

Pixar-Cars Changelog

v1.07 (2-7-13)

- Audit added to track games with an autopilot
- Audit added for tracking failed dongle writes
- Reduction of motion blur

v1.06 (2-6-13)

- Fixed bug where RIO reconnects weren't logging properly in menu screens
- Fixed a bug where game could go into maint reboot in test menu
- Minor wording change on calib menu to include instruction on using start button as well as test button
- AI skill changed to allow for fewer last place finishes
- AI cars have sidestep move removed

v1.05 (2-5-13)

- Handling changes
- Dongle retry for coin writes
- Handling changes
- Dongle retry for coin writes

Cars Changelog 0.98 - 1.04

- Steering & handling vastly improved & more forgiving
- Much improvement of static shock handling & I/O board reconnection
- Talk button added to allow players to make Cars chatter
- Robustness added to audits & adjusts file
- Added uptime/serial # display (hold Talk button for 5 seconds in attract)
- Messaging added for autopilot
- Menu visible timeout timer added to game menus
- Shortened menu select times, 15 seconds per screen
- Added adjust for number of laps, defaults to 1
- Improvements & fixes to all audit information in test menu
- Enabled USB output of audit dump
- Dongle serial number support added
- Steering wheel use cancels track playback in attract
- Added many audits for system performance tracking
- Test menus cleaned up
- Improved fonts

Changelog for 0.86-0.98

-
- Force feedback cap to prevent possible washing machine motion
 - Added maintenance reboot if framerate drops
 - Added audit for FPS-related maintenance reboot
 - Added a watchdog test to diagnostic menu
 - Start spam crash fix
 - Lamp animations added
 - Force feedback moved to a thread
 - Watchdog test added
 - Coin meter test added
 - New videos from Disney
 - New attract play videos added
 - Sound tests added
 - Steering tweaks to improve gameplay
 - Many RIO reconnect fixes

6/7/12 - 3

ArcadeData.h, Audits.cpp, ArcAuditMenu.cpp: Added an audit to track when the RIO board returns after a drop.

ArcadeHud.cpp/h: Tweaked out the existing text line positions/sizes to use them for the system message routine. Changed default system message to disable.

ArcadeManager.cpp/h: Added code to properly handle the RIO disconnects, made sure all audits for disconnects/reconnects are saved to the audits, and added routine to make the Cars driving around in the menus be able to be disabled.

CarsFrontEnd.cpp: Remove car actors (Holly, etc.) when displaying system messages

RioInterface.cpp/h, rio.c, usbdev.c: Finished up the RIO reconnect/disconnect stuff at the system level, fixing some bugs along the way that were introduced with the switch to winusb (well, things that didn't work the same needed to be worked out, and the main app checks for the reconnect instead of the thread, which is much better for a bunch of reasons).

6/7/12 - 2

- ArcTestMenu.cpp: Forgot to switch the reboot message over to the new system message call for non-debug.

6/7/12 - 1

- ArcadeHud.cpp/h: Changed messages to use a generic system message routine, and had them use existing strings. I may have to add a string to make sure it's centered, but for now, this prevents any creation of new strings, and it centralizes all of the system messages to one routine.

- ArcadeManager.cpp/h: Added code for RIO disconnect messages, using the new system message routine in ArcadeHud. Also improved the disabling of the maintenance reboot stuff, so it no longer even does the tally if the maint is disabled.
- ArcTestMenu.cpp: Changed handling of reboot message to use system message routine.
- RioInterface.cpp/h: Changed some of the flags to be static instead of part of the class. Since the Arcade stuff needs the RIO to exist before it does anything, it should be fine to check it regardless. It's all initialized correctly, so even if the static elements are accessed when RIO isn't present, everything should function ok.
- Controller.Rio.cpp: Changed a call to use the new static routines for RioInterface.cpp.
- usbdev.c: found a bug where the connect was returning a pointer even if there wasn't one. This was introduced with the switch to winusb.
- GameCommonLoop.cpp: Commented out define that was spitting out memory debug info.

6/6/12

- ArcadeHud.cpp/h: Added code to display a variety of system messages, including one for maintenance reboot, and the messages test mode uses when it exits
- ArcadeManager.cpp/h: Added a tally system for maintenance reboot. Basically, when a track is selected, a number is added to a global tally. When it reaches a certain threshold, the game starts to attempt a maintenance reboot. When it reaches a second threshold, it triggers a maintenance reboot after the next game finishes, regardless of idle time. In order to do this, a callback had to be added to the video system to get it to interrupt a video.
- ArcTestMenu.cpp: Changed display of rebooting to use new ArcadeHud system message.
- CarsPickup.cpp: Added a check for a null pointer based upon a crash JT & I saw.
- CarsFrontEnd.cpp/h: Added code for the maint reboot tally, and adjusted the autoplay code to work better in different situations (maint/no maint, hard code level selection, etc)
- RaceManager.cpp: Added two defines to handle timeouts based upon whether or not autoplay is going. Shortened the normal input timeout to engage the autopilot to 2 seconds.
- Memory.h: Commented out memory debug stuff
- Fmv.cpp/h: Added a callback to poke a timer abort into a movie being played
- fmvdriver.cpp: More abort callback code
- Input.arcade.cpp: Begun to add support to use keyboard/mouse for debugging
- RioInterface.cpp: Fixed a bug where the enable/disable of manual watchdog kicking wasn't checked properly
- rio.c: Added a check so that if the watchdog changed flag isn't set, the RIO code doesn't adjust variables
- StandAloneLoop.cpp: Removed debug memory define