

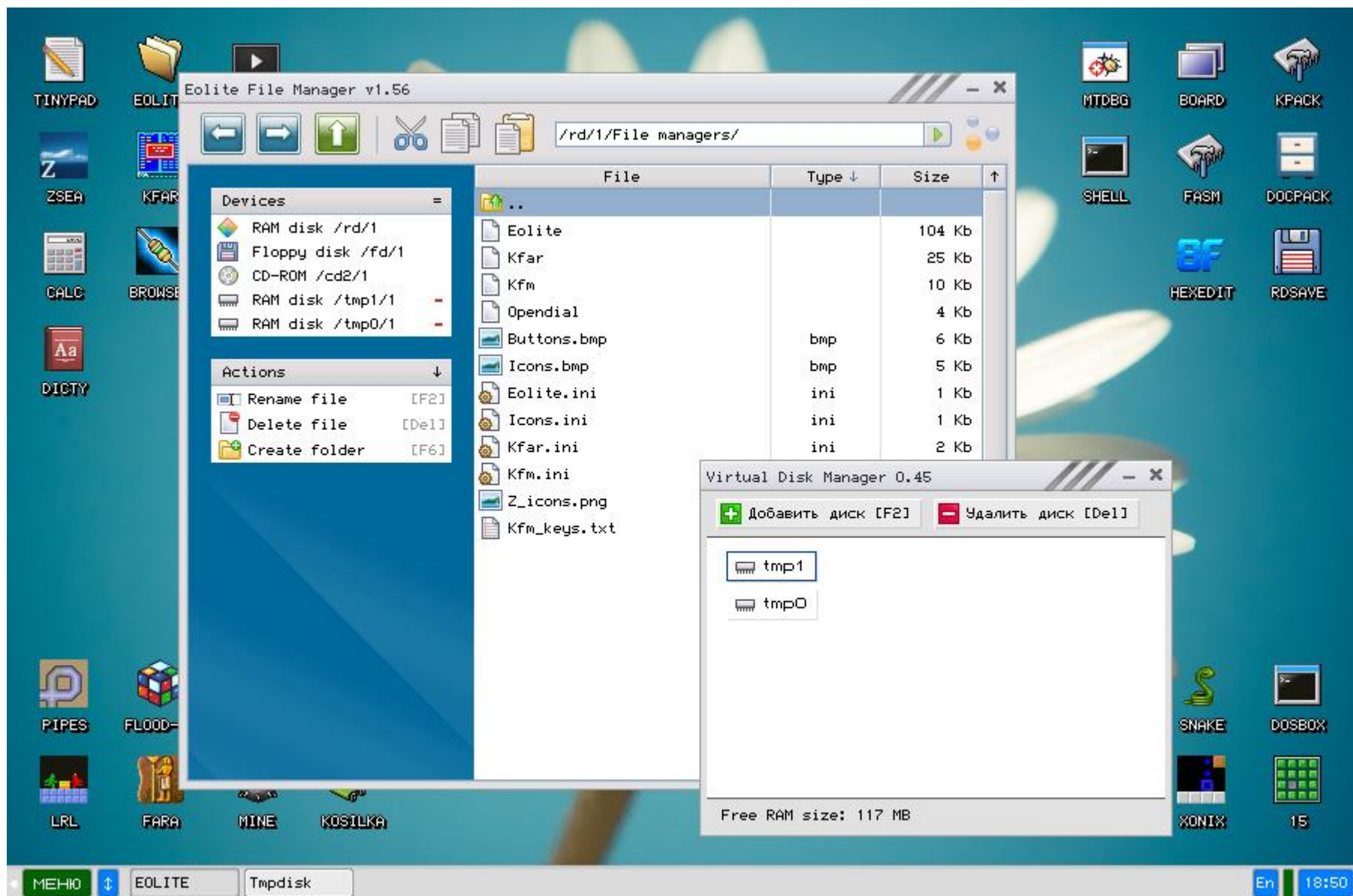


Kolibri OS

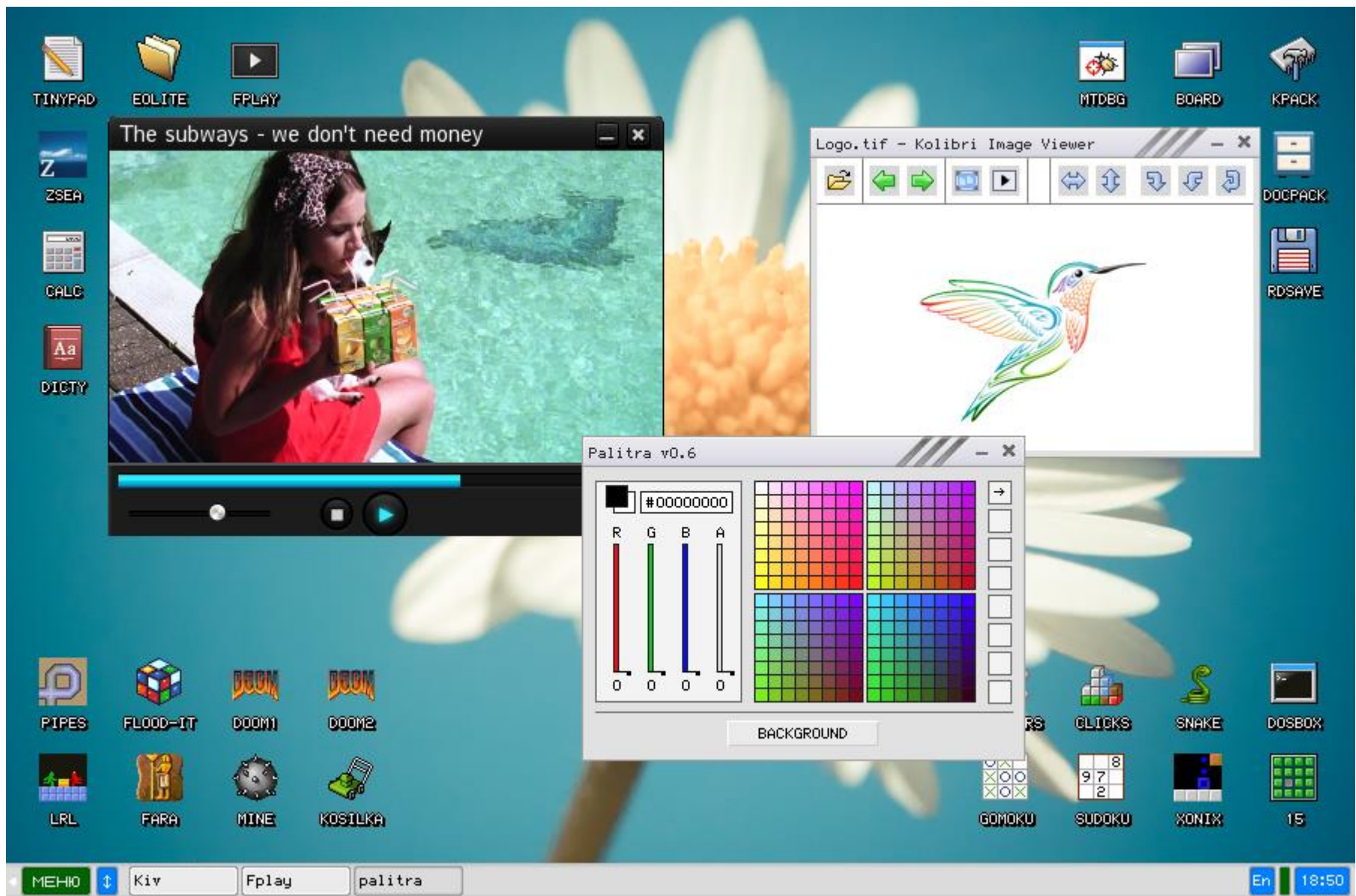
Докладчик: Липатов Кирилл

Особенности KolibriOS

- миниатюрность
- скорость
- графическое окружение



Файловый менеджер Eolite, менеджер виртуальных дисков (RAM-) дисков



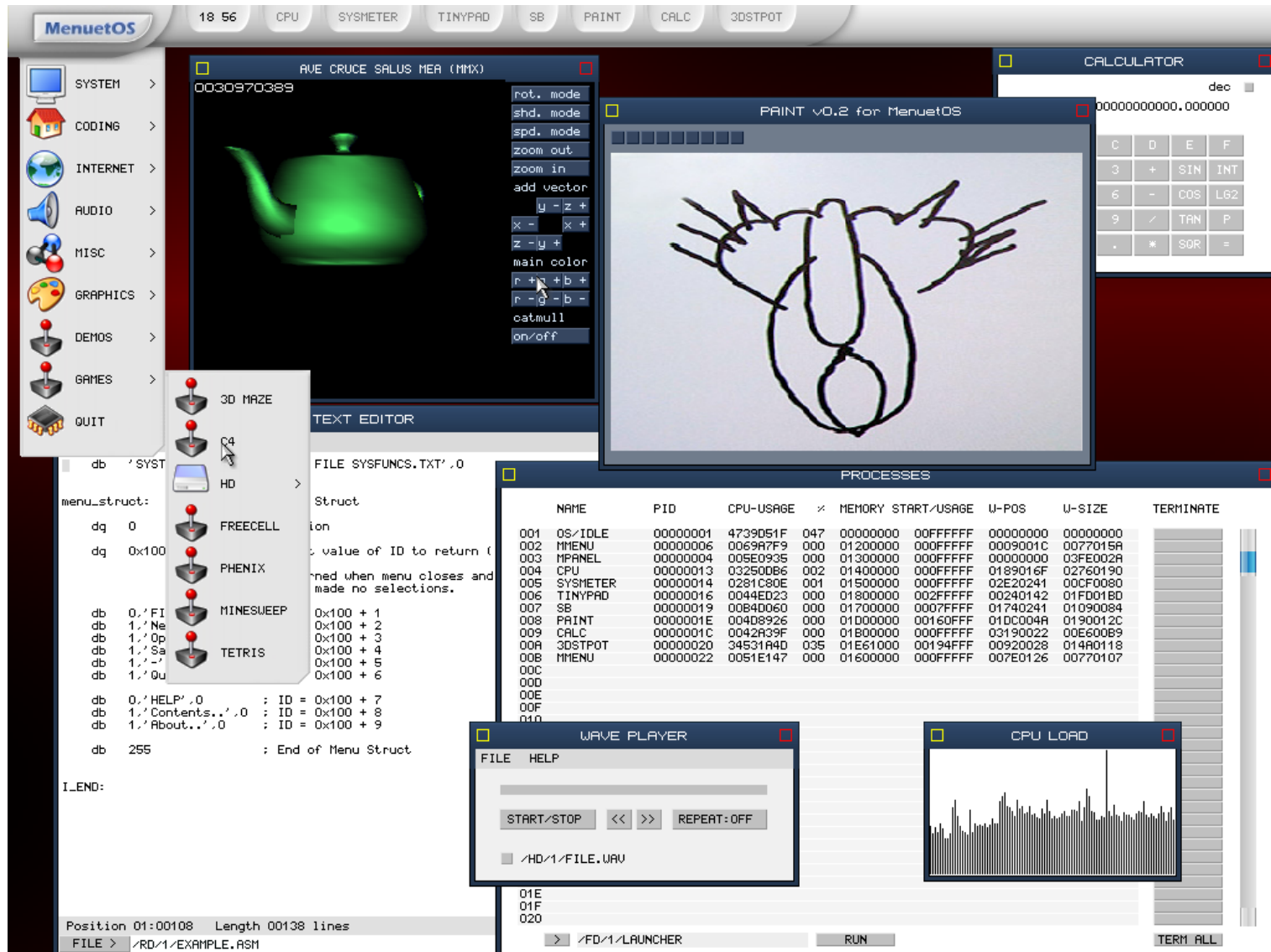
Видеоплеер FPlay, просмотрщик изображений KIV, Палитра

Минимальные системные требования

- Intel 586 совместимый процессор на частоте 50 MHz
- 8 Mb оперативной памяти
- VGA/Vesa совместимая видеокарта
- 1.44 Мб на загрузочном устройстве

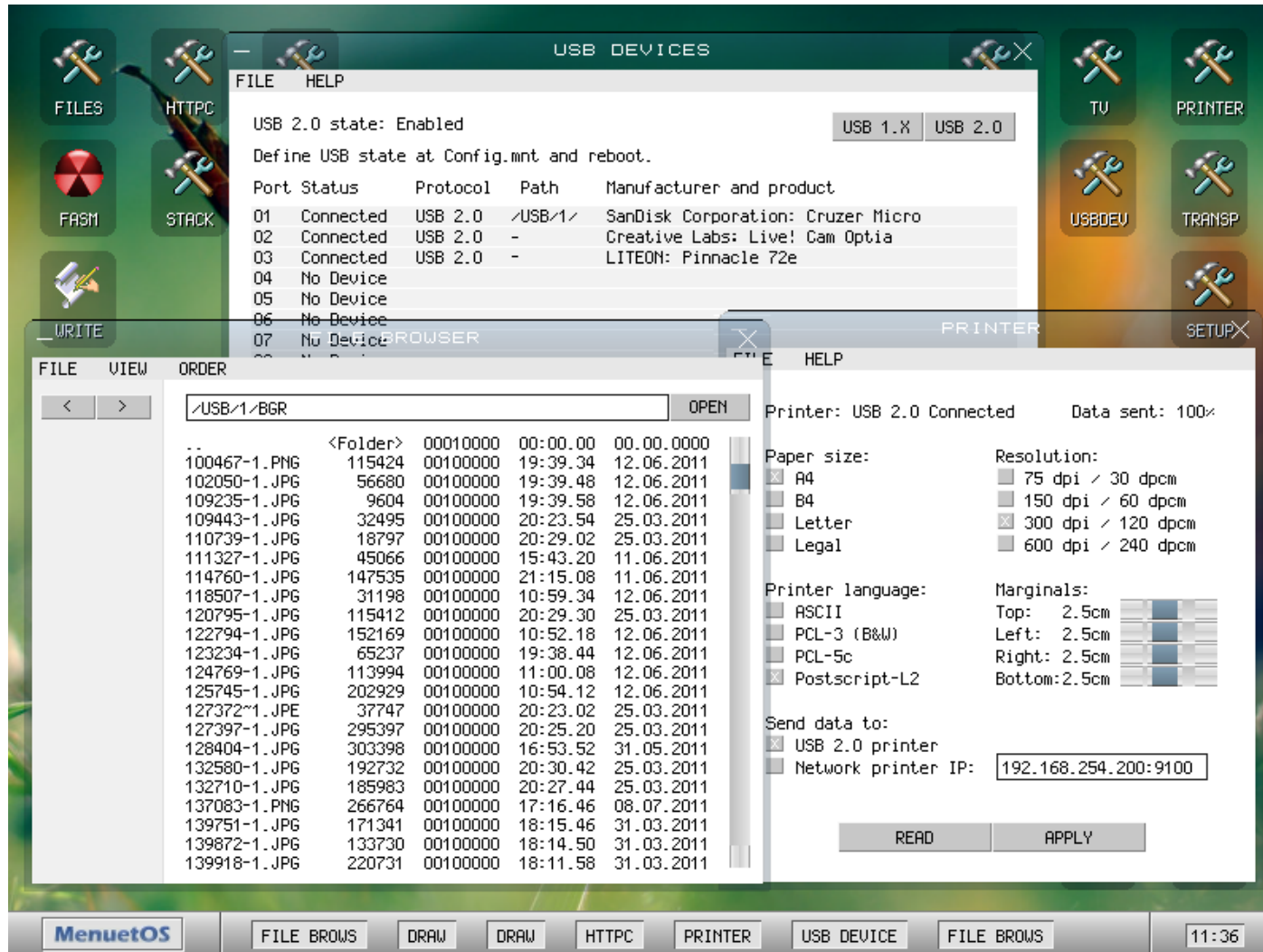
Menuet32

- предшественник Колибри
- проект закрыт



Menuet64

- учтены прошлые ошибки
- поддержка x64
- исходники закрыты
- требует 768 Мб ОЗУ
- неплохой браузер



Возможности KolibriOS

- вытесняющая многозадачность, возможность создания потоков, параллельное исполнение системных вызовов
- страничная адресация
- звуковые карты (AC'97, HDAudio, SoundBlaster)
- TCP/IP-стек, поддержка популярных сетевых карт
- поддержка аппаратного ускорения для видеокарт ATI и Intel, для других используется стандартный драйвер Vesa
- USB стек (OHCI, UHCI, EHCI), поддержка мышек, клавиатур и флешек
- полная поддержка FAT 12/16/32; только чтение ISO9660 (Joliet), NTFS, XFS* и EXT 2/3/4*

Программирование в KolibriOS

Описание KolibriOS с Википедии:

«Система рассчитана на использование ассемблера для написания приложений, но есть и программы, написанные на языках Ада, Си, С++, Free Pascal, Forth.»

Примеры написания кода на ассемблере

Документация [что такое регистры?..]

Функция 1 - поставить точку в окне.

Параметры:

- `eax = 1` - номер функции
- `ebx` = x-координата (относительно окна)
- `ecx` = y-координата (относительно окна)
- `edx = 0x00RRGGBB` - цвет точки
`edx = 0x01xxxxxx` - инвертировать цвет точки (младшие 24 бита игнорируются)

Возвращаемое значение: функция не возвращает значения

```
mov eax, 1
mov ebx, 10
mov ecx, 20
mov edx, 0x00CCCCC
int 0x40
```

Код на ассемблере

Ассемблер + макрос

```
mcall 1,10,20,0x00CCCCC
```

Примеры написания кода на ЯВУ

Процедура в
библиотеке
MSVC

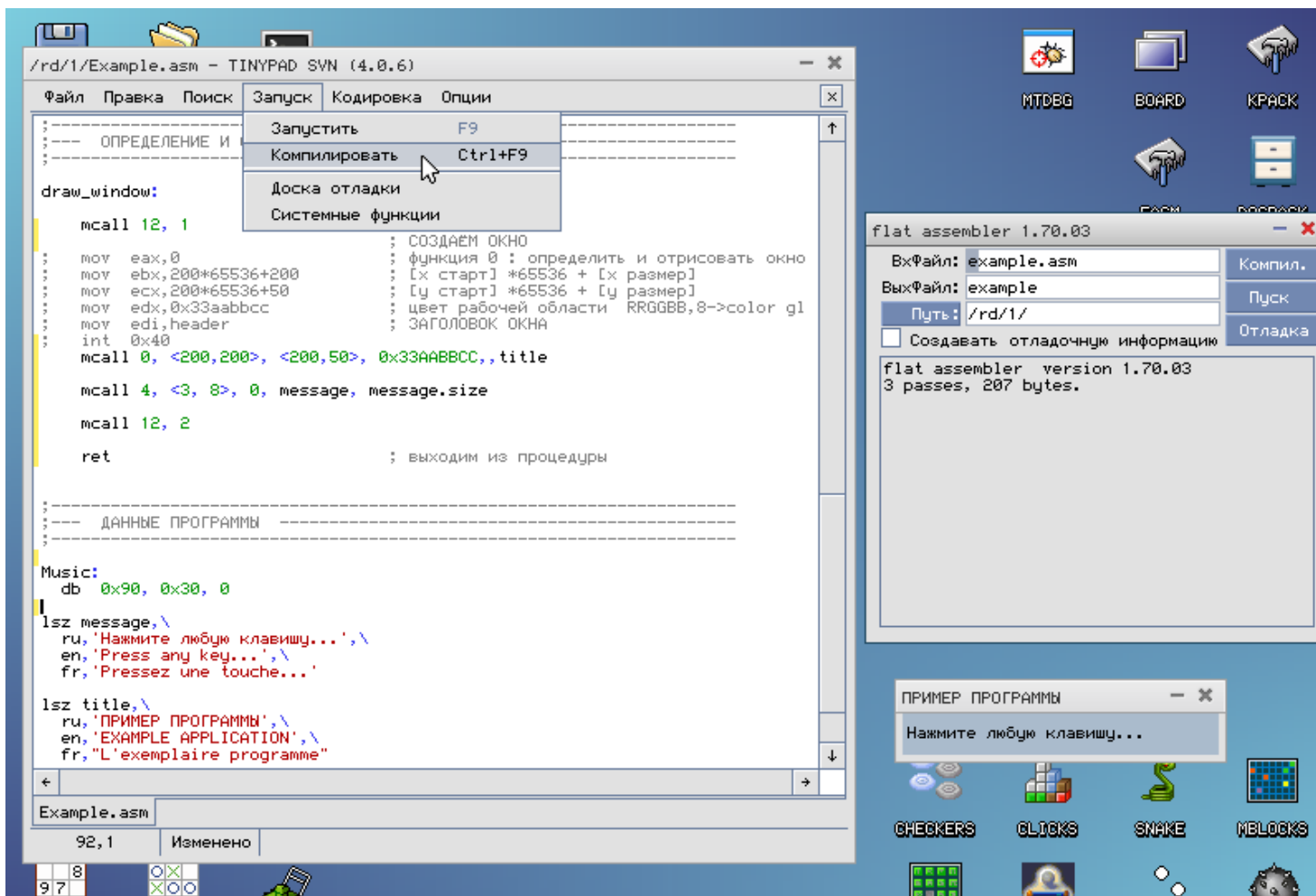
```
void kos_PutPixel( Dword x, Dword y, Dword colour )  
{  
    __asm{  
        mov eax, 1  
        mov ebx, x  
        mov ecx, y  
        mov edx, colour  
        int 0x40  
    }  
}
```

Код в программе

```
kos_PutPixel(10, 20, 0x00CCCCC);
```

FASM

- много библиотек
- много примеров
- компиляция через редактор TinyPad, файловый менеджер KFar и сам FASM
- простота использования и компиляции



Другие компиляторы и интерпретаторы под Колибри

- TinyC
- Порт LUA последней версии 5.2
- Порт Pyton (TinyPy)
- Интерпретатор Brainf#ck

Кросскомпиляция

- MSVC++
- GCC, G++ (библиотеки libc и newlib)
- Borland C++
- Sphinx C—
- Free Pascal и Pascal Pro
- Ассемблеры FASM, NASM, MASM
- XDS Modula-2/Oberon-2 compiler
- Среда Delphi
- Open Watcom

Примеры <http://kolibri-n.org/inf/hll/hll.php>

Проблемы Kolibri

- непертируемость
- слабая библиотека GUI элементов
- нехватка программного обеспечения и драйверов
- мелкие и не масштабируемые шрифты
- отсутствие нормального браузера

Варианты использования

- для восстановления данных после падения системы [цыфры]
- как сервер
- для маломощных дешёвых компьютеров [цифры для сравнения]
- как встраиваемая система [цифры для сравнения]
- для тестирования и наладки нового железа [что за патенты?]
- для домашнего и офисного использования
- [не закончено]

Планы

- больше открытости и социальности
- расширение поддержки сети
- лучше организация и координация
- хороший софт
- попытки покорения новых рынков, например, встроенного оборудования

