

# **ИНСТРУКЦИЯ К ПРОГЕ imitator\_1\_0.exe** версия 10.07.2017.

## **СОДЕРЖАНИЕ**

0.	О проге, системные требования, запуск .....	2
1.	Быстрое ознакомление с имитатором.....	3
2.	Управление дисплеем .....	7
3.	Работа с НМЛ.....	10
4.	Работа с ТПУ.....	13
5.	Запуск Бейсика, Фортрана, Выстры. «Бит b4» КД.....	17
6.	Работа с машиной без дисплея.....	38
7.	Регулировка быстродействия имитатора.....	45
8.	Перечень программ в txt-папке, прилагаемой к данной версии	47
9.	Приложение: фотографии.....	53

## 0. О проге, системные требования, запуск

- Благодарности:* благодарю – моего коллегу Сергея Александровича Р., предоставившего свои магнитофонные кассеты с программами для ДЗ-28 и фотографии плат согласующих устройств;
- авторов сайтов [www.leningrad.su/museum/](http://www.leningrad.su/museum/), [www.pc-history.com](http://www.pc-history.com) и [retropc.org](http://retropc.org), обеспечивших свободный доступ к фото ДЗ-28 и к документации по ДЗ-28; (на стр. 53-55 в приложении показаны в уменьшенном виде фото с копирайтами авторов сайтов, эти фото после ретуши и ещё некоторых преобразований были применены в имитаторе ДЗ-28).

*Системные требования:* прога запускается под управлением пока только 32-разрядных (старых) операционных систем Windows, таких как ME, XP, Vista, Win 7: эти варианты мной опробованы, другие – нет. ВIDEOSИСТЕМА должна поддерживать простейшие функции 2D-графики из OpenGL.

В компьютере должен быть вентилятор. Автор проги не программист, поэтому прога несовершенна, и один из её недостатков – возможность заметной нагрузки процессора (главным образом, при имитации быстрого вывода потока символов на экран дисплея в режиме работы ДЗ-28 по программе). Правда, в обновлённой версии проги эта проблема в значительной мере устранена, а при имитации режима останова ДЗ-28 она отсутствует совсем; пояснения см. в разделе 7. Однако, поскольку разные программы в ДЗ-28 ведут себя по-разному, и свойства компьютеров могут сильно различаться, а работа проги проверялась лишь на нескольких ПК, то для перестраховки вот рекомендация на всякий случай: если вы по звуку вентилятора или по другим признакам заметите, что возможен перегрев, то лучше закройте прогу; (как говорится, «от греха подальше»).

*Запуск:* достаточно распаковать архив в любое место и кликнуть двойным кликом файл `imitator_1_0.exe`. Прога не требует установки, не просится в интернет (он ей не нужен) и ничего сама не пишет в реестр; сама Windows, видимо, как-то отмечает в реестре запуск проги, поскольку она запоминает, из какой папки открывались файлы.

В папке с прогой imitator\_1\_0.exe должна оказаться папка [bmp] с файлами рисунков для проги. Их нельзя переименовывать или удалять, и нельзя переименовывать саму папку [bmp]; в противном случае прога не запустится. А с папкой [txt] можно поступать как угодно; в ней хранятся текстовые файлы, имитирующие магнитофонную запись с программами для ДЗ-28.

Прога имитирует работу старинного вычислительного комплекса "машина ДЗ-28 с 32-килобайтным ОЗУ, с терминалом 15ИЭ-00-013, подключенным к машине через согласующее устройство ПЕЛ2.240.001, и с печатающим устройством типа ТПУ 15ВВП80-002". Такой комплекс по своим возможностям близок к первым вариантам «Радио-86РК», но появился примерно на пять лет раньше, где-то на рубеже 1980-х годов. Информация выводится в комплексе на алфавитно-цифровой дисплей (или на алфавитно-цифровое ТПУ и на цифровое табло машины), а «графики» и «псевдографики» нет. Операционки тоже нет.

Для ясности договоримся о терминах:

ПК – персональный компьютер, на котором запускается прога имитатора ДЗ-28.  
ПК-клава – клавиатура персонального компьютера. (Ещё и мышь пригодится).  
Монитор – экран монитора персонального компьютера. Это всё, что надо проге.

Машина – изображение ДЗ-28 на мониторе или воображаемая ДЗ-28 в имитаторе.  
Пульт (и табло) – изображение пульта управления (и табло) ДЗ-28 в имитаторе.  
Дисплей – окно проги в режиме имитации экрана терминала 15ИЭ-00-013.  
Клава – воображаемая или изображённая клавиатура терминала 15ИЭ-00-013.  
КД – воображаемый контроллер дисплея, согласующий дисплей с машиной.  
ТПУ – воображаемое термопечатающее устройство, подключенное к ДЗ-28.  
НМЛ – изображённый или воображаемый кассетный магнитофон машины (накопитель на магнитной ленте).

Нажать клавишу – означает «нажать и отпустить». О необходимости удерживать клавишу нажатой будет сказано явно; а иначе – обязательно надо отпускать.

## **1. Быстрое ознакомление с имитатором**

1. Двойным кликом запустите imitator\_1\_0.exe. На мониторе откроется окно проги с изображением машины. Изображение машины и клави строится, к сожалению, не целиком, а из 6 текстур – рисунков 256x256 px; поэтому на разных ПК возможны разные «краевые» дефекты изображения в виде тонких вертикальных и горизонтальных полос, разделяющих текстуры. Если они очень заметны, то для их устранения попробуйте потянуть мышью вертикальные или горизонтальные края окна, меняя его размер. Изменять размер окна допускается.

Открытая крышка НМЛ и отсутствие имени txt-файла в верхней строке окна, то есть в строке с именем проги, служат признаками того, что в НМЛ не вставлена кассета с магнитофонной записью какой-либо программы.

2. В меню **File** выберите **Open txt-file as Tape**. В открывшемся стандартном окне обзора файлов найдите нашу папку [txt] и выберите из неё для открытия файл Basic\_D3-28\_v3A\_KP-157107\_\_N-11343.txt. Тем самым вы «поставили в машину кассету» с Бейсиком-157107. Признаки этого: НМЛ теперь изображается с закрытой крышкой, а сверху окна проги присутствует имя открытого txt-файла, имитирующего магнитную ленту (tape).

3. Нажмите на ПК-клаве клавишу <C> – это имитация кнопки C на пульте машины, т. е. команда «Сброс». Затем нажмите <L> – это как бы кнопка СЛ на пульте, команда «читать с ленты». При этом мигнёт окошко в крышке НМЛ; здесь это признак того, что файл «считался с ленты в ОЗУ машины Д3-28».

4. Нажмите <K> – это имитация кнопки КП на пульте, т. е. команда «вычислить контрольную сумму» файла, находящегося в ОЗУ машины. В нижнем табло машины (X-табло) будут видны цифры 157107, точка и нули.

5. Нажмите <C>. И затем <S> – это имитация кнопки S на пульте, т. е. команда Start (вернее, GO) – команда запуска программы из ОЗУ машины. Машина перейдёт из режима «останова с индикацией» в режим «работы по программе», в нашем примере – начнёт работать Бейсик-157107.

6. В меню **View** выберите **Display**. Поначалу прога имитирует дисплей, отключенный от машины, так что на дисплее нет текста (кроме служебной строки с цифрами), машина ожидает сигналов от терминала или от своего пульта. Нажмите на ПК-клаве клавишу <F5> – это имитация включения ДУП и ЛИН на клаве терминала. Бейсик выведет на дисплей начальный диалог.

7. Ответьте на вопросы в диалоге, причём в ответ на СНИМИТЕ КАСSETУ! и НОМЕРА ВНЕШНИХ ПОДПРОГРАММ? нажимайте <Enter> на ПК-клаве. Вообще, для ввода цифр можно нажимать на ПК-клаве обычные клавиши с цифрами или клавиши дополнительной цифровой клавиатуры (Numpad); точка на Numpad при вводе дробных частей десятичных чисел тоже будет работать. Завершайте ввод ПК-клавишей <Enter> – она имитирует клавишу ПС (перевод строки) на клаве терминала, или ПК-клавишей <Backspace> – она имитирует клавишу ВК (возврат каретки). Вместо ввода числа «ноль» во многих больших программах для Д3-28, в том числе в Бейсике, допускается просто нажимать ПС.

8. После того, как Бейсик выведет на дисплей слово ГОТОВ и двоеточие, нажмите (и отпустите) на ПК-клаве клавишу <Ctrl> – тем самым клавиша терминала переводится из режима РУС в режим ЛАТ; это необходимо для правильного ввода команд Бейсика. Когда потребуется режим РУС, надо будет снова нажать <Ctrl>.

Если хотите посмотреть текущее состояние клавиши, нажмите <F1> (а для возврата – ещё раз <F1>). Это работает, когда в меню **View** выбран **Display**; на мониторе изобразится клавиша терминала (а также знак «Выкл/Вкл ТПУ»), и можно будет посмотреть, как меняется состояние при нажатиях следующих ПК-клавиш:

- <Ctrl> – переключает РУС / ЛАТ,
- <Shift> – переключает ВР / НР («верхний регистр» / «нижний регистр»),
- <F5> – включает / выключает ДУП-ЛИН,
- <4> – вызывает окошко с информацией о «бите b4» контроллера дисплея,
- <D> – вызывает диалог о включении замедления; см. раздел 7,
- <F> – вызывает диалог об индикации оценки FPS («кадров в секунду»),
- <V> – показывает «версию имитатора» – дату компиляции exe-файла.

О включении ТПУ см. раздел 4. Прочие клавиши не следует нажимать в режиме просмотра клавиши. Вообще, лучше никогда не нажимать клавиши «от балды».

9. Выберите **File > Close txt-file as Tape** – так снимается кассета с НМЛ. Затем: **Open txt-file as Tape**, и откройте файл game\_1WAR\_\_bas.txt; тем самым в НМЛ ставится кассета с игровой тест-программой 1WAR на языке Бейсик.

10. Наберите команду LOAD и нажмите ПС, т. е. <Enter> на ПК-клаве. Успешное выполнение команды обозначится двоеточием в новой строке дисплея. При наборе команд и любых слов смотрите только на латинские буквы на клавишах ПК-клавы. Ведь имитатор с ПК-клавой не может воспроизвести расположение букв с клавиш старинного терминала, да ещё и множество кнопок на пульте машины, поэтому мы вынуждены принять некие правила соответствия; о таких правилах идёт речь в разделе 2: см. стр. 8-10.

11. Если хотите посмотреть, есть ли имя у загруженного Бейсиком файла, наберите PRINTOPEN и нажмите ПС; имя выведется вслед за буквой P. Вообще, в Бейсике любое действие подтверждайте нажатием ПС, т. е. <Enter> на ПК-клаве.

12. Для запуска программы подайте команду RUN. Описание игры 1WAR приводилось в другом pdf; напомним лишь, что для управления в этой игре назначены цифровые клавиши (их удобно нажимать на Numpad):

- <1>, <2>, <3> – выбор горизонтальной составляющей скорости точки;
- <7>, <8> – налево, направо; <9>, <6> – вверх, вниз; <0> – выход из игры.

13. Любую бейсик-программу можно остановить клавишей AP1 на клавиатуре терминала; на ПК-клавиатуре в имитаторе AP1 это <F3>. Но почему-то срабатывает это не всегда с первого раза; наверное, это один из недостатков имитатора. Другой способ: «подача из терминала команды AP1 с помощью нажатия клавиши СУ вместе с клавишей Р» (подробности о командах см. в документации терминала).

В имитаторе «прижатие/отпускание клавиши СУ» имитируется нажатиями (без удержания) клавиши <CapsLock> на ПК-клавиатуре. Отображается это только светодиодом самой ПК-клавиатуры, так что надо быть внимательным: если один раз нажали <CapsLock>, то не забудьте потом нажать <CapsLock> ещё раз, чтобы вернуть обе клавиатуры в исходное состояние.

14. После того, как бейсик-программа остановлена (или до того, как запущена), можно посмотреть на дисплее её листинг, набрав команду LIST. После вывода «одного экрана» Бейсик приостанавливает вывод; для продолжения нажимайте ПС. Чтобы вывести лишь несколько строк, надо указать первый и последний номер строк для вывода, например LIST 200,270 и затем нажать ПС. При этом, как и в начальном диалоге, в большей или меньшей степени (это зависит от быстродействия ПК) может проявиться ещё один недостаток имитатора – низкая скорость вывода символов на дисплей.

15. Командой CLEARP 1,7999 Бейсик очищает память машины от текущей бейсик-программы, подготавливаясь к вводу с НМЛ или к набору с клавиатуры новой бейсик-программы. Очистив таким образом память, наберём и запустим командой RUN простейшую программку (подробнее о наборе символов см. на стр. 8-10):

```
1 PRINT 'HELLO WORLD'
```

Если повторить опыт, заменив ЛАТ-буквы РУС-буквами, то при исполнении такой программки или при её листинге Бейсик-157107, увы, не выведет на дисплей русских букв:

```
:1 PRINT 'HELLO WORLD'
:RUN
HELLO WORLD
ostanow u stroke 1
:
:1 PRINT 'ПРИВЕТ РЕБЯТА'
:RUN
Privet rebata
ostanow u stroke 1
:
```

Видно, что здесь имеется проблема с выводом РУС-букв. О её решении см. раздел 5, стр. 21-22 про «бит b4». Есть и частичное решение, пригодное в тех случаях, когда требуется выводить одни только русские буквы, без латинских: после набора RUN, но до нажатия ПС, можно перевести дисплей в режим РУС, нажав на ПК-клаве клавишу <Ctrl>. Для примера проведите опыт: наберите латинскими буквами программку: 1 PRINT 'vopa'

в новой строке наберите RUN, затем нажмите <Ctrl> и только после этого <Enter>.

16. Клавиша <Escape> на ПК-клаве вызывает «ресет» имитатора; это всё равно что закрыть прогу имитатора и открыть её заново (за исключением того, что при ресете запоминаются текущий «бит b4» и значение параметра замедления, вывод FPS). Такое действие бывает более удобным для быстрой смены или повторного запуска программ ДЗ-28, чем закрытие txt-файла с программой, вновь открытие txt-файла и настройка режима дисплея. Ресет имитатора удобен ещё и тогда, когда что-то пошло не так, или «дэтришка зависла».

## 2. Управление дисплеем

<F1> – эта ПК-клавиша вызывает (при **View > Display**) изображение клавиатуры терминала; повторное нажатие <F1> – возврат к экрану дисплея.

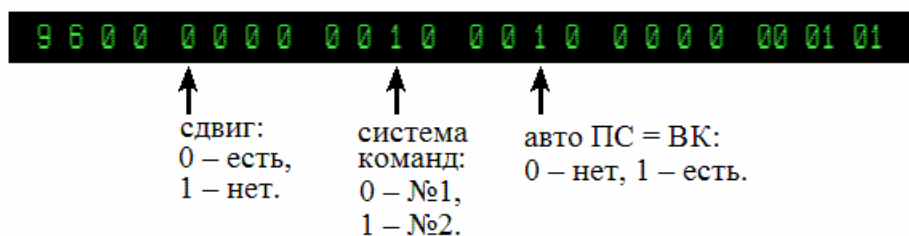
<F2> – нажатие этой клавиши имитирует нажатие AP2 на клаве терминала.

<F3> – нажатие этой клавиши имитирует нажатие AP1 на клаве терминала.

<F4> – нажатие этой клавиши имитирует нажатие C1 на клаве терминала.

<F5> – включает / выключает ДУП-ЛИН. (Текущее состояние см. нажатием <F1>).

<F8> – включает / выключает настройку служебной строки, т. е. имитирует СДВ. Прога пока ещё не умеет имитировать работу дисплея точно. В частности, в служебной строке доступно изменение только трёх параметров:



ПК-клавиша <F9> сбрасывает их всех в 0, установка параметра в 1 или 0 производится ПК-клавишей «стрелка вниз», маркер перемещается вдоль служебной строки ПК-клавишами со стрелками «вправо» и «влево».

<F9> – ОЧС экрана (при не СДВ), но такая «очистка» у нас не сбрасывает дисплей в начальное состояние после включения питания, а устанавливает показанную выше настройку служебной строки, и задаёт клавише терминала режим РУС (русские буквы) и ВР (верхний регистр, т. е. большие буквы). Так я сделал для того, чтобы меньше нажимать всяких клавиш при запуске Бейсика-157107 вслед за запуском имитатора. То есть считается, что работа с Бейсиком-157107 это основной начальный режим, а работа с другими программами и режимами – занятие «для знатоков», которые сумеют сами выбрать нужные им параметры дисплея и клавиш.

При **View > Display** (причём, не СДВ) клавиши со стрелками на ПК-клавиатуре имитируют аналогичные клавиши со стрелками клавиатуры терминала. Остальные «стрелочные» коды клавиш, специфичные для терминала 15ИЭ-00-013, можно посылать в машину путем имитации нажатия СУ вместе с другими клавишами.

<CapsLock> – при каждом нажатии имитирует прижатие / отпускание СУ.  
Нажав после <CapsLock> клавишу < I >, имитируем нажатие ГТ.

<Tab> – имитирует нажатие ТАБ; но не полностью: код ТАБ в машину поступает, а дисплей ни в одной из двух систем команд ничего не обрабатывает.

<Delete> – имитирует нажатие ЗБ, «забой».

<Backspace> – имитирует нажатие ВК, «возврат каретки».

<Enter> – имитирует нажатие ПС, «перевод строки».

Режим «не РЕД» (когда на дисплее должны мигать символы команд) у нас, к сожалению, не имитируется. В имитаторе пока нет такого ОЗУ дисплея, которое хранило бы два «экрана» и могло бы передавать в машину блоки кодов. Поэтому нет возможности редактировать целиком страницы текста и вести из терминала блочную передачу: режим ПРД в этом имитаторе не работает.

При наборе текстов важны клавиши ВР / НР – они влияют на выбор шрифта, но кодов в машину не посылают, и клавиши РУС / ЛАТ – они переключают русские и латинские буквы и при этом посылают свои коды в машину. Имитация:

<Ctrl> – переключает РУС / ЛАТ,

<Shift> – переключает ВР / НР, то есть «верхний регистр» / «нижний регистр».

В имитаторе при включённом ВР, т. е. при выключенном НР (индикатор НР при этом погашен) на дисплее набираются цифры и большие буквы.



При выключенном ВР, т. е. при включённом НР (индикатор НР при этом горит) набираются, как правило, маленькие буквы и следующие символы в режиме ЛАТ:

```

~ ! @ # $ % ^ & * ( ) _ + |
{ }
: "
< > ?

```

Нажатие тех же ПК-клавиш в режиме НР РУС даёт:

```

Ч ! @ # $ % Ч & * ( ) Ъ + Э
Ш Щ
: "
< > ?

```

Нажатие тех же ПК-клавиш в режиме ВР ЛАТ даёт:

```

` 1 2 3 4 5 6 7 8 9 0 - = \
[ ]
; '
, . /

```

Нажатие тех же ПК-клавиш в режиме ВР РУС даёт:

```

Ю 1 2 3 4 5 6 7 8 9 0 - = Э
Ш Щ
; '
, . /

```

Для набора русских букв (разумеется, в режиме РУС) не надо смотреть расположение русских букв на ПК-клаве; надо смотреть расположение латинских букв на ПК-клаве и нажимать латинские буквы по правилу соответствия:

$A_{\text{рус}} - A_{\text{лат}}$ ,  $B_{\text{рус}} - B_{\text{лат}}$ ,  $V_{\text{рус}} - W_{\text{лат}}$ ,  $G_{\text{рус}} - G_{\text{лат}}$ ,  $D_{\text{рус}} - D_{\text{лат}}$ , и т. д.

При этом можно пользоваться «хелпом» – изображением клавиш терминала, которое включается и выключается ПК-клавишей <F1> как раз в режиме **View > Display**. Каждая пара соответствующих друг другу русских и латинских букв изображена на одной и той же клавише терминала.

Исключение в имитаторе составляют буквы Ч и Ю (см. картинки выше), потому что на клаве терминала буква Ч попала на клавишу со шляпкой ^, буква Ю

попала на клавишу с собакой @, тогда как на ПК-клаве шляпка ^ и собака @ находятся на клавишах с цифрами.

Цифры и десятичную точку можно набирать на дополнительной цифровой ПК-клаве (Numpad), если она имеется; а можно и на основной ПК-клаве набирать.

Потренироваться набирать буквы и символы, а также управлять дисплеем при разных параметрах служебной строки можно, когда он отключен от «линии» связи с машиной, то есть – «в автономном режиме». В проге отключение или подключение к линии имитируется ПК-клавишей <F5>. Если машина работает в программном режиме, то отключение дисплея клавишей <F5> проявляется просто как пауза в работе программы; так можно вызывать паузу специально, когда захочется. Чтобы войти в автономный режим с возможностью печатать что-либо на экране, надо остановить машину (кнопками С или Ш на её пульте, см. раздел 6) или с самого начала не запускать на ней программу, т. е. нужно, чтобы машина была в режиме «останова с индикацией». Такое состояние реализуется, в частности, сразу после запуска имитатора.

Коды клавиш, посылаемые в имитаторе терминалом в машину, – 7-битные, с равным нулю битом «паритета» (чётности). Разбор имеющихся программ для ДЗ-28 показал, что в них бит паритета у принятого кода, вроде бы, всегда сбрасывается в ноль, поэтому я не стал пока формировать бит паритета, чтобы не усложнять и без того громоздкую процедуру анализа клавишных кодов.

Посмотреть коды клавиш, посылаемые в имитаторе из терминала в машину, можно с помощью программки display\_codes\_KP-1689\_N-106.txt. (Запуск: С, S).

### 3. Работа с НМЛ

Сразу после запуска имитатора (а также после нажатия кнопок С или Ш на пульте машины, см. раздел 6) машина находится в состоянии «останова с индикацией». При выборе меню **View > Machine** мы имеем на мониторе изображение машины с НМЛ (НМЛ – накопитель на магнитной ленте, кассетный магнитофон). При этом доступны следующие действия с НМЛ:

**File > Open txt-file as Tape** – открытие txt-файла, имитирующее установку кассеты в НМЛ. Признаком установленной кассеты является закрытая крышка НМЛ и наличие имени txt-файла в верхней строке окна имитатора. Либо:

**File > Close txt-file as Tape** – так мы вынимаем кассету из НМЛ.

Все действия с меню доступны и в режиме работы машины по программе.

ПК-клавиша со стрелкой «налево» действует как кнопка «перемотка назад», лента перематывается в самое начало (если машина в состоянии останова).

ПК-клавиша со стрелкой «направо» аналогичным образом имитирует «перемотку вперёд», но перематывает воображаемую ленту не до конца, а на одну программную единицу, заканчивающуюся командой END (с кодом 0512). Другими словами, в имитаторе перемотка вперёд действует как «пропустить один файл». Если в txt-файле есть всего одна программа, то, конечно, получается то же самое, что и перемотка до конца ленты.

Перемотка отображается в имитаторе коротким миганием окна в крышке НМЛ (ну, надо же было хоть как-то отобразить перемотку :-). Если кассета в НМЛ отсутствует, то перемотка не включается, и нажатие ПК-клавиш со стрелками никак не отображается. Если кассета в НМЛ есть, и с неё уже всё считано или перемотано вперёд до конца, то попытка ещё раз перемотать вперёд отображается в имитаторе зажиганием индикатора ОМ – ошибка магнитофона; это верхняя лампочка рядом с Y-табло машины. Назад можно перематывать сколько угодно.

Если машина находится в программном режиме, то перемотки (и вообще, большинство действий с пульта машины) в имитаторе не работают, блокируются; в реальности это не так, но в имитаторе пока так, чтобы не усложнять имитатор прерыванием имитации программного режима машины без серьёзного повода.

<L> – эта ПК-клавиша имитирует кнопку СЛ на пульте машины: если машина не в режиме «работа по программе», то происходит считывание программы с ленты (из открытого в имитаторе txt-файла); это действие также отображается миганием окна в крышке НМЛ. Если лента кончилась ещё «до того», то загорится индикатор ОМ; в этом случае чтобы считать программу, надо нажать <C> (индикатор ошибки погаснет), сделать перемотку назад, и только затем подать команду СЛ. Разумеется, если искомая программа не первая на ленте, то после перемотки назад надо нужное число раз сделать перемотку вперёд.

<K> – эта ПК-клавиша имитирует кнопку КП на пульте машины: команда вычисления контрольной суммы байтов в ОЗУ машины до байта с кодом команды END. Если команды END в ОЗУ машины нет, то загорится индикатор ОП – ошибка программы, и будет мигать X-табло. При успешном вычислении контрольной суммы результат индицируется на X-табло машины (т. е. на нижнем зелёном табло; верхнее зелёное табло называется Y-табло, см. раздел 6).

После выполнения команды КП программный счетчик машины указывает не на начало ОЗУ, а на ячейку ОЗУ с кодом команды END. Поэтому перед запуском считанной программы или перед записью программы из ОЗУ на ленту (см. ниже) надо подать команду Сброс – нажать <C> на ПК-клаве; тогда программный счетчик машины будет указывать на начало ОЗУ.

<Z> – эта ПК-клавиша имитирует кнопку ЗЛ на пульте машины: команда записи на ленту содержимого ОЗУ вплоть до команды END. Если такой команды в ОЗУ машины нет или не поставлена кассета в НМЛ, то загорится индикатор ОМ (так сделано в имитаторе, а как было в реальности – уже не помню).

Информация о разновидностях команд записи и считывания файлов машиной есть в документации по ДЗ-28; здесь в подробности не вдаюсь.

Надо быть внимательным, чтобы не испортить командой ЗЛ какую-нибудь уникальную запись. Желательно для записи заранее заготовить «чистую ленту»: создать «Блокнотом» пустой txt-файл с желаемым именем и поместить его в нашу папку [txt], чтобы перед записью можно было бы командой меню **File > Open txt-file as Tape** открыть именно его.

Аналогично можно создать «ленту» с записями многих программ или их дублей, попеременно считывая программы из одних txt-файлов, записывая их друг за другом в другой txt-файл, не забывая при этом должным образом «менять кассеты» в НМЛ и делать нужное количество перемоток вперёд (правда, в этом не всегда есть смысл, поскольку можно выполнять копирование txt-файлов и редактировать их содержимое более удобными программами в Windows :-)

После открытия файла командой **File > Open txt-file as Tape** имитатор перед его чтением проверяет наличие расширения txt в имени файла и смотрит начало файла, чтобы убедиться, что в нём записаны числа в соответствующем формате. Если в файле что-то не то, то при попытке чтения загорится индикатор ОМ, прога не станет читать такой файл, и вынет прочь кассету. При записи же прога проверяет только расширение (поскольку файл для записи может быть пустым), и поэтому легко обмануть прогу, специально заменив расширение не текстового файла на расширение txt, так что имитатор произведёт-таки в него запись. Это к тому, что подобным путём можно командой ЗЛ нарочно испортить любой файл; но я снимаю с себя ответственность за подобную порчу файлов, поскольку объяснил, чего и как не надо делать.

В сложных программах предусмотрены специальные способы работы с НМЛ; ими и надо пользоваться, когда машина находится в режиме «работа по программе». Например, в Бейсике работают команды REWIND – перемотка назад, SKIP N F – пропустить N файлов, SAVE – запись на ленту, LOAD – загрузка с ленты. В Фортране и в Быстре также есть команды работы с НМЛ. О формировании «библиотек записей» см. в разделе 5 (и в документации к программам Бейсик и Фортран).

## 4. Работа с ТПУ

<Home> – первым нажатием этой ПК-клавиши включается возможность имитации ТПУ – термопечатающего устройства; это «включение ТПУ». Прога создаёт txt-файл с уникальным именем типа «цифры\_\_цифры», где первая группа цифр это *час-минута-сек.* создания файла, а вторая группа цифр представляет дату создания файла: *день-месяц-год.* Сразу после создания файла в нём есть только одна строка: с текстом `tru_on`. Такой файл – как бы рулон бумаги, заряженной в ТПУ; в него пойдёт вывод букв, цифр и прочих символов, предусмотренный в программах.

Windows сама выбирает место для этого файла: если в имитаторе уже открывались txt-файлы из какой-то папки, то файл помещается в эту папку; если же мы включили ТПУ сразу после запуска имитатора, то файл создаётся в той папке, где находится exe-файл с имитатором.

Дальнейшие нажатия <Home> имитируют прокрутку бумаги в ТПУ: каждое нажатие добавляет в txt-файл пустую строку. Таким путём можно отделять друг от друга блоки текста, выведенные в один и тот же txt-файл по ходу работы с ТПУ.

<End> – первым нажатием этой клавиши имитируется, если ТПУ уже было включено, отрывание бумаги: созданный при включении ТПУ txt-файл закрывается, и с этого момента его можно куда угодно переместить и / или переименовать. При этом создаётся новый txt-файл, пустой.

Если сразу же нажать <End> второй раз, то ТПУ выключается; в пустой файл печатается текст `tru_off`, файл этот закрывается, а новый файл не создаётся. Такой файл-обрывок в дальнейшем надо не забыть удалить, он никому не нужен.

Если же после первого нажатия <End> выполнить какие-либо другие действия, то пустой txt-файл играет роль бумаги, оставшейся в ТПУ; в него пойдёт вывод, а после нажатия <End> он тоже будет «оторван» (и при немедленном повторном нажатии <End> «ТПУ выключится»; чтобы его снова включить, надо будет опять нажать <Home>. И т.д.).

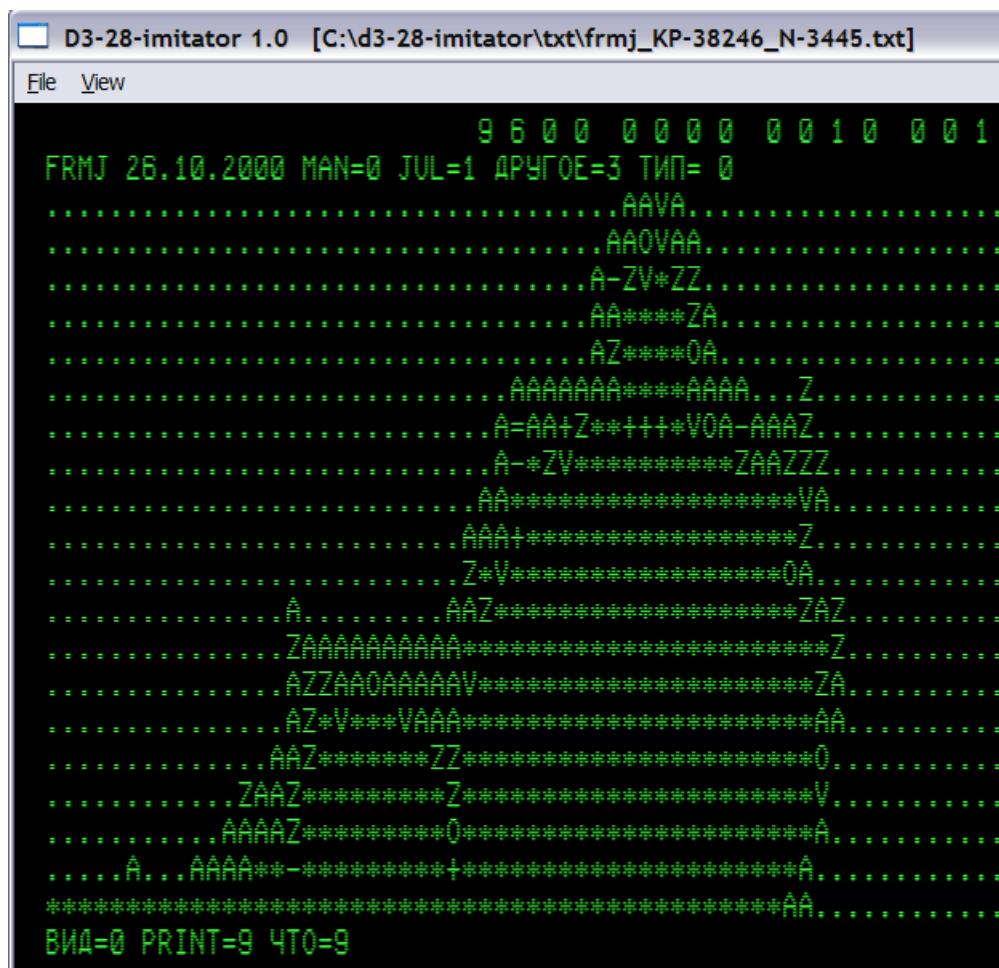
В самодельных программах печать надо программировать по шагам (пример: см. `raspechatchik_D3-28__KP-935__N-67.txt`) В готовых программах, таких как Бейсик, Фортран и Выстра, операции вывода на печать уже встроены и описаны в документации к этим программам.

Например, в Бейсике листинг программы пользователя выводится на печать командой `LIST#1`. Имитатор выполняет такую команду, что называется, в один

присест. Целиком листинг делается и в Выстре. В Фортране же листинг печатается частями; после каждой части надо нажимать клавишу <Пробел>.

Реальное ТПУ не умело печатать маленькие буквы, печатало только большие. В имитаторе тоже так сделано; при этом коды больших латинских и русских букв, а также цифр и прочих символов для вывода в txt-файл берутся из кодировочной таблицы “Windows Cyrillic (1251)”.

Имитатор не нарушает форматирования текста, если оно предусмотрено в программе для ДЗ-28. Однако есть нюанс, связанный с выбором шрифта в той программе, которая будет применяться для просмотра txt-файла, имитирующего распечатку на ТПУ. Проиллюстрирую это примером. Вот скриншот экрана дисплея с расчётом «фрактала» программой FRMJ в нашем имитаторе ДЗ-28:



```
9 6 0 0 0 0 0 0 1 0 0 1
FRMJ 26.10.2000 MAN=0 JUL=1 ДРУГОЕ=3 ТИП= 0
.....AAVA.....
.....AAOVAA.....
.....A-ZV*ZZ.....
.....AA****ZA.....
.....AZ****0A.....
.....AAAAAA****AAAA...Z.....
.....A=AA+Z**++*VOA-AAAZ.....
.....A-*ZV*****ZAAZZ.....
.....AA*****VA.....
.....AAA+*****Z.....
.....Z*V*****0A.....
.....A.....AAZ*****ZAZ.....
.....ZAAAAAAAA*****Z.....
.....AZZAA0AAAV*****ZA.....
.....AZ*V**VAAA*****AA.....
.....AAZ*****ZZ*****0.....
.....ZAAZ*****Z*****V.....
.....AAAZ*****0*****A.....
.....A...AAAA**-*+*****A.....
*****AA.....
ВИД=0 PRINT=9 ЧТО=9
```

Здесь видно, что в ответ на вопрос «ЧТО=» (имеется ввиду «что делать дальше?») был выбран вариант «PRINT». Имитатор с включенным ТПУ создал txt-файл с именем 2-19-41\_\_16-3-2017.txt. Ниже изображено содержимое этого txt-файла в том виде, как его показывает программа Lister в составе файлового менеджера типа Total Commander (Lister запускается нажатием <F3> или «Просмотр», в его меню «Вид» необходимо выбрать «кодировку Windows»):

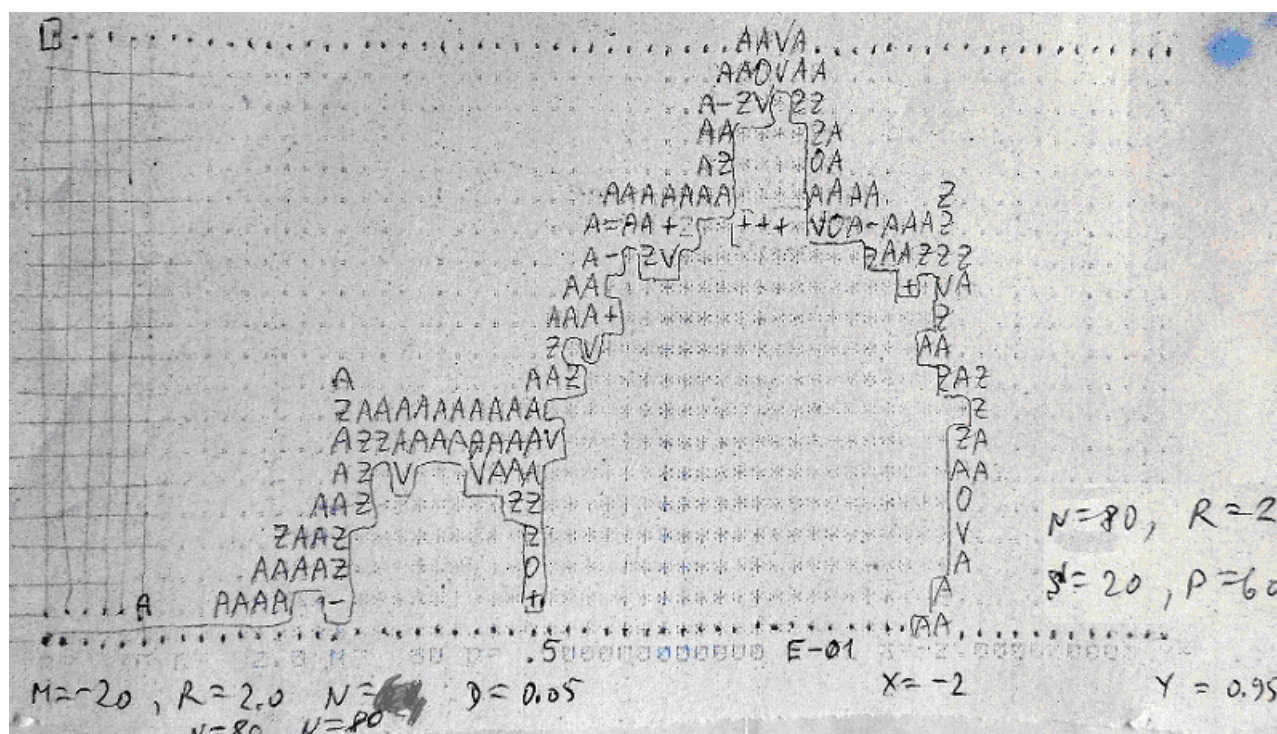
```

Lister - [C:\d3-28-imitator\txt\2-19-41__16-3-2017.txt]
Файл Правка Вид Справка

____tpu_on____
^
.....AAVA.....
.....AAOVAA.....
.....A-ZU*ZZ.....
.....AA*xxxZA.....
.....AZ*xxx0A.....
.....AAAAAA*xxxAAAA...Z.....
.....A=AA+Z*xx+++*UOA-AAAZ.....
.....A-*ZU*xxxxxxxxxZAAZZZ.....
.....AA*xxxxxxxxxxxxxxxxxVA.....
.....AAA+xxxxxxxxxxxxxxxxxZ.....
.....Z*U*xxxxxxxxxxxxxxxxx0A.....
.....A.....AAZ*xxxxxxxxxxxxxxxxxZAZ.....
.....ZAAAAAAAAAA*xxxxxxxxxxxxxxxxxZ.....
.....AZZAAOAAAAAV*xxxxxxxxxxxxxxxxxZA.....
.....AZ*U*xx*VAAA*xxxxxxxxxxxxxxxxxAA.....
.....AAZ*xxxxxxZZ*xxxxxxxxxxxxxxxxx0.....
.....ZAAZ*xxxxxxxZ*xxxxxxxxxxxxxxxxxU.....
.....AAAAZ*xxxxxxx0*xxxxxxxxxxxxxxxxxA.....
.....A...AAAA*-xxxxxxx+xxxxxxxxxxxxxxxxxA.....
*****AA.....
M= -20 R= 2.0 N= 80 D= .500000000000 E-01 X=-2.000000000 Y= .950000000

```

Сравним эту картину со старинной распечаткой на реальном ТПУ:





Старинная распечатка плохо сохранилась, обесцветилась, так что пришлось на ней бледные буквы подправить ручкой, но главное видно: имитатор распечатал в txt-файле всё правильно, а Lister всё правильно показал.

Но вот как выглядит содержимое того же txt-файла, если его открыть в «Блокноте»:

```

2-19-41__16-3-2017.txt - Notepad
File Edit Format View Help

____tpu_on____

^

.....AAVA.....
.....AAOVAA.....
.....A-ZV*ZZ.....
.....AA****ZA.....
.....AZ****OA.....
.....AAAAAAA****AAAA...Z.....
.....A=AA+Z**+++*VOA-AAAZ.....
.....A-*ZV*****ZAAZZZ.....
.....AA*****VA.....
.....AAA+*****Z.....
.....Z*V*****OA.....
.....A.....AAZ*****ZAZ.....
.....ZAAAAAAAAA*****Z.....
.....AZZAAOAAAAAV*****ZA.....
.....AZ*V***VAAA*****AA.....
.....AAZ*****ZZ*****O.....
.....ZAAZ*****Z*****V.....
.....AAAAZ*****O*****A.....
.....A...AAAA**-*****+*****A.....
*****AA.....

M=-20 R= 2.0 N= 80 D= .500000000000 E-01 X=-2.000000000 Y= .950000000

```

Видно, что картина искажена, поскольку ширина «знакоместа» у разных символов здесь разная, пробел вообще очень узкий.

Значит, если требуется воспроизвести формат, то следует для просмотра txt-файла, имитирующего ТПУ-распечатку, выбирать шрифт с постоянной шириной знакоместа (как это было в реальном ТПУ). Например, шрифт Courier.



## 5. Запуск Бейсика, Фортрана, Выстры. «Бит b4» КД

В этом разделе перечисляются некоторые нюансы, важные для безошибочного старта некоторых конкретных программ.

### *Бейсик-157107*

Сразу после открытия проги имитатора параметр «бит b4» сброшен в ноль (на что он влияет – рассмотрим ниже, пока не заботьтесь об этом). И установлена комфортная для запуска Бейсика-157107 настройка дисплея:

служебная строка: 9 6 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 ...

(т. е. система команд №2 и режим ВК = ПС, так что каждая из команд ВК или ПС производит две операции: «возврат каретки» и «перевод строки»),

режим клавиатуры терминала: ВР РУС.

Если «бит b4» мы не меняли, то команда ОЧС (очистка экрана дисплея ПК-клавишей <F9>) или «ресет» имитатора (ПК-клавишей <Escape>) возвращает дисплей к этой же настройке. Действия для загрузки и запуска Бейсика-157107:

**File > Open txt-file as Tape ,**

в папке [txt] открываем Basic\_D3-28\_v3A\_\_KP-157107\_\_N-11343.txt.

Запуск производится с пульта машины; так что, если мы смотрели экран дисплея, то необходимо перейти к изображению машины: **View > Machine.** ПК-клава теперь управляет машиной, а не дисплеем. На ПК-клаве нажимаем:

<C> – сброс счётчика команд (правда, поначалу он и так уже сброшен).

<L> – команда СЛ: загрузка программы в ДЗ-28 с магнитной ленты.

<K> – команда КП: проверка контрольной суммы. (Имитатор пока ещё не давал сбоев, а в реальности-то проверяли даже номер шага с кодом команды END (0512); в имитаторе для этого надо нажать <V> и смотреть на X-табло, там должно быть: 11343 05 12. Если посмотрели, то возвращаемся в режим «Работа» нажатием <R> ).

<C> – сброс.

<S> – старт; в этот момент машина переходит в «программный режим» (а до этого машина была в состоянии «остановка с индикацией»).

**View > Display** – переход от изображения машины к дисплею; при этом переходе в имитаторе автоматически меняется назначение ПК-клавиш.

<F5> – подключаем дисплей к «линии» связи с ДЗ-28, это вкл. ДУП-ЛИН, если до этого он был отключен от линии. Повторные нажатия <F5> отключают и включают ДУП-ЛИН; можно посмотреть текущее состояние, нажав <F1>; выход из просмотра – ещё раз <F1>.

Бейсик выводит на дисплей свой начальный диалог. Отвечая на СНИМИТЕ КАСSETУ! и НОМЕРА ВНЕШНИХ ПОДПРОГРАММ?, а также вместо ввода нуля, нажимаем на ПК-клаве <Enter> – это команда ПС; конечно, можно и ноль вводить, когда надо. Включение ТПУ, вроде бы, будет работать и в том случае, если в начальном диалоге мы не подтвердили наличие печатающих устройств.

<Ctrl> – перевод клави терминала в режим РУС; делаем его после того, как Бейсик выведет слово ГОТОВ и двоеточие.

Слова языка Бейсик (1970х-80х годов) широко известны и подробно описаны в справочной документации. Здесь не стану вдаваться в подробности; напомним только несколько самых важных команд:

LOAD – команда загрузки бейсик-программы пользователя с ленты. Важно не забыть перед этой командой сменить в НМЛ кассету:

**File > Close txt-file as Tape** – «вынимаем кассету» с Бейсиком,

**File > Open txt-file as Tape** – «ставим кассету» с бейсик-программами.

REWIND – перемотка ленты назад.

LIST (или LIST#0, если до этого был LIST#1) – листинг с выводом на дисплей.

LIST#1 – распечатка листинга. До этого надо включить ТПУ, т. е. нажать <Home>.

SAVE – запись бейсик-программы на ленту. Опять-таки: важно не путать кассеты!

RUN – запуск бейсик-программы пользователя.

CLEARP 1,7999 – удаление бейсик-программы пользователя из ОЗУ (обычно перед набором или перед загрузкой с ленты новой бейсик-программы).

Каждая команда в Бейсике выполняется после ПС, т. е. <Enter> на ПК-клаве.

В имитаторе работают также разновидности команд записи и чтения (подробно описанные в документации к Бейсику), такие как запись программы с именем, чтение нужной программы по имени с ленты, содержащей много программ, запись и чтение массивов, пропуск файлов на ленте.

Можно сформировать «библиотеку записей» – txt-файл, имитирующий ленту со многими файлами бейсик-программок (но, к сожалению, наш имитатор с Бейсиком не может читать бейсик-файлы из смеси записей разного формата, хотя с Фортраном может читать фортранные программы из смеси; «библиотека записей» для Бейсика у нас должна содержать только те файлы, которые сделаны Бейсиком). Это делается аналогично имитации ленты с записями программ в машинных кодах: попеременно набираем программы или считываем их командой LOAD из txt-файлов, и записываем их друг за другом командой SAVE'*имя\_программы*' в файл-«библиотеку», не забывая при этом должным образом «менять кассеты» и выполнять REWIND перед оператором SKIP N F с нужным количеством N. Если не предполагается в дальнейшем изменять содержимое библиотеки, то можно в роли последней программки записать список имеющихся в ней программ (т. е. их имён), условившись называть такую программку-список каким-то «стандартным» именем, например 'КАТАЛОГ'. Тем самым откроется возможность сначала командой LOAD'КАТАЛОГ' загрузить и узнать список имён программ на данной «ленте», а затем загрузить желаемую программу по её имени.

В самом конце «ленты» следует делать завершающую запись: SAVE END. Тогда при попытках повторного чтения последней программы без перемотки назад Бейсик не будет «зависать», а просто выведет на экран двоеточие – признак готовности к работе. Если же пытаться читать «окончившуюся ленту», в которой нет завершающей записи, то «ДЗ-28 зависнет», оставаясь при этом в программном режиме. В этом случае надо перейти к изображению пульта машины и остановить машину нажатием <C>; затем можно снова войти в Бейсик, нажав <T>, <M>.

Запущенную командой RUN программу (или слишком длинный листинг) можно остановить клавишей AP1, т. е. нажатием <F3> на ПК-клаве. При этом Бейсик продолжает работать: машина находится в «программном режиме».

Чтобы остановить машину, мы в любом случае должны перейти к управлению с её пульта, выбрав **View > Machine**, и нажать на пульте кнопку С или Ш, т. е. нажать на ПК-клаве <C> или <Enter>. Разница в том, что при нажатии С программный счетчик устанавливается в начало программы (а также происходит ресет указателя стека, сброс масок прерываний и т. п.), а при нажатии Ш счётчик после останова машины просто указывает на следующую команду из программы; её можно посмотреть в режиме «Ввод», нажав <V>.

Возобновить работу Бейсика можно попытаться нажатием <S> (сначала вернувшись из режима «Ввод» в режим «Работа» клавишей <R>), если счетчик команд указывает на упомянутую невыполненную команду. Либо – поочерёдным

нажатием двух ПК-клавиш: <Т>, <М> в режиме «Работа»; это имитация нажатия кнопок со значком «треугольник» и «М» на пульте машины. Тем самым подаётся команда «поиск метки» для «горячего старта Бейсика».

В папке [txt] примером «ленты с библиотекой бейсик-программ» служит файл bega\_\_5prg\_\_bas.txt. Вот так выглядит экран в начале работы с этим файлом:

```

                                     9 6 0 0  0 0 0 0  0 0 1 0  0 0 1 0

БЭЙСИК ДЗ-28, ВАРИАНТ 3А
СНИМИТЕ КАСЕТУ!
В КОМПЛЕКС ВХОДИТ ПЕЧАТАЮЩЕЕ УСТРОЙСТВО ( 0 - НЕТ, 1 - ДА ) ?0
НОМЕРА ВНЕШНИХ ПОДПРОГРАММ?

ГОТОВ
:LOAD 'КАТАЛОГ'
:RUN

BEGA-1
BEGA-2
SINUS
ХАОС-1
katalog

ostanow u stroke 60
:REWIND:LOAD 'ХАОС-1'
:PRINTOPEN
P ХАОС-1
```

Чтобы избежать проблем с РУС-буквами, желательно имена программ набирать только ЛАТ-буквами. Так, в приведённом примере имена всех четырёх программ, составляющих «библиотеку», образованы латинскими буквами, включая имя четвертой программы (она иллюстрирует явление «детерминированного хаоса»); только слово КАТАЛОГ при записи и чтении набиралось русскими буквами.

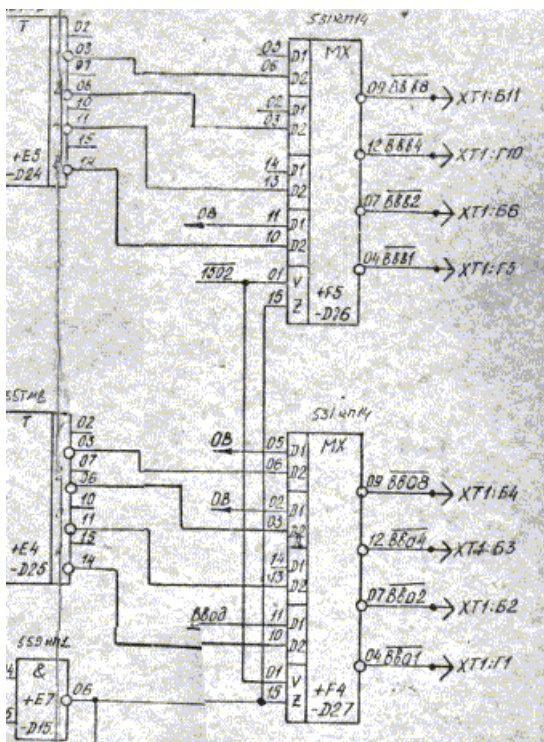
#### *«Бит b4» контроллера дисплея*

Прога имитатора позволяет экспериментировать с одним из параметров имитации контроллера дисплея – «битом b4», который, как оказалось, сильнее всего влияет на работу Бейсика-157107 (влияния же на работу других программ я не заметил). Немного поясню, о чём идёт речь.

Ввод данных в реальную машину из контроллера дисплея (КД) происходит байтами по 8-проводной шине «Ввод»; биты этой шины обозначим так:

b8, b4, b2, b1, a8, a4, a2, a1

Назначение байта на шине «Ввод» определяется байт-кодом, который машина выставляет на другой 8-проводной шине – шине «Управления» (УПР). Существенны три значения управляющего байт-кода; при УПР = 1507 контроллер дисплея выдаёт в шину «Ввод» байт 1515 – это признак того, что машина будет работать именно с КД, а не с каким-то другим периферийным устройством; при УПР = 1501 КД выдаёт в шину «Ввод» тот байт, который пришёл с клавиатуры терминала (при этом КД формирует для машины ещё и сигнал прерывания Пр8, но сейчас нам будет важно не это); при УПР = 1502 КД должен выдавать в шину «Ввод» байт, который несёт в машину информацию о состоянии дисплея и о типе дисплея – так называемое «слово состояния» КД.



И вот, именно со «словом состояния» имеется неясность. Можно заметить, что на схеме КД четыре из восьми входов мультиплексоров, которые формируют байт для шины «Ввод», висят в воздухе. Причём, как раз эти входы важны при появлении сигнала, соответствующего УПР = 1502.

Схема эта вообще нарисована плохо; из справочников по мультиплексорам выяснилось, что соответствие между номерами входов и выходов должно быть вот таким (при УПР = 1502):

Выход: b8, b4, b2, b1, a8, a4, a2, a1

Вход: 11, 14, 05, 02, 11, 14, 05, 02

(красным цветом отметил «пустышки»)

«Методом тыка», с учётом также и указанных на схеме соединений, удалось найти один из работоспособных вариантов формирования слова-состояния КД в имитаторе (при УПР = 1502):

Выход на шину «ввод» : b8, b4, b2, b1, a8, a4, a2, a1

Сигнал на входе: 0, 0, 1, 1, Пр8, 1, 0, 0

В этом варианте, как видим, В-тетрада равна  $0011_2 = 3_{10}$ ; в ней бит b4 = 0. Этот вариант по умолчанию устанавливается при запуске имитатора. Приведённое выше описание работы Бейсика-157107 относится именно к этому варианту (и в этом варианте у Бейсика-157107 возникает проблема с выводом русских букв, о которой шла речь в разделе 1, стр. 6-7).

Позже выяснилось, что работа Бейсика-157107 существенно меняется, если выбрать для В-тетрады значение  $0111_2 = 7_{10}$ ; т. е. установить бит  $b4 = 1$ . В этом случае усугубляется один из недостатков проги: уменьшается и без того невысокая скорость вывода на дисплей. Но появляются и преимущества: в этом варианте Бейсик-157107 правильно выводит на дисплей русские и латинские буквы, и позволяет менять размер шрифта при выводе на дисплей (точно так, как это указано в документации к Бейсику): оператор PRINT@1 выводит маленькие буквы, оператор PRINT@0 выводит большие буквы; пример:

```
:LIST
1 PRINT 'HELLO WORLD, ПРИВЕТ РЕБЯТА!'
2 PRINT@1 'HELLO WORLD, ПРИВЕТ РЕБЯТА!'
3 PRINT@0 'СНОВА БОЛЬШИЕ БУКВЫ'
4 PRINT 'БЕЗ СОБАКИ 1 ТАК И БУДУТ БОЛЬШИМИ'

:RUN
HELLO WORLD, ПРИВЕТ РЕБЯТА!
hello world, привет ребята!
СНОВА БОЛЬШИЕ БУКВЫ
БЕЗ СОБАКИ 1 ТАК И БУДУТ БОЛЬШИМИ
ОСТАНОВ В СТРОКЕ 4
:
```

Таким образом, приходится заключить, что вариант с  $b4 = 1$  – правильный! Вероятно, входы мультиплексоров внутри самих этих микросхем «подтянуты» к  $+E_n$ , поэтому разработчики КД посчитали возможным оставить «в воздухе» те входы, на которые должен подаваться сигнал логической единицы.

Переключение имитатора из режима « $b4 = 0$ » в режим « $b4 = 1$ » и обратно производится после нажатия ПК-клавиш <F1> и <4> в окошке с названием-вопросом «Change b4?» (в режиме **View** > **Display**, причём машина должна быть остановлена). Ответ «Да» меняет значение бита  $b4$  с нуля на единицу и обратно. После того, как вариант выбран, надо ответить «Нет»; тогда окошко исчезнет.

В режиме с « $b4 = 1$ » Бейсик в своем начальном диалоге, а также при исполнении операций LIST и PRINT, сам выполняет дополнительный перевод строки. Поэтому в настройке дисплея следует отключить «авто ПС=ВК» (см. об установке параметров служебной строки раздел 2, стр. 7):

9 6 0 0   0 0 0 0   0 0 1 0   0 0 0 0   0 0 0 0   ...

В режиме с « $b4 = 1$ » очистка дисплея (ОЧС) ПК-клавишей <F9> или ресет имитатора ПК-клавишей <Escape> автоматически устанавливает служебную строку в указанное состояние – с отключенным «авто ПС=ВК».

Наверное, следует вообще убрать из имитатора режим «b4 = 0», раз уж он «неправильный»; но я успел привыкнуть к нему, и, к тому же, на моём стареньком ПК он более быстрый, так что пока я его по-прежнему считаю основным :-)

### *Бейсик-132259*

Файл Basic\_D3-28\_v3A\_\_KP-132259\_\_N-9675.txt это «кассета» с более ранней версией Бейсика. Вроде бы, бит b4 на Бейсик-132259 не влияет, поэтому достаточно запускать его только в основном режиме имитатора, с «b4 = 0». Бейсик-132259 запускается в имитаторе так же, как и Бейсик-157107 при b4 = 0, с той же самой настройкой дисплея «по умолчанию»:

служебная строка: 9 6 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 ...  
режим клавиатуры терминала: ВР РУС.

Особенность только в том, что нажатие <Ctrl> для перевода клавиши терминала в состояние ЛАТ Бейсик-132259 воспринимает как ошибку; но это не препятствует дальнейшему набору программ ЛАТ-буквами.

В этом Бейсике LIST#1 не работает. И команды @1 и @0 в операторе PRINT не работают. Однако они и не нужны, потому что Бейсик-132259 и без них умеет выводить оператором PRINT маленькие буквы, притом различая ЛАТ и РУС:

```
D3-28-imitator 1.0 [C:\d3-28-imitator\txt\Basic_D3-28_v3A__KP-132259__N-9675.txt]
File View

9 6 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 ...
ВР РУС.

БЭЙСИК Д3-28, ВАРИАНТ 3А
СНИМИТЕ КАССЕТУ!
НОМЕРА ВНЕШНИХ ПОДПРОГРАММ?

ГОТОВ
:
ошибка 1 u stroke 0
:
:1 PRINT 'HELLO РЕБЯТА прубем world'
:
:RUN
HELLO РЕБЯТА прубем world
останов u stroke 1
:
=
```



Ещё особенность: Бейсик-132259 слегка тормозит при наборе букв, так как в нём работает периодическая самопроверка сохранности.

Интересно также, что программка, набранная и записная в Бейсике-132259, при запуске в Бейсике-157107 может правильно выводить на экран РУС-буквы даже при «b4 = 0». Пример: программы BEGA-1, SINUS и ХАОС-1 в упоминавшемся выше файле bega\_\_5prg\_\_bas.txt отредактированы и записаны Бейсиком-132259. Поэтому текст из них правильно выводится на дисплей в любом из двух Бейсиков и при любой настройке «бита b4» (только листинг может выглядеть на экране слегка странно); по этой же причине и многие другие программки, имеющиеся в папке [txt], отредактированы Бейсиком-132259. Программа же BEGA-2 редактировалась Бейсиком-157107; и каталог набирался в Бейсике-157107. Оказывается, текст из неё правильно выводится на дисплей только в Бейсике-157107: либо при «b4=1», либо (если «b4=0») при нажатии <Ctrl> после набора RUN, но до нажатия ПС. Чтобы указанный каталог прочитать с ленты по его имени в Бейсике-132259, надо его имя набирать латинскими маленькими буквами. Вся эта ситуация связана с тем, что Бейсик-157107 записывает только коды букв и цифр, а Бейсик-132259 записывает ещё и коды дисплейных команд 0015 и 0014, которые переключают РУС / ЛАТ.

### *Фортран-85*

Аналогично тому, как это делалось для загрузки Бейсика (см. выше), в свежезапущенный или освежённый ресетом имитатор «ставим кассету» Fortran-85\_p1-p2.txt – в этом файле содержатся обе части Фортрана-85. Нажатием ПК-клавиши <L> в режиме просмотра пульта машины загружаем первую часть. Нажатием <K> проверяем контрольную сумму: 45669. Делаем сброс нажатием <C>. Запуск производим нажатием ПК-клавиши <T> и ещё раз <T>. (Для запуска Фортрана не нажимайте <S> !) Зажигается индикатор «ожидание ответа ПУ».

Затем: **View > Display**, так мы переходим от изображения машины к дисплею. Нужный регистр клавиатуры терминала Фортран в начальном диалоге будет задавать сам, а мы должны перед диалогом установить нули в служебной строке дисплея; для этого нажимаем <F8>, <F9>, и опять <F8>.

Нажатием <F5> включаем ДУП-ЛИН. Фортран задает вопрос:

ТРАНСЛЯТОР ЗАГРУЖАТЬ? (Д/Н)

Нажимаем <D>, т. е. в ответ говорим «да». Фортран сам загружает «с кассеты» вторую часть и спрашивает, какой у нас дисплей:

ДИСПЛЕЙ-01



Нажимаем <3>, так как у нас «013-й» дисплей, фрязинский. Фортран выводит слово READY и ждёт дальнейших указаний в следующей строке. Нажимаем <Пробел>. На экран выводится: FORTRAN-85, и программа ждёт новых указаний.

Если мы не собираемся набирать новую программу, а хотим ввести готовую программку «с кассеты», то нажимаем <S> – это переход в «операционную систему» Фортрана. Затем нажимаем <M>, и на экране эта команда сама допечатывается до МЛ; клавиша терминала остаётся в режиме РУС. Чтобы вызвать «Чтение с ленты», надо ввести Ч; для этого нажимаем на ПК-клаве <Shift>, <~> и опять <Shift>; на экране появляется: ЧТ. Нажимаем <I> – это означает: «исходная программа». Появляется запрос: ИП ИМЯ?

На этом этапе важно правильно ввести первые четыре буквы имени программы, а иначе Фортран «зависнет». Важно также не забыть сменить кассету – открыть через меню **File** вместо Fortran-85\_p1-p2.txt файл с желаемой программой. Например, откроем 1WAR\_\_2WAR\_\_frtrn5m.txt. И введём имя: 1WAR. В ответ на запрос номера версии в виде буквы В нажимаем <1>.

Затем тоже надо не ошибаться: не надо нажимать привычную <Enter>, а надо нажать <Пробел>. Это характерная черта Фортрана – в нём большинство команд подтверждается нажатием клавиши <Пробел>. Фортран читает три копии программы с ленты и обозначает успех тремя восклицательными знаками. Вот так выглядит экран на этой стадии:

```

9 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ТРАНСЛЯТОР ЗАГРУЖАТЬ? (Д/Н) Д
ДИСПЛЕИ-013
READY

FORTRAN-85
S МЛ ЧТ ИП ИМЯ? 1WAR B1
    1.И 1WAR 1. 1070. 8526.!!!
S
=
```

Далее мы можем посмотреть листинг программы (для этого надо нажать <L>), и пытаться редактировать программу. Либо – транслировать её и запустить: в этом случае сначала нажимаем <Т>, на экране появляется запрос «канала трансляции», т. е. запрос варианта подготовки программы к запуску: ТП К. Нажимаем <1> и <Пробел>. Появляется запрос Л?, мол, с листингом или без? Если хотим увидеть не только адреса команд, но и текст команд, то надо нажать <L>, но выполняться листинг будет долго (и с остановками; для продолжения надо нажимать <Пробел>). Быстрее будет без листинга, поэтому нажимаем не <L>, а <Пробел>.

После вывода BD=3824 нажимаем <Пробел>. Появляется буква S, означающая приглашение к вводу очередного указания для «операционной

системы». Нажимаем ПК-клавишу с косой чертой: </>. На экран выводится слово READY. Нажимаем <R>, эта буква сама превращается в слово RUN, и сразу же происходит запуск оттранслированной программы 1WAR.

1WAR это игровая тест-программа, которая у нас тестирует взаимодействие машины с дисплеем и с клавиатурой. Управление в ней (ПК-клавишами):

<1>, <2>, <3> – выбор горизонтальной составляющей скорости точки;  
<7>, <8> – налево, направо; <9>, <6> – вверх, вниз; <0> – выход из игры.

После выхода из этой игры Фортран пишет: READY. Если мы нажмём <R>, то программка запустится снова. И т. д.

Если после выхода из программки, т. е. после того, как Фортран напишет READY, нажмём <Пробел>, то на дисплей выведется FORTRAN-85. После этого нажатием <S> попадаем в «операционную систему», и тогда можно подавать новые указания. Например, можем распечатать на ТПУ листинг программки:

<Home> – нажатие этой клавиши «включает ТПУ»; <A> – на дисплей выводится слово АЦПУ; <L> – нажатие этой клавиши запускает печатание; оно происходит частями, после появления каждого двоеточия приходится нажимать <Пробел>.

Команды и правила, необходимые для набора программ и редактирования приведены в pdf с конспектом описания Фортрана-85 для ДЗ-28; сюда их не перепечатаваю. В конспекте, однако, не очень-то ясно изложен способ работы с библиотеками записей. Рассмотрим этот вопрос подробнее.

Упомянутый файл 1WAR\_\_2WAR\_\_ftrn5m.txt служит примером имитации ленты, содержащей смесь записей разного типа. В начале этой «ленты» записан FORTRAN-5M (о нём речь пойдёт ниже), а затем – фортранная библиотека: исходные тексты и рабочие (транслированные в машинные коды) фортранные программки; они сопровождаются короткими заголовочными файлами. Чтобы увидеть каталог фортранной библиотеки надо, находясь в «операционной системе» Фортрана, подать команду МЛ К, т. е. нажать <М> и <К>. Результат:

```
READY

FORTRAN-85
S ML K
  НЗ Т ИМЯ      В      Б      КС  123
   1.И 1WAR    1.   1070.   8526.!!!
   2.И 2WAR    1.   2664.  21488.!!!
   3.Р 1WAR    1.   4306.  47581.!!!
   4.Р 2WAR    1.   5626.  63418.!!!
   5.И SF-T    1.    199.   1650.!!!
   6.И SF-T    2.    255.   2218.!!!x
S _
```

Обозначения столбцов в этой таблице: НЗ – номер записи, Т – тип (И – исходный текст программы на Фортране, Р – рабочая программа), ИМЯ – имя программы, В – номер версии программы, Б – базовый адрес памяти для данных, КС – контрольная сумма, 123 – номера копий на ленте для индикации качества чтения: удачно прочитанная копия отмечается восклицательным знаком, сбойная копия – знаком минус. Символ ⌘ в конце каталога означает наличие завершающей записи «Конец Библиотеки». В каталог попадают только те программы, которые записаны командами Фортрана и поэтому имеют специальную заголовочную часть; сам Фортран и прочие программы, если они есть на ленте, при выводе каталога не обнаруживаются.

Команда чтения программы с желаемым именем и номером версии уже рассмотрена выше. В Фортран-85 следует загружать только исходную программу; рабочую программу в него не стоит загружать: она зависнет или косяком пойдут ошибочные сообщения (вероятно, это говорит о наличии каких-то пока ещё не выловленных мной ошибок в самом имитаторе).

Можно продолжить формирование библиотеки, добавляя в неё новые фортранные программки или удаляя лишние. Для добавления новой исходной фортранной программки, набранной или прочитанной из какого-либо txt-файла, следует «поставить кассету» с библиотекой, выполнить указанным выше образом просмотр каталога программ в ней, и подать команду записи: МЛ ЗП И (т. е. нажать <М>, затем <З> и затем <И>). Если хотим записать рабочую программку (это удастся только сразу после трансляции), то вместо И подаём команду Р. Фортран не спрашивает имя записываемой программы, так как он уже его знает, а запрашивает лишь номер версии. Вводим номер версии, какой хотим, и нажимаем <Пробел>. Фортран автоматически дописывает программку в библиотеку после уже имеющихся в ней программ и присваивает ей следующий по порядку номер записи. Завершаем эти действия командой записи признака «Конец Библиотеки»: МЛ ЗП КБ (слово КБ набирается нажатием <К>).

Для удаления лишних записей сначала выполняем просмотр каталога (МЛ К), и затем набираем команду типа МЛ БНН2 3 5 0, указывая в ней номера записей, которые хотим сохранить. Здесь для примера в роли номеров сохраняемых записей указаны 2, 3 и 5 (причём, этот учебный опыт делаем с резервной копией файла 1WAR\_2WAR\_frtn5m.txt :-). Команда набирается нажатием <М> и <В>. Список сохраняемых номеров завершаем номером «ноль» – это признак конца списка. И нажимаем <Пробел>. Фортран переписывает указанные программы с начала ленты и сам дописывает в конце признак КБ, показывая это так:

```

$ МЛ БНН2 3 5 0
НЗ Т ИМЯ В Б КС 123
2.И 2WAR 1. 2664. 21488.
3.Р 1WAR 1. 4306. 47581.
5.И SF-T 1. 199. 1650.КБ

```

Если теперь выполнить просмотр каталога, то увидим, что Фортран, кроме того, перенумеровал записи, присвоив им естественные порядковые номера, начиная с единицы:

```

S ML K
H3 T ИМЯ В Б КС 123
1.И 2WAR 1. 2664. 21488.!!!!
2.Р 1WAR 1. 4306. 47581.!!!!
3.И SF-T 1. 199. 1650.!!!!x

```

(Присутствовавшая в начале файла «небиблиотечная» программа оказалась затёртой. Отметим также, что полученному файлу следует дать другое название, чтобы отличать его от первоначального файла, копию которого мы благоразумно сохранили перед нашим экспериментом по переделке библиотеки).

Чтобы положить начало новой библиотеке, надо иметь в ОЗУ набранную или считанную с какой-нибудь ленты фортранную программку. Она обязательно будет иметь какое-то имя. «В НМЛ ставим» созданный заранее пустой txt-файл с желаемым названием. В «операционной системе» Фортрана подаём команду МЛ ЗП (т. е. нажимаем <M>, <Z>) и нажимаем <B>. В ответ на запрос Б Н вводим число 0 – к нему в дальнейшем сама прибавится единица, и результат станет «библиотечным номером» первой записи. Завершаем командную строку нажатием клавиши <Пробел>. В новой строке снова подаём команду МЛ ЗП, указываем тип записи – нажимаем <R> для рабочей программы или <I> для исходной. На запрос номера версии В вводим число (желаемый номер версии), и нажимаем <Пробел>; происходит запись, причём имя программы сохраняется прежнее. И затем не забываем подать команду записи «Конец Библиотеки»: МЛ ЗП КБ.

Все упражнения (с чтением, записью, трансляцией, запуском программ) можно выполнять неоднократно, возвращаясь в «операционную систему» Фортрана, пока не произойдёт какой-нибудь сбой или «зависание ДЗ-28»; при неудачах можно клавишей <Escape> запускать имитатор с самого начала. Если Фортран вместо очередного чтения файла выводит значок ⌘, то надо «перемотать ленту назад» командой МЛ Н (буква Н вводится нажатием <N>).

Рабочую программу почему-то не удаётся запустить в Фортране-85. Однако рабочая программа может быть запущена без загрузки Фортрана, а прямо с пульта машины, как всякая программа в машинных кодах. Но до этого её надо «найти на ленте». Заголовочная часть у фортранной программки состоит из 4 небольших файлов, а саму основную часть Фортран записывает в трёх копиях. Таким образом, одна программка занимает 7 файлов. Значит, например, чтобы загрузить с «ленты» 1WAR\_\_2WAR\_\_frtrn5m.txt основную часть рабочей программы 1WAR (3-я программа в «каталоге», но в начале этой ленты есть ещё файл Фортрана-5М), надо «перемотать вперёд»  $1+7+7+4=19$  файлов. То есть после того, как «поставили кассету» 1WAR\_\_2WAR\_\_frtrn5m.txt, надо в режиме **View > Machine** нажать 19

раз ПК-клавишу <→>, имитирующую кнопку перемотки вперёд. Затем: <L> (команда СЛ – «считывание с ленты»), <K> (команда КП – проверка контрольной суммы; должно получиться число, которое мы видели в «каталоге»: 47581), <C> (сброс счётчика команд в ноль).

Теперь можно либо сразу запустить программу, либо сначала «поставить в НМЛ чистую кассету» и записать на неё программу, нажав <Z> (команда ЗЛ – запись на ленту). Если предполагается много раз экспериментировать с программой, то, конечно, лучше её записать в отдельный txt-файл, чтобы не извлекать её из «библиотеки» каждый раз многократной перемоткой вперёд.

Запуск: <C>, <S>. Переходим к **View > Display**, «включаем ДУП-ЛИН» нажатием <F5>, если «линия» была выключена. На экране появляется R. Дело в том, что программа 1WAR была записана не в Фортране-85, а в Фортране-5М; он обозначает своё состояние готовности не словом READY, а одной буквой R. Нажимаем <S> (в Фортране-5М в состоянии R это команда «Счёт по программе»), и программа начинает работать. Если бы мы аналогичным образом имели дело с программой, транслированной в машинные коды и записанной как «рабочая программа» в Фортране-85, то при аналогичном её запуске с пульта машины на экране появилось бы слово READY, и тогда для старта надо было бы нажать букву <R>, которая сама превратилась бы в слово RUN.

Можно заметить, что фортранная версия игры 1WAR работает гораздо шустрее, и притом чётче откликается на клавиши управления, чем бейсиковская версия (припоминаю, что на реальной ДЗ-28 бейсиковская версия оказалась вообще жутким тормозом...).

Остановим нашу тест-игру, как положено, ПК-клавишей <0>. Однако при этом происходит что-то непонятное: на экран выводятся некие числа с минусом прежде, чем появляется буква R. Возможно, это номера операторов с какими-то автоматически устраняемыми ошибками. Насколько помню, что-то подобное происходило и на реальной машине при запуске транслированных фортранных программ без самого Фортрана. Интересно, что программы так себя ведут только при первом запуске, а при повторных стартах работают совершенно нормально. Т. е. снова нажав <S>, запускаем программу, затем клавишей <0> останавливаем её, и видим на экране вполне нормальные сообщения, запланированные в этой программе; загадочные числа с минусом больше не появляются.

### *Фортран-5М*

FORTTRAN-5М – версия Фортрана в одном файле (в отличие от Фортрана-85, состоящего из двух частей). Предполагаю, что это более новая версия, поскольку она, вроде бы, менее «глючная»: во всяком случае в имитаторе она успешно запускает рабочие программки. В папке [txt] этот Фортран находится на «кассете» Fortran-5m\_KP-138177\_N-10414.txt, а также – в начале упоминавшейся выше «библиотеки записей» 1WAR\_\_2WAR\_\_frtn5m.txt.

Работа с Фортраном-5М в имитаторе мало отличается от работы с Фортраном-85. В свежезапущенный или освежённый ресетом имитатор «ставим кассету» 1WAR\_\_2WAR\_\_ftrn5m.txt, нажимаем <L>, <K> (контрольная сумма должна быть равна 138177, код команды END – на шаге 10414), <C>, <T> и ещё раз <T>. (Для запуска Фортрана не нажимайте <S>!)

Затем: **View > Display**, <F8>, <F9> и опять <F8>. Нажатием <F5> включаем ДУП-ЛИН. На экране печатается буква R, означающая, что Фортран-5М готов к работе (Ready). Нажимаем <T> (наверное, это означает вход в «транслятор»), на экране появляется текст «FORTRAN-5М», и курсор переводится в следующую строку. Для входа в «операционную систему» Фортрана нажимаем <S>. Далее можно выполнять все те же действия с файлами фортранных программ, которые рассматривались выше. Например, для просмотра каталога программ на уже «стоящей в НМЛ кассете» подаём команду МЛ К. Результат выглядит так:

```

                                     9 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0
RT
FORTRAN-5M
S МЛ К
  ПЗ  t  iM9  U      b      ks  123
  1. i 1WAR  1.  1070.  8526.!!!!
  2. i 2WAR  1.  2664.  21488.!!!!
  3. r 1WAR  1.  4306.  47581.!!!!
  4. r 2WAR  1.  5626.  63418.!!!!
  5. i SF-T  1.   199.   1650.!!!!
  6. i SF-T  2.   255.   2218.!!!!x
S _

```

Маленькие латинские буквы вместо больших русских – не глюк имитатора; похожая картина возникала и в реальности. Две версии программки «SF-T» («тест С. Фролова») позволяют заметить, что оба Фортрана по-разному работают со шрифтами: версия 1 написана в Фортране-85, при трансляции и запуске в Фортране-5М она в финальных сообщениях выводит маленькие русские буквы вместо больших латинских; в версии 2 формат сообщений отредактирован в Фортране-5М (это отмечено комментарием, посмотрите листинг программы, нажав <L>), и она выводит текст финальных сообщений правильно.

В Фортране-5М вполне удаётся запускать рабочие программы. Например, загрузим в ОЗУ рабочую программку 2WAR с помощью строки команд

S МЛ ЧТ РП ИМЯ? 2WAR B1 <Пробел>

Фортран выведет на экран строку с параметрами загруженной программы, и в новой строке напечатает букву S – приглашение к дальнейшим командам для «операционной системы». Нажимаем <S>, появляется буква R – признак того, что мы вышли из «операционной системы» и вернулись к начальному меню Фортрана-5M. Ещё раз нажимаем <S> – теперь (т.е. после R) это означает команду «Счёт по программе». Программа 2WAR начинает работать. Заодно опишу, что происходит в этой игре.

Игра 2WAR начинается с запроса «СКОРОСТЬ В = ». В ответ надо ввести число не большее 5 (оно может быть дробным; для начала можно попробовать ввести 1, а в следующих попытках уменьшить или увеличить его, в зависимости от быстродействия вашего компьютера). Оно задаёт скорость американского «бомбардировщика» и тем самым задаёт темп игры на данном компьютере. Бомбардировщик изображается буквой «В», взлетающей из правого нижнего угла экрана – из «USA»; он летит бомбить атомной бомбой Советский Союз – «SU» в левом нижнем углу экрана. Сверху чертится горизонтальная линия с цифрами; это условная «шкала дальности».

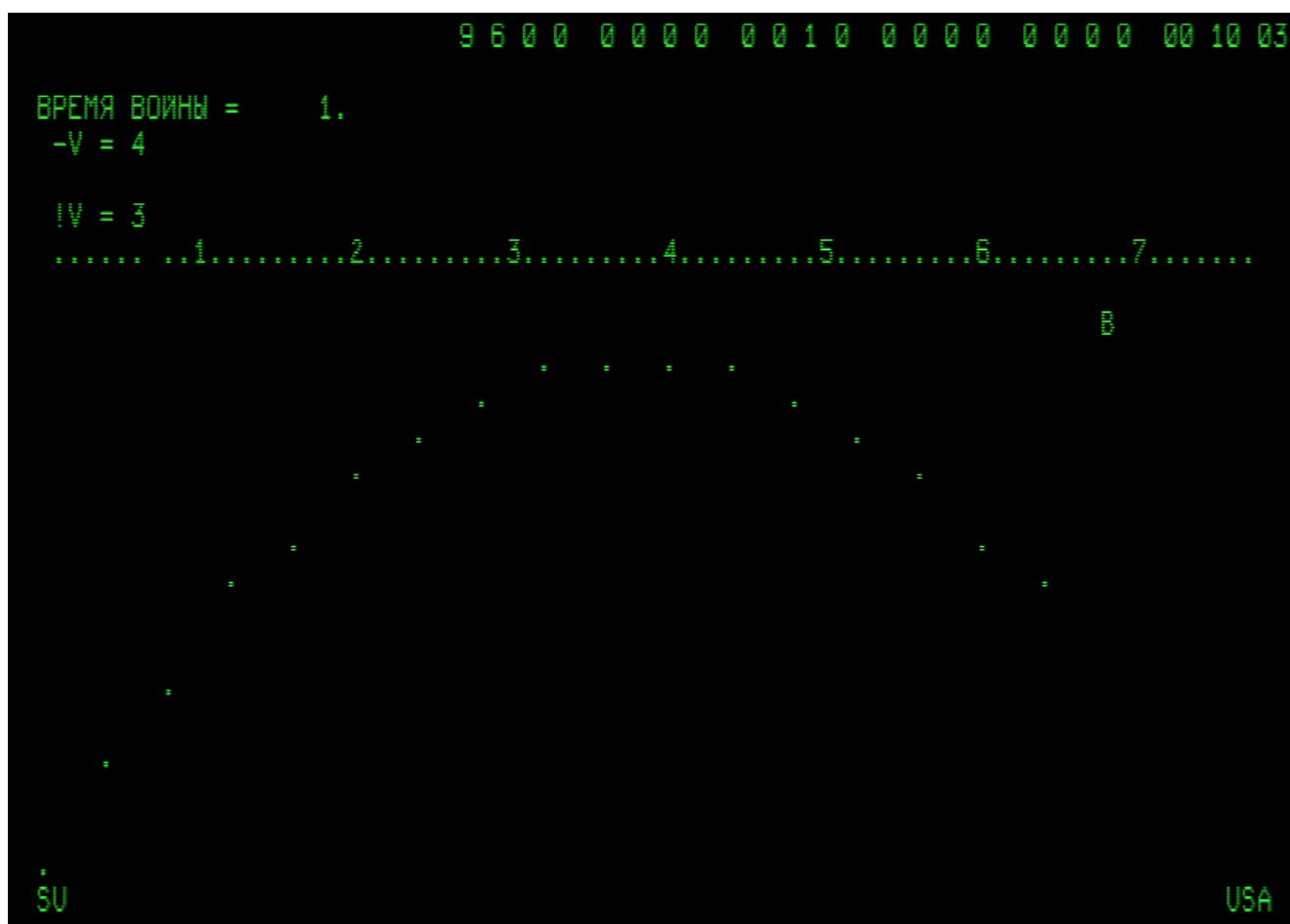
Оказавшись над SU, вражеский «бомбардировщик» сбрасывает бомбу, и она падает вертикально вниз... Экран очищается и появляется текст: «ВЫ УНИЧТОЖЕНЫ». Таков неудачный исход игры. Чтобы этого не случилось, надо в подходящий момент времени, пока «В» ещё в полёте к SU, нажать клавишу <Пробел>. Тогда на экране появляется запрос горизонтальной и вертикальной составляющих вектора скорости V ракеты, которая должна будет сбить вражеский «В» (причём, ввод отрицательного числа вызывает новый старт игры):

–V =  
!V =

Подбор значений для этих составляющих скорости – один из немногих «творческих» моментов в игре, как и выбор момента для ввода вектора скорости и тем самым запуска нашей ракеты. «Шкала дальности» на экране дисплея облегчает эту задачу, позволяя нам примечать раз от раза положение «В» и положение ракеты-точки при их сближении.

Вернее говоря, удачное начало игровых действий должно быть вот каким: как только «В» взлетел из USA, надо пустить ракету с прицелом на территорию USA, – ведь необходимо прежде всего уничтожить военные базы противника, а иначе оттуда будут вновь и вновь взлетать бомбардировщики, даже если мы наловчимся их сбивать без промаха. Затем, второй (или более) ракетой следует сбить «В»:







При редактировании фортранных программok выявился неприятный факт: вот такая, очень удобная форма оператора условного перехода

IF (арифм\_выражение) M<sub>1</sub> , M<sub>2</sub> , M<sub>3</sub>

переход к метке: M<sub>1</sub> если арифм\_выражение < 0  
M<sub>2</sub> если арифм\_выражение = 0  
M<sub>3</sub> если арифм\_выражение > 0

почему-то чаще всего не работает: при листинге в имитаторе появляется сообщение об ошибке (в обоих Фортранах). Избежать ошибки удаётся (и то не всегда) следующим приёмом: надо в предыдущей строке присвоить желаемое арифметическое выражение одиночной переменной, например: B1=арифм\_выражение, и затем в операторе IF указать эту переменную: IF (B1) M<sub>1</sub> , M<sub>2</sub> , M<sub>3</sub> . При работе с Фортраном-5M на реальной машине ДЗ-28 такой глюк никогда не возникал. Значит, этот глюк указывает на присутствие какой-то хитрой ошибки в имитаторе, в его интерпретаторе команд машины ДЗ-28. Её вылавливание может оказаться очень трудоёмким (для этого, наверное, придётся найти и дизассемблировать ту часть Фортрана, которая ответственна за обработку условных переходов), так что пока откладываю это дело «в долгий ящик»...

### *Выстра*

Как работать с Выстрой, здесь не рассказываю полностью, всё изложено в pdf с конспектом описания Выстры; (вернее, не всё, а кое-что из самого важного; многое приходится выяснять экспериментально). Рассмотрим запуск Выстры и некоторые первые шаги. Выстра запускается в имитаторе так же, как и Бейсик с b4=0: с настройкой служебной строки дисплея «по умолчанию»

9 6 0 0   0 0 0 0   0 0 1 0   0 0 1 0   0 0 0 0   ...

Открываем командой меню **File > Open txt-file as Tape** файл D3-28\_Vystra\_\_KP-113478\_\_N-9478.txt. Загружаем его в машину в режиме **View > Machine** нажатием <L>. Проверяем нажатием <K> контрольную сумму: 113478. Затем делаем сброс ПК-клавишей <C>, и старт ПК-клавишей <S>.

Переходим к **View > Display**, и включаем ДУП-ЛИН ПК-клавишей <F5>. На экране дисплея должно появиться приглашение к заданию функции: буква Ф.

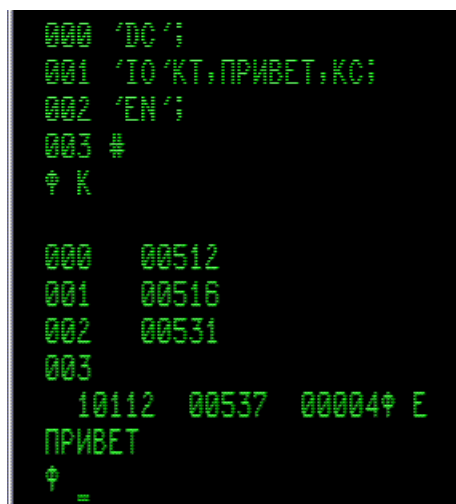
Например, наберём простейшую программку. Для этого нажимаем <I> или <И> и подтверждаем командой ПС, т. е. нажимаем <Enter>. Маркер переходит в следующую строку. Набираем три ноля 000 и подтверждаем нажатием <Enter>. Маркер сам опускается в нижнюю строку экрана дисплея, там выводятся три ноля и пробел. Такова специфика набора программы в Выстре: автоматически

выводится номер строки и за ним пробел, причём заполнение экрана идёт снизу вверх, так что при наборе каждой новой строки прежние строки заново выводятся над набираемой (долго, для ускорения попробуйте свернуть окно имитатора). Кроме того, Выстра всё время норовит сама переключить РУС/ЛАТ-регрстр клави; поэтому его надо часто проверять, нажимая <F1>, и при необходимости менять на нужный клавишей <Ctrl>. Допустим, мы набрали вот такой текст (номера строк и пробелы после номеров строк Выстра делает сама, их не надо набирать):

```
000 'DC';  
001 'Ю'КТ,ПРИВЕТ,КС;  
002 'EN';  
003 #
```

Здесь только слово ПРИВЕТ набирается в РУС-регистре; остальные символы в ЛАТ-регистре. Набор каждой строки, кроме последней, завершается командой ПС, т. е. нажатием <Enter>. Набор последней строки, с обязательным значком #, означающим конец программы, надо завершать командой ВК, т. е. нажатием <Backspace>; тогда Выстра в следующей строке выведет букву Ф.

Нажмём <K> и ПС, это означает команду компиляции. После компиляции, сопровождающейся выводом на экран номеров строк и операторов, в конце снова увидим букву Ф. Нажимаем <E> и ПС, это команда к выполнению (экзекуции) откомпилированной программки, и наблюдаем результат. Вот как это выглядит вместе с набором программки:



```
000 'DC';  
001 'Ю'КТ,ПРИВЕТ,КС;  
002 'EN';  
003 #  
Ф К  
  
000 00512  
001 00516  
002 00531  
003  
10112 00537 00004Ф E  
ПРИВЕТ  
Ф =
```

Поскольку текст программки ещё имеется в ОЗУ, его листинг можно распечатать. Для этого нажимаем <Home>, т. е. «включаем ТПУ», переводим клави терминала в режим РУС нажатием <Ctrl>, нажимаем <P> (при этом на экран выводится буква П, означающая команду «печатать»), и нажимаем <Enter>, т. е. нажимаем ПС. В Выстре почти все команды надо подтверждать нажатием ПС.

Можно текст программки сохранить «на ленте». С этой целью следует придумать и запомнить для будущих считываний имя программы. Назовём её, например, TEST. Для записи нашей программки «на ленту» надо не забыть вынуть кассету с Выстрой (**File > Close txt-file as Tape**), заготовить в нашей папке [txt] новый txt-файл с каким-то именем, «вставить его в НМЛ» (**File > Open txt-file as Tape**), и подать Выстре команды: <A> ПС (т. е. <Enter>) TEST ПС. По командам ПС маркер переходит в новую строку. В результате последнего ПС образуется желаемая запись в txt-файле.

Чтобы убедиться, что запись произведена, можно сделать ресет имитатора клавишей <Escape>, снова загрузить и запустить Выстру, «вынуть» её, «поставить кассету» с проверяемой записью и считать её командами чтения с ленты:

<C> ПС TEST ПС. Здесь, как и при записи, TEST это имя программы.

В случае удачного считывания имя программы появляется ещё раз, в следующей строке. Компиляция нажатием <K> ПС и нажатием <E> ПС показывают, что всё нормально.

Опыты можно продолжить следующим образом. Переведём клавишу в режим РУС, нажав <Ctrl>, и нажмём <V> ПС – это команда Ж, означающая уничтожение текста программки в памяти Выстры; при этом откомпилированная программка в ОЗУ машины сохраняется. Снова включим РУС и нажмём <G> ПС – это команда Г, означающая уничтожение самой Выстры. С этого момента машина уже не работает по программе, а находится в состоянии «останова с индикацией».

Откомпилированную программку теперь можно запускать командами с пульта машины: <C> – сброс счётчика команд, и затем <S> – старт. Возвращаясь к просмотру экрана дисплея, увидим, что после каждого такого запуска на экране появляется слово ПРИВЕТ, т. е. откомпилированная программка работает самостоятельно, без Выстры. Нажав «на пульте машины» <C> и затем <Z>, можно сохранить её «на ленту» – на новую, или на ту же; во втором случае программка запишется как четвёртый файл, потому что первые три файла это две части заголовка и сам выстровский текст нашей программки. Скомпилированная в машинные коды и записанная машинной командой ЗЛ программка считывается с ленты и запускается без всякой Выстры: <C>, <L>, <C>, <S>.

Для иллюстрации работы Выстры с НМЛ в папке [txt] есть пример «ленты с библиотекой программ»: frmj\_inpx\_test\_\_1prg\_\_3prg\_vystra-text.txt. В начале этого txt-файла записана одна небиблиотечная (т. е. не имеющая специального заголовка) программа «FRMJ» в машинных кодах; она запускается, как обычно (<C>, <S>), и строит на экране дисплея «фракталы» типа «Mandelbrot» или «Julia». Эта программа скомпилирована Выстрой в имитаторе из исходной программы, которая записана в «библиотеку» Выстрой в её формате (2 части заголовка плюс основной файл, содержащий выстровский текст программы) под именем FRMJ вслед за упомянутой рабочей программой FRMJ. В этой программе есть очень

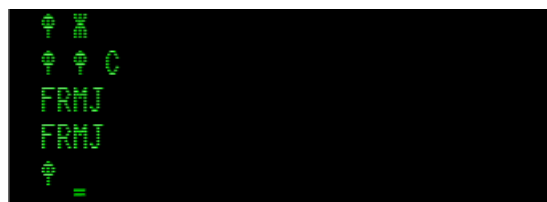
полезная для всех выстровских программ процедура INP(X) – ввод числа с клавиатуры дисплея (а в самой Выстре почему-то нет такого ввода; предусмотрен только ввод чисел с пульта машины). Выстра позволяет выделять полезные куски исходных программ и записывать их в виде отдельных программ.

Вторая запись в «библиотеке», с именем INPX, это как раз упомянутая полезная процедура. К голой процедуре INP(X), заканчивающейся в строке 033, там добавлено семь строк: с ними запись становится программкой, которую можно после считывания скомпилировать (командой К ПС) и запустить (командой Е ПС) для контроля. Для применения же INP(X) в новой программе эти лишние семь строк надо удалить.

Третья и последняя запись в библиотеке – та коротенькая программка TEST, которая рассматривалась выше. Здесь она играет роль метки «конец библиотеки», так как в Выстре нет аналога фортрановской команды КБ. Если условиться всегда заканчивать библиотеку программкой с именем TEST, то список имён выстровских программ на ленте можно получать командой чтения программы с именем TEST, потому что Выстра, отыскивая на ленте заданную программу, выводит на экран имена всех библиотечных программ, попадающихся перед заданной. В нашем примере это выглядит так: после буквы Ф подаем команду С, подтверждаем её нажатием ПС (т. е. <Enter>), набираем имя TEST, подтверждаем нажатием ПС. Выстра читает и выводит на экран имена попадающихся на ленте выстровских программ, пока не найдёт и не загрузит в ОЗУ заданную программу:

The screenshot shows a computer terminal with a black background and green text. At the top right, a hexadecimal address is displayed: 9600 0000 0010 0010 0000. On the left side, a list of programs is shown, each preceded by a green 'Ф' (Phi) symbol. The list includes: C, TEST, FRMJ, INPX, and TEST. Below the last 'TEST', there is a 'Ф' symbol followed by a hyphen '-'.

Допустим, мы решили загрузить FRMJ. Важны два нюанса. Во-первых: надо удалить из ОЗУ уже загруженную программку (т. е. TEST), в противном случае новая загрузка подстыкуется к концу предыдущей. Команда удаления программки из ОЗУ: Ж ПС. Во-вторых, надо сделать перемотку назад. В Выстре нет команды перемотки назад, но Выстра допускает многократную остановку и запуск. Поэтому переходим к **View > Machine**, и нажимаем <Enter> (это имитация нажатия кнопки Ш на пульте машины) – так в имитаторе останавливается Выстра. Нажимаем <←> – это имитация кнопки «перемотка назад». И нажимаем <C>, <S> – это запуск Выстры; при этом она снова выводит на дисплей букву Ф, так что на экране будут две буквы Ф рядом. Наконец, подаём команду считывания с ленты: С ПС FRMJ ПС. Выстра ищет и загружает программу FRMJ, выводит на экран в новой строке её имя, и выводит очередное приглашение Ф:



Допустим, теперь мы хотим вывести на экран листинг этой программы. Тут всплывает ещё один нюанс: Быстре надо задать номер строки, и тогда она выведет столько строк, сколько поместится на экране, с номерами меньшими заданного, включая и строку с номером равным заданному (такую строку условимся называть «текущей»). Но это не сработает, если задать номер строки, больший, чем последний имеющийся в программе номер строки. Значит, желательно заранее знать номер последней строки. Для этого следует придерживаться правила: в нулевую строку программы надо помещать комментарий с указанием номера последней строки; тогда можно будет сделать сначала листинг нулевой строки (или нескольких первых строк), и тем самым узнать допустимые номера строк.

Для примера посмотрим листинг первых трёх строк в FRMJ. Команда листинга: И ПС *трёхзначный\_номер\_строки* ПС. Результат:

```
000 x FRMJ 141 СТРОКА ; 'PR 'INP (X) ;
001 'DC 'B,W(6); 'KO '405,W,404,B,413,1312;B+=1; 'IF '(B/2-INT(B/2+.1))#0;B+=1;
002 'FI 'B=16*INT(B/2+.1); 'KO '405,B,413,1200,1100,1200,1104,5,703,701,413,
```

Комментарием является текст между символом  $\Phi$  и точкой с запятой; видно, что номер последней строки в FRMJ есть 141.

Маркер находится в начале текущей строки; если начать набирать текст, то начало текущей строки будет редактироваться – таким путём можно добавить туда комментарий или, например, метку. Если нажать ВК (т. е. <Backspace>), то появится признак ожидания новой команды: буква Ф. Если теперь подать команду ) ПС, то из программы будут удалены все строки перед текущей строкой, а если подать команду ( ПС, то будут удалены все строки после текущей строки. Так при должном усердии можно нарезать программу на части, записать части на ленту или на разные ленты, а затем считывать их, подстыковывая к концу набираемого текста. После каждой подстыковки следует подавать команду перенумеровки: <9> ПС; она восстанавливает нумерацию строк в порядке возрастания номеров.

FRMJ служит примером программы, требующей много памяти. Поэтому после её компиляции следует перед её запуском удалить из ОЗУ исходный текст командой Ж ПС и удалить саму Быстру командой Г ПС. После этого FRMJ запускается с пульта машины: <C>, <S>. В папке [txt] есть также файл frmj\_KP-38246\_N-3445.txt – это рабочая программа FRMJ, скомпилированная в давние времена на реальной машине ДЗ-28.

## 6. Работа с машиной без дисплея

В этом разделе речь идёт об имитации собственно ДЗ-28, без дисплея. Конечно, наиболее интересна работа с дисплеем и с «высокоуровневыми» программами, такими как Бейсик, Фортран, Выстра. Однако машина ДЗ-28 и сама по себе вызывает приятные, ностальгические воспоминания – о довольно сложном (и поэтому интересном) программировании в машинных кодах. Ниже описываются, в основном, правила соответствия между ПК-клавишами в имитаторе и кнопками пульта ДЗ-28. Для осознанной работы с имитатором по этим правилам необходимо пользоваться справочной информацией о пульте машины и о её системе команд из документации, поставившейся с ДЗ-28:

«Инструкция по эксплуатации»,  
«Справочник программиста».

Сразу после запуска проги `imitator_1_0.exe` на мониторе открывается окно проги с изображением машины. Напомню, что в данной версии имитатора изображение машины (и клави) строится из шести фрагментов – из bmp-рисунков 256x256 px; поэтому на разных ПК возможны различные дефекты изображения в виде вертикальных и горизонтальных полос, разделяющих фрагменты. Для их устранения попробуйте потянуть мышью вертикальные или горизонтальные края окна проги, меняя его размер.

Машина сразу после запуска имитатора находится в режиме «Работа» – горит индикатор слева от кнопки Р, на табло Х и Y светятся нули, означающие, что регистры Х и Y содержат нулевые значения. Это ещё не «работа по программе», а лишь состояние «останова с индикацией». Из режима «Работа» можно произвести запуск «работы по программе», если программа присутствует в ОЗУ машины. Чтобы в ОЗУ появилась программа, её надо туда загрузить «с магнитной ленты» (см. раздел 3) или ввести руками байт за байтом с пульта машины в режиме «Ввод»:

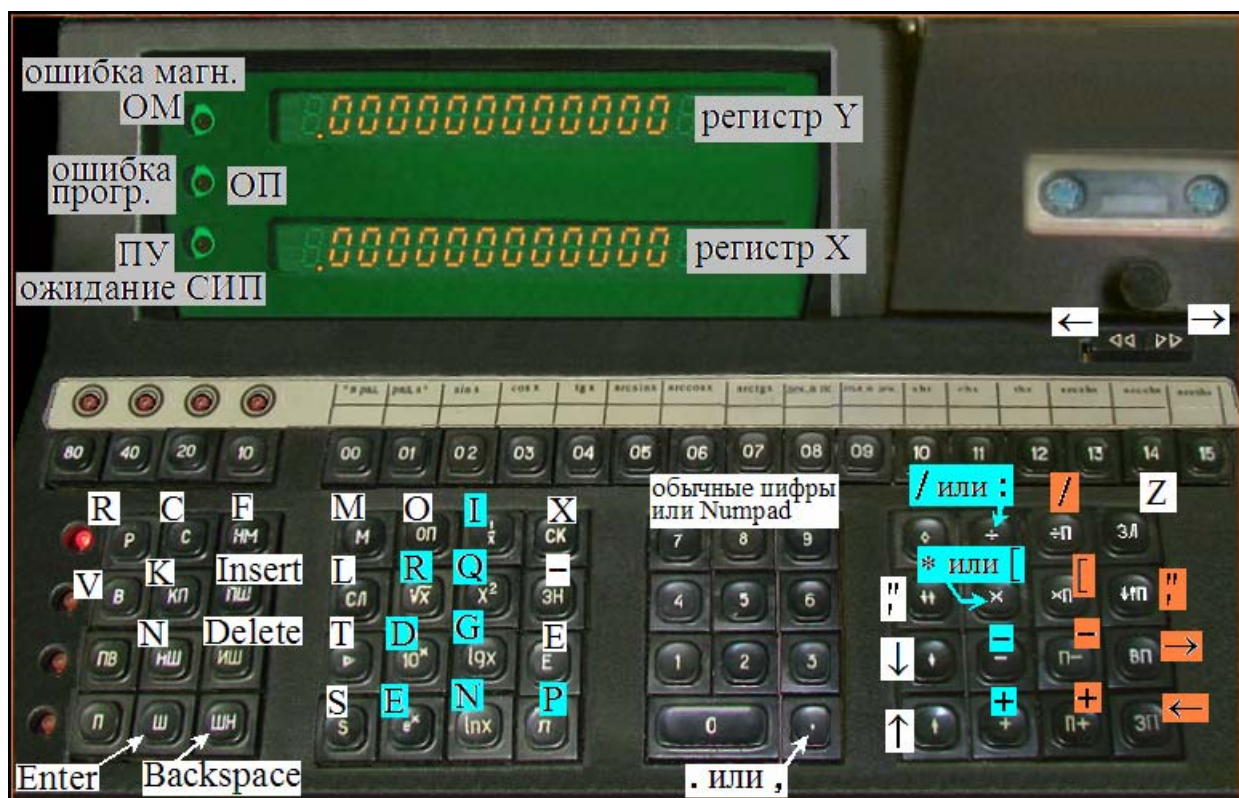
Режим «Ввод» включается нажатием <V>.

Нажатие <R> – переход в режим «Работа».

Режимы ПВ (печать программы при её вводе) и П (печать программы) в имитаторе не работают; на практике печать каждого байта при вводе никогда и не нужна, а отсутствие распечатки программы командой П у нас компенсируется возможностью распечатки программы «на ТПУ» специальной программой-распечатчиком, загружаемой в любую свободную область ОЗУ (см. раздел 8).

Не имитируется и кнопка с ромбиком – печать числа из регистра Х. Остальные кнопки имитируются ПК-клавишами; правда, имитация некоторых команд не полная и не точная; вот схема соответствия кнопок и ПК-клавиш:





На этой схеме надписи на зелёном табло – пояснения, где какой регистр, и что означают три индикатора, расположенные слева от табло регистров X и Y; они не относятся к кнопкам.

Белый цвет фона надписей на кнопках условно означает, что указанные в надписях ПК-клавиши работают без прижимания других клавиш. Напомню назначение тех из них, которые действуют в режиме «Работа»:

<R> / <V> – переключают режимы «Работа» / «Ввод».

<C> – сброс счётчика команд и масок прерываний, ресет указателя стека.

<K> – команда вычисления контрольной суммы программы в ОЗУ машины.

<L> – считывание с ленты.

<Z> – запись на ленту.

< ← > – перемотка назад.

< → > – перемотка вперёд.

<S> – старт работы по программе (команда GO с кодом 0514).

<T> – начало команды «найти метку и запустить работу по программе с этой метки». В имитаторе табло X гаснет, а мы должны ввести второй байт команды: либо цифрами, либо одной из букв: <T>, <S>, <M> (это зависит от того, как составлена программа; способ её запуска должен указываться в документации к программе). Номер метки в виде четырёх цифр в имитаторе вводится с постоянно нажатой клавишей <Ctrl>.

<F> – начало команды «найти (find) метку». В имитаторе табло X гаснет, а мы должны ввести второй байт команды: номер метки в виде четырёх цифр; он вводится с постоянно нажатой клавишей <Ctrl>. В имитаторе эта команда отличается от <T> тем, что здесь не происходит запуска программы. Перейдя нажатием <V> в режим «Ввод», мы увидим команду в ОЗУ машины, следующую за найденной меткой; саму метку мы увидим, выполнив шаг назад клавишей <Backspace>. Такой поиск меток удобен для изучения или отладки сложных программ.

<O> – гашение индикатора ОП. Сброс, т. е. <C> тоже гасит индикаторы.

<X> – гашение табло X; может применяться перед вводом в X нового числа.

<E> – сброс порядка у числа, находящегося в X; обычно применяется перед вводом порядка числа X.

< минус > – изменение знака на противоположный у числа в X, или изменение знака порядка у числа в X, если перед эти нажималась <E>.

Клавиши с цифрами – в режиме «Работа» служат для ввода числа в регистр X (или для ввода номеров меток, как пояснялось выше).

Клавиши с точкой или с запятой – ввод десятичной точки для числа в X.

< ↑ > – засылка в регистр Y того же числа, которое имеется в X.

< ↓ > – засылка в регистр X того же числа, которое имеется в Y.

Клавиша с кавычками ( “ и ’ ) – обмен числами между регистрами X и Y.

<Enter> – останавливает «работу машины по программе», если она шла; тем самым машина переводится в состояние «остановка с индикацией». При этом счетчик команд указывает на ближайшую невыполненную команду. (Сброс <C> тоже останавливает работу машины по программе, но при этом счетчик команд сбрасывается к нулевому шагу).

*В режиме «Ввод» действие цифровых клавиш и клавиши <Enter> иное:*

Пятизначный номер на X-табло это номер шага (если содержимое регистра ВР равно нулю, то номер шага равен содержимому регистра РС, т. е. программному счетчику; в общем случае  $PC = BP + \text{номер шага}$ ). Y-табло показывает двухбайтное значение РС в 16-ричной системе счисления; байты разбиты на тетрады; каждая тетрада принимает значения в диапазоне 0, ..., 15.



Рядом с номером шага высвечивается байт содержимого ОЗУ на этом шаге, т. е. содержимое байтовой ячейки ОЗУ машины с адресом, равным содержимому регистра РС; этот адрес выведен в 16-ричном виде на табло Y.

Нажатие <Enter> увеличивает на единицу номер шага, а нажатие <Backspace> уменьшает его на единицу. Таким путём можно байт за байтом просматривать содержимое ОЗУ машины (и её регистров в служебной зоне ОЗУ, см. «Справочник программиста» для ДЗ-28).

<N> – нажатием этой клавиши номер шага гасится, и машина ожидает ввода пятизначного числа – десятичного значения номера шага, который мы хотим посмотреть. Мы должны нажать пять раз цифровые клавиши; тогда рядом с введенным номером шага изобразится содержимое соответствующей ячейки ОЗУ.

Если в этом режиме вводить цифры без предварительного нажатия <N>, то машина воспримет их как ввод нового байт-кода в данную ячейку ОЗУ. Чтобы отказаться от начатого, но незавершённого ввода, надо нажать <Enter> или <Backspace>. Чтобы завершить ввод, надо ввести именно четыре цифры (две тетрады: по две цифры в каждой тетраде) и обязательно нажать <Enter>. Чтобы отказаться от завершения ввода после того, как уже введены четыре цифры (например, мы ошиблись при их вводе или передумали), надо нажать <Backspace>. Таким образом, в имитаторе можно вручную ввести в ОЗУ машины программу, написанную «в машинных кодах».

В реальности ввод байтов в ОЗУ с пульта машины производится кнопками прямого кодирования: старшая тетрада каждого байта вводится в коде 8-4-2-1 кнопками, расположенными под четырьмя индикаторами, а младшая тетрада вводится одной из кнопок в ряду 00, 01, ..., 15. В имитаторе такая процедура упрощена, а кнопки прямого кодирования отсутствуют.

Для отладки небольших программ и для их модификации могут быть полезными операции «Исключить Шаг» и «Прибавить Шаг»; на ПК-клавиатуре в имитаторе им назначены клавиши <Delete> и <Insert>. При исключении шага байт-код на текущем шаге теряется, на его место переписывается байт из следующей ячейки, и т. д. Прибавление шага заключается в том, что байты в ОЗУ, начиная с текущего, переписываются в ячейки с большим на единицу адресом, а на текущий шаг записывается байт-код команды GO, т. е. 0514. Затем можно его заменить любым желаемым байт-кодом, вводя новые четыре цифры. Следует помнить, что такая модификация не испортит работоспособность программы только в том случае, когда она не сбивает адреса ветвлений и переходов, имеющих в программе.

### *Простейший пример ввода программы с пульта машины*

Введите указанным выше путём с нулевого шага программку, содержащую всего 9 байт:

0700 0604 0701 0600 0412 0615 1402 0004 0512

Смысл этих команд прост:

0700 – ввод в регистр X числа 0,

0604 – пересылка содержимого X в Y,

0701 – ввод в регистр X числа 1,

0600 – сложение  $X+Y$  и засылка результата в Y,

0412 0615 – двухбайтовая команда «Пауза», для индикации X и Y на табло,

1402 0004 – двухбайтовая команда перехода к шагу «3», т. е. к сложению  $X+Y$   
и к засылке результата снова в Y.

0512 – команда END, необходимая здесь только для того, чтобы можно было записать эту программку «на ленту» (кстати, в нашей папке [txt] эта программка уже есть – с именем y-test\_\_KP-88\_N-8.txt.)

Это упражнение показывает, как вводить программу с пульта машины. В данном примере запрограммирован простой счётчик: после каждой паузы с индикацией число в регистре Y будет увеличиваться на единицу. Чтобы запустить программку, надо перейти нажатием <R> в режим «Работа», нажать <C> и затем <S>. Для останова можно нажать <Enter>; машина закончит выполнять текущую команду и прекратит счёт. После останова мы сможем клавишей <V> перейти в режим «Ввод» (пока велась работа по программе, такой переход был заблокирован; проверка этого факта служит способом узнать в имитаторе, находится машина в «программном режиме» или уже остановилась, если режим её работы неочевиден).

В качестве примера модификации программы установим номер шага равным 3, выполним «Прибавить Шаг», и введём вместо 0514 код 0700. Тем самым мы запрограммировали занесение в X числа 10. Запустив снова программку, мы увидим, что теперь счёт идёт с шагом 10.

Подобным же образом легко модифицировать программку так, чтобы в ней X и Y перемножались, а не складывались: для этого байт-код 0600 надо заменить на 0602. Чтобы получался не ноль, надо изменить и команды засылки чисел в X. Если запрограммировать засылку в X числа с порядком 10 или больше, то при выполнении программы будет быстро достигнуто «превышение разрядной сетки», так как в ДЗ-28 порядок числа задаётся всего двумя десятичными разрядами, и, значит, не может превышать 99. При превышении разрядной сетки имитатор начинает глючить и выводит на табло что-то бестолковое (потому что в имитаторе математика сделана пока в предварительном варианте, без должных проверок корректности действий, и не в соответствии с «математикой ДЗ-28»).

На практике написание тест-программок для имитатора удобнее делать в текстовом редакторе, например, в «Блокноте», и сохранять результат как txt-файл для последующей загрузки в имитатор через его «магнитофон». Байт-коды следует набирать столбиком, после каждого байта нажимая <Enter>, – такой формат предусмотрен в имитаторе при «чтении с ленты».

В папке [txt] есть три «стандартные» (обычно входившие в комплект поставки машины ДЗ-28) программы, работающие без дисплейного терминала:

statistika\_\_KP-7074\_\_N-806.txt – пакет подпрограмм «Статистика» (причём, в конце этого же txt-файла напечатаны пояснения об особенностях запуска «Статистики» в предыдущих версиях имитатора). Текст подпрограмм, а также правила и примеры работы с ними приведены в брошюре «Руководство оператора 3.857.100 Д12» из комплекта документации к машине ДЗ-28.

D3-28\_test\_028\_\_KP-60000\_N-5999.txt – тест системы команд,  
D3-28\_test\_017\_\_KP-5895\_N-527.txt – тест НМЛ.

О работе этих «стандартных» тестов рассказано в брошюре «Инструкция по эксплуатации» для ДЗ-28. См. также пояснения ниже – в разделе 8.

### *Выполнение одиночных команд с пульта машины*

Машина ДЗ-28 допускает выполнение команд, подаваемых с пульта кнопками прямого кодирования: старшая тетрада байта вводится в коде 8-4-2-1 кнопками, расположенными под четырьмя индикаторами, а младшая тетрада вводится одной из кнопок в ряду 00, 01, ..., 15.

Прога может имитировать подобную подачу команд, но не точно (ведь на ПК-клаве нет кнопок прямого кодирования). Для этого в имитаторе коды команд надо набирать цифровыми ПК-клавишами, удерживая постоянно нажатой клавишу <Ctrl>. Отпускание <Ctrl> до окончания набора команды имитатор воспринимает как отказ от команды. Если команда однобайтовая, то после набора её байта перед набором кода следующей команды можно отпустить <Ctrl>. Если команда двухбайтовая, и мы желаем её выполнить, а не отказываться, то отпустить <Ctrl> можно только после набора двух байтов.

Индикаторы 8-4-2-1 в имитаторе отображают набор только старшей тетрады каждого байта, а при наборе второй тетрады они гаснут – так в имитаторе индицируется факт получения команды машиной (на реальной же ЭВМ ДЗ-28 они продолжают гореть; программист ощущает подачу команды своими пальцами, поскольку реально нажимает одну из кнопок в ряду 00, 01, ..., 15).

Пример: наберём, удерживая <Ctrl> прижатой, цифры 0701. Результат будет такой же, как при вводе в регистр X числа 1 клавишей <1> без <Ctrl>. Затем наберём, удерживая <Ctrl> прижатой, цифры 0700; результат такой же, как если бы мы продолжили ввод числа в X нажатием <0> без <Ctrl>, т. е. на X-табло высветится число 10, если до этого там были одни нули.

Нюанс: перейдя в режим «Ввод», мы увидим, что в имитаторе каждая подача команд с пульта машины сопровождалась увеличением номера шага (увеличением РС). В реальности такого увеличения РС не должно быть; но я

не стал пока усложнять имитатор для борьбы с этой неточностью имитации. В руководствах по ДЗ-28 не рекомендуется подача с пульта команд, следствием которых должны быть переходы или ветвления, т. е. вполне определённые изменения РС; так что, одиночные команды не слишком-то и полезны на практике. Примерно из этих же соображений в имитаторе пока отсутствует режим пошагового исполнения программы, имеющийся в реальной ДЗ-28.

Некоторые одиночные команды в реальной ДЗ-28 могут быть исполнены нажатием специальных кнопок в правой части пульта, таких как хП, П-, П+, и др. – это команды арифметических действий с участием десятичных ячеек памяти (см. руководства по работе с ДЗ-28). В имитаторе им соответствуют ПК-клавиши, которые на приведённой выше схеме (см. стр. 39) показаны на красном фоне – их следует нажимать, удерживая постоянно нажатой клавишу <Ctrl>.

### *Применение машины в качестве калькулятора*

ПК-клавиши, показанные на синем фоне (см. стр. 39), следует нажимать, удерживая постоянно нажатой клавишу <Shift>, – тогда им соответствуют математические функции, обозначенные на кнопках пульта.

Пример: введём в X цифровой клавишей <1> число 1, клавишей <↑> скопируем его в Y, введём в X клавишей <2> число 2 (всё это делается без прижимания других клавиш) и, наконец, удерживая прижатой <Shift>, нажмём клавишу <+>. На табло регистра Y появится результат сложения чисел из X и Y (т. е. появится число 3).

Ещё пример: нажатием <P> с постоянно прижатой <Shift> мы вводим в регистр X число  $\pi$ . Аналогичным образом, если мы введём в X число 1, и нажмём с постоянно прижатой <Shift> клавишу <E>, то получим в регистре X результат вычисления  $e^1 = e$  – это число Непера, т. е. основание натуральных логарифмов: 2,71828182846 .

Такие вычисления можно комбинировать с командами математических операций, которые подаются с постоянно прижатой <Ctrl>. Например, после того, как указанным выше способом в X было введено число  $e$ , наберём цифровыми клавишами с постоянно прижатой <Ctrl> команду 0802. Тогда на табло X появится результат вычисления  $\sin(X)$ ; в данном примере это  $\sin(e) = 0,410781290502$  .

В имитаторе пока отсутствует возможность выполнения команды RES (вместо неё в имитаторе действует грубая подделка); в реальной ДЗ-28 эта команда позволяет вывести на табло «остаток» арифметического вычисления и тем самым повысить точность вычисления свыше 12 десятичных разрядов. Вообще, в имитаторе пока ещё не сделана точная имитации и самой 12-разрядной математики ДЗ-28; поэтому при исполнении примеров из руководств возможны отличия от результатов, указанных в документации к ДЗ-28.

## 7. Регулировка быстродействия имитатора

Имитатор, к сожалению, не воспроизводит динамику работы реальной машины ДЗ-28. Хотя на современном ПК вычисления выполняются намного быстрее, чем в ДЗ-28, вывод текстов на дисплей (или, например, вывод символов оператором PTINTTAB при построении графиков в Бейсике) происходит в имитаторе медленнее, чем следовало бы. Скорость имитации такого вывода зависит от быстродействия конкретного ПК и от свойств его видеосистемы.

Регулировать быстродействие имитатора в сторону увеличения невозможно. А замедлить работу имитатора можно – введением задержки между обращениями к интерпретатору машинных команд. Такая задержка может потребоваться, в основном, в двух случаях: 1) если ПК оказался очень шустрым, так что, например, «полёт точки-ракеты» в тест-игре 2WAR происходит слишком быстро; 2) если исследуется программа для ДЗ-28, в которой не запланирован периодический вывод информации на дисплей или на табло. Рассмотрим простой пример подобной ситуации.

Заменяем в разобранной на стр. 41-42 программке «y-test» двухбайтовую команду 0412 0615, выводившую «счётчик Y» на табло Y, двумя однобайтовыми командами 0514 (это «пустая» команда GO, она просто передает управление следующей команде в программе):

0700 0604 0701 0600 0514 0514 1402 0004 0512

Откроем в Windows «Диспетчер задач» на вкладке «Процессы» (или «Монитор ресурсов», или какую-то аналогичную программу, способную показывать процент использования центрального процессора ПК различными программами). Запустим имитатор ДЗ-28. Можно видеть, что он практически не потребляет ресурсов компьютера, пока в нём не запущена программка. Однако, если мы запустим в имитаторе указанную выше модифицированную программку «y-test», то увидим, что нагрузка процессора возрастает почти до максимально возможной; например, у меня на стареньком ПК с Windows XP использование процессора подскакивает до 90% и более! Процессор без передышки крутит цикл с нашей программкой, так что через некоторое время температура его заметно повышается, вентилятор начинает шуметь (поэтому лучше остановить машину, нажав <Enter>. На Y-табло будет видно, какое большое число циклов успел накрутить наш счётчик за время работы программки без вывода Y на табло).

Чтобы этого не происходило, надо чередовать работу процессора ПК над вычислительной задачей в имитаторе с гораздо более спокойной работой в операционной системе. Данная версия имитатора автоматически обеспечивает такое чередование (и тем самым полностью устраняет возможность перегрева процессора), отдавая несколько миллисекунд своих «квантов времени» операционной системе после каждого вывода «кадра» – вывода изображения

машины, клавиатуры или экрана дисплея. Но если обновление «кадра» не предусмотрено в имитируемой программке, то и получается работа процессора без передышки, как в только что рассмотренном примере. Подобная нежелательная ситуация может возникнуть при имитации любой такой программки, в которой вычислений много, а результат выводится редко.

В подобных ситуациях следует применить искусственную задержку вычислений (Delay), позволяющую командой `sleep(1)` отдавать операционной системе 1 мс времени каждый раз после заданного количества циклов работы имитатора. В данной версии имитатора это количество циклов обозначено как `dwN_delay`.

При запуске имитатора параметр `dwN_delay` равен нулю (а при ресете сохраняется его текущее значение). Чтобы его изменить, надо в режиме **View > Display** нажать `<F1>` и затем `<D>`. Появится окошко с названием-вопросом «Change delay?» и с информацией о текущем значении `dwN_delay` (также там будет информация о работе `GetTickCount` на данном ПК – сколько раз выполняется `sleep(10)` на «интервале времени 100 мс», заданном через команду `GetTickCount`; на эту информацию можно не обращать внимания). Чтобы ввести новое значение задержки, нажимаем «Да»; в появившемся окне вводим желаемое значение задержки (или 0, если хотим выключить задержку); на повторный вопрос «Change delay?» отвечаем «Нет», чтобы выйти из состояния настройки.

В реальной машине ДЗ-28 один такт длится одну-две микросекунды, а исполнение одной машинной команды занимает несколько тактов. Поэтому для оценки (очень грубой) значения `dwN_delay`, хоть как-то приближающего по порядку величины динамику имитатора к динамике реальной ДЗ-28, можно считать, что за 1 мс имитатор должен выполнять порядка 1000 циклов. Т. е. можно для начала задать `dwN_delay = 1000`, а затем скорректировать это значение; при значении 1000 у меня на стареньком ПК с Windows XP использование процессора в рассмотренном выше модифицированном «у-тесте» падает уже до 0-1%. При значении `dwN_delay` порядка 100 и менее имитатор может так сильно замедлиться, что он будет казаться «зависающим» в имитации работы с Бейсиком и с прочими программами, часто печатающими результат на экране или на табло. При увеличении `dwN_delay` свыше нескольких тысяч наличие задержки будет, наоборот, всё менее заметным.

Поскольку прога имитатора обречена пребывать в состоянии разработки, в ней предусмотрен контроль (необязательный для пользователя) частоты вывода «кадров» – параметр FPS (frames per second). Оценка FPS в проге вычисляется очень грубо – с помощью функции `GetTickCount`. Но всё-таки она помогает выяснить, как та или иная имитируемая программка распоряжается командами вывода информации. Если вывод очень редкий, то кадр будет обновляться, в основном, с частотой мигания курсора дисплея (примерно 5 раз в секунду), и значение FPS получится близким к 005. Если вывод частый, то в проге имитатора сработает ограничение частоты обновления кадра (предусмотренное,

чтобы избежать перегрева видеочипа), так что значение FPS, скорее всего, не превысит 060 ... 070. Значение FPS изображается белым шрифтом в позиции, соответствующей началу служебной строки дисплея; чтобы увидеть FPS (или, наоборот, скрыть его) надо в режиме **View > Display** нажать <F1>, затем <D>, и ответить на вопрос «Show FPS?».

В Windows 7 монитор ресурсов показывает не только процент использования процессора, но и частоту процессора в процентах от максимальной. Как выяснилось, во время вывода имитатором текстов на дисплей процент использования процессора невелик, но частота может подниматься до 90% и более, и при этом температура тоже повышается. Однако, в Windows 7 есть возможность ограничить повышение частоты и тем самым препятствовать нагреву системы при работе всех программ, не только имитатора. Для этого открываем: Панель управления > Электропитание > Настройка плана электропитания > Изменить дополнительные параметры питания > Управление питанием процессора > Максимальное состояние процессора; и вместо указанных там 100% задаём, например, 60% (при питании от сети и от батареи).

## 8. Перечень программ в txt-папке, прилагаемой к данной версии

Basic\_D3-28\_v3A\_KP-157107\_N-11343.txt и  
Basic\_D3-28\_v3A\_KP-132259\_N-9675.txt – две версии системы  
программирования на языке «Бейсик для ДЗ-28, вариант 3А».

Запуск: <C>, <S>. О выборе параметров служебной строки дисплея и другие подробности см. в разделе 5.

---

Fortran-85\_p1-p2.txt – система программирования «Fortran-85 для ДЗ-28».

Запуск: <C>, <T>, <T>. Подробности см. в разделе 5.

---

Fortran-5m\_KP-138177\_N-10414.txt – система программирования «Fortran-5М для ДЗ-28». Это вариант, очень близкий к Fortran-85, но состоящий из одного файла.

Запуск: <C>, <T>, <T>. Подробности см. в разделе 5.

---

D3-28\_Vystra\_KP-113478\_N-9478.txt – система программирования «Выстра».

Запуск: <C>, <S>. Подробности см. в разделе 5.

Описание правил работы с Фортранами и Выстрой не включено в данный архив;

( см. описание правил работы с этими программами в отдельных документах:

*conspect\_opisania\_Vystra\_\_9-07-2017.pdf*

*conspect\_opisania\_Fortran-85\_D3-28.pdf*

*opisanie\_FORTRAN-5M\_\_7-04-2017.pdf* )

---

game\_1WAR\_BAS.txt – тест-программа с игровыми элементами: для проверки взаимодействия машины, дисплея и клавиатуры; иллюстрирует некоторые приёмы применения машинных кодов в бейсик-программах пользователя.

Запуск: только под управлением Бейсика (загрузка командой LOAD, старт командой RUN).

---

bega\_5prg\_\_bas.txt – «тараканьи бега» – может использоваться для выявления особенностей вывода символов в разных режимах имитатора, в том числе для сравнительной оценки скорости вывода символов. В файле содержатся 2 записи программы «тараканьи бега». Первая запись называется BEGA-1, она сделана в Бейсике-132259. Вторая запись сделана в Бейсике-157107, она называется BEGA-2. Кроме того, в этом txt-файле содержатся бейсик-программы: SINUS – построение графика синуса, ХАОС-1 – построение графика хаотической динамики (внутри программы есть подробная инструкция), КАТАЛОГ – список программ на данной «ленте», т. е. в данной «библиотеке программ».

Запуск: только под управлением Бейсика – загрузка командой LOAD, старт командой RUN. По умолчанию подразумевается запуск в Бейсике-157107 при b4=0. Оказывается, при этом в программе BEGA-1 нормально воспроизводятся РУС-буквы. При запуске же программы BEGA-2 надо после набора RUN, но перед нажатием <Enter> нажать <Ctrl>, лишь тогда текст выведется в регистре ВР РУС (либо перезапустить Бейсик-157 в режиме с b4=1).

---

game\_luna\_\_bas.txt – «Посадка на Луну», стандартная игровая программа, набрана по листингу из документации к Бейсику.

game\_corolevstvo\_\_bas.txt – «Королевство Эйфория», игровая программа.

game\_casino\_\_bas.txt – «Казино», игровая программа.



game\_vojna\_\_bas.txt – «Война», игровая программа (сочинение студентов, впечатлённых военной кафедрой :-)

Запуск: все эти игровые программы работают только под управлением Бейсика.

---

bioritmy\_\_bas.txt – программа из «Руководства программиста» на Бейсике. Интересна тем, что в ней организовано построение одновременно трёх синусоид на экране алфавитно-цифрового дисплея. Запуск: только под управлением Бейсика – загрузка командой LOAD, старт командой RUN.

Для правильного вывода графиков на экран здесь необходимо воспользоваться Бейсиком-157107 с b4=1 (см. пояснения на стр. 22) и задать систему команд №1, сбросив в ноль разряды служебной строки дисплея нажатием клавиш <F8>, <F9>, <F8>. Тогда правильно отображается «нулевая линия» на графиках биоритмов, которая строится из символов I. Сначала печатается символ I, а затем он заменяется символом графика, если график должен пересечься с «нулевой линией». Интересно, что если бит-4 не установлен в единицу, то такой замены символов не происходит, и «нулевая линия» местами смещается направо.

---

1WAR\_1\_2WAR\_1\_\_frtrn5m.txt – файл, имитирующий «ленту» с библиотекой фортранных программ. В начале файла записан FORTRAN-5M; загрузка: <C>, <L>, <K> (КП=138177, код 0512 на шаге N=10414), запуск: <C>, <T>, <T>. Далее на этой «ленте» идут (см. стр. 30) исходные фортранные тексты программ 1WAR и 2WAR, рабочие (т. е. транслированные в машинные коды) программы 1WAR и 2WAR, причём у всех этих программ номер версии равен 1. И затем – исходные тексты SF-T версий 1 и 2. В конце есть запись «конец библиотеки».

Запуск исходных программ: только под управлением Фортрана-5M или Фортрана-85, после трансляции в машинные коды.

Запуск рабочих программ: под управлением Фортрана-5M, или запуск с пульта машины (подробности см. в разделе 5).

---

frmj\_inpx\_test\_\_1prg\_\_3prg\_vystra-text.txt – файл, имитирующий «ленту» с библиотекой выстровских программ. В начале – рабочая программа FRMJ, скомпилированная в Выстре в имитаторе. Затем – исходные тексты программ FRMJ, INPX, TEST (см. стр. 36). Запуск исходных текстов: только под управлением Выстры.

frmj\_KP-38246\_N-3445.txt – рабочая программа расчёта «фракталов» FRMJ, скомпилированная в Выстре на реальной машине ДЗ-28 в давние времена. Обе рабочих программы FRMJ служат примером автономно работающих программ, созданных в Выстре (примечание: контрольная сумма и количество шагов не обязательно получаются одинаковыми при разных попытках компиляции).

Запуск рабочих программ FRMJ: <C>, <S> на пульте машины. Загрузка Выстры не требуется.

---

display\_codes\_KP-1689\_N-106.txt – программа вывода на дисплей байт-кода, поступающего из терминала при каждом нажатии клавиши на клавиатуре терминала. Выводится символ (если клавиша символьная), и за ним байт-код. Если клавиша подаёт управляющий код, а не код символа, то выводится только этот код. Каждый код выводится в новую строку. Чтобы не было пустых строк, можно выключить в настройке служебной строки «авто ПС=ВК»:

9 6 0 0   0 0 0 0   0 0 1 0   0 0 0 0   0 0 0 0   ...

Запуск: <C>, <S> .

---

y-test\_KP-88\_N-8.txt – простейшая программка, поясняющая программирование в машинных кодах ДЗ-28; см. раздел 6, стр. 41-42.

Запуск: <C>, <S>. Смотреть на экран дисплея не нужно; программка выводит последовательные значения целых чисел на Y-табло. Для останова надо нажать <Enter> или <C>.

---

D3-28\_test\_028\_KP-60000\_N-5999.txt – стандартная тест-программа, проверяющая систему команд и ОЗУ машины ДЗ-28.

Запуск: <C>, <T>, <S>. Для работы этого теста дисплей не нужен.

Тест состоит из четырёх блоков, и в реальности должен работать непрерывно: после 4-го блока сам запускается снова 1-й блок. Но в имитаторе после 1-го блока каждый раз происходит останов, так как имитатор не воспроизводит строго ту погрешность вычислений в тесте, которая возникает в реальной ДЗ-28. При этом на табло мы видим:

Y = .383993903341-08  
X = .888000000000-08

Нижнее число (т. е.  $X$ ) это значение погрешности, которое должно накопиться в реальной ДЗ-28 по ходу многократных вычислений разных математических функций. Верхнее число – погрешность, накапливающаяся в имитаторе. Она примерно вдвое меньше той, что ожидается в тесте (потому что в имитаторе пока ещё не сделана точная имитация 12-разрядной математики реальной машины ДЗ-28), и это служит причиной остановки теста.

Но ничего страшного для нас в этом нет: просто надо после каждой такой остановки нажимать два раза  $\langle S \rangle$ . Тогда запускается продолжение теста, и дальше он идёт как надо, с автоматическим возвратом к 1-му блоку после 4-го.

Прохождение каждого из первых трех блоков индицируется на табло выводом номера блока в виде «цифр через одну»; через короткое время эти цифры меняются местами с погашенными разрядами табло. После 4-го блока теста его авторы своеобразно напоминают нам об одном историческом событии.

---

D3-28\_test\_017\_\_KP-5895\_N-527.txt – стандартная тест-программа, проверяющая работу НМЛ и пригодность магнитной ленты в «рабочей кассете».

Для работы этого теста дисплей не нужен.

Запуск:  $\langle C \rangle$ ,  $\langle T \rangle$ ,  $\langle S \rangle$ . После этого тест останавливается с индикацией числа 21 на обоих табло. Так программа напоминает нам о необходимости вынуть из НМЛ кассету с тестом и поставить чистую «рабочую кассету»! (Лучше это делать заранее, не дожидаясь напоминания).

После того как мы поставим «рабочую кассету», надо нажать  $\langle S \rangle$ .

Тест запускается и работает относительно долго. В имитаторе это проявляется как мигание НМЛ и смена цифр на табло. В итоге на табло фиксируются числа:  $Y = 1$ ,  $X = 0$ . В документации к тесту говорится, что такие числа означают полностью успешное прохождение теста.

rabochaja\_kasseta\_\_21-03-2017.txt – пустой файл, имитирующий «рабочую кассету» для теста НМЛ. Его можно использовать и для опытов с командами записи на НМЛ в других случаях.

---

statistika\_\_KP-7074\_\_N-806.txt – пакет подпрограмм «Статистика», (причём, в конце этого же txt-файла напечатаны пояснения об особенностях запуска «Статистики» в предыдущих версиях имитатора ДЗ-28). Набран по распечатке в «Руководстве оператора 3.857.100 Д12» из комплекта документации к ДЗ-28.

---

raspechatchik\_D3-28\_\_KP-935\_\_N-67.txt – программа для распечатки номеров шагов и байт-кодов программы из ОЗУ машины. (Она нужна с реальной машиной, а здесь не очень-то нужна, ведь наши программы это уже текстовые файлы.)

*Работа с распечатчиком:*

0. Включаем ТПУ (в имитаторе «ТПУ включается» ПК-клавишей <Home>).
1. Вводим в ОЗУ с ленты (или с пульта ДЗ-28 вручную, если это не слишком долго) ту программу, из которой надо распечатать часть или её всю.
2. Кнопкой КП проверяем контрольную сумму. Здесь и далее речь о кнопках на пульте ДЗ-28; их соответствие с ПК-клавишами в имитаторе указано в разделе 6.
3. Кнопкой В переходим в режим «Ввод», при этом индикатор покажет номер шага с кодом 0512 (END) программы в ОЗУ. И дописываем на следующих шагах код 0514 (это «пустая» команда GO) сколько-то раз – чтобы получился «круглый» номер шага, удобный для запоминания, с которого позже мы введём в ОЗУ машины свой распечатчик. Этот номер шага следует запомнить.
4. Кнопкой Р переходим в режим «Работа», и с пульта ДЗ-28 кнопками прямого кодирования подаём двухбайтовую команду 0413 0415; происходит засылка в регистр Х начального адреса дальнейшей загрузки. (Напомню, что в имитаторе цифры 0413 0415, поскольку они представляют собой байт-коды команды, надо набирать с постоянно нажатой ПК-клавишей <Ctrl>.)
5. Ставим магнитную ленту с распечатчиком и загружаем его в ОЗУ командой 1202 с пульта машины. (В имитаторе цифры 1202 надо набирать с постоянно нажатой ПК-клавишей <Ctrl>.)
6. Прямым кодированием заносим в регистры R0 и R1 соответственно начальный и конечный адрес того участка ОЗУ, который желаем распечатать. С этой целью поочерёдно пользуемся режимами В и Р (в имитаторе – клавишами <V> и <R>):
  - В: нажимаем НШ и вводим пять цифр (старшие из них могут быть нулями), это желаемый начальный адрес в десятичной системе; он сам собой оказывается в программном счётчике РС. (В имитаторе НШ это <N>).
  - Р: подаём двухбайтовую команду 1104 1500, это засылка РС в регистр R0 (в имитаторе цифры 1104 1500 набираем с постоянно нажатой ПК-клавишей <Ctrl> ).
  - В: нажимаем НШ и вводим пять цифр = желаемый конечный адрес.
  - Р: подаём команду 1104 1501, тем самым конечный адрес засылается в R1. (в имитаторе цифры 1104 1501 набираем с постоянно нажатой ПК-клавишей <Ctrl> ).

7. В: нажимаем НШ и вводим номер шага, на котором находится начало нашего распечатчика (этот номер мы запомнили раньше); тем самым мы подготовили стартовый адрес для запуска распечатчика.

8. Р: нажимаем клавишу S – старт; в имитаторе это клавиша <S>. Всё.

Распечатчик создаёт txt-файл, в котором левый столбец представляет десятичные значения адресов в ОЗУ с шагом 10, а в каждой полной строке содержатся 10 байт-кодов. Не следует пугаться пробелов, встречающихся в таком файле: дело в том, что имитатор, как и реальное ТПУ, не печатает нули, если они находятся в начале числа (так что, если, например, байт-код или номер состоит из одних нулей, то в ТПУ-распечатке он выглядит сплошным пробелом ).

## 9. Приложение: фотографии

Электронно-вычислительная машина (устройство специализированное управляющее вычислительное) «Электроника ДЗ-28»:



Image Copyright © Sergei Frolov, 2010  
<http://www.leningrad.su/museum/>

Изображение "машины ДЗ-28" в имитаторе не является фотографически точным. Оно построено из фрагментов фотографий, с ретушированием и с изменением пропорций – чтобы индикаторное табло выглядело покрупнее, и цифры на нём чтобы были видны подслеповатому пользователю (вроде меня) чётче.

Клавиатура дисплейного терминала «15ИЭ-00-013»:



Image Copyright © Sergei Frolov, 2009  
<http://www.leningrad.su/museum/>

В имитаторе на фото клавиатуры были немного обновлены пожелтевшие клавиши, кое-где вытерта пыль, и подрисована часть корпуса блока-логики терминала.



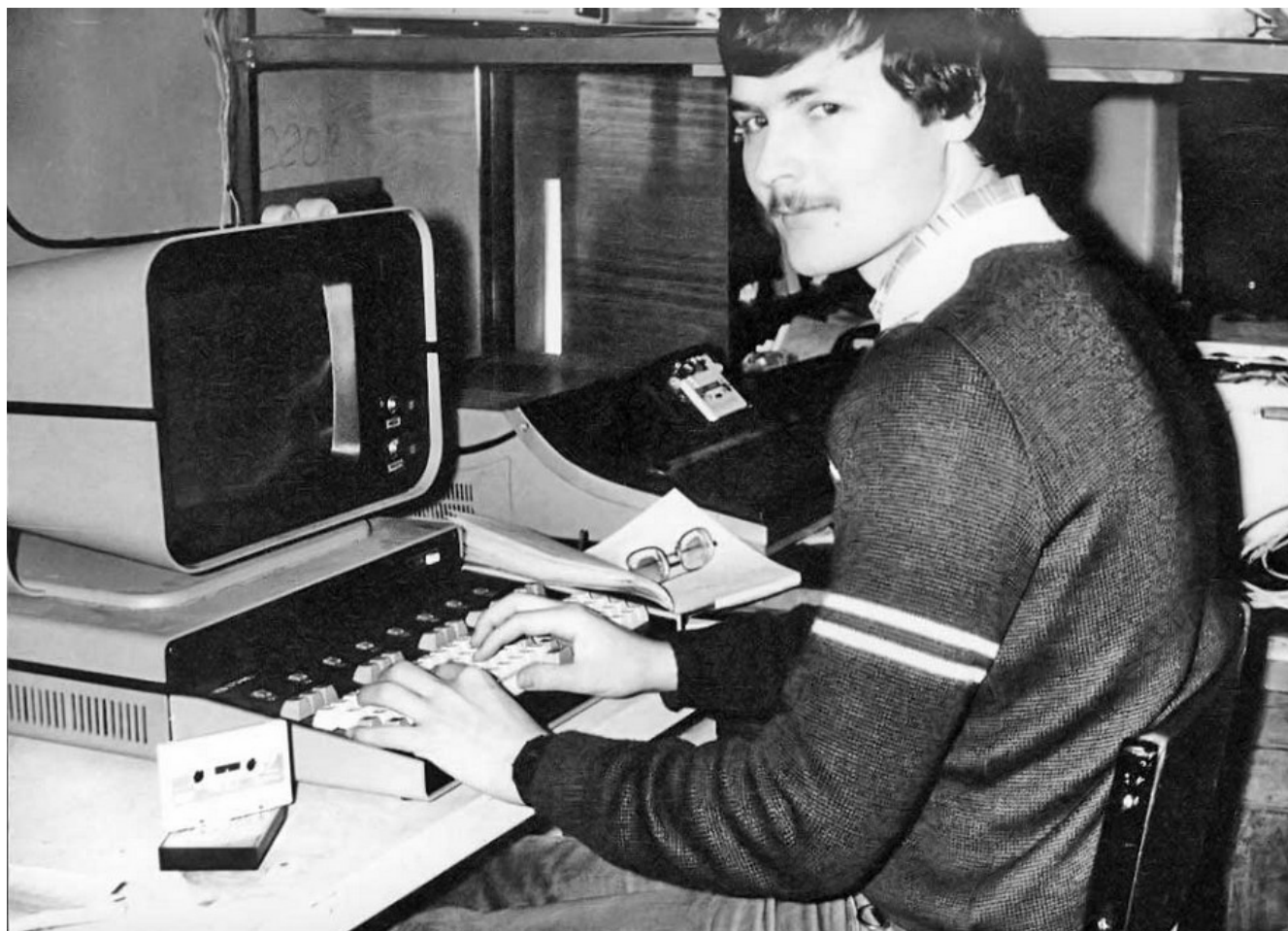
Ещё одна фотография ЭВМ «Электроника ДЗ-28»:



С этой фотографии для имитатора пригодилось изображение крышки НМЛ.

В завершение – две ностальгические фотографии из уже далёких 1980-х годов. Эти изображения не нашли применения в программе; но они передают атмосферу лаборатории политехнического вуза того времени. Обе фотографии найдены в сети в свободном доступе; авторам большое спасибо! (Однако сюда эти фотографии помещены без разрешения авторов, поэтому не сопровождаю их подробным описанием и персональной информацией.)

Вот здесь мы видим комплекс «ДЗ-28 с терминалом 15ИЭ-00-013» не на музейной витрине, а в его природной среде обитания:







А здесь надо взглянуть ещё и на самый верх фотоснимка – на полке видно (жаль, что не полностью) ТПУ 15ВВП80-002. Это единственная фотография из найденных в сети, на которой видно точно такое же ТПУ, какое мне запомнилось в те давние времена.

---

Июль, 2017 . Sinus . :-)