I'm not a robot

Render your favorite Dreamcast classics in 1080p or 4k like they deserve. No controller configuration, no BIOS or flash files, just add games and play. Over 95% of the Dreamcast's library is able to be played from start to finish. February 13th, 2020 Hot off the presses is our latest stable, version 1.5.0, marking the second stable release since the last progress report. In this past year, support has been added for multiple new platforms to make the emulator accessible, performance has dramatically increased, new features such as save states and cheat support have landed to make emulating more fun, and numerous accuracy improvements were made to continue polishing the overall emulation experience. Continue reading December 14th, 2019 Back in July the first video of redream running on the Raspberry Pi 4b was posted. The plan at the time was to release it within 2 weeks, but other priorities came up and getting the Pi release to production quality was put on hold. Continue reading August 13th, 2018 Last week we released our newest stable build, version 1.2.6; marking 6 months since our last progress report. During this time, many quality of life features have been implemented to make the emulator easier and more enjoyable to use, graphics have taken another step forward, and our audio emulation has finally reached a mature point where these features yet to be implemented, but what's implemented now works great. Continue reading July 21st, 2018 Earlier this month v1.2.3 was released, introducing a major change to our audio / video synchronization that I'm excited to talk about today. Continue reading January 30th, 2018 It's been 6 long months since our last status update, but for good reason - we've been hard at work on plenty of new features, improvements and optimizations that we'd like to talk about today. Continue reading July 19th, 2017 After a slow period at the beginning of the year, the pace has started picking up the past few months on redream's development. Continue reading Reddit and its partners use cookies and similar technologies to provide you with a better experience. By accepting all cookies, you agree to our use of cookies to deliver and maintain our services and site, improve the quality of Reddit, personalize Reddit content and advertising, and measure the effectiveness of advertising. By rejecting non-essential cookies, Reddit may still use certain cookies to ensure the proper functionality of our platform. For more information, please see our Cookie Notice and our Privacy Policy. Reddit and its partners use cookies and similar technologies to provide you with a better experience. By accepting all cookies, you agree to our use of cookies to deliver and maintain our services and site, improve the quality of Reddit, personalize Reddit content and advertising, and measure the effectiveness of advertising. By rejecting non-essential cookies, Reddit may still use certain cookies to ensure the proper functionality of our platform. For more information, please see our Cookie Notice and our Privacy Policy. Please take a look at the below list of commonly asked questions. If you can't find an answer, join our Discord server where we can personally assist you. The minimum requirements for Desktop are: Operating system Windows 7+ 64-bit, macOS 64-bit or Linux 64-bit Processor x86_64 compatible with AVX2 or SSE2 support Memory 512 MB RAM Graphics OpenGL 3.1 and for Android: Operating system Android 5.0 Processor 32-bit / 64-bit ARMv7 / ARMv8 CPU Memory 512 MB RAM Graphics OpenGL ES 3.1 There is no installation process for redream - download it, extract it to a folder and run. If you've upgraded and have a license, download and place it in redream's working directory: Windows and Linux Same directory as executable Mac ~/Library/Application Support/redream/ Redream supports any GDI, BIN/CUE, CHD or CDI disc image. However, not all images are created equal and we strongly advise against using CDI images. A note about CDI images CDI images take the original 1 GB GD-ROM and modify it to fit on a ~700 MB CD-ROM in order to be burned and used on a real Dreamcast. When the games are modified to fit on a CD-ROM, the audio and video are often compressed, the code is modified (sometimes resulting in crashes even on real hardware) and the metadata stored on each disc which is used to provide cover art and cheats inside Redream is altered. As a result of these modifications, CDI images can sometimes be less than 100 MB in size, In contrast, since each GD-ROM was 1 GB in size, GDI and BIN/CUE images will always be 1 GB in size as they are exact copies of the original game. If hard drive space is an issue, it's recommended to convert GDI images to the CHD format which provides lossless compression of the original game without altering its integrity. Loading a game through the UI Navigate to Library -> Add Directory to add a directory containing valid game(s) to the library. After doing so, each game should be available from the main menu: Loading a game through the command line For quick testing, or for integrating with a frontend such as Launchbox, it's often desirable to run a game directly from the command line. This can be done by passing the full path to the game as the first argument to redream: redream.exe C:\path\to\game By default, the keyboard is bound to the first player and the default binds are: The keyboard can be bound to all four players, with unique binds for each, enabling multi-player arcade sticks which register themselves as keyboard to easily be used. Using a controller is as simple as connecting it for most devices. To date, the only accuracy related option available in redream is the per-pixel polygon sorting. This is because requiring it would, at this time at least, exclude a double digit percentage of the user base due to its performance controller to, and then select the controller to use on the port: After selecting an input device, select Customize binds. In this menu, click the button you'd like to bind and then press the key or button you'd like it mapped to: Redream v1.5.0 is released. Redream (originally called Dreavm) is a work in progress emulator for the SEGA Dreamcast. Redream Changelog:AddedDisc swap support.Redump BIN/CUE support.Widescreen hack support.Codebreaker cheat support.Raspberry Pi 4 support.HLE BIOS font support.Exclusive fullscreen support.High DPI support on Windows and Mac.Nintendo Switch controller support.PowerVR trilinear filtering support.PowerVR secondary accumulation buffer support.PowerVR parameter selection modifier volume support.Register aliasing support to improve ARM32 register allocations. ChangedNumerous CPU and GPU performance improvements as a result of the Pi bringup.Removed GL_ARB_fragment_shader_interlock dependency for per-pixel sorting.Ignore removable media drives on Windows which have no disk in them.Library scanning responsiveness improved for large libraries.Library scanning is now limited to a single worker thread.Filter out noisy axes when catching key binds. FixedMajor HLE BIOS refactor, fixed all known incompatibilities.Refactored renderer to support rendering shadows before fog.Video output only filling half the screen in some PAL games.Video output flickering over component in some PAL games.Codebreaker main menu not rendering.Langrisser Millenium main menu not rendering.NFL 2K/2K2 field textures not blending correctly.NFL Blitz 2001 shadows not blending correctly.NHL 2K2 ice appearing black.Real Sound framebuffer vertical bands.World Series Baseball 2K2 blurry textures.V-Rally 2 main menu flickering.Numerous AICA stream fixes.Missing CDDA music in a few games.Decoding of ADPCM audio to be bit-perfect,Sega Bass Fishing audio going off the rails.Capcom Fighting Collection hanging on boot.Fast Striker crashing at the end of level 1.Gundam Side Story crashing during third mission.Tech Romancer crashing after a few rounds.Library scanning not working with exFAT drives. Download: Redream v1.5.0Source: Here Reddit and its partners use cookies and similar technologies to provide you with a better experience. By accepting all cookies, you agree to our use of cookies to deliver and maintain our services and site, improve the quality of Reddit, personalize Reddit content and advertising, and measure the effectiveness of advertising. By rejecting non-essential cookies, Reddit may still use certain cookies to ensure the proper functionality of our platform. For more information, please see our Cookie Notice and our Privacy Policy. Reddit and its partners use cookies and similar technologies to provide you with a better experience. By accepting all cookies, you agree to our use of cookies to deliver and maintain our services and site, improve the quality of Reddit, personalize Reddit content and advertising, and measure the effectiveness of advertising. By rejecting non-essential cookies, Reddit may still use certain cookies to ensure the proper functionality of our platform. For more information, please see our Cookie Notice and our Privacy Policy. In this past year, support has been added for multiple new platforms to make the emulator accessible, performance has dramatically increased, new features such as save states and cheat support have landed to make emulating more fun, and numerous accuracy improvements were made to continue polishing the overall emulation experience.Now on Android and Raspberry PiThe largest push this past year has been getting redream running on power-efficient but low-performance ARM platforms. To this end:The app was updated to work on 32-bit platforms.The rendering backend has been upgraded to support OpenGL ES.New JIT compilers were written for 32-bit and 64-bit ARM.User interface was overhauled to work well with touch devices.In addition to this work which was required to just get up and running, a huge effort has went into optimizing in order to ship the exact same product that's available for Windows / Mac / Linux on these platforms. That is, there are no speed hacks, no underclocking and no different or gimped code paths - users running on the Android and Pi get the exact same product.Outside of optimizing and improving redream itself, patches have been upstreamed to Mesa to grind out more performance on the Pi and patches have been pushed to SDL to expand on its KMS driver. These patches serve to not just improve redream, but any OpenGL applications running on the Raspberry Pi. Performance improvementsRedream has never slacked on performance, but during the past year a substantial amount of effort has went into optimizing it even further. The reason for the constant focus on optimization is twofold:The obvious, games needs to run full speed on these low-power devices.The less obvious, for each new feature and each accuracy improvement we refuse to compromise on performance.Refusing to compromise means that these new features and accuracy improvements aren't tucked away behind options and that each is thoroughly tested for performance regressions. If a change is fundamentally slower, all other code is on the optimization chopping block until performance breaks even.An example of this are the changes upstreamed to Mesa. Our renderer had been through the wringer for weeks and performance still wasn't where it needed to be on the Pi. Rather than claim the hardware was too slow or put a few features behind options, another ~10 percent was instead squeezed out of the driver. To date, the only accuracy related option available in redream is the per-pixel polygon sorting. This is because requiring it would, at this time at least, exclude a double digit percentage of the user base due to its performance impact.Jumping into the numbers, here's a video that was made right after Android came out but before Pi bringup. It shows side by side footage of a sample of games running uncapped between 1.3.0 and what became 1.4.0. On the low end, many games gained around ~25% speed. On the high end, games as much as doubled in speed: After Pi bringup games have again made another leap in performance. Here's a view of the raw frame rates on a few different platforms: These results were gathered by running the first 10,000 frames of each game's attract mode using redream's --runframes argument with a retail BIOS. When this argument is used frame skip is disabled (so all frames are fully rendered) as is vsync and the results are printed out to the log at the end of the run.General improvements For the uninitiated, save states are an additional way to save your progress in a game that's special to emulators. Save states are generated by saving the entire state of the emulator to a file, which can then be loaded at any time enabling users to save and load their progress outside of the constraints of the original game.This means that each save state stores the state of the emulated CPU, the GPU, the GD-ROM - everything. Given that, as the emulator changes and evolves so does this state, and therefore so does the save state format. When the save state format changes emulators can choose to either make old saves incompatible with new versions of the emulator, or they can do added work to make each new version backwards compatible with old saves. If the emulator's state is rapidly changing during development, maintaining backwards compatibility can quickly eat into development time for other features.With Android and ARM support complete, our code churn had settled enough to finally implement backwards compatible save states. Since release the format has been updated around 10 times without breaking compatibility. Disc swap supportMost multi-disc games on the Dreamcast don't require either disc swapping in order to be played through: they support saving at the end of disc 1, powering off and then starting up disc 2. However, a handful of applications and games (Codebreaker, D2, Pop'n Music 3 and 4, etc.) require that the second disc be swapped while the system is still powered on, which is something that's been sorely missing.Initial support for swapping GDI images with a real BIOS was added a few months back, but then we went down the rabbit hole of adding support for CDI images as well as swapping with the HLE BIOS. Supporting swapping GDIs with the real BIOS for games that require it was not work, but numerous issues popped up once the support matrix expanded to GDI x CDI x real BIOS x HLE BIOS x all multi-disc games.After a week or two of testing and prodding the GD-ROM hardware these issues were rooted out and disc swapping is now supported for all image types, with or without a BIOS. Cheat supportGoing into this I had never used an external cheat device (GameShark, Code Breaker, etc.) and by the end this became one of my favorite features from this release.While I hadn't used these devices, I'd seen cheats such as 011EC4B4 0000173E (which for Crazy Taxi gives you an infinite amount of game time) but what did this cheat actually do? As it turns out, there are many good resources scattered around the internet to help answer this question, and for most devices the cheats break down similarly:Each 4 byte word (11223344) is an instruction.The first (from left to right) byte represents an operation.The remaining 3 bytes are arguments for the operation.Additional words can follow to provide additional arguments and form conditionals.For 011EC4B4 0000173E, the cheat decodes like so:01 is the operation, which performs a 16-bit write.1EC4B4 is the offset in system ram to write the 16-bit value to.173E is the value to be written.So that's cool, these cheat codes are their own little programming language, but how should the code be run to actually apply the cheat? Manually applying a few cheats while a game was running worked fine, but some would apply and quickly stop. These devices weren't just applying cheats once at startup, they must be applying them periodically.After some digging around, I read about master codes which made the process somewhat click - they were somehow modifying games to call into their own code which would apply the cheats. However, master codes are read depending on which master code which would apply the cheats. However, master codes were read depending on which master code for CDI images as well as swapping specific, so there must be some generic way they do this, as a default. I couldn't think of a way to generically patch the games, but was reminded of how some of the release groups modified the BIOS system calls to bypass security checks. After remembering that and adding some checks to monitor writes to the system calls I found my answer - they were patching one of the frequently used GD-ROM system calls to jump to their own code which would apply the cheats.With the actual implementation of decoding and applying cheats sorted, a large number of publically available cheats were imported which show up automatically for GDIs in the Manage cheats menu. Support for adding custom cheats isn't available yet, but a menu for editing them will be added during the next round of cheats work. Widescreen cheatsBuilding on the cheat support mentioned above, we also rounded up and manually verified the widescreen cheats created by the members of the AssemblerGames forums (Espirral, yzb37859365, Radaron, ELOTROLADO.NE, Joel, S4ppb4rad and VIRGIN).These can be enabled for supported games from the same cheats menu shown above. Since 1.4.0, all known compatibility issues have been fixed with our replacement BIOS, support for the built-in BIOS font was added, and every issue (other than JP character support in the BIOS font, and support for some homebrew / cracked CDIs) has been closed out in the issue teacher.If a game runs with the retail BIOS, it will also run with our replacement HLE BIOS.Graphics improvements The Dreamcast's PowerVR supported what they referred to as "modifier volumes." These modifier volumes were user-defined 3D volumes that could be programmed to change how lighting was calculated for geometry that was either inside or outside of the volume.The PowerVR supported two different modes for processing the modifier volumes:Intensity Volume Mode. These are used by many games and have been supported for a long time. In this mode, if a pixel is inside a volume the colors used to shade the pixel are scaled by a single value, typically to darken the final color to look like a shadow.Parameter Selection Volume Mode. These are only used by a handful of games, but are now supported. In this mode, surfaces can send two different sets of texture and shading parameters - one set that is used if the final pixel is inside of a modifier volume and one that is used if the final pixel is outside a modifier volume. With this, games can do more dynamic effects such as changing the color or texture of a pixel based on if it's inside our outside of a modifier volume.This mode is used by Speed Devils in night time races to produce the headlight effect. Speed Devils at night without this effect Intermdiate results, pixels inside of a volume are red, pixels outside are black Final results, with different shading applied to pixels inside the volumes Fog SupportThe PowerVR supported both per-vertex and per-pixel fog modes, both of which are now emulated. In addition to emulating the fog, the renderer has been refactored to correctly render shadows when fog is used. Previously, the entire scene was shaded, and shadows were cheaply rendered on top of the final scene. This works great when there is no fog, but unfortunately fog processing comes after modifier volume shading in the graphics pipeline, which makes some scenes to look wrong with the original approach: Shadows incorrectly applied after fog Shadows correctly applied before fog This is now fixed, but the refactor was bittersweet - the improved accuracy comes with a large performance hit (particularly on mobile) for all games, while only a handful of games are noticably improved by it. Support has been added for a rarely used PowerVR feature - the secondary accumulation buffer. The secondary accumulation buffer was a secondary render target that could be drawn to, and then sampled from, during the same frame. This could be used to provide multi-texturing, bump mapping and trilinear filtering but in practice it was only used by a handful of retail games (Evil Dead and a few 2K sports games).NHL 2K2 without the secondary accumulation buffer output Ice and decal texture composited to the secondary accumulation buffer NHL 2K2 with the secondary accumulation buffer output Trilinear FilteringFor some background, trilinear filtering works by interpolating the results of two adjacent bilinear filtered mipmaps. Modern GPUs have dedicated hardware to select two mipmaps and filter between them, but with the Dreamcast the user had to do additional work in the form of rendering the polygon once for each mipmap level and manually blending them.In order to render a trilinear filtered, opaque polygon the user had to:Render polygon, tweaking mipmap selection to use the higher resolution mipmap.Render polygon, tweaking mipmap selection to use the lower resolution mipmap.To tweak the mipmap selection, a decimal coefficient was supplied for each polygon which could be used to select a higher or lower mipmap level. With this, games can do more dynamic effects such as changing the color or texture of a pixel based on if it's inside our outside of a modifier volume.This mode used to scale the final color enabling the results to be additively blended.Of this process, the mipmap level selection tweaking is not yet emulated, but scaling the final color by the fractional portion of the coefficient is. This causes the final color to be fractional blended incorrectly:D coefficient not honored, colors get added together twice D coefficient honored, blending is correct In ClosingThe past year has been a long one - it's been a delicate balance of adding new platforms, optimizing for these platforms as well as adding new features and improving compatibility and accuracy along the way. For a more complete list of changes, you can view the changelog.With these new platforms out of the way, we're really excited to focus purely on improving compatibility over the next few months, with some of the major items to soon be tackled are:Improved input device emulation. Adding support for emulating the Dreamcast's jump pack, keyboard, etc.Improved audio emulation. Adding DSP support, low-pass filter support and more.Windows CE support. Compatibility with non-Windows CE is nearly complete, it's finally time to add this.If you'd like to follow along with the development or suggest improvements, come join us on our Discord server: General Dreamcast discussion applies here. Before posting here please check the other forums in the Dreamcast section to see if your topic would fit better in those categories. retrofan01 rebel Posts: 20 Post by retrofan01 Sun Dec 31, 2023 12:25 pm Instructions - Version 2.6 (use only for OpenMenu/GDEmu - this will NOT fit on a standard 200 block VMU): Extract FCDCHEATS.BIN from the ZIP file then copy it to the following folder inside GDMENU Card Manager, overwriting the old one: OPENMENU\GDMENUCardManager.v2.0.1-win-x64\tools\openMenu data\cheats\ Of course, you must then open GDMENU Card Manager with your SD card loaded, then click "Save Changes" in order to rebuild openMenu with the new cheat database. Instructions - Version 1.5 (use for all other situations - this is the only version of the file that will fit on a standard 200 block VMU): If you have a VM2 or similar device, copy CODEBREA.VMU (found inside the VM2 folder) to the appropriate location on your VM2's SD card. If you don't have a VM2 or similar device, you can either:- use ateam's VMU Disc Builder to create a CDI image containing the supplied .VMI and .VMS files (found inside the VMU Disc Builder folder) - use a Dreamcast SD card adaptor to copy the FCDCHEATS file (found inside the DreamShell folder) over to your VMU. To do this: Copy the FCDCHEATS file to an SD card then insert the SD card into your Dreamcast SD card adaptor. Power on your Dreamcast and load DreamShell (I used 4.0.0 Beta 4 which is an older version) then run File Manager. File Manager must then be used to copy FCDCHEATS from the SD card (/sd) over to your VMU (/vmu/a1, /vmu/a2, /vmu/a3 or /vmu/a4 depending on which controller port your VMU is connected to). Importing the file onto a VMU file using VMU Explorer will not work. The FCDCHEATS file uses 194 out of 200 blocks so it will require its own VMU. The fcdcheats.txt file containing all the codes is also included. This can be loaded into Xplorer DC Code Compiler and edited further with additional or replacement codes. --- After several years of manually inputting Codebreaker codes using the DC controller whenever I wanted to try a new game, I discovered Xplorer DC Code Compiler recently and decided to put together a complete CodeBreaker cheats file containing: - all unique codes from GameHacking.org- codes already built into CodeBreaker- widescreen codes- other codes from the Dreamcast-Talk forum I used the following criteria to determine which codes to add: - one game one region (1G1R) - this is mostly USA with some Japan-only games but there are a few games which I included PAL cheats for due to one of the following reasons: - I've personally played through those games in their entirely using those codes and found they worked better than their NTSC equivalents (e.g. Record of Lodoss War, Silver) - cheat code creators have posted codes for a PAL version of a game which have no NTSC equivalent - all codes are decrypted (CodeBreaker/XplorerDC). Any encrypted CodeBreaker codes have been converted to decrypted format using DCrypt -- Changelog - Version 2.6 (2024-12-28):- Added codes for:Alienfront Online (megavolt85)Aqua GT (unknown, )Baldr Force EXE (bucanero)Buzz Lightyear of Star Command (unknown, megavolt85)Evil Twin: Cyprien's Chronicles (Sharpnull)Kao The Kangaroo (bucanero)Kidou Senkan Nadesico: Nadesico the Mission (bucanero)Metal Slug 6 (LuC-iTA)Moero! Justice Gakuen (bucanero)Prism Heart (bucanero)Puyo Puyo Fever (bucanero)Record of Lodoss War (bucanero)Shadow Man V1.1 (foxx)Shikigami No Shiro 2 (bucanero)Shin Seiki Evangelion - Ayanami Ikusei Keikaku (bucanero)Stunt GP (foxx)Tantei Shinshi Dash (bucanero)Trickstyle (bucanero)Tricolore Crise (bucanero)Walt Disney World Quest - Magical Racing Tour (foxx) - Altered Unlock Storyboards code for Spider-Man- Misc fixes Changelog - Version 1.5 (2025-04-24):- Added codes for:Alone In The Dark: The New Nightmare (LuC-iTA)Arcade Racing Legends:Road Trip Edition (LuC-iTA)Aqua GT (LuC-iTA)Baldr Force EXE (bucanero)Bikkun'Man 2000 - Viva! Festival (LuC-iTA)Border Down (dakidski, LuC-iTA)Buggy Heat (Europe) (LuC-iTA)Chaos Field (dakidski, LuC-iTA)Disney's Dinosaur (CodeJunkies)Exhibition of Speed (LuC-iTA)F1 World Grand Prix (LuC-iTA)F1 World Grand Prix II (LuC-iTA)Faster Than Speed (LuC-iTA)Fushigi no Dungeon: Furai no Shiren Gaiden (LuC-iTA)Half Life (CodeJunkies)Headhunter (LuC-iTA)Ikaruga (dakidski, LuC-iTA)Jet Set Radio (Code Master, Slab)Karous (LuC-iTA)Knights of Valour: The Seven Spirits (LuC-iTA)Le Mans 24 Hours (LuC-iTA)Maximum Speed (LuC-iTA)Maximum Speed 6 (LuC-iTA)NBA Showtime: NBA On NBC (Sage, Code Master)POD 2 - Multiplayer Online (LuC-iTA)Psyvariar 2 (dakidski, LuC-iTA)Puyo Puyo Fever (bucanero)Racing Simulation 2 - Monaco Grand Prix On-Line (LuC-iTA)Radirgy (dakidski, LuC-iTA)Shikigami no Shiro II (LuC-iTA)Sega Tetris (VB[Daryl])Sega Worldwide Soccer 2000: Euro Edition (CodeJunkies)Sonic Adventure (Sage, VB[Daryl])Stunt GP (LuC-iTA)Time Stalkers (Code Master)Trickstyle (LuC-iTA)Triggerheart Exelica (LuC-iTA)Trizeal (dakidski, LuC-iTA)Twinkle Star Sprites (LuC-iTA)Typing of the Dead (Raven-187)Under Defeat (dakidski, LuC-iTA)V-Rally 2: Expert Edition (LuC-iTA)Wacky Races (LuC-iTA)Walt Disney World Quest - Magical Racing Tour (foxx) - Magical Racing Tour (foxx) - Removed codes for:Buggy Heat (Japan)Sega Smash Pack Vol. 1 - Added disc names for all remaining non-multi disc games- Removed all remaining redundant encrypted codes- Altered Unlock Storyboards code for Spider-Man- Misc fixes Attachments CodeBreaker all cheats save file 1.5.zip (139.39 KiB) Downloaded 230 times CodeBreaker all cheats save file 2.6.zip (124.3 KiB) Downloaded 2467 times xpdccc.zip (80.28 KiB) Downloaded 3233 times Last edited by retrofan01 on Sat Apr 26, 2025 3:51 pm, edited 11 times in total. Woodneeded lithium Posts: 35 Post by Woodneeded Sun Dec 31, 2023 1:27 pm Awesome thank you for this. Apologies for my ignorance here but what is the best way to get this save file onto a memory card? I should know how to do this but I guess I do not MetroidTalon shadow Posts: 6 Post by MetroidTalon Sun Dec 31, 2023 3:23 pm Is there a what so I do with this or how to use guide? retrofan01 rebel Posts: 20 Post by retrofan01 Sun Dec 31, 2023 5:23 pm Woodneeded wrote:Awesome thank you for this. Apologies for my ignorance here but what is the best way to get this save file onto a memory card? I should know how to do this but I guess I do not Copy FCDCHEATS onto an SD card, then use a Dreamcast SD card adaptor and File Manager in DreamShell to copy the file onto a VMU. ateam wrote:Thank you so much for putting this together! Thank you for all your efforts in the DC translation space - Sakura Wars Columns 2 was one of the games I very much enjoyed in 2023. I have no programming or hardware skills so this is my small way of giving back to the community mistamontiel Shark Patrol Posts: 2155 Dreamcast Games you play Online: Errythan except Tetris o.0 Location: Miami, FL, CUBA Contact: Contact mistamontiel Post by mistamontiel Sun Dec 31, 2023 7:53 pm For the life of me CB refuses to save. I recall even trying other saves of it and would be ignored Tried with formatted VMU all same CB is only against meGreat to see Rent A Hero and Nanatsu Seven Mansions! When my VM2 finally lands I'm definitely doing this and hope FloDoc shadow Posts: 10 Post by FloDoc Mon Jan 01, 2024 3:32 pm Hahahaha! Man, I just finished making a txt file with all the codes I could find, but it was way too big for the VMU! I will check this one out for sure! Thanks for the effort here! I was trying to figure out if there would be a way to just have the codes added directly to the PELICAN.BIN so the VMU can just be avoided altogether. I think it's possible from what I've read, but I have no idea how to do it lol!Nice work here! seedsbower noob Posts: 1 Post by seedsbower noob Mon Jan 08, 2024 5:05 am MetroidTalon wrote:Is there a what so I do with this or how to use guide? retrofan01 rebel Posts: 20 Post by retrofan01 Mon Jan 08, 2024 7:08 pm This is awesome but is there a way to modify the Codebreaker ISO with these cheats and make a new bootable CDI with everything in it? I believe its been done with the Dream Explorer image at one point. deluxux Super Sonic Posts: 1551 Post by deluxux Mon Jan 08, 2024 9:57 pm The latest openmenu has codebreaker built in.. Can we use this file with it 1 Replies 4695 Views Last post by chrisrlink Fri Dec 24, 2021 11:33 pm Updated DreamPi Pso save not working Last post by Rainwarner88 Tue Jul 11, 2023 3:21 pm Posted in Online Replies: 2 by Rainwarner88 Tue Jul 11, 2023 2:39 pm in Online First post Hi all. I've downloaded the latest version of DreamPI but upon loading up PSO to test it won't let me play my character just keeps asking me to... Last post I reinstalled the dreamcast took out the vmu and back in off and on again and something happened because it works now. Thankyou for answering so... 2 Replies 3392 Views Last post by Rainwarner88 Tue Jul 11, 2023 3:21 pm Sega Tetris all characters save file Last post by Xiden Fri Aug 30, 2024 7:36 pm Posted in Support Replies: 3 by Mechasonik Thu Jun 10, 2021 9:35 pm in Support First post Apparently there were 2 character colors that were unlocked by connecting online, so Im asking if anyone has a save file with all the characters... Last post Just play 128 matches online and you can unlock everythingish... still have to get coins and get them out of the UFO catcher. 3 Replies 5625 Views Last post by Xiden Fri Aug 30, 2024 7:36 pm 0 Replies 3610 Views Last post by bshi02 Mon May 09, 2022 7:57 am 1 Replies 10840 Views Last post by krovellium Wed Jun 01, 2022 2:49 pm

How to use redream emulator. Redream dreamcast cheats. How to use cheats on redream emulator. Redream android cheats. Redream cheats.