Paragon File System 2.0

What Is Paragon FS

Paragon File System (Paragon FS, PFS) is a modular file system designed to extend flash memory lifetime, provide high performance, and operate in RTOSes. Due to its modular structure, Paragon FS is a perfect choice for pre-OS environments, Automotive, Industrial Autonomous Systems, and IoT use.

Purpose

High-performance, low wear, fail-safe, data-critical filesystem for embedded use with flash memory

Key benefits

For business	• Footprint/performance optimization — fine-tuning for a wide range of use-case scenarios - from tiny IoT to automotive VCE/ECU hardware • AUTOSAR® compliance — simple and easy way to meet compliance for integrated solutions with Paragon FS onboard
For developers	• POSIX® support provides well-known APIs as well as ACLs, hard links, and symbolic links • Any flash storage device is supported: both NAND and NOR • Easy integration — Paragon FS is already ported to various embedded and RTOSes. If your OS is not on our list, Paragon can assist with getting the system as a set of libraries to integrate them into your OS
For users	• Tree-based metadata enables quick data search and metadata versions navigation • Transaction support ensures all of your data, including metadata, is intact • Dynamic wear leveling and copy-on-write technology maximize flash memory life

Why Paragon File System 2.0

• AUTOSAR® compliant
• Fail-safe by design
• Flash optimized performance and wear
• RTOS/IoT optimized
• VFS emulation

• Block and file-level cache
• POSIX® compliance
• Symbolic and hard links
• Static Memory Allocation

Features

Feature	Description
Fail-safe	Paragon FS uses transactions to ensure the atomicity, consistency, isolation, and durability of your data. No intermediates are stored; all the data is final. In case of a crash or power failure, you can always return to the last well-known file system state.
Flash optimized	Paragon FS uses dynamic wear leveling and copy-on-write mechanisms that maximize your flash memory life. Paragon FS is designed to support unmanaged flash memory (NAND/NOR)
AUTOSAR® compliant	C++ code, C compatible interface
VFS emulation	Module to provide simple VFS integration
File Cache	Module to provide a file-caching layer
POSIX® compliance	Module to provide Paragon FS ease of integration with apps using POSIX®-like API
Block Cache	Module to provide Read-Write operations performance improvement by caching blocks of data
Symbolic and hard links	Support for symbolic and hard links
Customizable attributes	Provides an option to define an arbitrary set of extended file attributes — specifically for a target use case
Designed for flash	Set erase/program/read size-specific to flash memory media
Adjustable bitmap buffer size	Paragon FS is designed to operate with a user-defined bitmap buffer size if needed. This allows the developer to adjust bitmap buffer size to available memory constraints.
Static memory allocation	Paragon FS avoids dynamic memory allocation to run properly and stable in RTOS environments
RTOS optimized code	Paragon FS does not use recursion in its code, ensuring predictable code execution time, avoiding missed deadline issues in RTOSes
Compatible with low-resource environments	Paragon FS easily integrates into low-resource OS-like and preboot environments
Easy addition of AUTOSAR® compliant exFAT	Paragon FS integration enables easy one-step addition of AUTOSAR® compliant exFAT implementation. exFAT implementation is provided along with a direct license from Microsoft

AUTOSAR® is a trademark of AUTOSAR Gbr, München
POSIX® is a trademark of the IEEE.