

([//github.com/mamedev/mame](https://github.com/mamedev/mame))



Welcome to The Official Site of the MAME Development Team

What is MAME?

MAME is a multi-purpose emulation framework.

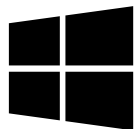
MAME's purpose is to preserve decades of software history. As electronic technology continues to rush forward, MAME prevents this important "vintage" software from being lost and forgotten. This is achieved by documenting the hardware and how it functions. The source code to MAME serves as this documentation. The fact that the software is usable serves primarily to validate the accuracy of the documentation (how else can you prove that you have recreated the hardware faithfully?). Over time, MAME (originally stood for Multiple Arcade Machine Emulator) absorbed the sister-project MESS (Multi Emulator Super System), so MAME now documents a wide variety of (mostly vintage) computers, video game consoles and calculators, in addition to the arcade video games that were its initial focus.

License

The MAME project as a whole is distributed under the terms of the GNU General Public License, 2 (<https://opensource.org/licenses/GPL-2.0>) (GPL-2.0), since it contains code made available under multiple GPL-compatible licenses. A great majority of files (over 90% including core files) are under the BSD-3-Clause License (<https://opensource.org/licenses/BSD-3-Clause>) and we would encourage new contributors to distribute files under this license.

Please note that MAME is a registered trademark of Gregory Ember, and permission is required to use the "MAME" name, logo or wordmark.

Supported Platforms



Download

([release.php](#))



Download

([//wiki.mamedev.org/index.php/SDL_Supported_Platforms](https://wiki.mamedev.org/index.php/SDL_Supported_Platforms))



Download

([//wiki.mamedev.org/index.php/SDL_Supported_Platforms](https://wiki.mamedev.org/index.php/SDL_Supported_Platforms))

News

[< Newer \(/\)](#)

[Older > \(/?page=2\)](#)

Updated requirements for MAME (/?p=563)

07 Apr 2026

I hope you enjoyed our April Fools' Day announcement. With that out of the way, it's time to discuss actual upcoming changes to MAME. We're upgrading the development language standard from C++17 to C++20 and winding back support for obsolete configurations. We'll also be reducing the frequency of releases a bit, so there will no longer be a release nearly every month. There will be no April release; our next release will be near the end of May.

A summary of updated requirements:

- A compiler and C++ standard library with a reasonable level of C++20 support. GCC 11 is the oldest version of GCC that we will support. You can also use a reasonably up-to-date version of clang.
- Windows releases will require an updated installation of Windows 10 or later. Microsoft has already discontinued mainstream support for Windows 10, as well as all prior versions of Windows Home and Pro, and Windows 11 is already four years old.
- MAME's Qt-based debugger will require Qt 6.

A summary of some functionality we're removing:

- The 32-bit x86 (i686) recompiler back-end. It's been over two decades since the x86-64 architecture was introduced. All major x86 operating systems have supported x86-64 for years, and 32-bit x86 support is being

wound back.

- Support for compiling on OpenSolaris and other System V UNIX systems. There are no actively developed OpenSolaris distributions remaining, and no other System V UNIX variants have a meaningful presence on desktop systems.
- Specific optimisations for PowerPC host systems. PowerPC and OpenPOWER currently have no meaningful desktop presence, and the Libre-SOC project to produce a completely free, high-performance OpenPOWER implementation has stalled.
- The obsolete aueffectutil tool for macOS. This tool is no longer relevant with MAME's new audio output system, and it had not been updated to work with recent versions of macOS.
- The pre-built MSYS2 environments with included development tools. There are multiple issues with our MSYS2 environment that we can't practically solve.

Read on for some more background and details.

Read the rest of this entry » (</?p=563#readmore>)

Future plans announcement (</?p=562>)

01 Apr 2026

Hello everyone,

It is rare that the MAME Team addresses the community with news of this magnitude, and we want to approach this moment with the transparency and care it deserves. What follows is a candid account of where this project stands, where it is going, and why the decisions we have made — difficult as some of them are — represent the most responsible path forward for a codebase that has grown into one of the most complex preservation efforts in the history of open-source software.

On the Question of Technical Debt and Organizational Capacity

MAME has, for most of its existence, operated through the extraordinary dedication of volunteers who give their time and expertise freely. That generosity has produced something remarkable: a codebase that accurately emulates thousands of distinct hardware architectures, often to a degree of fidelity that no commercial effort has matched or even attempted. We do not take that legacy lightly. What we must acknowledge, however — and what we have been reluctant to state plainly until now — is that the accumulated complexity of that codebase has begun to exceed the realistic capacity of any volunteer-driven review process to maintain safely and sustainably.

The decision we are announcing today did not emerge from a single conversation or a sudden shift in priorities. It is the result of a long period of internal reflection on what it means to steward a project of this scope responsibly.

The Confirmed Direction: Rust Migration and AI-Assisted Review

Following extensive deliberation, the MAME Team has reached a unanimous decision to pursue a **phased architectural migration toward Rust**, to be introduced incrementally across subsystems beginning with the May release. Concurrent with this migration, all pull requests submitted to the project will be subject to **mandatory AI-assisted code review** prior to human maintainer evaluation. Submissions that do not satisfy the automated review criteria will be closed without further escalation.

We wish to be unambiguous: this is not a request for community input, nor is it a proposal subject to revision through discussion. The decision has been made. We are communicating it now because the community that has supported this work deserves to understand the reasoning behind it, not simply to receive the outcome.

May Release Scope and Migration Priorities

The May release represents the first concrete milestone in this transition. The initial migration scope includes:

- **Memory safety wrappers:** The highest-priority concern from a security and long-term maintainability standpoint.

- **Sound driver subsystems:** An area of the codebase where Rust's ownership model offers the most immediate and demonstrable benefits.

Our governing principle throughout this process is one that long-time contributors will recognize: if a driver cannot pass a cycle-accuracy parity check against the existing stable build, it does not ship. We will not sacrifice correctness for the sake of modernity.

Platform Targeting and Contributor Requirements

We recognize this element of the announcement may provoke significant discussion, and we want to address it directly.

Going forward, MAME's primary development target is a PC running a **recent version of Windows**, equipped with a GPU compliant with at minimum **DirectX 11 (SM5), OpenGL 4.3, or Vulkan**. This requirement is not arbitrary. The AI-assisted toolchain that underpins both code review and regression testing requires local model inference capabilities, and we believe it is reasonable to ask that contributors have access to hardware capable of running those tools.

We want to be clear about what this does and does not mean. Ports to other platforms remain explicitly permitted under the terms of the existing license. Those ports will be upstreamed as they become available, subject to the project's monthly token budget constraints. Modifications to source layout, core APIs, and OSD-layer support remain at the discretion of relevant fork maintainers, provided that any such modifications pass the AI-administered unit test suite prior to upstreaming. Contributions that exhaust the project's monthly token allocation will be queued to the following development cycle; we appreciate your patience as we calibrate these limits.

On the Appropriate and Responsible Use of AI in This Context

We are aware that any announcement involving AI tooling will raise legitimate questions about the integrity of the work being produced. We want to address those questions honestly.

The language models integrated into our development pipeline serve three specific, bounded functions:

- **Structural refactoring:** Models are used to map established C++ memory-safety patterns into idiomatic Rust. Every output is reviewed and validated by a human contributor before it is considered for inclusion.
- **Test matrix generation:** AI synthesizes comprehensive regression test cases derived from hardware datasheets, covering timing and behavioral edge cases at a scale that would be impractical to produce manually.
- **Cycle-accuracy verification:** The AI-administered test suite exists to confirm, not to assume, that emulation fidelity is preserved throughout the migration process.

We are not using generative tools to write drivers, to make architectural decisions, or to substitute for human expertise in matters of hardware behavior. The goal is augmentation of human capacity, not replacement of human judgment.

Communication Going Forward

Official updates regarding this migration will be communicated through:

- **GitHub:** Technical specifications, commit history, and branch tracking
- **mamedev.org:** Milestone summaries and roadmap documentation
- **Discord:** Community discussion and beta testing coordination

We are transitioning away from mailing lists as the primary channel for core development announcements. We recognize that this represents a meaningful change for a portion of our long-term community, and we have made this decision thoughtfully.

A Final Word

The history that MAME exists to preserve is irreplaceable. The hardware it documents is dying. The window during which accurate emulation can still be validated against physical reference hardware is narrowing. These are the stakes that motivate every decision described in this announcement, and they are the reason we have chosen to act now rather than continue managing decline incrementally.

The first milestone release will be accompanied by detailed comparison data demonstrating cycle-accuracy parity across all affected subsystems. We invite scrutiny of that data. We are confident it will speak for itself.

Thank you, as always, for the trust you have placed in this project.

MAME Core Team

MAME 0.287 (/?p=561)

31 Mar 2026

It's the end of another month, which means it's time for another MAME release! As you'd expect, MAME 0.287 includes a wide-ranging array of emulation improvements to a multitude of systems. Interesting changes this month include better Namco System 23 graphics, improved lighting for Sega Model 3, and software-controlled volume control/panning for Philips CD-i (along with improved stability).

The GRiD Compass family has received a keyboard overhaul as well as an initial DAC sound output implementation. The Apple II family now handles tricky raster effects more realistically, as well as getting a substantial software list update (metadata for the MECC collection is in much better shape). And speaking of software lists, a couple of NES prototypes have been added.

As always, you can read about everything that changed this month in the `whatsnew.txt` file (https://www.mamedev.org/releases/whatsnew_0287.txt). You can find the source code and 64-bit Windows binary packages linked from our download page (<https://www.mamedev.org/release.html>).

Read the rest of this entry » (</?p=561#readmore>)

MAME 0.286 (/?p=560)

26 Feb 2026

It's time for MAME 0.286, bringing you more adventures in emulation. This month, we've added support for SDL3, which will be used by default when building on macOS (the default is still SDL2 on other UNIX-like platforms). You can choose SDL2 or SDL3 by adding `OSD=sd1` or `OSD=sd13` to your build options, respectively. There may be some teething issues, so if you build against SDL3 and things break, let us know.

In arcade emulation, we've added a rare early version of Mario Bros. (which may have served as the basis for the Apple II port), the original Tecmo release of Back Fire, and the elusive Monkichicchi no Fuwafuwa Puzzle. Outside arcades, dozens of systems have seen updates over the past few weeks, including the ZX Spectrum and its descendants, the Apple II family and clones, the Epson QX-10, the Canon X-07, and the Sharp MZ-80B. There are also some nice software list updates, including the latest prototype cartridge dumps and plenty of homebrew software.

If you want to read about everything that changed this month, check out the `whatsnew.txt` file (https://www.mamedev.org/releases/whatsnew_0286.txt). As always, the source code and 64-bit Windows binary packages are available from our download page (<https://www.mamedev.org/release.html>).

Read the rest of this entry » (</?p=560#readmore>)

MAME 0.285 (/?p=559)

30 Jan 2026

Are you ready for the first MAME release of 2026? MAME 0.285 is out now! We've added support for an early, unreleased version of Atari's Relief Pitcher running on their System 1 platform. Moon Raker, a Nichibutsu shooter from the late '70s, has also been found and emulated. Sega's Waku Waku series has been further filled out with the addition of Waku Waku Jumbo. And for something completely different, Apple's first computer built around the WIMP paradigm, the Lisa, is starting the year in substantially better shape.

If you play arcade games that utilised lightguns, you'll no doubt be aware that pulling the trigger while aiming the gun away from the screen was a common way to reload. You may also be aware that MAME had an option to make this a bit easier if you're using a lightgun to play. That option has been removed, with a new plugin taking its place. The great news is that the plugin works *even if you aren't using a lightgun*. That's right, you can now assign a button to reload when you're playing Virtua Cop or Lethal Enforcers with a keyboard, mouse or trackball. Check the plugin documentation (<https://docs.mamedev.org/plugins/index.html>) for more details.

The Hanimex Pencil II computer has had a bit of an overhaul, which should make more software usable. A few more of the many BBC Micro peripherals have been emulated. Other improvements this month include better Apple IIgs periodic interrupt emulation, another working Apple II clone from behind the Iron Curtain, better representation of base Grid Compass hardware configurations, and better default sound routing for people using Apple notebook computers.

As always, you can read about everything that changed this month in the `whatsnew.txt` file (https://www.mamedev.org/releases/whatsnew_0285.txt), and the source code and 64-bit Windows binary packages are linked from our download page (<https://www.mamedev.org/release.html>).

Read the rest of this entry » (</?p=559#readmore>)

MAME 0.284 (</?p=558>)

31 Dec 2025

MAME 0.284 is ready just in time for the end of the year! Two interesting arcade prototypes have turned up this month. One is Saurian Front, an early version of what became Strike Force, developed at Williams before they acquired the Midway name. The other is a completely unreleased Atari game called Jammin' that runs on Donkey Kong hardware. There are a few more Sega Model 2 fixes this month, helping Virtua Cop 2 in particular.

The Brainchild PLS-1000, a hand-held educational device from the '90s that you may not even have heard of, is now emulated. You can now use various replacement keyboards for early Apple II computers, some of which had advanced features like macro recording. There were some big software list updates this month for quite a few computers. The floptool disk image utility included with MAME has improved usability. You can even simulate receiving a Sega Channel broadcast on your emulated Sega Genesis.

Of course, there's far more in this release than we can talk about here, and you can read all about it in the `whatsnew.txt` file (https://www.mamedev.org/releases/whatsnew_0284.txt). The source code and 64-bit Windows binary packages are linked from our download page (<https://www.mamedev.org/release.html>). All the best in the new year from MAMEdev!

Read the rest of this entry » (</?p=558#readmore>)

MAME 0.283 (</?p=557>)

29 Nov 2025

With the end of the year barely more than a month away, it's time for MAME 0.283! As you may be anticipating, there are *even more* Sega Model 2 fixes this month. Trilinear luma filtering should be working now, and some glitches in tilemap layers are fixed. Microtexturing, used to good effect in The House of the Dead, is emulated for the first time. Some of the tilemap layer fixes have spilled across into Sega System 24 as well.

Several Ensoniq synthesisers have been promoted to working in this release. The Sequential Circuits Six-Trak analog synthesiser has also been promoted to working, with improvements there benefiting the related Bally/Sente 6VB sound board. There's been quite a bit of work on NEC computers this month, and there are some new software lists for Sharp home computers. MAME now emulates all supported (and some unsupported) video modes on the IBM PCjr.

To find out about everything else that's happened in MAME development this month, you'll have to read the `whatsnew.txt` file (https://www.mamedev.org/releases/whatsnew_0283.txt). As always, you can get the source code and 64-bit Windows binary packages from the download page (<https://www.mamedev.org/release.html>).

Read the rest of this entry » (</?p=557#readmore>)

[< Newer \(/\)](#)

[Older > \(/?page=2\)](#)