

Текстовый терминал VT52

Этот проект является аппаратной FPGA-реализацией текстового терминала, совместимого с терминалами DEC VT52 (и множеством его клонов), а также с 15ИЭ-00-013 (так называемый фрязинский дисплей). Его можно использовать вместо обычной связки компьютер+эмулятор терминала для работы с DEC-совместимыми ЭВМ (той же ДБК, например).

Физически для отображения информации может использоваться любой монитор, имеющий порт VGA. Картинка выводится в разрешении 640*480, 60Гц. Для ввода — обычная PS/2 клавиатура. Конечно, АТ-клавиатура сильно отличается от оригинальной клавиатуры фирмы DEC, и уж тем более - от клавиатуры 15ввв-97-006, используемой в 15ИЭ, но я постарался максимально использовать то, что есть — основная часть клавиш будет вводить именно то, что на них написано.

Поддерживаемые системы команд

Терминал поддерживает систему команд VT52, а также системы команд 1 и 2 терминала 15ИЭ (система команд 2 совпадает с VT52).

Протокол VT52 реализован в полном объеме. Поддерживается режим Hold Screen, а также набор псевдографических символов. Все клавиши оригинального VT52 отображаются на совпадающие по расположению клавиши PS/2-клавиатуры. Единственный режим, который не реализован — это управление принтером, подключенным к терминалу. В наше время этот режим не имеет практической пользы, но, при желании, легко реализуем.

Протокол 15ИЭ (система команд 1) реализован почти полностью. Часть клавиш оригинальной клавиатуры 15ввв-97-006 реализовать не удалось — все же у клавиатуры PS/2 только один полноценный дополнительный блок клавиш, а не 2. Но, при желании, любой управляющий код можно ввести сочетанием ctrl+соответствующая буква. Блочный режим (передача строки и страницы) не реализован в силу того, что применить его особо негде. Но при желании его нетрудно добавить. Также не реализован полудуплексный режим (локальное эхо) и поддержка принтера.

Переключение между протоколами можно делать вручную клавишей F4, а также автоматически с помощью управляющих кодов.

Также реализован автономный режим работы (локальная петля), когда вводимые с клавиатуры символы сразу передаются на экран, а обмен данными через интерфейс не производится. Режим включается клавишей F10. Он может быть использован для тестовых целей — проверки клавиатуры, например. Такой режим есть у всех существующих аппаратных терминалов.

Кодировка символов.

Терминал работает в 7-битном режиме. Формат кадра последовательного порта 8-битный (8-N-1), но старший бит байта игнорируется. Поддерживаются 2 набора кодировок символов — КОИ7 Н0/1, и КОИ7 Н2.

В режиме КОИ7 Н0 (режим по умолчанию) терминал работает со стандартной таблицей ASCII, в точности как оригинальный VT52. При этом отображаются малые и большие латинские буквы. Из кодировки Н0 (LAT) можно переключиться в

кодировку H1 (RUS), при этом терминал отображает малые русские буквы вместо больших латинских, и большие русские вместо малых латинских. На экране может отображаться смесь символов обоих кодировок. Так работает терминал 15ИЭ. Переключение между кодировками производится управляющими кодами 016 и 017.

В режиме КОИ7 H2 терминал отображает только большие латинские и русские буквы, при этом русские буквы занимают место малых латинских стандартной таблицы ASCII. Такой режим использовали терминалы, эксплуатировавшиеся совместно с СМ ЭВМ, например СМ7209 или ВТА-2000-3. Терминал 15ИЭ также имел такой режим работы, задававшийся через один из битов служебной строки. Переключение в этот режим и обрано в режим H0/H1 производится клавишей F11.

В режиме H2 эксплуатация оригинальных ОС от DEC невозможна, ибо все сообщения выводятся в искаженном виде (знаменитое ИНЖАЛИД ДЕЖИЦЕ). Но в этом режиме корректно будут работать адаптированные в СССР системы (РАФОС, например), а также многие написанные в те времена программы, использовавшие русский язык, к примеру, экранный редактор USED.

Неотображаемые символы из диапазона 00-37, не распознанные терминалом как управляющие команды, отображаются на экране в виде символов латинского алфавита с кодом 100-137, мигающих с частотой около 3 Гц. Эта особенность отображения присутствует в терминалах СМ7209 и 15ИЭ, но отсутствует в оригинальном VT52.

Формат экрана

Терминал работает в стандартном для всех таких терминалов разрешении 24 строки по 80 символов. Кроме этого, добавлен дополнительный режим в 38 строк по 80 символов. В таком разрешении работают все текстовые видеоконтроллеры с оренсоres, а также терминал, встроенный в проект rdp2011. В этом режиме на экран влезает больше строк за счет уменьшения межстрочных промежутков, что удобно для просмотра больших объемов текста. Но в этом режиме некоторые управляющие команды могут работать некорректно, и, кроме того, далеко не все ОС поддерживают режим с нестандартным количеством строк. Переключение в режим 38 строк и обратно производится клавишей F12.

Самая верхняя строка экрана терминала — служебная строка. Вместо маловразумительных битовых полей терминала 15ИЭ в эту строку выводятся буквенные индикаторы режимов работы красного цвета:

VT52/15ИЭ	текущая система команд терминала.
LINE	режим связи с ЭВМ.
LOCAL	автономный режим (локальная петля)
CAPS	включен CAPS LOCK.
ALT	включен альтернативный режим дополнительной клавиатуры.
L38/L24	количество отображаемых строк
КОИ7	включена кодировка КОИ7 H2
LAT	включена кодировка H0
RUS	включена кодировка H1
HOLD	включен режим HOLD SCREEN
WAIT	терминал ждет команды Scroll для продолжения вывода
MUTE	звуковой сигнал отключен.

У правого края служебной строки отображается текущая скорость последовательного интерфейса, а также время, прошедшее с момента включения дисплея (как в 15ИЭ). Вторая строка экрана — это разделительная черта синего цвета, отделяющая служебную строку от пользовательского экрана. Со строки 3 начинается собственно экран терминала. Пользовательские данные выводятся зеленым цветом. Выглядит все это так:

```
VT52 LINE CAPS L24 LAT 38400 00:04:19
000000
0B
8RKO
RT-11FB (S) 005.03
.SET TT QUIET
```

Терминал поддерживает 2 типа курсора — подчеркивание и блок размером со знакоместо. Форма курсора переключается клавишей F8. В случае блочного курсора символ, над которым он стоит, отображается инвертированным. Также курсор может быть мигающим или немигающим, этот режим переключается клавишей F7. Курсор отображается желтым цветом.

Клавиатура.

Алфавитно-цифровое поле клавиатуры используется для ввода букв, цифр и символов. Раскладка букв и символов во всех режимах примерно соответствует стандартной раскладке АТ-клавиатуры (qwerty/йцукен) с небольшими отличиями, которые легко выявить опытным путем (лень рисовать картинку с раскладкой).

Поле цифровой дополнительной клавиатуры используется в точности так же, как в оригинальном VT52 — в стандартном режиме вводятся цифры, в альтернативном — управляющие коды. Клавиши - и + соответствуют стрелкам вверх-вниз. Клавиши Numlock, /, * соответствуют кодам PF1, PF2 и PF3 стандартной клавиатуры VT52 (и расположены в тех же местах). Отсутствующие клавиши влево-вправо перенесены на поле управления курсором, там же продублированы стрелки вверх-вниз.

Управляющие клавиши АТ-клавиатуры имеют следующие значения:

F4	Переключение системы команд VT52/15ИЭ
F5	Уменьшение на одну ступень скорости последовательного интерфейса
F6	Увеличение на одну ступень скорости последовательного интерфейса
F7	Переключение формы курсора - блок/подчеркивание
F8	Включение-отключение мигания курсора
F9	Отключение-включение звукового сигнала.
F10	Переключение режимов Line (связь с ЭВМ)/Local (автономный режим)
F11	Включение/отключение кодировки КОИ7 Н2 (все большие буквы)
F12	Переключение режима 24/38 строк
L-ALT	Переключение кодировки RUS (H1) / LAT (H0)
R-ALT	Ввод кода LF (ПС) — переход на новую строку
Scrollock	Вывод следующей строки в режиме HoldScreen (с shift - вывод страницы)

В системе команд 15ИЭ также доступны следующие клавиши:

PgDn	Сдвиг текста вниз
PgUp	Сдвиг текста вверх
Home	Установка курсора в начало экрана
Ins	Размыкание строки
Del	Смыкание строки

Шрифт.

Терминал отображает знакоместа формата 8*12 (в отличие от древних терминалов, использовавших матрицу 8*8).

В схему терминала входит модуль ROM размером 4К (созданный в статической памяти FPGA) с образом шрифта знакогенератора. Исходные образы шрифта лежат в vt52-firmware/font/font-*.bin. Раскладка кодов символов внутри файла шрифта:

00-1f (000-037) большие латинские буквы (для индикации управляющих кодов)

20-3f (040-077) цифры и обычные символы

40-5f (100-137) большие латинские буквы

60-7f (140-177) малые латинские буквы

80-9f (200-237) псевдографика

a0-bf (240-277) не используется

c0-df (300-337) малые русские буквы

e0-ff (340-377) большие русские буквы

К проекту приложены 2 шрифтовых файла.

font-main.bin — основной шрифт 8*12, в основном выдраный из какого-то древнего досовского русификатора. Большинство линий символов имеет толщину в 2 пикселя, и шрифт смотрится вполне достойно на современных TFT-мониторах.

font-ksm.bin — шрифт 8*8, выдраный из знакогенератора платы КСМ (терминал 15ИЭ использует такой же шрифт). При использовании этого шрифта экран выглядит в точности таким же, как экран терминала 15ИЭ. Этот шрифт я приложил для ностальгирующих по старым временам, а также для наглядного сравнения корявых древних шрифтов с современными. Чтобы переключить терминал для работы с этим шрифтом, надо подключить файл font-ksm.mif к мегафункции fontrom в качестве файла инициализации памяти.

Также в каталоге vt52-firmware/font лежат некоторые утилиты для обработки шрифтовых файлов:

font2mif — преобразователь файл шрифта (bin) в формат mif для загрузки в FPGA.

fontlist — выводит на экран изображения всех шрифтов указанного bin-файла

fontextract — извлекает шрифт указанного или всех символов из bin-файла

fontreplace — заменяет указанный символ в bin-файле.

Извлеченный из двоичного файла образ символа представляет собой текстовый прямоугольник размеров 8*12 знаков, в котором пикселя, формирующие изображение, обозначаются символом О, остальные — точкой:

```
..000...  
.00.00..  
00...00.  
00...00.  
00...00.  
0000000.  
00...00.  
00...00.  
00...00.  
.....  
.....
```

Такая форма удобна для редактирования изображения символа. 2 нижние строки лучше не использовать для размещения изображения, поскольку на этих строках размещается курсор в режиме подчеркивания.

Аппаратная реализация

Терминал разрабатывался на fpga серии Cyclone 4 фирмы Altera (теперь, к сожалению, Intel). Он занимает около 2000 логических ячеек и 21 Кбайт внутренней памяти FPGA. Поэтому для его сборки можно использовать самую младшую микросхему серии Cyclone 4 — EP4CE6. Кроме собственно FPGA минимально нужен кварцевый генератор тактовой частоты на 50 МГц, разъем VGA, разъем PS/2, а также преобразователь 3.3v TTL-уровней сигналов последовательного порта Rx и Tx для согласования с оборудованием, к которому подключается терминал — обычно это или RS232, или токовая петля 20ма.

Самый простой способ запустить терминал — это купить простейшую китайскую отладочную плату на EP4CE6 (их навалом на aliexpress за копейки). Наличие на плате DRAM, LED дисплея, IR-порта и прочей периферии не требуется — достаточно наличия самой микросхемы FPGA, флешки конфигурационной памяти и тактового генератора. Большинство плат имеют тактовую частоту 50 МГц, и именно на эту частоту рассчитан проект. Если частота другая — плату тоже можно использовать, но придется подключить мегафункцию PLL для получения 50 МГц.

Еще проще купить плату, на которой уже есть VGA, PS/2, RS-232 — такая плата стоит подороже, но паять вообще ничего не придется, достаточно в Assignment Editor квартуса сопоставить порты с соответствующими выводами fpga согласно спецификации платы, собрать прошивку, и залить ее в конфигурационную память платы.

Проект состоит из 2 частей: модуля vt52, реализующего все функции терминала, а также корневого модуля-оболочки terminal, предназначенного для согласования терминального модуля с конкретной платой. Оболочка содержит в себе формирователь сигнала начального сброса, а также формирователь цветовых сигналов VGA.

Поскольку проект не поддерживает управление яркостью, то из модуля vt52 выходят однобитовые сигналы vgared, vgagreen, vgabluе. Их можно прямо подавать на ножки разъема VGA (через резистор около 300 ом). Если же на плате установлен ЦАП для формирования цветов разной яркости, то однобитовые сигналы надо преобразовать в

константу нужной разрядности, значение которой, подаваемое на вход ЦАП, определяет яркость конкретного цвета. Пример такого преобразователя имеется в модуле `terminal.v`.

Терминальный модуль умеет управлять звуковым сигналом через порт `buzzer`. Сигнал принимает значение 1, когда надо включить звук. Его можно прямо подавать на пьезопищалку со встроенным генератором звука. Либо создать в модуле `terminal` таймер/делитель на частоту около 400 Гц и использовать его для генерации звука. Или вообще обойтись без звука, не особо он и нужен.

Кроме создания законченного терминала, модуль `vt52` можно использовать для встраивания в другие проекты, которым нужен терминал. Например, у меня он встроен в FPGA-версию платы MC1201.02, при этом весь проект вполне помещается все в ту же EP4CE6. Получилась полноценная одноплатная ЭВМ ДВК-3. Для облегчения согласования скоростей последовательных портов из модуля `vt52` выходит наружу шина `vtSpeed`, на которую модуль выводит индекс текущей скорости своего последовательного порта (от 000 — 1200 до 111 — 115200). Эту информацию можно использовать для настройки скорости внешнего контроллера UART, сопряженного с терминалом.

Терминал построен на основе процессора KP1801BM2 (используется Wishbone вариант от уважаемого Vslav). В конфигурацию входит модуль статической RAM размеров 8K (созданный с помощью мегафункции `altsyncram`), в котором размещена управляющая программа, исходный текст которой лежит в `vt52-firmware/terminal.masm`. Там же размещен кроссассемблер `masco11` (из пакета поддержки эмулятора `simh`, насколько я помню), а также написанный мной преобразователь объектных модулей RT-11 в абсолютный бинарный файл (такой типа компоновщик). Компиляцию программы можно осуществить, используя скрипт `compile.sh`. Результатом работы скрипта будет файл формата `mif`, пригодный для загрузки в память FPGA.

Поскольку проект не использует каких-либо аппаратных особенностей серии Cyclone 4, его при желании можно портировать на многие другие типы FPGA, даже необязательно фирмы Altera. Единственной проблемой являются 2 мегафункции `altsyncram`, формирующих ОЗУ программ и ПЗУ знакогенератора. Их придется делать под конкретную микросхему.