

## 2025-06-07 Про Союз-Неон

2. Кто я такой -- Никита Зимин, вот это мой профиль на гитхабе. И, в общем-то, всё, что я делаю, оно есть на гитхабе. Я программист, всегда им был. И стараюсь всё делать открыто, всё сразу выкладывать, поэтому по активности [на гитхабе] можно сразу смотреть, что у меня вообще сейчас происходит. Например, последние несколько месяцев я занимался Saboteur 1 для ZX Spectrum. Его отреверсил, и сейчас его потихоньку портирую на УКНЦ. Вот это тоже здесь [на гитхабе] всё есть.

Сегодня я расскажу про Союз-Неон. Я больше известен эмулятором УКНЦ, наверное, вы его знаете -- UKNCBTL. И немного делаю другие эмуляторы. Недоделанный эмулятор Немиги у меня есть, недоделанный эмулятор БК, который хоть как-то работает, и эмулятор Неона.

Начиналось с того, что мы начинали делать эмуляторы Неона вместе с dr. Titus (Андрей Титов). Но я как-то сдался там быстро, сдулся, а он быстро-быстро достиг большого успеха. Ну а потом он его забросил, и ко мне пришёл Саша Холодов и говорит, ну, блин, давай сделаем нормальный эмулятор, всё же есть, как бы, а не хватает такого нормального и с отладчиком, как у тебя. Это было года полтора назад, и тогда, в общем-то, и появился рабочий эмулятор.

3. Собственно, Союз Неон. Как любой советский компьютер, начинается всё с программной статьи в издании «Микропроцессорные средства и системы». То есть, вот эта самая статья. 1987 год, январь. Но понятно, что подана она [статья] раньше. И обычно в такой статье те, кто авторы, это, в общем-то, и есть те, кто занимался этим компьютером. То есть, скорее всего, это те, кто делали аппаратную часть. Программной частью там другие люди занимались. Но здесь [в статье] описывается не Союз Неон, тут описывается некий компьютер «ПК-11». Есть определённое различие.

4. Здесь описывается вот такой компьютер.  
Он на ВМ2. Просто ВМ2, не 1806ВМ2. На частоте 9,25.  
ПЗУ там большое, 128 КБ. В этом ПЗУ есть Бейсик.  
ОЗУ от 256 КБ. Вроде бы, как у Союза Неона тоже.  
Разрешение больше по горизонтали, но меньше по вертикали -- 1024 на 288. А там [на Неоне] 832 на 300.  
И таймер тактируется, как написано, «по обратному ходу луча», то есть, очевидно, 50 Гц. Сетевой таймер обычно такой и есть.

5. Ещё фигурирует везде некий ИОН 11-12.  
Но я это так рассматриваю, что вот тот компьютер, который описан в статье, и ИОН-11/12, это такие, как бы, stepping stones. То есть, промежуточные какие-то этапы в создании вот этого Союза Неона. Ну, и сам Союз Неон, он тоже не выглядит совершенно законченной машиной.  
Всплыла фотография вот такого корпуса, кто-то её нашёл. И, заметьте, здесь написано «ИОН 11-12». И ещё есть такое [красное] окошко. В окошке индикатор из семисегментных красных индикаторов. Там такая линеечка вот этих индикаторов.

Ну, и корпус от УКНЦ.

6. И потом появился вот этот вот документ, про Персональную ЭВМ, ПК-11/16 уже. Тут уже год 1990-ый, то есть, прошло 3 года. И здесь уже описывается, в общем-то, практически в точности то, что сейчас из себя представляет Союз Неон.

Индикатор, который мы там видели, он исчез. Ну, причём, по-моему, там индикатором занимается микросхема таймера.

Ну, тут я должен ещё disclaimer такой сказать. Я программист? В железячной части у меня не очень. То есть, я вам просто пересказываю то, что я вот видел, слышал, то, что говорят многие люди. В основном я программной частью не занимаюсь подобных вещей.

Значит, индикатор у нас исчез, но на самом экране справа сверху появился вот такой же индикатор, как бы семисегментный. То есть, который пикселями имитируется, который повторяет то, что было там [на индикаторе в окошке]. Осталась микросхема таймера, на которой есть вывод на семисегментный вот этот вот индикатор, но она не задействуется больше. То есть, её можно было бы ещё использовать, но эта часть не используется, там используется только та часть, которая RTC, которая таймер.

Да, как бы такой appendix остался. Так, ну то есть, вот это описание, это то, что, по сути, есть Союз Неон.

7. Теперь, был вопрос, а сколько их [Союз-Неонов] на самом деле? Я озабочился тем, чтобы пособирать ответ на этот вопрос, я просто ходил к людям и спрашивал у них. Есть два источника информации: это шильдик, вот примеры справа, и многие шильдики просто пустые.

И ещё обнаружился номер на плате, вот слева внизу под клавиатурой, там просто ручкой, как бы нарисовано, или чем-то нарисован номер. Он такой вот, два-три знака, потом месяц, потом год. Исходя из этого, ну то есть, вот слева здесь, это то, что на шильдиках, а справа, это для этих же экземпляров, то, что на плате.

Это то, что мне удалось узнать. Больше, кроме вот этих экземпляров, реальных неонов, больше никакой инфы нет. Да, это вот всё, что было там на всех фотографиях, всё, что из всех опросов удалось узнать, то есть здесь сколько? 15 штук [на слайде 11], и это всё.

Здесь мы видим, опять же, что платы создавались в марте-июле 1991 года, а корпусированы они были, получается, в июне 1992 года. Но, опять же, инфа неполная. Вот это всё, что есть.

Может быть, где-то у кого-то в закромах есть ещё там один-два Неона, можно предположить, но, как бы, кто знает.

По одному источнику их было до 200 штук, по другому — до тысячи штук. Я склоняюсь всё-таки к тому, что, ну, наверное, не больше двухсот было выпущено, и всё, что было, это, видимо, как бы экспериментальная серия, то есть это не такое широкое серийное производство.

(Обсуждение с аудиторией)

-- Ну, вот же надписи на плате 184. Если второй и третий знаки — это месяц и год, то первый — это, может быть, порядковый номер платы.

-- Да, скорее всего, но непонятно, сколько годных было там и так далее.

Ну, то есть, вот, [это вся] информация.

8. Собственно, что из себя представляет Союз-Неон.

Процессор тоже ВМ2, но 1806ВМ2. На частоте строго 8 МГц, там [на ПК 11] было больше, 9,25, обозначено в первой статье. ПЗУ всего 16 килобайт. Естественно, без всякого Бейсика, Бейсик сюда не влезет. [ПЗУ] На двух РР-ках но, сейчас, насколько я знаю, РР-ки плохо достаются, поэтому ставят РЕмулятор.

ОЗУ от 512 КБ до 4 МБ. Не знаю, есть ли у кого-то реальная машина с четырьмя мегабайтами. Все что видели, по-моему, 1024 КБ.

Видео разрешение 832 на 300 пикселей строго. Значит, опять же, в ТО упоминается, что может быть еще режим 832 на 200, но непонятно с ним ничего. Реально, по-моему, его нет просто, есть только один строгий вот этот режим. Но там с видео все очень сложно, это мы дальше посмотрим.

Палитры, значит, 2 (черно-белые), 4, 16, 256 цветов. 256 цветов максимум. Ну и, соответственно, чем больше у вас цветов на экране, тем разрешающая способность по горизонтали становится меньше-меньше-меньше.

Контроллер прерываний. Все прерывания там, в общем-то, идут на HALT. Ну вот я по программной части здесь больше буду рассказывать.

Клавиатура и корпус от УКНЦ, тут все стандартно.

Звук довольно интересно устроен. ВМ53, которые две штуки подсоединены последовательно. Одна дает частоту, вторая регулирует громкость через ШИМ. То есть, получается трехканальный звук.

9. Часы реального времени есть, на [на контроллере] которых, еще есть возможность показывать символы на 7-мерных индикаторах, но это не используются уже. Но используются как часы реального времени.

Параллельный интерфейс. На него, в частности, Sovox вешается. Ну и принтер, естественно.

Контроллер гибкого диска и жесткого диска, он совмещенный. Причем диск жесткий только MFM. Там никакого IDE не заложено совершенно. И последовательность загрузки обязательно проверяет этот MFM контроллер, что он рабочий. Так что просто так туда IDE не поставишь, но поставили как бы, но немножко сбоку, дальше расскажу. Последовательный порт. И мышь почему-то стандарта MSX, не очень понятно почему, но вот так.

Два слота для расширения, так же как на УКНС. То есть там МПИ. И вот, в частности, туда IDE воткнули.

(Обсуждение с аудиторией)

-- Потому что сам корпус УКНЦ, извините, что перебиваю, это, в общем-то, по мотивам MSX. И MSX, это тогда «витало в воздухе». Это топовый компьютер для серии для ИИ.

-- Я думаю, что его когда-то показали кому-то из министерств. И они сказали, вот, такой компьютер нам нужен в школу. И вот после этого все оглядывались на этот MSX. Ну и показывали, очевидно, Yamaha.

Скорее всего.

10. Значит, вот тот экземпляр, который у меня есть, я получил от Михаила Гусева.

Спасибо ему огромное.

Что мы тут видим? Вентилятор есть, крупный динамик. Справа вот эти вот разъемы.

Справа там ПЗУшки.

Блок питания, естественно. Клавиатура.

11. Посередине под кулером это планки памяти. Вот здесь их 4 по 256 [КБ]. Все очень ужасно было с разъемами. Потому что [планки] вот в этих разъемах, они шатаются. И точно известно, что по крайней мере один экземпляр бросили в стену, потому что человек потерял работу свою за несколько часов от того, что просто задеваешь корпус, планка шатается, сброс (RESET).

То есть это большая беда. Кто-то их у себя заменял, кто-то полностью их перепайвал, пропаивал. То есть, разные подходы применялись, но в любом случае здесь надо что-то делать. Кто-то еще ставил такие сверху наставки, чтобы [планки] жестко держались, чтобы их придерживало. Ну то есть, вот с этим нужно что-то делать.

(Обсуждение с аудиторией)

-- А в реплике они должны шататься?

-- В реплике там другой разъем ставят, но тоже ставят планки.

Еще что тут интересно. Мы знаем, что УКНЦ были построены на больших на матрицах [БМК], которые зашивались на заводе. То есть там было 2 или 3 таких матрицы.

Здесь ПЛМ-ки применяются маленькие. Которые прошиваются программатором, насколько я помню. Но их тут прям много, штук 15 что-ли.

12. Вот собственно эти планки памяти многострадальные. Они греются. Проблемы с разъемами. Это очень узкое звено, которое конечно сильно мешало.

13. Карта памяти устроена очень здорово. С точки зрения программиста, она устроена просто прекрасно. Во-первых, вот этот BM2 и 1806BM2 -- у него есть четкое разделение режима HALT и режима USER.

Где-то [на других машинах] это используется так, чтобы часть памяти в режиме HALT иметь отдельной, а здесь она прямо вся разная совершенно. Вот слева показана память, внизу нулевые адреса, вверху верхние. Для режима HALT есть 8 регистров HR0 ... HR7, которые как раз управляют режимами памяти для каждого блока по 8 КБ. Но в режиме HALT нижние 16 КБ — это всегда по ПЗУ, а верхние 8 КБ это страница ввода-вывода, и в режиме USER тоже верхняя 8 страница ввода-вывода. Ну а остальные [блоки], в режиме USER, почти вся память управляется вот этими регистрами, то есть ты можешь переключать банки по 8 КБ вообще практически произвольно. Причем они там (далее это покажем), переключаются с точностью до 4 КБ. То есть, начало этого банка с точностью по 4 КБ.

14. Вот показаны эти вот регистры, как они устроены. Верхние два бита резерв, дальше идут вот эти вот биты, которые выбирают банк, и потом там еще опции как бы идут.

И все это дело складывается, складывается адрес, который в пределах 8 КБ, плюс то, что выбирается вот этим вот регистром, бит из 4 по 13, и всего этого получается 22-разрядный адрес. Ну а 22-разрядный адрес, это адресует как раз 4 мегабайта. То есть, соответственно, получаем максимальную память для этого компьютера.

Ну и больше 4 мегабайт просто так не расширишь.

Еще что тут интересно, пожалуй, это то, что есть маскирование. Оно применяется для того, чтобы выводить спрайты на экран, то есть можно один из цветов сделать прозрачным.

15. По поводу видео. Видео очень сложное, но оно и на УКНЦ, например, сложное. А здесь оно еще сложнее. Здесь видео устроено так. У нас есть 300 строк, у нас есть таблица строк, вообще все видео, оно выбирается 32-разрядными словами. Причем там используется какое-то расширение работы протокола МПИ.

МПИ же отдает 16-разрядное слово, а тут получается, что идет одно слово, а потом сразу выставляется второе слово. И вот видео, оно как раз тоже через память и работает через МПИ, естественно, как везде в DEC PDP-11. И выбирает по 32-разрядным словам.

Значит, частота всегда 50 Гц. Здесь почему-то написано еще, 72 Гц. Видимо, это для какого-то монитора в тех времен. И что число строк на экране могло быть 200.

Программно, ничего такого нет, никаких следов, в общем-то. Я считаю, что здесь один режим экрана, это 832 на 300.

В чем заключается поддержка многооконной работы? У нас есть таблица из как раз 300 строк. Каждая строка состоит из, так называемых «видео-отрезков». То есть, мы строчку можем поделить на отрезки и на каждом отрезке сказать, какое у нас разрешение, какой режим работы с палитрами, какой видеорежим вообще. И, соответственно, палитра меняется: черно-белый / 2 / 4 / 8 бит на точку используется. Ну и по горизонтали, чем больше палитра, тем больше дублирования строк по горизонтали происходит. Но все немножко еще на самом деле сложнее.

Причем, вот эти 256 цветов палитра, она выбирается из 64K цветов. По схеме 5-6-5. То есть, в слово (16 бит) укладывается по битам 5-6-5 RGB, на зеленый 1 бит больше.

16. Вот как устроены видеорежимы. Здесь достаточно подробно расписано.

Но и в ТО это хорошо расписано. То есть, есть куча бит в режиме, в видеотрезке, в описании вот этого видеорежима. И там используется термин, так называемая «информационная плотность». Но, в результате, я как программист просто пользуюсь таблицами, чтобы понять, где какой режим. Вот здесь показаны характеристики. Какой пиксельный блок у нас и так далее.

17. И в общем-то таблица видеорежимов, она вот так вот выглядит. То есть, здесь мы видим, сколько у нас точек в строке получается в результате в этом видеорежиме.

Но надо это понимать так, что это максимальное количество точек в строке. То есть, мы виде-отрезок можем там сделать из всего нескольких сегментов. То есть, там все делится на сегменты, по-моему, на 32 сегмента. И вот видео-отрезки, они вот по этим сегментам. И на каждом видео-отрезке вы, можете делать свое разрешение, свою палитру и так далее.

(Вопрос из зала)

-- Получается, что не было общего разрешения для всего экрана, а это все было по тайлам таким?

-- Да. Есть как бы ограничивающие общие разрешения 832 на 300. Но вот в каждой из этих 300 строк ты сам как хочешь режешь на отрезки. Но не до пиксела только, а там, по-моему, с точностью до 8 пикселей.

-- Ну до тайла.

-- Как бы да. Но тем самым окна можно как раз реализовывать тем, что у тебя вот в нужном тебе блоке в этих строках есть видео-отрезки, которые с тем же самым разрешением. Причем в каждой видеотрезке еще и указывается ссылка на память, есть указатель.

Есть один забавный видеорежим. Вот здесь написано запрещенная комбинация. Она не запрещенная [на самом деле], это можно использовать, но там получается, что два пиксела соседних, которые у нас описаны в данных, складываются в один. То есть как бы такое уменьшение, zoom такой.

18. Что еще сделано как бы очень круто в этой штуке, это они прямо «прыгнули выше головы», это виртуализация. Сделана виртуализация портов и виртуализация виртуальных прерываний (VIRQ).

Виртуализация портов. Есть определенный диапазон портов. Причем тут в ТО указан более широко, на самом деле он более узкий, как здесь [на слайде] сказано.

Виртуализация происходит только в режиме USER для той памяти. То есть для самих пользовательских программ. При обращении к этим портам на следующую команду происходит прерывание, и это прерывание HALT. Но при этом прописываются регистры, это [регистры] HR0, HR1, в которые записываются адреса, к каким портам мы обратились. В одной команде [конечно] мы можем обратиться к двум адресам, поэтому здесь два адреса записываются. А уже дальше, программно в прерывании HALT мы разбираемся, что случилось. Там в HALT складывается/сбрасывается всё, что бы ни случилось, очень много разных типов прерываний. В частности, обращение к [виртуальным] портам. И там [в прерывании] мы уже смотрим: если эти регистры заполнены, значит это [обращение в] виртуальный порт. [Тогда] мы идем в таблицу виртуальных портов, смотрим, на кого он сейчас назначен, на какой процесс. Вызываем этот процесс и ему передаем ему управление.

Ну и соответственно, можно в результате подставить, что результат команды вот он собственно такой. То есть с порта мы прочитали вот так.

И второе это виртуализация VIRQ. VIRQ я напомним, это прерывание, где говорится, что сейчас будет прерывание, а потом на шину МПИ выставляется адрес прерывания. Здесь реализовано это вообще очень необычно. Прерывание это же как бы устройство у нас выполняет, а здесь у нас устройство виртуальное.

И обратились к какому-то сегменту программы, к какому-то процессу и он решает, что сейчас у нас должно быть VIRQ. Он выставляет определенный бит в каком-то порте, который ставит нам VIRQ на шину. А потом происходит тайм-аут шины.

То есть, мы поскольку адрес на шину не выставляем, мы только выставили, что у нас прерывание собирается быть в VIRQ, то происходит так, что мы трапаемся по определенному адресу, который процессор задает. И в этом обработчике прерывания на то, что у нас не пришел адрес прерывания, мы как раз смотрим, ага, это на самом деле обращение, то есть это выставление виртуального прерывания и мы делаем опять же тот VIRQ [программно], то есть вызываем там какую-то программу.

Чем это плохо [в реализации], собственно, [тем что] тайм-аут долгий. Это 64 такта процессора. То есть, на каждое виртуальное прерывание долго получается ждать. Ну и опять же, виртуализация портов, она тоже долгая, потому что она делается программно. Мало того, она еще и делается перед предыдущей командой, на следующей команде.

19. Контроллер жесткого и гибкого диска, он совмещенный здесь, то есть там стоит между ними, ну то есть это на самом деле две микросхемы контроллера, но у них еще есть общий буфер и есть там через определенные порты общее обращение. То есть, здесь работа с программной части происходит так, что я выбираю, к какому устройству обращаться, потом говорю, прочитай мне столько-то секторов, то есть от 1 до 4, потому что буфер 2 килобайта. И [затем] просто ждём прерывания.

Прерывание пришло -- начинаем читать из буфера. То есть, все просто шикарно в этом плане. Это конечно не DMA, на самом деле, но что-то похожее не DMA.

(Реплика из зала)

-- Это асинхронных ввод-вывод?

-- Да, это асинхронный вот вывод. То есть, программно здесь не надо там соблюдать четко тайминги, выбирать...

-- Ты дал команду и пошел выполнять следующую инструкцию?

-- Да, просто ждешь, можешь там экран отправлять, еще что хочешь, звук там делать.

В этом плане чтение очень удобное. Ну и потом из буфера ты читаешь в том темпе, которые тебе интересен, и из буфера читается быстро.

Но за счет этого, IDE сюда воткнуть довольно будет сложно, потому что это нужно выкидывать полностью контроллер вот этого MFM, вставить здесь контроллер IDE, но и при этом модифицировать BIOS, потому что там в процессе загрузки проверяется вот этот контроллер, что он отвечает по крайней мере прерыванием.

20. Таймер. Я уже говорил, что в первом описании [на ПК-11] там понятно было, что таймер 50 Гц. Здесь [в Союз-Неон] таймер сделан странно, я бы сказал так. Обычно для PDP-11 у нас используется RT-11. В RT-11 у нас таймер либо 50 Гц, либо 60 Гц.

Понятно почему, он с кадрового прерывания обычно берется. Здесь таймер берется с RTC, он 64 Гц.

В результате сделан патч для RT-11, который пропускает некоторые тики этого прерывания. В результате мы получаем систему, которая немножко не «система реального времени». У нас уже поскольку не кадровые прерывания идут, а иногда пропускается, то если мы, например, звук туда поставим, он будет немного заикаться.

21. Вот тоже вот схема на ВИС3, как сделан звук. Я уже говорил, здесь первая микросхема задает частоту, 3 канала, а вторая [микросхема ВИС3] через ШИМ задает громкость.

Для меня это сложно было для эмуляции, ШИМ сложно эмулировать, то есть нужно средние какие-то считать и так далее. Пока вот как раз в эмуляторе Неона звук эмулируется плохо, но мы двигаемся в этом направлении. Что еще? Таймеры, естественно, можно как таймеры просто использовать, просто считать отчеты времени.

22. Двигаемся постепенно к программной части.

В процессе загрузки у нас происходит сначала проверка памяти, причем память может быть до 4 МБ. Он проверяет и может исключать из работы сбойные участки. Ну у него есть там сегменты, которые обязательно должны работать, иначе ничего не заведется. Но какие-то дальние более сегменты совершенно спокойно он может детектировать и оставлять, что [мы их] не используем. То есть как бы можно с какой-то памятью сбойной работать, видимо.

Дальше проверка устройств, подготовка процессов. Процессы сейчас объясню, что такое. И одним из процессов является эмулятор терминала VT-100.

Что интересно, используется распаковка в процессе работы BIOS-а. То есть, всего 16 КБ, но ОЗУ у нас гораздо больше, чем ПЗУ, поэтому то, что используется распаковка, это очень неплохой здесь ход, на мой взгляд. Причем явно они еще и подбирали параметры распаковки, чтобы это все ужималось очень хорошо. Интересно, что при ошибке начальной загрузки из BIOS-а происходит переход в пульт. То есть, там есть последний порт. И вот на этом последовательном порту можно с другой машины общаться с этой по пульту и смотреть, где что сбоит. И там даже что-то делать. Например, память можно немножко проверять.

23. Показан мини-пульт на COM-порту.

Если что-то произошло, он выдает вот такой аутпут (вывод на терминал). Причем, заметьте, здесь ПК11/12 еще называется, а не 11/16. И вот есть ряд команд характерных для пульта, которыми можно что-то посмотреть и даже G сделать -- начать выполнение с какого-то адреса программы выполнить. Саша Холодов говорил, что он с помощью этого пульта отлаживал Неоны, и это помогало тоже.

24. Собственно, мы уже дальше плотнее и плотнее передвигаемся в программную часть. Наверное, вам известны такие ребята "ASP Corp."

Они были известны тем, что они на ряд машин, начиная, наверное, от ДВК, делали ряд игрушек. Militer Marathon, Pif-Paf, Sheriff, Кот Рыболов, еще что-то.

(Реплика из зала) -- Крутые игрушки были.

Да, для того времени -- тем более. И мало того, это еще и переносимый был код. То есть он и на УКНЦе это работало, и на Немеге это работало. Это было круто. И очень здорово, что их взяли тогда, привлекли к работе над этой машиной. То есть вот подлинно известно, есть договор о том, что действительно они делали программную часть этой машины.

(Реплика из зала) -- Кто-то из них, кстати, сейчас владелец Авито.

25. По поводу процессов. Вот [картинка] после запуска и после загрузки RT-11, минимальный набор процессов у нас вот такой.

Что такое здесь «процесс»? У нас есть виртуализация некая, и есть ряд процессов, между которыми происходит постоянное переключение управления. Это невытесняющая многозадачность. То есть каждый процесс, он должен что-то поделывать, и отдать управление. И так происходит в цикле. То есть, крутится ряд таких резидентных процессов, которые каждый что-то делает. Например, это работа с диском, это обслуживание клавиатуры, это обслуживание оконной системы, это эмулятор VT-100, сам терминал.

Вот, собственно, то, что вы синее окно вот здесь видите, это эмулятор VT-100.

(Вопрос из зала) -- А черное вокруг, это твой эмулятор с вот этими обозначениями? Все черное, да, это все окно [эмулятора]. А вот справа вверху это эмуляция вот того семисегментного индикатора, который раньше был красным и физическим.

26. Один из процессов, который можно запустить, он называется pul.prc, вообще, запускаемые программы называются .sav, а процессы это тоже .sav, но переименованные в .prc. И вот один из процессов такой, что ты нажимаешь на клавишу СТОП, и вот такой вот рорир-чик появляется, в котором можно переключаться

между типом памяти, смотреть процессы, какие запущены там, работать с памятью, переключаться между типом памяти, смотреть процессы, какие запущены там, работать с памятью, просматривать порты даже, такие вещи. Довольно удобно, и для того времени тоже довольно продвинуто.

27. Большая часть описания, что сейчас есть к этой машине, это описание ВПО. Вообще, описание там сейчас довольно подробное и довольно объемистое. Наверное, ни по какой машине нет такого серьезного описания, по крайней мере, по программной части нет. Даже BIOS есть, есть исходники всего BIOS, и ребята даже перекомпиляли его, то есть, он вполне рабочий исходник. Значит, ВПО — это такой API, это такие программные интерфейсы, которыми можно что-то делать с системой. Ну, например, типично, когда мы запускаем программу, и мы хотим сделать окно, мы говорим, что у меня будет какая-то ария, в которой будут данные для окна, у меня будет само такое вот окно, я возьму вот такую палитру из тех, которая есть, собственно, заполнить данные палитры. Ну, то есть, вот это все можно через API-вызовы делать.

Эти API-вызовы, ну, это все на MACRO, естественно. API-вызовы, они выглядят вот так вот. Буквально мы что-то помещаем в виртуальный порт и ставим команду NOP. Почему команду NOP? Потому что прерывание происходит на следующей команде. Если у вас что-то там дальше еще есть, оно может интерферировать, поэтому лучше оставить NOP, чтобы у вас это прерывание исполнилось и нормально вернулось. В данном случае я как раз здесь вот делаю ария, это область памяти, которая будет под окно использоваться. Я говорю, что вот у меня ария такая-то, а потом уже в определении окна я эту арию подставляю.

(Реплика из зала) -- Блин, прикольно, очень на ARM похоже, там эти вот линии задержки тоже так же.

-- Ну, это то есть уже более высокоуровневое какое-то API, чем вот обычно предоставляется для компьютера [того времени].

28. RT11. Это обычная RT11SJ, да, то есть однозадачная, с несколькими патчами. По крайней мере, это патч таймера, о котором я говорил, да, потому что он работает на 64 Гц. И второе, это патч вывода строк, потому что, ну, вы знаете, вот для PDP11 там есть всегда стандартные регистры терминала, и туда отправляется по одному символу. Но здесь, если отправлять по одному символу, это будет немножко долго, потому что там как раз отработывает HALT, отработывает виртуализация, поэтому здесь можно в этот же стандартный порт передавать не байты, а слова, и слова — это адрес строки. Тот же самый HALT отработывает, но сразу вся строка выводится, а не по одному символу.

То есть за счет вот этих вот, того, что мы убираем лишние прерывания, все происходит гораздо быстрее. Ну а так, в целом, это обычная RT-11, также точно с обычными устройствами, да, то есть тут никаких сюрпризов.

(Вопрос из зала) -- Разве что вы будете эту тему дальше освещать, взаимодействие процессов и RT11. Вот эти процессы, я так понимаю, они относятся к BIOS-у системы?

-- Они с ВПО взаимодействуют, а не с RT11.

-- Ну, невытесняющая многозадачность — это они по ВИЛ переключаются, да?

-- Они [процессы] сами говорят, что продолжай, то есть освобождают.

-- Хорошо, а если программа, которую писали с RT-11, они про это все не знают? И они не обязаны ничего такого делать?

-- Ну, программа — это не процесс.

-- Ну хорошо, а как процесс получит управление, если программа ему не передаст?

-- Ну, процесс — это .SAV-файл, в котором определен вот этот участок, который нужно сделать на очередном цикле. То есть, например, ты обрабатываешь клавиатуру, тебе нужно опросить порты и записать, куда эти значения, и потом отдать управление.

-- Да, я понимаю. А как они диспетчеризуются? По прерыванию или как-то еще?

-- Просто в цикле вызываются все. Насколько я понимаю. Но я не вдавался.

-- Процессор один. Ну, это интересная вещь.

-- Нет, я туда глубоко не вдавался, но можно посмотреть в исходниках, как это делается.

29. Дальше программу часть рассматриваем. Значит, первое — это дема для ИОН 11-12. Но при этом она написана на языке высокого уровня, но я не помню, там Паскаль или Си. Но она работает на Союз-Неоне, то есть совместимость обеспечивается. И, ну, она что-то там покажет, какие-то окошки, какие-то рисунки, что-то такое.

30. Вот еще одна из самых старых программ.

Опять же, почему самых старых? Потому что 1989 год, и это тоже для ИОН-а программа. Это «электронная таблица». Но она работает в окне, здесь видно прямо, что вот это вот бордюр окна, вот тут видно, что он срезает там другое окно, там терминал под ним. А это вот окно с неким TurboCalc.

(Вопрос из зала) -- Это зачаток в Excel'е?

-- Да, это нечто похожее. До Excel'а тоже на Западе были программы «электронные таблицы», которые в DOS работали. То есть это как бы та же идея.

(Вопрос из зала) -- Так, а у машины был всегда по жизни графический режим, получается, да?

-- Да, то есть нет как такового текстового режима.

-- Знакогенератора аппаратного не было?

-- Это вот очень на Поиск 1 похоже. Там тоже самое.

-- Там [в BIOS Неона] встроен какой-то шрифт, причем, опять же, через ВПО шрифты можно переключать. И в BIOS встроен вот эмулятор терминала. То есть вот сейчас его это окно закрывает, но мы дальше на него ещё посмотрим. То есть ты всегда видишь, по крайней мере, окно вот с этим эмулятором терминала.

(Вопрос из зала) -- И это всё в 16 килобайтах упаковано?

-- Ну да.

31. Наиболее известно вот эта демо, которое показывает кадры из игрушки Defender of the Crown. По-моему, версия для Atari ST. Но это просто скриншоты, которые показываются в окнах. То есть здесь куча-куча окон показываются все время, они там друг на друга накладываются, они масштабируются по-разному. И там еще играет музыка, не трехголосая, одnogолосая мелодия. Причем мелодия реализована так, что запускается процесс snd.prg, и в этот процесс отправляется файл мелодии. И он как бы фоном играет.

32. Здесь опять же мы видим тоже эмулятор терминала, то есть это тоже та же самая машина. Здесь запущена система ASPECT. То есть это вот по названию ASP Corp. Так же

примерно, как вот в Windows 3.11, это оболочка над MSDOS, то ASPECT это оболочка над как бы RT-11 и вот Союз-Неон.

То есть здесь несколько другие окошки, здесь уже есть свое какое-то меню, из которого можно выбирать программы для запуска. То есть немножко все более наворочено, некий шаг в сторону более продвинутой графической операционной системы. Но в центре опять же мы видим вот это вот окно эмулятора терминала, в котором работает RT-11. То есть ASPECT, он запускается из-под RT-11, в виде вот ряда процессов, вот этих вот .PRC, которые мы здесь видим.

33. Это одна из программ, это некий двухпанельный файловый менеджер, который был в ASPECT, то есть это часть ASPECT.

34. Калькулятор. Ну и слева здесь видно меню, которое в ASPECT позволяет выбрать... (Вопрос из зала) -- А это какой год? 92 где-то? 93?

-- Ну сам ASPECT, по-моему это 91, 92, а вот программа для него может быть и позже.

35. Спрайт-редактор. Гречишкин это как раз из ASP, по-моему.

-- Дальше. Подождите, я не усмотрелся. Красиво.

-- «Их разыскивает милиция?»

-- Ну, видимо, это со сканера уже, кстати говоря.

-- С паспорта.

36. Это тот же спрайт-редактор.

37. Это редактор Пионер. Это графический редактор уже.

-- А на УКНЦ случайно его не было?

-- Нет, это все в ASPECT. Это просто так на УКНЦ не перенесёшь.

-- Вопрос, а какая палитра у него? Сколько он цветов может?

-- До 256.

-- И всяких этих историй там с атрибутами, естественно, у него нет?

-- Есть. История с разрешениями.

-- Нет, там нет такого, что атрибут на несколько пикселей нет. То есть, клешинга здесь нет.

-- А разрешение какое?

-- До 832 на 300.

-- Прямоугольники, естественно.

-- Я уже прикидываю, что рисовать.

-- Черно-белое.

-- Не вопрос, я дальтоник.

38. Это программа печати файлов на принтере. Ну, то есть, видимо, здесь выбираются файлы и отправляются на принтер.

Это тоже часть Аспект.

-- Крутое кольцо написали.

39. Калькулятор тоже. Но это не под Аспект, а просто под Союз-Неон. Ну, что-то такое. Тригонометрические функции есть, то есть, это, видимо, во флоат числах вычисления. И слева вверху там на фоне виден список процессов.

-- С процессами надо разбираться.

40. Под Аспект «Минное поле», ну это «Сапёр» очевидно игрушка.

41. Тетрис. Не под Аспект, просто под Союз-Неон.

42. Некий Пэкман. Я только не знаю, он под Аспект или нет, но похоже, что да.

43. Так, Максим Сапронов. Видимо, он увлекался ФИДОнетом и он написал для Союз-Неон ФИДО-мейлер. То есть, это взаимодействие с модемом и отправка/принятия файлов. То есть, это не то, что там редактор GoldEd, нет, это только отправка и получение.

-- Типа T-Mail?

-- Да.

Где-то в статьях я видел, не помню, кто писал, что ему не удалось завести Союз-Неон с модемом больше, чем на 19200. Я это связываю с тем, что они сделали из «системы реального времени» -- систему несколько нереального времени за счет виртуализации портов, за счет таймера.

44. Так, что еще было [из софта]? Вот, собственно, перечисление в одном из файлов для Aspect'a, что здесь было.

Но не показано только игра Diamonds.

-- Типа Boulder Dash?

-- Не совсем. Нет, там попроще что-то.

-- Boulder Dash тоже написан, но в последнее время, недавно BlaireCas его написал. То есть, вот, то, что ASP делали в Aspect'e, это вот список, он достаточно полный.

Непонятно, что такое «файловый монитор». То есть, мы там видели файловый менеджер. Вот это он или нет, не ясно. А остальные вещи, они, вроде как, есть.

45. Что еще было? Значит, был такой человек в RPC, по-моему, в тусовке он известен, но, по-моему, он уже умер, насколько я помню. Он занимался Союз-Неоном, и он написал для Союз-Неона эмулятор БК. Причем, эмулятор БК честный, в том плане, что, например, эмулятор БК на УКНЦ, там нужно патчить файл для БК, чтобы не обращаться к тем портам, которых нет, и делать маппинг портов на те, что есть.

А здесь RDC реализовал это честно, то есть, просто берёшь БК-шный файл и его запускаешь, но проблема в том, что системные регистры, некоторые БК-шные, они находятся вне поля эмулируемых портов, поэтому он [RDC] там [на своём Неоне] сделал перемычку, которая это поле расширяет. И за счет этого получилось, что те порты тоже можно эмулировать.

И от Михаила Гусева был просмотрщик GIF.

Вообще, я здесь показываю только тот софт, который был «тогда». Есть новодел, там тоже много чего написали уже, я этом здесь не рассказываю.

46. И, собственно, к новоделам всё-таки переходим. Как вы знаете, уже есть [реплика]. Одна из плат была полностью распаяна, с неё была полностью снята схема,

восстановлена, и заинженерена новая плата с какими-то небольшими изменениями. Например, здесь видно, что добавлена кнопка Reset.

Еще позже было выявлено, что в режиме работы с палитрой 256 цветов есть ошибки. Там делали такую программу, где известная актриса какая-то была показана и вот она прям скроллинг такой, и на некоторых машинах там появляется рябь, потом постепенно выяснили, что в какой-то из ПЛМ-ок там была ошибка и что-то именно на плате еще нужно было поправить, то есть вот в новых редакциях, кажется, это исправили. Ну, а так, в основном, это та же самая плата, которая была.

-- Панельки держат все хорошо?

-- Панельки держат, да. Но сменили разъёмы, как я понимаю, на что-то другое.

47. Так, новодел это вот этот IDE-контроллер, у него несколько необычная судьба. Есть такой человек S\_V\_B, он написал сначала драйвер для него [контроллера], драйвер без устройства.

То есть у него была идея, как сделать вот IDE на Союз-Неоне, и он написал драйвер.

Спустя год, что ли, сделали устройство. Драйвер заработал, потом Алексей Кислый его еще поправил, чтобы он [драйвер] там обрабатывал более правильно какие-то угловые ситуации. То есть сейчас это устройство работает, но проблема в чем? В самом BIOS'e у этой машины заложено то, что виртуализация как раз вот этих вот устройств, и они даже называются там нестандартно как-то.

То есть, свои драйверы для устройств, ну, как вот на УКНЦ MZ1, тут, естественно, свои драйверы. И эти драйверы предполагают общение с ВПО. И, соответственно, драйверы очень простые, они как бы редиректят запросы.

Но зато получается, что тот же АСПЕКТ система, она полагается вот на эти драйверы, что устройства будут называться именно так. Поэтому аспект, например, с IDE вот этим не работает. Или что-то там нужно еще доделать.

Но зато вот сейчас, по крайней мере, жесткий диск есть.

48. Эмуляция, собственно, и заканчиваем с этим.

Как я говорил, было два эмулятора. Сначала dr. Titus (Андрей Титов) написал его. Ну, у него эмуляторы EmuStudio, и у него они очень специфичные. Они только под Windows. Он всегда эмуляцию делает так, что он его делает от видео по скорости. Поэтому, если частота вашего видео отличается [от частоты кадров эмулируемой системы], то частота эмулятора становится другой. И частота звука за этим идет, например. Я иду либо от таймера, либо от звука. Поэтому я постараюсь синхронизировать все эти устройства, но у меня более страдает, что у меня менее точно по тактировкам эмулятор. Но, тем не менее, софт, в общем-то, работает. Большинство, там, 99,9%.

Вот то, как сейчас выглядит этот эмулятор, и у меня, как обычно, достаточно продвинутый отладчик.

-- И плюс у тебя есть кроссплатформенная версия и, по-моему, черная версия?

-- Есть кроссплатформенная, да, на Qt, но несколько недоделанная. Отстает от Windows-версии. Черная нет, там другое, для UKNCBTL.

49. Саша Холодов сделал полную эмуляцию, ну, как это эмуляцию, симуляцию Союз Неона в FPGA. То есть вот здесь это на Tang-e сделано, показано. То есть это полноценный Союз Союз. Он все делает так же, как и настоящий, вот такая маленькая коробочка.

Плюс он [Холодов] сделал UNEON для Mister'a, то есть на MiSTer FPGA тоже можно. На этом, как бы, все.

50. Следующий слайд с кучей благодарностей всем тем людям, которые помогают, которые ищут документы, выкладывают это все в открытый доступ. Масса людей помогали. Я бы, конечно, один не дошел до конца с этим эмулятором и не получил бы работающую систему.

Вопросы?

-- Ну вот, надо разбираться с тем, как процессы взаимодействуют в T11.

-- Да, интересный вопрос. Мне кажется, что взаимодействие через драйверы происходит, а сами процессы... Интересно, как многозадачность реализована на самом деле. Я не разобрался в деталях.

-- С современной точки зрения можно даже сравнивать с похожей системой типа Windows 3.11. Невытесняющая многозадачность, она как работает.

Диспетчеризируемые куски, процессы, неважно кто, они друг другу по согласованию, по договоренности передают управление с помощью директивы Yield. Она по-разному называется в разных программных средах, но она всегда есть.

Но ее должны придерживаться все. Если кто-то один не придерживается, система зависает. А тут у нас получается, в RT-11 она ничего про это не знает.

Пользовательские программы, например тот же Tetris, он тоже ничего про это не знает. Он может работать по прерываниям и на этом все. Получается, что вариантов процесса взаимодействовать с системой очень мало, это только прерывания.

-- На мой взгляд, здесь даже RT-11 ни при чем. Потому что RT11 крутится, взаимодействует с этим процессом, который эмулятор VT-100. То есть, процессы работают еще до RT-11.

-- У нас же камень один, физическое ядро одно.

-- Да. Как детально там реализовано, я не скажу точно. Но мне кажется, что по очереди вызывается каждый процесс, а он должен так можно быстрее сделать свою работу и отдать управление.

-- Да, но вот все процессы с ВПО отдали управление, а потом отдали куску хода, который ничего не знает про Yield. Он никому не отдал. Дальше только прерывания. Обычно в операционной системе диспетчеризация происходит по прерыванию. То есть, прерывание дернуло, мы перешли в режим ядра, погнали, переключили контекст, и операционка пока не решит, что все сделано, она не отдаст управление. Здесь, скорее всего, тоже так должно быть.

-- Здесь для это может быть прерывание таймера используется, возможно?

-- Я других не вижу вариантов. Кстати, как это проверить можно? Можно перехватить прерывание таймера, и если все повиснет, значит направо.

-- Здесь все проще, потому что есть исходник. Можно просто идти и смотреть, как процессы реализованы. Там прямо исходник-исходник, который с комментариями исходными, с макросами исходными.

-- Мне кажется, что здесь эти процессы, это некая такая оболочка ко всей RT-11. То есть, RT-11, она как бы не переключается.

-- Судя по вот этим вот двум, темам для виртуализации -- VIRQ и там, что еще там было?

-- Порты.

-- Да, они... А, еще плюс у нас там же есть MMU, блок управления памятью. Они активно использовали это всё.

-- Здесь, с точки зрения памяти, здесь как? Что у процесса, у каждого, у него набор вот свой из этих 16 регистров. Ну, не 16, из 8, да? Потому что USER mode. И когда ты передаешь управление процессу, ты перед этим восстанавливаешь вот эти вот регистры памяти. И тем самым ты переключаешь память на него. Для него память не изменилась. Она такая же, как была при его предыдущем запуске.

-- А это именно регистры или адреса памяти?

-- Ну, это порты, вот эти вот HR, которые...

-- Верхняя страница памяти?

-- Верхняя страница памяти, да. То есть, они из HALT будут доступны.

-- Система команд обычная PDP-шная?

-- Да, конечно. это VM2. Это тот же самый, что на УКНЦ. Но, только это 1806, в другой корпусировке. И разница между ними существенная ещё в том, что в 1801VM2 делится пополам частота внутри. Входная. А в этом не делится, насколько я знаю. Поэтому он побыстрее работает.