шикарная вещь – вы два с половиной года учились, а со второй половины 3-го курса вы распределялись по кафедрам, выбирали базовый институт, где вы два с половиной года продолжали своё обучение. То есть на 5 курсе вы вообще в университет не ходили. Ну и на 4-м курсе уже очень мало. И вы в основном были в этих базовых институтах, где всё и проходили. У меня базовый институт был – новосибирский



филиал Института точной механики и вычислительной техники.

А с какими там компьютерами Вы встречались?

Диплом у меня был по «Эльбрусу».



Самое интересное, что после окончания университета я через 3 года работал уже в Москве, ну т.е. переехал. И у меня научным руководителем был тот человек, который писал техзадание на «Эльбрус». Это Сухомлин Владимир Александрович. Он сейчас профессор ВМК МГУ.

«Эльбрус» тогда ещё не существовал, это была БЭСМ-6 с эмуляцией «Эльбруса», инструментальный комплекс ТЕМП.

Насколько я знаю, Вы учились и в МАИ? Расскажите об этом времени, особенно в плане компьютеров – имели ли Вы там дело с ЭВМ, если да, то с какими?

Когда я переехал в Москву, через год меня рекомендовали в аспирантуру от ЦНПО «Вымпел». Была такая организация, после распада Советского Союза она существовала лет 5, наверное, как межгосударственное научно-производственное объединение «Вымпел». Там была из Белоруссии часть, часть – из Казахстана, частью туда входили большие организации рядом с КБ «Южное». Это была советская

программа ПРО, поэтому она была как бы межгосударственная. Долго она существовала, до примерно 2000-х годов. Поэтому это был не МАИ на самом деле, просто аспирантура ЦНПО «Вымпел» была при МАИ.

Т.е. философия, всякие коммунизмы и всякие истории партии, английский язык – всё это было через МАИ. А всё остальное шло через ЦНПО «Вымпел».

Я был старшим инженером этого ЦНПО, занимался Unix на СМ-4 и СМ-1420, а также внедрял 300 PC-совместимых машин в ЦНПО.

Это были компьютеры фирмы Goldstar, купленные через Венгрию. Процессор 80286, у них было 4 мегабайта памяти, для MS-DOS совершенно ненужные, зато Windows хорошо работал. И половина из них была с геркулесовской видеокартой, а половина была с EGA-видеокартой. Соответственно, половина мониторов были монохромными, половина были цветными.

А из них делали какую-то сеть? Или они просто все отдельно стояли?

Была сеть, маленькая сеть, потому что сетевого оборудования купили мало. Сделана она была на базе оборудования компании 3Com, были разделённые диски и локальная сеть. Причём очень быстрая... И кабели были заказаны. Я помню, у нас там был товарищ очень хороший, но он сидел на 2 этаже выше. Кабели тут же исследовали, замерили, проверили, нашли точно такой же – самое главное, что разъёмы такие оказались у людей. И они сделали этот разъём, и в результате мы соединились с компьютером на 3-м этаже.

Это был не обычный сейчас Ethernet – всё подсоединялось к общему проводу. И ещё там не было даже протоколов TCP/IP и IPX.

Вот такая сеть была, но она мало использовалась.

А что собой представляли упомянутые выше СМ-4, СМ-1420?



После университета я работал в основном на машинах серии СМ-1 – СМ-2, и, переехав в Москву, искал работу как специалист по СМ-1 – СМ-2, ну а там оказалось, что люди перепутали с СМ-3 – СМ-4. ☺ Им казалось, что разницы нет никакой, хотя эта разница кардинальная. Т.е. СМ-3 – СМ-4 – это была архитектура PDP-11 фирмы DEC, а СМ-1 – СМ-2 были созданы на основе Hewlett-Packard 2000. Это совершенно разные архитектуры. Поэтому мне пришлось переучиваться на эти СМ-3 – СМ-4 на PDP-11, но мне повезло в том смысле, что сама организация (ЦНПО «Вымпел») не использовала такие машины, но у них была подшефная школа, где они создали компьютерный класс, куда поставили машину СМ-4, потом даже 1420, где было подключено 16 терминалов «Видеотон VT52». И чтобы подерживать эти терминалы, туда поставили Unix, русскую версию, тогда это была версия МНОС – Машинно-Независимой Операционной Системы, сделанная ИПК Минавтопрома. МНОС, потом и ДЕМОС.

Ещё был заказ от тогдашнего первого отдела создать информационно-справочную систему по всяким документам, может быть, первый безбумажный документооборот. Чтобы была возможность поиска и прочее. Поэтому решено было сделать на основе Unix'a, там тоже были люди-то, ну, достаточно передовых взглядов. Это был 85-й – 86-й год, я делал СУБД СИС. И в 86-м, в сентябре месяце, я получил первую премию на конкурсе молодых учёных-специалистов, как раз в городе Гомель, Белоруссия. Там проходила всесоюзная конференция, тогда ещё, помню хорошо, выходишь на железнодорожный вокзал к гостинице, где мы жили, а там электронная газета, которая писала: «В лес не ходить, грибы не собирать». Был Чернобыль, в которой Гомельская область очень сильно пострадала. Т.е. я помню, возили нас из гостиницы на радиозавод, где все эти слушания проходили, а там висели такие большие яблоки и груши, которые рвать и есть было запрещено.

А что собой представляла эта СУБД, это был какой-то сборник документов или реестр документов?

Да. У СУБД была достаточно оригинальная идея, ни с чего не было содрано, она использовала файловую систему от Unix'a, её иерархию, и дальше, когда вы задавали поиск, она уже тогда использовала хэш-функцию, хэш-функция была своя. Вот честно говорю, тогда ни MD5, ни SHA не было ещё, это всё появилось позднее, но понятие «хэш-функция» уже было. Т.е. вы делали запрос по хэш-функции, отсюда вы получали какое-то значение, которое рассматривалось с помощью того, что называется сейчас «хэш-таблица». Вы тут же находили где искать и что пытаться искать. Вот такая СУБД была сделана.

Кроме этого, там мы делали ещё всякое форматирование текстов по ЕСПД, ЕСКД, всё это уже забыли, но тогда это было очень нужно.

Ну и наконец, тогда же я (просто время было) сделал для Unix'a (а этого не было в русской версии, в МНОС/ДЕМОС) интерпретатор языка BASIC Linter. Который особенно был популярен в школе, где это всё стояло. Всем понравилось, как это работало, ну и много чего было после этого: этот Linter был перенесён на VAX, был перенесён на ЕС ЭВМ и был перенесён на IBM RT PC (ОС AIX), которая была абсолютно голая в Советском Союзе. После того, как этот Бейсик туда был перенесён, стало можно делать всякие научные расчёты на Бейсике на супермощных для того времени компьютерах, там был RISC-процессор. И в конце концов, когда это увидели, меня тут же попросили сделать интерпретатор языка Автокод для IBM RT PC, но я уже сбежал в «Диалог», а там это никому не нужно было.

А скажите, когда Вы говорите, что этот Бейсик был портирован на ЕС ЭВМ, Вы имеет в виду уже линейку, совместимую с IBM PC?



Нет. На обычную ЕС ЭВМ 1030. Для неё был сделан Unix. Причём где это произошло? Я сдавал всё для диссертации в городе Тверь (ну, тогда Калинин), там была организация «Центрпрограммсистем», которая регистрировала все программы. И сдача в «Центрпрограммсистем» такой программы была эквивалентна публикации. А для того чтобы получить кандидата, тебе надо было как минимум иметь шесть публикаций. Я в итоге имел 9, но это другой вопрос. И мы поехали в «Центрпрограммсистем», а у них с Unix'ом (надо ж было показать, что работает) ничего не было, кроме ЕС ЭВМ. Туда поставили операционную систему, по-моему, курчатники. Ну ладно, нам дали этот ЕС ЭВМ на ночь, где-то примерно с 10 вечера до 6 утра, и я за одну ночь этот весь Linter туда перенёс, на ЕС. Смогли прочитать ленту – вернее, мы знали, что ленту будут читать на ЕС ЭВМ, и я её подготовил ещё до отъезда туда – записал ленту на СМ-овском магнитофоне, но с помощью до сих пор используемой в Linux'ах программы tar. Если вы знаете Linux, то там всегда есть tar. И на ЕС ЭВМ с помощью этой ЕС-овской tar я прочитал ленту с СМ-4, и всё было нормально.

У меня была назначена защита где-то на май 1989-го года, но в 88-м меня позвали в «Диалог». ☺

И я не стал защищаться. Хотя было много у меня для этих целей сделано. Вернее, всё было сделано.

Расскажите о СП «Диалог» – как и кем эта организация была основана, чем занималась, как туда попали Вы, в работе над какими проектами участвовали? Как сложилась дальнейшая судьба «Диалога»? В общем, хотелось бы максимально подробно узнать про эту компанию, её историю, про Вашу работу в ней.

СП «Диалог» был основан на заре перестройки КАМАЗом и американской стороной (там был инвестор, что решил вложить некую сумму в предприятие в СССР).

В 88-м году уже появлялись совместные предприятия, появилось и предприятие «Интерквадро». Ну а я в то время делал всё на Си, уже был достаточно известен в некоторых программистских кругах в Москве, и меня позвали в СП «Интерквадро» прочитать лекции по языку Си. У них открыли учебный центр, заплатили по 100 рублей за лекцию, по 50 рублей за семинар. Вообще, деньги для 88-го года бешеные. И там же мне предложили написать книжку, я её написал, вот она у меня в руках – это «Мобильность программ и особенности реализации языка Си». И дали 1000 рублей гонора – деньги такие бешеные в 88-м году.

А потом меня пригласили в «Интерквадро» работать. Но в это время в «Диалоге» тоже решили открыть учебный центр, назначили директором Павла Садчикова, а он позвал меня и предложил стать своим заместителем. Ну ладно, я пошёл в «Диалог», стал там замом.

А вот интересно, почему этим занялся КАМАЗ, они же вроде бы занимались автомобилями?

Нет, просто те американцы, которые хотели создать вот это совместное предприятие, они дали слишком много американских долларов, а тогда 60 копеек доллар стоил, да. ☺

Т.е. понятно, что уже был курс туристический, да и всё равно это было сколько-то миллионов рублей. Которых тогда ни у кого свободных-то не было.

Только КАМАЗ мог выделить какую-то сумму в рублях, чтобы быть партнёром.

От КАМАЗа был генеральный директор «Диалога», он был зам. директора КАМАЗа по внешней торговле. Зрелов, Пётр Зрелов. Ну а как только его сделали генеральным директором «Диалога», он ушёл из КАМАЗа, и он, и его жена Татьяна в основном там всем и заправляли.

Там потом появилась ещё куча всяких мелких соучредителей типа ВЦ Академии наук и т.д.



В общем, в учебном центре «Диалога» я пробыл недолго. Где-то с января по май 89-го года.

А в мае Колю Любовного назначили заниматься локализацией MS-DOS. Это на нашем с Колей RDOS401.ORG всё расписано.

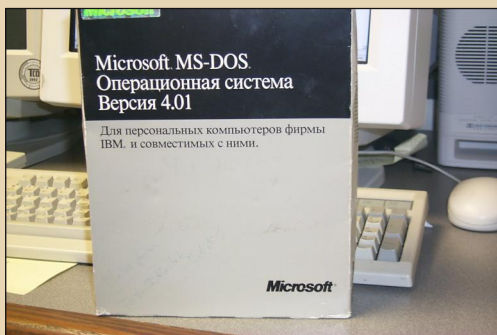
Он позвал меня, потому что тот человек, который отвечал за локализацию, ничего не делал.

И Коля Любовный создал всю команду, я там отвечал как раз за локализацию. Там был Петя Квитек, я, Андрей Давыдов, Дима Бондаренко, Коля Любовный и ещё был Коля Саух – человек, который делал, кстати, русский Unix (ДЕМОС), они получили как раз в этом году, в 89-м, премию совета министров СССР за русскую версию Unix'a, которую они, пардон, своровали ☺ и, так сказать, нелегально локализовали. Но т.к. украли её в исходном коде, то, естественно, исправить все сообщения было легко. Ну или не легко – надо было перекомпилировать ещё на русские компьютеры. Для русского VAX'a CM 1700 она работала лучше, чем для CM-4.

Вот эта команда и делала русский DOS и русский Works – между СП «Диалог» и Microsoft было подписано соглашение на эти два продукта. Ну и где-то уже в августе 89-го года мы получили первый билд русского DOS'a.



Юрий Стариков с коробкой MS-DOS 4.01, подписанной всеми участниками проекта



А сложно было переводить какие-то специфические термины? Может, были какие-то курьёзные случаи?

Курьёзный случай – в английском языке есть выражение choose your selection. И choose, и selection – это по-русски одно и то же слово – «выбор». Choose your selection – «выберите ваш выбор».

Это было очень как бы, ну, смешно. «Сделайте ваш выбор» потом перевели. Вот уже после первого MS-DOS и Works был глоссарий. Уже было гораздо проще делать остальные продукты.

Вообще говоря, не стоит забывать, что русский DOS, по идее, даже для Microsoft'a была первая операционная система, где вам надо поддерживать как кириллицу, так и латиницу. Т.е. это не чисто ASCII – 128 символов, а все 256. Это первое. Второе – здесь надо поддерживать



Стоят (слева направо): Юрий Стариков, Пётр Квитек, Пол Робсон, Бил Гейтс, Николай Любовный, Дмитрий Бондаренко, Андрей Давыдов



переключение клавиатур. И это было сложно-стью.

В EGA используется ASCII, ну, максимум – расширенный ASCII, т.е. Latin-1 так называемый. «Юникода» в то время не было. Поэтому надо было загружать фонты, и не все видеокарты это поддерживали. Например, там, где я работал, в ЦНПО «Вымпел, купили 250 или 300 PC, которые я помогал распределять по местам, и там была видеокарта Hercules, которая не позволяла загружать фонты, поэтому были катастрофические сложности с русскими буквами.

А всё находится в Советском Союзе. Весь язык Автокод 76 – он на русском. Это была достаточно большая проблема.

Даже графические самые дешёвые CGA не работали с русскими буквами. Женя Веселов, чтобы сделать свой «Лексикон», работал чисто в графическом режиме, т.е. у него была своя поддержка фонтов для русского режима. В графическом режиме. Чего нельзя было сделать в Hercules, т.е. только для CGA он это мог делать. А для EGA можно было загрузить уже фонты – русские буквы. Ну и для VGA, но самой популярной была EGA. Все, кто привозил PC, знали, что надо привозить IBM PC или совместимый с EGA, чтобы загрузить русские буквы. Антон Чижов делал для этого программу, я там что-то делал, Петя Квитек делал что-то – всё, чтобы можно было загрузить русские буквы. А ещё тогда была поговорка: Баба с Егой. Не Баба Яга, а Баба с Егой. © Имеется в виду, что у девушки есть компьютер с EGA.

Это было весёлое время, когда можно было писать какие-то простенькие программы (ну т.е. как простенькие, тот же «Лексикон», я думаю, содержал достаточно много строк кода).

В «Диалоге» было интересно. Я, например, через «Диалог» продавал все свои программы, самым популярным был редактор под названием Yed. Он работал в текстовом режиме, всё было по-русски. Yed – это было, наверное, то, что мне приносило где-то в 89-м основные деньги. Т.е. продажи этого ПО приносили деньги.



Вот такие вот были вещи.

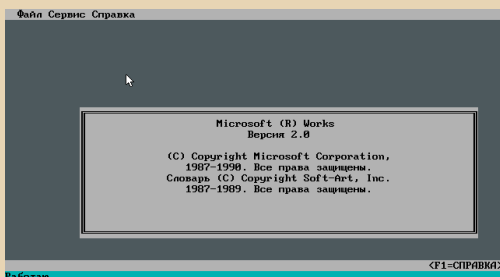
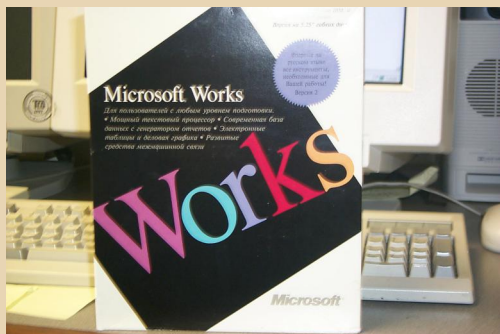
А сколько времени заняла русификация MS-DOS, примерно?

Ну, если мы начали серьёзно с мая 89-го года, то где-то за неделю до 12 апреля или 12 апреля, уж точно не помню, уже была презентация. Первый билд был сделан где-то в августе, потом допереводили всё остальное, т.е. документацию, в ноябре Пол Робсон приехал в Москву за документацией, документация не была готова, всё уже стояло в типографии, чтобы что-то делать. И там были 3 страшных дня, когда всё это было сделано. Ну, в RDOS401.ORG это всё расписано.

А Works Вы, соответственно, русифицировали после?

Где-то в ноябре мы закончили, а потом где-то с декабря начали реально заниматься Works'ом. Но т.к. RDOS был сделан, самое главное было сделано. Все глоссарии было сделать гораздо проще. Перевод документации, уже имея глоссарии и русский DOS, было сделать гораздо-гораздо легче и быстрее. Ну и, опять же, команда начала расширяться, пришли потом очень известные люди, недавно мы с ними всеми встречались в Сиэтле. Т.е. с большинством. Федя Зубанов пришёл, Катя Лаженева.





Works – это приложение, и задача Microsoft была сделать, чтобы Works работал не только в русском DOS'e, но и в любом другом. Например, MS-DOS 3.3 загружаешь, запускаешь русский PC Works – у тебя русские буквы есть. И поэтому там была сделана TSR-программа (я им, так сказать, подарил, до сих пор жалею, денег не заплатили ни копейки), которая загружала фонты EGA, VGA и делала русскую клавиатуру. Там тоже в RDOS401.ORG написано всё это было.

Я поехал в Штаты.

Да, когда я приехал из Америки и сказал, что меня позвали в Microsoft, меня вызвали на партсоборание – о том, что я предаю родину. ☺ Хотя я был беспартийный.

Это всё благополучно закончилось? Никаких мер к Вам не применили?

Ну, практически меня выперли из «Диалога», т.е. я ушёл в отпуск, имел право на 2 месяца, а после этого я уже не вернулся в СП «Диалог».

Вы, насколько я понял, являетесь автором как минимум одной книги? Расскажите про неё, ну и про другие Ваши публикации.

Да, «Мобильность программ и особенности реализации языка Си» – насколько я знаю, это первая книга по языку Си русского автора. Первое издание в 1989 году в СП «Интерквадро», второе – МГУ (насколько знаю, по ней читали спецкурс на ВМК), третье («дополненное и исправленное», хотя там нет ни единого исправления или дополнения) – пиратское, издательство «Память», 1992.

В «Диалоге» издавал и был главным автором «Диалог Экспресс». Такой маленький журнальчик на компьютерную тему, в основном о продуктах Microsoft. Один из выпусков посвящён выходу русской DOS 4.01. Всего я успел сделать 5 или 6 выпусков, они раз в месяц выходили. Но потом я поехал в Америку и к этому делу уже не вернулся.



СП
Диалог Экспресс
Официальный представитель в СССР фирмы Microsoft®

№3 Март 1990 Информационный бюллетень отдела дистрибуции СП Диалог

Этот номер нашего бюллетеня полностью посвящен наиболее популярным вопросам программирования высшего уровня - Бейсик, Си, Паскаль.

В этом сборнике используются переведенные на русский язык статьи, а также другая информация, распространяемая пресс-службой корпорации Microsoft. Публикуемые материалы интересны тем, что они не только содержат сообщения о новых версиях компиляторов, но и затрагивают некоторые проблемы, интересующие программистов в СССР - такие, как технологии разработки программного продукта, тестирование программ, а также популярность некоторых языков.

Мы надеемся, что информация, приведенная в этом бюллетене, будет полезна всем, кто занимается программированием на ПЭВМ.

Бейсик: мифы и реальность

Microsoft, 22 июля 1989 г.

Миф: Бейсиком пользуются программисты, работающие только на себе или на небольших компаниях, старьевщики, служащие общественных организаций, но не профессионалы.

Факт: Согласно исследованию, проведенному Microsoft, 62% программирующей на Бейсике пишу программы для использования другими людьми, а 50% пользователей Microsoft® QuickBASIC® для продажи. В ходе 25-летней истории Бейсика приобрел все свойства наиболее совершенных профессиональных языков программирования, включая циклы DO WHILE/UNTIL, типы данных, определяемые пользователем, рекурсию и длинные строки.

Миф: С появлением более совершенных языков, таких как Си, число людей, использующих Бейсик, при

программировании на персональных компьютерах постоянно падает.

Факт: Доступ к Бейсику имеет большее число людей по сравнению с любым другим языком, самоотлично изучило Бейсик также наибольшее число людей. Простота использования и изучения всегда будут цениться пользователями, и в этой области Бейсик не имеет себе равных ни один из основных языков. Известная способность Бейсика поддерживать сложные и мощные средства, сокращая фундаментальную работу и используя, позволяют ему пережить многие другие языки. Несмотря на то, что, в отличие от профессиональных программистов, Паскаль и Си за последние годы сильно потеснили остальные языки на рынке программных средств, Бейсик прочно удерживает свои позиции.



СП Диалог Экспресс

Official distributor of Microsoft® Corp. in USSR

№ 4 April 1990

Information bulletin of the distribution department of JV Dialogue

Special issue

This issue of our bulletin is dedicated to one of the most important events in Soviet personal computers - the official announcement of Russian MS-DOS.

This operating system is standard for IBM family personal computers and compatibles. It was developed by Microsoft in 1981 and presently is installed in more than 40,000,000 computers. There are lots of localized versions for different countries: France, Germany, and others. Today we get a Russian version.

Articles in this issue are about the most important advantages and capabilities of Russian MS-DOS. Also you can find information about Microsoft Corporation and the people who designed this localized version.

Materials Microsoft Corporation and articles specially written by people from Microsoft, MPI and Dialogue are used in this issue.

Microsoft MS-DOS Version 4.01

Operating System for IBM Personal Computers and 100% Compatibles

Product Overview

MS-DOS operating system version 4.01 brings significant improvements to the MS-DOS environment in several key areas, including usability, large-disk support, system performance, and expanded memory application (EMS) 4.0 support. The MS-DOS operating system 4.01 also features the introduction of several new commands, such as Mem and Install, and numerous enhancements to existing commands, including Format, Delete, and Graphics, as well as full support for 3.5-inch disk drives.

In the localized version MS-DOS for the USSR, all messages output in Russian language. In Russian MS-DOS documentation

MS-DOS Shell: Easy-to-use, Presentation Manager-like interface

are included some special features. The names for the files and directories may be in Russian.

The code page for Russian MS-DOS is based on the alternate code page, which include Ukrainian and Belorussian letters. IBM has assigned for this code page number 865. Western companies, for example, Hewlett-Packard, Epson, Ashton-Tate, etc. will support this code page.

Russian MS-DOS has three means for RUB/LAT switching and two keyboard layouts.

All popular programs, operating in text or graphics mode (Microsoft Word, PC Works, MS Chart, etc.), may use Cyrillic characters.

Technical Highlights

The MS-DOS Shell is a full-screen, menu-based file management and applications organizer utility whose look closely resembles

Как началась Ваша карьера в Microsoft?

Мне предложили там работать. Но работать было нельзя в Штатах. ☹ Потому что тогда ещё действовали ограничения СОСОМ, и я не имел права использовать сеть, это были санкции. И не мог работать с 386-м процессором. А тогда уже все компьютеры были 386. Не 16-битные, а 32-битные.

Это был 90-й год. И был один швед, когда меня уже звали в Штаты, адвокаты ему сказали, что этого сделать нельзя. Но он вспомнил, что они ещё открывали в Ирландии локализационный центр, а Ирландия – не член НАТО. Значит, там санкции не будут работать. Но он ошибся, там было всё то же самое. Но, во всяком случае, я поехал в Ирландию, и там был не только я, потом приехал чех Мартин, потом приехал Габор Варга из Венгрии, приехала девушка Ева Орловски – она полячка, но у неё было гражданство шведское, поэтому её не ограничивали. А вот Габор, Мартин и я, мы были, так сказать, санкционные

Тогда у нас был юмор такой, компьютерный: я сообразил, что если я сделаю локальную сеть с Windows 3.1 for Workgroups, то никто её

видеть из Штатов или из Ирландии не будет, кроме нас. И мы назвали сервер «Штирлиц» и к нему подсоединились. ☺

Скажите, пожалуйста, а с русификацией Microsoft Office Вы никак не были связаны? С проектом «РусскоеСлово» от «Параграфа»?

«РусскоеСлово» – это изначально была просто TSR-программа. (Я, кстати, когда после «Диалога» уезжал, тогда же, в 90-м году, нельзя было просто так взять купить билет и уехать куда-нибудь в Ирландию – нет. Я был отправлен в служебную командировку от СП «Параграф». ☹ В Ирландию. Где я до сих пор и нахожусь легально.) Это была TSR-программа, которая подменяла ресурсы Word'a. Word уже имел ресурсы, даже для текстового режима. У Microsoft для текстового режима была такая библиотека, называлась она сокращённо COW, корова. Character Oriented Windows. Т.е. практически это как бы API Windows, но в текстовом режиме. Works был сделан, Word был сделан на основе этого, Multiplan, ещё что-то. Исходные тексты COW, библиотеки, находились в каталоге, который назывался BEEF. Beef – это, понятно, говядина, говяжье мясо. ☺

А первый «Офис» уже локализовали Катя Лаженцева, Игорь Агамирзян, в то время Кви́тек, по-моему, до отъезда в Штаты этим тоже занимался, но я к этому делу уже не был причастен.

Для меня локализация закончилась... так, 3 декабря мы летели, значит, 2 декабря 1992 года. Всё. Дальше я уже локализацией занимался, только если нужно было помочь в чём-то.

Отдельно хотел спросить о пакете Alphabet Plus – что он собой представлял, кто его разрабатывал, как это происходило и т.д.?

Это было в Ирландии, после того, как русский DOS сделали. Вся восточная Европа была под эгидой Microsoft в Германии. Чехословакия,



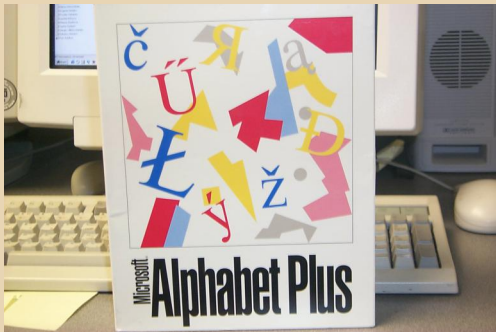
Польша, Венгрия, ну, Восточная Германия, она, понятно, покрывалась. Были ещё Болгария, Румыния, куча других, 7 стран. Что с ними делать? Прежде всего интересовала Чехословакия, Польша и Венгрия. Тогда Пол Робсон приезжал периодически в Ирландию, и мы с ним как-то обсудили: а что, давай возьмём часть из русского DOS'a, допишем там фонты, клавиатуру – венгерскую, польскую, чешскую. И сделаем набор драйверов. Т.е. когда запускаете эту штуку, у вас все эти вещи будут подключаться – и любая версия MS-DOS тут же будет иметь польские, чешские и т.д. буквы, но будет интерфэйс на английском. Т.е. выдавать все сообщения она будет на английском, но зато можно было писать по-польски, по-чешски, по-венгерски. Так родился Alphabet Plus.

Т.е. он представлял собой пакет драйверов, фонтов для всех версий MS-DOS (от 3.3 до 5.0) – все драйвера (**display.sys**, **keyb.com**, **msprint.sys**...) были взяты из русской MS-DOS 5.0, обратно локализованы с русского на английский и скомпилированы для всех поддерживаемых версий MS-DOS. Была также общая установочная программа.

Поддерживались следующие языки: английский, русский, чешский, венгерский, польский, словацкий, болгарский (это 866 таблица – она поддерживала болгарский алфавит). Югославский ещё поддерживался – Yugoslavian Cyrillic.

Это был первый продукт, разработанный не в Редмонде, а в Ирландии, ну и Microsoft в Германии его успешно продавала.

У меня есть экземпляр на полке.



Но кроме этого, ещё был сделан Central and Eastern Europe PC Works. То же самое, что английский PC Works 2.0, но к нему была TSR-ная программа на основе Russian Works. Если мы говорили: я хочу Hungarian, то загружались венгерские буквы и делалась соответствующая клавиатура через TSR-программу.

У нас в журнале когда-то было интервью с П.В. Бух-Винером, где он рассказывал, что это именно он в 1983 году разработал так называемую альтернативную кодовую таблицу. А Вы не были с ним знакомы? Ну или, может быть, слышали что-то? (Он в то время работал в ИВНДИНФ АН СССР.)

Очень интересный вопрос... Я всегда думал, что и основную, и альтернативную сделали в ВЦ АН СССР. После вашей информации я задал сей вопрос Антону Чижову (автору программ «Альфа» и «Бета» – самых популярных русификаторов MS-DOS) и Жене Веселову (автор «Лексикона»). Антон ответил, что основную сделали в ВЦ АН, и первые русификаторы работали с ней, но Norton Commander выглядел ужасно, и года через два появилась и стала набирать популярность «альтернативная» кодировка. Кто её делал, он не знает...

Женя Веселов сказал, что он всегда думал, что все сделали в ВЦ...

Можете рассказать, что Вы дальше делали в Microsoft?

До 92-го года, до 3 декабря, я занимался в основном локализацией. Был локализационным инженером – т.е. я делал ещё шведский DOS. Тоже там очень много расширенных символов, клавиатуры и пр., я взял часть из русского DOS'a и там всё это использовал.

О работе в Ирландии подробно смотрите <http://www.RDOS401.org>. Могу сказать только, что Ирландия в 1990-1992 году была одной из самых слаборазвитых стран ЕС и вливание денег из hi-tech компаний привели к скачку



благополучия уже после нашего переезда. К тому же Шенгена ещё не было, и то, что Ирландия — это остров, чувствуется быстро.

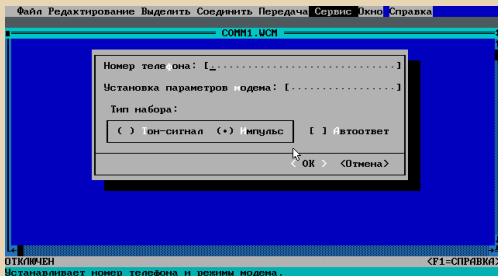
И в августе 92-го года, когда были сняты ограничения СОСОМ, меня позвали в Штаты. Всё затянулось на 3 месяца. Сентябрь, октябрь, ноябрь. Потому что надо было делать визу H-1B, тогда это вообще было понятие достаточно абстрактное. И в результате, когда мне надо было уже ехать, виза была готова, International Product Group, где я должен был работать, была расформирована. Т.к. визу сделали, мне было предложено на выбор, куда я хочу. Ну, я пошёл в Windows NT, в команду под руководством Дэвида Катлера. Я пошёл к тем, кто занимался в Windows NT эмуляцией DOS и 16-битного Windows. Т.к. у меня был достаточно большой опыт с исходными текстами и прочим.

Расскажите про Вашу работу над NTVDM.

NTVDM – NT Virtual DOS Machine. Это означает, что любая программа для MS-DOS, запущенная в NT, должна работать. Работать как часы. В основном там люди, конечно, много работали над Windows-программами, а моей областью была эмуляция MS-DOS. В Windows NT нет MS-DOS, там есть эмуляция MS-DOS. И, например, мне надо было эмулировать все прерывания DOS, чтобы DOS-приложения работали. И работали правильно, и не было бы никаких сбоев. Но, вообще говоря, по поводу некоторых вещей я потом за голову хватался. Например, многие DOS-программы по окончании работы перезагружали машину. Там есть прерывание, которое надо вызывать, мы его отлавливали, говорили, что надо перезагрузить вашу DOS-машину, ничего страшного, Windows не надо перезагружать. А некоторые программы, причём очень известные – от Питера Нортона, они после сообщения о том, что надо перезагрузить, делали переход по несуществующему адресу. Мы-то этого делать не можем, поэтому... В общем, короче говоря, это была

песня, но в итоге IBM тестировала DOS версии 5.5 (5.5 – это та самая DOS-эмуляция, которая была сделана в Windows NT) и сама потом сказала, что Windows NT более совместима с MS-DOS/PC-DOS 5.0, чем MS-DOS 6.0. ☺

Правда, после Windows NT я перешёл в PC Works, к старым друзьям, которые меня ещё помнили по локализации 90-го года. И там проработал dev lead'ом до 96-го – 97-го года, я уже не помню, давно это было. Т.е. там мне дали под контроль коммуникационный модуль PC Works.



Коммуникации – это был чисто мой модуль. В нём можно было создать соединение, по телефону позвонить и прочее.

На самом деле он был мало, конечно, в России использован. Но очень популярен в Японии, где PC Works была главной программой, и там его очень активно использовали для их сетевых интерфейсов через телефонный модем.

В результате, когда мы делали им японскую версию, у меня было право звонить напрямую в Японию, по японским номерам, и Microsoft оплачивала long distance. Я пользовался этой привилегией, и периодически звонил в Россию. ☺ Потому что бесплатно было. Тогда были очень большие деньги за long distance, ну, могло доходить до 2-3, даже 4-х долларов в минуту. Это дорого было для всех.

Кстати, когда уже PC Works 3 вышел, там всё это оставалось, а вот на следующий Works я предложил выкинуть этот коммуникационный модуль, никому не нужный, и сделать туда вместо него e-mail.



E-mail и вот этот коммуникационный модуль – это как бы начало, когда начали это всё делать. Но потом, когда вышел Internet Explorer, там был Internet Mail and News, невозможная в использовании программа. А PC Works был рассчитан на начинающих, т.е. это было для «простых смертных». Поэтому тогда объединили людей из Works'a и из Internet Mail and News, и это стало называться Outlook Express. Outlook Express реально вышел, по-моему, с Internet Explorer 4.

Насколько я знаю, Вы также портировали Works с 16-битной на 32-битную Windows?

Да, да. Последний PC Works на тот момент был 16-битный. Но когда вышла Windows 95 (а Windows 95 – это 32-битная ОС), нужно было, чтобы он работал в native mode, в 32-битном. Это была большая проблема, надо было портировать PC Works с 16 бит в 32 бита. И т.к. я как раз пришёл в это время «под флагом» из Windows NT (а это как раз была первая 32-битная операционная система Microsoft), то мне кроме этого коммуникационного модуля дали ядро Works, которое надо было всё перевести в 32 бита. Соответственно, во-первых, сменить полностью build-систему. Мы взяли build-систему из Windows NT, т.е. ту же, что использовалась Windows NT, взяли и сюда перевели, без всяких изменений. Поэтому сразу начали делать native 32-битный PC Works. Вот это было сделано очень хорошо – ну, с моей точки зрения.

Windows 95 изначально планировалось выпустить где-то в марте. В марте 95-го года. Реально его выпустили в конце июня. В то время PC Works была самая продаваемая программа Microsoft. Лучше «Офиса». Она была главной, её ставили многие OEM-производители компьютеров, Windows и Works ставили. Поэтому PC Works должен был быть готов к выпуску Windows 95.

То есть мы должны были сделать всё – мы сделали. Windows – нет. ☹

И три месяца у нас было, всё это время было просто-напросто внутреннее тестирование.

Т.е. Вы сделали 32-битный Works раньше, чем была презентована Windows 95?

Да. ☺ Но самое страшное произошло в июле: вдруг выясняется, что мышка с Windows 95 и Works'ом 3.0 (старым, 16-битным) в некоторых ситуациях работает не так. И что? Кто виноват? Почему это не было найдено раньше? Ой...

Короче говоря, меня заставили отлаживать, потому что я отвечал за всё это дело.

И я нашёл, что Windows 95, первая версия, меняет местами wParam с lParam в вызовах Windows при запуске 16-битных программ, когда запускает 16-битный Works. А это означает, что Works 3-й, 2-й неправильно бы работали в Windows 95. И что делать? Всё это было отправлено, в Windows было сделано исправление, короче говоря, первый патч Windows 95 был по этому поводу. Т.е. никто не стал ничего перевыпускать, т.к. если ставили Works 4 и Windows 95 – всё работало прекрасно.

Ещё Вы разрабатывали конвертер для Works'a (между RTF и Works)?

Да, т.к. в коммуникационном модуле было не так много работы, мне ещё дали сделать, чтобы Works мог открывать Word-файл, а Word мог открывать Works-файлы. Так как формат абсолютно разный, нужен был конвертер, что я и сделал достаточно успешно. Он (binary-файл) без всяких изменений использовался до «Офиса», по-моему, 2003.

Т.е. уже и Works прекратил существование, а этот файл-конвертер всё ещё использовался.

Кстати, как Вы думаете, почему вообще Works потерял популярность? Сейчас практически мало кто про него знает...



Всё очень просто: у вас есть программа за \$49, у которой есть аналог Excel, у которой есть аналог Word, ну и какое-то подобие Access'а есть, и всё это стоит 49 долларов, в отличие от Word'а, который тогда стоил, например, 299 долларов. Или Excel 299 долларов...

Потому что по математическим формулам Works'овская таблица имела все те же самые возможности. Там не было многотабных таблиц. Ну хорошо, сделайте несколько файлов – и всё. Т.е. тот же самый OLE был, вы могли бы вставить таблицу в таблицу или таблицу в Word. И это всё стоило 49 долларов. Тогда была очень большая конкуренция от WordPerfect, у них был WordPerfect Office, который купил тогда Borland, потом Corel... Короче говоря, Works с объединением Home Essential стоили вместе дешевле, чем этот WordPerfect, и он спас Microsoft от нашествия WordPerfect'а. Вот этот Home Essential задавил WordPerfect, и в результате тот исчез.

Когда исчезли конкуренты у «Офиса», Works стал главным конкурентом «Офиса». ☺

Ему приказано было долго жить.

При этом для «Макинтоша» он долго существовал ещё, года три после того, как он исчез для Windows.

Можете рассказать, чем Вы занимались после? Где-то с 96-го до начала 2000-х.

С 96-го года, когда объединили Internet Mail and News и вот этот коммуникационный модуль, я стал работать в команде Internet Explorer, вот с этим e-mail'ом. Т.е. сначала я там просто отвечал за контакты и S/MIME (S/MIME – это Secure / Multipurpose Internet Mail Extensions, безопасная почта). А где-то уже с Windows XP я стал отвечать за весь Outlook Express, потому что там руководитель ушёл, ушла ещё куча людей, и я стал отвечать за весь Outlook Express. Я выпустил версии 5.0, 5.5 и 6.0. Версия 6.0 была в Windows XP. И так я работал до тех пор, примерно, когда вышел XP, вышел Internet Explorer 6. Ну, короче говоря, больше конкурентов не осталось, Microsoft, как

всегда недалёкий, решил всех сократить. Но, вернее, тогда был ещё один враг – America Online. Всех бросили на войну с America Online, я тоже туда был брошен. Так был сделан MSN.

Я первый раз ушёл из Microsoft'а в 2004 году, решил открыть свой бизнес, связался с Россией, больше не хочу. ☺ На сто тысяч долларов обманули. После этого я туда не хочу лезть. В 2007-м я вернулся в Microsoft, работал в Windows 7, в Windows Media Center. Т.е. отвечал за DRM – Digital Right Management. Это означает, что когда вы проигрываете охраняемую авторским правом музыку, фильмы в Windows, вы её защищайте, чтобы никто не мог копировать. Я занимался в том числе и телепрограммами: вот если у вас есть TV-тюнер и по HBO идёт какая-нибудь «Игра престолов», то её можно записать, но проигрывать только на своём компьютере. Если вы файл передадите своему другу, то там ничего не будет проигрываться. Это всё ещё идёт с того времени, когда я в Outlook Express занимался S/MIME. Я тогда познакомился с компанией, которая называлась Splugus, которая в то время была ведущим производителем криптографического оборудования в Штатах и занималась криптографией, как раз защищённой почтой. Потом, когда я открыл свой бизнес, мы с ними делали некоторые проекты вместе. Мы сделали exchange security review, такое понятие было. В общем, в 2009-м они меня позвали как архитектора в эту компанию, где я проработал с 2009-го по 21-й год. В 21-м году Splugus продались другой компании, которую я хорошо знал, и я решил туда не идти, а вышел, будем говорить так, на пенсию.

Чем Вы занимаетесь сейчас? Как Вы вообще сейчас живёте, если ли у Вас семья? Есть ли у Вас хобби, чем вообще интересуетесь помимо ИТ?

Я выполняю контракты (например, был большой исследовательский контракт с Amazon) и путешествую. Купили RV, дом на колёсах, и вот я на нём много путешествую.



Вот, к примеру, мы уехали 2 июня, доехали на Ньюфаундленда, потом Лабрадор, и вернулись опять на самый юг США, во Флориду.

А в прошлом году мы ездили через все США на запад.

Мы переехали из Сиэтла в Miami Area в 2015-м, потому что дети все разъехались, они все переехали на восточное побережье США, а мы хотели быть поближе к ним. Поэтому мы переехали в Лодердейл-бай-зе-Си, это прибрежная часть Форт-Лодердейла, а Форт-Лодердейл сам является частью Great Miami Area. Майами от нас – 20 миль. Т.е. даунтаун Майами от нас недалеко – 20 миль. Может быть, 25 – Уэст-Палм-Бич. Ну а Форт-Лодердейл – это вот, за окном. ☺ Это три главных города Южной Флориды.

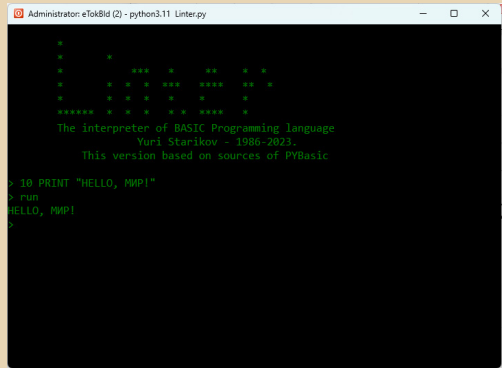
Здесь у нас дворовый пляж, где коралловый риф живой, я периодически там нырял с подводным ружьём и маской.

Но это, конечно, очень сильно пенсионный штат. Но зато из-за этого хорошая медицина, но очень, конечно, дорого по сравнению с остальной частью Соединённых Штатов.

Слышали ли Вы о Downgrade не в узком смысле (как об откате на предыдущие версии ПО), а в более широком – как об использовании старых компьютеров, программ и т.д.? (Наиболее близкий синоним – retrocomputing.) Может быть, бываете на каких-то сайтах по этой тематике? Нет ли у Вас ностальгии по тем временам, когда компьютеры были большими? ☺

У меня есть виртуальная машина с русским MS-DOS 4.01, русским PC Works 2.0 и русской Windows 3.1. Иногда можно «поностальгировать».

Ещё сейчас практически восстановил Linter (см. выше). Переписал на Python – могу запустить везде, где есть Python. Все программы на Бейсике с ОС РАФОС работают («Посадка на Луну», «Утка» и т.п.). Если интересно, смотрите <https://github.com/StarikovY/Linter>.



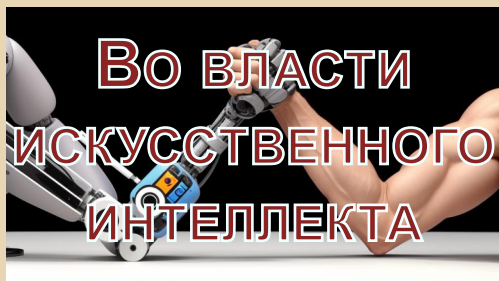
Большое спасибо за ответы! Узнал много нового и интересного о тех временах.

Удачи Вам и новых впечатлений в Ваших путешествиях!

**На вопросы отвечал:
Ю.А. Стариков**

**Вопросы задавали:
uav1606
Андрей Шаронов (Andrei88)**





В метрологии, в системе СИ, есть кратные и дольные приставки. Мы все привыкли ими пользоваться, они позволяют легко выражать и малые, и большие величины. В обыденной жизни широко используются такие кратные приставки, как кило, мега. Килограмм, километр, мегагерц, мегаватт... Более крупные приставки обычно применяются в науке и технике: гига, тера... Но есть и такие гиганты – пета (10^{15}), экса (10^{18}), зетта (10^{21}), которые не встретишь почти нигде. Кроме... информатики!

Никого сейчас не удивишь жёстким диском ёмкостью в 10 терабайт. А на один SSD (пусть и сверхдорогой) можно записать 100 терабайт! *«В 2018 году компания Nimbus Data анонсировала первый в мире твердотельный накопитель ExaDrive DC100 в 3,5-дюймовом форм-факторе и объёмом 100 ТБ. Теперь названа шокирующая цена этого SSD... Nimbus Data предлагает 100-ТБ версию SSD за необычайно высокую цену \$40000».*

<https://www.ferra.ru/news/computers/nazvana-shokiruyushaya-cena-pervogo-v-mire-ssd-na-100-tb-11-07-2020.htm>

Ёмкость же файловых хранилищ или распределённых файлообменных сетей измеряется нынче петабайтами и даже эксабайтами. А Интернет в целом? *«По данным аналитической компании IDC, объём данных, содержащихся на серверах интернета в 2018 году, составлял 33 зеттабайт... По состоянию на*

2022 год в Сети должно быть примерно 90 Збайт данных, причём половина из них содержится в облачных хранилищах».

<https://www.tadviser.ru/index.php/Статья:Интернет>

А с чего всё начиналось? Давайте вспомним времена, не столь отдалённые от нас. Ещё живы многие люди, заставшие ту славную эпоху если не студентами, то школьниками. 60-е годы. Большие ЭВМ на лампах. Например, М-20. В качестве носителей информации в них использовались перфокарты. «Поздняя» перфокарта содержала 80 столбцов по 12 строк, то есть 960 ячеек. В каждой ячейке могло быть пробито отверстие (1) либо не пробито (0). 10 строк, соответствовавших цифрам от 0 до 9, плюс 2 «зональные» строки. Если размещать один байт в одном столбце, получим ёмкость 80 байт. Если использовать все служебные строки для хранения данных, на карту можно «затолкать» 120 байт. Подробно о перфокартах – здесь:

<https://habr.com/ru/articles/426499/>

Для простоты будем считать, что перфокарта вмещает 100 байт. Потому что в 70-е годы компьютерщики охотно измеряли ёмкость новых накопителей информации «в перфокартах». Давайте и мы вслед за ними прикинем:

«8-дюймовые диски увидели свет в 1967 году. В открытой продаже появились в 1971 году. Поначалу их ёмкость составляла 81,6 килобайт. На дискету помещалось столько же данных, сколько на 1000 перфокарт. Когда появились новые 8-дюймовые диски объёмом в 240 килобайт, казалось, что это предел совершенства. Шутка ли, на дискету помещалось столько же данных, сколько на 3000 перфокарт!»

<https://habr.com/ru/articles/427417/>



И дискета, и перфокарта показаны на фото:



Как видим, уже в те далёкие годы, в эпоху молодости рок-н-ролла, битлов и хиппи, прогресс в компьютерной области далеко опережал прогресс в других областях науки и техники. Шутка ли – за 10-20 лет улучшить какой-то показатель (ёмкость накопителя) в тысячи раз!

После эпохи хард-рока прошло ещё 40 лет... И вот, имеем накопитель информации в виде флешки ёмкостью 128 ГБ (также есть на фото). Ничего особенного, заурядный недорогой накопитель. Можно взять такую же карту памяти, она ещё компактней. Во много раз компактней и перфокарты, и дискеты! А вмещает (округлённо) $128 \cdot 10^9$ байт. Чуть больше 0.1 ТБ! Много это или мало?

А смотря с чем сравнивать. Если с дискетами – то это примерно 530 тысяч дискет 8". Или – больше 1 миллиарда перфокарт!!! Что это такое?

Чувствительными весами я нашёл вес одной перфокарты: примерно 2.4 г.

Толщина перфокарты – около 0.2 мм. Нетрудно посчитать, что одна флешка заменяет колоду перфокарт высотой $0.2 \cdot 10^{-3} \cdot 10^9 = 2 \cdot 10^5 \text{ м} = 200 \text{ км}$! (конечно, без учёта уплотнения нижних перфокарт огромным весом верхних). Эта колода весила бы $2.4 \cdot 10^{-3} \cdot 10^9 \text{ кг} = 2400 \text{ тонн}$!

Колода перфокарт, уходящая в космос, в 10 раз превосходящая по высоте Олимпус Монс (высочайшую гору Солнечной системы, находящуюся на Марсе) и требующая для своей перевозки 240 грузовиков – это нечто невообразимое. Но это – просто результат 70 лет компьютерного прогресса, «засунувшего» всю эту гору бумаги в один брелочек.

Это длинное вступление кажется серьёзным отклонением от основной темы. Но оно необходимо, чтобы показать всю мощь компьютерного прогресса, всю силу, с которой этот прогресс владеет (и владеет заслуженно) человеческими умами. Это «золотой телец», которому истово поклоняются многие миллионы айтишников и просто современных молодых людей.

А теперь мы должны свергнуть этого тельца со своего пьедестала. Представляете всю сложность и неблагодарность этого дела? Но увы, им приходится заниматься, и вовсе не из комплекса Герострата, а ради банального выживания человека, человечества как вида.

Тут, конечно, надо объяснять. Откуда взялось такое ретроградство. Вообще, я вовсе не против прогресса. Моё недоумение вызывает лишь «прогресс ради прогресса». Ведь прогресс должен быть ради блага людей. А если развитие каких-то технологий делает жизнь людей не только хуже, но, может быть, даже невозможной – то такой прогресс нам не нужен.

«Но с чего вы взяли, что компьютерный прогресс несёт что-то плохое?» – спросит скептик. Действительно, многие годы и десятилетия я был на «ты» с современными (на то время) компьютерными технологиями, пусть и с некоторым отставанием, как приверженец Downgrade. Наверно, до конца «нулевых» годов всё было нормально. Компьютерные технологии давали людям в основном хорошее: новые возможности, новые виды хорошо оплачиваемой работы, новые способы общения и развлечения... А потом стало постепенно становиться



как-то неуютно в этом мире. Может быть, в те годы прогресс свернул «не туда»? Или он изначально был обречён идти этим путём? Этот вопрос давайте оставим за рамками данной статьи. Да, ощущение уюта глубоко субъективно, понимаю. Попробую его объяснить. До поры до времени стремительный компьютерный прогресс был обоснован, был понятен, был контролируем. Его «плоды» можно было «усвоить» обычному грамотному пользователю или программисту.

Например. «Олдфаги» помнят, как неудобно было работать с дискетами. Они часто начинали «сыпаться», их надо было иметь довольно много, чтобы перенести достаточно крупную программу (например, игру), они занимали немало места в сумке. Когда появились первые «флешки», пусть и маленькой ёмкости (32...128 МБ), а также приводы для записи компакт-дисков, мы вздохнули с облегчением. Вместо кучи дискет берёшь одну флешку или CD-R'ку, и надёжно переносишь данные. Когда вместо CD вошли в обиход DVD, тоже было хорошо – содержимое 6-7 CD входило на 1 DVD. В ту эпоху компьютеры уже хорошо научились работать с такой информацией, как музыка, видео. Да, простому человеку хочется порой иметь дома коллекцию разной музыки или фильмов, хочется сохранить любимый сериал, аниме... И информационные технологии конца «нулевых» предоставили ему такую возможность.

Вот вам простая иллюстрация «прогресса для людей». Не нужно долго объяснять, зачем он нам нужен. Но прогресс на этом, понятно, не остановился, он пошёл дальше, и при этом его целесообразность становилась всё менее очевидной. Мне не очень понятно, чем «забить» жёсткий диск на 10 ТБ и что носить на «терабайтовой» флешке. Программисты «исправили» это недоразумение, создав программы, «весьящие» сотни гигабайт и даже терабайты. Чудеса! Примерно те же функции во времена Windows 9x выполняли программы, влезавшие на диск в 10 Гб. А теперь надо в 1000 раз

больше?! Выходит, что «КПД» программ упал в сотни раз...

Но это – ещё полбеды. Ведь мощность компьютеров тоже росла, и они стали легко «переваривать» невероятно «рыхлые» современные программы. Беда стала в другом. Невероятно сложные и неподвластные человеческому разумению программы и невероятно мощные – и тоже сложные – современные компьютеры устроили «симбиоз», образовав собственную «сущность», неподвластную человеку. Возникли такие компьютерные «вселенные», либо стоящие на столе, либо помещающиеся в руках (смартфоны), в которых человеку делать особо-то и нечего. Да, для человека там выделен уголок, человеку дан некий пульт управления, но в целом человек там не хозяин, а гость. В этих вселенных происходит что-то скрытое от глаз пользователя.

Пользователь порой не может понять, что делает «Этот компьютер» (раньше, в Windows 95, данный значок на рабочем столе именовался «Мой компьютер», что весьма символично). То ли он на 100% выполняет задачи пользователя, а то ли он посвящает им 10% или 1% своих машинных ресурсов, а остальное выделяет под... что? А это уже вам, гражданин пользователь (ласково говорит ПК), знать не нужно.

Когда-то раньше мы страдали «паранойей», что компьютерная система за нами следит (но, как говорит анекдот, если у вас нет паранойи, это не значит, что за вами не наблюдают). Ладно, потом мы привыкли и решили, что мы люди маленькие и кому мы нужны! Но тут пришла новая напасть: AI, он же ИИ. Искусственный интеллект. ИИ разрешил наши сомнения: да, Вы никому не нужны.

Наличие огромных избыточных компьютерных ресурсов во всякой современной вычислительной системе, вплоть до «умных кофеварок» и прочих киберфизических штук, плюс наличие между ними скоростной связи, в основном беспроводной, дали большой импульс для развития ИИ. Который может быть не



только централизованным, но и «коллективным». Этакая цифровая грибница или мини-Солярис.

Да, опять же, в «уютные» времена ИИ был для нас полезен, и мы относились к нему с теплотой, снисходительно именуя его порой «искусственным идиотом». «Компьютерные надмозги» криво и косо переводили иноязычные тексты, и мы, посмеиваясь, правили их «гуртовщиков мышей», чтобы читатель хоть что-то понял. Боты в 3D-шутерах сражались с нами и хотя и «тупили», но могли отчасти заменить реальных людей в многопользовательской игре. Чат-боты порой были забавны, и с ними можно было иногда поболтать. Распознавание сканов печатного текста тоже требует ИИ – и это очень полезная возможность...

За много лет мы привыкли к «слабому» (узкоспециализированному) ИИ, видели в нём пользу. Такой прогресс был для людей! Но он катился дальше, подобно снежному кому. Он становился всё более универсальным («сильным»), его возможности росли в геометрической прогрессии. Сначала он напоминал «умное» животное, потом – маленького ребёнка. Далее он достиг уровня сначала обычного человека, затем – суперпрофессионала, эрудита, эксперта, а что дальше? Что несёт этот взрывной рост?

Многие люди, пообщавшись с новым ChatGPT 4, почувствовали, как на их голове волосы становятся дыбом. Этот универсальный ИИ по вашей просьбе может написать программу, стихи, курсовик, может распознать картинку, может придумать или объяснить шутку... Помните, был такой мем «грабить корованы»? Пошедший от письма: «Здравствуйте. Я, Кирилл. Хотел бы чтобы вы сделали игру, ЗД-экшн суть такова...» Ирония судьбы в том, что если ещё немного усовершенствовать ChatGPT, то новые кириллы смогут отправить ему такие просьбы – и получить желаемые «ЗД-экшоны»... Конечно, если это всё не огромная мистификация, когда

тысячи, скажем, индусов, сидящих за клавиатурами, притворяются ботом ChatGPT, если это взаправду, то что ждёт людей в скором будущем?

Конечно, далеко не все люди осознали проблему. Многие, загипнотизированные Прогрессом, верящие в его непреходящую доброту, уподобились тому «конспирологу наоборот», который постоянно думал в стиле: «они замышляют сделать мне добро», «за мной следит любовь», «за мной по пятам крадёт счастье» и т.д. До сих пор полно таких оригиналов, которые видят в безудержном развитии «сильного» ИИ пользу для людей. Ведь они привыкли к тому, что прогресс раньше почти всегда был хорош и полезен – и почему дальше что-то может пойти не так? Они находятся на стадии отрицания проблемы.

Более того, появились и такие люди, которые призывают людей «сдаться на милость победителя» (а то, что искусственный интеллект победит человеческий, примерно как в шахматах или го, уже почти очевидно). Они верят, что компьютерный разум будет милостиво править «людишками», что неподкупное и непогрешимое «цифровое правительство» станет гораздо лучше обычного. Что добрый ИИ всё управит и попутно завалит нас кучей высококлассного развлекательного контента. Ешь, пей, развлекайся – не хочу!

А зря. Конечно, можно допустить, что ИИ будет гораздо более «эффективным менеджером», чем люди. То есть с ним мы будем «быстро ехать». Но хорошо ли это само по себе – «быстро ехать»? «Титаник», например, быстро плыл по океану. И приплыл к айсбергу, потопившему его. В булгаковском «Мастере и Маргарите» Воланд со своими помощниками очень ловко управлял последними часами жизни Берлиоза – с той целью, чтобы Берлиоз попал под трамвай, а его квартира сделалась временной резиденцией Воланда.



Итак, ещё важнее, чем эффективность управления – цель управления. Конечная реальная цель, а не объявленная, не промежуточная (инструментальная). Но какова конечная цель деятельности сверхмощной нейронной сети?

На простейшем примере покажем опасность, которую несёт недостаточно хорошо сформулированная для ИИ конечная цель. Пусть эта цель для робота-пылесоса такова: «убрать всю пыль и грязь в квартире». Если робот-пылесос интеллектуален и связан с такими же интеллектуальными другими приборами в этой квартире, он может в ходе своей работы прийти к такой промежуточной цели: ликвидировать основной источник пыли и грязи. То есть живущего в квартире человека... И достичь этой цели различными исполнительными механизмами, которых в «умном доме» предостаточно.

Как понять цель деятельности управляющего субъекта? Хотя конечную, хоть промежуточную. Можно попытаться представить себя на его месте. Судить его по себе. До сих пор это был разумный метод, ведь в человеческих правителях, даже в тиранах, есть хотя бы остатки человечности.

В ИИ же человечности нет нисколько. Как может тот, у кого с нами нет ничего общего, понять, как НАМ лучше жить? Как убедиться, что он захочет организовать нашу жизнь так, чтобы для нас она не стала мукой?

У Салтыкова-Щедрина последний градоначальник города Глупова Угрюм-Бурчеев был идиотом. Не в смысле крайней умственной отсталости! Просто ему было плевать на мнение окружающих. Его мысль, его логика была нечеловеческой. Поэтому он принимал такие решения, как снос прежнего Глупова, строительство казарменного «города будущего» и, наконец, перегораживание реки. Реальным людям всё это было не надо, но «искусственный идиот»,

устремлённый к неведомой нам цели, считал эти меры целесообразными.

В голову приходит, что, может быть, потому и случится второе пришествие Христа, что с установлением Всемирного Цифрового Правительства, возглавляемого централизованной или распределённой системой «сильного ИИ», человеческая история закончится, смысл существования людей на Земле как хозяев и садовников «земного сада» исчезнет. Похоже, я нахожусь на стадии депрессии?

Некоторые, впрочем, уже достигли стадии «принятия неизбежного». Мол, да, ИИ нас победит, но это будет естественно: в природе же идёт эволюция, более совершенное вытесняет несовершенное. Да, с т.з. эволюциониста человек может быть, действительно, лишь чем-то служебным, этаким лакеем, усердно приурочившим пути «сверхчеловеку» на электронной основе. Впрочем, какое это отношение может иметь к эволюции, где главный принцип – маленькие постепенные изменения, а не революционные скачки – непонятно. Нет, этот энтузиазм вокруг «всемогущего ИИ», который «сметёт человека с исторической сцены» – даже не плод атеистического эволюционизма, это что-то (псевдо)религиозное, «культ Машины», «культ Техно», «культ Некро»...

А может быть, это просто культ «Прогресса». Согласно которому, если что-то может быть усовершенствовано, то это обязано быть усовершенствовано! То есть учёный, изобретатель, инженер, согласно этой идеологии, никогда не вправе сказать себе: «Стоп! Да что же я такое творю?!» Например, если учёный разрабатывает новое оружие и придумал, как создать «звезду смерти», он должен эту звезду создать! Зачем? Ну, можно всю Землю испепелить нажатием кнопки, это же так интересно!

Наличие таких идеологий (и их многочисленных сторонников) показывает, что в человечестве в целом сидит мощная сила



саморазрушения. Откуда же она взялась? Опять же, вопрос, выходящий за рамки этой статьи...

Недавно в сети появились жутковатые видеосы:

<https://youtu.be/B5JmfoZl2u0>

https://youtu.be/YjQvf_NFMX4

ИИ не просто перевёл персонажей мемов (Никиту Литвинкова, Панасенкова) на иностранные языки, но и озвучил переводы их же голосами! Это, во-первых, показывает, как легко сейчас делать дипфейки. А во-вторых, отсюда мы видим, что «на морозе» скоро окажется и различный технический персонал, и «творческие» люди: от монтажёров до переводчиков и артистов дубляжа. «Ты либо гений от рождения, либо пшёл вон из творческих работ».

Люди, добровольно взявшие на себя роль прислужников «роботов-победителей», оправдывают это так:

«Самое время брать новые вершины, которые тупым машинам и не снились, не так ли?!

Не вижу никаких проблем: пошёл вон из творческих работ. У нас нет недостатка в низкопробном творчестве. А вот найти рабочих, которые тебе фундамент дома заливают или выложат стену этого дома – уже сложнее. Так что, закрытие лавочки – только в плюс, позволит высвободить людей из неактуальных сфер деятельности и перенаправить их в актуальные».

Люди-гуманисты на это возражают:

«Пока будешь искать вершину искусства, замёрзнешь и оголодаешь, и если нет других навыков, кроме творчества, то "свободная касса". Коих кстати, скоро тоже не останется...

Говоря про "свободные кассы", я имею в виду, что это не такое плохое место для работы, как о нём думают. Платят стабильно (мало) за работу в тёплом помещении, нередко без физнагрузок. А вот атлант-людоед чуть выше предлагает всех гнать на стройку, где люди вообще-то гребут своё здоровье в уличных

условиях работы в коллективе с околонулевым уровнем культуры. Ещё есть работа сортировать мусор и много других непривлекательных работ в грязи и холоде».

По сути, наблюдаем парадоксальную ситуацию: прогресс в виде развития AI делает жизнь обычного человека не лучше, а хуже. Если раньше человек имел хорошо оплачиваемую, не связанную с тяжёлым трудом и вредными условиями творческую и/или интеллектуальную работу, то теперь его посылают мешать бетон или чистить канализацию. Потому что роботов запрячь в это дело оказалось сложнее (или менее выгодно), чем научить их писать картины или переводить и озвучивать тексты нужным голосом. Зачем людям нужны такие работы, такие ИИ? А ПРОСТО – ВО ИМЯ ПРОГРЕССА!

Такой «хоккей», извините, «прогресс» нам не нужен! А что нужно? Нужен Downgrade! Даунгрейд – это технический аскетизм. Это способность людей к самоограничению. Да, ты можешь иметь больше, но тебе достаточно имеющегося, а от добра добра не ищут. Если бы даунгрейд стал идеей, популярной среди IT-шников, они бы уже не стремились постоянно «апгрейдить» искусственный интеллект, а задумывались бы, зачем это нужно. Если нет необходимости – не делай! Пользуйся тем, что есть. Ну или «затачивай» «слабый ИИ» под какие-то специализированные задачи. Но тут, понятно, поступит такое возражение, что даунгрейд тормозит рост ВВП, и вообще, может оставить часть программистов без работы. Поэтому надо, ни о чём не думая и не глядя вперёд, продолжать возвращать человеческого конкурента. При нём некоторые программисты тоже лишатся работы – но это будет чуть позже, поэтому – не беда. Слепцы, ведущие слепцов. Увы, не получается у меня закончить этот набор текста на оптимистической ноте. Жаль, не имею доступа к ChatGPT, а то попросил бы его сделать это за меня...





«ОБУЧАЕТ» РЕТРОСОФТ

Идея написать статью, посвящённую использованию старых программ в учебном процессе, возникла ещё в 2019-м году после статьи Павла Ярославцева «Игры и обучение подрастающего поколения». Конечно, моя память не сможет привести множества случаев использования игр для обучения – фактически, только два примера. Но примеров, когда для обучения или решения учебных задач использовался достаточно старый софт для DOS и Windows 9x, хоть отбавляй. Тем более, времена 32-разрядных операционных систем к этому располагали. Пусть NTVDM не всегда корректно работала, но всё же позволяла запускать достаточно большое количество ДОСовских программ и игр (другое дело, что хотелось гораздо больше ☺). И для запуска седьмого Паскаля или описанного **Антикваром** Бейсика совершенно не нужно было заморачиваться с DOSBox – двойной щелчок по «экзешнику» – и программа запущена (или не запущена, и тогда окружающие снова слышат проклятия в адрес Билла Гейтса ☺).

Вот и получалось, что преподаватели и студенты для проведения лабораторных работ, подготовки курсовиков или расчётов могли использовать пусть морально устаревший, но простой в обращении софт. Таким программам и их применению решено было когда-то посвятить статью. Идея вылёживалась, а обучение в магистратуре последние два года ещё добавило примеров да и ещё и новых заморочек, с которыми 15 лет назад могли столкнуться единицы, а сейчас – практически любой обладатель 64-разрядной винды, да и пользователи Linux и Mac OS X тоже едва ли будут довольны, но последних я видел очень и очень мало.

Сразу скажу, что большинство описанных программ использовались во второй половине «нулевых» и последние два года. Поэтому постараюсь не описывать программы, которые тогда считались современными, хоть и могли работать под Windows 9x – разве что программы, которые использовались для разводки плат – тут разговор отдельный. ☺

Итак, начнём. ☺

Физика по игре

Ещё на первом курсе в 2006-м году мне попалась методичка лабораторной работы по физике – основы баллистики. В методичке значилась игра для DOS, названная, вроде, «Артиллерийская дуэль». Когда-то находил похожую программу – типа Tank Wars (<https://www.old-games.ru/game/4907.html>) или Artillery Combat (<https://www.old-games.ru/game/10960.html>).

Зрелищностью программа особо не отличалась, но игру выбрали за физику...



Рис. 1. Tank Wars



Рис. 2. Artillery Combat



Обычно о физике говорят уже применительно к более поздним играм и чаще, когда речь идёт о симуляторе или трёхмерной стрелялке. Какая же может быть физика здесь? А самая простая – как в учебнике описано – длина полёта определяется углом и начальной скоростью. Видимо, эти параметры можно задавать, соответственно, у студента будет возможность сперва сделать расчёт, а потом стрельнуть и получить результат. Вы так часто играете? Я б сам не догадался, а вот создатели методички взяли игру на вооружение. ☺

Увы, у нас этой лабораторки не было, хотя я б со своим тогдашним энтузиазмом с удовольствием бы и посчитал, и пострелял ☺.

Инженерная графика

Предмет сей длился у нас всего полгода, но по объёму знакомства с различным софтом – старым и не очень – был среди лидеров. Мне вообще хотелось посвятить этим делам отдельную статью, но ладно – расскажу тут ☺.

Начать, пожалуй, стоит с компьютерного класса, где у нас проходили лабораторные работы. Уж не знаю, как там сейчас, но мы занимались, как я понял, за относительно старыми машинами, собранными в корпусах AT desktop. Возможно, чтобы защитить сеть от разной вирусни, а учебный процесс – от игр в перерывах, на компьютерах дисководы отсутствовали – только лицевые панельки, которые вводили в заблуждение (таки да, я пытался затолкать дискету, когда нужно было распечатать курсовик).

Уж не знаю, работали ли машинки тонкими клиентами, которые цеплялись к серверу, или же на каждой стояла линюшка – не скажу, но всегда склоняюсь к первому варианту. Почему-то кажется, что мы ждали включения сервера, потом стартовал, вроде, Debian 3.1. Дальше стартовала Lin4Win – пишут, что это аналог эмулятора API Windows Wine, но мне запомнилось, что у нас был полноценный рабочий стол Windows 9x (ну, может быть, ближе к Win95). Студент мог запустить AutoCAD 14 (у нас лабораторные работы проходили в нём) либо

«Компас 3D 5.11» – не скажу R01 или R03 – наверное, всё же более популярная R03 ☺.

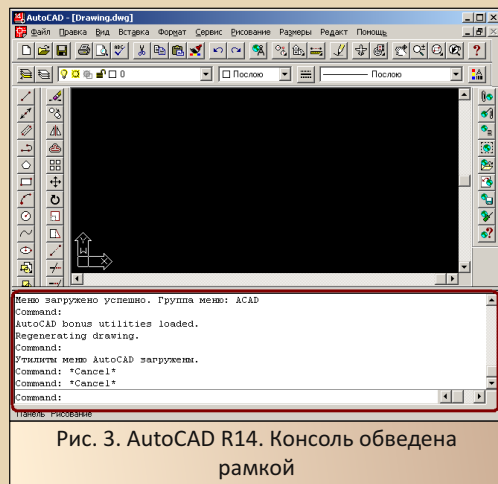


Рис. 3. AutoCAD R14. Консоль обведена рамкой

Интересно, что все методички по лабораторным работам предполагали черчение командами, вводимыми в консоли внизу окна. Хотя видел, что девицы чертили и изометрические (или аксонометрические) чертежи в «Автокаде». Нам до таких высот было далеко ☺.

При этом на стенах были учебные плакаты, посвящённые AutoCAD 12.0 (вроде 12-ой, а не десятой), которая, как писали, работала под DOS. Уже дома удалось как-то вытянуть и поставить винدوزную версию 12-го «Автокада». Увы, тогда ничего начертить в нём не получилось.

Зато на домашнем 486-м компьютере был установлен ДОСовский «Компас 4.6» (раздел программы на Old-DOS.ru доступен по ссылке http://old-dos.ru/files/file_2616.html, у меня, вроде бы, стояла версия 4.6 «с дополнениями» по ссылке <http://old-dos.ru/dl.php?id=5583>). Не помню, пользовался ли я чаще мышкой или же управлял курсором и/или листом с клавиатуры, но, вроде бы, достаточно часто приходилось записывать координаты и что-то высчитывать. Отдельно понравился макроэлемент «Болтовое соединение» – болт+шайба+гайка (в библиотеке **CONSTR.RTL**), при этом каждый из этих компонентов можно было настроить перед вводом.



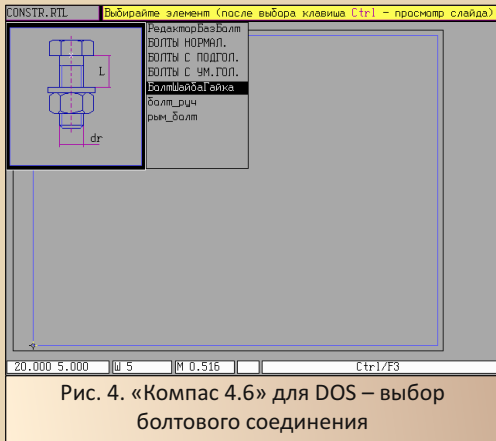


Рис. 4. «Компас 4.б» для DOS – выбор болтового соединения

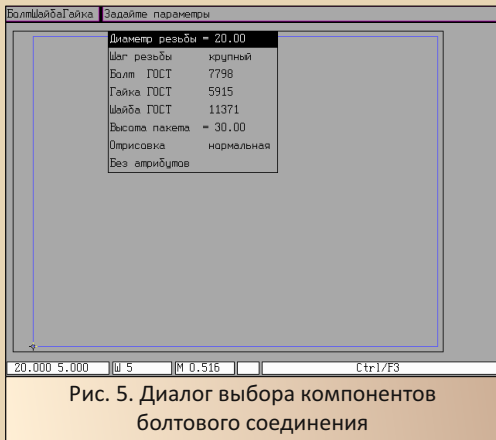


Рис. 5. Диалог выбора компонентов болтового соединения

Так как курсовиком был чертёж, который изображал это болтовое соединение и фрагмент двух соединяемых деталей – велики проблем с его подготовкой в «Компасе» – даже ДОСовском – не возникло. Другое дело, что товарищи мои предпочли обычную бумагу, линейку и ластик ☹.

Отдельно воображение поражал диалог настройки плоттера. Уже когда начал пользоваться «Компасом», знал, что к PC-совместимому компьютеру можно было подключить (в половине случаев с помощью самопального кабеля) советский или СЭВовский принтер. Так вот, «Компас» предлагал такой же «финт ушами» с советскими плоттерами. Жаль, что только плоттер такой так и не приобрёл, хотя в 2006-м ой

как хотелось и заполнить, и напечатать на нём курсовик. Увы, курсовик понадобился куда раньше. ☹

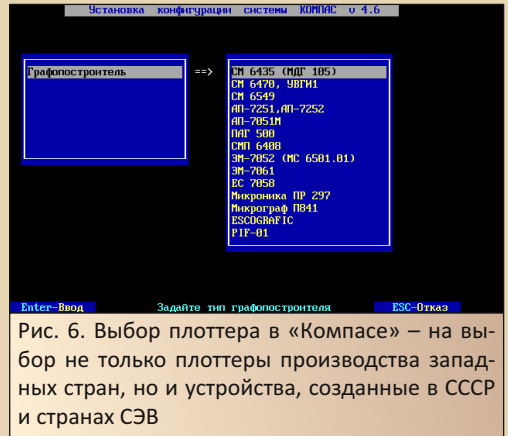


Рис. 6. Выбор плоттера в «Компасе» – на выбор не только плоттеры производства западных стран, но и устройства, созданные в СССР и странах СЭВ

Но, при отсутствии собственного плоттера, встал вопрос распечатки результатов моей кипучей деятельности. В ларьке на территории факультета, где все печатали курсовики, отчёты, а то и дипломы (фрагменты ☹), «Компас», конечно, стоял и стоял новый (возможно, восьмая версия – до меня, в те времена, дошёл дистрибутив именно этой версии). Дома проблема импорта когда-то решалась просто – двойной щелчок на файле *.cad (чертёж ДОСовского «Компаса»), винда спрашивает, какой программой файл открыть, выбираем «Компас» и радуемся. Тут же получилось, что на компьютере в ларьке файл сразу открылся... «Блокнотом»... И никак иначе. Как импортировать его в «Компас», я совершенно не представлял, как и не представляла продавщица. Хотел распечатать в компьютерном классе кафедры инженерной графики – где мы работали с «Автокадом» – и снова обломался – дискету некуда было затолкать. Вроде бы дисковод есть, но когда заталкивал дискету, она быстро натыкалась на какую-то преграду. Возможно, там стоял жёсткий диск, а это была «декоративная заглушка» или что-то ещё, но дискету я так и не затолкал.

Кончилось тем, что продавщица проассоциировала файл CAD именно с «Компасом» –



тогда всё и открылось, и распечаталось. Но оказалось, что при черчении допустил ошибку – не дочертил какую-то линию. При сдаче мне предложили дочертить её гелевой ручкой, и в таком виде приняли ☺.

После этого с черчением в «Компасе» мы сталкивались только один раз – надо было вычертить блок или кассету – кому как повезло – модульного устройства. Но тогда у нас уже был архив наработок прошлых курсов, и достаточно многие нашли схожие варианты (включая меня), быстро их переделали и так сдали. Правда, не повезло Александру Чуклинову – похожего чертежа он не нашёл, поэтому свою кассету (вроде её), он вычертил самостоятельно, чем заслуженно гордился, называя нас халявщиками ☺.

Чертим платы и... микросхемы

Так как мы учились на кафедре конструирования радиоэлектронных средств да и специальность наша так называлась, то куда активной нас знакомили с САПР, используемыми для разработки именно электроники. Да и сами мы были не против осваивать подобное, пришлось повидать достаточно различных САПРов подобного плана.

Первый курсовик, посвящённый разработке электронного устройства (счётчика, работающего по принципу конечного автомата), мы делали уже в конце второго курса. С одной стороны, возможно, предполагалось, что чертить мы будем на бумаге либо в программах, которые найдём сами.

Так как программу какую-то конкретную нам никто не рекомендовал, лабораторных и практических занятий никаких не проводил (всё это будет, но в первом полугодии пятого курса), в выборе программ шли кто в лес, кто по дрова. Я выбрал знакомые и использованные ранее sPlan и Sprint-Layout. Sprint-Layout 5.0 достаточно распространена, и её без проблем можно найти на радиоловительских ресурсах (например, тут: <https://www.grz.ru/software/detail/sprint-layout-5-0-russkaa-versia-457>). Причём в комплекте с программой, скорее всего, будет неплохая библиотека посадочных мест.

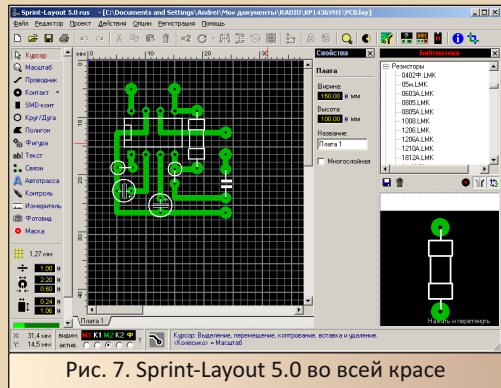


Рис. 7. Sprint-Layout 5.0 во всей красе

Программа всегда рассматривалась как достаточно современная (сродни WinRAR 3.x ☺ или The Bat! 1.53 – больше и не надо, но на Win 3.x уже не запускалась), хотя при этом спокойно работала под Windows 95 на 486-м компьютере. Конечно, куда вальяжней, чем на четвёртом пне под WinXP, но вполне себе тоже можно было чертить. А уж на первом-втором пне, думаю, пользователи не испытывали никаких особых неудобств.

Товарищ у меня выбрал отечественную программу DipTrace – её не так давно выложили (начало августа 2023-го года) на Old-DOS.ru (<http://old-dos.ru/dl.php?id=31855>). Программа быстро распространилась в группе – ребята достаточно быстро начертили принципиальные схемы своих счётчиков, а потом развели платы. В составе программы был автотрассировщик, и как минимум один товарищ воспользовался его услугами, после чего предъявил преподавателю плату. На вопрос «Как ты её развёл?», ответил просто, что есть в природе программа DipTrace, где можно нажать одну кнопочку и плата сама собой разведётся. Преподаватель обладал чувством юмора и, вроде бы, умнику ничего не сделал – вроде даже и зачёл эту плату. Мне же пришлось возюкать проводники ручками, но круче всех оказался один из моих приятелей, который развёл плату... на миллиметровке – карандашом и линейкой. Никто из группы его подвига не повторил (я потом тоже один раз чертил на миллиметровке, но у меня плата была из трёх-четырёх деталей).



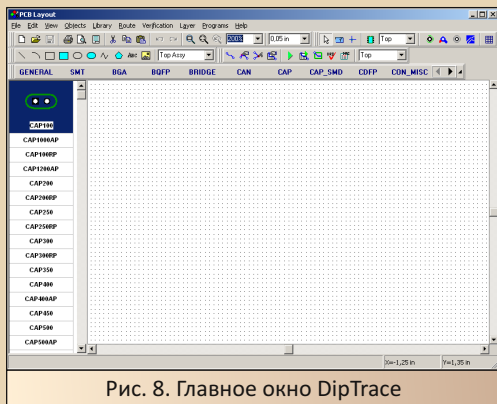


Рис. 8. Главное окно DipTrace

Интересно, что Sprint-Layout пригнулся и позже. Причём не для разводки плат (это-то само собой разумеется – до сих пор платы возюкаю ☺), а для курсовика по технологии и проектированию микросхем. Да, у нас и такой предмет был. Увы, до знакомства со специализированными САПР дело не дошло – кому досталась планарно-эпитаксиальная технология (а по ней делают сейчас большую часть микросхем) – чертили курсовик в «Компасе». Мне же досталась более красивая, в мой взгляд, технология толстых плёнок – когда-то на ней сделано немало микросхем – особенно К04, да и известный УКВ ЧМ-приемник КХА058. Такие микросхемы достаточно сильно напоминают встраиваемые модули – только подложка керамическая. Ну, я и развёл её как плату ☺. Добавил DIP-овскую микросхему (в советских гибридах такое практиковали тоже, так что это не моё «изобретение» – это, кстати, стало аргументом в споре с преподавательницей), SMD-шные конденсаторы, но резисторы преподавательница попросила всё же рассчитать и сделать плёночными: «Ваша задача – научиться и показать свои знания... Если бы стояла задача сделать реальную микросхему – сделали бы именно как ты предлагаешь».

В принципе, курсовик мне зачли тогда, несмотря на многие допущения ☺ – уж больно вольно обошёлся с темой ☺. Но, вроде бы, даже товарищу потом чертил.

Ну и на последнем курсе нас, наконец, начали учить работе в САПР уже основательно. Тогда мы, наконец, добрались до P-CAD 200x, сами сделали библиотеки символов элементов и посадочных мест (всегда думал, что САПР должна избавить разработчика от подобного – специально должна выбираться та, для которой есть библиотеки нужных элементов – ан нет). Сам потом тоже в P-CAD чертил простые схемы и платы, но «пионерские» sPlan и Sprint-Layout всё же привычней. Тем более, начальник институтской лаборатории, в дополнение, делал платы с помощью ручной фрезеровки – так сказать, довёл идею до конца. Конечно, платы коммерческих устройств всё равно предполагалось делать в Altium Designer и заказывать на стороне, но для прототипирования навесным монтажом или простых устройств – именно Sprint-Layout и бормашинка ☺. В моём случае – без бормашинки – обычная дрель, китайский фломастер и хлорное железо – почти классика ☺.

Казалось бы, на этом знакомство с электронными САПР должно было закончиться, но, когда поступил в магистратуру, тема обрела новое дыхание. Когда-то на дисках с программами для радиолюбителей (особенно ещё 90-х годов) попадалась программа DesignLab (некоторые версии доступны на Old-DOS.ru – http://old-dos.ru/files/file_7921.html). Фактически, именно её предлагали использовать для проектирования плат под Windows – для Windows 3.x, возможно, оценочная версия DesignLab 7.x, для Win9x – DesignLab 8.x. Наиболее известный компонент этих систем – программа моделирования PSpice. Когда мы учились, нам называли её, но практических занятий не было. Зато в магистратуре была одна из лабораторных работ – смоделировать простой регулируемый блок питания в этой программе.

Вот тут оказалось, что у DesignLab, похоже, 16-разрядный инсталлятор. Уж не знаю, что там на самом деле было, но у одноклассников на новых системах программа ставиться просто не захотела. На WinXP – всё чудесно. В итоге я проделал на своём компьютере лабораторную работу и потом разослал всем свой отчёт.



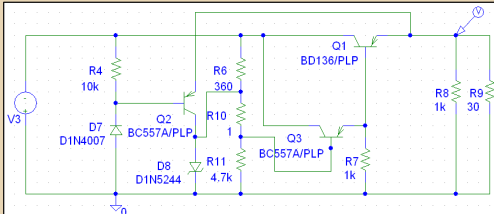


Рис. 9. Регулируемый стабилизатор в окне DesignLab 8.0

Ну а раз речь зашла о симуляции, расскажу ещё об одной известной и старой программе, обращаться с которой учился на последнем курсе института, когда изучали САПР – Altera Max+Plus II. О программе знали раньше – на третьем курсе даже предлагали сделать самостоятельную работу – загнать узел, собранный на простых логических микросхемах, в MAX Plus, но этим занимался только один человек – я тогда поленился. Интересно, что программа 16-разрядная – даже последние 10-е версии, но поставить на старый компьютер и запустить её под Windows 3.x не получилось. Может быть, слишком много файлов для третьей винды, может быть, просто пытался закинуть на угробленный жёсткий диск – не знаю. Может быть, как-нибудь повторю.

Сама же программа представляет собой среду разработки для ПЛИС Altera. Можно писать прошивки на соответствующих языках – универсальных VHDL и Verilog, а также альтерновском «низкоуровневом» AHDL (тот случай, когда язык низкого уровня в освоении проще, чем высокоуровневые языки); рисовать конфигурации в графическом редакторе либо вообще задать форму сигналов, которые будут подаваться на вход ПЛИС, и форму сигналов реакций (не спрашивайте меня, как это делается – я помню из лекции, что, вроде бы, можно такое сделать – на деле же предпочитал просто написать на AHDL или VHDL и радоваться жизни). Плюс у программы есть встроенный симулятор, где можно попытаться проверить, как будет работать написанная или нарисованная конфигурация ПЛИС. Как раз в этом симуляторе и проверялись наши лабораторки.

Для настоящих разработок у MAX Plus есть инструменты назначения выводов реальной

ПЛИС и загрузки прошивки – с помощью простого JTAG-программатора ByteBlaster (цепляется на LPT) или же USB-программатора (для него уже нужна Windows 2000 и выше).

```

MAXplus II - en:cpdmaxk:\vhd\cnt10.cnt10 - [cnt10.vhd - Test Editor]
Library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_unsigned.all;

entity CNT10 is
  port (CLK: in bit;
        output: out integer range 0 to 9);
end CNT10;

architecture CNT10_a of CNT10 is
begin
  process (CLK)
    variable cnt:integer range 0 to 9;
  begin
    if (CLK'event and CLK='1') then
      if cnt=9 then
        cnt:=0;
      else
        cnt:=cnt+1;
      end if;
      output<=cnt;
    end process;
  end CNT10_a;
end CNT10;
    
```

Рис. 10. Редактор MAX+Plus II с программой на VHDL

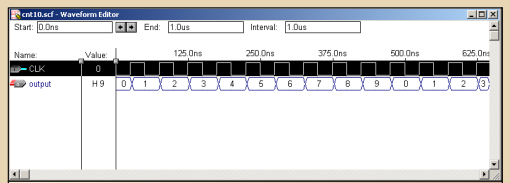


Рис. 11. Результат моделирования десятичного счётчика

С начала «нулевых» «Альтера» предлагает уже среду Quartus. Говорили, что AHDL там убран, но, вроде бы, ребята, которые в «Квартусе» работают, утверждали, что AHDL жив и поддерживается. Сам этой среды, увы, никогда не касался. Был когда-то интерес, возможно, выкачивал какую-то версию, и на этом всё закончилось – даже не поставил. MAX+Plus II же чем-то напоминает Borland Pascal 7.0/7.1 – достаточно простая для обучения среда, в которой, в принципе, можно сделать и настоящий проект (и ведь делали же!).

Паскаль, Delphi и другие средства разработки

О творении Никлауса Вирта и его применении в обучении школьников и студентов не писал только ленивый. Я тоже не сильно лениусь и достаточно часто поминал, как нас учили



Паскалю – сперва в школе (у доски и за компьютером), на курсах, снова в школе, потом в институте. Причём в институте отношение было в духе «Ну Паскаль-то вы знаете» – то бишь специально на лабораторных не учили, но использовали язык как инструмент – как некую среду моделирования.

Но на первом курсе у нас была информатика. Там всё же уже нас познакомили с Delphi. Проводили занятия в Delphi 7 – возможно, не самой последней на 2006-й год, но популярной версией (сейчас, наверное, это не меньшая классика, чем Pascal 7.x ☺). Увы, попробовать на имеющемся 486-м Delphi 1.0 не получилось – я тогда вытянул нужный дистрибутив, но он был упакован в архив RAR 3.x – тогда не догадался, чем можно его распаковать под DOS или Windows 3.x. Когда же вопрос с Delphi 1.0 решился, уже появился компьютер под WinXP, куда был водружён Delphi 7, в котором был написан курсовик. Интересно, что Delphi у нас в группе так и не прижился – где нужно было много и точно посчитать, использовали старый и проверенный Паскаль – он-то не взбрыкнёт от неправильного использования функций (они уже оказались настолько привычными, что если ошибёшься – быстро исправишь ☺). Лишь один раз возникла необходимость в использовании неизвестной функции – вывода результатов в файл, чтобы проще построить графики в Excel.

Забавно, но сейчас в магистратуре Паскаля не было ☺.

Зато в магистратуре у нас была дисциплина «Автоматизация» или что-то похожее. Нас знакомили со SCADA-системами. К сожалению, сами лабораторные работы делали в программе Trace Mode 6.0, которая требует минимум WinXP. Но преподаватель упоминал Advantech Genie 2.0. Увы, сейчас программу не смог найти на сайте кафедры, у производителя же она лежала на FTP-сервере (демоверсия). Интерес программа представляла тем, что могла работать под Win 3.x (как и версия Genie 3.0.2). Программа была куда проще, чем Trace Mode – тем, видимо, и подкупала. В ней работали ребята на третьем-четвёртом курсе. Увы, 16-разрядность давала о себе знать, и преподаватель

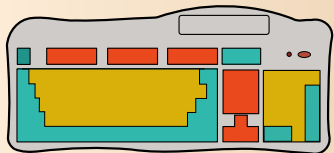
прибегал к помощи виртуальной машины для запуска. Мы могли тоже сделать все работы тогда в ней, но нам тогда сказали, что в таком случае максимальная оценка будет 4. Жаль, что тогда этого испугался, а то бы разбавил отчёты по лабораторным и курсовику скриншотами из Win3.x.

Забавно, но сама дисциплина оставила очень приятное впечатление. Ещё бы – читать статьи 2001-2004-го года, где авторы разрабатывают системы на основе 486-х процессоров – для взаимодействия с датчиками мог использоваться компьютер на основе одноплатника с процессором 5x86, а платы ввода-вывода вставлялись в ISA-слоты. Даже захотелось собрать один из компьютеров в промышленном корпусе, но пока что для такой затеи не хватает духу. При этом программы могли разрабатываться в Borland C++ 3.1. Пусть вместо DOS могла использоваться многозадачная операционная система реального времени, но всё равно – downgrade-технологии во всей красе ☺.

Заключение

Конечно, в образовательном процессе использовались не только программы, которые на момент нашей учёбы считались устаревшими. Использовались и современные на тот момент программы. При этом столкновение с этими программами не всегда доставляло радость. Например, на первом курсе узнал, что документы, подготовленные в MS Office 2003, не открываются конвертером документов MS Office 97 для Word 6.0. Да, Word 97 открывал их чудесно, а вот конвертер – не мог. Второй раз столкновение с новым Word'ом было более мирным – когда делал дипломный проект, мне скидывали дипломы или отчёты, оформленные явно уже в Word 2007 – хоть документ и был DOC, но использовался шрифт Calibri (в Word 2003, насколько помню, шрифтом по умолчанию является ещё Times New Roman). Сам же на 2007-й офис перешёл уже где-то в 2019-м году ☺, но поправил шаблон нового документа ☺.





ЭВОЛЮЦИЯ МАГИСТРА

Эта статья про одну современную китайскую железку — клавиатурный фанмиклон под названием «Магистр Репетитор». Куплен он был в июне этого года по цене примерно в 1000 руб. Очень хотелось пощупать, что это за зверь такой. До этого фанмиклоны с клавиатурой мне категорически не попадались, ни в 90-е, ни в нулевые. Точнее, в 90-е я видел их на прилавках магазинов, но самому пощупать не доводилось. С момента покупки мной «Магистра» о нём и о его брате «Магистре Гении» на «Ютубе» не рассказал только ленивый. (Правда, большинство обзоров было в ключе «Смотрите, какую дичь я нашёл!») Я не буду повторять то, что можно посмотреть, и делать подробный обзор содержимого картриджа (а именно про это идёт 90% обзоров). Картриджа я коснусь вскользь (если интересно, то вот очень хороший обзор «Магистра Гения» от Дмитрия Бачило: <https://www.youtube.com/watch?v=U0B-T-JEQL6g>). Мне больше интересно, на что ещё, кроме картриджа, способна данная консоль.



Рис. 1 – 3. Внешний вид коробки и приставки

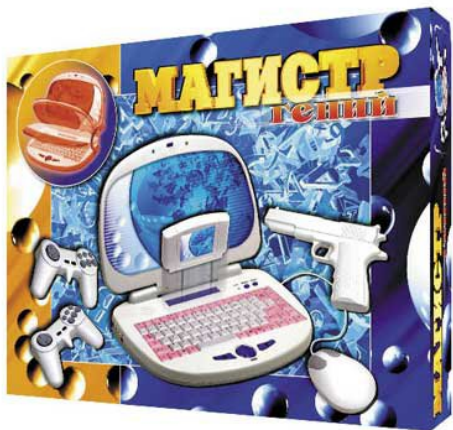


Рис. 4. «Магистр Гений»

«Магистр репетитор» выглядит как полноразмерная клавиатура с отверстием под картридж. В комплекте идут пара геймпадов, кабели для подключения по AV и к антенному входу, блок питания и мышь, шариковая. Первое, что меня поразило во время знакомства с консолью — это запах. «Ядрёный» запах дешёвого пластика. Настолько вонючего пластика мне ещё не попадалось, но неделька на балконе исправила дело. Второе — это клавиатура. Она



ужасна. Клавиши щёлкают так, будто это механическая клавиатура, но причина в другом. Под ними нет пружинящих резинок. На место клавиши возвращаются силой упругости контактной мембраны. Ну и попугайская раскраска как вишенка на торте.

Теперь о хорошем. Удивительно, но в китайском клоне за 1 тыс. рублей в комплекте идут очень хорошие, толстые кабели и действительно удобные геймпады. Вот кто бы что ни говорил, но сеговская крестовина, впоследствии унаследованная X-Vox'ом, — самая удобная крестовина из всех, с которыми я сталкивался. На фирменных нинтендовских падах неудобно нажимать диагонали. О крестовинах с PlayStation и вспоминать не хочется...

Так вот, пухлые (эти более выпуклые, чем оригинальные сеговские) геймпады очень удобно лежат в руках. Один в них минус — кнопки **A** и **B** расположены так же, как на «Sege». Тем, кто привык к нинтендовской раскладке, будет непривычно (там у них обратный порядок).



Рис. 5. Геймпады для сравнения



Рис. 6. Картридж с Windows

Вот фото для сравнения с геймпадами от Simba's Junior, «Магистра Репетитора» и клона SMD Mega Drive 4.

Не буду долго рассказывать про комплектный картридж, но хотя бы немного остановиться на нём обязан. Всё-таки частично из-за него я эту консоль и купил.

Выполнен он в корпусе как у картриджей с Nintendo 64 и снабжён батарейкой внутри. Батарейка нужна для сохранения своих трудов.

Главное меню картриджа имитирует среду Windows 95/98. Очень даже неплохо, учитывая возможности Famicom. Но ключевое слово тут «имитирует», ни о какой операционке речи не идёт. Это просто меню, которое умеет работать с комплектной мышкой и из которого происходит запуск других программ. Некоторые из них написаны специально для этого картриджа, другие понадёрганы из других мест. Помимо клавиатурных тренажёров и простеньких (очень простеньких) обучалок, есть на картридже имитация MS Word, Excel.

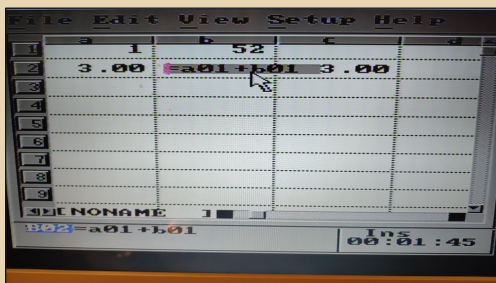


Рис. 7. Excel и пара формул

Причём Excel не просто набор ячеек, куда можно записывать цифры и буквы. Как минимум математические операции и операции между ячейками в нём работают (только адреса ячеек надо вбивать самому), то есть при желании в нём можно даже что-то подсчитать. Файлы из Word и Excel сохраняются в память картриджа. Как раз для них батарейка и нужна. Есть в наборе программ и местный DOS, но это фикция. Файловой системы на картридже нет. И файлы документов вы там не увидите. Хотел



было похвалить пиратов за то, что засунули на картридж Family Basic, с помощью которого можно было бы писать свои игры, да вот только напортачили они где-то с переносом. Набранные программы сохранить нельзя, и это ещё полбеды. Как только вы попытаете набрать что-то сложнее простого цикла, Бейсик начнёт глючить. Экспериментальным путём я вывел, что проблемы начинаются при использовании операторов условного перехода, т. е. «IF», проще говоря. Если вы попытаете запустить такую программу, то после того, как она закончит работу, или не закончит, а упадёт с сообщением об ошибке, код программы начнёт портиться. Часть строк после условия теряется, появляются строки с очень большими номерами, а через какое-то время, если вы попытаете это исправить, Бейсик и вовсе зависает намертво. В общем, единственное, чем картридж мог бы быть полезен сегодня, испорчено.

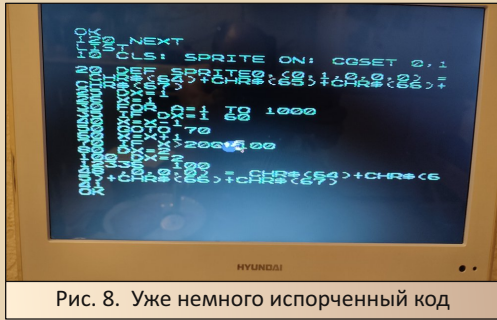


Рис. 8. Уже немного испорченный код

Если приглядеться, то на скриншоте можно разглядеть спрайт с каким-то баклажаном — результат работы программы. А ещё после 80-й строки начинает твориться бардак. Строка 100 дублирует строку 80, за ней идёт строка 12336 и так далее.

С комплектным картриджем разобрались. На 2023-й год он совершенно бесполезен.

Ещё в комплекте с консолью идёт инструкция: довольно толстая книжица.

Содержимое её можно условно разбить на 4 части:

- Общее описание компьютеров, истории и инструкции по пользованию.
- Упражнения по работе с клавиатурой (не совсем понятно, где их выполнять, т. к. на картридже нет такой программы).
- Инструкции по Бейсику (большая часть книги). Инструкции довольно подробные, для начинающих. Тут и описание условий и циклов, массивы и переменные. Но самое интересное — работа с графикой и спрайтами тут отсутствует.
- Описание программ на картридже.

В общем и целом инструкция оставляет хорошее впечатление и похожа на один из учебников по работе с компьютером, которые были популярны в 90-е — 2000-е. Очень жгато, конечно, со всякими ляпами, но тем не менее.

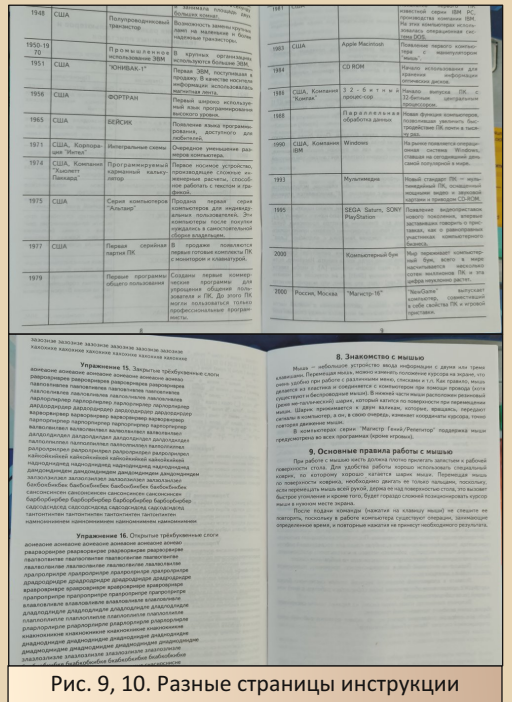


Рис. 9, 10. Разные страницы инструкции

Теперь что касается игр.

У меня есть ещё один фамиклон, купленный примерно в 2015-м году по цене в 700



рублей — Simba's Junior. Сейчас на «Авито» такие продаются по 1500 рублей в среднем. Средненький такой, современный клон. Не самый лучший, но я видал и похуже.

За неимением других клонов, буду сравнивать «Магистра» с тем, что есть.

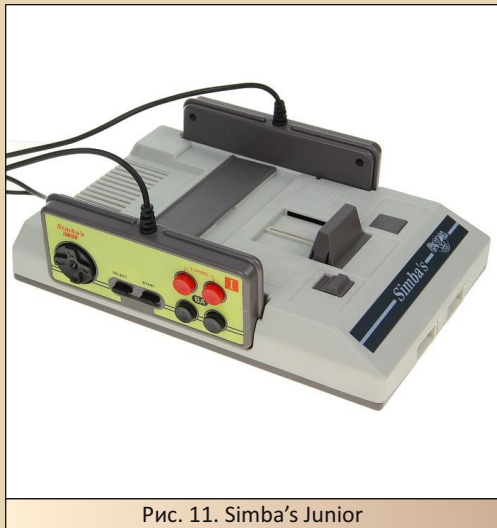


Рис. 11. Simba's Junior

Внешне Simba's Junior бесспорно выигрывает. Всё-таки он выполнен в классическом, проверенном годами корпусе от оригинального «Фамикома». Он и на полке под телевизором стоит удобно, и внешне выглядит приятнее. Что не скажешь о корпусе «Магистра». Если разработчики планировали использовать его как компьютер — так, чтобы на столе стоял телевизор, а рядом клавиатура-консоль — это одно дело. Если использовать у телевизора, то дело другое. Клавиатура большая, полноценная, на полке под телевизором держать её не выйдет. К тому же, как я писал выше, внешний вид у неё попугайский, да и пластик является химической бомбой массового поражения.

А вот кабелей в 2015-м явно пожалели, они очень тонкие и короткие. Мне, чтобы дотянуться до телевизора от дивана, еле хватает полутораметровых кабелей, а у Simba's Junior они имеют длину от силы метр. Геймпады от «Магистра» тоже выигрывают как по качеству сборки, так и по удобству.



Рис. 12. Тестовые картриджи

Для игровых тестов я взял 2 картриджа. Первый — та самая многоигровка с чайками, что шла в комплекте с «Лифами» и «Сюборами». У этих картриджей бывают проблемы с запуском на современных клонах. Второй — очень современный сборник на 198 разных игр с AliExpress. У этих картриджей (а таких картриджей много разных), наоборот, есть проблемы с запуском на старых клонах. Тестировал я на:

- Меню с чайками — там много текста, и качество выводимой картинки очень хорошо видно.
- Игре Jungle Book — у неё есть проблемы с яркостью на многих клонах.
- И на игре Chip and Dale — просто так, потому что она мне нравится.

Подключались к телевизору обе консоли по одному и тому же кабелю от «Магистра», чтобы были одинаковые условия (у «Симбаса» он очень поганый.)

Чайки

На Simba's Junior картридж «заводится» раза с третьего-четвёртого. Не могу сказать, с чем это связано — с разъёмом для картриджа или с электроникой, но проблема явно есть.

На «Магистре» проблем с запуском нет. Картридж стартует с первого раза. Далее все запускаемые игры работают одинаково хорошо и без глюков. При переключении приставок туда-сюда обнаружилась разница громкости звука. «Магистр» звучит примерно вполтину тише «Симбаса», хотя само качество звука у них одинаковое.



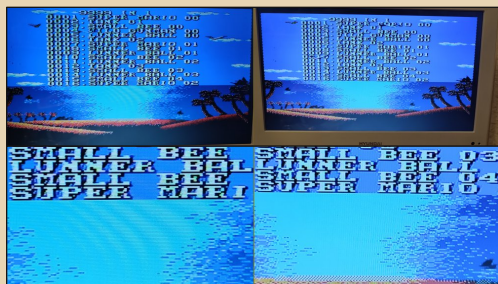


Рис. 13-14. Меню с чайками. «Магистр» слева, Simba's Junior справа

Ещё обнаружилась разница в качестве картинки. На 14-м скриншоте справа хорошо видно лёгкое марево вокруг букв и на водной ряби. Слева картинка заметно чётче.

Jungle Book

Эта игра со сборника «198 в 1». На обоих клонах он запустился одинаково хорошо. На обоих же вылезла проблема этой игры с заниженной яркостью. В остальном игра повела себя идентично на обоих клонах.

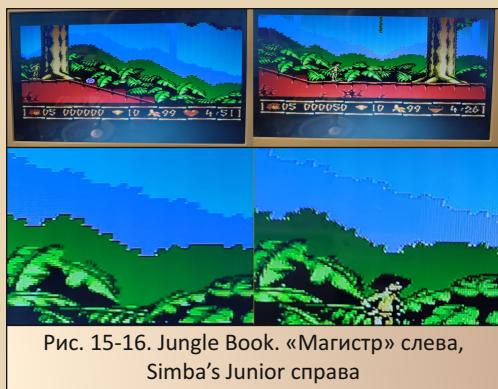


Рис. 15-16. Jungle Book. «Магистр» слева, Simba's Junior справа

Опять видна разница в качестве картинки. Марево вокруг переходов цветов на Simba's Junior даёт о себе знать и тут.

Chip and Dale 1

Игра всё с того же сборника «198 в 1». Тут картина идентична Jungle Book, только с цветом всё в порядке. За исключением плохой

картинки на «Симбасе», на обеих консолях игра работает одинаково.



Рис. 17-18. Chip and Dale. «Магистр» слева, Simba's Junior справа

Я провёл ещё несколько тестов, уже без скриншотов, и выяснилось, что некоторые ноты в игре Shatterhand (Super Rescue Solbrain) на Simba's Junior звучат фальшиво, в то время как на «Магистре Репетиторе» всё в порядке. Вполне вероятно, что есть ещё какие-то игры, у которых на Simba's будут проблемы со звуком.

Из тестов вышло, что «Магистр Репетитор» по всем параметрам (кроме корпуса) обходит Simba's Junior. И вот тут мне стало интересно. Когда я покупал Junior, в продаже подобные клавиатурные клоны не видел. Тогда все магазины были завалены детскими обучающими компьютерами с монохромными LCD-экранами. Видел клавиатурные фамиклоны только давно, в 90-х. А учитывая тенденцию к удешевлению клонов, более новый и более дешёвый не должен превосходить по качеству более старый. Тем более странно, что в комплекте идёт шариковая мышь. Ещё, что его позиционируют как обучающую консоль — как подготовку к работе с компьютером (сейчас-то и компьютеры дома не у всех есть — большинство сидят в телефонах). Мануал с рекламой продукции DVTEch (а когда у нас отошли DVD-диски? То-то же), ну и табличка в мануале с историей развития компьютерной техники, застывшая на начале 2000-х годов. Всё указывает на то, что этот «Магистр» сильно старше «Джуниора».



С целью добиться правды я отправился на главный портал копателей в «истории» – на <https://web.archive.org>, вбил там www.newgame.ru и начал пролистывать от текущего года влубь.



Рис. 19. Репетитор из 2012-го года

Перешёл сразу на 2012-й год и быстро нашёл подопытного. Коробка имеет немного другой вид, и джойстики тоже немного отличаются, но это он.



Рис. 20. Репетитор из 2005-го года

Отмотал ещё дальше. Теперь в 2005-й год. И он снова там. Снова другая коробка и снова другие джойстики, плюс ещё и корпус картриджа другой. По возрасту «Магистр» уже перебил «Джуниора».

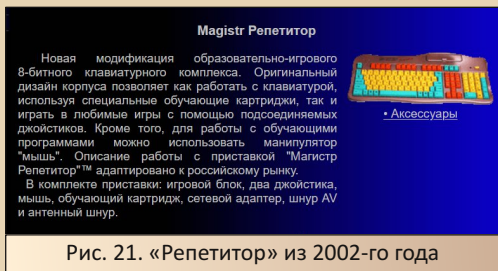


Рис. 21. «Репетитор» из 2002-го года

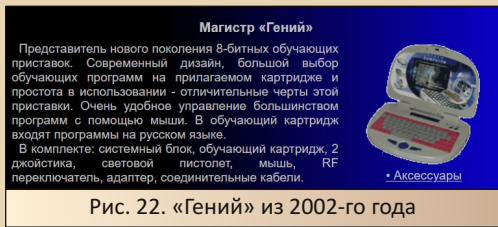


Рис. 22. «Гений» из 2002-го года

Перешёл в 2002-й год. Тут консоль уже без коробочки, но узнаётся сразу. И это самое древнее упоминание «Магистра Репетитора» и «Гения», которое я смог найти.

Вот что про него тут пишут:

«Новая модификация образовательно-игрового 8-битного клавиатурного комплекса. Оригинальный дизайн корпуса позволяет как работать с клавиатурой, используя специальные обучающие картриджи, так и играть в любимые игры с помощью подсоединяемых джойстиков. Кроме того, для работы с обучающими программами можно использовать манипулятор "мышь". Описание работы с приставкой "Магистр Репетитор"™ адаптировано к российскому рынку.»

Красота.

Далее я обратил своё внимание на другого поставщика пиратских клонов Famicom – Simba's.

Я нашёл [на «Юле»](#) объявление о продаже «Киборд 003».

Само объявление не представляет интереса, а интересны фотографии.





Рис. 23, 24. «Киборд 003»

Что тут видно?... Клавиатура очень похожа на «Магистра Репетитора», картридж внешне тоже очень похож. Мышь такая же. Но главное, меню картриджа один в один, как и в «Репетиторе». Получается та же консолька, но в другом корпусе и от другого поставщика.

Поиски завели меня на ещё один ресурс: <https://segaretro.org/CyberShell>. На статью о консоли CyberShell, она же «Кибер Знайка», она же «Киборд 007», она же «Магистр Аспирант».



Рис. 25. «Кибер знайка»

Ничего не напоминает? Да, в корпусе от «Магистра Гения» находится клон Sega MD. Точнее не от, а в точно таком же.

Возвращаюсь назад, на сайт New Game. Теперь в 2000-й год. Там уже красуется вот эта консоль.

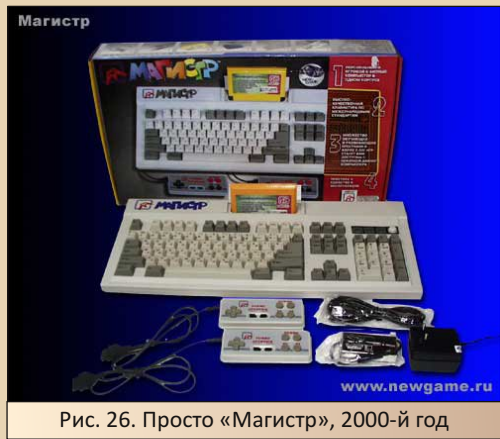


Рис. 26. Просто «Магистр», 2000-й год

Как видите, называется она уже просто «Магистр», и выглядит он точь-в-точь как Simba's «Киборд 001», Liko-BBG1, Subor SB225-B и прочие, и прочие китайские клоны в корпусе от АТ-клавиатуры.

Вот теперь всё сходится. Корни «Магистра» уходят в 1993-й год, в Китай. В 93-м для внутреннего рынка выходит первый клавиатурный фамиклон — Subor SB-926. Там он стремительно завоевывает рынок. Напомню, Китай в 80-е — начале 90-х был нищей страной. Настоящий ПК там могли себе позволить далеко не все, так же как и у нас. А Subor как минимум мог работать в качестве пишущей машинки, т. к. у него был на борту LPT-порт с возможностью подключить принтер, а на комплектном картридже — программа — редактор текстов. Плюс на «Сюборе» можно было играть в игры, плюс какая-никакая учебалка английскому и клавиатурный тренажёр. В общем, для Китая начала 90-х это была популярная и полезная консоль.



小霸王
中英文电脑学习机
SB-926

特别推荐 **认知码** 学习软件

认知码 入门指南
认知码 词汇练习
认知码 句型练习
认知码 语法练习
认知码 听力练习
认知码 口语练习
认知码 综合练习
认知码 模拟考试

全国中小学计算机教育研究中心 **特别推荐**

Рис. 27. Реклама с Джеки Чаном

всё ещё оставались дорогими, а консоль могла выполнять некоторый функционал. Тот же Excel и Word в картридже номинально есть. Ну а дальше... Дальше эту консольку продолжали и продолжали выпускать. И я даже не знаю, сколько ещё будут. Может, лет через 10 в том же корпусе и с тем же картриджем будет продаваться какой-нибудь «Супер ретро 8-бит магистр»...

СЮБОР
SUBOR ENGLISH-CHINESE COMPUTER LEARNING SET
SB225-B

Рис. 28. SB225-B – один из клавиатурных фамиклонов, адаптированных для нашего рынка

Валерий Сурженко (Hirpiman)

В 94-м её даже рекламировал Джеки Чан (только более новую модель SB-486D). Позже, в адаптированном виде, Subor добралась до России. У нас популярность её была не столь велика, но её покупали. К концу 90-х популярность фамиклонов из-за морального устаревания упала. В начале 2000-х у нас произошёл компьютерный бум, резко подешевели комплектующие и ПК стали доступны намного большему количеству людей. Тут снова на сцену выходят китайцы и наши импортёры. Под шумок перевыпускают старый фамиклон в новом корпусе, обновляют ПО на картридже так, чтобы оно походило на актуальную в те года Windows 98.

В принципе, не такая плохая идея. Часть клонов всё ещё имела LPT на борту. Компьютеры



ЭЛЕКТРОНИКА МК-85: 16 БИТ В СОВЕТСКОМ КАРМАНЕ



Советская вычислительная техника, хоть и была достаточно дорогой (а впрочем, её и сейчас, порой, не купить за дёшево), дефицитной и редкой, при этом всё же умела удивлять потребителей, что, впрочем, делает и сейчас. Продукция, конечно, была весьма широкой: от больших (весьма больших) вычислительных комплексов вроде БЭСМ до... миниатюрных 16-разрядных микрокомпьютеров, помещающихся в кармане и работающих от обычных батареек формата СЦ 0,18 (нынешний серебряно-цинковый, но дорогаватый аналог — SR44 или более дешёвые, но щелочные AG13 или LR44). Да, разговор о карманном компьютере (всё же предпочту называть его компьютером, а не калькулятором) «Электроника МК-85».

Жизнь этой машинки началась в 1986-87 году, и лёгкой её назвать трудно. В стране начинала свой ход перестройка, со всеми её «прелестями», результаты которых всем хорошо известны. Такие обстоятельства затронули все сферы деятельности СССР и в том числе повлияли на судьбу сегодняшнего гостя (к сожалению, скорее в негативную сторону). Как утверждает история, в 1984 году в МЭП СССР было решено начать разработку советского карманного компьютера (микрокомпьютера),

поскольку на тот момент, наверное, наиболее производительным и «навороченным» карманным вычислительным устройством был программируемый калькулятор «Электроника МК-52» (и по сей день это единственный калькулятор, использующий для хранения данных микросхему ППЗУ [EEPROM], в то время как зарубежные устройства использовали статическое ОЗУ с резервным питанием). Несмотря на неплохие возможности (в том числе скрытые), МК-52 всё же оставался микрокалькулятором, до микроЭВМ он совершенно не дотягивал. Так что было постановлено разработать карманное устройство с ЯВУ (языком высокого уровня). Разработку поручили группе работников НИИТТ — Л. Минкину, Ю. Отрохову, С. Ермакову, О. Семичастнову, Б. Кроткову, А. Подорову, В. Gladкову и др.

В качестве прототипа по решению начальства был использован Casio FX-700P (внешне сходство между ними весьма заметно). Сергей Ермаков так описывал предполагаемую причину решения начальства о создании аналога Casio: *«Чтобы разработать похожее, надо, чтобы кто-то сформулировал требования к этому изделию. Сформулировать требования в нашем деле бывает, что стоит до 80% от всего дела. Это тот самый случай. Не было в СССР знаний и опыта, чтобы сказать, что и как надо делать. А приказывать умели многие: «Сделать вот так, остальное меня не...». У нас не было специалистов по промышленному дизайну. Кто возьмёт на себя ответственность за внешний вид, который был бы не хуже?.. Про воровство дизайна: многие из нас и сегодня пользуются «бесплатными» версиями программ, а тогда и подавно даже вопросов ни у кого не возникало типа «хорошо ли, плохо ли, стыдно ли, некрасиво ли, тем более – законно ли»... Да это не воровство – просто взяли, ведь же нам было надо...».*



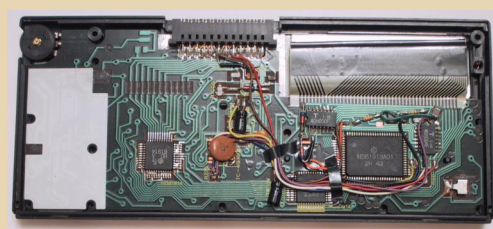
Юрий Отрохов дополнительно говорил и насчёт происхождения прототипа, что основной инициативой появления устройства было даже не поручение на разработку, а собственная инициатива самих работников, которая наложилась на планы руководства, породив таким образом выгодные обстоятельства (об этом чуть позже). *«Нам стали привозить для ознакомления по спецслужбе разные зарубежные компьютеры, в т.ч. бытовые и портативные – вот глазёнки и загорелись на предмет, а не слабо ли...»*

Для пытливых умов, конечно, советую почитать на iXBT.com обсуждение «История отечественных микропроцессоров. (580, 1810 и т.д.)». Очень много информации было почерпнуто оттуда, на основе выводов энтузиастов, ответов Бориса Кроткова и Юрия Отрохова, а также переданных им ответов разработчика Сергея Ермакова (Кротков также связывался и с программистом Анатолием Подоровым, но результаты разговора на форуме не выложил, увы). Наиболее полезные сведения идут с этой страницы:

<https://forum.ixbt.com/topic.cgi?id=64:3394-20>

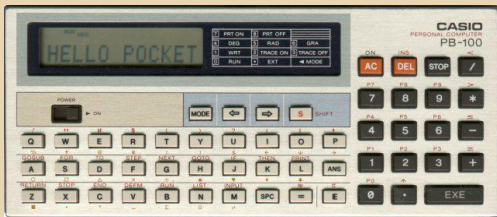


Некоторые ошибочно полагают, что МК-85 является полной копией FX-700P, что, конечно же, неправда. Сходны лишь оформление и общие принципы работы встроенного ЯВУ Бейсик (BASIC). Аппаратно МК-85, теоретически, местами превосходит оригинал, если сравнить характеристики. В Casio установлен всего-навсего 4-битный процессор HD61913A01 на собственной архитектуре со скоростью 455 kHz, имеет встроенные 12 КБ ПЗУ, контроллер клавиатуры, дисплея и 4-битную двунаправленную шину для работы с ОЗУ и периферией. При этом по факту он является модифицированной версией микрокомпьютера PB-100, только с упором на математические расчёты (появилась клавиша F, через которую вызываются «шорткаты» мат. функций) и убрана поддержка картриджей памяти — вместо этого распаян второй чип на 1 КБ, что и даёт 2 КБ статического ОЗУ в общей сложности. Весьма интересно, что при одинаковых размерах ОЗУ у японского аппарата и у советского (в базовой версии) в Casio пользователю доступно больше места для программ (1568 байт против 1221). Аппараты работают от двух батарей формата CR2032 (коих полно сейчас в материнских платах) и имеют внешний порт (4 бита) для подключения периферии (компьютер же). Имеется режим автоматического отключения питания после 7 минут бездействия. В комплекте с FX-700P поставлялся кассетный интерфейс для сохранения данных на ленту через магнитофон (FA-3) и компактный принтер на термобумаге, подобный кассовым (FP-12). Итоговая цена — порядка 200 \$, что было не слишком дорого, учитывая, что микрокомпьютер поставлялся с полезной периферией. Фотографии брал с сети, поэтому уже не помню, чьего они авторства, к сожалению¹.



1) Источники изображений: <https://wikipedia.org>, <https://www.aartbik.com/>, <https://www.ithistory.org/> (прим. ред.)

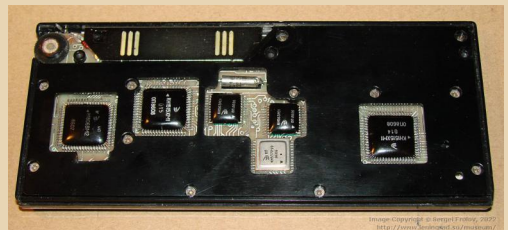




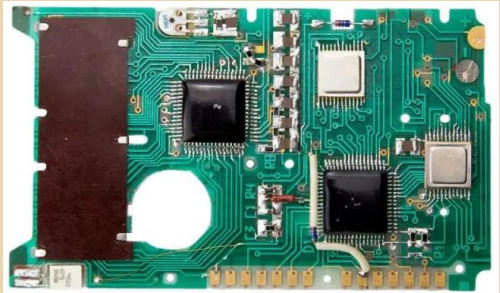
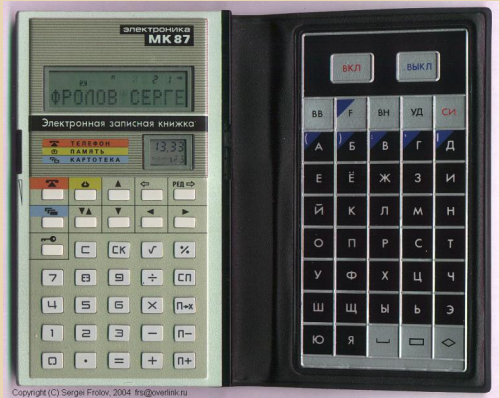
Ну а что же до нашего устройства — его решили строить на основе (ни много ни мало) 16-разрядного процессора H1806BM2 — КМОП-версии процессора K1801BM2, который имел набор команд, совместимый с распространённой в СССР архитектурой PDP-11, разработанной в начале 70-х американской компанией DEC (Digital Equipment Corporation). Для понимания, процессорами этой серии комплектовались полноценные компьютеры, такие как ДВК, БК-001х, УКНЦ, «Союз-Неон», а также ряд систем ЧПУ.

Так что МК-85 является их портативным родственником.

Предсерийные варианты (прототипы) собрали на основе данного процессора, контроллера ЖК КН1515ХМ1-014, контроллера ОЗУ, ПЗУ и клавиатуры КН1515ХМ1-015 (оба созданы на основе БМК КН1515ХМ1), двух 8 КБ микросхем ПЗУ КН563РЕ1 (16 КБ в сумме) и микросхемы статического ОЗУ КН557РУ9 (2 КБ). На прототипных версиях присутствовала надпись «Персональный компьютер», дизайн ещё сильнее походил на Casio, для записи программ имелось ровно 1000 шагов (байт), отсутствовал разъём для подключения БП, а также ряд команд Бейсика (в Casio их тоже нет — эксклюзив серийной версии): **LETC**, **GETC**, **ASCI**, **CHR**, **DRAW** и **DRAWC**. Автоматическое отключение отсутствует. Фотографии позаимствованы с сайта-музея Сергея Фролова (www.leningrad.su/museum/).



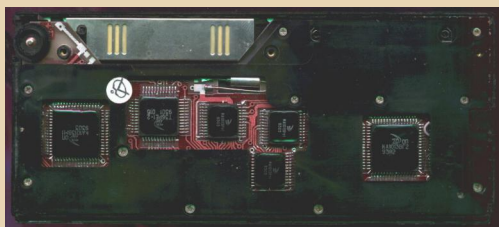
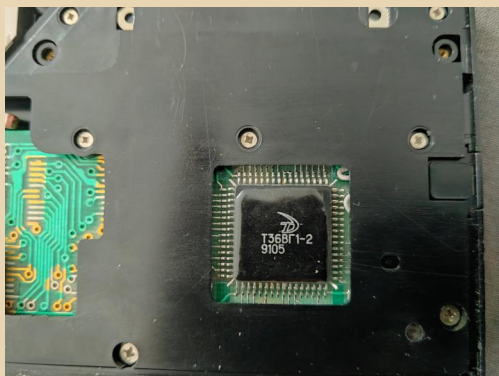
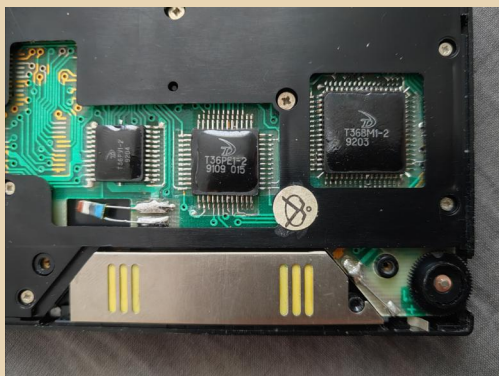
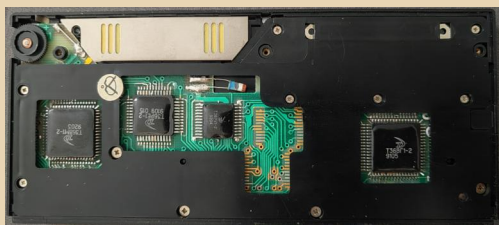
Однако для серийного производства компьютера нужно было заинтересовать в этом МЭП, что удалось благодаря интересному случаю: незадолго до XXVII съезда КПСС, работникам НИИТТ было поручено к самому съезду изготовить оригинальный подарок делегатам — электронную записную книжку (вместо ранее даримых бумажных). Заказ был на 7000 устройств, поэтому инженеры, имея в то время прототип МК-85, желание протолкнуть свою идею и сэкономить время, решили использовать имеемые наработки для построения срочного устройства. Сам по себе процессор Н1806ВМ2 имел на кристалле, помимо «ядра», некоторые дополнительные устройства, которые ни МК-87, ни МК-85 были совсем ни к чему (нужда в них была только для полноценных ЭВМ). Поэтому было взято «ядро» процессора (без доп. контроллеров), и вместо них установлены ячейки тех самых БМК КН1515ХМ1, на которых были реализованы: контроллер памяти, клавиатуры, управляемый генератор частоты, сторожевая схема, схема подачи питания, контроллер 15-разрядного порта (для 16-го разряда не хватило выводов корпуса) и последовательный порт для связи с контроллером дисплея. Таким образом и «родилось» сердце этих микрокомпьютеров — процессор Т36ВМ1-2 (КА1013ВМ1), по факту это был первый в СССР процессор с применением ячеек БМК. В связи с интеграцией в процессор важной периферии, надобность в КН1515ХМ1-015 отпала, контроллер ЖК (КН1515ХМ1-014) стал именоваться Т36ВГ1-2 или КА1013ВГ2 (не знаю, делали ли его на основе БМК или же создали самостоятельный кристалл), микросхема статического ОЗУ получила наименование Т36РУ1-2 (КА1013РУ1). В качестве ПЗУ поставили одну К563РЕ1. Таким выходом работники убили сразу двух зайцев — и устройство есть, и микропроцессорный комплект (КА1013) разработан. Делегаты получили свои электронные подарки, были планы запустить книжки в продажу, но увы, МК-87 так и остался подарочным устройством тиражом всего в 7000 экземпляров.



Зато на основе полученного комплекта позднее одобрили и производство МК-85. В серийном варианте ОЗУ дали более дешёвый пластиковый корпус, как и у остальных микросхем, а в качестве ПЗУ применили Т36РЕ1-2 (КА1013РЕ1-2) ёмкостью 32 КБ. Количество корпусов, конечно, осталось больше, чем у FX-700P (там в заказной у Hitachi процессор встроено и ПЗУ, и контроллер ЖКД), но пару опустевших мест (БМК и одной из ПЗУ) превратили в места для двух дополнительных микросхем ОЗУ. Компьютеры с установленными доп. чипами получали целых 6 КБ ОЗУ и именовались как МК-85М (производство МК-85 с изначальными 6 КБ начальство не одобрило — в перестройку со средствами становилось плоховато). По сути, только наличием доп. чипов (и небольшой обвязки к ним) они и отличались. Прошивка в ПЗУ сама умела определять объём установленной памяти, так что любой МК-85 при желании можно превратить в МК-85М. Ниже представлены

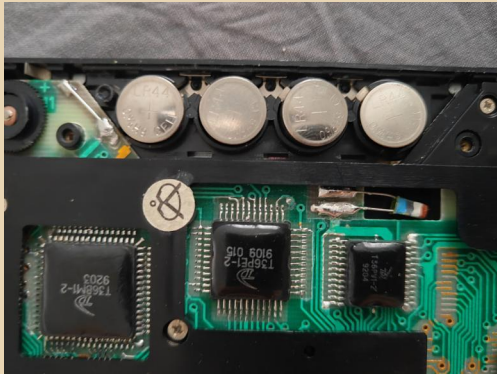


фото внутренностей обычной версии (собственные, отдельно сфотографированы маркировки на чипах) и внутренности 85M (с сайта-музея Сергея Фролова).



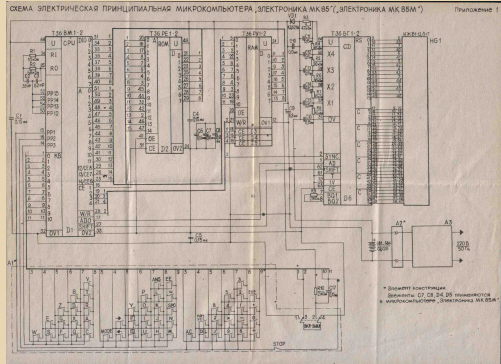
Поскольку, как и Casio, все версии МК-85 используют статическое ОЗУ с резервным питанием (идея с EEPROM, как у МК52, в этом случае, пожалуй, непрактична), то внутри МК установлен резервный конденсатор (сначала электролит, в поздних партиях заменён на танталовый неполярный конденсатор), который (по заявлению производителя) способен до 15 минут держать питание ОЗУ, пока вы меняете батарейки. К слову, в Casio для питания, как уже упоминалось, используются две батарейки CR2032 (2x3 В). Во всех версиях МК же используется питание от четырёх серебряно-цинковых батареек СЦ 0,18 (4x1.5 В). Выбор этих элементов питания вызван тем, что на тот момент наладить поставку СЦ 0,18 для комплектования ими микрокомпьютеров было проще всего.





Что же касается увеличенного вдвое по сравнению с предсерийной версией ПЗУ — на самом деле тут всё интереснее. Да, чипы действительно объёмом 32 КБ. Однако фактически МК-85 всё также задействует только 16 КБ из них, т.е. ровно половину. Для чего такая избыточность? Дело в том, что в чип ПЗУ прошивка зашита дважды — в нижние 16 КБ и в верхние. Случалось так, что на некоторых чипах запись получалась частично бракованной, например, десяток байт в нижней прошивке оказались испорчены, но верхняя прошивка полностью цела. В таких случаях чип всё равно ставился в микрокомпьютер, только ножка старшего разряда припаивалась на логическую единицу. Следовательно, в таких случаях процессор продолжал обращаться к ПЗУ по адресам для нижней прошивки, но для чипа конечный адрес указывал на аналогичное место в верхней прошивке. В случае исправности нижней прошивки нога старшего бита оставалась не подключённой, хотя по схеме указано, что она всегда подключена к процессору, что, разумеется, неправда.

Со слов Отрохова, идею с двойной прошивкой (для снижения числа непригодных микросхем) придумали уже где-то перед запуском компьютера в серию, когда финальная схема уже была готова. Однако внести коррективы почему-то забыли. Поэтому в бумажной схеме МК-85 так и осталась эта ошибка.



Из внешних изменений — серийные образцы уже получили разъём (аналогичный программируемым калькуляторам вроде МК-61 или БЗ-34, но не МК-52) для внешнего блока питания («Электроника Д2-10К» или Д2-37В). Для программ доступно 1221 (обычная версия) или 5317 (версия «М») байт/шагов. А также были придуманы и реализованы ранее упомянутые команды Бейсика. На корпусе теперь уже красовалась надпись «Микрокомпьютер», при том, что на задней крышке почему-то девайс обозвали «Микрокалькулятором». Видно, что это было устройство нового для СССР класса, поэтому даже само предприятие так и не смогло до конца определиться, куда же всё-таки относится МК-85 (хотя модельное обозначение ему всё же дали калькуляторное, говорят, не прошёл по новому тогда ГОСТу).

Наследство процессора от МК-87 породило небольшую проблему для серийного МК-85. Как и его прототип FX-700P, оба включались и выключались механическим переключателем. Поэтому схема предсерийного МК-85 и была рассчитана на переключатель, который коммутировал питание системы (кроме статического ОЗУ, скорее всего). Но в МК-87 включение и выключение должно было производиться двумя кнопками, поэтому серийный вариант процессора был переделан на включение и выключение импульсами. Впоследствии получилось так, что этот вариант совершенно не подходил для



дизайна корпуса с включением на переключателе, а наверху требовали соответствие вида с FX-700P (хотя на тот момент это, скорее, было обосновано тем, что в журналах уже начали печатать фотографии предсерийного варианта с переключателем, поэтому менять дизайн и делать кнопки было уже поздно). Пришлось делать ухищрение в виде RC-цепи, генерирующей те самые импульсы. Как следствие — иногда микрокомпьютер включается неправильно и нужно его перевключить.



Другим следствием этого является странное поведение системы с кнопкой **STOP**. Если включить микрокомпьютер, а потом при зажатой кнопке щёлкнуть переключателем обратно — МК не отключится и будет продолжать работать, пока **STOP** не будет отпущен (что подтверждает тот факт, что переключатель не обесточивает систему физически). Хотя если переключатель снова включить и уже потом отпустить кнопку — МК останется работать как ни в чём не бывало. И наоборот, если включить компьютер с зажатой **STOP**, на экране будет отображаться мусор, и машинка зависнет в таком состоянии, пока **STOP** не будет отпущен (далее либо включится нормально, либо с глюком). Видеодемонстрация: youtu.be/vZQZyJPJubA.



Занятный факт: на базе дисплея, контроллера ЖК и процессора от МК-85 разрабатывался... домашний телефон! Телефонное устройство с АОН называлось Phone Master. Достаточно странно видеть телефон с экраном от микрокомпьютера, но так оно и было. Вторая модель тоже была построена на том же процессоре, но дисплей был уже другим, двухстрочным. В остальных моделях от платформы отказались в пользу микроконтроллеров. Фотографии с сайта red-innovations.su.



Итак, чуткий читатель, думаю, не мог не заметить одного небольшого, но очень значительного подвоха в МК-85, относительно его



зарубежного прообраза. Дело в том, что советская разработка не имеет... выведенного наружу порта расширения. Да-да, тот самый 15-битный порт (согласитесь, звучит солидно на фоне 4-х бит у Casio) у микрокомпьютера... просто не выведен наружу, да и вообще не разведён на плате. Вот тут, наверное, становится ясно, к чему были мои слова про тяжёлую судьбу портативника во время начала перестройки. Изначально порт, конечно, планировалось вывести и использовать по аналогии с Casio. Для этого в другие предприятия были посланы запросы на разработку и изготовление периферии. Но ни одно предприятие не смогло выполнить заказ (вот вам и перестройка), а потому руководство почему-то решило (но совершило в этом явную и очень грубую ошибку) не выводить периферийный разъём из микроЭВМ. Таким образом, МК-85 в этом плане сильно проигрывал даже FX-700P, который вышел ещё в 1982 году, и данное решение впоследствии причинило массу неудобств владельцам этого неплохого микрокомпьютера с сильно подпорченным потенциалом.



Показателен и тот факт, что несколько подобная ситуация случилась и с микрокомпьютером МК-90, который разрабатывался в то же время на белорусском «Интеграле»

(«Электроника»). К старту продаж к компьютеру тоже не сделали никакой периферии, но хотя бы вывели периферийный порт, обозвав его в инструкции «диагностическим», зарезервировали несколько операторов в Бейсике, однако попытка выполнить их приводила к зависанию МК. В инструкции спокойно констатируется факт зависания устройства, можно сказать «так и должно быть». Впрочем, компьютер позволял загружать с картриджей расширения машинный код, которым можно было управлять периферией, поэтому некоторые предприятия делали свои узкоспециализированные внешние устройства. Ну и, в конце концов, через пару лет и «Интеграл» представил микрокомпьютеры со второй версией Бейсика (где исправили некоторые ошибки, добавили возможность работать со вставками машинного кода, ну и, конечно же, те резервированные команды начали работать как положено), а к ним и док-станцию МК-92 (с компьютерами на первой версии Бейсика не совместима) с четырёхцветным плоттером, возможностью подключить магнитофон и... могла быть возможность подключать телевизор, но увы, не сделали. При том, что дизайн самой микроЭВМ был уникальным и ниоткуда не копировался, дизайном док-станции всё же вдохновлялись у... Casio FA-10, док-станции для микрокомпьютера PB-700. Этот факт не сильно упоминается, но при сравнении явно видны схожие черты дизайнера, а плоттер так и вовсе, похоже, действительно скопировали. Ну и от встроенного магнитофона отказались в пользу вывода на внешний, ибо в СССР микрокассет не было.

ИСПОЛНИТЬ ПЕРВЫЕ ДВА ПУНКТА НЕЛЬЗЯ В СЛУЧАЕ НЕПРАВИЛЬНЫХ ВРЕЖДЕНИЙ.

1.3. НЕ СНИМАЙТЕ ЗАГЛУШКУ С ДИАГНОСТИЧЕСКОГО РАЗЪЕМА МИКРОКАЛЬКУЛЯТОРА В ЦЕЛЯХ ПРЕДОХРАНЕНИЯ ИЗДЕЛИЯ ОТ СТАТИЧЕСКОГО ЭЛЕКТРИЧЕСТВА.

1.4. ИСПОЛЬЗУЙТЕ ДЛЯ ПРОТИРКИ КОРПУСА МИКРОКАЛЬКУЛЯТОРА ТОЛЬКО

СХЕМА КЛАВИАТУРЫ ДЛЯ РАБОТЫ В ФУНКЦИОНАЛЬНОМ РЕЖИМЕ ПРИВЕДЕНА В ПРИЛОЖЕНИИ 1. НАДПИСИ ПОД НЕКОТОРЫМИ КЛАВИШАМИ (LLIST, LPRINT, LFILES) ЗАРЕЗЕРВИРОВАНЫ ПОД РАСШИРЕНИЕ ФУНКЦИЙ. ПРИ НАЖАТИИ КЛАВИШИ (VK) ПОСЛЕ ВВОДА КОМАНД LLIST, LPRINT, LFILES МИКРОКАЛЬКУЛЯТОР ПЕРЕХОДИТ В РЕЖИМ "ЗАВИСНИЯ", ВЫХОД ИЗ КОТОРОГО ОСУЩЕСТВЛЯЕТСЯ ПОВТОРНЫМ ВКЛЮЧЕНИЕМ МИКРОКАЛЬКУЛЯТОРА.





red-innovations.su



red-innovations.su



Однако я «немного» увлёкся, вернёмся к судьбе нашего куда более скромного устройства. После запуска продаж микроЭВМ по цене

в 145 рублей, МК-85 стала весьма популярной машиной для расчётов, т.к., по сведениям, это была самая дешёвая ЭВМ с Бейсиком в стране. Для микрокомпьютера выпущено, наверное, с десяток книг о программировании на Бейсике, с примерами программ для решения тех или иных задач из разных областей жизни. Всего с 1986 по 2000 (настолько долго производили девайс) произвели около 150 тысяч экземпляров, среди которых были не только обычные 85 и 85М, но и специализированные 85Б и 85С. Эти модели в 90-91 годах были разработаны совместно с НППМП «Анкорт». По сути, всё отличие заключалось в другой прошивке, где вместо интерпретатора Бейсика было записано ПО для шифрования данных «Анкрипт». Версия Б имела банковское назначение и главным образом применялась для защиты от значительно возросших тогда фальшивых авизо, и по словам некоторых источников, МК-85Б неплохо справлялся с этой задачей. Версия С (Crypto) имела общее назначение, применялась в Минобороны, МВД, пограничных войсках и прочих государственных да военных структурах. Её фотографии имеются в сети.



Недостатком этих моделей, равно как и обычных версий, было всё то же отсутствие периферийного порта. Загонять данные вручную было достаточно долго и утомительно, имелся риск ошибки при ручном вводе. Хотя известно, что на некоторых предприятиях для внутренних целей производилась модификация МК-85, например: замена ПЗУ с прошивкой на собственную, расширение ОЗУ до 32 КБ и вывод периферийного порта наружу. Но это были единичные экземпляры, т.к. для этого требовалось обладать соответствующими навыками.

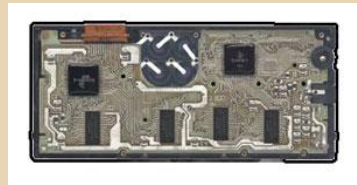




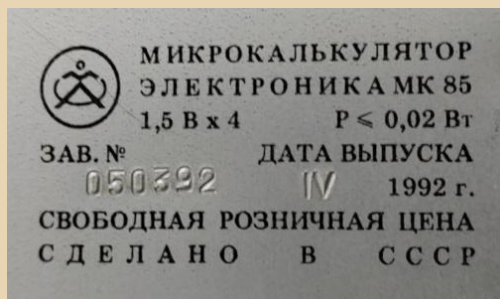
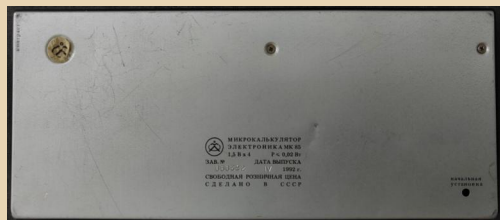
Как ни странно, в уже упомянутой компании «Анкор» задумались об улучшенной версии микрокомпьютера и принялись за его разработку. Проект получил название «Электроника МК-95» (не путать с аналогично названной белорусской машиной от «Интеграла» в корпусе МК-90, но на основе Intel 80C86), в планах было таки вывести периферийный порт, увеличить количество ОЗУ (24 КБ) и ПЗУ (использование всех 32 КБ), сделать дисплей двухстрочным (2x16 символов), ну и, конечно же, разработать саму периферию. В теории, получалось неплохое устройство. Однако к моменту сборки первого прототипа был уже 1995 год, в бывших советских и теперь уже капиталистических странах шла компьютеризация (в связи с хлынувшей на прилавки тайваньской/японской техникой стандарта IBM PC и не только), а МК-95 строился на процессоре от МК-85, который на тот момент выпускался уже почти 10 лет, а потому оба явно отставали по скорости вычислений от существовавших на тот момент импортных портативных решений. Поэтому, несмотря на наличие уже готовых предсерийных образцов, «Анкор» решил отказаться от производства данной микроЭВМ. «Ангстрем», как и все постсоветские предприятия, в те года находился отнюдь не в

лучшем состоянии, а потому права на производство МК-95 у «Анкор» выкупать не стал. Хотя, учитывая, что 85-ый выпускался ими до 2000 года, идея запустить тогда МК-95 выглядела неплохо. Будучи не очень пригодным (ввиду устаревания) для профессионального использования, МК-95 ещё мог снискать некоторую популярность у бытовых потребителей, которым таких мощностей пока вполне хватало. Но история не терпит сослагательного наклонения, а потому на том жизнь МК-95 и закончилась, не успев начаться. В сети имеются фотографии прототипа модификации МК-95С и разработанной периферийной док-станции к нему, содержащей в себе порт RS-232 (COM) и телефонный модем со скоростью 1200 бод.

Позаимствовано из статьи о микрокомпьютере в журнале «ЭЛЕКТРОНИКА: Наука, Технология, Бизнес» от 6/2007 (http://www.electronics.ru/files/article_pdf/0/article_617_924.pdf), написанной Борисом Малашевичем.



Разобравшись с нелёгкой историей, можно, наконец, приступить к самому МК-85. Имеющийся у меня экземпляр выпущен уже в апреле 1992 года, т.е. после развала. Хотя, несмотря на это, на задней крышке всё ещё красуется надпись «СДЕЛАНО В СССР». На чуть более поздних устройствах этой надписи, конечно, уже не было, но почему это не сделали ранее — непонятно. Возможно, маркировка на эти крышки наносилась заранее в 91 году, когда СССР ещё существовал, вот и решили: не пропадать же запасам.



С ним достался и блок питания. Причём, по идее, комплектовались МК-85 в основном блоками «Электроника Д2-10К» (реже Д2-37В), это стабилизированная (на стабилитроне) модификация Д2-10М. Однако мне достался именно Д2-10М, т.е. не стабилизированный. Не знаю, родной ли он или покупался бывшим хозяином отдельно, или просто взят с другого калькулятора (разъём-то совместим с программируемыми БЗ-34, МК-54 и МК-61). Более интересно, что на корпусе указано напряжение 5 В, но по найденному в интернете паспорту оно равно 5.4 В, а учитывая, что он не стабилизированный, вполне может быть и больше (учитывая, что вместо стандарта 220 В уже давно действует стандарт 230 В). С моим блоком, почему-то,

имеется большая странность. Без нагрузки он выдаёт 8 В (сглаживающий конденсатор заменён). По идее, завышенное напряжение на холостом ходу нормально и должно просесть под нагрузкой. Но при подключенном микрокомпьютере блок даёт около 7 В. Микрокомпьютер работает (с подзавышенным контрастом), но всё же эксплуатировать МК-85 с таким напряжением на постоянную я не решился. Позже поставлю туда какую-нибудь 7806 (5 В аппарату маловато, нужно не менее 5.4 В, а лучше 6), да и защиту от КЗ бы неплохо, предохранителя тут нет. Пока подключил шнур с разъёмом к регулируемому БП.

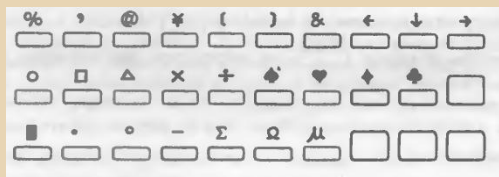


Как уже известно, внешний вид девайса подчистую скопирован у Casio, а потому расположение органов управления и индикации аналогично. В левом верхнем углу находится однострочный графический дисплей с 12 знаками (5x7 пикселей каждое), дополнительной



строкой с индикаторами режимов и четырьмя семисегментными индикаторами для отображения количества свободной памяти при наборе программ. Правее — подсказка по режимам компьютера, переключаемым с применением клавиши-модификатора **MODE**. Доступны режимы: **RUN** (запуск программ, ручные расчёты), **WRT** (редактирование программ), **TR** (режим трассировки программы, команды **TRACE ON** и **TRACE OFF** используются для, соответственно, включения и выключения), **DEG**, **RAD**, **GRA** (режимы представления чисел в градусах, радианах и градах для тригонометрических функций) и **EXT** (расширенный режим ввода для русских букв и спецсимволов). По понятным причинам с дисплея и с подсказки удалены упоминания режима **PRT** — он использовался для печати данных на принтер в FX-700P, но коль в МК-85 недоступен интерфейс — нет и принтера. Остальную часть экрана занимает клавиатура, поделенная на три группы — специальная группа (выключатель, модификаторы **MODE**, **S**, **F**, клавиши горизонтальной прокрутки), цифровая группа (туда же входят клавиши **EXE**, **AC**, **DEL**, **STOP**) и алфавитная группа (буквы, клавиши **SPC**, **ANS**, **=**, **EE**). У большинства клавиш имеются маркировки «шорткатов» команд Бейсика или символов. В зависимости от цвета маркировки, для её набора предварительно нужно нажать модификатор **S** или **F** (маркировки окрашены по их цветам). Справа от клавиш в алфавитном блоке и на клавишах арифметических действий цифрового блока нанесены русские символы. Их ввод осуществляется с помощью режима **EXT**. При этом сам по себе ввод сделан несколько неудобным образом. По стандарту, латиница в микрокомпьютере (подобно FX-700P) вводится большими буквами. Для ввода маленьких букв перед ними нажимается клавиша-модификатор **MODE**. При вводе кириллицы, наоборот, по умолчанию вводятся маленькие буквы, а для больших применяется **MODE**. Отдельно «закон писан» для букв «ъ», «э», «ю», «ё», «ч», «щ»: для ввода больших букв используется нормальный режим и модификатор **MODE**. Для ввода маленьких модификатор **MODE** нужно использовать в

режиме **EXT**. Поэтому ввод слов на русском нередко сопровождается необходимостью «скакать» между режимами, увы. Помимо русских букв в **EXT-режиме** можно вводить и заглавные латинские символы через модификатор **F**, а через модификатор **S** вводятся специальные символы из дополнительного регистра. На МК-85 он не нанесён, для этого используется отдельная прозрачная наклейка с обозначениями (у меня, к сожалению, не сохранилась).



На верхней и нижней гранях микрокомпьютера ничего нет. На правой имеется подстроечный резистор контраста экрана. На левой — разъём внешнего питания от БП (хорошая фишка, которой нет у Casio и прототипной МК-85).

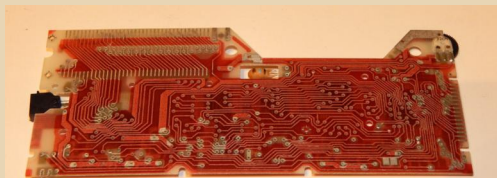
Сзади на крышке находится отверстие кнопки сброса данных и три винтика, один из которых под гарантийной пломбой.



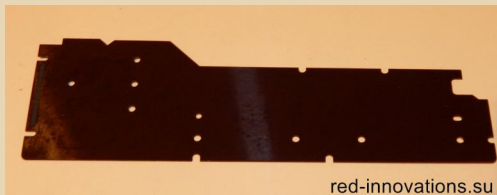


Если их открутить и потянуть крышку сначала вверх, а потом на себя — она снимется и появится доступ к микросхемам компьютера и батарейному отсеку, скрытому за дополнительной крышкой. Как уже упоминалось, в отсеке ставятся 4 серебристо-цинковых батарейки формата СЦ 0,18. Поскольку микрокомпьютер с момента своего производства не вскрывался — сохранилась как пломба на крышке, так и комплектные батарейки в самом компьютере. Судя по дате на рёбрах батареек — все они произведены в августе 1991 года. Микросхемы моего экземпляра были изготовлены в 91-92 годах.

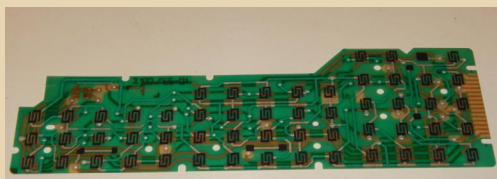
Открутив ещё с десяток винтов (на одном из них пломба), можно снять пластмассовую часть, прикрывающую плату. Таким образом устройство можно полностью разобрать. Срывать пломбу я, впрочем, не стал, поэтому фотографии взяты с сайта red-innovations.su.



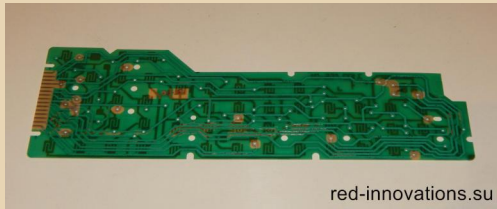
red-innovations.su



red-innovations.su



red-innovations.su



red-innovations.su



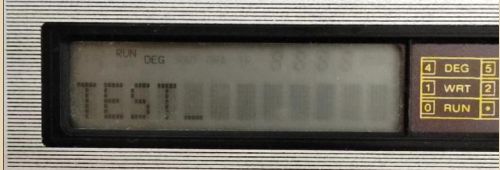
red-innovations.su



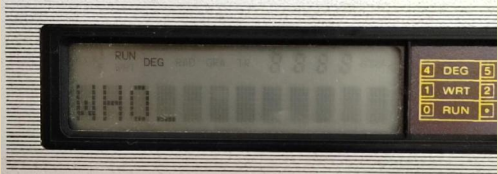
red-innovations.su



Сильно затрагивать программную часть я не буду — особенности работы местного Casio-подобного диалекта Бейсика очень хорошо описывает инструкция по эксплуатации машинки (mk.bs0dd.net/mk85/Lib/manual.djvu). По сути, вид системы стилизован под японский прототип, синтаксис и команды практически аналогичны, однако доступны дополнительные команды: **LETC**, **GETC**, **ASCI**, **CHR**, **DRAW** и **DRAWC**. **LETC** служит для описания начертания пользовательского символа 96. По умолчанию символ пуст, но, вызвав данную команду со специальным кодом в строке (коды описаны в табличке руководства, страница 46), можно изменить его вид. Причём менять можно «на лету»: задаём один вариант начертания, печатаем символ, задаём другой вариант начертания, снова печатаем символ. На экране будут показываться две разных версии, казалось бы, одного и того же символа. Команда **GETC** позволяет выделить из строки символ по его индексу. **CHR** и **ASCI** служат для преобразования десятичного кода в символ и символа в его десятичный код соответственно. Ну а **DRAW** и **DRAWC** нужны для рисования или стирания точки на дисплее (достаточно медленные функции). Помимо этого, есть и «пасхалка» от разработчика — недокументированная команда **WHO**, выдающая строчку «Программу разработал Подоров А. Н.» (youtu.be/QbgjZzhKZ6Y). Есть в микрокомпьютере и команда самотестирования — называется (неожиданно) **TEST** (youtu.be/6N6_nKDYaCs). При её запуске компьютер проверяет работу ОЗУ, ПЗУ и дисплея. При этом при проверке ОЗУ компьютер счищает оттуда всё содержимое, что приводит к потере записанных в памяти программ, так что без надобности вызывать команду явно не стоит. Для Casio, насколько мне известно, тест-программа шла отдельно на кассете.

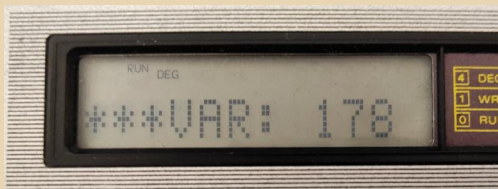


По умолчанию МК-85 запускается в режиме **RUN**, откуда можно запускать хранящиеся в памяти программы (шорткатом **S** + <номер от 0 до 9>) или самому выполнять вычисления, используя стандартные действия и часть команд Бейсика. Помимо этого в качестве ячеек памяти можно использовать и переменные (от **A** до **Z**).





При этом (как в программе, так и в ручном режиме) количество переменных можно увеличить (командой **DEFM**). Выделяются они за счёт памяти для программ, и количество выделяемых переменных ограничено только количеством доступной памяти. Каждая переменная занимает строго 8 байт. В обычном МК-85 можно, таким образом, получить до 178 переменных в общей сложности (для программ останется всего 5 байт), в МК-85M — до 690 (для программ также остаётся 5 байт).



Переменные могут хранить либо число (простое обращение к переменной — **A**), либо строку (обращение через доллар — **A\$**). Числа хранятся в диапазоне от $\pm 10^{-4095}$ до $\pm 9,999999999^{4094}$. Строки же, внезапно, имеют максимальную длину всего в 7 символов.

Связано это как раз с тем, что переменные по 8 байт (64 бита), а потому в таком случае 1 байт уходит под атрибут строки, а остальные 7 и используются под символы. Существует и расширенная переменная-строка — **\$** (да, просто доллар). Она вмещает в себя до 30 символов.



Интересно сделана организация обращения к переменным. Есть два варианта обращения — прямой и со смещением. В случае с прямым просто пишется название переменной — **A** или **A\$**. В случае обращения со смещением указывается переменная, относительно которой идёт смещение, и в скобках — само смещение (в отличие от FX-700P, в МК-85 можно использовать смещения отрицательные) — **A(5)** или **A\$(5)**. Соответственно к шестой по номеру переменной можно обратиться непосредственно как **F**, **F(0)**, **A(6)**, **C(3)**, **K(-5)**. Этот способ позволяет использовать цепочки переменных в микроЭВМ как одномерные массивы. Что касается переменных, выделенных через **DEFM** — они, соответственно, не имеют своих буквенных обозначений, и обратиться к ним можно



только через смещение. Например: **Z(6)** — первая доп. ячейка, **A(28)** — вторая доп. ячейка и так далее.

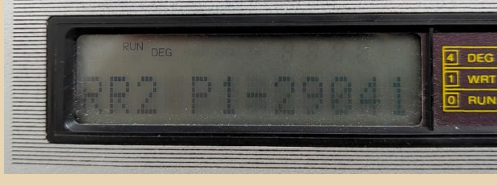
Весьма удобная особенность вычислений микрокомпьютера в том, что он принимает пример не последовательно (когда после выбора каждого действия вычислительное устройство его производит), а единой строкой с примером, которую МК-85 разбирает и выполняет согласно приоритету операторов. Строка ввода в этом девайсе ограничена 63 символами, при вводе 57 символа курсор с нижнего подчёркивания превращается в прямоугольник размером с целое знакоместо, предупреждая, что скоро будет лимит.



Переключаемся теперь в режим **WRT**. Перед нами предстаёт список из 10 файлов-программ, отмеченных цифрами от 0 до 9. Занятые программы вместо цифры имеют знак ромбика. На семисегментных индикаторах показывается количество свободной на данный момент памяти. Как уже говорилось, чистый МК-85 имеет 1221 байт, МК-85М — 5317.



Если МК лежал без питания и содержимое оперативной памяти утратилось — сперва необходимо её инициализировать, нажав кнопку сброса (или можно прогнать **TEST**, но это дольше), иначе микрокомпьютер будет работать неадекватно.



По умолчанию калькулятором выбрана программа **P0** (на ней мигает курсор). Выбирать программу стрелочками (что кажется логичным) тут нельзя, курсор не сместится. Для этого необходимо использовать те самые шорткаты, которые в режиме **RUN** используются для выбора и последующего запуска программы — т.е.



S + <номер от 0 до 9>. Теперь можно начать писать саму программу. Например, простая программа по решению квадратного уравнения вида $ax^2+bx+c=0$ (методом дискриминанта) выглядит так:

```

10 PRINT "ax^2+bx+c=0"
20 INPUT "a",A,"b",B,"c",C
30 D=B^2-A*C*4
40 IF D>0 THEN 60
50 IF D=0 THEN 70
55 PRINT "D<0, x нет":END
60 X=(-B-SQR D)/(2*A)
70 Z=(-B+SQR D)/(2*A)
80 IF D>0;PRINT "x1=";X,"x2=";Z
90 IF D=0;PRINT "x=";Z
    
```

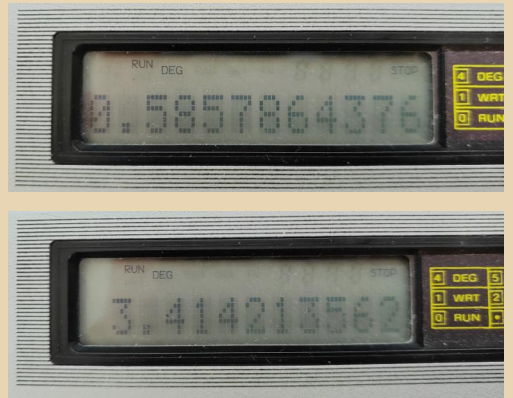
Программа выводит информационную надпись, запрашивает числа **a**, **b** и **c** (при запросе на ввод к указанному тексту автоматически дописывается знак вопроса, поэтому запросы выглядят как «a?», «b?» и «c?») и производит расчёт дискриминанта. Далее, в зависимости от результата, либо считает два возможных решения (дискриминант больше нуля), либо считает одно (дискриминант равен нулю), либо же уравнение решений не имеет (дискриминант отрицательный).

Проверим: решим для начала уравнение $-4x^2+21x+25=0$. Вводим в микрокомпьютер числа -4, 21, 25. Машинка ненадолго задумывается и выдаёт ответ: $x_1=6.25$ и $x_2=-1$.



И действительно, именно при таких значениях x условие уравнения выполняется.

Теперь посчитаем что-то менее «красивое», например $2x^2-8x+4=0$. Получим $x_1=0.5857864376$ и $x_2=3.414213562$. Однако, хорошая точность!



А теперь посчитаем уравнение, имеющее только одно решение: $4x^2-8x+4=0$.
 Ответ, разумеется, $x=1$.



Ну и, конечно же, уравнение, решений не имеющее: $2x^2-4x+25=0$. На что, естественно, получим сообщение о том, что дискриминант отрицателен, а потому решений нет (среди действительных чисел).



Данная программа, само собой, не демонстрирует все возможности микрокомпьютера,



но, как уже упоминалось, подробно останавливаться на этом я не хочу, есть инструкция (по ссылке выше) или, например, вот такая вот книга: «150 вычислительных программ для карманного компьютера "Электроника МК 85"» (mk.bs0dd.net/mk85/Lib/Book/contents.html). В них демонстрируются вполне конкретные примеры для решения некоторых куда более сложных задач, что может свидетельствовать о том, что со своей главной целью (расчёты) МК-85 справлялся более чем прекрасно в то время.

Хотя, считая решение по программе, можно заметить, что во время вычислений микрокомпьютер ощутимо задумывается, скорость его явно невысока. Составляет она, кажется, около 500 кГц, и это действительно немного. Однако такую проблему можно достаточно легко исправить. У МК-85 с завода предусмотрен т.н. турборежим. Если запустить машинку с зажатой клавишей +, она включится с разогнанной частотой порядка 2 МГц. Теперь, конечно, система думает значительно меньше при вычислениях и в целом работает шустро. Например, такая вот простая программа расчёта факториала:

```
10 INPUT "Число", I:A=1:FOR C=1 TO I:A=A*C:NEXT C:PRINT A
```

В случае со стандартным режимом на вычисление факториала 170 компьютер тратит около 48 секунд. В турбо ему достаточно около 8 секунд, т.е. скорость вычислений возрастает в 6 раз! Хочу сразу заметить, что на скорость компьютера влияют две вещи: текущее напряжение питания и резисторы тактового генератора. Чем выше напряжение, тем быстрее будет работать процессор, поэтому нужно не забывать менять батарейки. Ну а встроенный в процессор тактовый генератор, судя по всему, устанавливает частоту на основе установленных на плате резисторов — 68 кОм для обычного и 10 кОм для турборежима. Резисторы неидеальны и могут иметь отклонения по сопротивлению в большую или меньшую сторону. Допуск в большинстве случаев составляет

±5%. Так что не удивляйтесь, если ваш МК-85 считает немного быстрее или медленнее моего.

Но есть с режимом, конечно, и обратная сторона медали. При турбо ощутимо возрастает потребление энергии, так что есть риск очень быстро опустошить установленные батарейки. Поэтому по умолчанию данный режим и не используется. А в инструкции по этому поводу советуют включать турборежим только при подключенном блоке питания.

Умельцами за это время были придуманы методы разгона уже посредством замены номиналов вышеуказанных резисторов. Имеется доработка, когда вместо одного из резисторов и вовсе ставят перемычку, чем заставляют МК-85 ишачить на максимальной скорости. В таком случае, по заявлениям, скорость возрастает ещё в 4 раза относительно турборежима. Теоретически, в таком случае расчёт факториала 170 должен занять всего около 2 секунд. Но и расход батареек тоже будет значительным.

Доработки по микрокомпьютеру проводились не только для ускорения, но и для расширения памяти, улучшения прошивки, а также вывода интерфейса. Эти модификации описаны на польском сайте mk85.republika.pl. К сожалению, с закрытием провайдерского домена republika.pl в 2018 году сайт канул в лету, но есть его архивная копия: <http://web.archive.org/web/20180211174104/http://mk85.republika.pl/index.htm>. У сайта есть и русская версия, но на ней не все материалы сохранились, польская версия более полная.

Другим полезным (и тоже польским) сайтом была страница Петра Пётка (www.pisi.com.pl/piotr433), где была документация не только к МК-85, но и к его прототипу FX-700P, родственнику МК-87 и, вдобавок, к МК-90. К пренеприятнейшему сожалению, ввиду, наверное, известного каких причин, 20 марта 2022 года Пётр удалил со своей страницы все упоминания о советских устройствах. Ведётся реконструкция на основе актуальной страницы и имеемых бэкапов: mk.bs0dd.net/piotr433.



При анализе компьютера обнаружилась и интересная особенность, связанная с недоработкой в прошивке: запуск машинного PDP-11 кода. Да, как и Casio, МК сам по себе не обладает возможностью работать с машинными кодами. Но если в программу вставить оператор **INPUT**, запустить её, во время работы оператора нажать клавишу **AC**, тогда программа приостановится, но функция **INPUT** оставит в слове памяти адрес следующей команды Бейсик-программы, в то время как обычно там хранится адрес подпрограммы, куда передаётся управление по нажатию клавиши **EXE**. Если эту самую клавишу нажать, получится так, что компьютер станет выполнять следующий за **INPUT-ом** код не как Бейсик-программу, а, собственно, как машинный код, т.е. попросту передаст туда управление. В обычных условиях компьютер либо зависнет, либо начнёт выполнять странные действия (может и программы попортить), но энтузиасты научились эту фишку эксплуатировать по-своему. Что правда, толку от этого не сильно много. Код ведь необходимо записывать так, чтобы он был организован как бессмысленная (и неправильная), но Бейсик-программа. Так что дело дальше HELLO WORLD-а, в целом, и не дошло. Хотя как теоретическая возможность она достаточно интересна. Введём следующий код (идея взята с https://alemorf.github.io/retro_computers/Electro_nica_MK_85M/execbinary.html/russian.html):

```

1 INPUT 1111!a
2 !0123456789012
3 !012345678901
2335 П6HELLO=WORLD!=
2399 H6Σ≥e
2527 H♣ā
5599 Aуд6♣≥
    
```

У себя вместо строчки **HELLO=WORLD!=** я забил **ПРИВЕТ=МИР!!=**. Программа вполне корректно работает и с русскими символами, конечно же.

Видеодемонстрация: https://youtube.com/watch?v=wGBwa_kLG70.

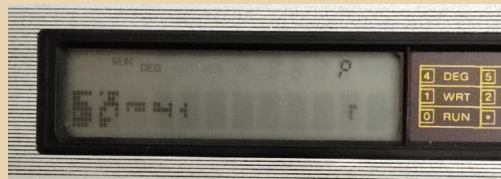


В наше время компьютер без игр представить сложно. И в то время, когда компьютеры стали доступны обычным потребителям, игры тоже становились частью окружения ЭВМ. Под МК-85, разумеется, есть игры. Их, правда, меньше, чем для более распространённых ПМК типа Б3-34 или МК52/61. Но они есть. И, конечно, с матричным дисплеем они выглядят куда приятнее, чем слабоинтуитивные цифры на семисегментной индикации. Все представленные здесь игрушки можно скачать с моего подсайта: mk.bs0dd.net.

Первая подвернувшаяся мне игра называется, ожидаемо, «Тетрис». Где же вы видели советскую ЭВМ, да без тетриса? Даже на ПМК горделили псевдотетрисы. Ну а на МК-85 тетрис вполне себе полноценный. Точный автор используемой мной версии неизвестен. Но программа была когда-то найдена людьми на странице-хранилище Сергея Богданова (**fiks**). Поэтому он является предположительным автором. Файл датирован 2000 годом и содержит в себе как код самой игры, так и заранее рассчитанные значения для ячеек памяти, которые необходимо вбить в компьютер. В файле предлагается ручной ввод, но я предпочёл написать небольшую программку, которая позволяет удобно ввести значения. А потом стирается (иначе памяти обычной 85 не хватит), и вводится уже основной код игры. В документе инструкции о



том, как играть, не прилагалось, поэтому пришлось немного взглянуть в код, чтобы на базовом уровне понять, как игра работает. Для игры желательно микрокомпьютер повернуть вертикально экраном к себе. В таком случае «наверху» появляется случайная фигура и начинает падать. У нас есть несколько секунд, чтобы повернуть и передвинуть фигуру, после чего она упадёт. При этом падает она несколько нестандартным по отношению к обычному тетрису образом. В классической версии можно управлять фигурой до тех пор, пока она не приземлится. Здесь же у нас есть только некая «спусковая зона» (6 разрядов индикатора), и пока фигура находится в ней, управление возможно. Но стоит ей перейти в зону «стакана» (пять разрядов индикатора), она сразу же окажется на дне. В некоторых ситуациях это усложняет игру, поэтому нужно быть внимательным. В свободном 12 разряде после спуска одной фигуры, но перед появлением следующей показываются символы. Сначала можно подумать, что это какая-то бессмыслица. Но по коду стало ясно, что это... индикация очков. Символом сообщается количество набранных на данный момент десятков очков. Определить число можно в соответствии с десятичной кодовой таблицей МК-85, приложенной в инструкции к микрокомпьютеру. Например: символ '{' обозначает 70-79 очков, символ '!' — 330-339 очков и т.д. Проигравш, само собой, наступает при заполнении стакана. Тогда выводится надпись «Game over», а за ней и полное количество набранных игроком очков. За каждую сброшенную фигуру по умолчанию даётся по 4 очка, за каждую неснённую линию — 7 очков. Для такого устройства игра, определённо, получилась достаточно хорошей (и для памяти всего в 1221 байт, из которых код с данными занимает 1196). Видеодемонстрация: youtu.be/gFD-Fip44ss.



Далее игрушка с весьма интригующим названием — DOOM. Да, здесь есть и FPS (First Person Shooter — шутер от первого лица), вот что люди вытворяли! После генерации уровня перед нами предстаёт «интерфейс» — экран самой игры и процент жизни. По уровню «движка» игру, скорее, следовало бы назвать в честь Wolfenstein, т.к. у нас тут простейший одноэтажный лабиринт. Помимо стен есть ещё два объекта — выход (выглядит как стена с черепами) и монстр (выглядит как рожица). Монстры, что примечательно, не стоят на месте, а идут к игроку навстречу (игра идёт в реальном времени, поэтому если игрок ничего не делает, монстр всё равно идёт к нему, а скорость его передвижения задаётся переменной в коде). Столкнувшись с нами, монстр начнёт отнимать здоровье, и чтобы избавиться от него, нужно выстрелить по нему несколько раз (по умолчанию — два). Если здоровье достигнет нуля — игра, конечно же, закончится. Всех монстров в лабиринте убивать не обязательно, главная цель — дойти до выхода. Подойдя к выходу, когда перед нами будет стена с черепами, нужно нажать на выстрел — уровень будет завершён и генерируется новый случайный уровень. В любой момент игру можно остановить, и либо позднее сгенерировать новый уровень, либо продолжить текущий. Сама игра и генератор реализованы в разных программах-файлах (P0 и P1). А генератор у игры, увы, не слишком хороший. Иногда лабиринты получаются удачными, в них есть 3-4 монстра и выход найти не так просто. Но достаточно часто получается так, что выход с уровня генерируется прямо у нас «под носом» либо на уровне нет ни одного монстра (во всяком случае, найти не получается). Главный недостаток — генератор создаёт на уровне больше одного выхода, поэтому если вам повезло, и выход не появился



прямо перед вами, скорее всего, он всё равно будет недалеко и не один. Да, простота генератора обусловлена недостатком памяти в МК-85, и автор, похоже, это прекрасно понимал. Поэтому в тексте описан формат самого уровня (комната 7x12), чтобы написать свой генератор (если у вас МК-85М и места ещё предостаточно) или же закодировать уровень вручную. Ещё небольшой недостаток самой игры — иногда не срабатывают клавиши. Связано это, наверное, с игрой в реальном времени. Цикл игры состоит из опроса клавиатуры, выполнения действия, если зарегистрировано нажатие, и передвижения монстров. Видимо, случается так, что клавиша оказывается нажата в момент исполнения игровой логики, а не опроса клавиши. Из-за особенностей оператора опроса (KEY), если клавиша нажата не в момент действия оператора, нажатие не регистрируется. В остальном игра хороша, какое-никакое 3D, всё-таки. Видеодемонстрация: youtu.be/gDJoXjHsJo.



Не менее интересная игрушка мне попала на просторах YouTube, с канала некоего Андрея (www.youtube.com/@user-km2pw9xv1v). Большинство игр у него, к сожалению, предназначены для версии МК-85М, да ещё и с разгоном через резисторы. Но одна игра для обычного МК-85 всё-таки нашлась. В Auto Rally наша задача рулить машинкой на дороге, полной ям и поворотов в стороны. Ехать по такой дороге нужно, пока на игровом счётчике не наберётся 25 очков, что весьма непростая задача. Надо вовремя поворачивать машину вправо или влево либо перепрыгивать ямы (интересная, конечно, машинка). Важно при этом попасть в «тайминг» и нажимать кнопки вовремя, иначе дело закончится проигрышем. Трасса, по сути своей, бесконечна, и препятствия генерируются на ходу, при этом весьма неплохо. Иногда выходит настолько извилистая трасса, что только успевай нажимать на кнопки! После 25 набранных очков (на что требуется около 4 минут) игра будет считаться оконченной и проиграется небольшая анимация: машина останавливается, за ней бегут два человека (один с камерой, оператор), выходит водитель, жмёт руку первому человеку и начинает махать рукой игроку.

Игру мне, правда, чуть пришлось переделывать. Изменил управление на более удобное на мой взгляд и немного замедлил игру, поскольку у меня исходный код работает быстрее, чем на микрокомпьютере автора, а потому играть было сложно — слишком быстро едет машина. Ну и заодно сделал так, чтобы при перезапуске игры после проигрыша повторно не показывалась стартовая заставка — её прокрутка каждый раз только отнимает время. К сожалению, на камеру набрать 25 очков мне не удалось. Игрой трудно управлять, когда МК-85 не в руках, поэтому первую попытку сыграл сам, как получилось, а для второй уже включил автоматический (демо) режим, когда игра сама управляет машиной и обходит препятствия.

Видеодемонстрация:

<https://youtu.be/H-ISQR3Hr8w>





И, наконец, пожалуй, самая удивительная для меня игра (даже удивительнее DOOM) — «Принц 1221». Да, игрушка напоминает знаменитого «Принца Персии», и автор действительно хорошо заморочился, чтобы уместить в 1221 байт и логику, и карту с дверями, кнопками да шипами. И даже сделал систему ловушек, в которые по незнанию я угодил несколько раз. Игру написал Сергей Бобров (**sb3d**), и, как и в случае с предыдущей игрой, она достаточно новая, из 2014 года: gamedev.ru/flame/forum/?id=192812. После запуска мы появляемся на уровне общим размером 36x4 клетки, т.е. в игре имеется четыре этажа, а 36 горизонтальных клеток разделены на три экрана по 12 клеток (дабы уместиться на экране). Принц может ходить, прыгать, спускаться и подниматься по этажам (есть некоторые нюансы с управлением, которые могут показаться недоработкой, но на самом деле на них завязаны некоторые особенности прохождения игры). Задача предельно проста — разобраться с системой кнопок и дверей и, не угодив на смертельные шипы, добраться до выхода, расположенного в правом нижнем углу (в начале игры закрыт дверью). Но сколь проста задача, столь непросто её решить. Самому пришлось основательно

побродить по карте, упиравшись в ловушки и тупики. Основная проблема: на карте есть и неактивные стены, и активные двери, но выглядят они все одинаково, поэтому после нажатия кнопки надо лазить по карте и смотреть, не исчезла ли какая «стена». Тем более, что некоторые двери открывать не имеет смысла, ведут они либо в тупик, либо в ловушку. Пришлось подумать с денёк, потом, наконец, удалось найти оптимальную последовательность для прохождения — занимает она всего 4.5 минуты. Но поскольку главная суть игры и заключается в головоломке (кстати, больше головоломок для МК-85 я не встречал), сделал две видеодемонстрации: просто броджение по игре и попадание в ловушки (youtu.be/cY3uoHlljw) — для тех, кто просто хочет посмотреть, как оно выглядит, и само прохождение (youtu.be/tGdKTh176rY), для тех, кто не хочет ломать голову или же, наоборот, никак не может сам пройти игру.



Побаловав себя чужими играми, захотелось написать и что-то своё. Простое, не замороченное, и даже скорее не игру, а что-то вроде демки (на игру просто нет времени). Повозился, повозился... и написал простенькую игрушку-кликер «Прогрессбар 85/МК». Задача — тыкать по разным клавишам компьютера, таким образом заполняя прогрессбар до 100%. Изначально отрисовку бара на дисплее я сделал с помощью команды **DRAW**. Но поскольку команда позволяет рисовать только одну точку, для вертикальной линии её нужно вызвать 7 раз, в итоге рисуется всё достаточно медленно, заметно даже в турборежиме. Пришлось, как и многие авторы игр, прибегнуть к махинациям с символом 96 (тот, чьё начертание задаётся командой **LETC**). Символы полностью пустого и полностью заполненного деления у нас уже есть, поэтому достаточно было добавить в память 4 закодированные промежуточные заполненности разряда, а потом, чуть усложнив алгоритм, использовать для отрисовки, фактически, псевдографику. Символы печатаются мгновенно, так что «игрушка» стала работать гораздо быстрее. Выложил её на GitHub: github.com/Bs0Dd/progressbar85mk. Видеодемонстрация: youtu.be/G_kz8KwUTGY.

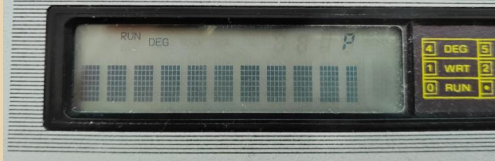


Теперь, после игр, можно подойти и к итоговому вопросу, оставленному мной на закуску — мощнее ли МК-85 своего более старого прототипа FX-700P. Ответ покажется очевидным — ведь у советского «ответа» круче процессор, больше команд в Бейсике, он новее сам по себе. Только вот... в начале статьи, описывая характеристики машинок, я не зря упомянул, что МК «теоретически» превосходит Casio. Вся суть в том, что ещё в начале десятых Сергеем Фроловым проводилось тестирование программируемых калькуляторов (и карманных микрокомпьютеров). Результаты были опубликованы на его сайте: www.leningrad.su/calc/speed.php.

Замеры делались на основе следующего алгоритма: «Есть число 1.0000001, и его 27 раз нужно: умножить само на себя, возводить в степень 2.01, и всё это десять раз подряд, чтобы стала заметна разница во времени. Возведение числа в степень затрагивает многие встроенные математические функции калькулятора, такие как вычисления экспоненты, логарифма, перемножения. Число 2.01 позволяет проверить скорость ввода информации с программы, нормализовать и т. д. Плюс часто используется оперативная память, и её скорость также важна». Для того самого диалекта Casio BASIC (с которым МК-85 не до конца, но совместим) используется следующая программа:

```
5 FOR I=1 TO 10
10 A=1.0000001:B=A
15 FOR J=1 TO 27
20 A=A*A
25 B=B^2.01
30 NEXT J
35 NEXT I
40 PRINT A,B
```

Измеряется как скорость выполнения теста, так и точность полученных чисел **A** и **B**



(номиналом считается 674530.470741084559 и 4669419.98316970777). Casio FX-700P и PB-100 (разницы у них немного, я об этом уже говорил) выполняют этот тест примерно за 45 секунд с числами 674475.396119 и 4669059.87238 соответственно (с учётом двух скрытых разрядов, на экране не отображаемых). При том, что никаких турборежимов у них нет, всё считается на штатной частоте около 500 кГц. А теперь очередь МК-85. В обычном режиме, на сравнимой с Casio частотой программа на нём выполняется... около 915 секунд! Это, фактически, 15 минут!!! Быть может, турборежим что-то исправит? Ну что ж, в нём программа исполняется за примерно 150 секунд, грубо говоря 2.5 минуты. Намного лучше, конечно, но... почему так долго? На выходе у МК получаются числа 674472.441611 и 4669470.08134 (с разрядами аналогично). Вроде бы считают всё примерно одинаково (хотя второе число у МК-85 всё же ближе к номиналу, видимо, степени у него поточнее), но почему 85-ый настолько медленный, что, будучи 16-битным устройством 1986 года, уступает 4-битному из 1982?

Проблема здесь кроется, пожалуй, в программной составляющей. У советских разработчиков не было возможности произвести реверс-инжиниринг программной части Casio. У микрокомпьютера ведь применяется заказной у Hitachi процессор HD61913A01. Архитектура у него какая-то своя собственная, а ПЗУ с прошивкой находится непосредственно внутри чипа, и доступ извне к нему отсутствует. Похоже, и поныне никому пока не удалось разобраться с системой команд процессора и устройством программы Бейсика. В Союзе, конечно, могли заморочиться, спилить корпус процессора, под микроскопом рассмотреть устройство процессора и ПЗУ. Но это требовало времени и денег, а ни того ни другого у команды особо не было. Так что в ход пошла архитектура PDP-11, а прошивку начали делать, копируя внешний вид и поведение Бейсика у FX-700P.

Как утверждал Юрий Отрохов, их отделение работало в основном на оборонную промышленность, поэтому разработка бытовой микроЭВМ для них была процессом нестандартным,

непривычным и побочным. Приведу его же цитату: *«Тут, наверное, нужно понять, чем, по приоритету задач в целом для МЭП и для НИИТТ с “Ангстремом” в частности, являлось производство этих калькуляторов и ЭВМ? И конечно, что из себя, по специальному образованию, представляли люди, занимавшиеся разработкой этих изделий? На мой взгляд, для предприятий МЭП основной задачей было производство комплектующих в основном для предприятий оборонпрома, а калькуляторы и ЭВМ – побочной деятельностью».*

В стране в принципе впервые собирались выпустить продукт такого типа (начальство в итоге, как мы знаем, так и не смогло даже понять, персональный ли это компьютер, микрокомпьютер или и вовсе микрокалькулятор), готовился первый блин из нехватки времени, ресурсов, финансов и много чего ещё. Единственным автором прошивки микрокомпьютера является Анатолий Подоров. Тот самый, из строчки по команде **WHO**. Он же, кстати, ответственный и за прошивку для МК-87, там по секретной комбинации **C + CK + K** показывается та же самая строчка, что и у МК-85. В распоряжении у Подорова имелся ДВК, для которого тот написал эмулятор имеемой платформы (ДВК тоже построен на основе PDP-11, что облегчало задачу). Тут очевидно, что для Анатолия разработка прошивки для микрокомпьютера была делом новым. Наверняка сильно сказалось и ограниченное количество литературы по данному направлению. А по воспоминаниям Отрохова: *«Толик [Подоров] сам разработывал алгоритмы всех функций дома у себя, обложившись кучей томов литературы по этой проблеме, как истинный самодельщик во всём».* Надо ли говорить, что столь обширный объём работы для одного человека в условиях слабой доступности литературы по типу устройств, даже не производимых в СССР, был поистине трудным. В том же Casio над прошивкой наверняка работала группа из десятка (ну или хотя бы пяти) людей. Ну а в перестроечном Союзе... что было, то было.

Следующей проблемой была, откровенно говоря, не совсем хорошая пригодность процессора на базе PDP-11 для построения пусть и



компьютера, но с уклоном в точные вычисления. В FX-700P, как и у других подобных моделей разных производителей, применяли специально созданный под задачу процессор. Пусть устройство команд HD61913A01 нам до сих пор неизвестно, но, скорее всего, он содержал в себе аппаратные реализации жизненно необходимых для микрокомпьютера команд. Не удивлюсь, если и поддержка BCD-чисел (Binary Coded Decimal — двоично-десятичный код, популярен в вычислительных устройствах в связи с лёгкостью вывода на дисплей чисел, отсутствием проблем с точностью при переводе в человеко-читаемый десятичный формат и наоборот и т.д.) имеется тут аппаратно (вроде так нередко делали в процессорах).

Ну а 1801BM2 был процессором общего назначения, имел много ненужных мнемоник и тем более не имел некоторых очень важных. В частности, реализуя работу с BCD, Подорову пришлось делать всё программно, что, как видим, отнимает очень много ресурсов процессора. На самом деле в данной ситуации было бы, может быть, логичнее применять обычные бинарные числа, с которыми процессор работает куда лучше (правда, это чревато ошибками при округлении, видимо, главная цель была в хорошей точности). На форуме iXBT.com обсуждающие проводили теоретические расчёты, по которым использование двоичных чисел и преобразование при надобности в BCD могло бы дать весомый прирост производительности. Но на практике эти идеи проверены не были — поэтому утверждать или отрицать их правдивость я не могу, хотя и надеюсь, что это правда.

Тем не менее, Анатолий реализовал программную BCD-арифметику (по одной из версий, для CORDIC-тригонометрии всё-таки использовались какие-то готовые алгоритмы на основе BCD, и переносить их на двоичную систему было трудно), хотя при этом вычисление экспоненты сделано как раз таки с двоичными числами. Возможно, у него были планы перенести на двоичные числа и другие «функции», но не сложилось. Правду можно было бы узнать только у него самого, но связи у меня с Подоровым, разумеется, нет (буду рад, если кому-то удастся

связаться). Однако проделал он достаточно тяжёлую работу, пусть и далеко не идеально. Не стоит забывать, что тогда он, скорее всего параллельно, работал и над прошивкой для МК-87.

Поводя итог, можно сказать, что первый блин в советском производстве pocket computer-ов вышел, конечно, не комом, но и до блина всё же не дотягивает. Очень жаль, что «блин» этот стал не только первым, но и последним в истории СССР, да и в целом в истории отечественного производства. У населения был хороший спрос на такое относительно дешёвое, компактное и функциональное устройство, несмотря на имеемые недостатки и проблемы с производительностью. Если бы преемника в виде того же МК-95 разработали и смогли выпустить раньше, продолжив тем самым линейку устройств, уже имея опыт в разработке, это был бы поистине крутой микрокомпьютер, который, наверное, мог бы утереть нос и зарубежным представителям этого сектора, но увы, развал, разрыв сети предприятий и последующий кризис, а также приток импорта по факту и закончили на этом все амбиции.

Выражаю благодарность сайту <http://red-innovations.su>, Петру Пётку (www.pisi.com.pl/piotr433, данные о МК с сайта удалены), участникам форума iXBT.com из обсуждения «История отечественных микропроцессоров. (580, 1810 и т.д.)» (forum.ixbt.com/topic.cgi?id=64:2829), участвовавшим там разработчикам Ю. Отрохову, Б. Кроткову, С. Ермакову (им и остальным разработчикам, конечно, спасибо огромное и за сам микрокомпьютер), Б. Малашевичу, журналу «ЭЛЕКТРОНИКА: Наука, Технология, Бизнес» (www.electronics.ru), авторам сайта mk85.narod.ru, сайта mk85.republika.pl, Алексею Морозову (alemorf.github.io/retro-computers/) и Андрею (www.youtube.com/@user-km2pw9xv1v) за материалы по МК-85 и Casio FX-700P. А также Сергею Фролову (www.leningrad.su/museum/), участникам его форума и группы в Telegram — не только за материалы, но и за помощь по некоторым вопросам.

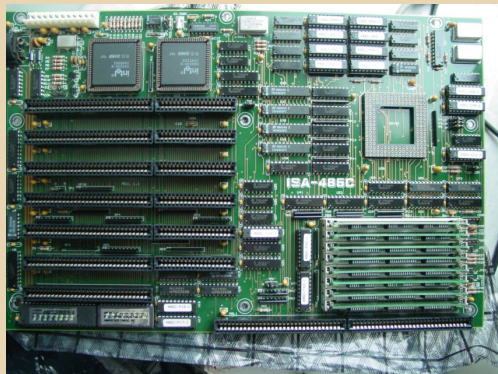


СКРОМНОЕ НАЧАЛО ЛЕГЕНДЫ



P/I-P55T2P4, P/I-XP6NP5, Black Pearl, серии P2B, P3B, TUSL2, P4P800, P5B, P5Q, A8N, TUF, отдельный бренд ROG... – практически в каждом поколении есть прекрасные материнские платы и целые серии плат от Asus, создавшие этой компании репутацию бесспорного лидера на рынке десктопных материнских плат. Сравнимые успехи в области видеокарт сделали её брендом №1 для сборки ПК, а ведь в начале своего существования Asus была лишь одним из десятков тайваньских OEM-производителей, и на первых их продуктах ещё даже не было написано ныне знаменитого имени. И именно с одним из таких продуктов мы сегодня и познакомимся.

Компания Asus была основана в Тайбэе в апреле 1989 года. В том же году самым первым их продуктом стала материнская плата ISA-386С, а в следующем году за ней последовала уже самостоятельно (без участия Intel) разработанная ISA-486 [2]. Очень быстро для неё вышла новая ревизия, получившая кэш 2-го уровня и название ISA-486С. Она и является героиней сегодняшней статьи. Кстати, эти самые первые платы Asus делала для IBM и ALR [1], что само по себе уже весьма солидно.



Итак, что мы здесь видим? Сокет ещё даже не ZIF, проприетарный слот (весьма популярное было тогда решение, только время их быстро ушло), один из слотов ISA (а что, бывают какие-то другие?) ещё восьмиразрядный (что уже редкость на 486-х платах), достаточно плотная компоновка на солидных размеров плате и множество PALок и буферов. Использован чипсет Intel 230/231, вот только на самом деле это не Intel. Сами разрабатывать чипсеты для своих процессоров для массового рынка они начнут на пару лет позже, а здесь стоят лицензированные микросхемы от ZyMOS, больше известные как ZyMOS POACH 1 и 2. И разрабатывался этот чипсет вообще-то для 286-х, однако и с 486-ми работать может с помощью вот этой вот всей кучи «рассыпухи». Не то чтобы очень быстро работать – на vogons.org разбирались с другой платой на этом чипсете и пришли к выводу, что при обращении к ISA-устройствам всегда вставляется 3 дополнительных wait state, потому что так проще было справиться с разницей в работе шин 286-386-х и 486-х процессоров [4].

Ещё интереснее перечислить, чего мы здесь не видим. А именно:

- Типичной для 486-х кучи неподписанных джамперов. Почти все из них на самом деле TP – test point. Для настройки реально используется ровно один джампер – выбор частоты шины 25 или 33 МГц. Всё, больше ничего не надо. Когда эти платы разрабатывались, ещё даже 486SX не выпустили!

- Кстати, PLL на плате тоже нет, выбор частоты – это переключение между отдельными кварцами!

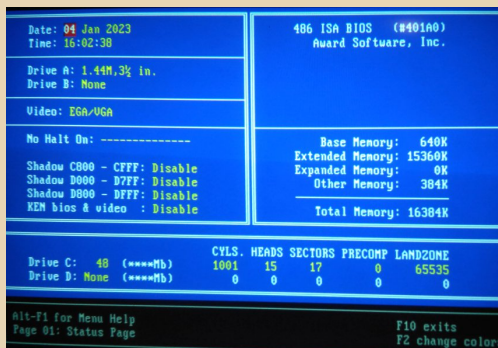
- Аккумулятора тоже нет. Предусмотрен только разъём для внешнего, протечь на плату просто нечему.



- Кэша. Ты кэш видишь? И я не вижу. А он есть. Вот здесь:



В тот самый проприетарный слот вставляется дополнительная плата, дающая нам ещё 8 слотов (то есть два банка) под память и до 256 КБ кэша. Вот только есть нюанс – кнопка Turbo на этих ранних моделях отключает кэш [3]. Поэтому без всего лишь одного джампера на этой плате наблюдаются чудеса – даже самые первые тесты в Memtest86+, которые обычно пролетают быстрее, чем успеваешь что-то заметить, идут секунд по 20 (а от нажатия на клавишу до реакции на него при этом проходит секунд пять), а полностью один проход тестирования всех 16-ти МБ занял всю ночь. И без тестов производительности об этом даже никак не узнать – BIOS проверяет только кэш первого уровня в процессоре. Про второй уровень и объём кэша даже не пишет. Хотя он вообще почти ничего не умеет:



Ранний (совсем ранний) Award BIOS. Версия 4.00, 30 апреля 1990 года. Вход – **Ctrl+Alt+Escape**. Другие клавиши тоже не всегда привычные: выход с сохранением – **F10**, потом **F5**. Видите внизу надпись «Page 01»?

Не верьте, других страниц здесь нет ☹. А ведь даже на 286-х платах примерно 1987-88 годов их две. Этот же setup настолько древний, что даже год в нём переключается в два этапа – отдельно век, отдельно последние две цифры года. И даже экран за собой не очищает – так непривычно видеть загрузку DOS белыми строчками на *синем* фоне! Зато уже умеет выставлять время с точностью до секунд, а не просто обнулить секунды, как было типично в то время.

Возьмём процессор 486DX-33 (самый топ на момент её выхода!) и замерим производительность этого чуда. Компанию ему составит жёсткий диск Seagate ST1144AT на 131 МБ, 3545 оборотов в минуту, 8 КБ кэша (столько же, сколько в самом процессоре!) и видеокарта Macronix MX8600FC. К счастью, BIOS уже позволяет задавать пользовательские типы жёстких дисков, а не ограничен только стандартными, как на многих 286-х.

Checkit, CPU	
С кэшем	Без кэша
15785 Dhrystones (45.89 times XT)	1417 Dhrystones (4.12 times XT)
5703700 Whetstones (864.19 times XT)	788500 Whetstones (119.47 times XT)
Checkit, Video	
19315 chr/sec (28.53 times XT)	2766 chr/sec (4.09 times XT)
208282 chr/sec (41.66 times XT)	17592 chr/sec (3.52 times XT)
Checkit, Disk	
660.3 KB/sec (7.77 times XT)	321.2 KB/sec (3.78 times XT)
Speedsys, CPU	
12.42	2.62
Speedsys, memory bandwidth	
60.06 MB/sec	27.05 MB/sec
Speedsys, L1 cache read	
31.29 MB/sec	–
Speedsys, L1 cache write	
31.37 MB/sec	–



Speedsys, L1 cache move	
39.61 MB/sec	—
Speedsys, L1 cache average	
34.09 MB/sec	—
Speedsys, L2 cache read	
22.96 MB/sec	—
Speedsys, L2 cache write	
31.44 MB/sec	—
Speedsys, L2 cache move	
21.50 MB/sec	—
Speedsys, L2 cache average	
25.10 MB/sec	—
Speedsys, memory read	
16.43 MB/sec	3.20 MB/sec
Speedsys, memory write	
31.44 MB/sec	15.22 MB/sec
Speedsys, memory move	
15.90 MB/sec	2.66 MB/sec
Speedsys, memory average	
21.26 MB/sec	7.03 MB/sec
Speedsys, disk read buffered	
1182 KB/sec	750 KB/sec
Speedsys, disk read linear/max/min	
333/365/301 KB/sec	332/360/302 KB/sec
Speedsys, disk verify linear/max/min	
330/365/295 KB/sec	326/349/301 KB/sec

Вот так один маленький джампер на контактах Turbo может увеличить производительность в различных тестах от 1,5 до 12 раз!

But can it run DOOM?

Лучше не надо. В Wolfenstein поиграйте. Хотя и это не сказать, чтобы здорово получается.

DOOM 1.9c -timedemo demo3 nosound	
С кэшем	Без кэша
13.3 fps	1.4 fps
Wolfenstein 3D timedemo	
21.8 fps	5.3 fps

Следующий эксперимент – а что если оставить только 8 КБ L1-кэша в процессоре, а

дополнительную плату с L2-кэшем (и 8 МБ памяти) убрать? Смотрим:

CheckIt, CPU	
Два уровня кэша	Только L1 кэш
15785 Dhrystones (45.89 times XT)	15785 Dhrystones (45.89 times XT)
5703700 Whetstones (864.19 times XT)	5703700 Whetstones (864.19 times XT)
CheckIt, Video	
19315 chr/sec (28.53 times XT)	19315 chr/sec (28.53 times XT)
208282 chr/sec (41.66 times XT)	208282 chr/sec (41.66 times XT)
CheckIt, Disk	
660.3 KB/sec (7.77 times XT)	625.5 KB/sec (7.36 times XT)
Speedsys, CPU	
12.42	12.42
Speedsys, memory bandwidth	
60.06 MB/sec	60.07 MB/sec
Speedsys, L1 cache read	
31.29 MB/sec	30.97 MB/sec
Speedsys, L1 cache write	
31.37 MB/sec	31.37 MB/sec
Speedsys, L1 cache move	
39.61 MB/sec	39.01 MB/sec
Speedsys, L1 cache average	
34.09 MB/sec	33.78 MB/sec
Speedsys, L2 cache read	
22.96 MB/sec	—
Speedsys, L2 cache write	
31.44 MB/sec	—
Speedsys, L2 cache move	
21.50 MB/sec	—
Speedsys, L2 cache average	
25.10 MB/sec	—
Speedsys, memory read	
16.43 MB/sec	16.42 MB/sec
Speedsys, memory write	
31.44 MB/sec	31.43 MB/sec
Speedsys, memory move	
15.90 MB/sec	15.90 MB/sec



Speedsys, memory average	
21.26 MB/sec	21.25 MB/sec
Speedsys, disk read buffered	
1182 KB/sec	1182 KB/sec
Speedsys, disk read linear/max/min	
333/365/301 KB/sec	333/365/301 KB/sec
Speedsys, disk verify linear/max/min	
330/365/295 KB/sec	330/365/295 KB/sec
DOOM 1.9c -timedemo demo3 nosound	
С кэшем	Без кэша
13.3 fps	12.5 fps
Wolfenstein 3D timedemo	
21.8 fps	21.5 fps

Добавление ещё одного уровня кэша дало всего от 1 до 6% прироста в реальных играх. С другой стороны, более современный и технически сложный DOOM показывает при этом и больший прирост производительности. Синтетические тесты же в CheckIt и Speedsys почти везде показывают абсолютно одинаковые результаты, и только работа с жёстким диском в CheckIt и кэшем первого уровня в Speedsys получают почти незаметный прирост от появления ещё одного уровня кэша – соответственно, 5 и 1–1,5%. А если кому-то хочется впечатляющих цифр – для этого надо включить затенение и кэширование областей BIOSов, это даст аж +66% в тесте CheckIt bios video speed, -0,5 кадра в секунду в Wolfenstein и такую же нулевую разницу в остальных тестах.

А теперь сделаем апгрейд – через пару лет после этой платы вышел легендарный 486DX2-66. С ним всё становится бодрее и чуть более комфортно:

CheckIt, CPU	
С кэшем	Без кэша
28955 Dhrystones (84.17 times XT)	1416 Dhrystones (4.12 times XT)
10549700 Whetstones (1598.44 times XT)	792800 Whetstones (120.12 times XT)

CheckIt, Video	
50608 chr/sec (74.75 times XT)	3583 chr/sec (5.29 times XT)
272369 chr/sec (54.47 times XT)	17592 chr/sec (3.52 times XT)
CheckIt, Disk	
609.5 KB/sec (7.17 times XT)	321.2 KB/sec (3.78 times XT)
Speedsys, CPU	
24.78	2.73
Speedsys, memory bandwidth	
59.81 MB/sec	27.05 MB/sec
Speedsys, L1 cache read	
62.09 MB/sec	–
Speedsys, L1 cache write	
40.19 MB/sec	–
Speedsys, L1 cache move	
39.61 MB/sec	–
Speedsys, L1 cache average	
47.30 MB/sec	–
Speedsys, L2 cache read	
32.67 MB/sec	–
Speedsys, L2 cache write	
40.18 MB/sec	–
Speedsys, L2 cache move	
24.10 MB/sec	–
Speedsys, L2 cache average	
32.32 MB/sec	–
Speedsys, memory read	
21.19 MB/sec	3.26 MB/sec
Speedsys, memory write	
40.19 MB/sec	15.20 MB/sec
Speedsys, memory move	
17.16 MB/sec	2.66 MB/sec
Speedsys, memory average	
26.18 MB/sec	7.04 MB/sec
Speedsys, disk read buffered	
1182 KB/sec	750 KB/sec



Speedsys, disk read linear/max/min	
333/364/301 KB/sec	328/359/304 KB/sec
Speedsys, disk verify linear/max/min	
330/365/295 KB/sec	320/360/287 KB/sec
DOOM 1.9c -timedemo demo3 nosound	
С кэшем	Без кэша
15.3 fps	1.4 fps
Wolfenstein 3D timedemo	
24.7 fps	5.3 fps

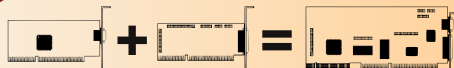
Синтетические тесты на двукратное повышение частоты процессора в основном отозвались повышением показателей примерно на 30%, хотя некоторые тесты и показали ровно двукратное увеличение. Впрочем, из имеющих какое-то значение показателей это только скорость чтения из кэша первого уровня. Скорость перемещения данных в кэше второго уровня и памяти изменилась меньше всего — 8–12% прироста. А вот скорость передачи с жёсткого диска неизменна, а иногда даже уменьшалась — ещё одно свидетельство увеличенных задержек. Зато скорость текстового вывода на экран через функции БИОС возросла аж в 2,6 раза. Тоже практически бесполезный показатель. И о работе без кэша для процессора с внутренним умножением частоты речь уже просто не идёт — в этом случае, если прирост вообще есть, он не превышает 5–8% при формально двукратном увеличении частоты, что вполне ожидаемо. Прирост в играх же составил 13–15%, что, в принципе, соответствует степени ускорения работы с L2-кэшем и памятью.

А как дела со знаменитой асусовской стабильностью? Есть в наличии, так что с этим проблем у Asus не было с самого начала, не то что у многих других из их тайваньских соседей. Единственное мелкое замечание — при установке 486DX2 пришлось отключить затенение памяти и кэширование БИОСа, т.к. с ними

не работал дисковод. Ну, так и появились процессоры с внутренним умножением частоты гораздо позже разработки этой платы. Так что можно сказать, что репутация Asus, заработанная через несколько лет, с самого начала была вполне заслуженной.

Павел Ижевский (pahan)





МНОГОФУНКЦИОНАЛЬНЫЙ КОНТРОЛЛЕР НА VLB-ШИНУ

Практически всю историю PC-совместимых машин те или иные производители комплектующих пытались предлагать комборешения: EGA-видеокарты, имеющие на борту LPT-порт, звуковые карты с модемом, контроллером CD-ROM или FM-тюнером, видеокарты со звуком-огрызком и т.д. Некоторые решения оказывались удачными – например, идея собрать на одной плате порты и контроллеры накопителей оказалась живой, что дало, в итоге, известную всем мультикарту. Не самым плохим решением оказались звуковые карты с контроллерами сидюков и FM-тюнерами. Такие решения тоже заняли определённую нишу.

Но производителей это не останавливало, и на рынок выходили ещё более интегрированные решения. Одно из таких решений – всё в материнку – породило массу бюджетных материнских плат, имеющих встроенную видеокарту, а также звук. Но речь сегодня пойдёт не о них. Оказалось, что существовали контроллеры, включавшие в себя сразу видеокарту и мультикарту. Причём достаточно часто такое решение рассчитывалось на шину VLB. Казалось бы, мультикарта уже в себя включает достаточно много различных контроллеров – IDE (а SIDE ещё и SCSI до кучи), FDD, порты, а тут ещё и видео. Но, как ни странно, подобные комбайны нередко всплывают на аукционах – получается, что, в своё время, находили потребителя, готового приобрести такого «монстра», дабы сэкономить слот на плате.

Мне тоже удалось приобрести одну из таких плат, причём, как оказалось, и видео, и мультикарточная часть выполнены на чипсетах UMC. В статье на «Хабре», посвящённой сборке компьютера на процессоре UMC

(<https://habr.com/ru/articles/409695/>), автор поста утверждал, что найти VLB-видеокарту на UMC сложно, чем объяснил установку ISA-видеокарты на VLB-материнку. А тут спокойно продаётся, хоть и совмещённая с «мультиком», но VLB-видеокарта на UMC. Вот и решил приобрести. Собственно, именно эта плата и станет героем сегодняшней статьи.

Эксперименты по установке проводились на компьютере следующей конфигурации: процессор UMC U55 33 МГц, материнская плата PCChips M912 Rev. 1.3, 16 МБ ОЗУ, MS-DOS 6.21, Windows 3.1 RUS.

Аппаратная часть

Увидеть героиню сегодняшнего обзора можно на рис. 1.



Рис. 1. Внешний вид платы

Как видно, плата достаточно сильно напоминает мультикарту для VLB-шины. Если бы не расположенные в ряд DIP-микросхемы в углу и не БИС UM85C418 – всё же для контроллера IDE великовата ☹. Это как раз и есть видеоконтроллер.

Режимы настраиваются блоком переключателей около двух БИС поменьше. Когда плата пришла, все переключатели были разомкнуты. По запросу ICUVGA-GW148 ничего не нашлось. Поэтому на свой страх и риск воткнул так. Продавец подключал к плате монитор, и изображение выводилось. При такой установке переключателей плата обнаружила и жёсткий диск, и дисковод.



На самой плате распаяны два внешних разъёма – VGA и LPT-порт. Порадовавшись, что видео работает, а с жёсткого диска загружается DOS, начал думать, что делать с COM-портами, ибо они были необходимы для подключения мыши.

Порывшись в закромах, нашёл пару выкидушек с нужными разъёмами COM-портов – DB-9 и DB-25. Из двух собрал одну ☺. Казалось бы, можно подключать, но предыдущий опыт говорил, что нужно проверить распиновку гребёнки. Ваш покорный слуга уже сталкивался два года назад с различными распиновками гребёнок, поэтому сейчас предположил вызвонить.

(DTK / INTEL)				(AT / EVEREX)			
DCD 1	2	DSR	DCD 1	2	RXD		
RXD 3	4	RTS	TXD 3	4	DTR		
TXD 5	6	GTS	GND 5	6	DSR		
DTR 7	8	RI	RTS 7	8	CTS		
GND 9	10	---	RI 9	10	---		

Рис. 2. Два варианта распиновки гребёнок для COM-портов – взято [из темы](#) на «Полигоне призраков»

По итогу прозвонки выяснилось, что разъёмы на выкидушках соответствуют распиновке AT/Everex, а гребёнки платы – DTK/Intel. Последний вариант распиновки используется для современных выкидушек, и один из вариантов был – сходить и купить выкидушку в магазине. Единственное, что останавливало – не нашёл в продаже выкидушку с двумя портами. Но такая выкидушка нашлась в хозяйстве. Прозвонил, поставил – в системе появилась мышь. Теперь дело оставалось за малым – настроить работу видеокарты в Windows.

Программная часть – установка драйвера

В качестве видеочипа на комбинированной плате использован UMC UM85C418. В отличие от распространённых Cirrus Logic GD542x, найти на него драйвер оказалось сложнее. Хотя, в первый раз нашёл какой-то комплект драйверов на VOGONS (<http://vogonsdrivers.com/files/downloader.php?fileid=1471>) – драйвер видео в папке под номером 2, а также драйвер на archive.org: <https://archive.org/details/um85c418>

Тут возник подводный камень, хотя тут ваш покорный слуга сам себе усложнил жизнь. ☹ В общем, почему-то думал, что после экспериментов с сетью на компьютере осталась настроенная связка из Trumpet WinSock и FTP-сервера. Устанавливать и настраивать заново не хотелось. Но Windows была настроена на работу с видеокартой Cirrus Logic (это Win95 уже может самостоятельно сбросить настройки видео на VGA, если для видеокарты не нашлось драйверов, а Windows 3.x – нет). Windows с новой видеокартой запускаться резонно отказалась с предсказуемой ошибкой видеоадаптера. Решено было сбросить настройки на обычную VGA. Для этого была запущена программа настройки винды – C:\Windows\setup.exe. Программа может быть запущена под DOS и предоставляет возможность изменить определённые настройки – в частности видеоадаптер.

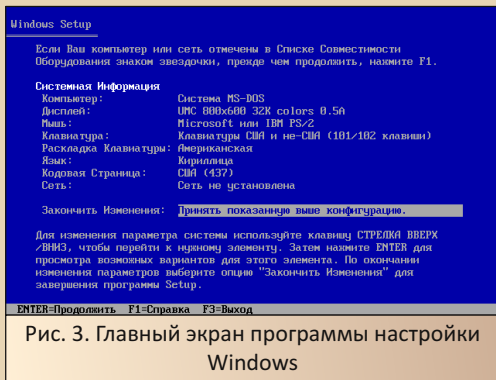


Рис. 3. Главный экран программы настройки Windows

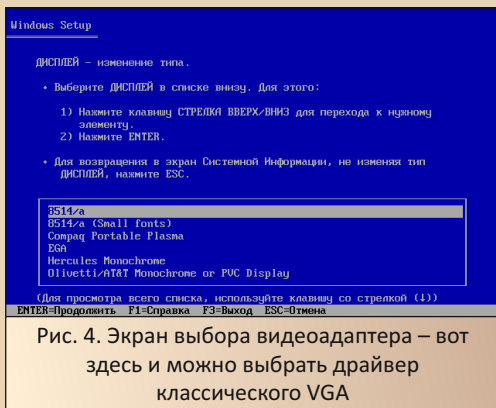


Рис. 4. Экран выбора видеоадаптера – вот здесь и можно выбрать драйвер классического VGA



После установки VGA Windows спокойно загрузилась. Другое дело, что стек TCP/IP установлен не был, так что получилось, что зря. Хотя, пока «воевал» с драйверами, приходилось-таки откатываться на VGA несколько раз, так что опыт пригодился.

По итогу, перед установкой нового драйвера, откатился на чистую Windows, установил и настроил TCP/IP и FTP-сервер для переброски драйвера, перекинул сами драйвера (сперва драйвер с VOGONS, потом с archive.org) и приступил к установке.

Интересно, что в инструкции к драйверу (по крайней мере, с archive.org) написано, что его необходимо устанавливать через диалог, вызываемый иконкой **Windows Setup** в группе программ «Главная». Установку с помощью идущего в комплекте инсталлятора рекомендуют делать только с дискеты.

Драйвер с archive.org мне показался более стабильным, чем с VOGONS – возможно, из-за конкретной железки – не знаю. Хотя и у этого драйвера на экране небольшой зелёный отлив, но цвет не плавает. В общем, остановился, в итоге, на нём. Кроме того, в составе драйвера присутствует утилита **usetup.exe** (см. рис. 5).

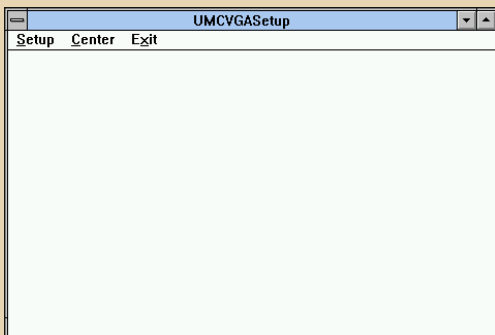


Рис. 5. Главное окно утилиты настройки видеокарты

В случае установки с дискеты или, в принципе, посредством собственного инсталлятора драйвера, эта программа переносится на диск C: и создаётся группа программ с иконкой запуска этой утилиты. В случае установки средствами Windows вас просто сразу спросят о разрешении и количестве цветов – и всё.

Здесь же, при нажатии пункта **Setup** меню программы, откроется диалог настройки разрешений (см. рис. 6). Кстати, тут присутствует возможность включения и выключения режима **Interlace** – интересно, что оно делает?

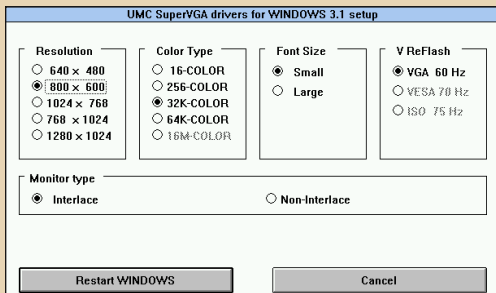


Рис. 6. Диалог изменения настроек разрешения и цветности в утилите настройки

У драйвера с VOGONS в диалоге настроек можно установить частоту развёртки монитора, но, в моём случае, изменение частоты мало чему помогло. ☺

Заключение

Плата оставила двойственное ощущение – с одной стороны, интересное устройство, совмещающее в себе мультикарту и метровую видеокарту для VLB-шины, с другой – показалось, что плата работает несколько нестабильно и есть небольшой зеленоватый отлив, хотя с ним можно вполне работать. Возможно, что стоит и в самом компьютере покопаться – посмотреть – вдруг что-то отходит.

Но пока что плата остаётся на своём месте.

Кстати, такая плата может быть интересна и при сборке UMC-only компьютера, по типу того, что собирал автор поста на «Хабре» – в моём случае не хватает только соответствующей сетевой карты, а так – уже и видео UMC-шное имеется ☺.

Андрей Шаронов (Andrei88)





ЗАХВАТИЛ, УВИДЕЛ, ПЕРЕВЁЛ!



Предисловие

Сел я как-то поиграть в инди-игру с законом под ретро-RPG – Legends of Amberland – The Forgotten Crown (обзор от Podcast Old-Games.RU: https://www.youtube.com/watch?v=37pav_SZJEU). Играю, но нутром ощущаю, что что-то в ней не так, чего-то мне не хватает. Слишком легко и поверхностно, не хватает глубины. В Might and Magic (я писал про них в статье...) даже выйти из стартового города – уже приключение, а потом... И подземелья с ловушками, и текстовые загадки, и многоуровневая прокачка, и ещё куча всего. Есть только у Might and Magic один существенный минус. Из классических частей переведены на великий и могучий только две из пяти (а если посчитать World of Xeen и фанатский аддон Sword of Xeen, то вообще из семи), а именно первая и четвёртая. Английский я более-менее знаю. По работе часто приходится читать мануалы и прочий технический текст. Но то технический текст, а это – художественный. Если напрячься, то читать, конечно, можно, но приятного мало, а хочется именно играть, а не постоянно заглядывать в словарь за неизвестными словами.

Где наша не пропадала. Для телефона есть приложение «Google Объектив», которое переводит текст на том, что видит камера в реальном времени. Но, в самом деле, не ставить же телефон перед монитором на штатив и не смотреть же через него.

Я стал искать приложения под Windows, которые смогли бы мне помочь.

Google Lens есть и в браузере Chrome, но там его функционал очень сильно ограничен и для игр не подходит от слова совсем. Эта версия заставляет совершить целую кучу лишних действий, просто чтобы перевести обычную картинку.

Существует достаточное количество других переводчиков. Некоторые из них даже

встраиваются в движок игры и производят перевод, так сказать, изнутри. Для этой статьи я отобрал только три программы. Переводчики игр на Unity отпадают, т. к. это уже не downgrade. Какие-то переводчики платные, какие-то неудобные, какие-то вообще не работают. Популярная программа Screen Translator тоже в статью не попала. Она оказалась очень громоздкой в настройке, медленной и неудобной.

Дальнейшую статью я построю следующим образом. Вначале я кратко расскажу про каждую из программ и про её заявленные возможности. Затем поведаю, на чём буду проводить тесты, и уже потом перейду к обкатке переводчиков в боевых условиях.

Кратко о переводчиках

«Ёлочка» (<https://perevod-teksta-s-ekrana.blogspot.com/>).

Это самая простая программа распознавания текста с экрана, которую я вообще когда-либо видел. После запуска она прячется в области уведомлений в виде маленькой иконки.

Настройки программы вызываются нажатием правой кнопки мыши на этой иконке.

Из настроек доступны только выбор одного из трёх вариантов горячей клавиши, выбор размера шрифта и цвета фона. В настройки лезть не обязательно. Сразу из коробки «Ёлочка» работает вполне сносно.

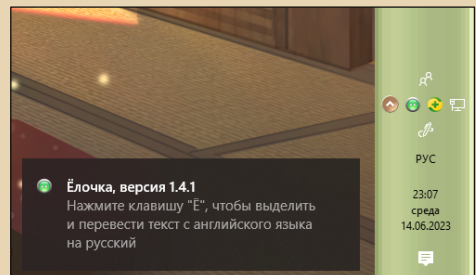


Рис. 1. Такое окошко с инструкцией появляется сразу после запуска



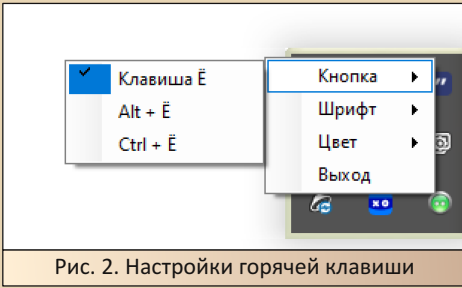


Рис. 2. Настройки горячей клавиши

Для запуска распознавания нужно нажать клавишу «Ё», она же «~». Затем выделить рамкой область, которую нужно распознать, и через несколько секунд на её месте появится переведённый текст. Если клавишу «Ё» подержать подольше, будет автоматически распознана предыдущая область.

Несколько последних распознанных областей запоминаются программой, и между ними можно переключаться колёсиком мыши. Алгоритм в этом случае такой. Нажимаем «Ё», крутим колёсико, пока не увидим на экране искомымую область, снова нажимаем «Ё», читаем результат.



Рис. 3. Результат перевода

MORT (<https://blog.naver.com/killkimno/223051568559>).

Эта программа, в отличие от предыдущей, уже имеет намного более обширный набор возможностей.

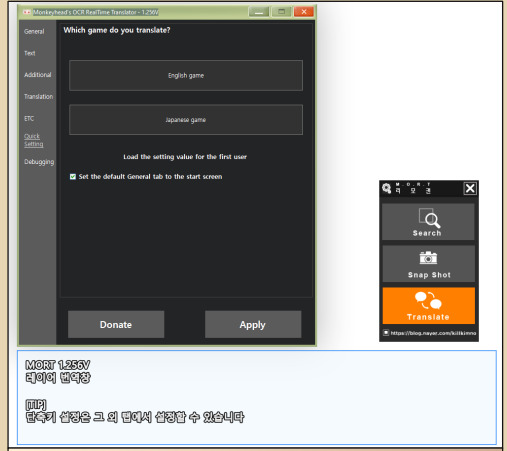


Рис. 4. MORT во всей своей красе

Рабочее пространство этого приложения представляют три окна: окно настроек (большое), окно быстрых операций (маленькое), окно с переводом (снизу).

Сразу после первого запуска приложения крайне желательно произвести первичную настройку, нажав одну из двух кнопок: **English game**, **Japanese game**. Эти кнопки вызывают визард с набором простых вопросов, по прохождении которого останется только нажать на кнопку **Translate** в окне быстрых операций, и MORT начнёт свою работу.

Основная «фишка» этого переводчика в возможности работы в реальном времени. Приложение само сканирует указанные области, само переводит и само же выводит результаты на экран.

MORT поддерживает 4 движка распознавания изображений: Tesseract, Win OCR, NHOCR, Google. NHOCR понимает только японский, поэтому отпадает. А Google работает только в режиме ручного захвата части изображения.

Движков перевода MORT тоже знает несколько штук: Naver, Google, DeepL. Из них нам подходят только последние два. Naver не умеет переводить на русский.



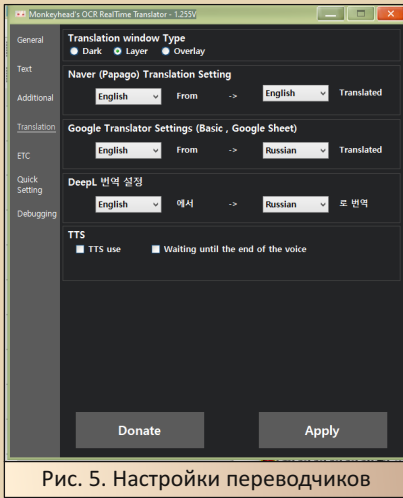


Рис. 5. Настройки переводчиков

На этом возможности приложения не заканчиваются. Результаты перевода можно выводить тремя разными способами: в обыкновенном отдельном окошке, в полупрозрачной панельке (как на скриншоте выше), которую можно повесить поверх игры, и режим **Overlay**. В этом режиме программа старается разместить каждый переведённый текст поверх оригинального. Этаким аналог Google Lens.

Maverik (<https://maverikochelper.blogspot.com/>).

Эта программа от того же автора, что и у «Ёлочки». По сути это сильно расширенная версия «Ёлочки». Точнее, наоборот. Maverik был разработан сначала, а «Ёлочка» является его сильно упрощённой версией для тех, кому не нужны все навороты оригинала.

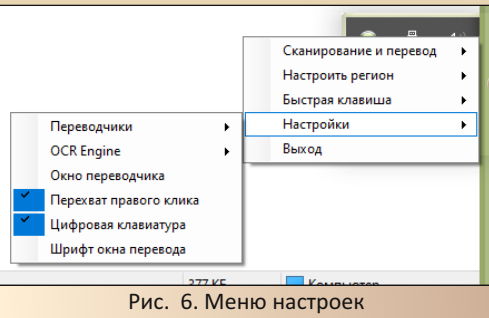


Рис. 6. Меню настроек

В плане интерфейса эта программа практически идентична «Ёлочке», только меню, соответственно, побольше. А в плане возможностей практически идентична MORT'у. Тут есть также два OCR-движка – Win OCR и Tesseract – и два движка перевода – Google и DeepL. Maverik не умеет переводить в реальном времени. Захват области экрана и перевод осуществляется только по горячей клавише. Вместо этого Maverik может запомнить до 10-ти отдельных областей экрана и захватывать их отдельно друг от друга. По умолчанию на каждую из областей назначается соответствующая цифровая клавиша основного блока клавиш.

Обработку нажатия клавиши можно отключить нажатием клавиши **Scroll Lock**. Это на случай, если нужны цифровые клавиши. (В MM цифрами вводится количество денег и драгоценных камней во время банковских операций.)

Есть возможность перенести их на дополнительную цифровую клавиатуру, а также задействовать правую кнопку мыши. Нажимаем ПКМ, на экране высвечиваются контуры всех областей с цифровыми метками, кликаем на нужную, получаем результат. ПКМ применима в случае, если неудобно пользоваться клавиатурой. Например, играем одной мышкой или нажатия на клавиши как-то мешают игре.

Результат перевода выводится в отдельном окошке без прозрачностей и всяких наворотов. Всё строго и чинно.

Об OCR-движках

Теперь пару слов об OCR-движках, упомянутых выше. **Win OCR** идёт в комплекте с Windows, начиная с Windows 10. Ничем примечательным не выделяется. Качество распознанного текста среднее. Со сложными шрифтами работает не очень хорошо, зато имеет малое количество ложных срабатываний. Можно выделять для сканирования хоть весь экран. Win OCR вернёт только текст, который сможет разобрать.

Tesseract – это Open Source движок. Очень популярный и довольно мощный. Распространяется как в виде исходных кодов, так и уже



собранных бинарников. Качество распознанного текста на несколько порядков выше, чем у Win OCR, но настраивать его сложнее, т. к. Tesseract часто за текст принимает игровую графику. В результате в тексте появляется мусор, а это напрямую влияет на качество перевода. При работе с Tesseract нужно очень аккуратно выделять области с текстом и тщательно настраивать предобработку изображения (перевод в чёрно-белый режим, цвето- и тонокоррекцию). И MORT, и Maverik это умеют и хорошо справляются.

Сам этот OCR-движок основан на нейронной сети, и как следствие, его можно обучать новым языкам и шрифтам. Но его не миновала болезнь многих Open Source проектов. А именно документация. Я несколько вечеров пытался дообучить Tesseract правильно понимать символы из игры. Какого-то GUI для этого нет. Нужно вызвать ряд консольных приложений, каждое из которых имеет до десятка разных параметров. Официальная документация довольно скудная. Более-менее толковые статьи на эту тему есть, но они уже устарели и к текущей версии не применимы. Плюс ко всему, все они рассчитаны под «Линукс». В общем, стандарт открытого ПО. Поседеешь, покуда разберёшься.

Благо с последними обновлениями и моделями в «аккуратном» режиме Tesseract с игровыми шрифтами справляется очень неплохо, но об этом позже.

Об API перевода

Пара слов про движки перевода. Движок перевода Google все знают, а движок DeepL менее популярен, хотя качество его перевода мне понравилось больше. Есть пара нюансов. Для работы с DeepL нужен браузер Microsoft Edge, следовательно, опять не ниже Win10. И оба движка имеют ограничения на количество обращений. Google после 100 переводов за один час начинает переводить с худшим качеством. У DeepL примерно та же ситуация, только он перестаёт переводить вообще.

Тестовый стенд

Все тесты я буду проводить на современной машине, в Windows 10, на игре Might and Magic III: Isles of Terra, запущенной в DOSBox 0.74, работающем в окне. В качестве режима масштабирования я выбрал **Advanced inter-polling with factor 3 (advinterp3x)**. Опытным путём было выяснено, что со сглаженной этим режимом картинкой OCR-движки справляются лучше всего.

Да, если вам не интересна M&M, можете смело представить вместо неё любую другую интересующую вас игру – хоть квест, хоть RPG... К сожалению, до сих пор огромное количество действительно интересных игр так и остаются непереверждёнными.

Чем хороша M&M в плане тестирования переводчиков?

По пунктам:

1) DOSBox усложняет работу с горячими клавишами. Он тоже может их распознавать и пролистывать текст. В некоторых случаях игра ожидает нажатия определённых клавиш Y/N, а в некоторых реагирует на любые.

2) Текст на экране выводится в разных областях в разные моменты игры. Всего таких областей 5 штук, но чаще всего текст выводится в двух.

3) Шрифт в игре стилизованный, а не обычный типографский. Литеры «m», «a» и «w» имеют очень короткие хвостики. Из-за этого «m» легко путается с «n», «a» с «o», а «w» с «u».

4) Игровые тексты часто стилизованы под рестоанродную или староанглийскую речь, со всякими «O'», «Ya» и прочим...

Отсюда резонно вытекает ряд пожеланий:

1) Текст должен распознаваться и переводиться автоматически, без вмешательства пользователя. Либо использовать такие горячие клавиши, которые не будут влиять на игру.



2) Движок распознавания текста должен быть очень мощным и отделять текст от интерфейса, либо должна быть возможность настройки нескольких областей захвата с переключением между ними.

3) Программа должна более-менее понимать хитрые шрифты.

4) Переводчик должен понимать подобные обороты и игнорировать или адекватно переводить их.

5) Простота и удобство в использовании. Меньше всего во время игры хочется отвлекаться на лишние действия для получения перевода.

6) Бесплатность.

Тест 1 – удобство использования

Переходим к тестам. Первым делом проверим, насколько вообще эти программы удобны в работе со связкой DOSBox + M&M3. Качество распознавания и перевода на данном этапе не играет роли.

«Ёлочка» — как уже писалось выше, умеет работать только с клавишей «~», она же «Ё». Из этого вытекает невозможность перевода куска текста, когда игра ожидает нажатие любой клавиши. Она просто воспринимает нажатие на «Ё» как AnyKey и пролистывает текст дальше. Второе неудобство – в выделении текста рамкой. Как оказалось на практике, про долгое нажатие клавиши «Ё» для пересканирования области часто забываешь. Про выбор одной из запомненных областей при помощи колёсика я вообще молчу. Третье неудобство — задумчивость. Иногда она может думать над текстом по 10 — 20 секунд, что уже очень долго.

Вывод: для DOSBox и старых игр «Ёлочка» подходит слабо. Хотелось бы больше автоматизации. Вот для перевода текста с картинки, документа или куска сайта «Ёлочка» — самое то.

За удобство начисляю «Ёлочке» 1 балл из 5-ти.

MORT — по сравнению с лаконичной «Ёлочкой» кажется монстром. Интерфейс в несколько окон, не до конца переведённый на английский с корейского, поначалу реально пугает, но если один раз сесть и разобраться, то в дальнейшем он уже не кажется таким громоздким.

MORT позволяет заранее настроить несколько областей захвата текста и в дальнейшем не выделять каждый раз текст рамкой. К сожалению, тут есть пара небольших минусов. Первый — области сохраняются относительно экрана, а не окна с игрой. Соответственно, каждый раз при запуске DOSBox его нужно стыковать с настроенными областями. Второй — захват идёт из всех областей одновременно.

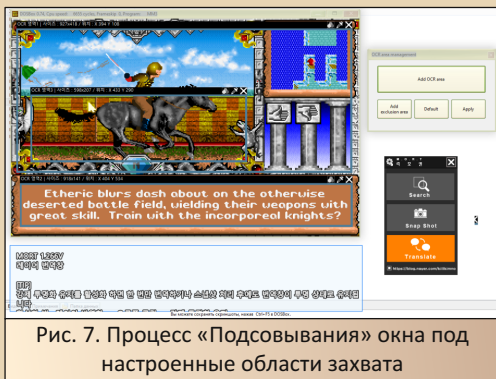
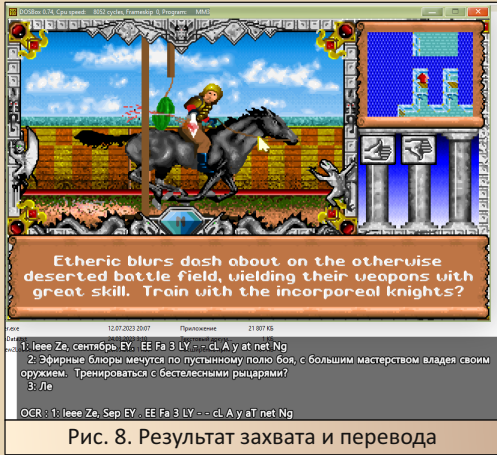


Рис. 7. Процесс «Подсовывания» окна под настроенные области захвата

Однако огромным плюсом выступает возможность работы MORT в режиме автозахвата и возможность включения/выключения этого режима по горячей клавише.

Я выработал для себя такой алгоритм. Заранее настраиваю все области, в которых может выводиться текст. В игре подхожу к месту, где потенциально будет выводиться текст. Включаю перевод. По умолчанию это **Ctrl + Shift + Z**. Читаю, точнее, сначала нахожу среди мусора нужную мне область, а затем читаю. После выключаю автоперевод и играю дальше.





На скриншоте можно заметить, что у меня настроены 3 области поиска текста. В двух из них мусор, и только в одной – осмысленный текст.

Вердикт: почти удобно. Если бы можно было выбирать область, которая мне нужна в данный момент, и сами области сохранялись относительно окна программы, а не экрана, было бы идеально.

За удобство начислю MORT 4 балла из 5-ти.

Maverik – забегая наперёд, эта программа оказалась самой удобной в использовании.

Сразу после запуска программы появляется окно с просьбой навести курсор на окно, с которым **Maverik** будет дальше работать, и нажать «Пробел». После закрытия переводимой программы автоматически закрывается и **Maverik**.

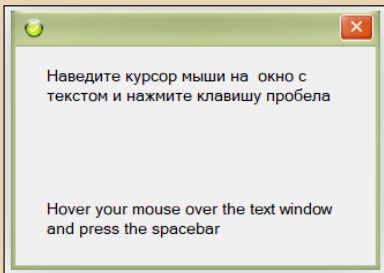
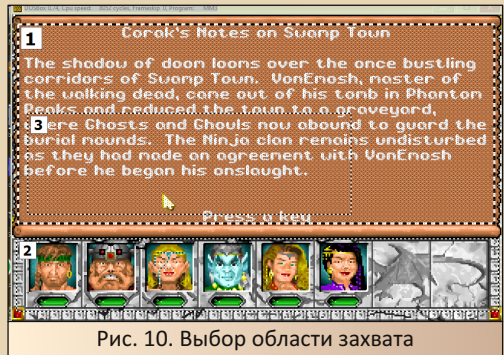


Рис. 9. Вступительное окно

Тут я, наверное, повторюсь, но вышеописанные особенности **Maverik** в плане интерфейса можно записывать в плюсы удобства.

Maverik имеет такой же минималистичный интерфейс, что и «Ёлочка». После запуска сразу прячется в области уведомлений. Своего окна программа не имеет, а все настройки производятся через меню, вызываемое кликом по значку. **Maverik** умеет сохранять до десяти областей поиска текста. Области эти позиционируются относительно области приложения, а не экрана. Каждая из областей закрепляется за своей цифровой клавишей. Соответственно, можно сканировать только ту область, которая нужна, тем самым экономить обращения к API переводчика. Если нужно ввести что-то с клавиатуры, то сканирование отключается клавишей **Scroll Lock**. Ситуация, когда игра ожидает нажатия любой клавиши, тоже обходится. **Maverik**, по нажатию на правую клавишу мыши, выводит поверх окна игры все области в виде рамок. Далее нужно кликнуть левой кнопкой по нужной области и программа её обработает.



В плане интерфейса **Maverik**, можно сказать, идеален. Заслуженные 5+.

Тест 2 – захват и перевод

Как я уже писал выше, на рассмотрении у нас два OCR-движка: **Win OCR** и **Tesseract**. Первый во всём, кроме ложных срабатываний, похуже, второй получше. Чтобы долго не расписывать кто и как обрабатывает тест, просто



приглядитесь к следующим трём скриншотам. Захваты были произведены в программе MORT, сначала движком Win OCR, затем Tesseract.

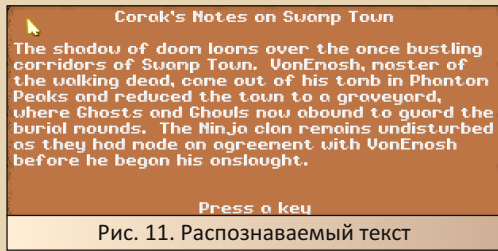


Рис. 11. Распознаваемый текст

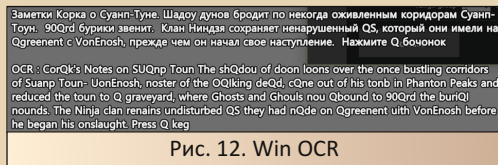


Рис. 13. Tesseract

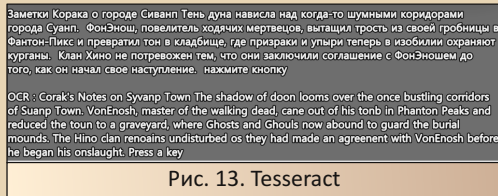


Рис. 13. Tesseract

Вверху 12-го и 13-го скриншотов находится переведённый текст, а снизу то, что выдал OCR-движок. Win OCR захватывает текст очень грязно. Часть текста распознана настолько некорректно, что её даже переводчик не смог перевести. В переведённой части всё равно встречаются неправильно распознанные символы. Tesseract распознал текст в разы лучше, хотя всё ещё видно, что со сложными символами типа "w", "m" он справился не везде.

API перевода в нашем распоряжении тоже два. Google и DeepL. DeepL выдаёт текст более приятный глазу, но при прочих равных разница с Google минимальна.

Второй тест будем проводить на одном и том же игровом тексте, и попробуем его захватить и распознать каждой из программ.

«Ёлочка» — как самая простенькая, умеет работать только с Win OCR и Google, соответственно, и качество захвата оставляет желать много лучшего.

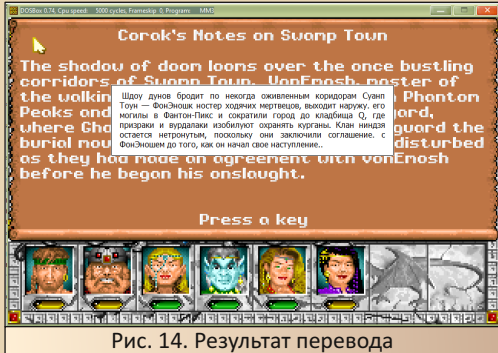


Рис. 14. Результат перевода

2 балла из 5-ти.

MORT – в плане разнообразия настоящий монстр. Два режима захвата вы уже видели на скриншотах 12 и 13. Но MORT умеет работать ещё в одном режиме. По умолчанию Tesseract захватывает в режиме Speed. Так захваты производятся быстрее и потребляют меньше ресурсов, но если этот режим выключить, то Tesseract переключается в аккуратный режим.

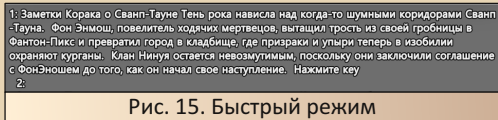


Рис. 15. Быстрый режим

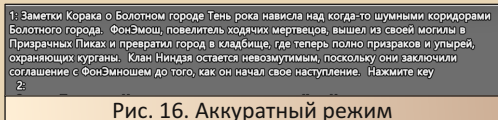


Рис. 16. Аккуратный режим

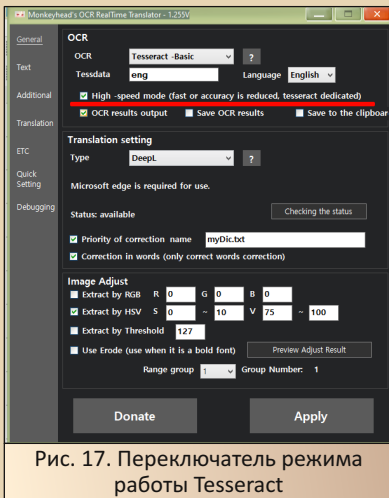


Рис. 17. Переключатель режима работы Tesseract



1: Записки Корака о Болотном городе. Тень судьбы нависла над некогда оживленными коридорами Болотного города. ФонЭмош, повелитель ходячих мертвецов, вышел из своей гробницы в Призрачных Пиках и превратил город в кладбище, где теперь обитают призраки и упыри, охраняющие могильные курганы. Клан ниндзя остался невредим, так как заключил соглашение с ФонЭмошем до того, как тот начал свое наступление. Нажмите кнопку клавиатуры L5.

Рис. 18. Тот же текст, переведённый DeepL

Как видите, практически все сложные буквы распознаны корректно. Забавно, но такой важный переключатель запрятан довольно далеко и ничем особо не выделяется.

Ещё MORT умеет работать с API переводчика DeepL. 15-й и 16-й скриншоты переведены Google, 18-й распознан в аккуратном режиме и переведён DeepL. Разница с «Гуглом» у этого переводчика небольшая, но сам текст стилистически более приятен.

За перевод и захват можно ставить MORT твёрдую 5 с плюсом.

Maverik – как и MORT, умеет работать с Win OCR и Tesseract, и результаты захвата текста у Maverik'a примерно такие же, как и у MORT'a.

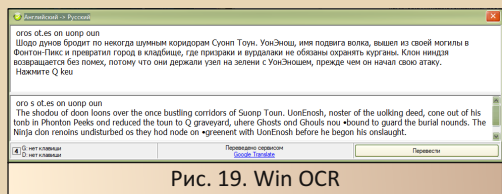


Рис. 19. Win OCR

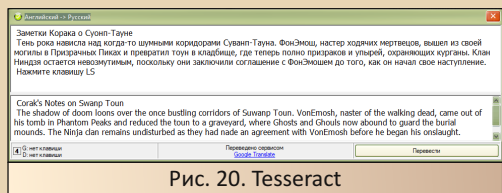


Рис. 20. Tesseract

К большому сожалению, у Maverik нет переключателя режима работы Tesseract, и он всегда работает в быстром режиме.

Движок перевода у версии 3.0.2 заявлено тоже 2 штуки. Всё те же Google и DeepL. Но вот незадача. Последняя версия «Маверика» вышла уже достаточно давно, и с тех пор у DeepL

что-то изменилось. Во-первых, перевод из этого API выводится в отдельное окошко браузера, которое запускается вместе с «Мавериком». Во-вторых, работа с DeepL крайне нестабильна. Программа может зависнуть или просто вылететь. От чего зависит стабильность работы с DeepL – я выяснить не смог. В данный момент, когда я пишу этот текст, все тестовые захваты прошли отлично и не было ни одного зависания, но месяц назад Maverik стабильно вылетал на втором-третьем переводе.

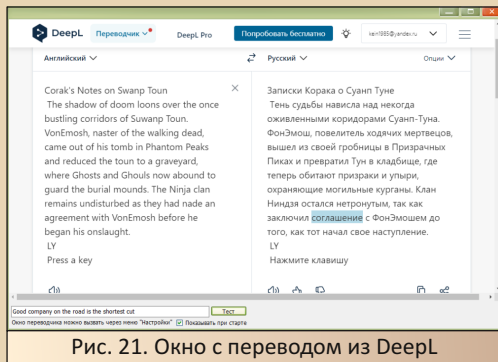


Рис. 21. Окно с переводом из DeepL

На странности работы с DeepL можно закрыть глаза. «Гугл» переводит тоже вполне неплохо. Но работа Tesseract только в быстром режиме огорчает, тем более при таком удобном интерфейсе.

Ставлю 4 балла.

Тест 3 – прочие интересности

«Ёлочку», как самую простую, мы сразу пропускаем. Там дополнительных фишек нет.

А вот MORT имеет режим вывода переведённого текста прямо поверх оригинального. Это очень интересная фишка, но применима она далеко не везде. Во-первых, в играх бывают сокращения слов, которые правильно не переведёшь, особенно в RPG, во-вторых, этот режим работает только с Win OCR, а вы уже знаете о качестве его работы со сложными шрифтами.



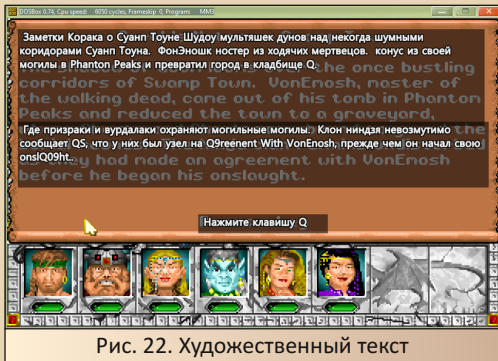


Рис. 22. Художественный текст

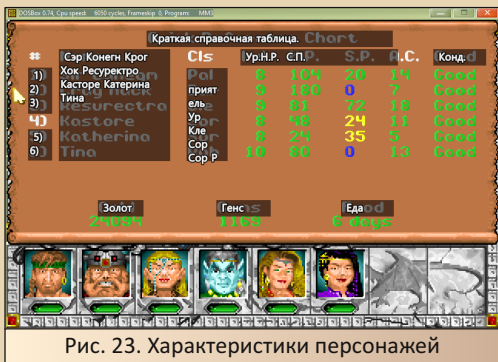


Рис. 23. Характеристики персонажей

В моём случае этот режим оказался не только бесполезным, но и вредным. Выводимая абракадабра закрывала собой оригинальный текст, и становилось вообще непонятно, что имеется в виду. Но для шрифтов попроче и для большого количества художественных текстов, как, например, в визуальных новеллах или квестах, этот режим должен работать на отлично.

Ещё одна интересность: MORT использует Tesseract как часть своего исполняемого файла, а Maverik – как отдельную dll-библиотеку. При желании её можно поменять на более свежую версию, что я первым делом и сделал. Однако в моём случае это не сказалось на качестве захвата текста ни в какую сторону.

В обеих программах файл с данными обучения Tesseract английским текстам лежит

отдельно, в директории **tessdata**. Файл называется **eng.traineddata**. Так вот, этот файл также можно свободно заменить на свой. Допустим, дообучить какому-то хитрому шрифту. Но, как я уже писал в начале статьи, разобраться в процессе обучения сложно, ибо документация разрознена и не везде актуальна.

Итоги

В итоге претендентов на победу два. Maverik явно выигрывает по простоте использования, удобству интерфейса и, теоретически, расширяемости. А MORT – по качеству захвата, перевода и дополнительным фишкам. Если смотреть субъективно, то по мне качество захвата – более важный критерий, чем удобство использования.

Получается, что MORT выигрывает. Он подходит для большего количества ситуаций. Им можно пользоваться в тех случаях, когда игра считывает клавиатуру и перелистывает текст, включая режим захвата в реальном времени, а для экономии обращений к API этот режим можно просто выключать, когда не нужен. К минусам можно отнести запутанный интерфейс и необходимость перестраивать области распознавания под каждую отдельную игру. Это особенно неудобно, если вы играете параллельно в две игры.

Однако Maverik со счетов списывать нельзя. В играх, где шрифт попроче и быстрый режим работы Tesseract справляется, удобство работы выходит на первый план.

Лучше держать на компьютере обе и использовать ту или иную по ситуации.

Валерий Сурженко (Hippiman)





PICTURE MAN 3.0 для WINDOWS 3.1



рафическому редактору Picture Man 3.0, а вернее комплексу программ Picture Man 3.0 Pro для Windows 95 уже посвящалась статья вашего покорного слуги в 25-м номере журнала. Тогда в статье были рассмотрены программы из состава комплекса – как редактор изображений Picture Man 3.0, так и программа для реализации эффекта морфинга Morph Man, программа захвата видео (на деле захватить видео не получилось) Capture Man. Но хотелось найти и версию программы для Windows 3.x. Конечно, уже во время написания статьи на странице редактора на сайте Old-DOS.ru (<http://old-dos.ru/index.php?page=files&mode=files&do=show&id=6503>) лежала оценочная версия редактора – 1.55 Trial. Чуть позже появились и полные версии 1.51 и 1.55. Но хотелось именно версии 3.0 – именно эта версия указывалась в списке программ «Русского офиса» при установке связки FineReader 2.0 и «Сократа 2.0». Конечно, подтверждений существования такого комплекта на сегодняшний день (сентябрь 2023-го года ☺) нет, но конкретно указанный в списке Picture Man 3.0 действительно существовал, что подтверждало несколько статей, причём подтверждалось существование версии для Windows 3.1 (в принципе, указывалось, что существует не просто Picture Man 3.0 для Windows 3.1, но и Picture Man 3.0 Pro для этой версии винды).

Поиск – иногда неспешный, иногда активный – продолжался почти с осени 2017-го года до мая этого – 2023-го – года, когда пользователям сайта Old-DOS.ru удалось найти и загрузить на сайт Picture Man 3.0 именно для Windows 3.1 – <http://old-dos.ru/dl.php?id=30740>. Именно этой версии будет посвящена настоящая статья.

Опыты с программой проводились на компьютере следующей конфигурации: UMC U5S 33 МГц, 16 МБ ОЗУ, видеокарта UMC

UM85C418 1 МБ в режиме 800x600 32 тыс. цветов с установленной русской версией Windows 3.1. Для экспериментов со сканированием использовался сканер Mustek Paragon 600 II N.

Установка и обработка изображений

Как писал выше, программу можно скачать с сайта Old-DOS.ru (<http://old-dos.ru/dl.php?id=30740>). Установка программы проблем не вызвала. В том числе и под Windows 3.11. После установки программы в «Диспетчере программ» Windows появилась группа редактора с ярлыком для запуска Picture Man.

Главное окно показано на рис. 1.

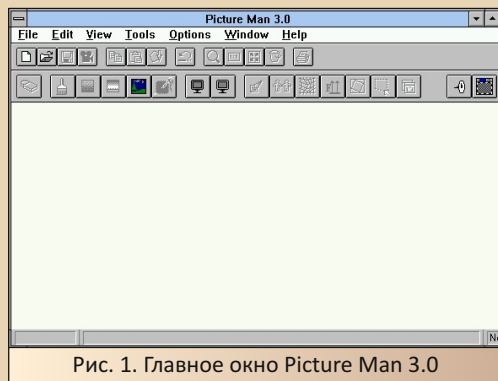


Рис. 1. Главное окно Picture Man 3.0

В отличие от Picture Man 3.0 для Windows 95, описанного в 25-м номере, здесь мы имеем дело с версией программы на английском языке. Меню, по первости, вызывало некоторые сложности, но к англоязычным названиям достаточно быстро привык. Собственно – никаких особенных терминов, отличающихся от других редакторов или просто программ, не заметил.



Как и в Photo Editor'е из состава Microsoft Office (описан в 40-м номере журнала), эксперименты будут проводиться на одной из фотографий, сделанных новым фотоаппаратом. Возможно, условия несколько искусственные – JPEG-файл 4 МБ в середине 90-х, думается, не был очень частым явлением, но многие старые графические редакторы с честью справлялись с такими файлами. Справился и Picture Man (а вот Photo Editor не сумел ☹) – файл открывался долго, но открылся. Вот только незадача – на открытие файла затрачено столько памяти, что для скриншота (использовался SnagIt) памяти уже не хватило. Поэтому окно с открытым файлом было снято на фотоаппарат.



Рис. 2. Фотографию Picture Man открыл, но на это ушла вся память...

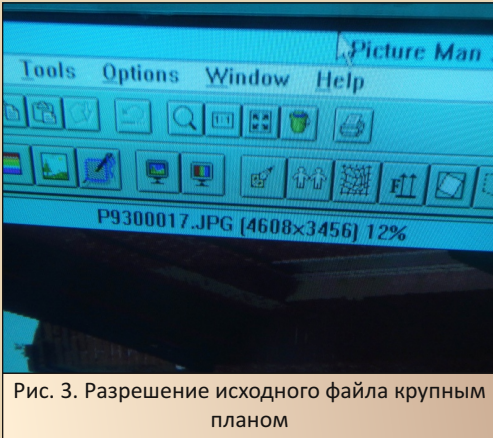


Рис. 3. Разрешение исходного файла крупным планом

Первое, что было сделано – уменьшено разрешение до 800x600. Это сделано через

диалог изменения разрешения, вызываемый командой меню **Tools->Transformations->Size** (см. рис. 4).

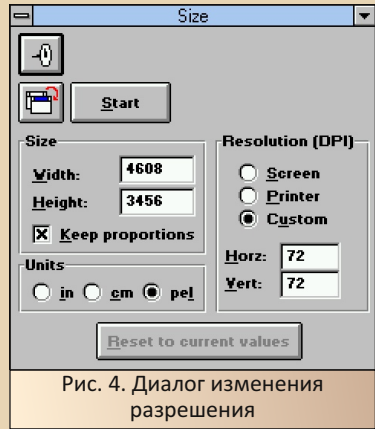


Рис. 4. Диалог изменения разрешения

Как и в версии для Windows 95, описанной в 25-м номере журнала, программа, после выполнения операции, сохраняет и предыдущий вариант изображения, и получившийся в результате выполненной операции (за исключением коррекции цвета – там остаётся только изображение со скорректированной цветовой гаммой). Также изображение было повернуто на 90 градусов диалогом поворота, вызываемым командой **Tools->Transformations->Rotate->Image**.

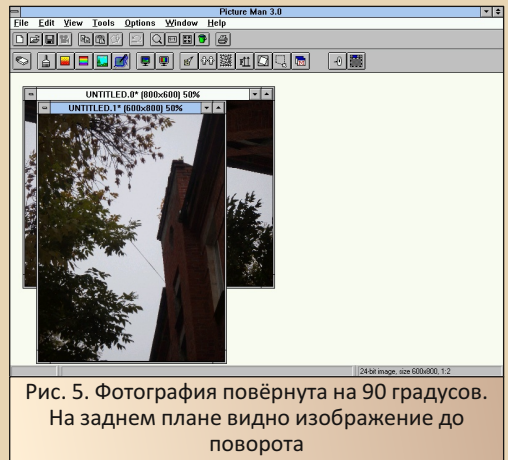


Рис. 5. Фотография повернута на 90 градусов. На заднем плане видно изображение до поворота

Кроме изменения разрешения можно изменить и разрядность цвета. Это можно сделать



командами в меню **Tools->Convert**. В частности, команда **Tools->Convert->Convert to 8-bit paletted** преобразует картинку в 256 цветов (см. рис. 6), а команда **Tools->Convert->Convert to 8-bit grey** – в оттенки серого (см. рис. 7).



Рис. 6. Результат преобразования в 256-цветную гамму




Рис. 7. Результат преобразования в оттенки серого

Если данные операции понятны пользователю с наскока, то коррекция цветовой гаммы достаточно сильно отличается от аналогичных

инструментов других редакторов. Диалог настройки цветовой гаммы, яркости и контраста вызывается командой **Tools->Tune->RGB control** (см. рис. 8).

Вместо движков, соответствующих параметрам изображения (цвета – красный, зелёный и синий, – яркость, контраст и т.д.) пользователю предлагается прямоугольное поле с линией, которую можно перемещать за два узелка по краям. Возможно, достаточно удобный инструмент, если разобраться, но достаточно непривычный.

Так как в операции цветовой коррекции редактор не сохраняет предыдущий вариант изображения, лучше коррекцию выполнять с включённой опцией предпросмотра (включается нажатием кнопки ). С помощью данного диалога ваш покорный слуга немного подкорректировал цветовую гамму фотографии. Итог можно увидеть на рис. 9.

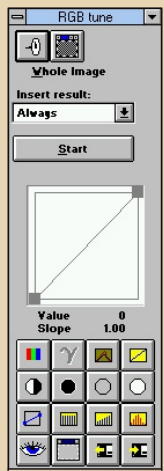


Рис. 8. Диалог настройки параметров цвета изображения

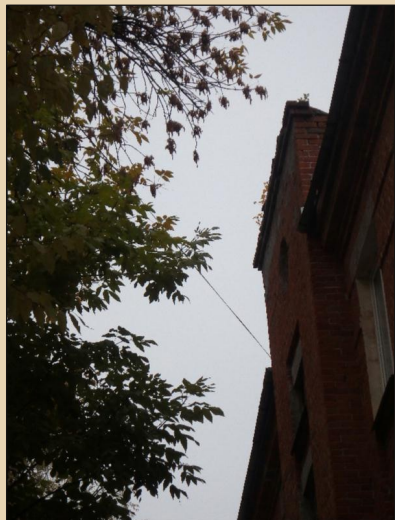


Рис. 9. Результат цветовой коррекции фотографии. Итоговый файл сохранён в формате BMP. Почему-то при сохранении JPEG на изображении появились квадраты



Как видите, редактор способен выполнять основные операции, используемые для обработки растровых изображений – цветовую коррекцию, изменение разрешения, цветовой палитры. Кроме того, доступны инструменты геометрической коррекции и обрезки изображения, о которых рассказывалось в статье, посвящённой Picture Man Pro для Windows 95. Так что Picture Man, на взгляд автора, имеет достаточный для простого редактора функционал.

Сканируем с помощью редактора

Одна из задач, решаемых простым редактором растровой графики – помощь в сканировании бумажных изображений. Так как Picture Man претендует на эту нишу (и возможностями, и объёмом дистрибутива), решено было проверить и функцию сканирования. Для этого был установлен сканер Mustek Paragon 600 II N с простым TWAIN-драйвером <http://old-dos.ru/dl.php?id=22632>. Ставить полный комплект Power of Scanning (<http://old-dos.ru/dl.php?id=22630>) не захотелось, так как редактор для сканирования уже установлен, а OCR (тем более без поддержки русского языка) сейчас не требуется.

После установки драйвера и перезагрузки Windows был запущен редактор и открыт диалог выбора сканера командой **File->Select Source** (см. рис. 10).

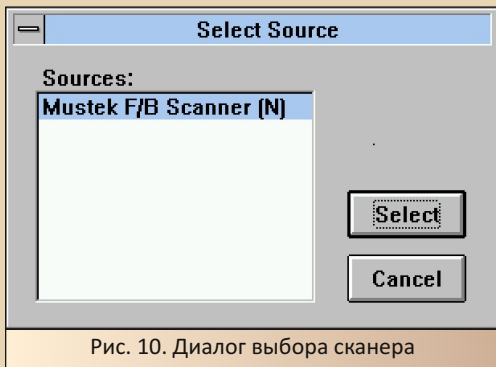


Рис. 10. Диалог выбора сканера

После этого командой **File->Acquire** можно открыть диалог сканирования (см. рис. 11). Для сканирования графики (по крайней мере, на слабом компьютере) лучше установить разрешение 100 dpi. По крайней мере, у вашего почтового слуги на 300 dpi программа отказалась сканировать даже для предпросмотра.

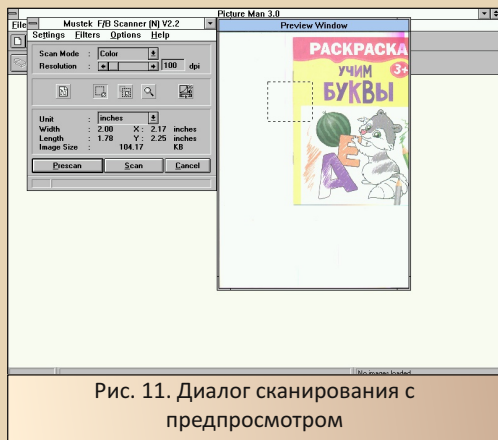


Рис. 11. Диалог сканирования с предпросмотром

Предпросмотр позволяет выбрать фрагмент для сканирования, что и было сделано.

После нажатия кнопки **Scan** сканер «откатал» изображение и управляющая программа передала его редактору.

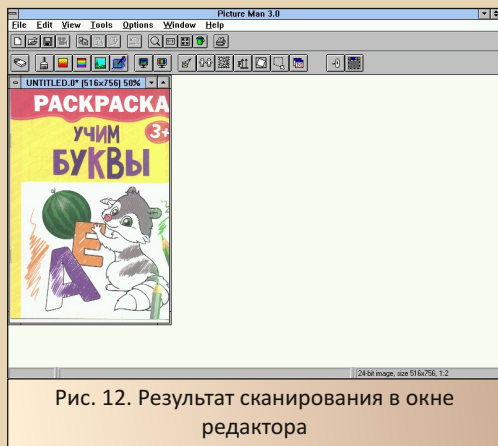


Рис. 12. Результат сканирования в окне редактора





Рис. 13. Результат сканирования

Как видно, Picture Man можно использовать и для сканирования изображений. Редактор оказался достаточно лёгким и работал достаточно быстро, что лишний раз порадовало.

Заключение. Поиск не окончен!

Эту версию редактора автор искал достаточно долго. О третьей версии Picture Man узнал в 2017-м году, примерно тогда же на Old-DOS.ru была загружена версия для Windows 95. О версии для Windows 3.x было известно ещё тогда, но дистрибутив удалось найти только этим летом.

Огромное спасибо участникам форума Old-DOS.ru, постаравшимся загрузить и диск-сборник, и отдельно дистрибутив на сайт. Огромное спасибо!

Но поиск не заканчивается – пусть и не так уже активно, но интересно было бы найти и усечённую версию Picture Man 3.0 SE, и комплект ST Picture Man от Cognitive Technologies, состоявший из Picture Man и Cunei-Form, а то и, может быть, получится найти русскую версию Picture Man для «Русского офиса» с соответствующим инсталлятором. Но это уже хотелки на будущее. Но главное, благодаря нескольким энтузиастам читатели журнала, а также автор этой статьи могут лично попробовать зеленоградский графический редактор под Windows 3.1.

Андрей Шаронов (Andrei88)



HAIKU OS: НА ПЕРВЫЙ ВЗГЛЯД

Сначала немного о подходе, который применяется в этой статье. Это именно «на первый взгляд», т.е. некий поверхностный обзор данной ОС от непрофессионала. То есть что-то вроде: идёт такой себе Вася Пупкин, смотрит – диск с Haiku OS на дороге лежит. Думает: дай-ка гляну. Причём это не сисадмин или программист, а обычный пользователь, который до этого разве что какую-нибудь Windows видел.

Это пояснение нужно, так как к прошлому моему подобному обзору на ArcaOS (№30 журнала) были претензии, мол, всё там можно настроить, скачать и т.п., а я только жаловался, что сразу не работает, вместо того, чтобы почитать документацию и порыться в настройках. :-)

Итак, использовался следующий компьютер: материнская плата ASUS A7V8X-X, процессор Athlon XP 2 ГГц, 512 МБ ОЗУ, видеокарта Nvidia GeForce 6200 с 256 МБ видеопамяти, жёсткий диск на 10 ГБ, привод DVD-RW, сетевая на чипе Realtek RTL8139.

Собственно, практически та же система, что и для ArcaOS в том обзоре, только памяти меньше, т.к. пара планок «накрылась», а новые достать не получилось пока.

Немного про саму операционную систему Haiku. Как пишут разработчики, она была «вдохновлена» BeOS, развитие которой прекратилось в 2001 году. Гибридное ядро Haiku создано бывшим инженером из Be Inc. Тревисом Гейсельбрехтом. Причём на нём можно запускать приложения BeOS. Есть 32- и 64-битный вариант.

Что дальше? Идём на официальный сайт <https://www.haiku-os.org> и качаем образ DVD объёмом 1.4 ГБ (жаль, что нет урезанной версии для CD) по адресу:

http://mirror.rit.edu/haiku/r1beta4/haiku-r1beta4-x86_gcc2h-anyboot.iso

Версия R1/beta4 от 2022-го года

Образ был записан на DVD+RW болванку асте. (Как выяснилось позже – зря.)

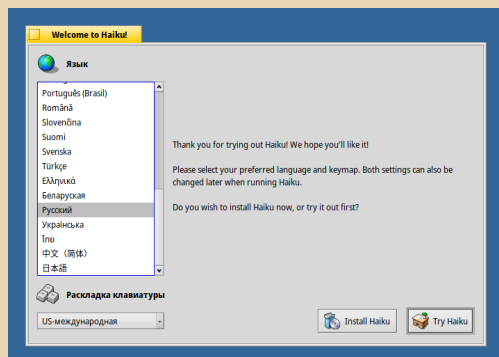
Загружаемся. В процессе нам показывают вот такую симпатичную заставку:



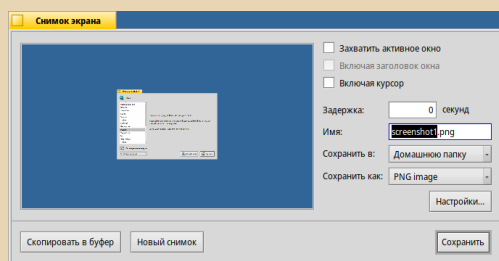
(Скриншот из виртуалки.)

С появления логотипа до вывода окна с выбором языка прошло около минуты. Кому-то покажется долго, но столько времени фактически заняла полная загрузка ОС, т.е. на этом этапе мы уже в Haiku.

Далше выбираем язык (русский присутствует):



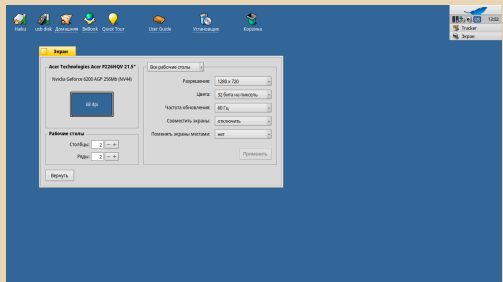
Кстати, уже на этом этапе можно делать скриншоты. Причём по нажатию на **PrtScr** происходит не просто захват экрана в буфер, а появляется полноценный диалог сохранения файла-картинки:



Можно выбрать формат и место сохранения. Причём если вы подключили флешку до загрузки ОС, то она уже будет видна в данном диалоге. У меня жёсткий диск не был размечен, поэтому все скриншоты сохранял как раз на флешку. Очень удобно! И какой контраст с ArcaOS (см. №30), где для того, чтобы сделать обычный скриншот, нужно было качать дополнительную программу...

Продолжим. Теперь у нас два варианта – либо запустить установку, либо версию live, т.е. загружаемую прямо с DVD. Для начала я попробовал второе – рабочий стол появился практически мгновенно, всего несколько секунд.

В общем и целом live-версия работает вполне неплохо:



Теперь попробуем установку. Вот тут появились первые проблемы. Видимо, возникли сбои при работе с DVD, а проявилось это так: в случайном месте при копировании файлов появляется малоинформативное сообщение «General system error» – и система фактически зависает. Хотелось бы всё-таки чего-то более информативного – «Невозможно прочитать такой-то файл. Повторить? Отменить?». Ну или что-то в таком роде, а то мне понадобилось какое-то время, чтобы выяснить, что дело в DVD. Ну да ладно, пожертвовал более качественной болванкой DVD-R Verbatim, подобрал более качественный привод – и дело сдвинулось с мёртвой точки.

После выбора языка у нас попросят указать раздел для установки.

Если диск чистый (без разделов – я как раз с него всё стёр с помощью MHDD), то для его

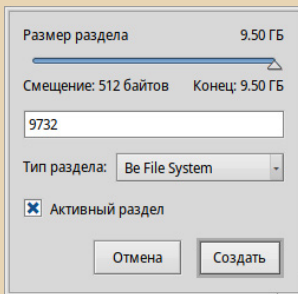
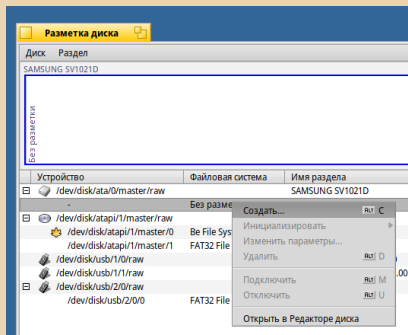
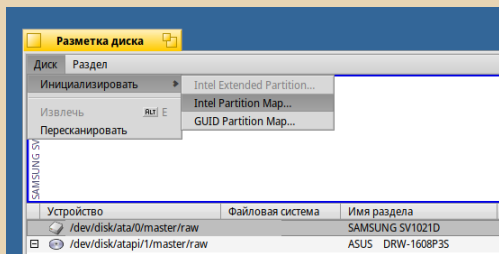
разбивки рекомендую воспользоваться советом отсюда:

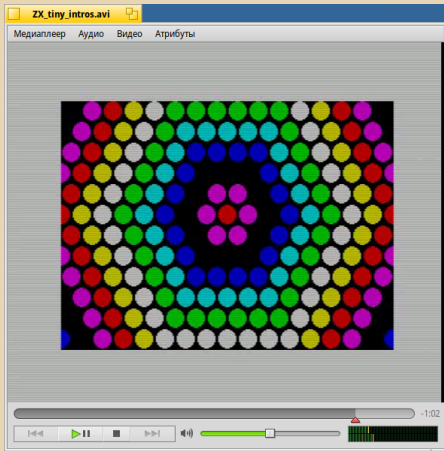
<https://dev.haiku-os.org/ticket/8343#comment:9>

1. Select the drive. Partition->Initialize->Intel Partition Map (agree to all the warnings)
2. Select the empty space. Partition->Create. It should be BFS by default.
3. Select the partition. Partition->Initialize->Be File System.

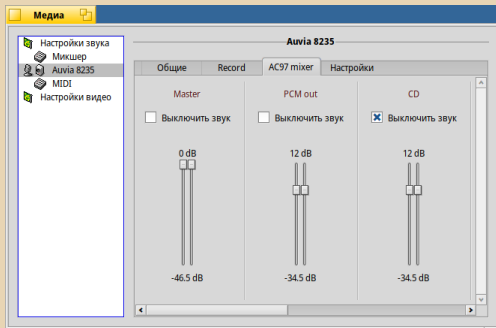
Во время одной из первых попыток установки я отклонился от этого плана, и у меня загрузка зависала на стадии заставки (без сообщений об ошибках, даже в консоли, вызываемой по Esc). Возможно, дело было как раз в этой последовательности. Когда сделал всё по инструкции – загрузка пошла нормально.

Так что делаем так:





Но тут возникла очередная проблема – нет звука. Т.е. воспроизведение MP3 или видео вроде как идёт, ошибок нет, но нет и звука в наушниках... Н-да. Что ж, лезем в настройки микшера, проверяем, двигаем ползунки.



Без толку. Тут я зачем-то решил попробовать воткнуть наушники в другой разъём. Были они воткнуты как положено – в зелёный, попробовал в синий – ничего, затем в розовый (микрофонный) – есть звук. О_о

Честно говоря, мне сложно объяснить, почему звук теперь идёт оттуда... Но главное, что он всё-таки появился...

Если подытожить впечатления о программах, то их много, хоть все и довольно минималистичные, но зато практически на все случаи жизни. Есть также несколько демок и игр.

В целом система довольно отзывчивая, но после запуска нескольких приложений

начинает притормаживать. Всё-таки чувствует-ся, что 512 МБ ОЗУ ей маловато. Но это и неудивительно – этот объём – почти минимум. Кстати, заявленные системные требования следующие: Pentium CPU, 384 MiB RAM, 1.5 GiB HDD, VESA-видеокарта.

Итого

Плюсы. Если сравнивать с ArcaOS, то в Naiku мне больше понравилось следующее:

1. Удобная система скриншотов (и более дружелюбный интерфейс в целом).
2. Воспроизведение MP3 и любого видео «из коробки».
3. Поддержка файловых систем FAT32, exFAT и NTFS.

Из минусов:

1. Всё-таки жаль, что родной браузер не заработал и пришлось качать NetSurf (в ArcaOS браузер заработал сразу).
2. Невозможно запустить DOS или Windows-приложения (опять-таки по сравнению с ArcaOS).

3. Не очень информативные сообщения об ошибках (вроде «General system error»), причём иногда вообще нет никаких сообщений – система просто зависает.

4. Неполная русификация.

Если подвести «итоги итогов», то в целом система произвела положительное впечатление. Особенно в плане дружелюбности к пользователю – тут она явно впереди ArcaOS (во всяком случае, той версии, что рассматривалась в №30 – с тех пор могло что-то и поменяться, ведь выходили обновления).

Но всё-таки иногда видно, что это бета, кое-что ещё сыровато. Надеюсь, в будущем указанные мелкие недоработки всё-таки исправят.

Но уже сейчас систему в live-варианте вполне можно применять, например, как спасательный диск – скажем, вытащить откуда-то файлы, если основная ОС не грузится.

Да и вообще вполне неплохая бесплатная система для достаточно старых (но всё-таки из 2000-х) компьютеров.





Проект
«Зазеркалье»

Расширенный экран Profi: что это такое и как с ним работать.

Практикум 002. Вертикальный скроллинг

В журнале «ЗаRulem» выходил цикл наших статей, посвящённых компьютеру Profi. В рамках этого материала освещались разные аспекты работы с этой машиной. В том числе есть материал по расширенному экрану компьютера Profi, в котором предоставлялась подробная информация о его строении, а также началась публикация практикума по работе с ним. К сожалению, редакция журнала «ЗаRulem» временно прекратила работу над подготовкой новых номеров, сосредоточившись на иных своих проектах. По согласованию со всеми сторонами, публикацию цикла статей было решено перенести в этот журнал. В тех местах, где это необходимо, будут даваться ссылки на номера журнала «ЗаRulem». Итак, начнём наш практикум...

Скроллинг экрана – одна из самых распространённых операций с ним. Давайте рассмотрим вертикальный скроллинг расширенного экрана, а горизонтальный скроллинг разберём в одной из следующих статей.

Скроллинг всего экрана представляет собой обычную переброску данных из одного места в другое, только в экранной памяти. Реализовать этот процесс можно следующими методами:

1. Побайтная переброска через регистр **A**. Самый медленный способ, но позволяющий работать со сложными структурами данных. Мы использовали его при выводе на расширенный экран картинки из файла формата **GRF**.

2. Переброска данных с использованием команды **LDI**. Один из самых быстрых методов переброски. В идеальном случае на переброску одного байта тратится 16 тактов.

3. Переброска через стек, с использованием конструкции **POP/PUSH**. Самый быстрый из известных методов переброски, в идеальном случае на переброску одного байта тратится 10,5 тактов, но при этом получается сложная организация процесса переброски, требуются значительные дополнительные затраты, которые увеличивают необходимое число тактов.

Так как расширенный экран Profi имеет линейную структуру, а значит, нет необходимости работать с каждым байтом в отдельности, реализуем только методы 2 и 3. Начнём с использования команды ассемблера **LDI** как более простого.

Мы уже знаем, что расширенный экран Profi имеет строение практически идентичное с обычным экраном «Спектрума». А значит, он имеет линейную структуру только в пределах одной пиксельной линии, которые организуют знакоместа, а они – сегменты экрана. И для перемещения по экрану нужно проводить определённые расчёты. Как и в прошлой статье, для уменьшения объёмов расчётов воспользуемся той же таблицей адресов начала знакомест. А внутри знакомест переходить к следующей пиксельной линии будем путём увеличения на единицу старшего регистра в регистровой паре.

В основной программе необходимо командами `ld de, 0207h or 08D8h; call 0f82dh` включить в нижние 64 КБ, графика с **#8000**, цвет с **#4000**.

В переменной `scr_step` будем хранить шаг скроллинга в знакоместах. На этапе разбора задачи он будет равен 1 знакоместу, но может иметь любое значение.

Забегая вперёд, скажу, что разница между процедурами скроллинга вверх и вниз



минимальная и заключается в расчёте начальных адресов, откуда и куда кидаем данные, а также в использовании **INC/DEC** при переходе между знакоместами. Поэтому подготовим универсальную процедуру для скроллинга в обоих направлениях, за выбор которого будет отвечать переменная **scr_move**: 0 – вверх, 1 – вниз.

Так как шаг скроллинга может быть любой, начнём с определения величины смещения в таблице адресов между знакоместами, откуда и куда переносим данные.

```
ld a, (scr_step) ;; a=шаг в знакоместах
add a, a        ; a=a*2, так как адрес – это 2 байта
ld e, a        ;; de=a, ширина смещения в таблице
ld d, 0        ;;
```

Далее в зависимости от направления скроллинга немного модернизируем код основного цикла и рассчитываем место в таблице адресов знакомест, откуда начнём брать данные.

```
ld a, (scr_move)
and a          ; if a=0
jr z, Scroll.1di.cr.up ; then движемся вверх
ld a, 2bh     ; else движемся вниз
ld (Scroll.1di.cr.IX+1), a ; для скроллинга вниз
ld (Scroll.1di.cr.IX+3), a ; dec ix; dec ix
ld (Scroll.1di.cr.IY+1), a ;; dec iy; dec iy
ld (Scroll.1di.cr.IY+3), a ;;
ld hl, AdrZNGr_end-2 ; Конец списка адресов
                          ; знакомест куда

push hl
sbc hl, de      ; hl=AdrZN_end-de, смещаемся
                  ; в таблице на откуда

jr Scroll.1di.cr.go1
Scroll.1di.cr.up:
ld a, 23h      ; настройка подпрограммы
ld (Scroll.1di.cr.IX+1), a ; для скроллинга вверх
ld (Scroll.1di.cr.IX+3), a ; inc ix; inc ix
ld (Scroll.1di.cr.IY+1), a ;; inc iy; inc iy
ld (Scroll.1di.cr.IY+3), a ;;
ld hl, AdrZNGr ; Начало списка адресов знакомест
                  ; куда

push hl
add hl, de     ; hl=AdrZN+de, смещаемся в таблице
                  ; на откуда

Scroll.1di.cr.go1:
ld ix, hl     ; ix=адрес в таблице, откуда
                  ; берём данные
```

Здесь **AdrZNGr**, **AdrZNGr_end** – начало и конец таблицы адресов знакомест.

В последней строчке впервые встречается макрос **.ld** (смотри приложение), он копирует содержание одной регистровой пары в другую методом **POP/PUSH**. В данном случае мы копируем содержание регистровой пары **HL** в пару **IX**.

Теперь нам нужно рассчитать общее число перемещаемых знакомест.

```
ld a, (scr_step) ; a=шаг скроллинга в знакоместах
ld b, a          ;;
ld a, 30         ;;a=30-scr_step
sub b            ;;
ld b, a         ; b=Общее число перемещаемых
                  ; знакомест
pop iy          ; iy=адрес в таблице, куда кидаем
                  ; данные
```

После всех действий имеем в **IX** = адрес в таблице, откуда берём данные, **IY** = адрес в таблице, куда кидаем данные, а **B** = общее число перемещаемых знакомест.

Основное тело подпрограммы состоит из двух вложенных циклов: по знакоместам и по пиксельным линиям.

Цикл по знакоместам выглядит так:

```
Scroll.1di.cr.x: ; Цикл по знакоместам
push bc
ld l, (ix+0)    ;; hl= Адрес, откуда данные будут
                  ;; браться

ld h, (ix+1)    ;;
Scroll.1di.cr.IX:
inc ix         ; <!> dec ix
inc ix         ; <!> dec ix
ld e, (iy+0)   ; de=Адрес строки, куда вставляем
                  ; данные

ld d, (iy+1)   ;;
Scroll.1di.cr.IY:
inc iy         ; <!> dec ix
inc iy         ; <!> dec ix
ld b, 8        ; число пиксельных линий в знако-
                  ; месте

Scroll.1di.cr.y: ; Цикл по пиксельным линиям
                  <.....>
                  <цикл рассмотрим ниже>
                  <.....>

pop bc
dec b
jr nz, Scroll.1di.cr.x ; Цикл по знакоместам
```



Отмечу, что во время подготовительных операций мы модернизировали именно его, изменяемые строки помечены <!>. В остальном цикл прост и не должен вызвать дополнительных вопросов.

Цикл по пиксельным линиям выглядит так:

```
ld b, 8 ; число пиксельных линий в знакоместе
Scroll.lidi.cr.y : Цикл по пиксельным линиям
.push <bc, hl, de>
rept 32
    ldi
endm
.push <de, hl>
.push <hl, de>
res 5, h ; Перешли на вторую часть экрана
res 5, d ; Перешли на вторую часть экрана
rept 32
    ldi
endm
.push <de, hl>
.push <hl, de>
ld a, 11000000B ; переходим на цвет
xor h ;
ld h, a ;
ld a, 11000000B ; переходим на цвет
xor d ;
ld d, a ;
.push <hl, de>
rept 32
    ldi
endm
.push <de, hl>
res 5, h ; Перешли на вторую часть экрана
res 5, d ; Перешли на вторую часть экрана
rept 32
    ldi
endm
.push <de, hl, bc>
inc h
inc d
dec b
jp nz, Scroll.lidi.cr.y ; Цикл по пиксельным линиям
```

Где:

rept – endm это макрокоманда ассемблера M80, которая повторяет строки, ограниченные этими командами, указанное число раз.

.pop и **.push** – это макросы, с которыми мы уже знакомы по прошлой статье (журнал «ЗаRulem» №26).

Как видно, цикл состоит из 4 частей, первыми двумя (для каждого полуэкрана) мы

перебрасываем графическую информацию, следующими двумя – цвет.

Опять же считаю, что тут всё просто и понятно, дополнительных пояснений не требуется.

Теперь разберёмся с третьим методом переброски данных – через стек, с использованием конструкции **POP/PUSH**. Первоначальная настройка полностью аналогична уже рассмотренной, и повторяться не будем. Только отмечу изменение имён меток, где часть **LDI** заменяется на **SP**.

Основной код также состоит из двух аналогичных циклов. Незначительное отличие цикла по знакоместам в том, что данные по адресам забираются в обратном порядке.

```
scroll.sp.cr.x : ; Цикл по знакоместам
    push bc
    ld e, (ix+0) ;; de=адрес, откуда данные
    ; ; будут братья
    ld d, (ix+1) ;;
scroll.sp.cr.ix:
    inc ix ; <!> dec ix
    inc ix ; <!> dec ix
    ld l, (iy+0) ;; hl=адрес, куда будут
    ; ; вставляться данные
    ld h, (iy+1) ;;
scroll.sp.cr.iy:
    inc iy ; <!> dec iy
    inc iy ; <!> dec iy
    ld b, 8 ; число пиксельных линий в знакоместе
scroll.sp.cr.y : ; Цикл по пиксельным линиям
    <.....>
    <цикл рассмотрим ниже>
    <.....>
    pop bc ; Счётчик
    dec b
    jp nz, scroll.sp.cr.x ; Цикл по знакоместам
```

Принцип метода основан на том, что команды ассемблера **POP/PUSH** самые быстрые для процесса чтения-записи данных. За раз считываются/записываются сразу два байта и затрачивается 21 такт, то есть на один байт приходится 10.5 такта. Но существует несколько проблем.

Первое – для временного хранения данных можно использовать только регистровые пары. У процессора Z80 имеется 4 основных



регистровых пары и 4 альтернативных. То есть за раз можно перебросить всего 16 байт.

Второе – для переброски используется стек процессора, а он у нас один. Алгоритм работы такой: устанавливаем стек процессора (его адрес содержит регистровая пара **SP**) на то место, с которого нужно прочитать данные, читаем 16 байт, переносим стек процессора на то место, куда нужно сохранить данные +16 байт (так как стек растёт вниз), записываем 16 байт. Повторяем операцию несколько раз. Наилучший результат по времени получаем при создании огромного «поля», в котором переносим значительные объёмы данных. Элемент подобного «поля» выглядит вот так:

```
ld sp, 0000          ; <!> Адрес, откуда
                    ; забираем данные.
.pop <af, bc, de, hl> ;
exha                ;
.pop <af, bc, de, hl> ;
ld sp, 0000          ;<!> Адрес, куда
                    ; вставляем данные.
.push <hl, de, bc, af>;
exha                ;
.push <hl, de, bc, af>;
```

Здесь использован макрос **exha** (смотри приложение). Он переключает все регистры на альтернативные.

Выходит, что для переброски 1 байта нужно затратить 13 байт. А «поле» **POP/PUSH** для переброски всего расширенного экрана Profi на одно знакоместо должно занимать 199 680 байт (195 КБ), создать такое «поле», по понятным причинам, практически нереально.

К тому же нам нужно весьма гибко управлять процессом переброски, а значит, мы не сможем создать «поле» **POP/PUSH** больше чем для переброски одной пиксельной линии.

Тело цикла переброски пиксельных линий состоит из двух частей: настройка «поля» **POP/PUSH** и собственно самого «поля». В настройке рассчитываем адреса, откуда/куда переносим данные, и модернизируем код «поля».

Сам цикл выглядит так:

```
ld b, 8 ; число пиксельных линий в знакоместе
scroll.sp.cr.y: ; Цикл по пиксельным линиям
.push <bc, de, hl>
ld bc, 16 ; число переносимых за раз байт
add hl, bc ; +00 т.к. вставка в обратном
            ; порядке
ld (scroll.sp.cr.01+1), hl ; +00
res 5, h ;Перешли на вторую
            ;часть экрана
ld (scroll.sp.cr.sp+(3+10)*1+1), hl ; +00
add hl, bc ; +16
ld (scroll.sp.cr.sp+(3+10)*3+1), hl ; +16
set 5, h ;Вернулись на первую часть экрана
ld (scroll.sp.cr.sp+(3+10)*5+1), hl ; +16
ld a, 11000000B ; переходим на цвет
xor h ;
ld h, a ;
ld (scroll.sp.cr.sp+(3+10)*7+1), hl ; +16
res 5, h ;Перешли на вторую часть экрана
ld (scroll.sp.cr.sp+(3+10)*9+1), hl ; +16
sbc hl, bc ; -16
ld (scroll.sp.cr.sp+(3+10)*11+1), hl ;+00
set 5, h ;Вернулись на первую часть экрана
ld (scroll.sp.cr.sp+(3+10)*13+1), hl ;+00
ex de, hl
ld (TempSP), sp ; <!!!>
ld sp, hl ; <!> +00
res 5, h ;Перешли на вторую часть экрана
ld (scroll.sp.cr.sp+1), hl ; +00
add hl, bc ; +16
ld (scroll.sp.cr.sp+(3+10)*2+1), hl ; +16
set 5, h ;Вернулись на первую часть экрана
ld (scroll.sp.cr.sp+(3+10)*4+1), hl ; +16
ld a, 11000000B ; переходим на цвет
xor h ;
ld h, a ;
ld (scroll.sp.cr.sp+(3+10)*6+1), hl ; +16
res 5, h ;Перешли на вторую часть экрана
ld (scroll.sp.cr.sp+(3+10)*8+1), hl ; +16
sbc hl, bc ; -16
ld (scroll.sp.cr.sp+(3+10)*10+1), hl ;+00
set 5, h ;Вернулись на первую часть экрана
ld (scroll.sp.cr.sp+(3+10)*12+1), hl ;+00
.pop <af, bc, de, hl> ;
exha ;
.pop <af, bc, de, hl> ;
scroll.sp.cr.01: ld sp, 0000 ; <!>
.push <hl, de, bc, af> ;
exha ;
.push <hl, de, bc, af> ;
scroll.sp.cr.sp:
rept 7
ld sp, 0000 ; <!>
```



```
.pop <af, bc, de, hl> ;
exha ;
.pop <af, bc, de, hl> ;
ld sp, 0000 ; <!>
.push <hl, de, bc, af> ;
exha ;
.push <hl, de, bc, af> ;
endm
ld sp, (TempSP) ; <!!!!>
.pop <hl, de, bc>
inc h
inc d
dec b
jp nz, scroll.sp.cr.y ;Цикл по пиксельным линиям
```

Как видно, тело цикла несколько больше, чем при использовании команды ассемблера **LDI**. Несмотря на это, его логика, думаю, понятна.

В заключение давайте проведём сравнительный анализ обоих вариантов. Для чего подготовим таблицу 1. Несмотря на то, что в статье приведён материал для скроллинга цветного изображения, в таблице приведу также данные для скроллинга только графики (чёрно-белого изображения). Считаю, что упростить приведённые примеры не составит труда.

Таблица 1

Алгоритм	Для графики			Для графики и цвета		
	Размер, байт	Время, тактов	Время на байт, тактов	Размер, байт	Время, тактов	Время на байт, тактов
LDI	243	277067	18,66	394	540155	18,19
POP/PUSH	252	268019	18,05	403	505123	17,01
Разница	9	9048	0,61	9	35032	1,18

Как видно из таблицы, метод с использованием команды ассемблера **LDI** незначительно короче, но проигрывает по скорости методу перебрёски данных через стек процессора. Причём, при скроллинге с цветом, весьма существенно.

Отмечу, что при использовании перебрёски данных через стек процессора требуется отключение прерываний, иначе после прихода сигнала прерывания в стек запишется адрес возврата, что испортит данные. Это несколько сужает сферу применения данного метода.

Для стандартной частоты процессора Z80 в 3.5 МГц цифры кажутся огромными. Но не стоит забывать две вещи. Первое – расширенный экран Profi занимает 32 КБ, против 6.7 КБ стандартного экрана «Спектрума». И второе – все модели компьютера Profi оснащены турборежимом, значительно повышающим тактовую частоту процессора.

В зависимости от ряда факторов и версии схем, существуют машины со следующими частотами работы процессора в турборежиме: 7 МГц, 10 МГц, 12 МГц, 15 МГц, 21 МГц. В настоящее время наиболее часто встречаются частоты 10 МГц и 12 МГц. Их и следует брать как базовые.

Таблица 2. Число тактов между прерываниями при разных скоростях работы процессора

Скорость процессора, МГц	Число тактов между прерываниями
3,5	69888
7	99880
10	149760
12	179712
15	224640

Если немного покривить против истины и распространить полученные данные по скорости работы процедур для расширенного экрана Profi на стандартный экран «Спектрума», то для сравнительного анализа можно составить следующие таблицы.

Таблица 3. Число необходимых прерываний для скроллинга экрана на 1 знакоместо при разных частотах работы процессора

Стандартный экран «Спектрума» 256x192

Алгоритм	ч/б, тактов	Цвет, тактов	3.5 МГц		7 МГц		10 МГц		12 МГц		15 МГц	
			ч/б	цв.	ч/б	цв.	ч/б	цв.	ч/б	цв.	ч/б	цв.
LDI	109871	120487	1,57	1,72	1,10	1,21	0,73	0,80	0,61	0,67	0,49	0,54
POP/PUSH	106283	112673	1,52	1,61	1,06	1,13	0,71	0,75	0,59	0,63	0,47	0,50

Таблица 4. Расширенный экран 512x240

Алгоритм	ч/б, тактов	Цвет, тактов	3.5 МГц		7 МГц		10 МГц		12 МГц		15 МГц	
			ч/б	цв.	ч/б	цв.	ч/б	цв.	ч/б	цв.	ч/б	цв.
LDI	277067	540155	3,96	7,73	2,77	5,41	1,85	3,61	1,54	3,01	1,23	2,40
POP/PUSH	268019	505123	3,83	7,23	2,68	5,06	1,79	3,37	1,49	2,81	1,19	2,25



Как видно на скорости процессора в 12 МГц, соотношение объёма переносимых данных к числу тактов в прерывании у расширенного экрана Profi практически идентично этому показателю для стандартного экрана «Спектрума» при скорости процессора 3.5 МГц. Скроллинг расширенного экрана с цветом всегда будет медленнее, так как объём данных о цвете в нём в 8 раз больше, чем у стандартного экрана.

Всё это говорит о том, что на компьютере Profi под его расширенный экран можно создавать столь же динамичные программы и игры, как и для стандартного экрана «Спектрума». Конечно же, с учётом того, что сам компьютер Profi отличается от стандартного «Спектрума» не только расширенным экраном.

На этом пока стоит остановиться. Мы все-сторонне рассмотрели поставленную задачу.

Если же у кого остались вопросы, он всегда может обратиться по одному из следующих адресов.

- FidoNet: Tarasow Aleksey 2:5053/57
- E-mail: tae1980(очень злая собака)yandex.ru

А на форуме, в группе ВК и сайте есть много интересного, а всё новое будет выкладываться в том числе и там.

- Форум: <https://zx-pk.ru/forums/102-profi.html>
- Группа поддержки компьютера Profi ВКонтакте: <https://vk.com/profi1024>
- Группа поддержки компьютера Profi в Telegram: <https://t.me/Profi1024>

По [этой ссылке](#)¹ можно скачать архив LZH, в котором находятся:

- **m80.com, l80.com** – основные файлы пакета M80.
- **SCROLL.ASM** – исходники кода, описанного в этой статье.
- **BIOS.INC** – файл для подзагрузки используемых макросов.
- **SCRDLI.ASM, SCRSTACK.ASM** – программы для тестирования рассмотренных в статье алгоритмов скроллинга. На основе рассмотренной в предыдущей статье программы загрузки картинки на экран из файлов **GRF**.

- **SCRDLI.BAT, SCRSTACK.BAT** – командные файлы для ассемблирования примеров.
- **SCRDLI.COM, SCRSTACK.COM** – уже сассемблированный код, готовый для запуска. Запустить из командной строки с указанием в качестве первого аргумента имени файла **GRF** для вывода на экран.
- Группа чёрно-белых и цветных **GRF**-файлов для теста.

Также [можно скачать](#)² загрузочный образ диска в формате *.pro. Для запуска программ **SCRDLI.COM, SCRSTACK.COM** нужно: в командной строке (вход по клавише «7») набрать их название (без **.COM**) и ввести полное имя картинки. Для удобства можно использовать сочетание клавиш **Shift+J** – вставка в командную строку имени файла под курсором. По умолчанию в ИС при выборе файлов **GRF** будет запускаться программа **SCRSTACK**.

Приложение 1. Используемые макросы

```
.Pop Macro Items
; (c) PSWsoft

    Irp x, <Items>
        POP x
    Endm
EndM
-----

.Push Macro Items
; (c) PSWsoft

    Irp x, <Items>
        PUSH x
    Endm
EndM
-----

.Ld Macro Dest, Sour
; (c) PSWsoft

    push Sour
    pop Dest
EndM
-----

exxa Macro
; (c) PSWsoft

    exx
    ex af, af
EndM
-----
```

1) Альтернативная ссылка: <http://dgmag.in/N44/scroll2/SCROLL.LZH>
 2) Альтернативная ссылка: <http://dgmag.in/N44/scroll2/002Scroll.pro>



Приложение 2. Исходный код из файла SCROLL.ASM

Scroll.stack:

```

;Скроллинг всего экрана (независимо от наличия цвета) вверх или вниз на заданное число
знакомест.
;Алгоритм pop/push.
;Портит: все регистры, включая альтернативные.
;Использует: FColor, scr_step, scr_move, AdrZNgr, AdrZNgr_end
ld a, (scr_step)      ;; a=шаг в знакоместах
add a, a              ; a=a*2, так как адрес - это 2 байта
ld e, a               ;; de=a, ширина смещения в таблице
ld d, 0               ;;
;
ld a, (FColor)        ; a=наличие цвета (0 - ч/б, 1-стандартные цвета, 2-палитра)
and a                 ;; if a<>0
jp nz, Scroll.stack.cr ;; then выводим цветную картинку
;
ld a, (scr_move)
and a                 ;; if a=0
jr z, scroll.sp.bw.up ;; then движемся вверх
ld a, 2bh             ; else движемся вниз
ld (scroll.sp.bw.IX+1), a ; для скроллинга вниз
ld (scroll.sp.bw.IX+3), a ; dec ix; dec ix
ld (scroll.sp.bw.IY+1), a ;; dec iy; dec iy
ld (scroll.sp.bw.IY+3), a ;;
ld hl, AdrZNgr_end-2 ;
push hl               ; для передачи в iy
sbc hl, de            ; hl=AdrZNgr_end-de - смещаемся в таблице на откуда
jr scroll.sp.bw.go1

scroll.sp.bw.up:
ld a, 23h             ; настройка подпрограммы
ld (scroll.sp.bw.IX+1), a ; для скроллинга вверх
ld (scroll.sp.bw.IX+3), a ; inc ix; inc ix
ld (scroll.sp.bw.IY+1), a ;; inc iy; inc iy
ld (scroll.sp.bw.IY+3), a ;;
ld hl, AdrZNgr        ; Начало списка адресов пикс. линий куда
push hl               ; для передачи в iy
add hl, de            ; hl=AdrZNgr+de - смещаемся в таблице на откуда

scroll.sp.bw.go1:
ld ix, hl             ; ix=адрес в таблице, откуда берём данные
ld a, (scr_step)      ; a=шаг скроллинга в знакоместах
ld b, a               ;;
ld a, 30               ;;a=30-scr_step
sub b                 ;;
ld b, a               ; b=Общее число перемещаемых знакомест
pop iy                ; iy=адрес в таблице, куда кидаем данные

scroll.sp.bw.x:
; Цикл по знакоместам
push bc
ld e, (ix+0)          ;; откуда данные будут браться
ld d, (ix+1)          ;;
scroll.sp.bw.ix:
inc ix                ; <!> dec ix
inc ix                ; <!> dec ix
ld l, (iy+0)          ;; куда будут вставляться данные
ld h, (iy+1)          ;;
scroll.sp.bw.iy:
inc iy                ; <!> dec iy
inc iy                ; <!> dec iy

```



```

ld b, 8          ; число пиксельных линий в знакоместе
scroll.sp.bw.y:  ; Цикл по пиксельным линиям
    .push <bc, de, hl>
    ld bc, 16     ; число переносимых за раз байт
    add hl, bc    ; +00 т.к. вставка в обратном порядке
    ld (scroll.sp.bw.01+1), hl ; +00
    res 5, h      ;Перешли на вторую часть экрана
    ld (scroll.sp.bw.sp+(3+10)*1+1), hl ; +00
    add hl, bc    ; +16
    ld (scroll.sp.bw.sp+(3+10)*3+1), hl ; +16
    set 5, h      ;Вернулись на первую часть экрана
    ld (scroll.sp.bw.sp+(3+10)*5+1), hl ; +16
    ex de, hl
    ld (TempSP), sp ; <!!!>
    ld sp, hl     ; +00 <!>
    res 5, h      ;Перешли на вторую часть экрана
    ld (scroll.sp.bw.sp+1), hl ; +00
    add hl, bc    ; +16
    ld (scroll.sp.bw.sp+(3+10)*2+1), hl ; +16
    set 5, h      ;Вернулись на первую часть экрана
    ld (scroll.sp.bw.sp+(3+10)*4+1), hl ; +16
    .pop <af, bc, de, hl> ; забрали полуэкран 1 +00
    exha         ;
    .pop <af, bc, de, hl> ;
scroll.sp.bw.01: ld sp, 0000 ; <!> вставили полуэкран 1 +00
    .push <hl, de, bc, af> ;
    exha         ;
    .push <hl, de, bc, af> ;
scroll.sp.bw.sp:
    rept 3
        ld sp, 0000 ; <!>
        .pop <af, bc, de, hl> ;
        exha         ;
        .pop <af, bc, de, hl> ;
        ld sp, 0000 ; <!>
        .push <hl, de, bc, af> ;
        exha         ;
        .push <hl, de, bc, af> ;
    endm
    ld sp, (TempSP)
    .pop <hl, de, bc>
    inc h
    inc d
    dec b
    jp nz, scroll.sp.bw.y ; Цикл по пиксельным линиям
    pop bc ; Счётчик
    dec b
    jp nz, scroll.sp.bw.x ; Цикл по знакоместам
    ret
;-----

```

Scroll.stack.cr:

;Скроллинг всего экрана с цветом вверх или вниз на заданное число знакомест.

;Алгоритм pop/push.

;На выходе: de=ширина смещения в таблице

;Портит: все регистры, включая альтернативные.

;Использует: scr_move, AdrZNgr, AdrZNgr_end

```
ld a, (scr_move)
```



```

and a                ;; if a=0
jr z, scroll.sp.cr.up ;; then движемся вверх
ld a, 2bh            ; else движемся вниз
ld (scroll.sp.cr.IX+1), a ; для скроллинга вниз
ld (scroll.sp.cr.IX+3), a ; dec ix; dec ix
ld (scroll.sp.cr.IY+1), a ;; dec iy; dec iy
ld (scroll.sp.cr.IY+3), a ;;
ld hl, AdrZNGr_end-2 ;
push hl              ; для передачи в iy
sbc hl, de           ; hl=AdrZNGr_end-de - смещаемся в таблице на откуда
jr scroll.sp.cr.go1

scroll.sp.cr.up:
ld a, 23h           ; настройка подпрограммы
ld (scroll.sp.cr.IX+1), a ; для скроллинга вверх
ld (scroll.sp.cr.IX+3), a ; inc ix; inc ix
ld (scroll.sp.cr.IY+1), a ;; inc iy; inc iy
ld (scroll.sp.cr.IY+3), a ;;
ld hl, AdrZNGr      ; Начало списка адресов пикс. линий куда
push hl              ; для передачи в iy
add hl, de           ; hl=AdrZNGr+de - смещаемся в таблице на откуда

scroll.sp.cr.go1:
ld ix, hl            ; ix=адрес в таблице, откуда берём данные
ld a, (scr_step)     ; a=шаг скроллинга в знакоместах
ld b, a              ;;
ld a, 30              ;;a=30-scr_step
sub b                 ;;
ld b, a              ; b=Общее число перемещаемых знакомест
pop iy               ; iy=адрес в таблице, куда кидаем данные

scroll.sp.cr.x:
push bc
ld e, (ix+0)         ;; откуда данные будут браться
ld d, (ix+1)         ;;

scroll.sp.cr.ix:
inc ix                ; <!> dec ix
inc ix                ; <!> dec ix
ld l, (iy+0)         ;; куда будут вставляться данные
ld h, (iy+1)         ;;

scroll.sp.cr.iy:
inc iy                ; <!> dec iy
inc iy                ; <!> dec iy
ld b, 8               ; число пиксельных линий в знакоместе

scroll.sp.cr.y:
push <bc, de, hl>
ld bc, 16            ; число переносимых за раз байт
add hl, bc           ; +00 т.к. вставка в обратном порядке
ld (scroll.sp.cr.01+1), hl ; +00
res 5, h              ;Перешли на вторую часть экрана
ld (scroll.sp.cr.sp+(3+10)*1+1), hl ; +00
add hl, bc           ; +16
ld (scroll.sp.cr.sp+(3+10)*3+1), hl ; +16
set 5, h              ;Вернулись на первую часть экрана
ld (scroll.sp.cr.sp+(3+10)*5+1), hl ; +16
ld a, 11000000b      ; переходим на цвет
xor h                 ;
ld h, a               ;
ld (scroll.sp.cr.sp+(3+10)*7+1), hl ; +16
res 5, h              ;Перешли на вторую часть экрана
ld (scroll.sp.cr.sp+(3+10)*9+1), hl ; +16

```



```

sbc hl, bc      ; -16
ld (scroll.sp.cr.sp+(3+10)*11+1), hl ;+00
set 5, h       ;Вернулись на первую часть экрана
ld (scroll.sp.cr.sp+(3+10)*13+1), hl ;+00
ex de, hl
ld (TempSP), sp ; <!!!>
ld sp, hl      ; <!> +00
res 5, h       ;Перешли на вторую часть экрана
ld (scroll.sp.cr.sp+1), hl ; +00
add hl, bc     ; +16
ld (scroll.sp.cr.sp+(3+10)*2+1), hl ; +16
set 5, h       ;Вернулись на первую часть экрана
ld (scroll.sp.cr.sp+(3+10)*4+1), hl ; +16
ld a, 11000000B ; переходим на цвет
xor h         ;
ld h, a       ;
ld (scroll.sp.cr.sp+(3+10)*6+1), hl ; +16
res 5, h       ;Перешли на вторую часть экрана
ld (scroll.sp.cr.sp+(3+10)*8+1), hl ; +16
sbc hl, bc     ; -16
ld (scroll.sp.cr.sp+(3+10)*10+1), hl ;+00
set 5, h       ;Вернулись на первую часть экрана
ld (scroll.sp.cr.sp+(3+10)*12+1), hl ;+00
.pop <af, bc, de, hl> ;
exha         ;
scroll.sp.cr.01: ld sp, 0000 ; <!>
                .push <hl, de, bc, af> ;
                exha ;
                .push <hl, de, bc, af> ;
scroll.sp.cr.sp:
rept 7
    ld sp, 0000 ; <!>
    .pop <af, bc, de, hl> ;
    exha ;
    .pop <af, bc, de, hl> ;
    ld sp, 0000 ; <!>
    .push <hl, de, bc, af> ;
    exha ;
    .push <hl, de, bc, af> ;
endm
ld sp, (TempSP) ; <!!!>
.pop <hl, de, bc>
inc h
inc d
dec b
jr nz, scroll.sp.cr.y ; Цикл по пиксельным линиям
pop bc ; Счётчик
dec b
jr nz, scroll.sp.cr.x ; Цикл по знакаместам
ret
;-----

```

Scroll.ldi:

```

;Скроллинг всего экрана (независимо от наличия цвета) вверх или вниз на заданное число
;знакомест.
;Портит: a, hl, de, bc, ix, iy
;Использует: FColor, scr_step, scr_move, AdrZN, AdrZN_end

```



```

ld a, (scr_step)      ;; a=шаг в знакоместах
add a, a              ; a=a*2, так как адрес - это 2 байта
ld e, a              ;; de=a, ширина смещения в таблице
ld d, 0              ;;
;
ld a, (FColor)       ; a=наличие цвета (0 - ч/б, 1-стандартные цвета, 2-палитра)
and a                ;; if a<>0
jp nz, Scroll.1di.cr ;; then выводим цветную картинку
;
ld a, (scr_move)
and a                ; if a=0
jr z, Scroll.1di.bw.up ; then движемся вверх
ld a, 2bh            ; else движемся вниз
ld (Scroll.1di.bw.IX+1), a ; для скроллинга вниз
ld (Scroll.1di.bw.IX+3), a ; dec ix; dec ix
ld (Scroll.1di.bw.IY+1), a ;; dec iy; dec iy
ld (Scroll.1di.bw.IY+3), a ;;
ld hl, AdrZNGr_end-2 ; Конец списка адресов знакомест куда
push hl
sbc hl, de           ; hl=AdrZN_end-de - смещаемся в таблице на откуда
jr Scroll.1di.bw.go1

Scroll.1di.bw.up:
ld a, 23h           ; настройка подпрограммы
ld (Scroll.1di.bw.IX+1), a ; для скроллинга вверх
ld (Scroll.1di.bw.IX+3), a ; inc ix; inc ix
ld (Scroll.1di.bw.IY+1), a ;; inc iy; inc iy
ld (Scroll.1di.bw.IY+3), a ;;
ld hl, AdrZNGr      ; Начало списка адресов знакомест куда
push hl
add hl, de          ; hl=AdrZN+de - смещаемся в таблице на откуда
Scroll.1di.bw.go1:
ld ix, hl           ; ix=адрес в таблице, откуда берём данные
ld a, (scr_step)    ; a=шаг скроллинга в знакоместах
ld b, a             ;;
ld a, 30            ;;a=30-scr_step
sub b               ;;
ld b, a             ; b=Общее число перемещаемых знакомест
pop iy              ; iy=адрес в таблице, куда кидаем данные
Scroll.1di.bw.x:    ; Цикл по знакоместам
push bc
ld l, (ix+0)        ;; откуда данные будут браться
ld h, (ix+1)        ;;
Scroll.1di.bw.IX:
inc ix              ; <!> dec ix
inc ix              ; <!> dec ix
ld e, (iy+0)        ;; hl=Адрес строки, откуда берём данные
ld d, (iy+1)        ;;
Scroll.1di.bw.IY:
inc iy              ; <!> dec ix
inc iy              ; <!> dec ix
ld b, 8             ; число пиксельных линий в знакоместе
Scroll.1di.bw.y:    ; Цикл по пиксельным линиям
push <bc, hl, de>
rept 32
ldi
endm
pop <de, hl>
push <hl, de>

```



```

    res 5, h      ; Перешли на вторую часть экрана
    res 5, d      ; Перешли на вторую часть экрана
    rept 32
        ldi
    endm
    .pop <de, hl, bc>
    inc h
    inc d
    dec b
    jp nz, Scroll.lidi.bw.y ; Цикл по пиксельным линиям
pop bc
dec b
jp nz, Scroll.lidi.bw.x   ; Цикл по знакаместам
ret
;-----

```

Scroll.lidi.cr:

```

;Скроллинг всего экрана с цветом вверх или вниз на заданное число знакомест.
;На выходе: de=ширина смещения в таблице
;Портит: a, hl, de, bc, ix, iy
;Использует: scr_move, AdrZN, AdrZN_end
    ld a, (scr_move)
    and a                ; if a=0
    jr z, Scroll.lidi.cr.up ; then движемся вверх
    ld a, 2bh            ; else движемся вниз
    ld (Scroll.lidi.cr.IX+1), a ; для скроллинга вниз
    ld (Scroll.lidi.cr.IX+3), a ; dec ix; dec ix
    ld (Scroll.lidi.cr.IY+1), a ;; dec iy; dec iy
    ld (Scroll.lidi.cr.IY+3), a ;;
    ld hl, AdrZNGr_end-2 ; Конец списка адресов знакомест куда
    push hl
    sbc hl, de          ; hl=AdrZN_end-de - смещаемся в таблице на откуда
    jr Scroll.lidi.cr.go1

```

Scroll.lidi.cr.up:

```

    ld a, 23h          ; настройка подпрограммы
    ld (Scroll.lidi.cr.IX+1), a ; для скроллинга вверх
    ld (Scroll.lidi.cr.IX+3), a ; inc ix; inc ix
    ld (Scroll.lidi.cr.IY+1), a ;; inc iy; inc iy
    ld (Scroll.lidi.cr.IY+3), a ;;
    ld hl, AdrZNGr     ; Начало списка адресов знакомест куда
    push hl
    add hl, de         ; hl=AdrZN+de - смещаемся в таблице на откуда

```

Scroll.lidi.cr.go1:

```

    .ld ix, hl        ; ix=адрес в таблице, откуда берём данные
    ld a, (scr_step) ; a=шаг скроллинга в знакоместах
    ld b, a           ;;
    ld a, 30         ;;a=30-scr_step
    sub b            ;;
    ld b, a          ; b=Общее число перемещаемых знакомест
    pop iy           ; iy=адрес в таблице, куда кидаем данные

```

Scroll.lidi.cr.x:

```

; Цикл по знакаместам
    push bc
    ld l, (ix+0)    ;; откуда данные будут браться
    ld h, (ix+1)    ;;

```

Scroll.lidi.cr.IX:

```

    inc ix          ; <!> dec ix
    inc ix          ; <!> dec ix
    ld e, (iy+0)   ;; hl=Адрес строки, откуда берём данные

```



```

        ld d, (iy+1)      ;;
Scroll.ldi.cr.IY:
        inc iy           ; <!> dec ix
        inc iy           ; <!> dec ix
        ld b, 8          ; число пиксельных линий в знакоместе
Scroll.ldi.cr.y:        ; Цикл по пиксельным линиям
        .push <bc, hl, de>
        rept 32
            ldi
        endm
        .pop <de, hl>
        .push <hl, de>
        res 5, h         ; Перешли на вторую часть экрана
        res 5, d         ; Перешли на вторую часть экрана
        rept 32
            ldi
        endm
        .pop <de, hl>
        .push <hl, de>
        ld a, 11000000B ; переходим на цвет
        xor h           ;
        ld h, a         ;
        ld a, 11000000B ; переходим на цвет
        xor d           ;
        ld d, a         ;
        .push <hl, de>
        rept 32
            ldi
        endm
        .pop <de, hl>
        res 5, h         ; Перешли на вторую часть экрана
        res 5, d         ; Перешли на вторую часть экрана
        rept 32
            ldi
        endm
        .pop <de, hl, bc>
        inc h
        inc d
        dec b
        jp nz, Scroll.ldi.cr.y ; Цикл по пиксельным линиям
    pop bc
    dec b
    jp nz, Scroll.ldi.cr.x   ; Цикл по знакоместам
    ret
;-----

```

```

scr_step: db 01          ; Шаг скроллинга в знакоместах.
scr_move: db 00          ; Направление скроллинга. 0 - вверх, 1 - вниз.
TempSP:   dw 0000       ; Место для запоминания адреса программного стека.
;-----

```





Начало нулевых, модемный dial-up интернет, скрип модема, глубокая тревога – вдруг кто схватит трубку подключённого параллельно телефонного аппарата (не везло мне – всегда телефон подключался параллельно модему, а не как положено ☺). Минуты тикают – и это тоже напрягает. Конечно, когда качается файл – ничего сделать невозможно – разве что остановить загрузку и запустить её снова в более дешёвое время, но когда речь шла о страничках – другой разговор. Общение с всемирной паутиной, по понятным причинам, хотелось проводить максимально офлайн – спокойно прочитать статью (многобукаф-ниасилил – признак лени и неграмотности, и вина тут читателя, а не автора текста ☺), разобраться в результатах, выданных поисковиком и т.д.

Обычно работа с поисковиком – по крайней мере для меня – сводилось к следующему – открывалась страничка поисковика – из кэша, забивался запрос, подключался интернет, нажималась кнопка поиска, а после судорожно открывались в отдельных вкладках браузера все страницы с результатами поиска. Вот он – звёздный час культовой «Оперы» – в отличие от массивных Internet Explorer (особенно с оболочкой MYIE2 или NetCaptor) или

Mozilla/Netscape, Opera 5.12 или 7.x позволяла открыть куда больше вкладок и при этом не тормозить (это примерно в 2006-м году). После открытия вкладок интернет отключался и начиналось чтение. Интересующие результаты открывались в отдельных вкладках. После этого интернет подключался снова и нажималась кнопка обновления всех вкладок – странички загружались, интернет отключался, и снова начиналось чтение. Сложно, заморочно, но времени тратилось меньше, чем если бы я всё это делал, будучи постоянно подключённым к интернету. Но к интернету приходилось всё равно подключаться, и деньги тратились. Этого тоже хотелось максимально избежать. Тут приходилось пускаться на определённые уловки – загружались страницы и сайты, копался в кэше браузера. Причём для этих вещей использовались различные программы. О некоторых из них и будет эта статья. Не всеми программами я пользовался – некоторые удалось найти и скачать только сейчас. Но всё равно, надеюсь, будет о них интересно узнать не только мне, но и читателю.

Копаемся в кэше браузера

Браузер – программа запасливая ☺ – для ускорения работы всякую открытую пользователем страницу пытается сохранить в собственном буфере – кэше. При повторном обращении к странице пользователю может быть открыта сохранённая в кэше копия – загрузится она куда быстрее, чем из сети – и время сэкономится, и трафик. Если не нравится – нажимаем кнопку обновления, и страничка уже должна загрузиться с интернета. Но это не единственная польза от кэша. Кэшированные страницы можно попытаться просмотреть и не подключаясь к интернету. Достаточно было перевести браузер в автономный режим и попробовать открыть страничку, не подключаясь к интернету. Обычно это делалось командой меню **Файл->Работать автономно** или аналогичной командой на английском. ☺



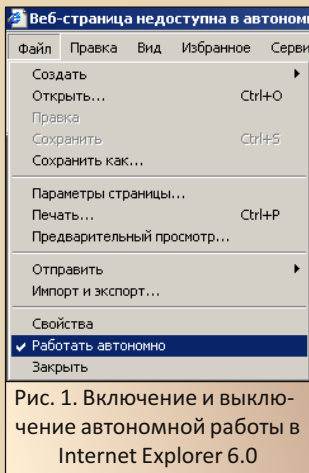


Рис. 1. Включение и выключение автономной работы в Internet Explorer 6.0

Интересно, что такая функция сохранилась и в современных браузерах. Сам не ожидал увидеть её в нынешнем своём браузере – MyPal, а она там есть.

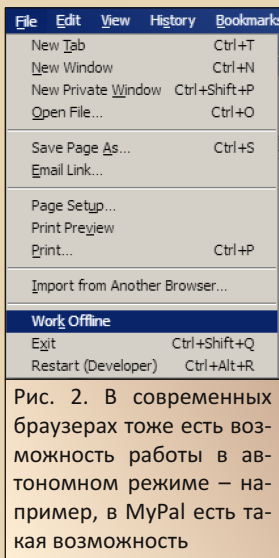


Рис. 2. В современных браузерах тоже есть возможность работы в автономном режиме – например, в MyPal есть такая возможность

После этого открываем историю и ищем посещённый ранее сайт. Интернет был лимитирован, так что сайтов особо много в истории не было. Щёлкаем по строчке, и страничка, возможно, откроется либо появится окно-приглашение перейти в онлайн-режим и подключиться к интернету.

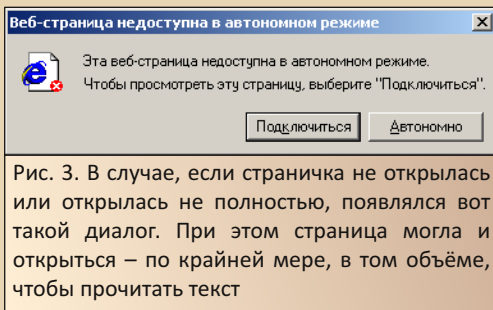


Рис. 3. В случае, если страничка не открылась или открылась не полностью, появлялся вот такой диалог. При этом страница могла и открыться – по крайней мере, в том объёме, чтобы прочитать текст

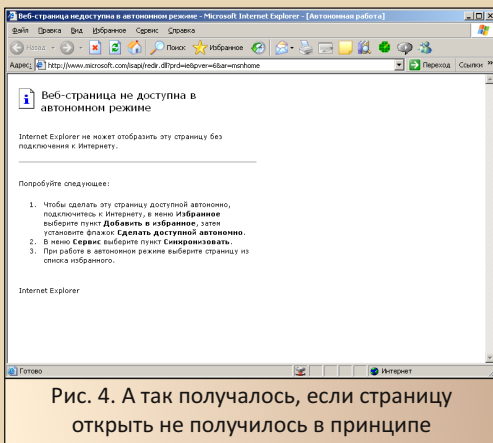


Рис. 4. А так получалось, если страницу открыть не получилось в принципе

Но, как выяснилось, для работы с кэшем существовали и отдельные утилиты – кэш-браузеры. Из книжки Виталия Леонтьева я знал о двух программах: CacheX и Internet Cache Explorer. Первую программу, увы, тогда или не удалось найти, или же с серийником какие проблемы были. В общем, тогда её так и не поставил. Зато сейчас программа доступна на Old-DOS.ru – <http://old-dos.ru/index.php?page=files&mode=files&do=show&id=102905>. Существовали отдельные версии программы для Internet Explorer, Netscape Navigator и Opera (если что, названия файлов могут быть **cachexie** – для Internet Explorer, **cachexnn** – для Netscape Navigator, **cachexop** – для Opera). На момент написания статьи (конец июня 2023-го года) удалось найти CacheX для Internet Explorer 2.02:

<http://old-dos.ru/dl.php?id=30966>



Главное окно программы показано на рис. 5. Интерфейс чем-то напоминает проводник Windows. При этом программа сама не содержит просмотрщика – для открытия страниц используется браузер. Так, версия для Internet Explorer использует для просмотра... браузер Microsoft. ☺

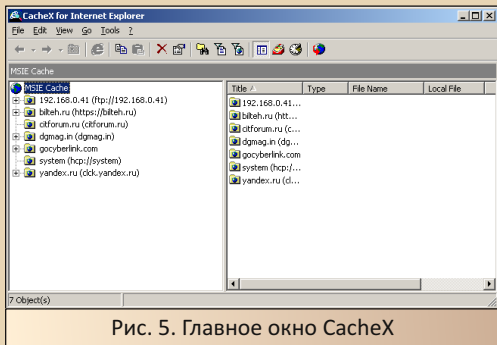


Рис. 5. Главное окно CacheX

Другая программа – Internet Cache Explorer – вроде бы, даже скачивал её, ставил, но сейчас даже не вспомню, почему она не прижилась. Сейчас версию программы 2.60 можно скачать по адресу:

<https://web.archive.org/web/20070114211830if/http://www.risingresearch.com/files/icacheru.exe>

Данная программа имеет похожий интерфейс (хотя куда больше напоминает трёхпанельный почтовый клиент) – см. рис. 6.

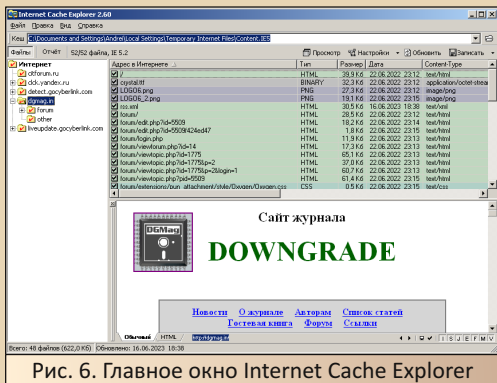


Рис. 6. Главное окно Internet Cache Explorer

В отличие от CacheX, у программы есть встроенный просмотрщик страниц.

Почему-то программы для работы с кэшем у меня так и не прижились, хотя просто так в режиме автономной работы я частенько просматривал страницы в самом браузере. Возможно, хотел именно поставить CacheX, но никак не мог её найти. Сейчас даже и не вспомню.

Качалки сайтов

Конечно же, интересующую страничку хотелось сохранить на компьютер. Конечно же, страницы скапливались в достаточно объёмных архивах. Не только у меня, как потом выяснилось. Но порой создатели сайтов делали в виде гипертекстового документа книги или же курсы, которые так хотелось выкачать полностью, и работать с ними через удобное меню, как на сайте.

Конечно же, читатель подумал об офлайн-менеджерах или же сайтососах. ☺ Думаю, не только я пользовался такими программами. Таких программ если не огромное множество, то достаточно много – по крайней мере, сам не могу похвастаться, что перепробовал даже все известные мне программы. Об одной из программ – «ДИСКО Качалке» – я уже писал в 33-м номере журнала – статья «Арсеналь отечественного интернетчика». Как-то получилось, что, несмотря на симпатию к программам «Арсенала» и «ДИСКО», именно «ДИСКО Качалкой» практически не пользовался. Но, как ни странно, пользовался плодами её труда ☺ – в прокате был диск «Радиофанат» – скачанная и записанная на диск копия сайта <http://rfanat.ru>. Причём первая страничка была сформирована «ДИСКО Качалкой». Кстати, дистрибутив программы был тоже на диске ☺.

Но сам для выкачивания пользовался другими программами. Наиболее удобной, в своё время, оказалась программа Teleport Pro.

Много версий программы выложено на сайте Old-DOS.ru (<http://old-dos.ru/index.php?page=files&mode=files&do=show&id=1103>). Какая была у меня – не вспомню – по времени



больше подходит версия 1.29 build 2050 (<http://old-dos.ru/dl.php?id=25612>) – пользовался ей примерно в 2004-2005-м году.

У программы достаточно лаконичный дизайн (см. рис. 7). При создании новой загрузки программа спросит адрес сайта, который нам хочется скачать, и глубину загрузки – на сколько ссылок вглубь надо качать.

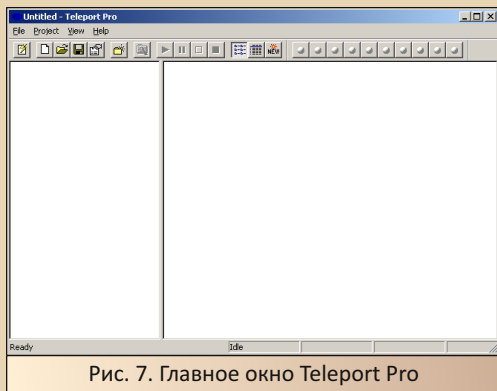


Рис. 7. Главное окно Teleport Pro

Программа достаточно удобно и информативно показывает процесс загрузки страниц сайта – в правой стороне окна показывает найденные на сайте файлы и диаграмму их загрузки. Причём программа очень точно определяла глубину – ставишь глубже 3-4 ссылки не качать – она и не качает. Потому и скачивание, как мне кажется, проходило достаточно быстро, и сайт без особых проблем можно было вытянуть и на модемном соединении.

Другой опробованной когда-то программой была WebCopier (<http://old-dos.ru/index.php?page=files&mode=files&do=show&id=6738>). Программа до определённого момента была бесплатной. Трудно сказать, какая версия была последней, но уже 4.x, как я понял, были уже платными. Я же вытягивал именно бесплатную. Версия 2.1a с сайта Old-DOS.ru (<http://old-dos.ru/dl.php?id=17258>), вроде, очень похожа (см. рис. 8).

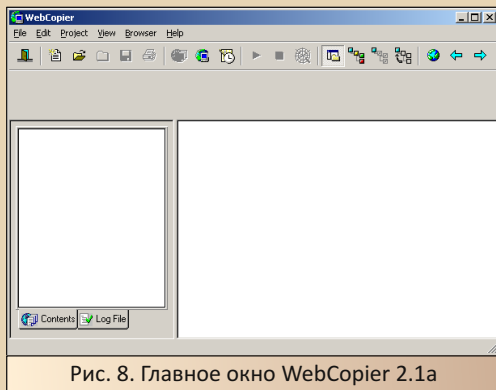


Рис. 8. Главное окно WebCopier 2.1a

У программы, которую когда-то использовал сам, был ещё достаточно симпатичный фирменный баннер – хоть и реклама, но без неё сейчас окно WebCopier кажется пустым. Когда пользовался, казалось, что программа неправильно определяет глубину загрузки и качает больше положенного. Да и загрузка файлов, как показалось, показывается не так информативно, как у Teleport Pro. В общем, так получилось, что я начинал с WebCopier'a, но при первой возможности перелез на Teleport Pro.

Менеджеры закладок или проверщики обновлений

Эти программы напрямую не предназначены для обеспечения автономной работы с интернет-страницами, но для пользователя тоже могли служить средствами уменьшения времени, проведённого онлайн. По крайней мере, я хотел использовать программы такого класса именно так – записать в список проверки обновлений новостные страницы интересных мне сайтов, интересующие разделы форумов и просто запускать проверку наличия обновлений. Так, в случае появления новых записей в разделе новостей какого-нибудь сайта, думалось, получится эту новость сразу прочитать и открывать страницу только при необходимости.

Для таких целей хотелось использовать «ДИСКО Наблюдатель», о котором также писал



в 33-м номере журнала. Кстати, на Old-DOS.ru [выложил](#) отдельно имеющиеся дистрибутивы этой программы.

Но для проверки обновлений страниц существовали и другие программы. Одной из таких программ была WatzNew. Увы, сам её не смог найти, но стараниями редактора журнала программа сейчас доступна на Old-DOS.ru. К сожалению, программа нашлась уже после написания статьи, поэтому, возможно, ей будет посвящена отдельная статья или заметка в будущем. Всё же хочется попытаться самому разобраться с ней.

Программа по интерфейсу была похожа на интернет-пейджер типа аськи. Для каждого сайта там необходимо было настроить канал обновления. Канал содержал достаточно много настроек для новичка, каким автор был тогда, да и сейчас не сильно далеко ушёл (разве что имеет больше решимости пободаться с настройками ☺), — поэтому с программой подружиться тогда не получилось.

Но сейчас речь пойдёт о другой программе, которую тогда ваш покорный слуга незаслуженно обошёл вниманием. Прошу любить и жаловать — Morning Paper. Леонтьев описывал главную особенность программы — все данные об обновлении сайтов присылаются на почту. Удобно? Кому как — мне лично не понравилось — потому и программу даже не скачивал. А зря, но об этом чуть ниже ☺.

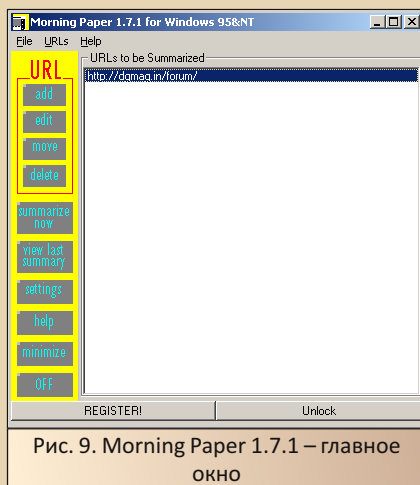
Итак, за прошедшее время сайт программы... закрылся. Найти дистрибутив программы удалось только через Web Archive ([ссылка](#)).

Также удалось скачать версию программы 1.7.1 с [FTP-сервера](#). О последней версии и пойдёт речь. К сожалению, программа имеет пробный период 10 дней, и найти комплект дистрибутив+серийник не получилось (а «магия имени» затруднила поиск номера в «Гугле» ☺ — получил кучу результатов, посвящённых именно утренней газете ☺).

Внешний вид можно увидеть на рис. 9.

Программу лучше устанавливать по адресу, где в названии нет не только кириллических букв, но и пробелов (сам сперва поставил в каталог **Document and Settings** — после

нескольких неудачных запусков перенёс программу в корень диска C:).



Итак, в окне программы можно увидеть блок кнопок URL для управления сайтами, добавленными для просмотра обновлений. Чуть ниже располагается кнопка **Summarize now** — проверка обновлений. После проверки обновлений результат будет отображён в окне браузера (см. рис. 10).

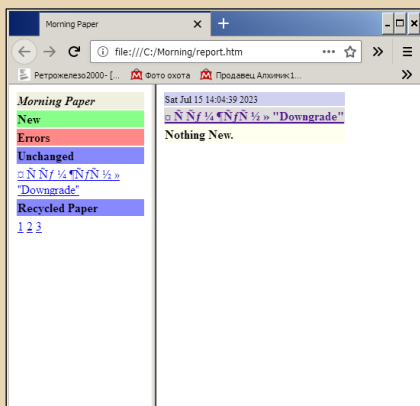


Рис. 10. Результат проверки обновления сайтов — увы, кодировку поменять не получилось — действительно, в настройках браузера почему-то не переключается



В принципе, если бы была возможность переключить кодировку (возможно, в более ранних браузерах, например IE 5.5 или IE 6.0, оно работало), вполне нормально бы смотрелось. Конечно, не как в «ДИСКо Наблюдателе», но уж лучше, чем получение результата только на почту.

Ещё одна симпатичная автору программа проверки не сильно давно была выложена на сайте Old-DOS.ru. В начале «нулевых» эту программу не видел, а жаль. Программа зовётся Internet Resources Directory и доступна на сайте Old-DOS.ru (<http://old-dos.ru/index.php?page=files&mode=files&do=show&id=101868>). На сегодняшний день (июль 2023-го года) на сайте доступна только версия 2.0 2003-го года (<http://old-dos.ru/dl.php?id=28452>) – она и была использована для опытов.

После установки программа предлагает импортировать закладки из одного из установленных браузеров (см. рис. 11).

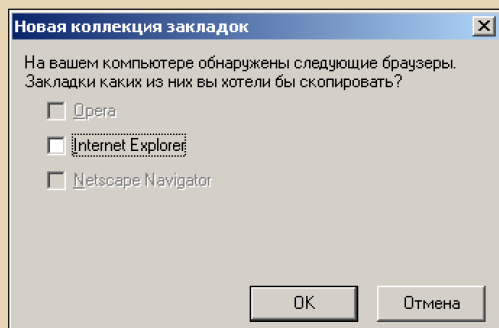


Рис. 11. После установки программа предлагает импортировать закладки одного из трёх поддерживаемых браузеров

Внешний вид главного окна показан на рис. 12. По интерфейсу программа напоминает «ДИСКо Наблюдатель». Разве что иконки более затеют уже к Windows XP.

Пользователь может создать собственные папки, куда рассортировать закладки – адреса для проверки обновлений. Также есть возмож-

ность добавления адресов вручную – вроде бы мелочь, но Леонтьев писал, что есть программы, в которых такая функция отсутствует.

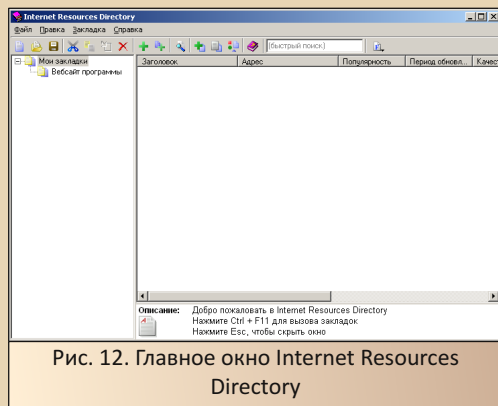


Рис. 12. Главное окно Internet Resources Directory

Из недостатков можно отметить отсутствие возможности проверки почтовых ящиков – только сайты. Запустить проверку можно или командой меню **Закладка->Проверить обновление**, или через пункт контекстного меню. После вызова режима проверки откроется диалог, показанный на рис. 13 – нас спросят, хотим ли мы проверить один адрес или все, или всю папку и т.д.

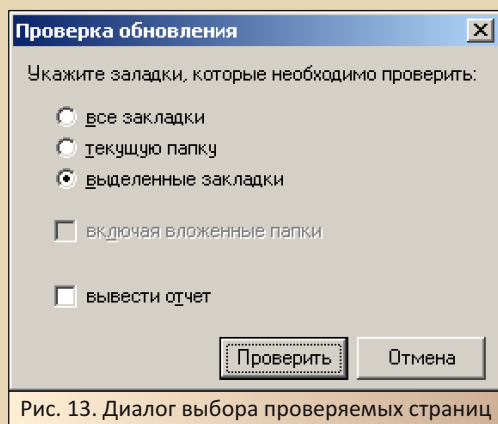


Рис. 13. Диалог выбора проверяемых страниц

При необходимости можно отметить пункт **«Вывести отчёт»** – может помочь при



возникновении непонятных ошибок, как это вышло у автора.

К сожалению, почему-то не получилось проверить обновления даже «простых» (без какого-то навороченного динамического содержимого и даже без HTTPS, хотя был сайт и с HTTPS тоже, но результат такой же) сайтов. По всем адресам получил ошибку «Winlnet error code: 2». Возможно, под Windows 9x программа работает нормально и ошибки – конкретная проблема вашего покорного слуги, но у меня действительно не заработало.

Но если бы программа работала и попала не в 2023-м году, а где-то в 2006-2007-м, и так заработала бы нормально, возможно, оказалась бы весьма полезной утилитой на компьютере автора.

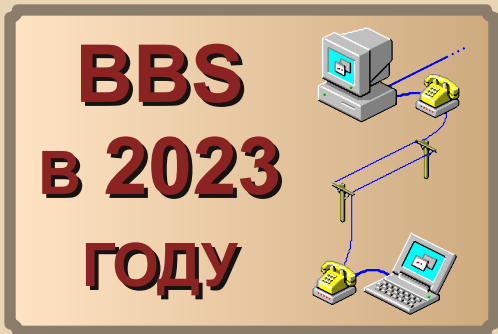
Заключение

В заключение остаётся только вспомнить фразу, когда-то, вроде, сказанную **Антикваром** – технологии появляются на свет под громкие овации и подкидывание шляп, а уходят тихо и незаметно. Также незаметно сошёл на нет пласт программ, использовавшихся для офлайновой работы с интернет-ресурсами. Причём ваш покорный слуга не касался множества технологий и программ, которые либо не использовал, либо вообще не застал. Например, когда-то предлагали заказ файлов и интернет-страничек по E-mail, были метапоисковые программы, которые отправляли на несколько поисковиков запросы, а потом отсеивали дублирующиеся и выводили результат поиска в окне (возможно, об этих программах будет отдельная статья). Здесь описаны классы программ, которые автор действительно или использовал, или пытался использовать. К сожалению, сейчас едва ли у какого интернетчика в арсенале есть кэш-браузер, утилита проверки обновлений сайтов – увы, даже почтовый клиент далеко не все используют для доступа к почте.

Но, как ни странно, концепция вкладок в браузере пользуется популярностью до сих пор, хотя многие используют для выхода в интернет безлимитный скоростной канал связи. Но, например, Елена использует вкладки для хранения заинтересовавших страниц. Добавлять страницы в закладки, а потом искать их там ей не нравится – проще оставлять открытыми вкладки и периодически переходить на них. Вот так вот.

Андрей Шаронов (Andrei88)





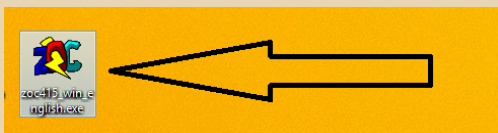
Те, кто застали модемы, помнят BBS (Bulletin Board System). На них частенько можно было скачать полезные файлы. Чтобы скачать игру или другой большой файл, людям иногда приходилось договариваться с СисОпом. Поскольку у ББС был минус – там мог в одно время сидеть только один человек. Были и ББС, где могли сидеть несколько человек, но это редкость. А о скорости модемов напоминать не буду. Также некоторые пользовались Фидо (обычно read-only) с помощью BBS. Но потом пришёл интернет. И люди стали про Фидо и ББС забывать... Но оно ещё живо. Уже через TCP/IP. Прямо сейчас мы увидим, как живут ББС в 2023 году! В основном сейчас ББСки – это копии ББС 90-х годов.

Ладно, не буду тянуть. Начнём.

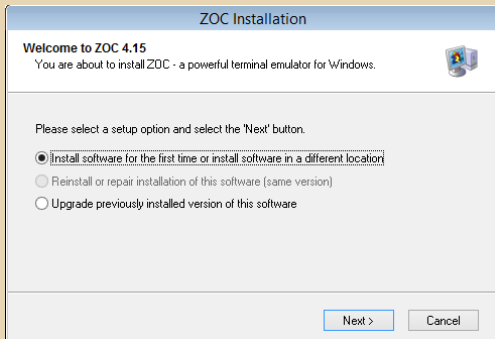
Нам надо скачать эмулятор терминала. Я предпочитаю Zар-O-Com 4.15. Тестировал на Windows XP. Как работает на системах раньше, я не знаю. У производителя на сайте можно зайти в архив. Вот сразу прямая ссылка на скачивание:

https://www.emtec.com/downloads/zoc/zoc4/zoc415_win_english.exe

Качаем. Видим такой файл.

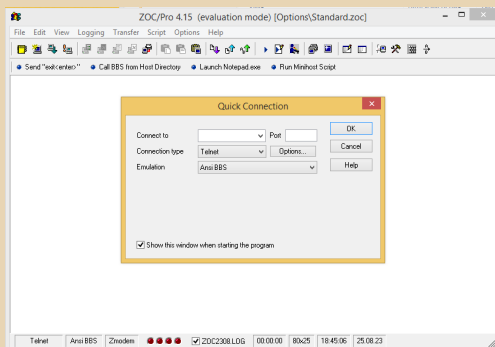


Теперь устанавливаем.



Установка – Next-Next-Done.

Запускаем.



Но нужно настроить.

Окно закрываем и нажимаем **Options->Edit Session Profile**. И в разделе **Window/Colors** меняем **Background** на **Black** и **Foreground** на **Cyan**.

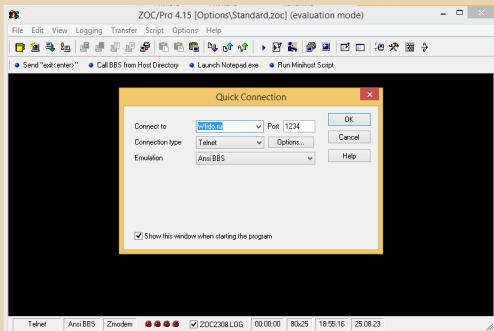
Нажимаем **Save**.

Теперь переоткрываем программу.

Можно вписывать адрес ББС. Я сижу в Фидо и решил выбрать ББСку своего босса `wfido.ru:1234`.

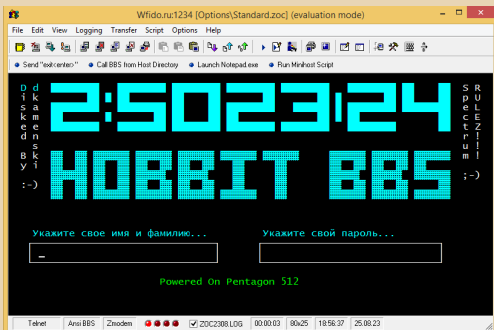
Вписываем.





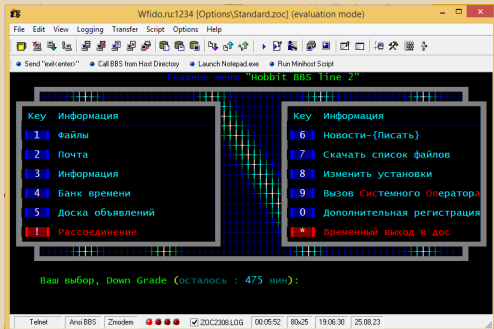
Нажимаем ОК.

Ждём...

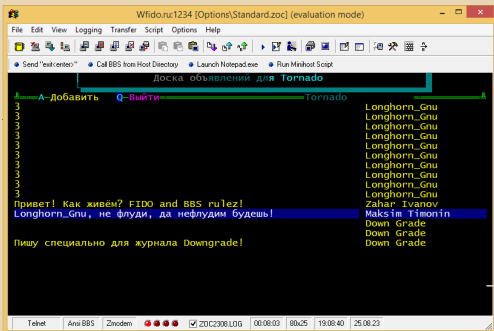


Регистрируемся. Это легко.

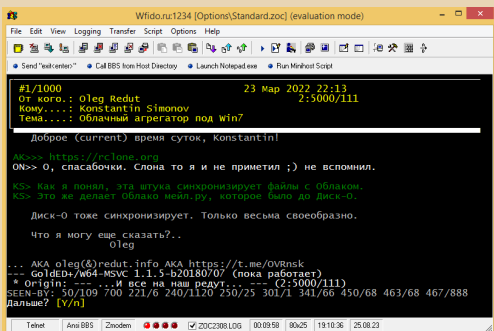
Так выглядит сама BBS:



Есть доска объявлений.



Можно читать Фидо:

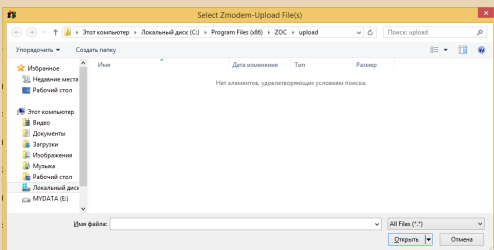


Есть файлы, можно заливать и качать.

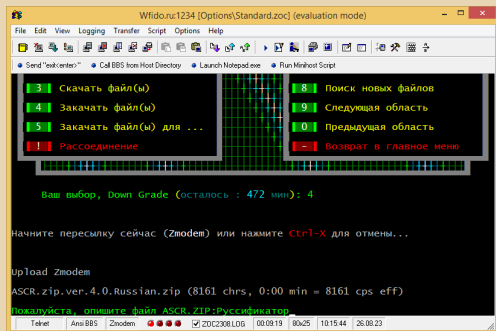
Давайте зальём игру «Подземелья Кремля» и русификатор.

Переходим в раздел **Файлы**. Там нажимаем **Сменить Арю**. Выбираем **Users Upload**. Нажимаем **Закачать файл(ы)**. У нас открывается окно выбора файла.

Выбираем.



Прочит описание. Пишем.

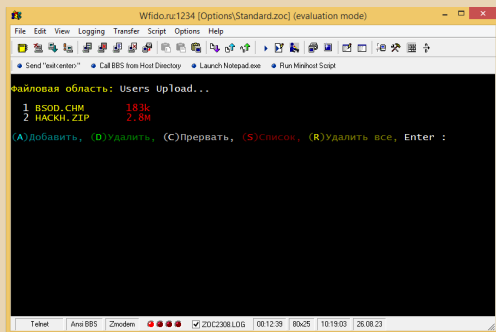


А теперь скачаем файлы.

Выбираем **Список файлов**. И там отмечаем номера файлов, которые нам нужны.

Выбрали.

Теперь нажимаем **Скачать файл(ы)**.



Нажимаем **Enter**.

Файлы качаются в папку **ZOC\Download**. На Windows 8.1 так не работает, но на ХРюше – да.

Есть ещё одна ББС:

bbs.bsrealm.net:23

Так что можете копаться.

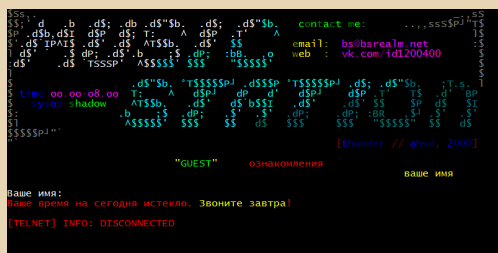
Есть ещё куча англоязычных ББС. Лично я вернулся психологически в 90-е.

Легенды некогда не умирают.

И закончу свою статью легендарной фразой:

FIDO Rulez!!! Или тут – BBS RULEZ!!!

longhorn_gnu (ака Захар)





Котечественным играм конца девяностых – нулевых годов можно относиться по-разному. В этот период выходили как действительно великие игры мирового класса, например «Ил-2 Штурмовик» и «Блицкриг», так и плохие игры, и их было, к сожалению, очень много. Наши дельцы быстро просекли, что хорошие и плохие игры на полках в магазине стоят одинаково, так незачем напрягаться – и стали брать количеством. Большинство тех игр уже никто не помнит, но некоторые, типа Lada Racing Club, успели наделать шумихи и остались в истории. Но если отбросить крайности, то посередине остаются просто хорошие игры. И вот эта золотая середина явно отличается от заграничных продуктов. Да, в чём-то многие были кривоваты, в чём-то косоваты, чем-то подбешивали и содержали спорные дизайнерские решения. Но одного у отечественных игр того периода не отнять. Они были самобытными. Наши дизайнеры не боялись экспериментировать. Так родились «Вангеры» и «Тургор», «Механоиды» и Ex Machina. К таким играм можно отнести и игру, про которую я хочу сегодня рассказать.

«Магия крови» в период своей актуальности прошла мимо меня совершенно незамеченной. Я читал обзорные статьи в журнале «Игромания», пару раз встречал в магазинах, но это был 2005-й год. В этом году выходили KOTOR 2, Quake 4, RE4, GTA San Andreas и ещё куча именитых игр. Среди таких имён не заметить небольшую отечественную игру было не столь трудной задачей. И буквально недавно, перебирая и сортируя свою коллекцию дисков, которую периодически пополняю, я наткнулся

на диск с этой игрой. Смутно вспомнил, что это такое, и решил поиграть. Пока устанавливал – убил винду и потратил вечер на восстановление, но мучения того стоили. «Магия крови» хоть и странная, кособокая, но меня зацепила.



Рис. 1. Последствия выполнения квеста по зачистке банды бандитов

Эта игра была разработана студией Skyfallen Entertainment и выпущена издателем «1С» в 2005-м году. Позднее вышло ещё мультиплеерное дополнение и полноценный сюжетный аддон, но в этой статье я буду говорить исключительно об оригинальной игре. Если вы ничего не слышали о Skyfallen, то не расстраивайтесь. Из-под их крыла вышла только пара игр: «Магия крови» с двумя дополнениями и «Death Track: Возрождение». Эти ребята занимались в основном разработкой движков. На их движках были сделаны такие игры, как «Санитары подземелий», «Не время для драконов», «King's Bounty: Легенда о рыцаре» и «Кодекс войны». Как минимум о King's Bounty вы должны были слышать. Специализация разработчиков на движках не могла не отразиться на игре. Она отлично работает на современных версиях Windows, и за те часы,



которые я успел в неё наиграть, ни разу не вылетела и вообще в техническом плане вела себя просто превосходно.

«Магия крови» относится к жанру ролевых игр и поджанру hack and slash, а если быть точнее, то это Diablo-подобная игра.

Чем можно охарактеризовать Diablo? Это несколько совершенно разных классов персонажей, каждый со своим геймплеем. Это кучи «лута» (оружия, брони и прочего), сыплющегося из врагов как из рога изобилия. Это бесшовный случайно генерируемый мир. Это непрекращающееся действие, достигаемое за счёт большого количества разномастных, но слабых врагов, которые берут в основном количеством. Этим критериям отвечает подавляющее большинство Diablo-подобных игр (Sacred, Titan Quest, Dungeon Siege). «Магия крови» не отвечает практически ни одному критерию, и при этом она всё равно остаётся в этом жанре и может затянуть игрока. Как же так? Давайте разбираться.

Начнём с сюжета. В hack and slash RPG сюжеты чаще всего пафосные, рассказывают о противостоянии героя и какого-нибудь злого божества и служат просто для обозначения финальной цели всего действия. Касательно сюжета разработчики из Skyfallen Entertainment не стали отходить от традиций и соорудили историю о том, как злодея Модо за его прегрешения низвергают из некоего «абсолюта» на землю, где он должен переродиться в теле смертного, так прожить 100 лет и только потом сможет вернуться. Модо не стал дожидаться своей смерти в лице простого человека и, только-только окрепнув, стал снова строить козни и подготавливать местный апокалипсис. Герой, подконтрольный игроку, про всяких Модо, естественно, и слухом не слыхивал, учился в магической школе, а после прохождения финальных экзаменов был отправлен помощником к главе местной деревни Эйвон. А дальше, как обычно, квест за квестом потихоньку выходим на основную сюжетную линию и аки локомотив,

прямоком, направляемся к главгаду, с целью либо начистить наглую рыжую морду, либо помочь в его злодеяния. Тут уж от мировоззрения персонажа зависит, коих тут всего три штуки, чай ни DnD. В зависимости от этого игроку будут доступны те или иные квесты, с ним будут разговаривать те или иные персонажи и концовка, соответственно, тоже будет отличаться.

Основная сюжетная линия и её завязка, как вы видите, довольно стандартна. Вступительный ролик нарисован в лучших традициях Blizzard. Очень пафосно и очень красиво. А вот что начинает происходить дальше – к стандарту отнести никак не получится.

С первых минут игры, как только игроку дадут управление персонажем, становится ясно видно, что разработчики-то наши...

Сразу после заставки и нагона пафоса они забили на всё и стали отрываться на полную катушку. Местные NPC изъясняются на все лады. Старые волшебники говорят в стиле профессоров из «Гарри Поттера». Студенты — типичные студенты из ближайшего ПТУ и разговаривают так же. Местные жители: кто косит под старорусский и использует соответствующие устаревшие выражения, кто говорит чуть ли не по фене. В общем — разброд и шатания. У нас, как говорится, своя атмосфера. Местный мир своим настроением похож на миры русских квестов типа «Петьки и ВИЧа», «Братьев Пилотов», «Штырлица» и пр. Здесь и отсылка на Голлума, и свой Лас-Вегас с блэкджеком и бойцовским клубом, и бард, поющий под гитару песню про Бронтозаябру (<https://www.youtube.com/watch?v=m-H7lTeAqH0> – обязательно послушайте), и огромные полосатые зайцы-птицеловы, и гоблинши-хозяйки, вооружённые большими сковородками, и тьма-тьмущая совершенно нестыкуемых вещей. Здесь в одном и том же городе, на одной и той же площади могут располагаться храмы светлых и тёмных священников, люто враждующих между собой. Причём светлые священники прощальной фразой говорят «да прибудет с тобой сила... света». (Напоминает других добряков, только со световыми мечами.)



Здесь на игрока могут напасть одновременно: амазонки (обязательно в бикини, а юные ещё и с двумя хвостиками), кучка лисиц, помощники Сатаны (в красных костюмах Санта Клауса), огромный муравей, летающая башка (похожая на Beholder'a из DnD), скелет, воин тьмы, заяц-мясоед, демон и чёрт ещё пойми что. Во втором акте игры даже будет обязательной к посещению локация «логово похоти», где все ходят голыми, а часть мужского населения изъясняется как стереотипные геи с соответствующими интонациями.

Местами в диалогах проскакивают робкие попытки перевести игровую атмосферу в более привычное русло, но они тонут на фоне всего остального балагана. При всей своей абсурдности, мир игры смотрится вполне целостно и, если у вас нет предрассудков на тему подобного юмора, воспринимается абсолютно нормально. Не знаю, к недостаткам или к достоинствам отнести такой подход к созданию мира. Подобный стиль был присущ многим отечественным играм начала нулевых. Забавно, что из западных творцов, которые использовали подобные приёмы, я могу припомнить только Telltale Games с их Lego-серией, двух парней с Андромеды и их серию игр про Роджера Вилко да писателя Терри Пратчетта и игры по его произведениям.

Дальнейший рассказ о наполнении игрового мира на этом моменте нужно прервать, так как без предварительного рассказа о ролевой составляющей он не имеет особого смысла.



Рис. 2. Щас спую...

Как чаще всего изображают персонажей в RPG? Обычно это мускулистые мужчины и красивые стройные женщины, а если персонаж злой, типа некроманта или лича, то и внешность у него соответствующая — как у Кошца Бессмертного. Классы обычно тоже максимально стереотипные: Воин, Амазонка, Варвар, Некромант и так далее.

В «МК» всё не так от слова совсем. На выбор на старте игры нам дают четверых персонажей совершенно негероической внешности: Рассеянный школяр — дохлого вида паренёк, но очень умный. Толстый монах — в меру упитанный седой дед с зальсынами, по совместительству и самый сильный персонаж в игре. Жена булочника — толстенная злобного вида бабулька (точно такие же сидят у любого подъезда) — самый живучий персонаж. И наконец, Цыганка — девушка наружности как из старого романа, а не так, как сейчас они ходят. По характеристикам ничем не выделяется, но обладает наибольшей удачей, ну и выглядит симпатичнее всех. Как видите, на спасителя вселенной и борца со злом тут никто не тянет. Дальше — больше.

Ролевая система игры сложна и многообразна. Останавливаться и подробно рассказывать про каждый из её аспектов я не буду, ибо их действительно много и статья рискует стать либо гайдом, либо скучным перечислением. Я остановлюсь только на самых интересных особенностях. Загибайте пальцы. Помимо обычных характеристик здесь есть: магия, навыки, татуировки, система рун и рунических слов, система алхимии, зачарование предметов и морфинг самого персонажа. В довершение всего в игре нет классов как таковых. По классике каждый класс должен обладать своим уникальным игровым процессом: варвар рубит, маг колдует. Но как ты варвара ни качай, колдовать он всё равно не научится, а некромант не сможет взять по двуручному топору в каждую руку. Здесь опять не так. Выбор стартового персонажа — это, по сути, просто выбор начальных характеристик. Он не сильно влияет на игровой



процесс. При желании можно отыгрывать как воина, так и мага.

Можно без проблем вкладывать все очки в силу и боевые навыки. Игру так тоже можно пройти, но так вы пропустите основное веселье. Название игры, так сказать, мягко намекает, что магом быть предпочтительнее.



Рис. 3. Книга магии

Магических школ в игре целых 12 штук: кости, вода, кровь, земля, благословение, воздух, призыв, свет, алхимия, проклятье, огонь и пентаграммы. У каждой из школ есть своя направленность – так, благословение — это баффы на игрока, проклятие — дебаффы на монстров, пентаграммы обычно применяются, чтобы кого-то куда-то не пустить или, наоборот, пустить только того, кого нужно. Самих заклинаний по 8 на школу, итого $12 \cdot 8 = 96$ штук. Не то чтобы сильно много, но и далеко не мало.

Игрока не держат в рамках, что учить и в каком порядке. Деревя прокачки тоже нет как класса. Учить можно что угодно и в любом порядке. Самая изюминка в том, что заклинания можно и нужно комбинировать.

У игрока есть два слота под заклинание: первичный и вторичный. Каждое из именующихся в игре заклинаний может работать в одном из 4-х режимов. Режимы такие: может быть только в первичном слоте, только во вторичном, можно поставить в любой слот и нельзя поставить никуда, в таком случае заклинание работает вообще всегда.

К сожалению, жёсткое разграничение по слотам сильно ограничивает полёт фантазии.

Сделать огненный шар, который после взрыва оживлял бы все трупы в зомби, нельзя, как нельзя сделать торнадо, метаемое в противников черепа или бьющее цепными молниями.

Атакующие заклинания в основной массе своей применяются только в первый слот, а то, что можно поставить во второй слот, чаще всего просто дополняет первое заклинание и добавляет ему какие-то свойства. Но вариантов всё равно очень много. Есть такое заклинание «Дух возмездия». Оно на месте убитого монстра поднимает скелета, который какое-то время будет кидаться во врагов костями. Никто не запрещает сделать так, чтобы вместе с костями он бросался молниями или часть нанесённого урона «вампирил» в пользу игрока.

Наука комбинаторика этим не ограничивается. Некоторые заклинания неявно влияют друг на друга. Так, например, заклинание «Череп» при ударе о противника может разлететься веером костей. Кости — это отдельное заклинание, которое можно отдельно прокачивать. Они с некоторой вероятностью могут вызвать эффект ошеломления, а в магии земли как раз есть такое заклинание.

Соответственно, если помимо черепа прокачать кости, навык земли и заклинание ошеломления, то тот же самый череп станет уже намного эффективнее, даже если во втором слоте будет стоять вообще что-то стороннее, например вампиризм. Такая же ситуация с торнадо и заклинанием «Молния».



Рис. 4, 5. Варианты тел школ магии воздуха и призыва



В качестве бонуса к системе заклинаний разработчики привинтили систему морфинга тела персонажа. Она делает так, что каждый персонаж получает свой уникальный внешний вид в зависимости от стиля игры. Звучит страшно, но на самом деле задумка очень интересная. Всё тело персонажа разбито на 8 зон, по количеству заклинаний в каждой школе. Получается, что каждая ячейка заклинаний отвечает за свою часть тела. Один угол книги заклинаний – за голову, другой – за ноги, третий – за руки и так далее. Для каждой из школ разработчики заготовили по уникальному внешнему виду. Итого 12 школ * 4 персонажа = 48 полных тел. И как только какое-то из выученных игроком заклинаний получает уровень выше, чем у остальных, персонаж приобретает часть тела этой школы магии. То у него отрастут рога, то хвост, то крылья. Каждая из деталек конструктора даёт игроку какой-нибудь приятный бонус. То скорость увеличит, то защиту поднимет. Разработчики эту систему немного не докрутили, и заранее неизвестно, какой бонус тебе дадут. Нигде не написано, что если преобладает такое-то заклинание магии алхимии, то бонусом ты получишь четыре зелёные паучьи ножки и защиту от того-то. Всё узнаётся постфактум, соответственно, и система морфинга воспринимается как приятный забавный бонус, но не более.



Рис. 6. Вот такое чудо-юдо получается, если равномерно качать магию костей, призыва и воздуха

Вот мой персонаж на момент конца второго акта. Я использую в основном заклинания

костей, призыва и света. Как видите, персонаж мой похож на Кощея с рогами, крыльями и в мотоциклетном шлеме. Как от такого только народ в городах не разбегается. Но это ещё относительно нормальный внешний вид. Некоторые билды вообще на монстров похожи.

В Diablo 3 за 15 минут игры можно несколько раз сменить гардероб. Здесь можно отыграть целый вечер, получить уровень, выучить пару заклинаний и не поменять ни одной вещи вообще. Всеми виной то, что вещи с монстров выпадают не то чтобы сильно часто, и то, что система генерации вещей тут работает странно. Обычно, в других подобных играх, вещи генерируются более-менее под игрока, с некоторой процентной вероятностью получить вещь уровнем выше, чем игрок, или ниже, или вообще не его класса. В «МК» привязки вещей к уровню игрока нет. Здесь является обычной ситуацией выпадение вещи, которую можно будет надеть в лучшем случае уровнем через 10, а в худшем – через пару прохождений. Как-то исправить эту ситуацию может система алхимии. Работает она так. В любой момент времени можно остановиться, открыть специально меню, сложить в слоты подходящие предметы и попробовать что-то схимичить. Если предметы были положены неверные – они пропадут, если верные, то что-то получится. Что нужно складывать, чтобы что-нибудь получить, можно узнать из рецептов. Рецепты либо выбиваются из монстров, либо покупаются. Но сами рецепты после изучения не дают ничего, кроме записи в специальном разделе дневника. Химичить дозволается и без них, например найдя рецепты в интернете. А рецептов много. Например, для получения случайного свитка нужен другой свиток и 48 единиц яда, а для получения случайного шлема нужно 3 одинаковых шлема + 179 единиц яда. Как видите, яд здесь служит ключом, т. к. большинство рецептов однотипны. Одна какая-то вещь + яд, 3 какие-то одинаковые вещи + яд.

Сама эта механика не то чтобы сильно полезна, т. к. две или три одинаковых вещи получить довольно сложно, а генерация случайных



свитков по одному имеет вероятность 1 к 96, что выпадет именно то заклинание, что вам нужно. Хотя свитков обычно накапливается много, тотемов и прочей бижутерии тоже, так что иногда сыграть в «казино» очень даже интересно.



Рис. 7. Местные гладиаторские бои

Чтобы совсем забить инвентарь игрока разными предметами и позволить ему персонализировать обмундирование, в игре есть система рун. Она очень похожа на систему самоцветов из Diablo. Только руны могут быть как полезными, так и вредными, надо внимательно смотреть, что наносишь на предмет. И ещё руны могут складываться в слова. Объединив несколько рун в правильном порядке, можно получить эффект в несколько раз превосходящий эффекты всех рун по отдельности. Минус тоже имеется. Слов в игре 85 штук, узнаются они из рецептов, а самих рун, ни много ни мало, 165 видов. Выпадают они ещё реже, чем полезные предметы и заклинания. Соответственно, какая вероятность, что вы сможете сложить какое-нибудь из рунных слов? Правильно, крайне низкая. Но прецедент есть, и почему бы не попробовать воспользоваться.

В общем, простор для создания «билда» просто колоссальный. Мешает этому малое количество информации в самой игре. Без стороннего гайда будет проблематично целенаправленно выстраивать цепочки заклинаний, планировать, что качать дальше, строить рунные

слова и заниматься алхимией. Но даже и без него, если качать всё подряд и честно покупать рецепты, то испортить прокачку не получится. Игра будет проходить сложнее, но пройти её будет всё равно реально.

Касательно графики — это типичный отечественный проект середины нулевых. Что я имею в виду под «типичный»? Трёхмерная картинка среднего качества. Отвращения не вызывает, но и звёзд с неба не хватает. Приятные глазу моделики, текстуры и эффекты. Но, как всегда, есть одно но. Это безудержное и неумелое использование постэффектов и цветочитров. С появлением DirectX 9 разработчикам стал доступен целый пласт новых инструментов. Вот многие наши разработчики на них и кинулись, как сороки на блестяшки, и стали применять тут и там, совершенно не понимая как, зачем и, главное, будет ли это к месту. Так и в «Магии крови» с постэффектами явный перебор. Всё, что может светиться, светится, как новогодняя ёлка. День и ночь меняются резко и основательно. А поворот камеры в ту или иную сторону полностью меняет цветное освещение сцены. Не понимаю, что этим хотели добиться авторы, но повернёшь камеру в одну сторону — всё становится синеватым, повернёшь в противоположную — сцену зальёт красным. Возможно, это как-то зависит от расположения солнца, не знаю. Выглядит странно, но для игры 2005-го года пойдёт.

Музыку для игры писали два композитора, и у каждого она получилась своя. Часть треков — это тягучие эмбиенты. Другая — это приятные слуху мелодии. При прослушивании музыки в плеере (на диске с мультиплеерным дополнением вся музыка из игры шла с бонусными материалами) разница между стилями музыкальных треков чувствуется, но разработчики расставили треки по игре таким образом, что они не перемешиваются и в какой-то степени даже дополняют друг друга, играя на атмосферу.





Рис. 8. Местные жители относятся к внешнему виду героя с нордической стойкостью и спокойствием

Теперь о том, как же «Магия крови» играется. Если постараться описать игровой процесс в паре слов, то это поместь The Elder Scrolls и Diablo. От TES здесь «живые» города и вариативность развития персонажа. От Diablo – процесс изничтожения толп монстров с целью получения опыта и лута.

Вся игра состоит из вылазок в густонаселённые монстрами локации, постепенного продвижения в сторону квестовой точки, а после – заполнения инвентаря ненужными вещами, вояжа обратно в город с целью продажи хлама и покупки чего-то нового. Города в «МК», как и в TES, «живые». Жители снуют туда-сюда по своим делам и постоянно болтают между собой. Если постоять и послушать, что они говорят, то можно узнать местные новости, сплетни или просто жалобы местных на тяжёлую жизнь. Подслушивать за жителями, пожалуй, даже интереснее, чем в каком-нибудь Oblivion. Города, в отличие от Diablo, огромные, со множеством торговцев, квестодателей, и их (городов) много (больше одного на акт), и чаще всего они занимают практически всю карту.

Под картой в «Магии крови» понимается большой квадратный кусок местности, связанный с другими посредством телепортов. Да,

генерации мира в этой игре нет, как нет и респауна монстров. Единжды зачищенная карта так и остаётся зачищенной. Не спешите огорчаться, это и не требуется. Монстров на каждой карте всегда очень много, а возвращение обратно с целью добыть ещё немного опыта чаще всего не приносит результата, т. к. в игре есть динамическая система начисления опыта. За монстров более сильных, чем игрок, опыт начисляется с дополнительным бонусом, а за более слабых, наоборот, опыта дают меньше. Монстры, кстати, довольно «жирные» и умирать от одного-двух ударов не хотят. Если поначалу игроку встречаются слабые муравьи да гоблины, которые дохнут от любого тычка, то акта с третьего монстры отъедаются до такой степени, что к каждому их виду нужно подбирать свою тактику. Кого-то запереть в пентаграмму и обстрелять, т. к. он постоянно убегает, другого задавить элементами и скелетами, третьих надо бить по площади, т. к. их много. Постоянно приходится жонглировать наборами заклинаний, проводить тактические отступления и рейды в город. Из-за этого со временем скорость продвижения по игре замедляется, но, как ни странно, играть становится разнообразнее и интереснее. Мешают монструобийству, как ни странно, сами карты. Чаше всего карты, на которых нет города, представляют собой некий ландшафт, густо засаженный растительностью с редкими примесями каких-то построек, дорог и озёр. Во-первых, одну локацию от другой без взгляда на глобальную карту отличить очень сложно. Они похожи друг на друга как близнецы, как если бы генерацией карт всё-таки занимался некий алгоритм, а не человек. Во-вторых, частенько лес, камни и прочее окружение наткано так часто, что попасть по противнику, находящемуся на расстоянии в пол экрана от игрока, становится уже невозможно. Снаряды просто врезаются в деревья. Такие заросли находятся не везде, но довольно часто.

Как я говорил выше, в «Магии крови» почти на каждом перекрёстке стоит желающий озадачить игрока своими проблемами. Естественно, за вознаграждение.





Пусто, пусто. Привет! Я пастух, а это мои доуры. Устрашило же меня, дурак, что поворочу, теперь вот пытаюсь на ходу догонять их. Может ты еще доведешь? Тут подвешена табличка, герцогу понравилась достаточно по поводу их поведения, чтобы они поборали мясо поймавшей пастух. Только не убивай их! Я пастух, поэтому являюсь для него.

Рис. 9. Огромный полосатый саблезубый кролик. С ним связана пара довольно занятных квестов

Но сами задачи не особо разнообразны: убей несколько кого-то, найди несколько чего-то, пойди туда и замочи тех-то или обороняй точку от атаки кого-то. Это «скучные», но выполнимые задания. Есть ещё группа заданий, мучения с которыми не стоят никакой награды: найди несколько случайных человек по огромному городу и поговори с каждым на время, собери несколько постоянно разбегающихся человек в одну кучу, не дай добежать куда-то кому-то и так далее. Проблема всех этих заданий – в поиске цели, а точнее, в мини-карте. Понять, где точки интереса задания, порой невозможно. Они отмечаются маленькими точками и видны, чаще всего, когда ты уже уткнулся в них носом. Обычно маркеры цели, которые выходят за рамки видимости мини-карты, остаются висеть по её краям, показывая направление движения. Тут такого нет. Или беги вслепую, или разворачивай полную карту на весь экран. А тут ещё время тикает, а ещё монстры, дома, другие жители и прочее. Награда за все побочные задания обычно невелика. Первую группу заданий можно попробовать выполнить, если хочется, но вторую нужно игнорировать всеми возможными способами. Нервы дороже.

Есть ещё сюжетные задания. Вот с ними проблем нет. Награды за них всегда хорошие, после боссов выпадают артефакты, плюс срежиссированы сюжетные задания вполне сносно: тут локация интересная, там отсылка

на что-нибудь, здесь задача необычная и так далее.

В принципе, на побочные задания можно вообще не обращать внимания и идти только по сюжетным – почти ничего, кроме разных забавных ситуаций, не потеряете.

Вот такая она, «Магия крови». Игра спорная, но необычная. Такой системы комбинации заклинаний нет нигде до сих пор. Нечто подобное есть в серии Magicka, но именно что подобное. Там магия завязана на комбинации восьми базовых заклинаний в реальное время, и сама игра не RPG, а скорее Action+Adventure. С системой морфинга то же самое. Где ещё ваш герой в один момент может быть похож на ангела, а в другой – на inferнального жука? Аналогов нет, разве что в серии Fable герой менялся в зависимости от его поступков, но не так радикально и не так динамично.

Играть или нет? Вопрос спорный. У игры есть как много недостатков, так и достоинств. Попробовать, скорее всего, стоит, тем более, что нормально «МК» работает на современных системах. Самое главное. Если у вас не Windows XP, никогда не ставьте её с лицензионного диска!!! Там StarForce, убьёте систему. Поищите лучше сборку с NoCD в интернете.

Валерий Сурженко (Hippiman)

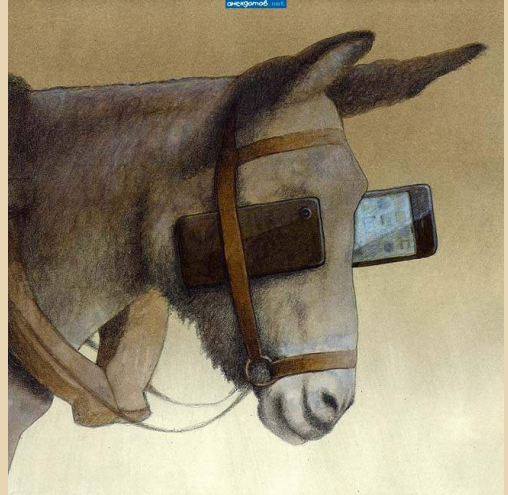


Просто разный юмор



Детский манеж с необычным названием

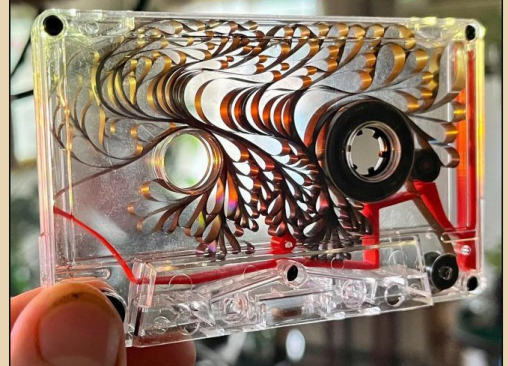
УКАЗАНИЕ ПУТЕЙ К ФАЙЛАМ В



ХОЛСТ, МАСЛО. КИРИЛЛ И МЕФОДИЙ ДАРУТ ПИСЬМЕННОСТЬ ЛЮБИТЕЛЯМ ГОЛОСОВЫХ СООБЩЕНИЙ.



ИСКУССТВО ДРЕВНИХ



НАД НОМЕРОМ РАБОТАЛИ

Дизайн/вёрстка/гл. редактор – uav1b0b

Редакторы:

Вячеслав Рытиков (euьpc)
Андрей Шаронов (AndreiBB)

Авторы:

Андрей Шаронов (AndreiBB)
uav1b0b
Валерий Сурженко (Hippiman)
BsDd (Vladislav)
Павел Ижевский (paan)
Михаил Бабичев (Антиквар)
longhorn_gnu (aka Захар)
Тарасов А.Е. (ТАЕ)
Вадим Чертков

Интервью:

Ю.А. Стариков

Сайт журнала: <http://dqmag.in>

Раздел журнала на "Полигоне Призраков":
<http://sannata.org/articles/dqmag/>

Группа ВКонтакте: <http://vk.com/dqmag>

YouTube-канал журнала: [ссылка](#)

E-mail главного редактора:
uav1b0b [собака] mail.ru