STEEM SSE MANUAL

v4.2.1

Atari ST

The Atari ST was a legendary 16bit computer of the Eighties.

Beside the price, its forces were a no-nonsense graphical operating system (the GEM, that also existed on the PC), a flicker-free high resolution monochrome display for professional use, and MIDI ports. Thanks to the latter, to usable hard drives and to tight timings provided by the hardware, the Atari ST was for a long while the computer of choice for musicians and studios, running the best MIDI software available at the time, such as Cubase and Notator. In this domain, the ST was number one. In fact it is still used in some studios.

Its native sound is produced by the modest YM2149 chip, which is not too good at sample playback but had some good tunes composed for it, and even a cool speech synthesiser that has been used in commercial records.

The ST struggles on scrolling action games because its chipset wasn't dedicated to that sort of childish programs, it was a serious computer in the beginning. The STE version is better in this respect, in other words, less serious.

The ST has a great collection of games nevertheless, including action games, including scrollers.

Unlike computers today, the ST featured two useful keys: Help and Undo (mapped to Page Up and Page Down in Steem!)

Downsides of the ST were that earlier models were sold with single-sided drives, the OS in ROM couldn't be easily fixed, memory wasn't easily expandable and that bad business practices and viral piracy readily killed the platform, especially in the USA.

Steem

Steem Engine, by Anthony & Russell Hayward

Steem is a legendary Atari ST emulator for the PC, both accurate and easy to use thanks to its great user interface. Its development was stopped in 2004, at v3.2. The Steem authors probably got sick of all the lamers. In 2011 the Hayward bros released the source code for further development.

Steem SSE

Steem SSE stands for ***ST** Enhanced **EM**ulator **S**ensei **S**oftware **E**dition*. It is an updated version of Steem. Some say an improvement, some say it's worse!

Features have been added and emulation has been refined. Steem SSE's goal is to keep the Atari ST alive in the

hearts and minds of old and new generations (the former goal, being the #1 ST emulator, having already been reached).

Steem SSE Web site

https://sourceforge.net/projects/steemsse/

There are downloads, a blog, a forum, and the source code repository (SVN).

About the manual

This manual is based on the original readme.txt included with Steem 3.2. It's been updated to reflect all modifications introduced with Steem SSE.

This manual applies to the Windows build. Not all of it is relevant to the Linux build, which has fewer features and some differences. There is specific documentation for the Linux build in its package, please check the steem sse linux readme file.

The manual doesn't cover the debugger.

Licence

Steem and Steem SSE are free open source software. The licence is GPL3 as chosen by Steem authors.

Installation

There's no setup program. Extract files to a folder of your choice. Text (and RTF) files should be in a *doc* subdirectory and plugins (dll and other files) in a *plugins32* (32bit builds) or *plugins64* (64bit builds) subdirectory, although both types can also be in Steem's directory.

When you run a Steem SSE build for the first time, based on the file name, it proposes to add a link to the new version in your start menu if you care about that.

On starting, Steem SSE looks first in the application folder for your current steem.ini file, then in Documents/steem then in the registry and lastly in the roaming "AppData/steem" user folder.

If it can't find it, it asks you where it is or it should be created. All your settings will be recorded in that file.

Steem also needs a temp folder for archive handling and such. First it checks in the steem.ini file if the path is specified ([Main]TempPath). If not it tries to use "AppData/steem", if not writable (which would be surprising) the system temp folder (adding /steem), if not writable (very surprising) the place where steem.ini is, if not writable the folder where the executable was started.

You must also select a TOS file (see Getting emulation started). One is provided (default tos.img).

Builds

Four packages of Steem SSE are available. The Windows packages contain each four executables. For Linux, it's simpler, there's just a choice between 32bit and 64bit.

Windows 32bit

Steem32D3D.exe	Direct3D9	(main build)
Steem32DD.exe	DirectDraw7	
SteemDebug32D3D.exe	Direct3D9	with Debugger
SteemDebug32DD.exe	DirectDraw7	with Debugger

Windows 64bit

Steem64D3D.exe	Direct3D9	(main build)
Steem64DD.exe	DirectDraw7	
SteemDebug64D3D.exe	Direct3D9	with Debugger
SteemDebug64DD.exe	DirectDraw7	with Debugger

Linux 32bit

xsteem32

Linux 64bit

xsteem64

When there is a b at the end of one of those files, it's a beta.

Notice that you don't really need to run the 64bit Steem build on your x64 system, the 32bit build works too.

D3D builds are better for VSync and true fullscreen.

DirectDraw has been deprecated since long by Microsoft but it is very compatible with all sorts of configurations, while D3D may require an update. It is also convenient for virtual machines (for Linux users who want to use the Steem debugger).

Requirements

Of course you need a PC with a SSE2-compatible processor.

Windows builds require at least Windows XP and the 2008 C++ runtime. The D3D builds require DirectX 9.0c. If Windows tells you it can't find some related library, try to update your computer:

D3D 9.0 https://www.microsoft.com/en-us/download/details.aspx?id=35

Visual Studio runtimes (there are good chances they're already installed on your computer):

https://docs.microsoft.com/en-us/cpp/windows/latest-supported-vc-redist?view=msvc-160

32bit https://download.microsoft.com/download/5/D/8/5D8C65CB-C849-4025-8E95-C3966CAFD8AE/vcredist_x86.exe

64bit https://download.microsoft.com/download/5/D/8/5D8C65CB-C849-4025-8E95-C3966CAFD8AE/vcredist_x64.exe

Note: Microsoft regularly changes its site structure so links to runtimes may be outdated.

Third-party

Steem SSE works with various other programs or libraries.

-ArchiveAccess.dll (7Z, RAR, ZIP and more support)

http://sourceforge.net/projects/archiveaccess/

This is included in the download package.

-CAPSImg.dll (IPF, CTR support)

http://www.kryoflux.com/

-Drive image (ACSI hard disk image manager)

http://atari.8bitchip.info/drimus.php

-FreeImage.dll (for screenshots, though you already have PNG in D3D builds)

http://freeimage.sourceforge.net/download.html

If it's not up-to-date, it will not work well with recent versions of Steem SSE.

-msa.exe (converts files to ST/MSA disk images)

http://msaconverter.free.fr/

-pasti.dll (STX support), 32bit only

http://pasti.fxatari.com/

Check at atari-forum for the most recent version.

-stvl32.dll or stvl64.dll (low-level ST video logic emulation)

This plugin has not been released yet.

-unrar.dll or unrar64.dll (RAR and ZIP support)

http://www.rarlab.com/rar_add.htm

-unzipd32.dll (ZIP support)

You can find it in older Steem versions. 32bit.

-d3d9sweetfx.dll (D3D)

SweetFX shaders

The plugin will silently fail to load if it doesn't find the Microsoft Visual C++ 2010 SP1 Redistributable Package in your system (or in Steem's main directory). Contrary to VC2008, they're not included in every Windows 10 installation, don't ask me why.

VC2010 32bit http://www.microsoft.com/en-us/download/details.aspx?id=8328

VC2010 64bit http://www.microsoft.com/en-us/download/details.aspx?id=13523

You need the plugin for the CRT emu option. It is part of the download package, so no worry. d3d9sweetfx.dll is the name for Steem SSE.

-ShaderGlass

ShaderGlass also works with Steem SSE. There are lots of effects to check, including RetroArch shaders.

https://github.com/rohatsu/ShaderGlass

For best effect, disable the tool and status bars.

All the plugins can be stored in a *plugins32* or *plugins64* folder, it's the first place where Steem SSE will look. After that, it looks in a *plugins* folder, then, for DLL files, in a folder with the same name as the DLL (pasti,

FreeImage...). Finally in Steem's main directory. However Steem SSE doesn't control where those plugins look for their own files if any (can be in the app folder).

Steem SSE uses additional data files, that are included in the download package:

-Folder *DriveSound* with some samples. You can use your own samples instead.

-Folder *PRG* to help PRG/TOS files support, which should be copied into Steem's folder.

You can put some files in Steem's folder to replace internal resources:

HD6301V1ST.img, ym2149_fixed_vol.bin

if you have other versions you prefer (not likely).

Emulation limitations

Emulation is mostly high-level with some low-level elements where it helps most. Define low-level! Steem SSE has a "low-level" 6301 emulation, in that the chip's opcodes are emulated instead of the high level commands sent to the chip, but the emulation of that chip is higher level than Steem's 68000 emulation, which is "high-level." Contrary to what you might think, really low-level emulation of all chips composing the ST is not realistic on current PC systems, as of 2021.

Emulation of the Mega ST and of the Mega STE is partial.

The light gun is not emulated (hard to do on a LCD screen!)

STE digital sound filters are not emulated (should we?). MicroWire isn't accurate.

Only one mouse in port 0 is emulated.

The MC68901 MFP's *Pulse Width Measurement Operation* isn't emulated. Timer B emulation is pragmatic and a bit dodgy.

SCP: Can't write disk images (they're all read-only anyway). Protections with weak bits can be trouble.

GEMDOS hard drive emulation is not guaranteed to be compatible with all programs (it's a hack after all).

Disk DMA transfer timing is approximate.

...

Window menu

Press Alt-space for the Window menu, or click top left of the window, on the icon.

Restore Position is useful if some window was opened on another display which isn't currently switched on or similar circumstances. If Steem's main window is invisible, you can access the menu through the task bar.

When automatic resizing is off you may find it difficult to get the window to a precise size, making ugly lines appear.

Normal size will resize the window to the double (low res) size. Restore aspect ratio will maintain the current size of the window but will alter it so that its aspect ratio matches that of the selected size for the ST resolution. Also on this menu is Bigger and Smaller Window, they are a quick way to resize the window without going to the Options dialog, and border settings, which also affects the size of the window.

The window menu also lets you enable the tool bar and the classic menu, and start or stop emulation.

Menu and Keyboard

If you press the Alt key while emulation is stopped, this toggles the classic menu. Most options of Steem SSE can be changed with just the keyboard.

Getting emulation started

To use Steem, you need a copy of an Atari ROM with the operating system, called "TOS", and image disk files. A TOS file is 192KB long for the STF (except v2.06), 256KB for the STE. The extension is img or rom. When you first launch Steem SSE (no ini file), you must point to a TOS file, and you can change that later. One important point is to match ST model and TOS (see further), for example STF + TOS 1.02, STE + TOS 1.62. In Steem v4.2 (and some older versions), TOS v2.06 (from Atari or from EmuTOS) works with the STF too.

Use the disk manager or a shortcut to select the disk image, there are various ways to start emulation.

Disk images are files that contain the data of a floppy disk. They are easy to find. For example:

http://planetemu.net/index.php?section=menus&machine=34

http://www.atarimania.com/

Unfortunately, full archives of ST programs are less available than they used to be.

TOS:

http://www.avtandil.narod.ru/tose.html

Legally this is still copyrighted material, and if they really wanted, companies that own it could call a halt on emulation, but they don't.

To get you started, a legal EmuTOS image is included in the download as the default tos.img.

Tip It crashes on accessories (ACC) and doesn't load DESKTOP.INF. Other than that, its compatibility is great. Both GEMDOS and ACSI hard drive emulations work with it. It works on both the STF (because we simulate hardware hacks!) and the STE.

Tool bar

The tool bar is optional. On it you find some useful icons; from left to right:

1st group on the left

- Reset
- Play
- Fast forward
- Memory snapshot
- Screenshot
- Paste

2nd group on the right

- Disk manager
- Joysticks
- Patches
- Shortcuts
- Options
- Debugger (Debug builds only)
- Info

If icons are too small on your screen, there's a way to make them bigger (startup options).

Some words of explanation are given for each icon.

Reset

A left click on the reset button emulates the reset button of the ST (warm reset).

Notice that some demos were programmed to react on reset (hidden screen and such).

A right click emulates switching off, waiting as long as necessary for the RAM chips to clear (which could be minutes on a real ST), switching on again, except it's instant (don't think we'll get requests for precise delay).

Keyboard shortcuts may be assigned to warm or cold reset as well.

Play

You start/stop emulation by using the play button or the **F12** function key.

If you right click on *Play*, you advance in slow motion. The speed can be set in Options/General. For technical reasons, sound still plays if option *Emulation thread* is active.

When emulation starts, the mouse cursor may be grabbed by Steem. You may change this (tricky) behaviour in the options. In that case, the mouse still controls Windows, and you can stop emulation by clicking on *Play* again. If you click in the window, the mouse will be grabbed in *auto* mode. Once it is grabbed, you may wonder how to free it? **F11** toggles mouse capture.

So **F11** and **F12** are the main Steem SSE control keys. They were chosen because the ST had function keys F1 to F10. Yet you may change this in the options, it is possible to use **pause** and **shift pause** instead.

Fast forward

Use this to skip boring intros. You may still expect accurate emulation. Steem uses more CPU power.

Right clicking on this also activates the *lightsearch* (check if there is some picture hidden by colour definitions).

If you double click on this button (with the left or right button) it will stick down, so you don't have to hold down the button for particularly long waits.

For technical reasons, sound still plays if the *Emulation thread* option is active.

Memory snapshot

The camera and chip button on the Steem toolbar brings up a menu from which you can load and save memory snapshots. Save snapshot will save out the current state of the ST (not just RAM) to a file.

This can be useful for games without save facilities.

When you load a snapshot it will change your TOS version, monitor type, memory size and the current disks in the drives. The last ten snapshots that have been used appear also on this menu so you can quickly reload

them.

There is also an option to undo the last snapshot load or reset in case you did it by mistake or it failed.

Tip Memory snapshots of 32bit and 64bit builds are not mutually compatible

Screenshot

Use the right mouse button to choose your options and the left one to record a screenshot.

The D3D builds of Steem SSE can natively save a screenshot as a BMP, JPG, PNG or NEO file.

The DD builds of Steem SSE can natively save in BMP or NEO format.

With the FreeImage.dll plugin, more formats and options are available (hint: use the last version).

PNG is the recommended format because it is both small and lossless. Notice that JPG is a lossy format, only appropriate for a very colourful screen.

NEO is the format used by Neochrome on the ST. It is only appropriate when there is no overscan and no palette trick.

It is also possible to save multiple screenshots in a row using a shortcut.

Note: Recording the video

Although you can assign a shortcut to internal ST video recording (DirectDraw builds only), this isn't recommended as it is far too slow. It is better to use a third-party tool or Windows 10's own recorder (Windows key + G), which efficiently produces MP4 files.

Paste

This button allows you to quickly and easily paste text from Windows into the ST. Just click on the button and Steem will type in the text at incredible speed. Some programs may struggle with the top speed, to slow it down right click on the button and choose a longer delay.

Disk manager

The icon toggle Steem's disk manager (see below).

Joysticks

You can assign keys or real (PC, USB) joysticks to ST joystick ports 0 and 1 and to STE joysticks, and even to parallel port joysticks.

Tip Most games use standard port 1 for one player game

Tip If the standard port 0 is selected, the mouse is disabled, as on a real ST

Tip Key definitions for joysticks may interfere with keyboard emulation

Steem handles analogue as well as digital PC joysticks.

Six setups may be recorded. In each, joysticks may be always active, never active, or active or not according to the state of your PC keyboard Num Lock and Scroll Lock keys. If those keys are not on your keyboard, you may assign shortcuts (Toggle Joystick ... select) instead with the same effect. A shortcut can also select the setup.

For "up" you have 2 boxes, so you can map a joystick button for jump as in arcade games, in addition to the joystick movement. Unfortunately, ST joysticks only had one button.

F11 or Pause to remove a key selection.

Patches

Patches (file extension STP) change the memory of the ST after a program has been loaded to change its behaviour.

This is useful to go around a bug or a protection, cheat in some game, etc.

Tip You may use the debugger to probe the ST program and create your own patches with a text editor

Shortcuts

You can assign key combinations (up to 3 keys each) to either some Steem control (start/stop, fullscreen, etc.) or some Atari ST key.

Not only keys are taken as shortcuts but also mouse and joystick buttons.

You can load and save as many key sets as you wish and use several of them at the same time.

To avoid overwriting your changes, it's advised to name your own sets rather than edit the Default set.

Tip You can scroll the list using only the keyboard when the focus is on a Del button. F11 to remove a key selection.

Settings

The icon makes the extensive options dialog of Steem (see below) appear or disappear.

Debugger

This icon, an ST bomb, is only available in the Debug builds. It toggles the Debugger windows. For some information about the debugger, consult the Debugger release notes. It is not really covered by this manual.

Info

The blue 'i' icon opens an information box that can display this very manual and other useful documentation, and the *TRACE.txt* debug file if it exists. Pressing F1 when the emulator is stopped also opens that box.

Status

A page in this window displays program statistics and the current state of emulated chips (68000 etc.)

Most figures are reset at each run (from start to stop by using the icons or F12 or some shortcut), those that are not are displayed in deep blue.

- The 4096 colour palette is an STE feature, but notice that a program written for the STF may accidentally use the extra STE bits, and the STE TOS itself uses that extended palette on boot. TOS also sets up default MicroWire settings, and it uses the video timer B on boot. So you need to be careful when interpreting statistics, that's why some are split between TOS and program use.
- Computing statistics may take a while if the current frame contains many colours. For this reason, they are computed only when the *Status* info page is active (or when you open it). The colour count comprises OSD if present and maxes out at 4096.
- The reported CPU usage, in %, can be higher than in the task manager if your CPU is throttled.

Status bar

Steem SSE optionally features a status bar.

There are 4 parts.

1. Frequency/mouse coordinates
   If VSync is engaged, the first figure is for the ST, the second for the current PC screen.

   Mouse coordinates are displayed if the ST address where they are stored is made available to Steem SSE. There are two ways to do it: By specifying MouseAd in an ini file, or by using the *emu detect* capacity on ST address $ffc10a.
2. Useful info
3. Configuration info and icons
4. CAPS NUM SCRL state/disk info
   CAPS NUM SCRL info is handy if you don't have leds on your keyboard (remote keyboard, laptop...)
   Disk info is an alternative to the OSD option.

Tip Multiple status bar refreshes due to drive operation can be a drag on the system

Steem SSE may use the status bar to communicate some funny issues such as...

HALT

In some conditions, the ST processor (MC68000) can *halt*, that is stop operating.

It generally means the program it was running crashed, and it's not possible to recover from this situation. This may be an emulation bug or, more likely not (bad disk image, buggy ST program, etc.) The ST won't run again, it must be reset or rebooted.

Tip On real hardware, the rest of the system may sometimes still run somewhat, but this is not emulated in Steem SSE because it would uselessly add load to the emulator.

BLIT ERROR

Problem with video rendering. In some cases, Steem SSE can recover, in others you have to leave or even use the task manager to kill Steem SSE. Sorry.

It happens regularly when you stop emulation. In that case it isn't important.

If you see GDI on the status bar, it means DirectX stopped working and you'd better restart Steem SSE. Those issues mostly happen with fullscreen endeavours going wrong, or power management interfering. Or maybe the *No DirectX* startup option is checked.

6301 CRASH

The keyboard controller (6301) emulation hanged.

SYSTEM ERROR

Steem SSE caught a system exception (application crash). The most prudent approach is to leave, but in some cases, you can resume emulation. If you quit while this message is showing, Steem SSE won't attempt to save the configuration in *steem.ini* in order to avoid file corruption.

Settings

Steem is a very complete emulator and there are really many options, spread on several pages (or tabs). All are explained here, tab by tab.

Tip In Steem SSE you can change almost all options live, that is when the emulator is running. This is handy but not guaranteed to work all the time. After all, buffers are created, moved, deleted on the fly, and changing an option could alter a buffer while it is being used.

This is especially the case when the *Emulation thread* option is checked: you really have two threads running on the same data set, and there are only a few locks in Steem SSE. Freezes and crashes when changing display and sound are possible.

As from Steem SSE 4.2.1, you can move the setting tabs up or down by just dragging them.

Startup

You need to quit and restart Steem for those options to have effect.

Only one Steem SSE

If you want to start Steem SSE (v4.1+) again, directly or by launching an associated file, it will go to your current instance instead.

Big GUI

This setting may help for high DPI screens. You can also change the font size, though the GUI was designed with the defaults in mind.

Tip Since v4.1, Steem SSE disables Windows scaling by default. If you wish you can override this by changing the DPI properties (right click on the icon, properties, compatibility, high DPI settings, check *High DPI scaling override*, select *Scaling performed by: System*).

Restore Previous State

You can choose to have Steem remember the state of the ST when it quits, so you can return to what you were doing next time you run it. Below the option is a box where you can type in the filename that will be used to save the state, this file will be saved in the directory Steem is in.

Tip Sometimes, the file gets corrupted and makes Steem SSE fail, possibly crash. The solution is to delete it (the file).

If this option isn't checked, Steem SSE will load and display the startup.jpg picture instead. It looks for one in your settings path then in the application directory.

Restore window positions

If you like your windows to appear at the same positions you left them.

Run on startup

I did it to compare CPU use of instances of Steem SSE starting at the same time, but it can be handy in other cases.

Start in fullscreen mode

If DirectX is enabled and working, then you can make Steem fill the whole screen when it first runs, to impress your friends.

Lock window size

When it's checked you can't change the size of the window by dragging the borders so that you won't be able to sabotage Steem's nice display.

Never Use DirectDraw/Direct3D /DirectSound

If you have DirectX errors when you boot up, or DirectX doesn't work properly, then you can tell Steem to not use it. Sound doesn't work without DirectSound and drawing is slower without DirectDraw. If you are having problems, it is best to install the latest version of DirectX through Windows Update.

Tip Really it shouldn't happen, this option was more useful in the time of Windows 95 when Steem first was released. What could happen is that those options are checked by accident. People using Wine to run Steem (for example the debugger, that is Windows-only) may also prefer GDI rendering.

Check for updates

If ticked, Steem SSE checks at sourceforge if a new build is available. If that's the case, it proposes to download it. It is based on the file name.

TRACE.txt

If checked, Steem SSE will write some debugging info in that file in its running directory. It will also dump your options (as options.txt) when you close it.

Tip This option doesn't exist in beta builds, a trace file is always produced.

If Show paths is checked, full paths are traced, if not only the file names (privacy concern).

Quick settings

New in Steem SSE 4.2.1, various settings are grouped in one page for convenience. You can change the ST model, RAM, ROM, the screen options, switch on or off the hard drive and even select the floppy disk currently in drive A:

Tip To enable TOS versions, you just need to click them once in the TOS (ROM) page. After that if you select an ST model, a default TOS will be selected (like TOS 1.0 for STF), which you can change.

General

Reset settings

Restore settings to generally compatible values.

Tool bar

The tool bar can be hidden, in case you don't like it or you need Steem to blit on the full client area of the window. It's a technical point with practical implications: third party tools like ShaderGlass work with the client area, for example.

Menu bar

You can control Steem SSE with the menu instead of the tool bar.

Status bar

If you don't want the magnificent status bar, this is where you can toggle it.

There's also a TOS flag checkbox for the nationalist.

Save backup snapshot on reset

Notice the operation can cause a glitch during emulation.

Show pop-up hints

This little option toggles those pop-up hints on and off; now you've read the manual you don't really need them.

Make Steem high priority

When this option is ticked Steem will get first use of the CPU ahead of other applications. This means Steem will still run smoothly even if you start doing something in another window at the same time, but everything else will run slower. It's at your discretion which processes are more vital on your computer. The spreadsheet or Steem?

Tip This does make a difference for a stable frame rate when your computer gets busy (internet browser, disk activity...)

Emulation thread

This option could make Steem SSE more responsive. Steem gets a tad more system resources for its supplementary thread. When it's checked, you may change any other options you like live (emulation running), but there's no guarantee it will not crash!

Noticeably, with the apart thread, emulation isn't stopped when you move any Steem window and the sound doesn't loop. When the emulation thread crashes or stalls, you can kill it using F12 or the classic menu.

Tip The shortcut for double clicking in the ST requires this option

Tip With the debugger, effects are unpredictable

Pause emulation when inactive

When this option is checked Steem will pause emulation when you switch to another program. Steem uses some PC CPU time when running which may slow other programs down, if you are switching between programs regularly you may want to use this option.

Disable system keys when running

This option allows you to send certain key combinations that are used by Windows to the ST instead. When it is checked Alt+Tab, Ctrl+Esc and Ctrl+Alt+Delete will go to the ST instead of to the PC. Although Steem is very stable this option could cause a bit of a disaster if Steem stops responding for any reason, you won't be able to shut it down with Ctrl+Alt+Delete, so for safety reasons it is best only to use this option when you really need it.

Unfortunately, this option doesn't always work in fullscreen mode. Depending on your version of Windows, it may not be possible to capture certain key combinations - you can use shortcuts (see below) to trigger the keys you want from other keys.

Start emulation on mouse click

When checked clicking on Steem's main window will start emulation.

Display

Frameskip

This is a very useful option for slower processors. Drawing the screen takes quite a bit of PC CPU time so skipping frames is a very effective way to speed things up to the same speed as a real ST. Auto frameskip makes Steem skip up to 8 frames dependant on how fast it is running.

If possible, *None* is the best option because auto frameskip always induces some glitches.

Borders

Here you can choose whether you want to see the ST borders, and which size.

Some games (and quite a lot of demos) use a technique called "overscan" to display graphics in the borders, that's one reason you might want to see them.

DirectDraw. The border option affects the fullscreen mode: with borders on, Steem displays fullscreen in the 800x600 screen resolution (flip screen or straight blit modes).

Steem SSE offers no less than 5 border sizes:

Resolution	Low	Medium	High
Border			
Off	320x200	640x200	640x400
Normal	384x270	768x270	768x540
Normal (US TOS)	384x242	768x242	768x484
Large	412x275	826x275	–
Max	416x283	832x283	–

The normal size for the US TOS has fewer scanlines because the ST typically operates at 60Hz. This is applied whenever Steem SSE is resized.

Why various sizes, while other emulators only offer one? There are many reasons.

- Bigger display sizes in Steem SSE use various tricks and hacks and there's always a risk something is broken, so we keep the original size available. Palette change timings were especially trouble during development.

- There's a performance hit for larger displays.

- Not all real monitors display the full border, in particular modern LCD screens or the original Atari monitors (SC1224) would typically display the same borders as Steem does in normal mode, therefore it is most realistic.

- Normal size may be merciful when the program displays trash in the border, typically because the programmer couldn't see it.

- Only the normal (or no) border makes sense for monochrome emulation.

- Most programs, including games, don't display video in the borders at all, and they shouldn't as this is beyond specs.

Stretch Filter (Direct3D)

One of the standard D3D filters is applied when the image is stretched. It can blur the picture a little and reduce distortion. This setting applies to both windowed and fullscreen modes.

Tip Even if you don't explicitly select a stretched display option, Steem SSE may need to stretch the picture to make it fit in the window

Timing Loop

Steem SSE will start a hard polling loop so many milliseconds before the planned time of blit (video rendering). A higher setting means more CPU use. Generally 2ms seems to be a good compromise (1ms is the previously hardcoded setting).

However, if your PC is very busy, no matter how high the setting, Steem SSE may lag. The general option *Make Steem high priority* has a greater impact here.

This setting is ignored when VSync is engaged.

Scanline buffer

Steem SSE will prepare each scanline in an intermediate buffer before copying it to the video buffer (wherever it is, controlled by the driver). The effect on speed and CPU use may vary according to the system and what Steem is emulating, usually unchecked is more efficient, checked more stable.

There's also a difference in the way optional scanlines are rendered.

Microseconds

Use precise hardware counters to time video rendering, if it's not checked milliseconds are used. Microseconds allow emulating the real video frequencies, which are fractional. Timing of display still depends on the hardware and OS (better with FreeSync).

This setting is ignored when VSync is engaged.

Old sync

Use this if your computer has trouble with the way recent versions of Steem SSE handle video synchronisation and you feel nostalgic for good old Steem.

When it's checked, settings for microseconds and the hard loop have no effect.

Aspect ratio

The typical ST screen's pixels were vertically stretched, this option emulates that aspect ratio. The 60Hz mode on a 50Hz screen, very stretched, is emulated if borders are disabled.

Scanlines

This makes Steem display black lines between actual video lines ; this may or may not be quicker than doubled lines but looks pretty ugly to some people. To others, it may be part of the emulation. Some people insist on crisp scanlines, others want filters and interpolation.

Pixels

The same as scanlines, but vertically.

CRT (D3D)

Emulates a Cathode Ray Tube screen (maybe the youngest people have never seen a real one!) using SweetFX

by CeeJay.dk, which should be in your plugins folder, under the name of d3d9sweetfx.dll.

You can change all parameters, but not inside Steem. In Steem SSE, there's just this option to toggle the effect, keys won't be transmitted to the plugin (depending on the settings).

If you don't like that integration, it's possible to use SweetFX the usual way, Steem D3D was already compatible with the hack.

This option is only available in D3D builds, and uses the power of your video card, not the CPU.

Tip Creating the surfaces and switching to and from fullscreen takes some time, don't panic

BFI (D3D)

BFI stands for *Black Frame Insertion*. Steem SSE will update the display at twice the ST rate, one frame will be black, one frame will be the emulated image. It results in less movement blur, but video gets dimmer and some flicker may be perceptible.

This option operates in windowed and fullscreen modes, with or without VSync, but better with VSync. The screen must be set at a high frequency or in FreeSync.

Triple Buffering (DirectDraw)

This used to be useful to eliminate tearing or ghosting, at the price of quite higher CPU use in windowed mode on Windows XP.

Window compositing, a feature introduced in Windows Vista and enforced (can't disable) in Windows 8, could interfere with this. Compositing does its own triple buffering, without CPU use penalty.

Generally, this option isn't necessary anymore.

Windowed Mode

Four sizes are available. With *Show all settings* option, you can choose for each resolution.

Stretch makes Steem use the video card to convert the image from the ST output size to the size of the window. No Stretch will make Steem draw the ST screen larger in order to avoid stretching.

If *biggest* is still too small for you, it's possible to just make the window bigger by stretching it the usual way. Check on the status bar for a precise multiplier. To keep the size through session, uncheck the following option.

Automatic resize on resolution change

There is also the option of whether Steem should automatically resize its window or not when the ST resolution changes (also on reset, starting Steem).

This can have an impact on performance if an ST program changes resolution all the time.

Lock aspect ratio

With this you may drag all you want, the window will keep the same proportions. Steem SSE also keeps the normal aspect ratio for all window sizes, which may introduce blur or artefacts. For instance, the *big* size in medium resolution (corresponding to x3 in low resolution) uses a non integer multiplier.

If the option isn't checked, you're free to change the aspect ratio of the emulated screen and some window sizes have a different aspect ratio. For instance, the medium resolution may be twice larger horizontally.

VSync

This version of vertical synchronisation is for windowed mode.

DirectDraw: Steem uses some CPU power to adjust speed.

Direct3D: Speed is adjusted at no CPU cost by the system.

Speed adjustment means that emulation speed isn't realistic anymore if the display refresh rate significantly differs from the ST rate. Note that even if your hardware displays at 50, 60 or 72Hz precisely, the real ST has a slightly different refresh rate based on its main clock.

VSync works with double frequencies (e.g. ST 50Hz/PC 100Hz).

Tip The ST can operate in 3 frequencies, 50Hz (PAL, "Europe"), 60Hz (NTSC, "USA") and 72Hz ("Monochrome"). Some programs change the frequency all the time to trick the hardware, but hopefully there's still one main frequency, generally 50Hz.

Tip It may depend on hardware/OS, but VSync only works on the primary monitor (main display), even if Steem is on the second monitor. Notice that it is very easy to switch the main display.

Tip For a long time, LCD screens defaulted to 60Hz and had no 50Hz mode. It was a miserable period for emulation. Thanks to the HDMI requirement, a lot of screens now do 50Hz. Steem SSE can also use 100Hz as if it were 50Hz.

Tip Many screens and video cards now offer G-Sync or FreeSync. With this you can get smooth animation at every ST frequency without changing your display settings. You can check that with the early Automation menu disks (0 for 60Hz). It's still not always as smooth as VSync, for various possible reasons, but the emulation speed is more realistic.

Tip If you use G-Sync or FreeSync, don't check VSync

Tip Process priority (general options) makes a difference

Tip Don't use VSync if your screen frequency is no match, it will alter emulation speed

Auto VSync

With this option, VSync will be enabled only when the main display refreshes at a frequency close to the ST frequency (+-2Hz).

Screenshots
These options configure where screenshots are saved.

You may also choose the format options here.

The last option is *Minimum size screenshots*, when checked this will make Steem always save screenshots at the smallest size possible for the resolution, this is handy for people taking screenshots to put on a website.

Warning: Sometimes this might make a mess as Steem will have to shrink the picture, but this is never a problem if you are using *Stretch* mode for drawing (see the Fullscreen options and Display/Window Size options).

Sound

Device
This lets you configure which DirectSound driver Steem will use.

Format
You can change the sample rate. If your soundcard isn't able to deliver one of the higher rates, Steem will settle for the maximum available. It makes a difference in some cases where the emulated PSG vibrates faster than 22KHz. It does happen, as the max PSG frequency is 125KHz! Most audio equipment can't reproduce above 22Hz, and your aging ears go even lower.

Note that 56KHz is precisely 56066 (the maximum, and rarely used rate for STE sound), 25KHz is precisely 25033 and 22KHz is 22050.

16bit stereo is recommended, except for some rare STE demos that insist on mono sound.

Mute
You can change it live, hopefully it won't crash.

when inactive
Sound will stop when you're busy with other apps, and play again when Steem has the focus.

Vol
That's volume.

YM2149 Sound
The YM2149 was the original sound chip of the ST. It is also called PSG for *Programmable Sound Generator*.

Low-level activates a lower level YM2149 emulation inspired by MAME (thx Couriersud). When it's checked, Steem SSE also uses a big sample table of the chip made up by ljbk based on measurements on an actual STF (which includes filters). Thanks to this accuracy of PSG emulation is much improved. The sound is more saturated, which is correct.

Vol allows you to adjust the volume of the chip, which is a hack. It's expressed in bits. The maximum of 16 bits may saturate on some tunes.

You can then set the anti-alias filter frequency, it should be somewhere around 10KHz (the ST was no CD player). Totally right turns it off, which will produce aliasing artefacts on some tunes (except if you set the sample rate very high).

When *low-level* isn't checked, Steem's original PSG emulation runs. It is not too bad and uses less CPU. You may then choose a filter type. Those filters are primitive but do the job.

Digital sound (STE)

It is generally called DMA sound, also by Atari, but DMA (Direct Memory Access) on the ST is a procedure that uses bus arbitration, for disk operations and for the blitter, so it's called Digital sound in Steem SSE!

Vol allows you to adjust the volume of digital sound, which is a hack.

Unfortunately, emulation of the ultra-cheap analog MicroWire tone controller is rather primitive.

Slow fade is a hack which may avoid pops when the program changes the volume too fast (*Sea of Colour, Talk Talk 2*).

The bass and treble shelf settings adjust the filter frequency. Bass and treble proper are set on the GEM Control Panel.

Drive sound

Yes! This SainT feature is imitated in Steem too.

Record

These options allow you to record Steem's output to a wav file. Clicking on the red button begins recording, clicking on it again (or stopping emulation) will end recording. You can choose the name and location of the wav file to output by clicking on choose.

Alternatively, Steem can save a YM file. This is usable by STSound and other utilities. The file is uncompressed.

GEM Control Panel

Imitation of the GEM Control Panel

The full control panel is a hack giving you direct control on some ST internals.

Note that if the ST program changes the palette, the panel is updated. This can be demanding on processing power.

Block palette

This will lock all ST colors to their current palette definition

Clock always correct

When this option is checked, the various ways the system can get the date and time will be hacked to report the PC time (the little clock in your taskbar). So that if you stop emulation for a while, you still get the correct time when you resume, and not the emulated time. This covers the Gemdos software clock, the Mega ST Ricoh clock and the 6301 keyboard chip. This option can interfere with some programs.

Emu detect

Steem includes a facility to interact with ST programs that want to communicate with the emulator. If you uncheck the option, this facility is disabled. It's still possible to detect Steem using emulation inaccuracies, but this way is unreliable since further versions are supposed to be more accurate (right?)

The ST program writes $456D753F (ASCII for *Emu?*) in both D6 and D7, and calls xbios function VSync ($25).

Steem returns with $53544565 (STEe) in D6, $6D456E67 (mEng) in D7 and $FFC100 in A0, which is the base of *secret* addresses used for interaction.

Read

$ffc100 Steem version

$ffc102 slow motion

$ffc104 fast forward

$ffc106 Debugger present

$ffc107 snapshot loaded

$ffc108 emulation speed

$ffc10a average frame time

$ffc10c CPU cycle counter (32bit only)

$ffc110 time of last vbl (32bit only)

$ffc114 time of last hbl (32bit only)

$ffc118 scan_y (scanline)

$ffc11a write logs to printer

$ffc122 runstate (Debugger, option Emu detect necessary, but no need to write Emu?)

Write

$ffc100 create disk image from source (effective address) string:

Name, Sides, TracksPerSide, SectorsPerTrack, each ending with /0

$ffc104 leave emu detect

$ffc105 set cpu speed in mhz

$ffc107 snapshot loaded

$ffc108 set run speed (in %)

$ffc10a write byte

- $4D ('M') informs Steem that D5 contains the address of mouse coordinates, to be displayed in the status bar
- $51 ('Q') to quit Steem
- $52 ('R') to reboot (cold reset)
- $51 ('r') to reset
- $53 ('S') to stop emulation

$ffc11a write logs to ST printer (Debugger)

$ffc1f0 write source string to log (option Emu detect necessary, but no need to write Emu?)

$ffc1f4 wipe log (Debugger, option Emu detect necessary, but no need to write Emu?)

$ffc123 trigger breakpoint (Debugger, option Emu detect necessary, but no need to write Emu?)

Fullscreen Mode

Go Fullscreen Now/Go Windowed Now

The top button in this page allows one to toggle between windowed and fullscreen modes. You may also assign shortcuts to enter or toggle fullscreen mode (alt-enter also works).

Fullscreen on Maximize Window
You can change Steem's behaviour on hitting the maximize button.

Fullscreen GUI
The fullscreen GUI is handy if it works. No guarantee. Blame Microsoft and video card drivers for all bugs.

Tip On Windows 10, it doesn't work well in D3D builds for true fullscreen, and when it hangs, it's hard to come back to a normal desktop. The option is disabled for true fullscreen modes, but this can be overridden in the ini file for those who have an older OS (see INI file options).

If the option is checked, you can leave fullscreen mode by clicking on the blue icon (second from the right) of the interface, or pressing alt-Enter.

Stretch (D3D)
Choose between the two rendering modes for fullscreen. Stretch allows you to use a filter. Non-stretched has smaller OSD.

VSync
This version of vertical synchronisation is for fullscreen, including Windowed Borderless. See the remarks in *Display*.

Auto VSync
See *Display*.

Triple Buffering
One more buffer is used in video rendering. It could fail on some systems. The exact way it works (or not) depends on your system, generally it's a FIFO queue, which means an extra buffer may increase latency.

Aspect ratio
Screen: resolution of your PC monitor, **correct**: magnified resolution of the ST, **crisp**: could be (much) smaller but no distortion.

Mode
The option is different in DirectDraw and Direct3D builds.

In DirectDraw builds, this is a bit of a confusing option, you really don't need to know what the possible settings mean, but which is best will be different depending on your computer's hardware, try them all to see which is quickest. *Stretch blit* makes Steem draw small and then use the video card to stretch the resulting image to fill the screen, at your chosen resolution. This can be slow on very old, bad video cards, and on many

video cards it causes the image to be slightly blurred ("alpha-blitted"), which makes it better resemble the bad televisions usually plugged into STs. If this blurring annoys you, switch to the first modes. In those modes, colour pixels are doubled and Steem uses the 800x600 resolution if borders are displayed, 640x480 if not. Screen Flip is not recommended for monochrome monitor emulation if the screen refresh rate is inferior to 72Hz. In the DirectDraw builds, Windowed Borderless is one of the listed modes. In Windowed Borderless mode, Steem SSE uses the native resolution and its own calculations of the drawing zone. In medium and high resolutions, the picture could be crisp but small, or big but distorted. It's because there are already many pixels in these ST resolutions.

Tip Try with border off

In Direct3D builds, the option isn't available when the Windowed Borderless Mode option is checked. This lists all D3D 32bit fullscreen modes that are available on your video card. You just select the one you like. Note that whether a mode works or not also depends on your display and your other options. Only some monitors will display 320x200 or 320x240 properly. On a LCD screen, you may go for the maximal resolution (the same as your desktop). Steem SSE will then compute how to display the biggest possible ST screen for the aspect ratio.

All true fullscreen modes require some time to adjust to the settings. It's a Windows feature. In D3D builds, true fullscreen on a secondary monitor is possible.

Scanlines (DirectDraw)

When you are in *Straight Blt* or *Screen Flip* drawing mode, *Scanlines* makes every other vertical line black, this can make the display a bit dark but it can also make Steem run quicker.

Use 640x400 (DirectDraw)

This option makes the screen switch to 640x400 pixels when you run in fullscreen. This means the ST display will fill the entire screen instead of having small borders to the top and bottom. You can only use 640x400 mode when borders are set to *Off*, and if the mode is available on your video card.

Use desktop refresh rate (D3D)

This will bypass the Hz setting in Mode, using the current desktop refresh rate instead. It can be useful if the video card can force a refresh rate that is not officially supported by the display (or its driver).

Windowed Borderless Mode (D3D)

This 'fake' fullscreen mode bypasses most other settings. You lose the benefits of true fullscreen, such as refresh rate choice, hardware lower resolution, triple buffering and possibly better performance (Steem isn't as demanding as a 3D game so no worry), but it may be more compatible with a variety of computers. It is also faster to go in and out.

Colour Control

Some PC monitors/video cards can be very dark, this can make the ST display unrealistic, or even make some

things that should be visible black. Here are some simple options to fix this problem, just fiddle with the two values until in the colour bar display you can see colours above the number 2 and the colours above 15 and 16 are a different shade. Sometimes other colours on the screen can make this difficult, to get around that click on the image of the bars to make it fill the whole screen.

Steem SSE also includes controls for gamma correction.

Reset

The power of computers, no need to reset everything yourself.

Full spectrum

We adapt the scales to get brighter colours on both the STE (4096 colours) and the STF (512 colours). It's not a trick, we just extend the values used on the PC side.

B/W

Black & white screen using the greyscale technique.

Green

A nostalgic emulation of the Philips CM8833 Green button.

Keyboard/Mouse

Chip HD6301

Low level 6301 emulation (C1)

Using code by Arne Riiber and doc by Stephane Catala - Belzebub / ST Connexion, Steem SSE features low-level emulation of the HD6301 *intelligent keyboard* (IKBD) chip. The main goal was to be able to run some (very rare!) software reprogramming that chip but it does more than that now, being better than high level emu in some cases. When you change this option, you should reset the ST. In case you're curious, the titles that reprogram the chip are: *Audio Sculpture*, *Dragonnels*, *Froggies over the Fence*, *Transbeauce 2*.

The option also commands ACIA (the MC6850 chip) improvements, mainly more precise timings. Both the IKBD and the MIDI connector use a 6850 to communicate with the CPU, though at different speeds.

Battery

The keyboard chip can be used to keep time, and when equipped with a battery, keep time when the ST is switched off too. There's an option for being 2000-ready, which original hardware wasn't. Ticking this option off may be more compatible as most STs had no such battery.

Keyboard

Here you can configure the keyboard. This is an area that many people have problems with because most non-English PC keyboards are vastly different from their ST counterparts. The important thing is that your PC keyboard language, TOS language and ST keyboard language all match, that gives you a chance of having the right characters appear for the right keys.

When the *Shift and alternate correction* option is checked this makes Steem do its best to get round the differences between ST and PC keyboard layouts by fooling the ST into thinking you are pressing keys you aren't. This works well for GEM programs but it could cause problems for games and any other program that reads the keyboard directly.

The last choice in the language list, *Mapping file*, allows you to select an ini file that contains key definitions. Definitions are included in the package for the German and the French keyboards, which should be the most used beside English. Also included are definitions to use US or UK keyboard on UK or US TOS, it could be handy.

The point of the feature is to allow you to create your own file in case you're not satisfied with Steem SSE's hardcoded mapping.

The file format is as follows.

```
# = comment, at start or end of line
entry format: 5 fields, each followed with a ; and return
1;22;33;44;
1: PC key to type (1 character)
2: ANSI code of PC key to type (2 characters)
3: Alt (a) and Shift (s) modifiers of PC key to type, if any (2 characters)
4: ST scancode, hexadecimal without header (2 characters)
5: Alt (a) and Shift (s) modifiers of ST key (2 characters)
The length of each entry must be the same, insert spaces where appropriate
The ; separator must end each field including the last one
Empty lines allowed
You may type the PC character, the ANSI code, or both. If there's a code,
Steem uses it. If not it uses the character. Both ANSI and UTF-8 encoded
characters should work, but the code, if any, must be ANSI (Windows-1252
actually), not UTF-8.
When you need to press shift or alternate (Gr) on the PC, field 1 contains
the 1st character of the key, not the character you want to see.
For instance, on a German keyboard, you need to press Alt Gr and < to
```

get |. The entry is: <; ;a ;2b;s ;

In Steem, you don't use Alt Gr, you use Alt instead, like on a ST.

Both ANSI and ST scancode are two digit hexadecimal numbers.

Please share it if you get it working. The file may be in Windows-1252 or UTF-8 format (unicode).

There is also the option to manually remap the keys using shortcuts (see below).

F11 F12 Emu control

When checked (default), F11 toggles mouse capture and F12 starts and stops emulation. If you prefer other shortcuts, uncheck...

Pause Emu control

When the option is engaged, Pause will toggle emulation run, and Shift Pause will toggle mouse capture. Note that the emulated ST will get the Shift key, it's a problem you don't have with F11 (F11 and F12 don't exist on the ST keyboard) .

Click (Linux, hack)

People like some feedback when they type. Atari thought about it, but instead of equipping the ST with expensive clicky keyboard, they provided the sound in software. Note that the Mega ST had a mechanical keyboard, excellent but not clicky (Cherry MX Black).

The sound may be disabled in the control panel, but this isn't always available (games that must boot the system...)

So in Steem SSE you can disable it directly. In some cases it should remain enabled though (see Hints), so the best is to disable it only when it's annoying you.

In Windows builds, this option doesn't exist anymore, you can switch the click with the control panel (but the option isn't saved).

Mouse

Capture

Off: the mouse isn't captured unless you press F11 (even if you click in the window).

On: Steem SSE grabs the mouse when running, frees it when stopping. F11 can free the mouse.

Auto: Steem SSE grabs the mouse if it was captured at the last stop. F11 toggles and clicking in the window grabs the mouse.

Auto-release: it's like auto but the mouse will be released each time a program is run. Useful if you need the mouse only to launch the program in GEM.

VM-friendly

This option helps when Steem is run in a virtual machine (works with VirtualBox, VMware). Another method is used to poll the mouse, and the cursor isn't bounded by the window frame (haven't found a way TBH). Some people actually prefer this way, even without a virtual machine.

Hide pointer on blit

Some video cards can make a mess when the PC mouse pointer is over the Steem window, with this option on that will never happen as Steem hides the mouse before drawing to its window. This can however cause the PC mouse to flicker while Steem is running, if that is a problem try turning it off and see if your video card is one of the messy ones.

Speed

There's also a mouse speed dial.

Tip Changing your PC mouse settings may help, the fact is that mice today are fast and not so precise. They are set up to rapidly go through thousands of pixels, the ST mouse was much slower and not accelerated.

Machine

ST Model

The STE was the better computer but most programs were coded with the older STF (and 520ST) in mind.

Notice that there was no model called Atari STF or Atari STE, they had a number like 520ST, 1040STFM, 520STE. There were also Mega models.

In Steem SSE, the Mega ST is like an STF with a blitter and a Ricoh real time clock. Its CPU clock is different (8010613Hz instead of 8021247Hz). You choose the memory. If you want to emulate the keyboard with Cherry black switches, you need to purchase a mechanical keyboard!

If you use a US TOS, the CPU clock will also be different on the STF (8010600Hz) and the STE (8053976Hz): those are the NTSC variants.

The Mega STE uses the same CPU clock as the STE, but it can programmatically switch to 16MHz too. It also has the Ricoh RTC. It can use 1.44MB disks. By convenience, other models can too (hack). The turbo button (hack) commands the Mega STE's CPU cache and speed.

The STFM is almost like the STF, it just has some more bus errors on STE-only device addresses. Not all STF and STFM behaved that way, but some exemplars certainly did and we emulate those.

RAM Memory

Here you can select how much memory will be available in the virtual ST.

The most compatible choice is generally 1MB, some apps need more, some apps crash with 4MB.

12MB includes 8MB of ALT-RAM as on the MonSTer board. A driver and an appropriate TOS (e.g. 2.06) are necessary to use that RAM.

Please note that 14MB wasn't supported on the ST range of computers without special hacks, so not everything will work either with that option selected.

dbi

This is the cycle at which the interrupt lines are scanned by the CPU.

Clock (MHz)

You can set the clock of both the CPU. It is a hack that can boost the speed of the emulated ST, to improve those painfully slow games that were released. The ST's CPU ran at 8 megahertz but here you can whack it up to get things going faster. A couple of points: setting the CPU speed above 8MHz will cause the emulator to run **slower**, because it has more processing to do for each screen displayed. It will also cause many ST programs to crash/not work properly, because they rely on the processor running at the correct speed. If you want really high speed (512mhz and above), you may prefer the 64bit builds of Steem SSE because it has larger cycle counters that make it safer.

Tip You don't need to change the setting if you want to run the Mega STE at 16MHz as on real hardware. On real hardware, the CPU clock is internally commanded. There's no *turbo* switch on the original machine, not even a GEM option (design fault in my opinion).

Blitter start

On some machines, the cycle on which the blitter starts is 2.

Clocks (Hz)

In Steem SSE you can fine tune the clock of the CPU and the MFP timer clock (crystal or "xtal"). The default values correspond to typical machines. Apparently there was quite some variation, especially for the MFP xtal (was there a typo in the STE schematics?) So contrary to the CPU MHz setting, those sliders are not just hacks, they allow precise emulation of real models.

Monitor

The monitor type can be *Colour* (for ST low and medium resolutions) or *Monochrome* (for ST high resolution). It was an important choice on the ST, because only the appropriate monitor could display such or such resolution. Like the ST, Steem runs in one or the other mode.

Also available are extended monitors; these are larger screens than a standard ST could manage but are achieved by some tinkering with the ST operating system. The only programs that have any chance of working with extended monitors are GEM applications (windows/menu bars), all other programs will crash or make a

mess of the screen (there is nothing that can be done about that). Most GEM apps will fail too, but some will work and allow you to fit more on the screen than anyone could ever need! NOTE: Extended monitors only work with TOS versions above 1.02, also if you use a 4-plane resolution (e.g. 800x600x4) the ST can get very confused if it tries to change to medium resolution, so try to avoid it. You have to do a cold reset of the ST before any changes to this option take effect.

Instant machine change

Normally, changes to the ST model, memory size or ROM file apply only after a cold reset. With this option, you can make the change instant. It's a dangerous option, as it will often crash the ST or even the PC.

Cartridge

Steem can emulate ROM cartridges using cartridge images, here you can choose to insert one. For details how to make a cartridge image see the *cart image howto* file that comes with Steem.

tip Few cartridges were actually available. Generally the ST operated without a cartridge.

Switch off makes the cartridge transparent (or crashes Steem SSE if I forgot a bug). It's the same as if you removed the cartridge.

Freeze emulates the freeze button of the *Multiface* and *Ultimate Ripper* cartridges. For the latter, you need to select the appropriate Port connexion.

If you insert the (fake) MV16 cartridge image, Steem SSE will play the samples in the original B.A.T game.

Steem SSE also emulates playback of the Replay 16 cartridge (not recording).

Active cartridges

Steem SSE is ready to use active cartridge files, that is files that contain routines in binary code.

If a cartridge file starts with CUB3 as first four bytes (at $0), then Steem SSE will read the two 4 byte values at $20 (32bit builds) or $40 (64bit builds) as an offset to the binary of a C function, and as the lentgh in bytes of this function.

In the far future Steem SSE for Linux will read at offsets $60 and $80 instead of $20 and $40 (not implemented yet).

That way, one cartridge can be used by all 32 or 64 builds.

The function is declared as:

```
/*  pointer to function
*   ad=address bus value, 23 significant bits A1-A23
*   rpin=table of booleans representing pins of the chip rpin[0]=bit 0
*   for Cubase 2, rpin[8] to rpin[15] represent D8-D15
*   for Cubase 3, rpin[8] represents D8
*   return value=data bus value
*/
extern "C" WORD (*CartridgeCheck)(DWORD ad,BYTE *rpin);
```

The routine will be called at each cartridge access (ROM3 signal).

Notice that "CUB2" cartridge files are not supported for the moment because a check is necessary at each Upper Data Strobe, which is some load.

Because Cubase 2 and 3 for Atari are not freeware, those cartridge files are not released anyway, but if you're smart you can make your own!

Hardware overscan

Steem SSE can emulate two hardware overscan circuits, either *Lacescan* or *Autoswitch*. You need appropriate ST drivers and should use TOS 1.04. For technical reasons those devices weren't available on the STE.

Wakeup

This is a very technical point, related to variable latency in some electronic chips. Normally you'll use these options only if the display of some demo isn't correct without (see the Hints file).

You can adjust the wakeup settings for the GLU(E) and for the Shifter.

The GLU wakeup option allows you to choose one of the 4 wake states of the STF. On the STE, the value is 1 and can't be changed.

The Shifter wakeup option changes the precise cycle when the pixel is matched to the palette. If it isn't what the program expects, you can see some artefacts on *Spectrum 512* pictures (blame the ST and the programmers for that, not the emulator!) The default (0) should be most compatible.

Random on boot randomises GLU and Shifter wakeups so that you get the delightful surprise as on real hardware.

TOS (ROM)

TOS officially stands for *The Operating System*.

Including the BIOS, the GEM graphic environment and other layers, the full OS, except on very early models (also emulated), was printed on ROM chips. You need an image of one of those roms to use Steem.

Generally, TOS 1.02 is the most compatible for STF and TOS 1.62 for STE mode.

The Mega ST can have TOS 1.02, 1.04, or 2.06, the Mega STE TOS 2.05 or 2.06.

Some rare programs prefer another version (check Hints file).

1985: TOS 1.00 (ST/STM/STF) - original ROM version

1986: TOS 1.02 (STM/STF/STFM/Mega ST) - blitter support

1989: TOS 1.04 (STF/STFM/Mega ST) - coloured Atari logo, better disk I/O, many bugfixes, faster.

1990: TOS 1.06 (STE) - STE hardware support

1991: TOS 1.62 (STE) - bugfixed version of above

1990: TOS 2.05 (Mega STE) - new desktop, customizable icons, HD disk support

1991: TOS 2.06 (Mega STE) - bugfixed version of above, support for future hardware. It could also work on a modified STF (emulated in Steem).

TOS 3.0X TT (not supported in Steem)

TOS 4.0X Falcon (not supported in Steem)

US TOS versions

Using a United States TOS causes games to be run in the NTSC screen mode (60Hz). This is okay emulation-wise but a lot of ST games and demos do not work in this screen mode, and often just crash or act oddly rather than telling you why. We recommend that American users of Steem use UK TOSs for games and demos, the only time a US version is required is for word processing, due to tiny differences between the US and UK keyboards.

Tip UK and US keyboard definitions are available.

TOS 1.00

The first version of TOS isn't as bugged as is often said. TOS 1.00 is much decried for its hard drive management, but it's not really better on TOS 1.02 (Mega ST), and not that bad actually, the Mega ST was a capable computer. It is relatively slow. TOS 1.04 and above feature better handling of hard drives.

TOS 1.06 and 1.62 UK

Watch out, somne copies of these TOSs are corrupt.

Language

Some localised version of the TOS may be necessary for some programs.

If you have some trouble with a program, you may give it a try. For instance, digits need shifting on a French keyboard but no an English one.

EmuTOS

EmuTOS is a GEMDOS compatible operating system for Atari ST series computers. It is made from Digital Research's GPLed original sources and is a free and open source replacement for common TOS images for Atari ST emulators. It hasn't exactly the same functions and compatibility as regular TOS, but most software will run on it. It is regularly improved.

http://sourceforge.net/projects/emutos/

The last stable version of EmuTOS is shipped with Steem SSE.

On Screen Display (OSD)

The Steem OSD is there to look pretty and give information, it can be configured to suit your needs.

Logo

You can configure how long the 'Steem' logo will be displayed after you start emulation.

Speed bar

The blue bar in the bottom-left corner indicates the current speed of the emulator compared to the refresh rate of the monitor on a real ST - if the bar is full it is drawing at the ST sync rate.

State icons

These icons tell you whether the emulator is running, fast-forwarding, sound recording or just stopped.

CPU speed indicator

This icon appears next to the speed bar when you have ST CPU Speed set to more than 8 megahertz.

Time

If you ever wonder how much time you waste on Steem, this is the control for you. Mercifully it resets at each cold reset.

Debug info

Some information will be displayed on the top left corner of the window or on the status bar:

EXC	CPU exception number
OVR	keyboard ACIA overrun
PRT	printing
CRC	disk checksum error

FPS (Frames per Second)

With this frame rate counter, Steem can reveal the hard reality about some of those games.

Palette changes are ignored. It's not guaranteed to work with everything due to the technique used, based on video registers VBASE and VCOUNT, and a simple checksum of the video memory.

Disk access light
This is a small light that appears in the top right of the screen, it emulates the light that was on the right hand side of a real ST, except that the led is green for reading, red for writing, yellow for hard disk.

Drive track info
Alternatively, this will display current drive, side, track and sector, like those devices some people had attached to their ST. Track is the current drive cylinder, sector is the WD1772 SR register.

Scrolling messages
After a while, they're not so useful anymore and can be distracting, so you can turn this off.

Bad jokes
For those scrollers that never were useful.

Disable on screen display
This will disable all elements of the on-screen display.

No OSD on stop
Stop is then delayed by one frame. This is useful if you want to take screenshots or want the colour count.

DirectDraw: could be necessary for correct screenshots

I/O Ports

This page configures the emulated ST's link to outside hardware. There are sections for parallel and serial ports , as well as a section for various adapters. The MIDI port is handled on the MIDI page. To set up a port select one of the options from the box next to *Connect To*. When you do the section will fill up with options related to the connection you chose.

MIDI Device
When you select this you must then select the PC MIDI device you want the ST's messages to be sent to and the PC MIDI device you want to be able to send data to the ST.

Parallel Port (LPT)
This allows you to connect an ST port to one of your PC's parallel ports. The only option is to choose which one. Warning: This may not work on all versions of Windows. See the FAQ for information on printing from Steem. Note: that port is generally not featured anymore.

COM Port

By selecting this option, you can connect an ST port to one of your PC's COM ports. Again, just select the required port. Warning: This may not work on all versions of Windows.

File

Using this option, you can send all output from an ST port to a file. Click on the *Change File* button to open the file selector where you can select the output file (you can create a new one by typing in a new name and clicking *Open*). The file will record the raw data (see below, SMM804).

The *Reset Current File* button will delete everything that is currently in the file, it will also close (not delete) currently open RTF or PBM files and reset the page counters for those.

You can't use a file for input.

Loopback (Output->Input)

Selecting this will cause anything that is sent to the port to be then received by the port, what use that is I really don't know!

TCP/IP

With this, you should be able to connect with someone else on the internet through Steem SSE. One instance of Steem SSE can play server (write *server* as address), the other can play client (type the address or the IP of the server). Reselect the TCP/IP option to start the connection if you change the address or the port. The server should be started first. It will probably trigger your firewall. The server can have multiple clients, if you think it's a bad idea or if your machine is client, just leave the option at 1. To terminate the connection, select the *None* option.

Various

Some (simple) dongles are emulated. You can select one in the Ports page. For joystick port dongles, option C1 ('Low-level emulation' on the Keyboard/Mouse option page) must be checked.

This section also covers some other adapters:

-Pro Sound Designer: soundcard used by *Wings of Death* and *Lethal Xcess*.

-Switches for Ultimate Ripper Cartridge and Multiface ST.

This is also the place where you can engage options for a dot matrix printer (SMM804) or a laser printer (SLM804).

When printing is detected, Steem automatically opens an RTF file in the selected print folder. This file may have formatted output (bold, italics, etc.)

When graphics printing is detected (such as printing the screen using alt-help), or the Laser printer is used, Steem automatically opens a PBM file per page in the same folder. Sorry, no PDF!

MIDI

Here are some options that affect the PC side of Steem's MIDI emulation.

Remember that the ST side is also affected by ACIA emulation.

Use DirectMusic

The DirectMusic drivers allow greater precision and potentially eliminate MIDI jitter, provided your audio interface has the correct drivers. This uses the DirectMidi library by Carlos Jiménez de Parga.

Use timer

Try to time MIDI out using a Windows timer. This is not very precise and could be worse accroding to your system.

Use Sleep

The "Sleep" Windows function is a way to wait for an event. If it isn't checked, Steem SSE will use SwitchToThread instead. Normally SwitchToThread is better and Sleep consumes more CPU power.

Volume

Not that useful but here nevertheless. It won't work for all devices.

Device choice

Like the *I/O Ports* page.

Sysex (system exclusive)

Raw MIDI

Don't interpret data as MIDI messages (no SYSEX at all).

With this and the TCP/IP port, you could be able to play MIDI Maze with other idiots on the Internet. That is, if that works, which would be surprising.

Allow running status for output/input

Normal MIDI messages are made up of one status byte that describes the message and then one or two data bytes. If a program wants to send the same message again it can leave off the status byte and just send the

data. By default, Steem doesn't allow this and adds the status byte itself if it has been omitted, but this could cause problems to programs that require a very high transfer rate.

These options mean that running status will be passed on to Windows which has to decide what to do with it, so if you are having problems with MIDI tempo you can try this.

Sysex Buffers

The number of system exclusive buffers affects MIDI when it is sending/receiving many small system exclusive messages quickly. If the number of buffers is too small, it is possible for there to be so little time between messages that Steem is still busy with all the buffers when a new sysex message arrives (and therefore part of it gets lost). If you find some MIDI program having input or output problems that sound like this it's worth a try increasing the number of buffers, something like 6 should do the job for almost anything. The size of the buffers allows you to choose the maximum length of message Steem can send or receive. This is important if you are transferring large memory banks from the device to the ST and vice-versa.

Input speed

This option allows you to slow down the speed Steem feeds any received data from the MIDI device to the ST. Due to restrictions of Windows Steem sends all data as fast as possible to the ST regardless of the speed it was actually sent by the device, but some ST programs can't handle that. If you have problems with receiving bank dumps it might be an idea if you slow this down a bit. Also related to this is the shortcut action *Pause until next SysEx*, this can get around time-outs on ST programs waiting for a bank dump to start. See the shortcuts section for details of how to assign this action to a key/combination of keys.

Record Input

This is a handy feature that will allow you to record sequences of input in to the ST and play them back later as macros. At the top there is a display of your macro folder: each macro is stored in a file with the extension stmac, and they are displayed here (if you have any). To record a macro, first click on *New Macro*, name it something snazzy and then click on the big red record button.

Now go to the main Steem window, start emulation and do some sort of input (move the mouse, press a key, move a joystick). When you have recorded what you want to go back to the macro display and click on the record button again to turn it off. Don't worry about doing this too quickly, Steem doesn't start recording until some input is received and it will cut off any time on the end that doesn't have any input. Now you can replay the actions you recorded by pressing the green play button (next to the record button).

You can have as many macro files as space on your hard drive will allow.

This might mean they get a bit difficult to find on the list, so you can organise them into folders using the macro folder display. Just right click to create a new folder and drag appropriate macros in there. You can also right click on the bold *Macros* folder and choose to open it in Windows explorer if you prefer to use that for your organising.

There are 2 options that affect how macros work:

Maximum mouse speed

This affects recording and playing macros. Some ST programs won't accept really fast mouse movements as ST mice couldn't generally move very far in a short time. This option lets you allow for this problem by configuring the maximum speed for each macro individually.

Playback event delay

For most macros you will want your actions to be played back as quickly as possible (for instance in the case of clicking on a shoot button in a game).

This option allows you to configure how long Steem will wait between input changes, as some programs won't accept lots of very quick inputs. If speed isn't an issue then you can always set this to *As Recorded*, then playing the macro will replay the recorded events at the speed you recorded them at.

Configurations

This page allows you to store many different Steem configurations and use them when you like. To make a new file containing the current Steem configuration click on *New*. You can then restore it at any time by selecting it and clicking on *Load*. Be careful though as your configuration before you load another one will be lost forever.

Below is a list of sections that affect loading configurations, you can enable or disable each section. This allows you to only load specific options in Steem and keep other options that you don't want changed. As in macros (see below) you can organise all your configurations using the *Configurations* folder display.

For example, the 520 ST configuration would have 512KB memory and TOS 1.0, maybe a single-sided drive, the 1040 STFM would have TOS 1.02.

You can set up some configurations for STF, STE, Mega ST, Mega STE colour or monochrome, with GEMDOS or ACSI hard drive emulation, etc., and then select one in one click rather than manually changing all the settings. The config must have the exact same name as the model. STF will work, 520 STF will not.

If you want to change startup options, such as starting in full screen, you should first select the config then change the option.

Please be advised that the whole configuration is saved, the checkboxes are for loading only some parts of it.

When you save a configuration, the paths to the ROM folder and to the home folder aren't saved. Paths to the current TOS and disk image files in floppy drives are made relative to those folders. That way, config files are more portable.

Icons

You can customise Steem's look by altering its icons. On this page you will see a box containing all the icons, just left click on one to be able to select an ico file to replace it with. Right clicking on an icon restores it to its

default, click on the *All Icons to Default* button to restore them all. You can also download icon schemes to change all or just some of Steem's icons with a single click on the *Load Icon Scheme* button, go to:

http://steem.atari.st/icons.htm (link still working)

or

https://sourceforge.net/projects/steemsse/files/Other/

to see what is available.

You can create your own full set based on the stico file.

This option page is not available when the big GUI is active.

File Associations

Windows programs love to associate and de-associate themselves with file types. This dialog allows you to choose which files you want to associate with Steem. Just click the *Associate* button next to the desired files, it changes to *associated*.

Click again to de-associate.

Note: if you used an old version of Steem and associated files, you must re-associate with the current version (or rename the program 'Steem'). If you used a version before v.3.5.3, you should manually remove the associations in the registry, HKEY_CLASSES_ROOT part. This is because in newer versions, associations are recorded for each user, in older versions, they were for the computer, and you needed administrative rights (as was natural on older Windows versions).

Emu parameters 1

Software overscan

There is a trade-off between emulation accuracy and performance.

None: Normal video output, most video tricks (abusing the GLUE, the MMU, the Shifter (1), generally called "Shifter tricks", are not emulated as there is no overscan check during emulation. This is actually enough for most programs, including most games, as overscan wasn't part of the ST specification, but many demos won't display correctly. Who watches demos anyway, right? However, many unofficial game compilations use overscan to display scrollers. Spectrum-512 emulation (colour palette changes on the fly) will still work without overscan.

(1): GLU(E): General Logic Unit; MMU: Memory Management Unit; Shifter: combines and displays video pixels

High-level (C2): Efficient emulation using knowledge of often used "Shifter tricks". It should run almost all games and demos. CPU use varies with what is being emulated but is generally low.

Low-level (C3): Directly emulates the video chips, the option is available only if the *STVL* plugin was detected. It hasn't been released yet, so it's normal if the option is greyed out. CPU use is constant (per resolution) but high.

Tip Please don't turn this off then complain that programs don't work

Some emulation parameters that normally should be left alone.

Tracking VC

Some demos change the video RAM right after it's been displayed.

Unstable Shifter

This alters some aspects of high-level (C2) or low-level (C3) software overscan emulation.

C2: Option needed to emulate the *beeshift* effect (some technical demos by LJBK) and some other cases (*Overdrive/Dragon* for instance).

C3: Likelier to get Shifter bands and other funny glitches if you change the wakeup.

Rendering delay

Those parameters adjust the timing of video rendering in different situations for better compatibility with option C2 checked. They are expressed in 8MHz cycles.

CPU	The MC68000 is writing in the video memory, close to the video counter
RES	Resolution change
VC	Video counter change (STE)
PAL	Palette change
END	End of scanline
LW	Linewidth (STE)
SCR	Horizontal Scroll (STE)

MC68901

All the MFP parameters show that for that chip, Steem SSE still doesn't have a 100% accurate model.

Emu parameters 2

Run speed (hack)

Sometimes you will find a game/program that isn't significantly affected by changing CPU speed, it is still too slow. In that case you can use this option to change the speed Steem runs and make sure a program speeds up (or even slows down if you want). The disadvantage is that sound doesn't work above 105% or below 80%.

Slow motion speed

This option determines how fast Steem will run when you have slow motion turned on (right-click on the play button).

Maximum fast forward speed

Using this option, you can limit Steem's speed, fastest isn't always best (you can find you have zipped through the bit you wanted to see).

Disk

Some emulation parameters that normally should be left alone.

Hacks

This option covers three kinds of things:

- Convenience hacks, like accelerating STE booting (hard drive detection is slow)
- Options that go beyond the plain vanilla ST, like faster drive, faster CPU, bigger screen, etc. Without *Hacks*, those options are not available
- Emulation hacks, at least Steem SSE is straightforward about it

Uncheck for an authentic ST experience.

Disk manager

This window controls all the disks on the ST, and the two boxes up the top hold the disks currently in the drives. This window is resizable.

Disk Images (ST, STX, MSA, DIM, STT, IPF, HFE, SCP, STW, CTR)

Disk images are floppy disks turned into files, this is how most programs are accessed on ST emulators. For details of how to create disk images from ST floppy disks have a look at the *disk image howto.txt* that comes with Steem. You will find most images on the internet however.

The large box at the bottom of the disk manager window shows all the disk images in the current directory. The disk manager is like a little Explorer window, you can go into a folder by double clicking on it or by selecting it and pressing return. All disk image files, zip/rar/7z/bz2/gz/tar/arj archive files and any Windows shortcuts to those types of files are shown.

Steem SSE needs some DLL files to handle archive files.

-unzipd32.dll for zip files (32bit)

-unrar.dll or unrar64.dll for rar files

-ArchiveAccess.dll for 7z/bz2/gz/tar/arj files

ArchiveAccess.dll manages zip and rar files too if the other plugins are not available to Steem SSE. ArchiveAccess can't handle newer rar archives (v5).

At first Steem doesn't know if some archive contains a disk image or not, so all archives are displayed in the Disk manager with a ZIP icon regardless. There is an option to make Steem SSE check inside archives and only display disk images, but it is slow. If you hover over a disk file icon, the hint will display what kind of image the archive contains.

To insert a disk image into a drive, drag it to the box to the right of the big drive icon (or press A or B). To remove it from the drive just drag it back out to the directory view (or press Delete).

If you right click in the directory view a menu will pop up allowing you to create a new standard size disk image (ST), a custom size disk image (ST, MSA or DIM), a STW disk image or an HFE (Rev.1.1) disk image. High density versions are available for ST and STW.

Custom disk images can be up to 2 megabytes in size, although larger ones can be emulated correctly, most TOS versions didn't support them, you may find they will only work with TOS 2.05+ or maybe not at all. Only some real Mega STE could handle high density disks. In Steem SSE, other models can only handle such images when Hacks are enabled.

ST disk images contain raw data bytes without any meta information (so that emulators must guess the disk

configuration, if it isn't correctly written on the disk image). MSA and DIM images also contain extra information. MSA images can be created on an ST using Magic Shadow Archiver. Steem can read compressed images, but writes back uncompressed so you should have backups, beware! DIM images can be created on an ST using *FastCopy Pro*. Steem needs all sectors.

STW and HFE disk images include clock bits as well as data, encoded into MFM words. Those STW and HFE images are more than twice larger than ST images but allow emulating the WD1772 disk controller at a lower level, and so handling the Format command. Because they're formatted inside the emulated ST, there's no sides/tracks/sectors options. You get a raw disk full of random data, lucky you! It's up to you to format the disk exactly as you would on a ST. STW is the simplest such format. If you open a disk image in a hex editor, you will find sync marks coded as $4489. HFE is optimised for the HxC floppy emulator hardware. It's quite handy if you own such device.

STW images may even be bigger if they are version 2, like images converted from STX or SCP, because they contain fuzzy data and timing information. It is the ultimate ST disk image format.

Tip If you plan to write on STW or HFE disk images, turbo drive is less reliable

Beside those ST, MSA, DIM, STW and HFE disk images Steem can create itself, it is also able to handle:

-STT images (read-only)

-STX images (read-only if using pasti.dll, read-write otherwise)

-IPF and CTR images created with Kryoflux hardware (read-only, need CAPSImg.dll)

-SCP images created with Supercard Pro hardware (read-only)

A right click option allows you to convert DIM, MSA and ST images to STW. Make sure both drives are empty when you do that to avoid any interference. There's also a MFM emulation option that does that on the fly, but the extra steps take some time.

About IPF and CTR images, as they are handled by a third-party plugin, whether they work or not is also up to this plugin. Make sure you have the last version.

Steem SSE directly handles SCP images, there's no plugin, so any bug should be fixed in Steem SSE itself. SCP images can also be converted to STW (v2) images.

STT is a Steem-specific format that is to be considered legacy. It is a sort of "stx light", it was abandoned when the STX format appeared, but Steem and some other programs still support it.

STX images are handled by the pasti plugin if available, directly by Steem SSE otherwise.

If you right click in the box to the right of the big drive icon, a context menu allows you to toggle the read-only or write-protect property (important for some games), swap disks A and B, get the previous or next disk (Disk Swapper), remove the disk, or go to where the disk is stored. Notice that it is possible to eject a writable disk without saving the changes, this is handy in case you made some mistake.

Write-protect is internal: it will not change the file property, when it's checked it's like the sliding tab on the floppy was up. Read-only will change the file property. When you insert a disk, the option will be set if it's Read Only or an archive.

Right clicking on a disk image displays a menu with a myriad of options to allow you to organise and use your disk images. *Get CRC32 and Contents* is useful if the disk image is a compilation disk, selecting it will display a list of what the disk contains. For it to work, *disk image list.txt* must be in the doc folder, and it must contain the entry. This list uses CRC32 checksums to identify files. It was based on a previous TOSEC version and misses some disk images. Some have been added in the version shipped with Steem.

If you right click on an archive that contains multiple disk images, you can select the image thanks to the little arrows in the menu.

Tip Steem is good at finding images in multiple files archives, but it can't open an archive inside an archive

MSA Converter is a great program that allows you to edit the contents of a disk and convert it to a different format.

Tip It can convert a zip file containing ST files to a ST disk image in a single click

Properties allows you to change the way Steem uses the disk, it is very unlikely you will have to change the setting from the default.

You can also run config files (INI) from the disk manager.

If you double click in some empty space of the main window, you go to the parent folder of the current one.

If you don't want to use the Disk manager to select disk images, two shortcuts are available to use Windows' classic file selector instead, and you also may use the classic menu by pressing the Alt key.

Custom disk pictures (New in v4.2.1)

The disk manager will display custom disk pictures. The images must be located in the same place and have the same name as the disk image or archive, extension included, plus the picture extension (jpg, png, bmp).

For example:

Barbarian (Psygnosis) Disk A.rar

Barbarian (Psygnosis) Disk A.rar.jpg

Barbarian (Psygnosis) Disk B.rar

Barbarian (Psygnosis) Disk B.rar.jpg

If there's a picture file called exactly "diskimage.png" in the directory where the disk images are, this will be used for all those disk images (except for those with an individual custom disk picture). This can be useful for collections.

Drive icons

Left-clicking on the B: drive icon toggles drive B:.

Right-clicking on the drive icons opens a context menu with some options to:

- Stop the motor if it's running you crazy. It's a hack and will not work every time.
- Change your fine internal or SF314 double sided drive into a miserable internal or SF354 single-sided drive (if you were unlucky back then and feel really nostalgic, or you just need to check things). For example, *Star Ray* loads sampled music from side B if it can.
- Select the drive or the side of the disk as if you were using a *freeboot* switch (don't forget to reset that later!) On drive A:, it selects the second side but on drive B:, it swaps drives so that a game can boot on the external drive, which was very useful back then.
- Select drive sound samples. You pick the folder, the files must have those names:

  drive_startup.wav, drive_spin.wav, drive_click.wav, drive_seek.wav

  Tip Steem SSE for Linux recognises Wavosaur's loop markers inside a wav file

Disk Swappers

Steem SSE has two disk swappers. One is called smart because you don't need to create a list, Steem SSE will look for the next or previous disk in the current folder or archive. Accessible through shortcuts, the classic menu or the context menu.

If the disk image is found, Steem SSE will insert it and mention the name in the status bar (or on the OSD) for a while. If not, nothing happens. Searching may take a while especially going backwards because Steem SSE

must check inside archives.

You may assign two shortcuts for *Previous disk* and *Next disk*. Otherwise, right click on the inserted disk in the disk manager to reveal the items in the context menu.

(v4.2.1) The second disk swapper uses a m3u file containing a list of disks. To activate the swapper, insert the file as you would a disk image, in the disk manager. The swapper will deactivate automatically if you insert another disk.

An example of list is (file test.m3u):

```
# same directory

Arkanoid.MSA

MGT -STK.STW

# in another directory

../Demo/3615GEN4 ALF.ZIP

# same directory

Buggy Boy (1988)(Elite).zip
```

It's as simple as that. It works with absolute or relative paths.

The smart disk swapper is always active, except when a m3u file is in use.

GEMDOS

Another handy button on the disk manager is the GEMDOS hard drive icon. A right click opens the GEMDOS hard drive manager. You can have up to 10 virtual hard drives. To set one up select the folder on your PC and then select the letter that you want it to be on the ST. Those virtual drives are called GEMDOS drives because they're based on interception of GEMDOS commands.

Tip Like in good old TOS and DOS days, limit your file and folder names to 8 characters and max 3 character extensions: OKAY.PRG; NOT_OKAY_NONO.FOLDER

You need to know that this type of emulation is very convenient for you, because you just use your PC hard drive, but it is based on hacks, and in some cases, it could not work as on a real Atari, or something could work in Steem but not on a real Atari.

Hard drives aren't as reliable as disk images for games, most things will work but not everything. If a program isn't working properly copy it to a disk image (using the ST desktop) to see if that helps. Another option in this box is to choose which hard drive to boot from if floppy drive A is empty. You can force Steem to boot from a hard drive even when drive A contains a disk by holding down the CONTROL key when you run after performing a cold reset of the ST. ALT may force loading the floppy disk (on some TOS).

Left-clicking on the icon will toggle GEMDOS hard drive emulation on/off.

Notice: this emulation based on interception will only work with TOS and compatible OS like EmuTOS, not MagiC for instance.

ACSI Hard Drives

There's also a hard drive icon for ACSI hard drives. Left-clicking on this icon will toggle ACSI hard drives on/off, right-clicking opens a box with setup details. ACSI (Atari Computer Systems Interface) was the cheap SCSI-like interface between the ST and a hard disk or a laser printer. ACSI emulation uses img files that contain all sectors of an Atari hard drive (data only, like an ST floppy disk image). Note that unlike GEMDOS hard drives, each ACSI device may be split into several partitions (C:, D:...). The driver you use must be able to handle each of those drives (Atari's AHDI is).

Two ready examples are provided in the download, images of the familiar SH204 and MEGAFILE 60 hard drives, empty save fore the drivers.

It is also possible to enable GEMDOS virtual hard disks at the same time as ACSI virtual hard disks, you need to adapt letters, of course. ACSI drives will take the first letters. It's handy to copy files.

It is possible to create hdd images using Steem SSE's interface. Enter the number of megabytes you wish (1MB=1,000,000 bytes). The file is filled with zeroes. It's up to you to make it a working hdd image. ICD Pro and other ST software may help. The feature isn't much but without it you need special utilities just to create the file.

Some programs will work with a GEMDOS hard drive but not with an ACSI one, sometimes depending on the driver. In turn, ACSI emulation being lower level, it should be compatible with any alternative OS, such as Magic.

Pasti also provides ACSI emulation, for it to work it must be enabled in the disk manager and Steem's ACSI emulation must be disabled. The hard drive image is selected in pasti itself.

Tip Make regular backups of your image files, you never know if Steem SSE will damage them. Bugs are always possible.

The Atari SLM804 Laser printer also uses the ACSI connection. You don't need to enable ACSI hard drives to use the printer, the option is on the *I/O Ports* page. The printer controller is hardcoded as ACSI device #7.

Home Folder

The disk manager allows you to select a home folder. This is where all your disk images or shortcuts to disks

should be stored. You can go to your home folder at any time by clicking the third button from the left in the disk manager window. If you are away from your home folder you can quickly move/copy/create shortcuts to disks in your home folder by right mouse button dragging them to the home button. The button to the right of the home folder button sets the current folder as your home folder. Right clicking on the home button and the set-home button brings up menus allowing you to go to/set 10 quick-folder links.

Options

To the right of the set-home button is the disk manager options button, it brings up a menu with the following choices:

Turbo Drive (hack)

Floppy disk drives are slow, horribly tediously slow, so optionally Steem does away with the waits and emulates disk access at impossibly high speed.

Unfortunately, some ST software uses, ahem, colourful ways of accessing the STs floppy disk controller that only work if the disk access is horribly, tediously slow. Even some contemporary demos poorly program the controller.

Though the *Drive sound* and *Drive track info* options may make this ordeal more bearable, as you can see and hear what the drive is doing, you don't get the feeling the emulator is stuck. It's emulated the way it was.

Generally, we don't recommend leaving this option checked with unknown disk images, it is less compatible.

The option doesn't operate with CTR, IPF, and SCP images. Speed for images run by Pasti is set in *Pasti configuration*.

Count DMA transfer cycles

On real hardware, each time disk data is transferred to or from memory, the CPU must surrender the bus. It is correct emulation to do the same, but it can be trouble according to the floppy or hard disk emulation method (especially accelerated). This option allows you to disable the feature.

Automatic fast forward on disk access

Sometimes disk access can be really slow, so this option makes Steem always fast forward when the ST is doing it. There are some programs that continue while accessing the disk, if you are going to use one of them then you should turn this off.

This feature is compatible with accurate disk access times, with MFM, Caps (SPS) and Pasti emulation, which means you can get both accurate and fast emulation.

Read/Write Archives (Changes Lost on Eject)

To access disk images in archives Steem extracts them to a temporary file and uses that like a normal disk image file. Unfortunately re-archiving the temporary file can be difficult according to the archive type. Besides,

modifying an archive is questionable, maybe we just shouldn't.

Because of this by default all disk images in archives are treated as read-only, so they can't be changed.

By turning on this option you can make Steem allow writes to the temporary file, but any changes will get lost when the temporary file is deleted (the disk is ejected).

Some ST programs require their disk to be read/write, this option will allow you to run them without having to extract them.

Protect image files

When this option is checked, your image files won't be changed by Steem SSE.

Connect drive B:

By default, Steem emulates an ST with only one internal floppy drive. It can also emulate an external drive, this is usually very useful. Unfortunately, most STs only had one drive and some programs will get confused when there are two.

It is usually a good idea to reset after you change this option as the ST might still think there is a drive B when really there isn't.

Enable ghost disks for protected disks

Check this option to harmlessly save high scores or game progress with CTR/IPF/STX/SCP disk images, or other images that are read-only. STG files are automatically created when the program tries to save data on the disk, leaving your image file untouched. The image may be in an archive. STG files are simple and compact collections of sectors, only those that have been changed. The feature may interfere with some programs, in that case you only need to disable the option.

Run PRG and TOS files

This enables direct support of PRG and TOS files. To run standalone PRG or TOS files, you need to set up a Z: hard drive that points to a PRG folder, preloaded with a DESKTOP.INF file and an AUTO folder, as provided in the download. Steem SSE will copy the file in this folder (always under the same name, so don't worry about disk waste). This should work with files inside archives. The main limitation is that it only handles single files. If the PRG needs a data file, it won't work. If you want it to work, just use the regular GEMDOS hard drive feature. Also, some programs just aren't meant to work that way. In this case they should be run from a floppy or hard disk.

Automatically Insert Second Disk

Lots of games come on two disks, if this option is on when inserting a disk into drive A Steem will look for a second disk image with a similar name and insert it into drive B. Note that even if Steem manages to insert the second disk in drive B:, the game may still require a disk swap in drive A:. Many games had only one drive in mind unfortunately. The same for demos.

Disk Swapper Patterns

When checked, the smart disk swapper will look for a name where only one number or letter changes, for example [1], 2 of 4, or Disk B. It should work in many cases, and it is fast.

If unchecked, the swapper will just look for the next or previous disk image in the directory or the archive. If you so fancy, you can go through the full *Automation* set with just the disk swapper. It can be slower in large directories with lots of archives, especially going backwards, because Steem must look into all archives until a match is found.

When there's no previous or next file in an archive, Steem warns you first, then if you insist it looks for a file out of the archive, in the folder. The order of files is logical but can sometimes be unexpected.

For instance:

Chrono Quest (B).STX
Chrono Quest (C).STX
Chrono Quest (D).STX
Chrono Quest.STX

The first disk is last in the sorted directory. There are ways to name your files so that this problem doesn't occur. Chrono Quest (A).STX comes to mind. For the game, however, disks B to D are data-disks 1 to 3!

Eject Disks When Quit

This makes Steem eject all the disks from the drives when you quit.

Close Disk Manager After Insert, Reset and Run

If you don't like the disk manager open when you run a game, give this a go.

Pasti options

Pasti, by Jorge Cwik (ijor at http://www.atari-forum.com), is a way to use the STX floppy disk format in Steem. This format allows emulation of original (protected) floppy disks of yore. Try to use the last version of the plugin (check at atari-forum).

Pasti for all floppies and ACSI

Pasti can handle ST, MSA and ACSI disk images, if you want that check this option (for examplae to go around a Steem SSE bug).

If the option is checked, the SLM804 Laser printer isn't available.

If the option is unchecked, pasti will only handle STX images.

Configuration

It opens a dialog box managed by pasti.dll. It is recommended to set all speed options to *slow* for precise emulation, which is what you should expect of STX images, and to disable warnings.

MFM emulation

Auto-convert

That option allows Steem SSE to internally use an MFM format for ST, MSA and DIM disk images. Steem automatically converts those images to STW on insertion and back on ejection. MFM emulation is more precise. It can be slower and disk image conversion on the fly takes some time too.

If option *Use Pasti for all floppies and ACSI* is checked, this option has no effect on images handled by pasti.

Low-level

STW, HFE, and other images if *Auto-convert* is checked, will be handled bit by bit as if they were SCP images.

Double Click Action

This allows you to configure what happens when you double click the left mouse button on a disk image (or a shortcut to a disk image). By default, doing this will insert the disk into drive A, reset the ST and then run.

The same effect is assigned to the Enter key.

Hide

Hidden files

Files hidden in Windows explorer won't show in Steem.

Extension

File extensions such as zip or msa won't show.

Non-Disk Archives and Broken Shortcuts

This option hides shortcuts that point to a file that doesn't exist. If you show these shortcuts, then you can right click on them and select Fix Shortcut to allow Windows to search for the target file. You may want to check this if you have many broken shortcuts that are going to stay broken to speed up displaying of their folders.

The option also hides archives that don't contain any actual disk images. It can be really slow if there are many archives in the folder.

Icon Size and Icon Spacing

Save space with small icons, see more clearly with larger spaces.

Sizes go from 16x16 to 128x128 (which causes some bloat of the executable since v4.2.1).

Sort by

The disk manager can sort by date, newest first, it's useful when you point to your download folder.

Tools

Next to disk manager options is the disk image management tools button, it brings up a menu with the following options.

Search Disk Image Database

Steem uses the great disk image database created and maintained by Keili to be able to recognise disk images. Using this option you can search for a program to find out what disk images contain it. To download the disks have a look at Steem's comprehensive links in the General Info dialog.

Tip The database, as a text file, is now updated and versioned in the Steem SSE project itself

Open Current Folder in Explorer

Opens the folder the disk manager is looking at in Windows Explorer.

Folders Pane in Explorer

When checked causes the folders pane to appear when you open a folder in Explorer.

Find in Current Folder

Opens a Windows find dialog with the *Look in* directory set to the current folder, useful if you have a large number of disks in many folders and you want to find one quickly.

Run MSA Converter

This loads the MSA Converter program that can be used to edit disk images. It needs to be there, of course.

Command-line options

Here are some useful command line options for Steem. The easiest way to run Steem with them is to make a shortcut to Steem.exe, right click on it and select Properties. In the text box labelled Target add the command line options on the end (after the " if there is one).

This is also how a frontend will launch Steem.

You don't need to add - before the options.

NODD, GDI - don't use DirectDraw or Direct3D, only the slow Graphics Device Interface

NODS, NOSOUND - don't use DirectSound.

NODM - No DirectMusic

.st, .msa, .stt, .dim, .stx, .ctr, .ipf, .scp, .stw, .hfe, .zip, .rar, .7z, .stz file - load disk & go.

If two disks are passed, then the first will go in drive A and the second will be in drive B. Autoinsert second disk won't work.

.prg, .tos: same as disk but those are individual ST files.

.img, .rom: path to the TOS file

.sts file - load snapshot & go.

.stc file - insert specified cartridge.

NONEW - means that if one of the above files is passed to Steem and a Steem window is already open it will be opened in the current Steem and not in a new window.

OPENNEW - means the opposite of NONEW, the files will always be opened in a new Steem window.

NOLPT - this removes the option to connect an ST port to your PC's LPT ports (that option can cause a crash on some versions of Windows).

NOCOM - this removes the option to connect an ST port to your PC's COM ports.

INI=[filename] - load [filename] instead of steem.ini.

TRANS=[filename] - use [filename] as translation file.

SOF=[freq] - stands for Sound Output Frequency. Output will be forced to [freq]Hz and all other factors will be ignored (unless setting the sound card output to [freq]Hz fails, in that case Steem will ignore this option and use its usual method of determining output frequency).

WINDOW - force Steem to boot in windowed mode.

FULLSCREEN - force Steem to boot in fullscreen mode. There's also a Startup option for that.

DOUBLECHECKSHORTCUTS - this makes Steem's disk manager check shortcut files more thoroughly, some people have had problems when a shortcut's target file name is in a different case to the actual file. This option makes the disk manager slower to display folders with shortcuts in them.

SCLICK - this option makes Steem output a click when you run and stop instead of a bump.

NOPCJOYSTICKS - don't look for any PC joysticks, sometimes joystick drivers do some weird things!

OLDPORTIO - we changed how I/O to LPT and COM ports worked in v2.4, if you are using Win9x/ME then you can use this switch if it doesn't work anymore.

SCREENSHOT[=path] - this tells the currently open Steem to take a screenshot. If no Steem is open, then it won't do anything.

ALLOWREADOPEN - this command line option makes Steem open hard drive files as read only. TOS has a bug in it that allows programs that open files for read to write to them, most versions of Windows and all versions of Linux don't allow it. Steem gets around this by opening all files as read and write.

This works well except in circumstances when another Windows program is accessing the same file. If you want to do that then use this command line option to open the file as read-only.

NONOTIFYINIT - disable the "Steem is Initialising" box.

STFMBORDER - if it is not overruled by the saved options or memory snapshot, this makes Steem start as an STF.

NOTRACE - Steem SSE won't create a Trace.txt file with debugging info (there is also a more convenient option in the startup page).

INI file options

Those options are inside the steem.ini file to avoid bloating the emulator interface (GUI) itself.

DefaultSnapshot

You can define a shortcut, called "Load Default Memory Snapshot", that will load the memory snapshot you specify in the [Main] part of steem.ini, for example:

DefaultSnapshot=[path to Steem]\memory snapshots\mysupergame.sts

ScreenShotUseFullName

If you change the line ScreenShotUseFullName in the [Display] part of steem.ini to:

ScreenShotUseFullName=1

Steem will use the full disk name as the screenshot name, rather than just the first word.

This is now the default behaviour, so if you want only the first word write:

ScreenShotUseFullName=0

ScreenShotAlwaysAddNum

If you change the line ScreenShotAlwaysAddNum in the [Display] part of steem.ini to:

ScreenShotAlwaysAddNum=1

Steem will add a number to the screenshot file name even if there is no such file already.

WindowTitle

You can change the name that is displayed as window title and in the task bar by adding a line in the [Main] part of steem.ini, for example:

WindowTitle=Microsoft Excel

It is useful if you mean to run several specialised instances, each in its own folder.

Limitation: mustn't be longer than 100 characters.

MonochromeDisableBorder

If you add this line in the [Main] part of steem.ini:

MonochromeDisableBorder=1

Borders will be off in monochrome mode, except if you switch them on after the cold reset. You can also use different configurations instead.

NoNotify

If you add this line in the [Main] part of steem.ini:

NoNotify=1

It does the same as argument NONOTIFYINIT.

NoIntro

If you add this line in the [Main] part of steem.ini:

NoIntro=1

Steem won't run its intro proposing to add a shortcut. Useful for frontends, maybe.

TrueFullScreenGui

By default, on the D3D builds, only the Windowed Borderless fullscreen mode can show the GUI while in fullscreen.

If you add this line in the [Display] part of steem.ini:

TrueFullScreenGui=1

Steem will enable the GUI option for true fullscreen too, just in case it works on older systems (XP?).

Tip For example it doesn't work on my Windows 10 system!

PreciseModel

This is for config files, not steem.ini.

It is possible to have the exact model you want displayed in the status bar, you just need to add a line in the [Machine] part of your configuration file.

For example, in the 520 ST.ini config, you have

PreciseModel=520ST

You can create a 520ST with only a single-sided floppy drive running TOS 1.0, a 520STF with a double-sided

drive...

Notice that drive A: is always included, it cannot be disconnected.

MouseAd

This is for config files, not steem.ini.

In the [Patches] section of the configuration file you can set the ST address where Steem will read mouse coordinates which it can display on the status bar.

The address must be in integer format and point to 2 words for x,y.

For example in Barbarian, the address is $62C:

[Patches]

MouseAd=1580

Tip The *Patches* section of the configuration must be checked

Tip The setting will be remembered through reset, until another ini is loaded. It is also saved with snapshots.

Here's a few if you want to create ini files.

Title	TOS	Address (hex)	MouseAd
Barbarian	102	$62C	1580
Captain Blood	102	$17926	96550
Defender of The Crown	102	$2886C	165996
Dungeon Master	102	$2F024	192548
	162	$2CEBA	183994
The Final Conflict	100	$33226	209446
	162	$339BE	211390
Flight Simulator II	102	$055CE8	351464
The Guild of Thieves	162	$FC46	64582
King's Quest 1 + 3	102	$021788	137096
	162	$01F83E	129086
Maupiti Island	102	$04E9FE	322046
The Pawn	102	$13418	78872
Phantasie 3	102	$3C43E	246846

RunOnStartup

This option is also in the GUI, and it also works when you double click on a configuration file in the Options dialog.

It should be placed in the [Options] part of your configuration file.

RunOnStartup=1

Spurious

A spurious interrupt occurs when the MC68000 microprocessor is interrupted but the interrupt request (IRQ) clears during the *interrupt acknowledge* (IACK) cycle. On the ST, this may happen with the MFP. Emulation of this is tricky, you can deactivate it by changing this line in the [Options] part of your configuration file:

Spurious=0

Tip In Windows builds, this can be changed in the settings/emu parameters

UsersPath and TempPath

At startup, Steem tries to determine where the main settings path and the temp path are. You can customize those in the ini file.

UsersPath=[path]

TempPath=[path]

This must be the full path. UsersPath will be used only if the ini file is part of the command-line.

The ini file may also contain the path of folders such as memory snapshots, shortcuts, etc.

DirectMusic

If you add this line in the [Main] part of steem.ini:

DirectMusic=0

Steem SSE won't try to initialize DirectMusic.

Debugging info

If you check the TRACE.txt startup option, Steem SSE saves a log with that name containing some useful information in its running directory each time it is stopped.

It is possible to read the trace file from inside Steem SSE, in the info dialog. Notice that Windows will need

time to update the dialog if the file is large (but it eventually does).

Sometimes, a fatal error happens (system exception). Windows builds of Steem SSE tries to catch them instead of just crashing. To help reporting, an alert box describes the error. The message is also copied into the Trace file. If the bug happens in Steem SSE itself, it's sometimes possible to track it down using this message.

If you post a bug report, please read the rules before (on the website forum). Ask yourself if you're giving all useful information to get the bug fixed or if you're just venting frustration like a lamer.

Don't forget to copy or join the TRACE.txt (report on the last run) and options.txt (complete dump of your options when leaving Steem) files if appropriate.

Bug reports:

https://sourceforge.net/p/steemsse/forum/bugs/