Small Satellite Research Laboratory

Franklin College of Arts and Sciences

UNIVERSITY OF GEORGIA

FileRecycler Component Documentation

Prepared for
The Small Satellite Research Laboratory
Fall 2024 - Spring 2025

Principal Investigator
Dr. Deepak Mishra

The University of Georgia
Geography-Geology Department
Athens, Georgia
United States of America

Prepared by
The MEMESat-1 CDH Team
University of Georgia
Athens, Georgia
United States of America

Contents

1 List of Acronyms

BBS Bulletin Board System
CDH Command and Data Handling
EPS Electrical Power System
GDS Ground Data System
FSM Finite State Machine
FSW Flight Software
MCU Microcontroller Unit
MS-1 MEMESat-1
OBC On-Board Computer
OSAL Operating System Abstraction Layer
OTA Over-The-Air
OTAU Over-The-Air Updater
TLM Telemetry
UART Universal Asynchronous Receiver-Transmitter
UHF Ultra High Frequency

2 Revision History

Changes	**Authors**	**Version**
Initial Draft	Aiden Hammond	1.0.0
Updated to fit new design	Aiden Hammond	1.0.1
Update format	Vansh Arora	1.0.2
[04-03-2024] Reviewed for Feasibility Review. Updated format, added block diagram and dependency section.	Olivia Beattie	1.1.2

3 Purpose

The document is meant as an overview for the FileRecycler component which is meant to delete old files on a schedule from the image and message storage, as well as allow the ground station to delete files, and allow other components to delete files from storage.

4 Overview

The component has access points at the ground station and through input ports. Through the ground station, files can be deleted before or after a specified time, or a specified number of files are deleted from oldest to newest. File information is accessed through the stat method, and they are evaluated based on modification time (which can be changed by mknod(2), truncate(2), utime(2), and write(2)). Modification time was chosen because the time of last access and the time of last status change may not give accurate information about how old the file actually is. For example, if the file is read(2) then the time of last access changed, or if the inode information is changed (i.e., by changing owner, group, link, count, mode, etc.) then the time of last status is changed. Also through the ground station, the schedule aspect can be toggled or changed.

Description of Directory Files:

- FileRecycler.fpp - contains fprime interactions for the component.
- FileRecycler.hpp - Used as a container for function signatures.
- FileRecycler.cpp - Implementation of the fprime interactions.

5 Interactions

5.1 Components

5.1.1 Scheduler

The scheduler component is responsible for running the file recycler on a specified schedule. The file recycler contains the schedule at which is should be run, and the scheduler will call the file recycler at that cycle rate. The routine of the file recycler will then be initiated each time the scheduler calls it. The file recycler will then delete files based on the schedule that is set.

5.1.2 BBSMessageStorage

The BBSMessageStorage (or Message Storage) component communicates with the file recycler to dispatch specific file-system locations which should be cleaned by the file recycler.

5.1.3 OTAUpdater

The OTAUpdater should communicate with the file recycler through a port instantiation to ensure that any files which need to be cleaned should be handled accordingly. The OTAUpdater will be able to call the file recycler to delete files which are no longer needed, or to delete files which are not needed for a specific OTA routine.

5.1.4 StateMachine

The StateMachine component will communicate with the file recycler to ensure that when a system reset or shutdown is initiated, the file recycler will be able to prepare for it. Due to the nature of the file recycler, it will need to be able to handle a system reset or shutdown in a way that does not cause any issues with the file system.

6 Design

6.1 Requirements

Requirement	Description	Verification Method
FR-010	Delete files before or after a specified time.	Unit Test
FR-011	Delete specific number of files from oldest to newest.	Unit Test
FR-012	Execute functions on a specified schedule.	Unit Test

6.2 Ports

Through the input ports, other components can request (on a queue) file deletion from the image storage. Another input port is used for receiving status information about a schedule change request. The last input port interacts with a 0.25 Hz (1/4) rate group to clear messages stored on the queue for the component. Three output ports interact with the Scheduler.

Port Data Type	Name	Direction	Kind	Usage
FileRecyclerOpIn	fileRecyclerOpIn	Input	Asynchronous	Allows other components to delete files from the storage directory.
FileRecyclerOpOut	toFileRecyclerHandler	Output	N/A	Internal port for calling fileRecyclerOpInHandler (Primarily for scheduledHandler)
SchedulerModule SendSchedule	sendSchedule	Output	N/A	Sends the name and schedule to the Scheduler Component.
SchedulerModule RunSchedule	scheduledHandler	Output	Synchronous	Called by the Scheduler component and handles schedule status. Recycles files.
readyForPowerOff	FileRecyclerStatusPort	Input	Synchronous	Tells the component to lock directory access and be ready to power off.

6.3 Custom Ports

FileRecycler defined three port types. Two input ports, and one output port. FileRecyclerOpPort and MemeRecyclerOpPort are both ports that can be used by other components to perform certain operations (i.e., delete last x files). The FileRecyclerStatusPort can then report back to the requesting component whether the operation was successful or not.

Name	Parameter and Type	Description
FileRecyclerOp	count: U32 op: FileRecyclerOp	Determines which files are deleted. What method of deletion is used.
FileRecyclerStatus	result: bool	Whether the operation was completed successfully or not.
MemeRecyclerOp	op: CronOp	

6.4 Custom Types

These custom types are defined to be used within the FileRecycler's ports. The FileRecycler and CronOp types, at a high level, allow components to customize how they want files to be deleted. The various operations are listed below.

Name	Type	Argument/Parameter	Description
FileRecyclerOp	enum	BEFORETIME	Deletes all files created before specified time.
		AFTERTIME	Deletes all files created after specified time.
		NUM	Deletes specified number of images in order of age, beginning with oldest.
		MAX	Defines the maximum number of images to be kept in storage.
CronOp	enum	START	Start the cron job.
	enum	STOP	Stop the cron job.
Status	enum	COMPLETED	Status completed.
		FAILED	Status failed.
FauxOsStatus	Struct	status string status_message: string	

6.5 Commands

CLEAN_UP_DATA specifically deleted the files based on the instructions (i.e., which operation to perform). This is the crux of what this component does.

Mnemonic	Arguments	Synchronization	Description
CLEAN_UP_DATA	count: U32 op: FileRecyclerOp	Synchronous	Initiates the file. Deletion action.

6.6 Parameters

The parameters are high level arguments that determine how the component behaves at an abstracted level. MAX_FILES determines the maximum amount of files to store. SCHEDULE is the deletion schedule upon which the component operates.

Name	Data Type	Default Value	Description
MAX_FILES	U32	1000	Max files that can be stored.
SCHEDULE	String	”0 * * * * ?”	The deletion schedule.
SYS_RESET	U8	0	System reset.

6.7 Events

The events are fired when a command is successful, an operation was performed, or an error was ecountered. these help when debugging and testing the component.

Name	Severity	Description
COMMAND_RECV	command	An event that is triggered when a command is received.
CLEAN_UP_DATA_COMPLETED	diagnostic	the deletion is complete.
SCHEDULE_ERROR	warning low	There was an error with the scheduling.
SCHEDULE_RAN	diagnostic	The schedule has run.
SCHEDULE_CHANGED_TO	diagnostic	The schedule was changed.
FILE_REMOVE	diagnostic	The file was removed.
CANNOT_OPEN_DIR	warning high	Cannot open a directory.
RECYCLER_STOPPED_BY_SYS_RESET	warning low	Recycler was stopped by being set for system reset.
STAT_ERROR	warning low	Stat object returned non-zero value.
UNEXPECTED_DIRECTORY_STATUS	warning low	There was an unexpected directory status.

6.8 Telemetry

The telemetry values relay data they may need to be extracted from the component to evaluate its performance during run-time.

Name	Data Type	update	Description
MAX_FILES	U32	On change	Max number of files.
NUM_FILES_REMOVED	U32	On send	Number of files removed.

6.9 Dependencies

The FileRecycler component relies on Scheduler components. The Scheduler will receive schedules and send notifications to the FileRecycler, to which it will execute operations. Thus, the FileRecycler relies on some port types defined by the Scheduler.