
Rtfs

Installation Guide Read Me First

©2007 EBS, Inc
Revised June 2008



EBS Inc. 39 Court Street Groton MA 01450 USA
<http://www.ebseembeddedsoftware.com>

TABLE OF CONTENTS

Introduction _____	3
Installation Directory and Purchased Options _____	3
Documentation _____	4
Installing Rtfs _____	5
Building Rtfs _____	7

Introduction

Thank you for purchasing Rtfs. Rtfs has been deployed since 1987 and has been selected as the file system for thousands of designs over the years. It is a component in millions of units including scientific and medical devices, cell phones, digital cameras, and programmable set top boxes.

Please study the manual sections and call us at (978) 448 9162 or email us at support@ebseembeddedsoftware.com

Installation Directory and Purchased Options

The name of the installation directory depends on what options were purchased. The top directory is always Rtfs6.xx where .xx is the release ID. The distribution is contained in a subdirectory of Rtfs6.xx. The name of the subdirectory indicates what options were purchased.

The following configurations are available:

Configuration: (Install directory name)	Description
RtfsBasic	FAT12, FAT16, 8.3 File names only
RtfsPro	FAT32, VFAT, High Performance Free Cluster Management.
RtfsProFailsafe	RtfsPro features with Journaling
RtfsProPlus	High performance extended file API, Asynchronous operations, control of specific clusters or cluster ranges during file extends. Ddirect swap, extract and removal of sections of file cluster chains, cluster pre-allocation, zero fragmentation options, DMA features, file extent and free space map queries. Real time & deterministic operations.
RtfsProPlusFailsafe	RtfsProPlus features with Journaling
RtfsProPlus64	RtfsProPlus features but adds 64 bit metafile support. In RtfsProPlus, Extended file operations can only be performed on native 32 bit files. With RtfsProPlus64 these API calls also work on 64 bit metafiles.
RtfsProPlus64Failsafe	RtfsProPlus64 features with Journaling
RtfsProPlusDvr	RtfsProPlus64 features with additional API for implementing video recording and other similar devises.
RtfsProPlusDvrFailsafe	RtfsProPlusDvr features with Journaling

- All Rtfs distributions are similar.
- They all share the same manual set.
- They all share the same basic source code.
- As additional packages are added, the added source code files are placed mainly in new subdirectories. In some cases new source files are added to existing directories.
- All source distributions use the same configuration files.
- Each distribution contains a differently configured instance to the file named rtfspackages.h. This file contains a set of compile time flags that are used in Rtfsconf.h and a few run-time startup routines.
- Not all compile time options are available for all packages, but the manuals and source code indicate what configurations specific constants are available for.
- The project files for all releases are similar, differing only in the subdirectories and source files that are included.

Documentation

Before you start, please check the **Manual** directory for a read me file that may contain important release information.

Rtfs documentation is provided in both printed form and PDF. The PDF versions of all documents are supplied in the **Manual** subdirectory of the release. The following documents are provided.

Document	Description
Rtfs Configuration Guide	This document describes in detail the steps necessary to configure Rtfs features. The document describes both compile time and run-time configuration issues.
Rtfs API Reference Guide	This document provides programmer documentation for the Rtfs API.
Failsafe Technical Reference	Describes the theory of operation and technical information for Failsafe users
Device Driver and Porting Guide.	This document describes in detail the steps necessary to port Rtfs and device drivers to your target system and get it up and running.
Application Notes	Application notes explore certain topics that are not mentioned or briefly mentioned in the reference guides.
Revhist	Contains the current package revision number and history of changes and feature set enhancements by revision.
Read Me	A version specific read me file may be included.

Installing Rtfs

To install the software you must copy the directory named Rtfs from your CD to your workspace. The components included with the installation are described in the following table.

Component	Description
Manual (Included with all releases)	This subdirectory contains PDF versions of Rtfs documentation. It also contains some additional documents like read me files and revision histories that are not provided in print form.
Rtfsprojects (Included with all releases)	This subdirectory contains project files for building Rtfs libraries and test applications. The projects directory will always contain subdirectory for building Rtfs to run in Windows under simulation. Additional projects may also be included. Please inspect the directory.
include (Included with all releases)	<ul style="list-style-type: none"> This subdirectory contains common include files. You must add this directory to your include path when compiling Rtfs modules.
rtfscommon (Included with all releases)	<p>This subdirectory contains source and header files for the basic Rtfs package. It contains two subdirectories:</p> <ul style="list-style-type: none"> Source - Common source files. You must add all files in this directory to your project or make file when you build the Rtfs library. Apps – Application packages that use common source files. Use of these files is optional. This directory contains a useful portable shell program.
Rtfspro (Included if Pro or ProPlus options were purchased)	This subdirectory contains additional source files for the Rtfs pro package. The Rtfs Pro package adds FAT32 and VFAT support to the basic release.
Rtfspackages (Included if ProPlus options were purchased)	<p>This subdirectory contains additional, specialized source code and header files for Rtfs. It contains three subdirectories:</p> <ul style="list-style-type: none"> ProPlus - Additional source files. You must add all files in this directory to your project or make file when you build the Rtfs library.

	<ul style="list-style-type: none"> ▪ ProPlusDvr - Additional source files. You must add all files in this directory to your project or make file when you build the RtfS library. ▪ Apps – Application packages that use additional source files. Use of these files is optional. This directory contains several regression tests and an extension to the portable shell program that provides access to additional features.
Rtfsfailsafe (Included if Failsafe option was purchased)	<p>This subdirectory contains source code and header files for RtfS's Failsafe journaling package.</p> <p>If you plan to use Failsafe you must:</p> <ul style="list-style-type: none"> ▪ Add this directory to your include path when compiling RtfS modules. ▪ Add all files in this directory to your project or make file when you build the RtfS library.
Rtfstargets (Included with all releases)	<p>This subdirectory contains additional subdirectories which contain source code for porting files. Ports are provided for several hardware and software environments.</p> <ul style="list-style-type: none"> ▪ You must add a file named portkern.c and optionally portio.c to your project when building the RtfS library. ▪ If you are building for a supported target you may use files from the targets subdirectory. Otherwise you should make copies of files in the targets directory and modify them for your target.
Rtfsdrivers (Included with all releases)	<p>This subdirectory contains additional subdirectories which contain source code for device drivers.</p> <ul style="list-style-type: none"> ▪ You may add all files in all subdirectories to your project or make file. If you do this unwanted device drivers may be conditionally excluded by disabling them in portconf.h. ▪ Alternatively you may add files to your project or make file from subdirectories containing drivers you

	<ul style="list-style-type: none"> require. <ul style="list-style-type: none"> If you are providing your own device drivers there is no need to add any driver files to your project or make file.
Application Notes (Included with all releases)	Application notes explore certain topics that are not mentioned or briefly mentioned in the reference guides.

Building Rtfs

Follow these instructions to build a generic Rtfs library for your target. Once you have completed this section you can then reconfigure Rtfs and add physical device drivers by following the instructions provided in the Configuration Guide and Porting Guide.

- To compile the source code.
 - Add Rtfs source to your project or make file following the suggestions in the previous section.
 - Add Rtfs include paths to your project or make file following the suggestions in the previous section.
 - Edit **includes/rtfsarch.h** following the suggestions in the Configuration Guide.
 - Disable all device drivers by setting all possible options in **includes/portconf.h** and **includes/rtfsconf.h** to zero.
 - Edit **rtfscommon/source/apirun.c** and set **CALL_TEST_SHELL** to 0.
 - Add **portkern.c** to your project. The Device Driver and Porting Guide explains portkern.c in detail. A generic version of this file which compiles in any environment is provided in the file named **targets/template/portkern.c**.
- To compile the source code, link to your application and call Rtfs API functions, accessing the ram and rom disk drivers.
 - Implement the minimum kernel support functions required to run Rtfs according to the instructions in The Device Driver and Porting Guide.
 - Modify your application to call the routine **pc_ertfs_run**.
 - Modify **includes/portconf.h** to include these drivers and add the source code from **rtfsdriver/romdisk** and **rtfsdriver/ramdisk** to your project.
 - Note: by default the drive letters for these devices are G: and H: respectively.
- Now you may want to experiment with the Rtfs test shell. The test shell is very useful and an excellent debugging tool. If you do have access to a character console device or TCP-IP, it is strongly advised that you take advantage of the command shell features.
 - Follow these instructions to add shell support:
 - Implement telnet support or terminal IO support as explained in the Porting Guide
 - Add these four files to your project:
 - **rtfscommon/apps/appcmdshrd.c**

- **rtfscommon/apps/appcmdshwr.c**
- **rtfscommon/apps/apputil.c**
- **rtfscommon/apps/appcmdshformat.c**
- Edit the file **rtfscommon/apps/appcmdshrd.c** and set **INCLUDE_ESHELL** to 0.
- Edit **rtfscommon/source/apirun.c**. Set **CALL_TEST_SHELL** to 1.
- You should now be able to interactively experiment with your port from a console.