

Квалифицированная русификация игровых программ на примере игры PYJAMARAMA

© Михаил Судаков, г. Воркута, 1994 г.

Прежде всего, я хотел бы поточнее объяснить, что такое "полная русификация". Все любители аркадных адвентюр (да и не только они), наверное, играли в "SCEPTRE OF BAGDAD" или, в русифицированной версии: "ЖЕЗЛ БАГДАДА". Так вот, это - далеко не полная русификация. При всем моем уважении к автору перевода (а ведь именно "Жезл Багдада" подтолкнул меня переключиться с русификаций игр типа "FIREBIRDS" к более сложным программам), я не могу без содрогания смотреть на явно нерусские надписи типа "AIR BAGDAD" или русские, но сокращенные или разбитые на части.

Девиз полной русификации: "Ни одного нерусского слова!" (понятие "нерусский", как Вы понимаете, включает в себя много чего). С Божьей помощью я разобрался с русификацией и постараюсь показать, как это делается, Вам. Почему я выбрал для этой цели программу PYJAMARAMA? Очень просто - она достаточно сложна и одновременно достаточно проста. Сложна - по сравнению с той же FIREBIRDS, а проста - по сравнению, например, с DIZZY (особенно с последними частями этой игры). Сразу разбирать сложную программу я не буду по двум причинам: 1-я - лучше все-таки начинать с простого, а 2-я - с такими программами я и сам-то ещё не полностью разобрался. Итак, приступим.

1. Сначала давайте-ка определимся, с какой программой мы будем работать. Не спешите смотреть в заглавие статьи и протирать глаза - у меня имеется три версии игры PYJAMARAMA: первая - простая, абсолютно незащищенная и открытая для всех; вторая - закоренная (термин "ксорка" я взял у Н. Родионова, и он обозначает, что программа закодирована оператором XOR - для тех, кто не знает); третья - незащищенная, но с измененной музыкой и испорченной графикой. Наиболее распространена все-таки 2-я версия, и именно её я возьму за основу. Владельцы первой версии могут пропустить строки, посвященные "расксориванию", а владельцам третьей версии я могу посоветовать только одно - выкиньте её куда подальше и приобретите 1-ю или 2-ю. Да, кстати, как отличить 1-ю версию от 2-й - в загрузчике 2-й версии после всех LOAD-ов должна стоять команда: RANDOMIZE USR 64649 или DRAW USR 23760,USR 64649, или что-то в этом роде, а в 1-й - скорее всего RANDOMIZE USR 33280.

2. Для того, чтобы можно было нормально работать с версией 2, её необходимо преобразовать в версию 1, чем мы сейчас и займемся. PYJAMARAMA имеет 3 файла: BASIC, SCREEN\$ и CODE. Файл CODE имеет адрес загрузки 33280 и длину 31392. Загрузите его в память командой LOAD "<имя>" CODE (я не привожу дисковой версии по простой причине - экономия места и времени, но, так как у меня самого дисковод, то все же напомним, что перед каждым LOAD или SAVE те же счастливики, что и я, должны ставить "RANDOMIZE USR 15619: REM:"). Теперь давайте посмотрим, что же содержится по адресу 64649 (кто-то может смотреть в журнал, а кто-то - на экран, если загрузит MONS, Mon2 или ещё какой-нибудь монитор по адресу 33280 минус <длина монитора> или ниже):

64649	LD	HL, 46319	INC	DE
	LD	DE, 16384	DEC	BC
	LD	BC, 6912	LD	A, B
64658	LD	A, (DE)	OR	C
	XOR	(HL)	JR	NZ, 64658
	LD	(HL), A	JP	48874
	INC	HL		

Как видите, здесь декодирование области памяти с 46319 по 62702 происходит с использованием заставки, а затем управление передается команде по адресу 48874. Посмотрим, что там у нас:

```
48874      LD      SP,0      ;стек пользователя размещается по адресу 0
          CALL    48844      ;вызов подпрограммы по адресу 48844
48880      JP      33280      ;переход по адресу 33280
```

Подпрограмма 48844 выглядит следующим образом:

```
48844      DI
          LD      BC,15766    ;декодируется область памяти
          LD      HL,48883    ;начиная с адреса 48883 и
          CALL    48860      ;кончая адресом 64648
          LD      BC,15564    ;
          LD      HL,33280    ;
48860      XOR     A          ;
          LD      R,A          ;то же происходит и с областью памяти
48863      LD      A,R          ;с 33280 по 48843
          XOR     (HL)        ;
          LD      (HL),A      ; подпрограмма декодирования использует
          DEC     BC          ; регистр регенерации R
          INC     HL          ;
          LD      A,B          ;
          OR      C           ;
          JR      NZ,48863    ;
          RET              ; возврат из подпрограммы
```

Вот, в принципе, и вся процедура декодирования. Чтобы после её запуска вернуться в Бейсик, надо по адресу 48880 поставить команду RET (код 201). Кто уже сделал POKE 48881,201 не спешите давать RANDOMIZE USR 64649, сначала сделайте POKE 48874,0 для отключения команды LD SP,0 которая переносит стек в ПЗУ, откуда Вы уж точно не вернётесь в Бейсик. Ещё понадобится POKE 48844,251 - для замены команды DI (запрет прерываний) командой EI (разрешение прерываний). Но и теперь не спешите запускать декодирующую процедуру. Нельзя забывать, что она использует заставку, поэтому лучше наберите и запустите небольшую программу:

```
10 CLEAR 33279
20 LOAD "pijam.$" CODE 16384
30 LOAD "pyjam.c" CODE 33280: REM предполагается, что у Вас есть два
  файла с такими именами
40 POKE 48881,201:POKE 48874,0: POKE 48880,251
50 RANDOMIZE USR 64649: SAVE "pyjama.c" CODE 33280,31392
60 NEW
```

А ещё наберите такую программу:

```
10 PAPER NOT PI: INK NOT PI: BORDER NOT PI: POKE VAL "23624",NOT PI:
  CLEAR VAL "33279"
20 LOAD "pyjama.$" CODE VAL "16384"
30 LOAD "pyjama.c" CODE
40 RANDOMIZE USR VAL "33280"
```

Здесь, надеюсь, всё понятно. Вторую программу сохраните где-нибудь на видном месте, так как это - готовый загрузчик для Вашей русифицированной "PYJAMARAMA".

3. Теперь приступим собственно к русификации. Тут я сделаю небольшое отступление: все, что я буду излагать и объяснять, понадобится только тем, кто хочет пойти в русификации подальше PYJAMARAMA. Они могут просто читать журнал и пытаться всё понять мысленно, а могут загрузить монитор-отладчик и работать вместе со мной. Во втором случае, Вам придётся неоднократно сохранять кодовую часть, а также бейсик-программу для русификации на внешнем носителе (впрочем, бейсик-программа почти необязательна - можно всё изменять с помощью монитора-отладчика - так даже интереснее). Поэтому я не буду через каждое слово вставлять: "А теперь, если вы ещё не загрузили кодовую часть после перебора в сети, случайного нажатия RESET или ещё чего-

нибудь, то сделайте это командой LOAD "pyjama32" CODE, а владельцы дисковода - командой RANDOMIZE USR 15619: REM: LOAD "pyjama32" CODE ". Также я не буду учить Вас, как набрать токен "GO TO" в середине строки и прочим мелочам - об этом столько раз писали на страницах ZX-РЕВЮ, что я просто боюсь кого-либо оскорбить. Тем, кому просто хочется поиграть в русскую РYJAMARAMA, я советую набрать бейсик-программу "rus", приведенную в конце статьи, запустить её, сохранить кодовую часть и т.д. (в общем, смотрите в конце статьи).

Итак, начали. Самое первое, что мы должны сделать, это изменить английский набор символов на русский. Для этого пригодится моя небольшая программа "prosm":

```

9900 LET adr=15616: LET st=8: LET h=1: LET w=32
9901 LET s$=" !""#%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ
[\]^_`abcdefghijklmnopqrstuvwxyz{|}~@"
9902 LET n=1: GO SUB 9921: PRINT AT 0,10;adr: RANDOMIZE adr: POKE
23606,PEEK 23670: POKE 23607,PEEK 23671: FOR m=1 TO h: PRINT INK 1;s$(n
TO n+w-1): LET n=n+w: NEXT m: PAUSE 0: IF INKEY$="9" THEN LET adr=adr-
st: GO TO 9902
9903 IF INKEY$="1" THEN GO SUB 9921: INPUT "Step? ";st: GO TO 9902
9904 IF INKEY$="3" THEN GO SUB 9913: GO TO 9902
9905 IF INKEY$="2" THEN GO SUB 9921: INPUT "Adress? ";adr: GO TO 9902
9906 IF INKEY$="5" OR INKEY$=CHR$ 8 THEN LET w=w-1: GO SUB 9915: GO TO
9902
9907 IF INKEY$="6" OR INKEY$=CHR$ 10 THEN LET h=h+1: GO SUB 9915: GO TO
9902
9908 IF INKEY$="7" OR INKEY$=CHR$ 11 THEN LET h=h-1: GO SUB 9915: GO TO
9902
9909 IF INKEY$="8" OR INKEY$=CHR$ 9 THEN LET w=w+1: GO SUB 9915: GO TO
9902
9910 IF INKEY$="4" THEN GO SUB 9921: INPUT "Byte? ";byte: POKE adr,byte:
GO TO 9901
9911 IF INKEY$<>" STOP " THEN LET adr=adr+st: GO TO 9902
9912 GO SUB 9921: STOP
9913 GO SUB 9921: FOR k=0 TO 7: FOR l=167 TO 160 STEP -1: IF POINT
(k,l)=1 THEN PRINT AT 181-l,k; INK 5; PAPER 0;".": NEXT l: NEXT k:
RETURN
9914 PRINT AT 181-l,k; PAPER 5; INK 0;".": NEXT l: NEXT k: RETURN
9915 CLS : IF w*h>96 THEN LET h=h-1: RETURN
9916 IF w=0 THEN LET w=1
9917 IF w=33 THEN LET w=32
9918 IF h=0 THEN LET h=1
9919 IF h=22 THEN LET h=21
9920 RETURN
9921 POKE 23606,0: POKE 23607,60: RETURN

```

Если вы её уже набрали, то будет нелишним её сохранить - пригодится для последующих деяний. А перед её загрузкой не забывайте ставить CLEAR 26000. Теперь о том, как работает программа. Она начинается со строки 9900, чтобы не мешать программе, которая может уже находиться в памяти (в этом случае, естественно, адрес в CLEAR должен быть соответственно увеличен). Запустив её командой GOTO 9900, вы увидите вверху число - это текущий адрес, а под ним - строка символов - это окно отображения символов. В начальном состоянии число равно 15616 - это, как вам известно, адрес расположения в ПЗУ стандартного набора. Используя клавиши управления, вы можете менять этот адрес, изменять величину окна и даже вносить изменения в память. Вот эти клавиши:

"STOP" (SS + "A")	- выход в бейсик с восстановлением CHARS
"1"	- ввод значения шага с запросом "Step?"
"2"	- ввод значения адреса с запросом "Adress?"
"3"	- показ левого верхнего символа в увеличенном виде
"4"	- ввод байта по текущему адресу с запросом "Byte?"
"5" или CS + "5"	- уменьшить окно символов по оси X
"6" или CS + "6"	- увеличить окно символов по оси Y
"7" или CS + "7"	- уменьшить окно символов по оси Y
"8" или CS + "8"	- увеличить окно символов по оси X
"9"	- уменьшить адрес на значение шага
Любая другая клавиша	- увеличить адрес на значение шага.

Удобна следующая последовательность действий: установите адрес равным примерно 25000, шаг равным 256 и просматривайте память, пока не обнаружите что-нибудь, похожее на символы. Затем сделайте шаг равным 1 и чуть-чуть увеличьте или уменьшите адрес, чтобы символы не были "разорванными". А теперь установите шаг равным 8 и сделайте, чтобы в левом верхнем углу стоял символ ПРОБЕЛ, за ним - "!" и т.д. В случае "PYJAMARAMA" адрес будет равен 49803. Набор должен выглядеть так:

!"#\$% '()*+,-./0123456789:;<=>? ABCDEFGHIJKLMNOPQRSTUVWXYZ

Выгрузите символьный набор командой: SAVE "pyj.fnt" CODE 49803,768 (заметьте, что длина обязательно должна быть равна 768, потому что редактировать символы Вы будете, скорее всего, в ART STUDIO). Теперь запускайте ART STUDIO и редактируйте символы таким образом, чтобы приведенная выше строка выглядела так:

!"#Щ%Э'()*Ё+,-.Ш0123456789:;Ч=Ъ?ЮАБЦДЕФГХИЙКЛМНОПЯРСТУЖВЫЗ

Только ни в коем случае не изменяйте ту белиберду, что находится после символов. Теперь сохраняйте набор под именем "pyjr.fnt" и выходите из ART STUDIO. Опять загружайте файл "pyjama.c" (я уже говорил, как это сделать), затем загружайте набор командой LOAD "pyjr.fnt" CODE 49803 и сохраняйте файл командой SAVE "pyjama.c" CODE 33280,31392 (эту операцию Вам тоже желательно делать почаще - во избежание нежелательных последствий, а владельцам дисководов неврдно будет перед сохранением сделать ERASE "pyjama.c" CODE). Вот у Вас уже и есть PYJAMARAMA с русскими буквами, но все ещё с нерусскими словами, чем мы сейчас и займёмся.

4. Самые первые слова, которые мы будем переводить на русский - это названия предметов. В памяти они начинаются с адреса 57960. Посмотрев внимательнее, мы увидим, что каждое название завершается кодом 255, что уже наводит на мысль о kernale, то есть таблице, в которой содержатся адреса каждого из предметов. Поискав в памяти упоминание об адресе первого предмета - 57960 (или двух байтов - 104 и 226), находим его по адресу 58489. Проверяем два следующих байта - они равны соответственно 120 и 226 и представляют собой число 57976, а это - адрес второго предмета. Итак, наша догадка подтвердилась - начиная с адреса 58489, каждые два последующих байта определяют адрес соответствующего предмета, то есть по адресу 58493 будет содержаться информация о расположении 3-го. Таким образом, адрес названия n-ного предмета можно вычислить по формуле:

РЕЕК (58489+2*(n-1))+256*РЕЕК (58490+2*(n-1))

что программа, в принципе, и делает. Но нам это не нужно. Кстати, Вы заметили, что между двумя словами в таких названиях, как "THE ROUND KEY" или "THE DOOR HANDLE" стоят три байта - 252, ещё какой-нибудь байт и 1. Байт 252 означает своего рода команду, за которой должны стоять два аргумента. В подробности того, что они означают, мы вдаваться тоже не будем, хотя надо заметить, что байт, следующий за байтом 255, изменяет положение второго слова относительно первого (за это отвечает и второй байт, но немного по-другому). Тут все просто - уменьшая первый байт после байта 252, второе слово (например, "HANDLE" в названии "THE DOOR HANDLE") мы смещаем соответственно влево на один символ. Увеличивая - смещаем вправо. Можно, правда, и

проще - вычитаем длину первого слова из числа 256 и получаем то, что нужно. То есть, в названии "КРУГЛЫЙ КЛЮЧ" слово "КРУГЛЫЙ" имеет длину 7, но $256-7=249$, следовательно, память, начиная с адреса 57960, должна выглядеть так:

КРУГЛЫЙ (252) (249) (1) КЛЮЧ (255)

(здесь числа в скобках представляют собой соответствующие байты). Теперь обратимся к программе "rus" (приведенной полностью в конце статьи). Чтобы долго не мучиться, предлагаю Вам набрать и запустить пока что строки с 200 по 320 (и дальше ещё много будет таких фрагментов, так что можете их и не запускать, но сохранить советую). Вы можете изменять названия предметов, как хотите (в разумных пределах, конечно), если последуете следующим советам: длинные названия разбивайте на две части, а после первого слова ставьте пробел. Например, название "ДВЕРНАЯ РУЧКА" разбиваем на "ДВЕРНАЯ" и "РУЧКА", а в листинге это будет выглядеть, как "ДВЕРНАЯ_", "РУЧКА". После названия предметов из одного слова пробела ставить не нужно ("МАГНИТ"). Если Вы изменяете название предметов "THE WATER BUCKET", "THE LASER GUN" или "THE FUEL CAN", после которых на экране должно стоять слово "FULL" или "EMPTY", то ставьте пробел и после второго слова тоже. Так, название "ВОДА В ВЕДРЕ" в листинге будет выглядеть как "ВОДА_V_", "ВЕДРЕ_". Здесь у Вас наверняка появится вопрос, почему именно "ВОДА В ВЕДРЕ" или "ТОПЛИВО В КАНИСТРЕ", а не "ВЕДРО ВОДЫ" или, соответственно, "КАНИСТРА ТОПЛИВА"? Дело в том, что в этом случае после английских слов "THE WATER BUCKET" ("ВЕДРО ВОДЫ") стоит слово "EMPTY" ("ПУСТОЙ") или "FULL" ("ПОЛНЫЙ"), но ведь в английском языке прилагательные не имеют рода, а в русском - имеют. Как вам нравится: "ВЕДРО ВОДЫ ПОЛНЫЙ"? Это, скорее, не русский, а грузинский. Или "КАНИСТРА ТОПЛИВА ПОЛНОЕ"? Но это ещё полбеды (в крайнем случае, можно было бы обойтись сокращениями типа "ПОЛН."), а как можно допустить словосочетание "ЛАЗЕРНЫЙ ПИСТОЛЕТ ПОЛНЫЙ"?! А заменив "ПОЛНЫЙ" на "ЕСТЬ" и "ПУСТОЙ" на "НЕТ", да сделав соответствующие изменения в названиях предметов, мы получим вполне приличные словосочетания "ВОДА В ВЕДРЕ ЕСТЬ" или "ЭНЕРГИЯ В БЛАСТЕРЕ ЕСТЬ" (на экране это, правда, получше смотрится, чем на бумаге). Осталось только объяснить, что значит РОКЕ 58083,246 и 58177,246. Просто-напросто "КАНИСТРЕ ЕСТЬ" (это кусок из "ТОПЛИВО В КАНИСТРЕ ЕСТЬ") никак не умещается на экране, поэтому слово "КАНИСТРЕ" пришлось сдвинуть влево (до команды РОКЕ по адресу 58083 находилось число 247 - это тот самый первый байт после команды 252). Вы, конечно, можете немного подумать и поставить вместо "КАНИСТРЕ" какое-нибудь другое слово, но ведь моя задача - показать Вам приёмы русификации, а не учить русскому языку. Та же история и с "ЭНЕРГИЯ В БЛАСТЕРЕ".

Итак, с предметами покончено. Правда, не радуют взгляд надписи "ЕМПТЫ" и "ФУЛЛ", но с этим мы ещё разберёмся.

Теперь перейдём к тем словам и сообщениям, которые можно изменить, не прилагая особых усилий. Просмотрев память, Вы найдете их несколько, это:

"TEA" - по адресу 56502
 "LIFE", "SNOOZE ENERGY" - 33394
 "CHANGE" - 56191
 "PRESS A KEY TO CONTINUE" - 45240
 "PRESS KEY FOR LEFT" - 46171
 "PRESS KEY FOR RIGHT" - 46192
 "PRESS KEY FOR JUMP" - 46214
 "FULL" - 58806
 "EMPTY" - 58811
 "HELP" - 54106

(Слово "LIFT" переводить не надо, так как лифт он и по-английски лифт, а слово "HELP" я заменяю на "ЯЩИК", потому что помощь-то в игре и есть - появление ящика, и надпись "ЯЩИК ЕСТЬ" будет неплохо смотреться).

Перевести их на русский язык несложно, для этого наберите ещё один фрагмент программы "rus" - строки с 300 по 340.

Здесь тоже все должно быть понятно. Я лишь объясню насчёт "LIFE" и "SNOOZE ENERGY". Я поставил их вместе, так как они соединяются некоторыми командами (известной нам 252 и командой 254 - она управляет цветовыми атрибутами надписей и имеет один операнд). Наша задача по переводу этих слов усложняется необходимостью расчёта байтов после 252 - дабы слова "ВАША ЭНЕРГИЯ" вдруг не появились в центре экрана. Хотите - посчитайте (все опять сводится к увеличению и уменьшению злополучных байтов). А не хотите - смотрите во фрагмент, приведенный выше. Да, насчёт фрагмента - не удивляйтесь, если компьютер остановится с сообщением "Out of data" - впереди ещё много строк с DATA.

Теперь неплохо бы перевести на русский такие большие сообщения, как меню ("KEY:- 1.SINCLAIR..."), сообщение о поражении ("YOU DIPSTICK!..."), о результатах игры ("WALLY HAS WALKED...") и о победе ("CONGRATULATIONS!..."). Сначала займёмся меню. Все слова в нем, начиная от "KEY..." и кончая "PRESS ENTER ...", взаимосвязаны, ибо составляют одно сообщение. Его-то нам и нужно подредактировать. Чтобы хватило места в памяти на полноценную надпись "НАЖМИТЕ 'ВВОД' ДЛЯ СТАРТА", придется отказаться от соблазнительной замены:

"KEY:- 1.SINCLAIR_JOYSTICK" на "1.СИНКЛЕР_ДЖОЙСТИК"

и им подобных. Придётся опять изменять байты после 252. Меню находится по адресу 46265 и выглядит так:

```
(254) (68) KEY: 1.SINCLAIR_JOYSTICK
(252) (229) (5) KEY:- 2.KEMPSTON_JOYSTICK
(252) (229) (5) KEY:- 3.DEFINE_KEYS
(254) (70) (252) (236) (5) LEFT=(252) (12) (0) RIGHT=
(252) (239) (3) JUMP/FIRE=
(252) (242) (4) (254) (7) PRESS_ENTER_TO_START (255)
```

(что означают числа в скобках, я уже говорил). Чтобы слово "1.СИНКЛЕР ДЖОЙСТИК" находилось в центре строки, надо изменить начальную координату сообщения по оси X. Но эта координата в самом сообщении не задаётся, следовательно, надо поискать упоминание об адресе 46265 в памяти. Задайте вашему монитору-отладчику функцию поиска двух байтов 185 и 180 ($185+256*180=46265$) и наткнетесь на адрес 46502. Уменьшаем указатель адреса на 1 и смотрим, что мы обнаружили:

```
46501      LD      HL, 46265 ;адрес сообщения
46504      LD      DE, #0300 ;координаты начала сообщения
46507      CALL   57743      ;вызов процедуры печати
```

Число #0300 приведено в шестнадцатиричной системе, так как она удобнее для отображения обоих байтов - 3 и 0. Первый засылается в регистр D и отвечает, по-видимому, за X-координату сообщения, а второй - в регистр E и отвечает за Y-координату. Итак, увеличив байт 3 на 4, мы сместим "1.СИНКЛЕР ДЖОЙСТИК" на 4 символа вправо и надпись будет точно посередине, чего мы и добивались. То есть, необходимый эффект достигается РОКЕ 46505,7 - правда, во фрагменте, приведённом ниже, это сделано немного по-другому - DATA 46505,CHR\$ 7 (дело в том, что когда компьютер остановится с "Out of data" - если Вы решили сделать пробный запуск программы, то до оператора РОКЕ он попросту не дойдёт). Можно было бы, конечно, поставить РОКЕ в другом месте, но вариант с DATA выглядит даже изящнее. Дальнейшие же манипуляции с сообщением представляют собой изменения первых байтов после 252 (смотри строки 350 и 360 из программы "rus").

Теперь - сообщение о поражении. Его первые три строки перевести достаточно легко - не надо ничего изменять, кроме букв, и даже можно рассматривать эти строки, как отдельные сообщения, каждое из которых имеет свой адрес - так удобнее. В четырех же нижних единственная проблема - последняя строка. По-английски она выглядит как "UP THIS TIME.", а по-русски - "УОЛЛИ ВОВРЕМЯ.", что на один символ длиннее, чем английская версия. Это тоже обходится очень легко - первая строка из этих четырёх ("IT'S ALL YOUR FAULT.") длиннее, чем русский вариант ("ВЫ В

ЭТОМ ВИНОВАТЫ!") опять же на один символ, который мы и используем для последней строки. Так как в нашем варианте строка "ВЫ В ЭТОМ ВИНОВАТЫ!" короче на один символ, чем в исходном, то для того, чтобы все остальное осталось на своих местах, надо байт после 252 увеличить на 1, сделав это, мы получим число 237, которое и присутствует в листинге (смотри строки 370 и 380 из программы "rus").

Сообщение о победе располагается по адресу 45072 и выглядит примерно так:

```
C O N G R A T U L A T I O N S !
(252) (225) (5) (254) (67) WALLY'S WOKEN UP. FOR THE FIRST
(252) (225) (4) TIME IN HIS LIFE HE WILL BE
(252) (225) (4) EARLY FOR WORK.
(252) (248) (5) (254) (68) NOW WATCH OUT FOR
(252) (240) (4) (254) (69) 'LIFE OF WALLY' (255)
```

Единственная проблема, которая здесь есть, это - как перевести "NOW WATCH OUT FOR 'LIFE OF WALLY'"? Я перевел на русский как "ТЕПЕРЬ СМОТРИТЕ 'ЖИЗНЬ УОЛЛИ'", но что-то сомнения одолевают. А в остальном - смотрите на листинг программы "rus", строки 390 и 400.

Теперь перейдём к сообщению о результатах игры. Оно начинается с адреса 45510 и выглядит следующим образом:

```
(254) (67) WALLY HAS WALKED (252) (249) (5) PACES
(252) (242) (5) AND HAS SOLVED (252) (249) (4) (254) (71) %
(254) (67) (252) (246) (4) OF THE ADVENTURE (255)
```

"WALLY HAS WALKED" переводится как "УОЛЛИ ПРОШЕЛ", но в программе я поставил "ШАГИ УОЛЛИ:", а сделал это потому, что в русском языке, в отличие от английского, есть падежи, и словосочетание "УОЛЛИ ПРОШЕЛ 31 ШАГОВ" будет выглядеть нелепо. По этой же причине я вообще убрал слово "PACES", и теперь вы увидите на экране такую картину: "ШАГИ УОЛЛИ: 31". Остальное - смотрите в строке 410 из программы "rus".

И, наконец, последние сообщения, которые ещё остались, это: "MAGNETIC LOCK", "SMASH GLASS", "FUEL", "ON" (для лифта) и "ON" (для ящика). Рассматриваю я их в конце по той причине, что они слишком малы, чтобы заменить их русскими переводами. "Нет проблем", - скажете Вы, - "надо всего-лишь поместить их в другое место". Все проблемы заключаются в том, что, например, короткое на экране слово "ON" является не отдельным словом, а частью большого сообщения, которое в основном состоит из графических объектов - стул, кровать, лампа и т.д. Поищите команду 255 после слова "ON" - не близко, правда? Даже если после слова "GLASS" и стоит 255, то начинается-то сообщение задолго до слова "SMASH". Поэтому мы не будем что-то куда-то переносить, сделаем так, чтобы программа стала понимать ещё одну команду - 255,255 (все остальные команды уже забиты). Команда, как видите, двойная, то есть, когда компьютер встретит команду 255, то проверит, стоит ли после неё ещё одно число 255. Если нет, то это будет воспринято как обычный конец сообщения, а если стоит, то считает два байта, которые будут находиться после второго числа 255, и эти байты укажут ему, с какого адреса будет начинаться продолжение сообщения. То есть, встретив последовательность цифр 255, 255, 3, 100, программа изменит адрес сообщения на 25603 ($25603 = 3 + 256 * 100$). Сделать это не так уж и трудно. Посмотрите, как выглядит процедура печати сообщений в программе:

```
57743      LD      A, (HL)      ;Если код символа по адресу HL
          INC     HL           ;меньше 128, то переход
          BIT     7,A          ;на адрес 57802
          JR      NZ,57802     ;(печать простого символа).
57749      CP      255         ;Если код равен 255, то это - конец
          RET     Z            ;сообщения (выход из процедуры печати).
57752      CP      254         ;Если код не равен 254, то переход на
          JR      NZ,57760     ;адрес 57760.
          .....

```

Что там находится дальше, нам знать не обязательно. Свою дополнительную процедуру расположим по адресу 64649 (там раньше находилась одна из процедур декодирования), а чтобы программа обращалась к ней, заменим команды CP 255 и RET Z по адресу 57749 на одну команду JP 64649, что можно сделать тремя операторами POKE: POKE 57749,195: POKE 57750,137: POKE 57751,252. А теперь сама дополнительная процедура:

```
64649      CP      255      ;Если байт по адресу HL не равен 255, то
          JP      NZ,57752 ;продолжение стандартной процедуры печати.
          LD      A,(HL)    ;Если следующий байт
          CP      255      ;не равен 255, то возврат
          RET     NZ        ;из процедуры печати.
          INC     HL        ;
          PUSH    BC        ;
          LD      A,(HL)    ;В регистр HL пересылается тот адрес,
          LD      B,A        ;который находится после двух чисел 255.
          INC     HL        ;
          LD      A,(HL)    ;
          LD      H,A        ;
          LD      L,B        ;
          POP     BC        ;
          JP      57743      ;Печать следующего символа.
```

На листинге, приведённом ниже, эта процедура формируется в строках 500 и 510. Теперь о самих предметах, точнее о том, как их названия перевести на русский. Это, конечно, сделано в строках с 420 по 460, но кое-что объяснить все-таки нужно. Возьмем, к примеру, "ON", которая относится к лифту. На русский её можно перевести, как "ВКЛЮЧЕН". "ВКЛЮЧЕН" длиннее "ON" аж на 5 символов, так что придётся часть сообщения расположить где-нибудь в другом месте. Кстати, о свободном месте. Когда мы переводили на русский меню, то у нас осталось незанятое пространство с 46387 по 46405 длиной 19 байтов, а когда переводили названия предметов, то появился свободный участок с 58448 по 58488 длиной 41 байт. Слово "ВКЛЮЧЕН" мы расположим, начиная с адреса 46393. Итак, по адресу 54070 (где начиналось слово "ON") мы располагаем такую последовательность кодов: (255)(255)(57)(181) (сосчитайте: $57+181*256=46393$). Этими четырьмя числами мы забили следующие символы: ON(254)(68) "ON" мы всё равно заменяем на русское слово, а вот (254)(68) надо будет не забыть поставить после слова "ВКЛЮЧЕН" - ведь это - код управления цветом. Также важно запомнить адрес, с которого продолжается сообщение после слова "ON" - он равен 54074. Теперь по адресу 46393 помещаем следующую последовательность:

ВКЛЮЧЕН (254) (68) (255) (255) (58) (211)

(по команде 255,255,58,211 адрес становится равен 54074, так как $58+211*256=54074$). Вроде бы с "ON" всё. Но не забывайте, что "ВКЛЮЧЕН" длиннее его на 5 символов, то есть на экране появятся два лишних символа справа и три - слева, и если мы ничего не сделаем, то, во-первых, надпись "ВКЛЮЧЕН" будет не посередине, а смещена вправо, и, во-вторых, все изображение после этой надписи тоже будет сильно смещено. Поэтому давайте-ка поищем ближайший к надписи "ON" код 252 - он находится по адресу 54067 и уменьшим число после него на 2, это можно сделать командой POKE 54068,251 (раньше там было число 253). Затем найдём ближайший код 252 после "ON" - по адресу 54077. За ним находится число 254, уменьшим его на 3 командой POKE 54078,251. Вот теперь все.

С остальными словами надо разбираться точно так же. Не надо никаких POKE только для надписи "РАЗБЕЙ СТЕКЛО", так как она прижата к левому краю и туда её уже никак не сместить, к тому же она является последней надписью в сообщении (стоит код 255), следовательно, и после неё ничего не надо изменять. Все остальное смотрите в листинге программы "rus", строки с 420 по 510 (все слова расположены в строках с 420 по 460).

Итак, всё, что можно было, мы перевели. Но так ли это? Ведь остались ещё две надписи - "VIDEO GAMES" ("ВИДЕО ИГРЫ") и "STANDARD" ("САЛЮТ"). Они выполнены графически - если

вы просматривали память с помощью программы "prosm", то, наверное, заметили "VIDEO GAMES" по адресу 53715 и "STANDARD" (надпись почему-то разбита на две половинки, которые лично у меня никак не стыкуются) по адресу 48469. Если хотите, можете изменить эти надписи с помощью "prosm", но гораздо удобнее это сделать с помощью ART STUDIO (сделав, например, SAVE "video.fnt" CODE 53715,768 и SAVE "stand.fnt" CODE 48469,768). А ещё лучше - наберите несколько дополнительных строк из программы "rus" - с 600 по 730.

И всё-таки несколько английских надписей осталось: "JAM" (на баночке джема), "PETE" (на книге в библиотеке), "BAT" (на силовом узле), "BP" (на канистре), "3 TON" (на гире), "R" (на круглом ключе), "T" (на треугольном ключе) и "S" (на квадратном ключе). Не удивляйтесь, что Вы не обнаружили эти надписи, подобно "VIDEO GAMES", при просмотре памяти с помощью "prosm", - ведь предметы, на которых они нанесены, являются спрайтами. Спрайты лучше всего просматривать и изменять с помощью программ типа "SCE" ("SPRITE CRACKER & EDITOR"). А у кого таких программ нет, тот пусть добьёт ещё несколько строк из программы "rus": с 800 по 940.

Несколько слов о том, что я собственно изменил: вместо "JAM" я нарисовал ягоду, так как слово "ДЖЭМ" никак не вменялось. "PETE" переделал в "ПЕТЭ" (хотя, может быть, "PETE" означает "ПИТ" - это имя такое). "BAT" было несложно переделать в "БАТ", а "3 TON" превратилось в "3 т". Аббревиатуру "BP" я заменил более наглядным "Л" (то есть 1 литр). Буквы "R", "T" и "S" по логике вещей следовало бы заменить первыми буквами от слов "КРУГЛЫЙ", "ТРЕУГОЛЬНЫЙ" и "КВАДРАТНЫЙ", но беда в том, что первое и третье слово начинаются на одну и ту же букву, поэтому вместо букв я нарисовал соответственно кружок, треугольник и квадратик (что смотрится, по-моему, даже лучше). Вот теперь уж точно конец.

Теперь - вся программа "rus" целиком. Кто ещё ничего не набирал, пусть её наберёт, запишет на кассету или дискету и попробует запустить. Никаких "Out of data", "Nonsense in Basic" и прочих ошибок быть не должно, а если подобное сообщение все-таки появилось, то тщательно проверьте программу и проверяйте до тех пор, пока не найдете ошибку (я имею в виду ошибку, допущенную при наборе).

Программа "rus".

```

200 RESTORE 240: LET ad=57960
210 FOR i=58489 TO 58551 STEP 2: RANDOMIZE ad: POKE i,PEEK 23670: POKE
    i+1,PEEK 23671: READ a$: LET l=LEN a$: FOR j=1 TO l: POKE ad,CODE
    a$(j): LET ad=ad+1: NEXT j: POKE ad,255
220 IF a$(1)=" " THEN POKE ad-1,252: POKE ad,257-1: POKE ad+1,1: LET
    ad=ad+2: READ a$: LET l=LEN a$: FOR j=1 TO l: POKE ad,CODE a$(j): LET
    ad=ad+1: NEXT j: POKE ad,255: IF a$(1)=" " THEN POKE ad-1,255: POKE
    ad,250
230 LET ad=ad+1: NEXT i: POKE 58083,246: POKE 58177,246
240 DATA "KRUGLYJ ","KL@<","RADIO","KWADRATNYJ ","KL@<","DWERNAQ
    ","RU<KA","MAGNIT","FUNT ","STERLINGOW","PLQVNYJ ","MQ<","CWETO<NYJ
    ","GOR/OK","TOPLIWO W ","KANISTRE "
250 DATA "TREUGOLXNYJ ","KL@<","OGNETU/ITELX","KL@< DLQ
    ","KOROBKI","BIBLIOTE<NYJ ","BILET","&NERGIQ W ","BLASTERE
    ","MOLOTOK","ZA$ITNYJ ","/LEM","WODA W ","WEDRE ","OSTRYE ","NOVNICY"
260 DATA "UPRAWLENIE ","KONWEJEROM","BIBLIOTE<NAQ
    ","KNIGA","PENS","POLOTENCE","DVOJSTIK","POWARSKAQ
    ","<A/A","WODITELXSKIE ","PRAWA","KREST","KL@<I ","ZAVIGANIQ"
270 DATA "LUNNYJ ","KRISTALL","ORBITA ","KRISTALLA","SILOWOJ
    ","UZEL","ME<","<ASOWOJ ","KL@<"
300 RESTORE 320

```

```

310 FOR J=1 TO 36: READ a,a$: FOR i=1 TO LEN a$: POKE a+i-1,CODE a$(i):
    NEXT i: NEXT J
320 DATA 54106,"Q$IK",56191,"RAZMEN",56502,"<AJ",45240," NAVMITE L@BU@
    KLAWI/U ",46171,"NAVMITE 'WLEWO'"
330 DATA 46192,"NAVMITE 'WPRAWO' ",46214,"NAVMITE
    'PRYVOK'",58806,"ESTX",58811,"NET"+CHR$ 255
340 DATA 33393,"VIZNI"+CHR$ 252+CHR$ 1+CHR$ 1+CHR$ 254+CHR$
    67+"WA/A"+CHR$ 252+CHR$ 251+CHR$ 1+"&NERGIQ"
350 DATA 46265,CHR$ 254+CHR$ 68+"1.SINKLER DVOJSTIK"+CHR$ 252+CHR$
    238+CHR$ 5+"2.KEMPSTON DVOJSTIK"+CHR$ 252+CHR$ 237+CHR$ 5+"3.NOWYE
    KLAWI/I"+CHR$ 254+CHR$ 70+CHR$ 252+CHR$ 237+CHR$ 5+"WLEWO="+CHR$
    252+CHR$ 11+CHR$ 0+"WPRAWO="+CHR$ 252+CHR$ 241+CHR$ 3+"PRYVOK="+CHR$
    252+CHR$ 240+CHR$ 4+CHR$ 254+CHR$ 7+"NAVMITE 'WWOD' DLQ STARTA"+CHR$
    255
360 DATA 46505,CHR$ 7
370 DATA 45267,"WY NE SMOGLI RAZBUDITX UOLLI",45302,"PO&TOMU ON
    PROSPAL SWO@ RABOTU",45337,"I BOSS EGO UWOLIL! "
380 DATA 45366,"W &TOM WINOWATY WY!" +CHR$ 252+CHR$ 237+CHR$ 5+CHR$
    254+CHR$ 67+"NADE@SX, W SLEDU@$IJ RAZ U WAS"+CHR$ 252+CHR$ 224+CHR$
    3+"POLU<ITSQ LU</E, I WY RAZBUDITE"+CHR$ 252+CHR$ 224+CHR$ 3+"UOLLI
    WOWREMQ."
390 DATA 45072," W E L I K O L E P N O ! ",45109,"UOLLI PROSNULSQ
    I PERWYJ RAZ W",45143,"VIZNI RANO PRI/*L NA RABOTU!",45177,"&TO
    POBEDA! "
400 DATA 45197," TEPERX SMOTRITE ",45219," 'VIZNX UOLLI' "
410 DATA 45512," /AGI UOLLI: ",45531," ",45539," ON POSETIL
    ",45566," WSEH KOMNAT "
420 DATA 55150,"RAZBEJ"+CHR$ 252+CHR$ 250+CHR$ 1+"S"+CHR$ 255+CHR$
    255+CHR$ 51+CHR$ 181,46387,"TEKLO"+CHR$ 255
430 POKE 54068,251: POKE 54078,251: DATA 54070,CHR$ 255+CHR$ 255+CHR$
    57+CHR$ 181,46393,"WKL@<EN"+CHR$ 254+CHR$ 68+CHR$ 255+CHR$ 255+CHR$
    58+CHR$ 211
440 POKE 54111,252: POKE 54121,253: DATA 54113,CHR$ 255+CHR$ 255+CHR$
    80+CHR$ 228,58448,"ESTX"+CHR$ 254+CHR$ 68+CHR$ 255+CHR$ 255+CHR$
    101+CHR$ 211
450 POKE 54460,251: POKE 54475,250: DATA 54467,CHR$ 255+CHR$ 255+CHR$
    90+CHR$ 228,58458,"ZAPRAWKA"+CHR$ 255+CHR$ 255+CHR$ 199+CHR$ 212
460 POKE 54544,252: DATA 54525,"MAGNITNYJ"+CHR$ 255+CHR$ 255+CHR$
    102+CHR$ 228,58470,CHR$ 252+CHR$ 249+CHR$ 1+"ZAMO"+CHR$ 255+CHR$
    255+CHR$ 11+CHR$ 213
500 RESTORE 510: FOR i=64649 TO 64669: READ a: POKE i,a: NEXT i: POKE
    57749,195: POKE 57750,137: POKE 57751,252
510 DATA
    254,255,194,152,225,126,254,255,192,35,197,126,71,35,126,103,104,193,19
    5,143,225
600 RESTORE 610: FOR i=53715 TO 53762: READ a: POKE i,a: NEXT i
610 DATA 110,255,139,171,154,169,139,255
620 DATA 182,255,115,107,107,107,65,255
630 DATA 182,255,17,117,21,117,17,255
640 DATA 127,237,237,233,229,237,255,118

```

```

650 DATA 255,17,117,113,119,119,255,186
660 DATA 254,119,119,23,87,23,255,182
700 RESTORE 710: FOR i=48469 TO 48501 STEP 8: READ a,b,c: POKE i,a: POKE
    i+1,b: POKE i+2,c: NEXT i
710 DATA 184,183,183,227,93,221,194,186,186,198,187,59,13,189,189
720 RESTORE 730: FOR i=48504 TO 48536 STEP 8: READ a,b,c: POKE i,a: POKE
    i+1,b: POKE i+2,c: NEXT i
730 DATA 183,183,184,193,93,221,186,186,58,187,187,199,189,189,189
800 RESTORE 810: FOR i=39926 TO 39934: READ a: POKE i,a: NEXT i
810 DATA 99,255,221,128,93,255,221,0,99
820 FOR i=39988 TO 39996: READ a: POKE i,a: NEXT i
830 DATA 33,255,237,128,45,255,237,0,33
840 FOR i=40210 TO 40222: READ a: POKE i,a: NEXT i
850 DATA 13,0,25,255,245,128,45,255,245,0,25,248,13
860 FOR i=40183 TO 40198: READ a: POKE i,a: NEXT i
870 DATA 247,195,231,147,199,147,231,147,231,147,231,147,231,147,195,19
880 POKE 40863,194: POKE 40865,222
890 FOR i=41455 TO 41480: READ a: POKE i,a: NEXT i
900 DATA
    63,252,127,254,127,254,99,254,93,254,125,254,125,254,243,131,253,239,25
    3,239,221,239,227,239,255,255
910 FOR i=43733 TO 43742: READ a: POKE i,a: NEXT i
920 DATA 209,1,213,109,213,41,213,109,213,41
930 FOR i=41783 TO 41796: READ a: POKE i,a: NEXT i
940 DATA 51,252,57,204,63,156,62,124,60,60,44,52,46,116

```

И в заключение хочу дать несколько советов, которые помогут Вам в русификации других программ:

1. В такой известной игре, как "DIZZY" символьный набор состоит всего-лишь из 26 букв алфавита и еще нескольких знаков, что не позволяет заменить его на полноценный 32-символьный русский набор. Тут Вам придется или писать дополнительную процедуру печати или найти где-нибудь свободное место и загнать свой набор туда (например, если Вы добились определенной экономии в названии предметов, то можете эти две области памяти обменять). Так же можно поступить, если программа вообще не использует своего шрифта, а берет стандартный из ПЗУ. В таких случаях еще может помочь буфер принтера.

2. Не во всех программах текстовые сообщения находятся в таком виде, как в "PYJAMARAMA". В "DIZZY 5", например, где-то в определенном месте находится таблица, в которой беспорядочным образом размещены все используемые символы, и если, например, где-то находится такая таблица: "AOMRENT", то слово "MOTHER" зашифруется так - (3)(2)(7)(6)(5)(4), а слово "МАМА" - (3)(1) (3)(1), то есть, каждая буква заменяется своим номером в таблице (отсчет может начинаться и с нуля). А в игре "ROBIN OF SHERLOCK" каждая буква получается после вычитания кода из числа 255. То есть если на экране слово "МАМА" (или, кодами: 77,65,77,65), то в памяти это слово хранится как (178)(190) (178)(190), так как 255-77=198 и 255-65=190.

3. Обязательно найдите где-нибудь (если у вас нет) программу типа "SCE" - для выламывания и редактирования спрайтов.

4. О том, что программы бывают, закодированы, я уже не говорю.

УДАЧНЫХ ВАМ РУСИФИКАЦИЙ!

Послесловие ИНФОРКОМА.

Вы узнали некоторые из секретов, связанные с высококвалифицированным переводом программ на русский язык. В то же время, ни для кого не секрет, что немаловажным фактором

является оснастка. Приятно работать, имея в руках хороший инструмент! Да и автор тоже упоминает, в частности, программу "SCE". Огромную помощь в работе Вам могут оказать программы с дистрибутивного диска "УТИЛИТЫ-1". Это "FILES DUMP EDITOR" (компании "STALKER" из Пскова) и "FIND EDITOR FONT", загружаемая в экранную область - неброская на вид, но очень эффективная программа (Германа Мясникова из Сыктывкара). Эти программы позволяют искать и редактировать спрайты и символьный набор прямо в существующих файлах, не создавая новых, а также могут кое-что еще. Особо следует отметить подготовленную для дистрибуции программу "TURBO V-SCREENER" (Алексея Мартынова), специально созданную для русификации программ (поиск и редактирование шрифтов, спрайтов и текстовых сообщений). В этой программе, кстати сказать, предусмотрена возможность расширения возможностей самим пользователем при помощи собственных процедур в машинных кодах. Кроме того, специально для работы со спрайтами предназначен подготовленный для дистрибуции пакет из двух программ (Красноярской группы "FAMOUS FACES FACTORY"): "THE REAL SPRITES TRANSFORMER" (авторы: В.Савенков и Ю.Батенко) и "THE REAL MAP EDITOR" (Ю.Батенко и А.Леготин).

Все эти программы созданы людьми, так или иначе на себе испытывшими проблемы русификации. Поэтому мы искренне надеемся, что благодаря совместно начатому делу по дистрибуции авторских программ, на свет появятся не просто высококвалифицированные переводы, а настоящие шедевры!

* * *