

UZIX




UZIX — это реализация UNIX для компьютеров MSX (Unix Implementation for MSX). Она основана на UZI, разработанной Дугласом Брауном, и была портирована на MS-DOS / MSX Аркадием Щекочихиным и Адриано да Кунья (Adriano C. R. da Cunha). UZIX может быть запущена на PC (из MS-DOS) или на MSX2/2+/TurboR. Исходный код в основном написан на Си, и транслировался с помощью Turbo-C (ПК) и Hitech C (MSX).

<http://uzix.sourceforge.net/>

История

MS-DOS версия разрабатывалась Аркадием, практически самостоятельно, с 1997 по 2001. Адриано, в свою очередь, почти целиком ответственен за портирование на MSX. В версии 1 система занимала в ОЗУ менее 30Kб (а точнее, 25.8Kб). Это, наверное, самая компактная реализация [W 7th Edition](#), какую только возможно найти.


Для UZIX было создано и портировано много подсистем и приложений, таких как стек TCP/IP, клиенты telnet, ftp, irc, демоны telnetd и httpd, и даже графический Web-браузер  FudeBrowZer, разработанный Рикардо Битенкуром (Ricardo Bittencourt).

В 2002 году вышла версия UZIX 2. Это был большой шаг вперед по сравнению с первой версией. Основным отличием стало большее ядро, позволяющее включить в него больше функций, и, в то же самое время, большее пространство ОЗУ, доступное для прикладного кода — что открыло возможности для запуска более крупных приложений. В этой версии UZIX получил и множество других улучшений: в частности, графическую систему (не X11), модульное ядро (с возможностью горячей загрузки и выгрузки модулей ядра), была также предпринята попытка создать собственные средства разработки - FOCA - компилятор Си (не завершен), ассемблер Z80, оптимизатор. Разработкой FOCA занимались Фелипе Берго (Felipe Bergo) и Адриано Кунья. Последней версией UZIX стала 2.1alfa от 24/09/2003. Последняя правка в файлы UZIX вносилась 18/03/2005, после чего разработка, очевидно, прервалась.

Архитектура ОС

Архитектура UZIX 1, за небольшим отличием — в целом, идентична UZI:


Поскольку на компьютерах 8080 семейства нет стандартного механизма менеджмента памяти (MMU), *многопроцессность* достигается методом «total swapping» (полного своппинга). Но, в отличие от UZI, для «total swapping» используется не жесткий диск, а память маппера. На PC — для этого используется «расширенная память». В обоих случаях, UZIX доступно 64Kб адресного пространства (всё адресное пространство Z80 или один полный сегмент на PC). Собственно UZIX располагается в верхних 32Kб памяти, а работающий в данный момент процесс занимает нижние 32Kб. UZIX не может обойтись без некоторых дополнительных устройств. Во-первых, должен присутствовать, в том или ином виде, таймер или часы, которые могут обеспечить периодическое прерывание. Также, дополнительно используются часы реального времени (RTC) для получения времени для временных меток файлов, и т.п. ТTY-драйвер опрашивает клавиатуру в цикле ожидания — а не предполагает работу от прерываний, как в UZI.

Существует специальная «лайт-версия» UZIX для MSX1, работающая в режиме 40 символов в строке, и эмулирующая недостающий RTC для своих нужд. Но, в виду отсутствия в MSX1 штатного RAM-маппера, она использует  [бразильский MegaRAM](#), что замедляет работу системы, по-сравнению с MSX2. Скачать её можно [здесь](#).

Архитектура UZIX 2 существенно отличается от предыдущей версии:


Операционная система, аналогично, использует маппер MSX2/2+/TR для «total swapping». Но само ядро UZIX занимает все 64Кб памяти, разделенное на две части — 48Кб для выгружаемой части ядра, и 16Кб для невыгружаемой (резидентной). Резидентная часть является интерфейсом между userland-программами и остальным ядром. Во время системных вызовов или прерываний, выгружаемая часть ядра свопится так же, как любой другой процесс. Приложения занимают в ОЗУ, как минимум, 16Кб, а максимум — 48Кб (в зависимости от их размера). UZIX 2 поддерживает до 4096Кб памяти. Максимальное количество процессов 31.

Возможности:

- UZIX работает на любой MSX ([MSX2](#), [MSX2+](#), TR а также в эмуляторах);
- Полнофункциональная многозадачная среда;
- Полнофункциональное многопользовательское окружение;
- Стабильная файловая система, почти без багов;
- Чрезвычайно стабильное ядро, шелл и окружение;
- Поддержка модулей (модульность используется пока только для TCP/IP стэка);
- Поддержка жестких дисков ( [ESE MegaSCSI](#), [Sunrise IDE](#) и [Beer IDE 202^{1\)}](#));

Попробовать UZIX 2 можно в эмуляторе  [WebMSX](#). Логин — root без пароля.

Примечания:

1. Если Вы захотите запустить UZIX 2 на реальном железе, то нужно учитывать, что он не заработает на MSX без маппера ОЗУ, и заработает крайне ограничено на машинах с ОЗУ 128Кб (а это такие модели, как Yamaha YIS-805/128R и Yamaha YIS-503IIIR) — запустится shell, но, при попытке запуска чего-либо, будет выдаваться сообщение о невозможности порождения нового процесса (fork failed). Для нормальной работы требуется, как минимум, 256Кб. Тут выручить может [Carnivore](#), либо другое расширение памяти.
2. Некоторые DiskROM-ы не очень хорошо работают с UZIX. DiskROM для MSX Turbo-R также в этом замечен, и проблема связана с остановкой дискеты. Если дисковод прекращает вращение, то следующее обращение к нему выдаст ошибку 'not ready' после длительного таймаута. Эта проблема решается установкой соответствующей галочки в 'Advanced settings' в инсталляторе UZIX, либо нажатием пробела во время загрузки ZILO.
3. Работа UZIX была протестирована только на контроллерах ESE MegaSCSI, Sunrise IDE и Beer IDE; Контроллеры Novaxis SCSI выдают ошибку при попытке доступа к диску; UZIX, в принципе, должен работать и на других устройствах (Bert, Gouda, и т.д.), но может потребоваться некоторая тонкая настройка.
4. Инсталляция на Sunrise IDE с [NEXTOR](#) kernel, к сожалению, невозможна. Для этого необходимо будет прошить [оригинальный ROM Sunrise](#) (или, если речь об эмуляторе, прописать его в [настройках](#)). Образ HDD с установленным UZIX 2.1 для использования с оригинальной прошивкой Sunrise можно [скачать тут](#). Там используется старый  'MSX-style' формат таблицы разделов, отличающийся от MS-DOS partition table.
5. Инсталляция на Beer IDE возможна, но затруднена. Поэтому для работы с BEER 202 предлагается готовый загрузочный [образ](#) с установленным UZIX. Используется MS-DOS MBR, стандартная Partition Table.

Примечание 1

Разработчиком поддержка изначально не заявлялась, но в 2022 году была [случайно обнаружена](#) де-факто, и полностью подтверждена тестами на эмуляторе и реальном железе.

Beer IDE

Есть тонкости при инсталляции. В частности, UZIX видит разделы жесткого диска BEER IDE как /dev/fd*

Ссылки

<https://github.com/marioaugustorama/uzix-kernel>

[Логотип](#)

¹⁾

Смотри [Примечание 1](#)

<http://sysadminmosaic.ru/msx/uzix/uzix>

2024-02-16 14:22

