OpusRead v2.0.0

Opus Discovery Floppy/Image Reader for Windows

Freeware

Overview

OpusRead is a freeware Win32 program -written in **VB6**- that allows a user to read, analyze and save **Opus Discovery** floppy images and physical disks. The different **Opus** floppy files' types can be viewed with the included Screen, Program Listing and Binary viewers, and these can also be saved in several formats as well as printed. It does not, however, write physical floppies.

Requirements

The program was developed under **Windows 2000**, but it should also work on **Windows 9x**, **XP** and **2003**. To read physical floppies it relies in the **FDRawCmd** driver by Simon Owen (http://simonowen.com/fdrawcmd/) but please note this driver runs under **Windows 2000**, **XP** and **2003** as stated on Simon's site. **Windows 9x** users will thus be limited to use **OpusRead** with existing floppy image files.

Installation and Removal

Due to several limitations and annoyances with standard installers, I decided against an installer package. Instead, I smacked all needed files in a Zip file and distribute it. The steps to install **OpusRead** are (assumed Windows in English):

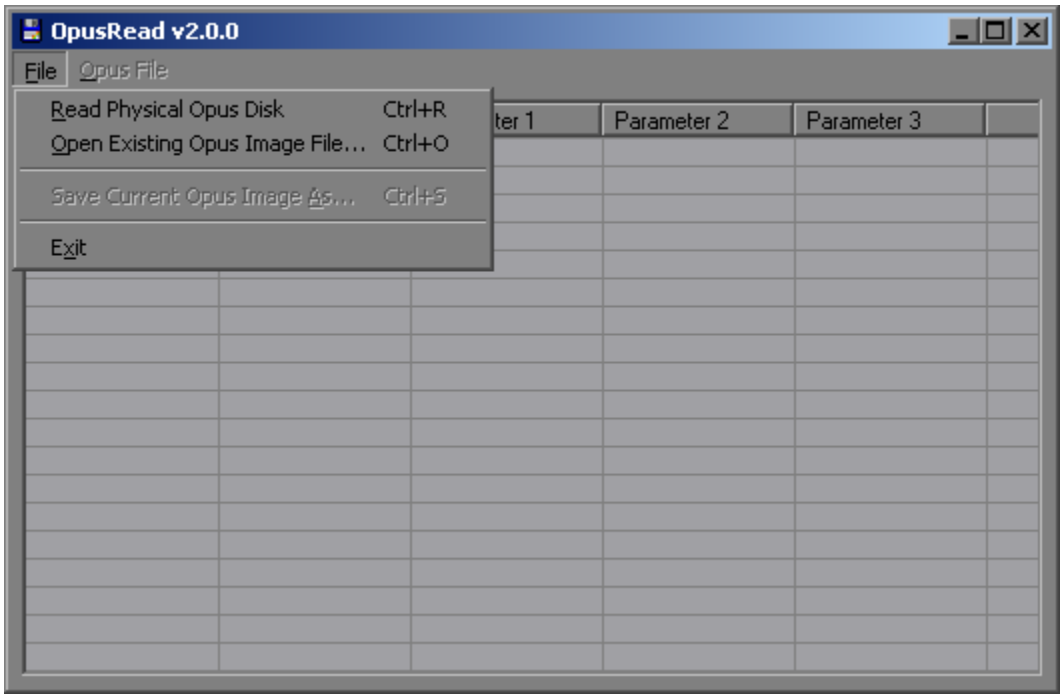
- Important note for **Win9x** users: Download and install the **Visual Basic 6** runtime libraries from Microsoft (search for **vbrun60sp5.exe**). Last time I checked it was here: http://www.microsoft.com/downloads/details.aspx?FamilyID=bf9a24f9-b5c5-48f4-8edd-cdf2d29a79d5&DisplayLang=en
- Create a temporary folder, say **C:\Temp**
- Unzip all files in **OpusRead_v200.Zip** to the temporary folder
- (Optional) Run **C:\Temp\FdInstall v1.0.1.9.Exe** to install **FDRawCmd** to read physical **Opus** floppies (if you skip this step you won't be able to read physical floppies, but the program will be able to read existing floppy images)
- Create the application folder, **C:\Program Files\OpusRead**
- Copy the program files **OpusRead_v200.Exe** and **OpusRead.Exe** to the application folder
- Unzip **C:\Temp\ActiveX.Zip** to the application folder
- Change to the application folder and run **Register.Cmd** to register ActiveX components
- Create a shortcut to **OpusRead.Exe** wherever you like (desktop, start menu, etc)

The installation does not add any files to any Windows folder(s), but might change some ActiveX registrations in the registry. If you're not comfortable with this, the files in **ActiveX.Zip** can be unzipped to **C:\WinNT\System32** or **C:\Windows\System32** (depending on your Windows version) and registered in that folder instead. And, since the registration is manual, you choose whether to overwrite existing files or not.

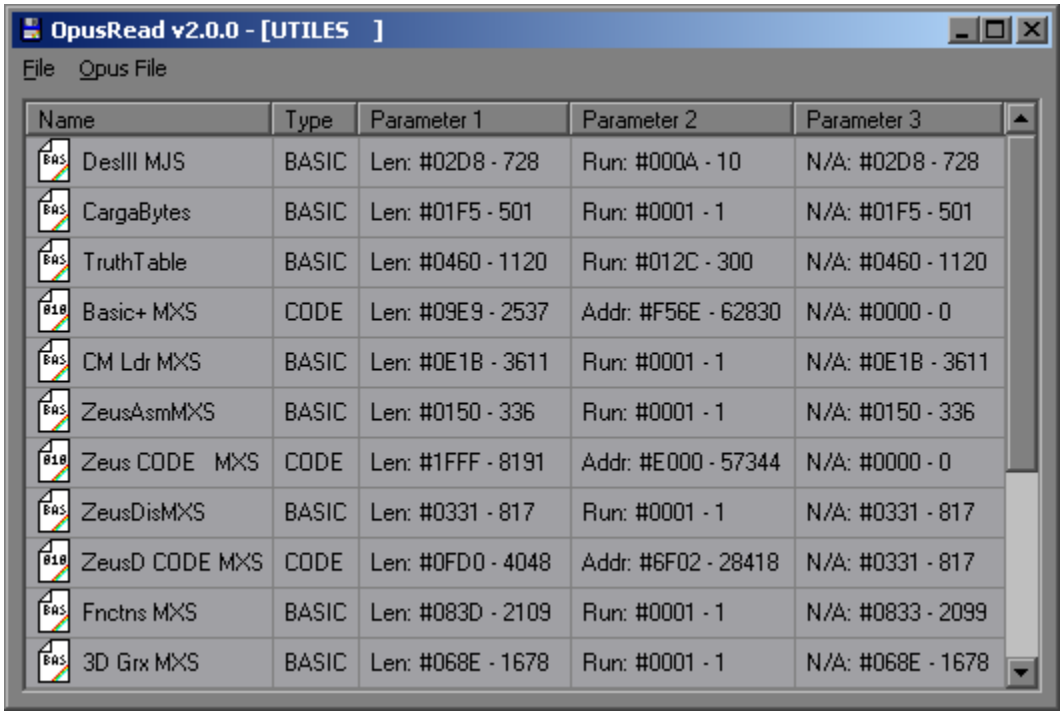
To uninstall simply reverse the install steps, but use **UnRegister.Cmd** instead of **Register.Cmd**. Please note that this step might render some ActiveX components unusable for other applications, so do this **only if you know what you're doing**. If you don't want to know anything about ActiveX, simply don't unregister anything.

OpusRead usage

The program starts blank, with no images or floppies read. You have to either open an image file or read a physical **Opus Discovery** floppy. This is a screenshot, the *File* menu open:

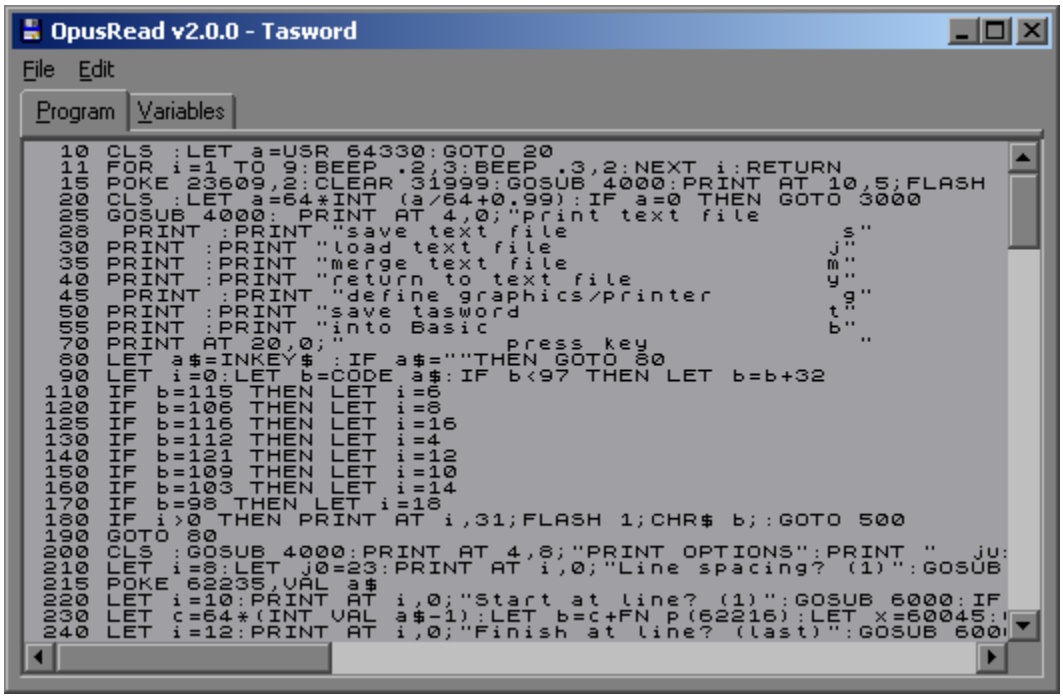


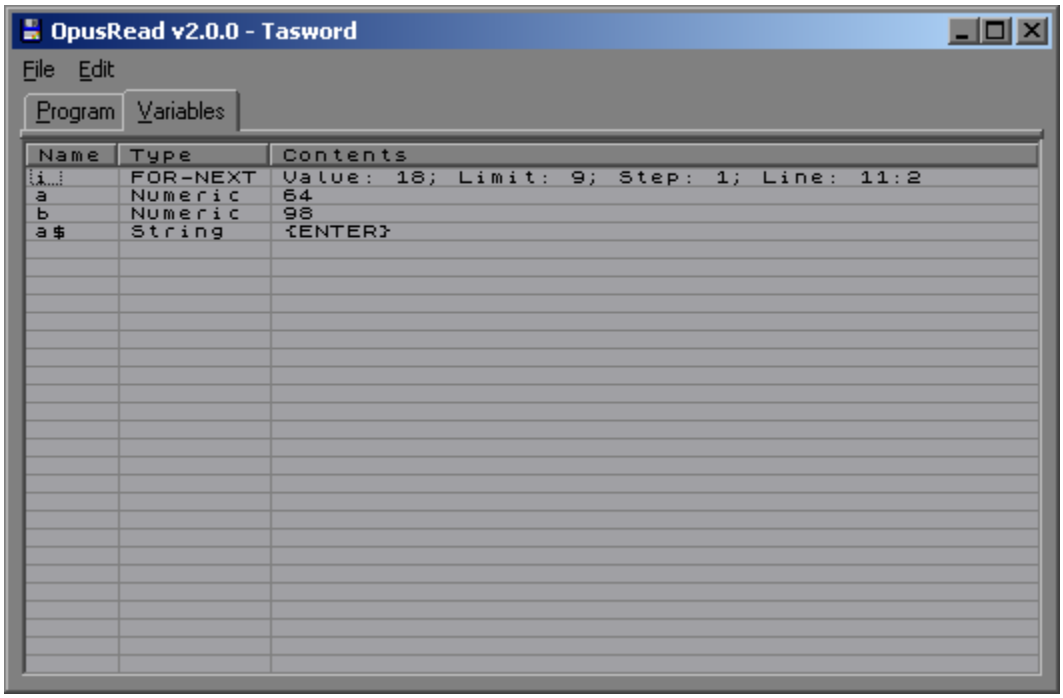
Note: If the **FDRawCmd** driver is not present, the *Read Physical Opus Disk* menu will be disabled.
When an image is read in, the screen changes to reflect its contents:



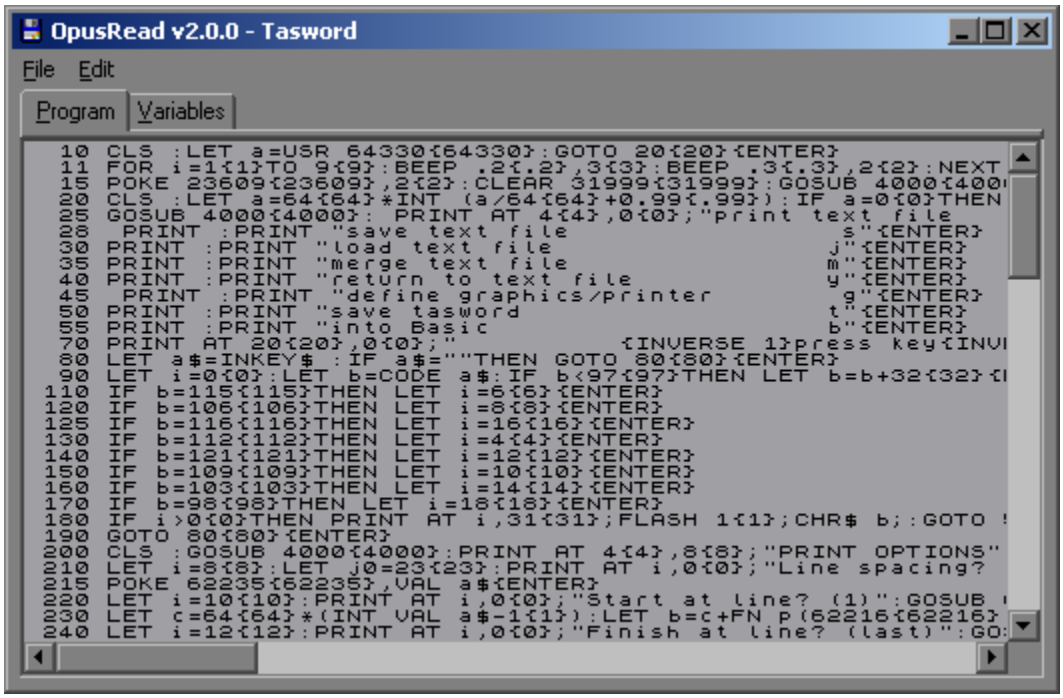
The program title bar changes to reflect the floppy label (**UTILES** in this case). The files are listed as they appear in the floppy, with no sorting whatsoever. The first column is the ZX Spectrum's file *Name*; the second is the file *Type* (can be BASIC, CODE, SCREEN$, Char Array or Num Array); and the remaining three column's meaning depend on the file type, but the *Parameter 1* is normally the file's length. All numbers are given in both Hexadecimal and Decimal notations.

If we double-click a file, a specialized viewer for the file type will be launched. Next are a program (*BASIC*) and its variables viewer:





The *Edit* menu has a pretty useful item: *Expand Control Codes (Ctrl-D)*. This option toggles expansion of the hidden – and invisible to Windows– **ZX Spectrum**'s print control codes, as well as number expansion. We all know a **ZX Spectrum** program has numbers which are in two forms: ASCII and floating point. Sometimes, for copy protection, the ASCII numbers are tampered with, and if the line is edited the floating-point representation is recalculated thus rendering the listing unusable. This is the same program as above but with the codes expanded (this example has no altered numbers, though):

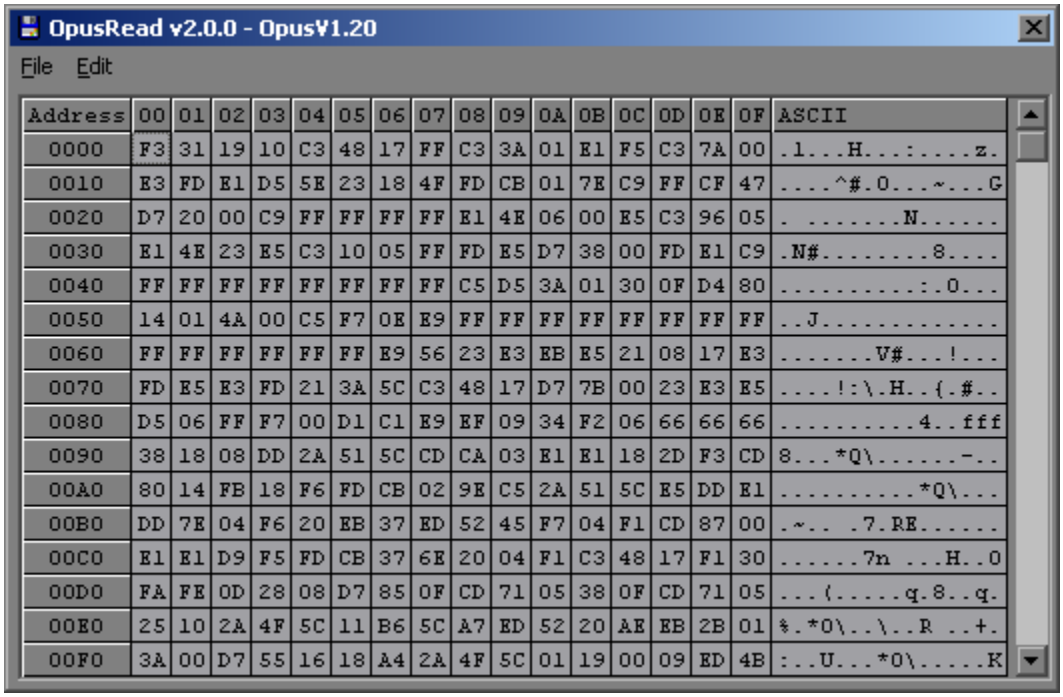


For example, line 10 reads as follows, with expansions in blue:

10 CLS :LET a=USR 64330{64330}:GOTO 20{20}{ENTER}

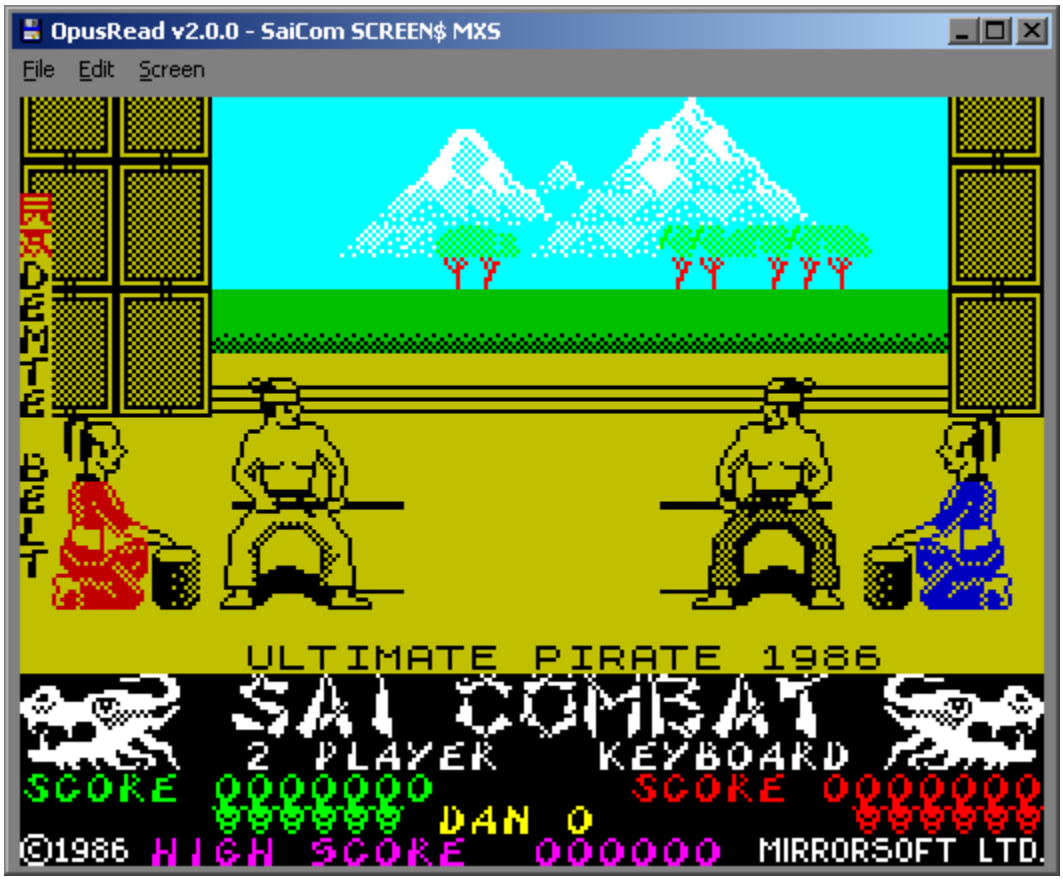
Print control codes like **FLASH**, **INVERSE**, etc, and non-PC-standard ASCII codes are also expanded.
Via the *File* menu, listings can be saved as Emulator **TAP** file, as Windows **TXT** file, or printed. Variables –if any– are also saved and printed.

A *CODE*, *Character Array*, or *Numeric Array* file triggers the following viewer (showing a *CODE* dump for the **Opus Discovery ROM v1.20**):

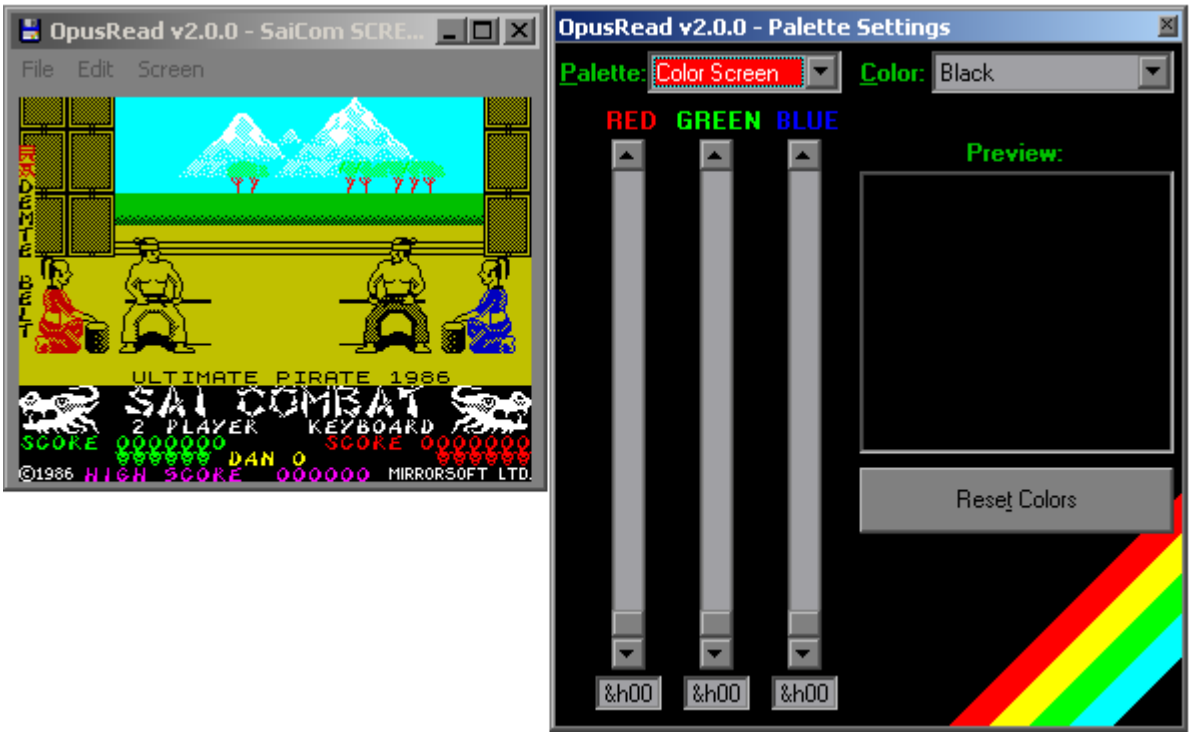


Please note the *Search* menu item does not always work correctly. This is also to be fixed.
Binary dumps can be saved to either Emulator **TAP** file or Windows **BIN** (binary) file. Files saved as **BIN** will not have any headers or descriptors and will only hold the raw bytes.

If a *CODE* file is 6912 bytes long, its type will be set to *SCREEN$* in the main screen, and the viewer will be the following (showing a pirated **Sai Combat** loading screen):



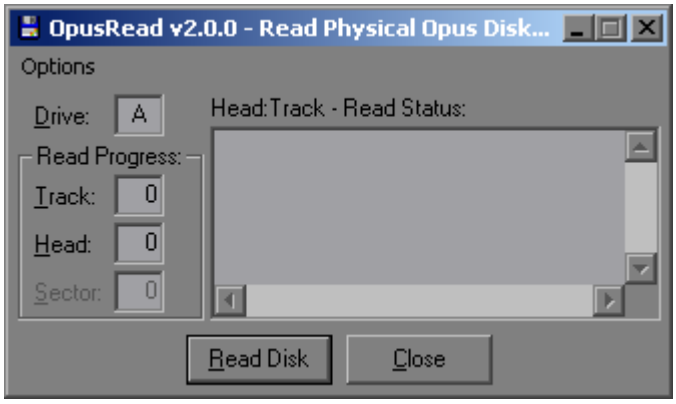
Screen files can be saved as Emulator **TAP** or **SCR** files, or Windows **BMP** files. The *Edit* menu only has the *Copy (Ctrl-C)* entry to copy the screen to the clipboard. The *Screen* menu has options to size the screen to 1x, 2x or 3x the normal size, to enable/disable **FLASH**, to see or not the **ATTR**ibutes, and to adjust the **ZX Spectrum** screen colors (palette). The palette adjustment screen looks as follows:



The *Palette* combo allows choosing between the color screen and the black on white bitmap palettes. The *Color* combo sets the color to be adjusted; depending on the selected *Palette*, the *Color* choices are 2 or 16. The sliders allow fine-tuning colors, and the *Preview* box shows how the adjusted color looks. The changes on the **ZX Spectrum** screen are immediate. Should the original colors be restored, pressing the *Reset Colors* button does it.

Reading Physical Floppies

To read physical Opus Discovery floppies, the **FDRawCmd** driver must be installed in the system. Clicking the *File – Read Physical Opus Disk (Ctrl-R)* menu shows then the next screen (note: **OpusRead** reads only standard **Opus Discovery** floppies; this is, single-density, single-side, 40 track, 17 sector floppies):



Clicking *Read Disk* starts the floppy read process. This usually takes a minute or two. If no read errors are reported, the user can click on *Close* to have the image dumped in the main **OpusRead** screen. Now it is a good time to go and click *File – Save Current Opus Image As... (Ctrl-S)* to save the **Opus Discovery** floppy image to hard disk as either an **OPU** or **OPD** file. Once safe in the hard disk, the floppy image can be browsed as any other preexisting floppy image.

Acknowledgements:
This idea of having a Windows program to read physical **Opus Discovery** floppies was in my mind for a long time, but with the absolute lack for a Windows low-level floppy driver it was near to impossible to do. Now, thanks to **Simon Owen**'s driver I finally did it, and even with a quirk or two I'm releasing it. There's always time for improvement and version numbers are free, but no advance can be done if no one else knows about the program. This program also "forced me" into writing a custom control for manipulating **ZX Spectrum** screen files (**ctlZXScreen_v103.Ocx**) with which I learned a lot. This guide has been "PDFised" with the freeware **PrimoPDF** printer driver. Of course, all brand names given here are © by their corresponding owners.

Disclaimer:
All efforts have been taken to make this program as unobtrusive as possible, as well as safe. It does not read your credit card numbers, does not have advertising, does not read your mind, pick your pockets or whatever. But as you might know by now, any Windows program can go bananas at any time for unknown reasons. What I mean is, the program is designed not to harm your computer, but if it does *I am not to be held responsible for any loss*. You're supposed to back your system up regularly, and it is your failure, not mine, if you don't.

About The Author:
This program is © 2006, zxMarce (zxmarce@datafull.com). Should you feel something's incomplete or wrong in this guide, please email me. You can also find me occasionally lurking in the forums of www.worldofspectrum.org as Marcelo.

Version Notes:

- **v2.0.0**: First public release. Known bugs are in the token parser, and code's Search.