



libtrf

API Release Notes v1.4.0

Apr 26, 2023

Table of Contents

1 Introduction	2
2 New Features	2
3 Enhancements	3
4 Fixed Defects	3
5 Known Issues and Limitations	4
Contact Us	4

1 Introduction

thinkRF's **libtrf** API is a multi-platform API library that can be used for interfacing with existing generation thinkRF RTSA devices (R5xx0) and will be maintained into the future to support interfacing with future thinkRF devices. In addition, the API supports and will be evolved to increasingly provide sophisticated signal processing capabilities that can be easily integrated into end-user application code.

The libtrf library is currently released as builds for Microsoft Windows 64-bit (.h/.dll/.lib) and Linux (.h/.so) for both x86_64 (Intel/AMD) and aarch64 (ARM v8+) targets. The interface is presented as undecorated 'C' compatible with the C99 (ISO/IEC 9899:1999) and later standards.

The original libtrf public release was version 1.1.1. This document describes the updates to libtrf leading to the current 1.4.0 release. Refer to the companion 'libtrf API Reference Guide' document for full libtrf and its usage details.

1.1 Supported devices

The versions of libtrf currently released support all thinkRF's RTSA products R55x0/R57x0 products (408/418/427 models).

2 New Features

The following new features were added between the 1.1.0 and the 1.4.0 releases:

1. Implemented a user-accessible processor abstraction. A processor provides a means of processing and analyzing radio data within libtrf before presentation to user code. The following processors are currently implemented:
 - a. AM demodulator (DSB, LSB, USB)
 - b. FMDemodulator
 - c. IQ to Spectrum converter
 - d. SpectrumCharacterization (add Peak-hold, Average, Min-hold additional spectra)
2. Added finite-duration spectrum capture with time-duration or frame-count specification.
3. Added many new examples and with richer command-line options.
4. Added the following new functions:
 - a. Stream related: trfGetNextBaseband(), trfFreeBaseband(), trfAllowDiscarding().
 - b. File related: trfAttachStreamToFile() , trfConvertBasebandFileToWavFile()
 - c. Processor related functions as mentioned in point 1. above.
5. Added the following new parameters:
 - a. RTSA info related: TRFGNSSValid, TRFRxSampleRate.
 - b. File related: TRFLooping.
 - c. Capture Frames Related: TRFDurationFrames, TRFFramesExpected.

- d. AM/FM Demodulation parameters: TRFAMSubtype, TRFFramesProduced, TRFOutputSampleRate.
 - e. IQ to Spectrum Processor parameters: TRFFollowIQ, TRFOverlap, TRFMatchIQ, TRFOutputSize, TRFRecentFrames.
 - f. Spectrum Characterization Processor parameter: TRFAverageCount, TRFClearAverageTrace, TRFClearMaxHoldTrace, TRFClearMinHoldTrace.
6. Added the following new trfStatus values: trfNotABasebandStream, trfFileTypesNotBaseband, trfBasebandStreamStartFailure, trfSampleRateInvalid, trfCannotObtainOutputStream, trfCannotConnectOutputStream, trfBadFilename, trfWindowTypeInvalid, trfDeviceCommunicationsLost, trfUnknownStreamType.
 7. Added the following new trfIQFrame struct members: uSequenceNumber, pJSONInfo.
 8. Added the following new trfSpectrumFrame struct members: uSequenceNumber, pJSONInfo, uAdditionalSpectra, pAdditionalSpectra. See the “Streams” section of the Reference Manual in particular regarding the new stream buffer handling with these new members.
 9. Introduced a ‘discarding’ flag on streams to facilitate efficient operation when reading data from files for post-processing. See documentation for trfAllowDiscarding().
 10. Added the following new trfBasebandFrame struct members: uSequenceNumber, pJSONInfo.
 11. Added FMinHz and FMaxHz to parameters for a spectrum stream reading from a file.

3 Enhancements

The following other changes were made between the 1.1.0 and the 1.4.0 releases:

1. Updated definition of many parameters and functions. Refer to the Reference Guide or libtrf.h for the details.
 - a. Parameters: TRFBufferSec, TRFBufferFrames, TRFDiscardProportion, TRFIFBWHz, bDiscontinuity, iDurationNanosec
 - b. trfCreateIQStream()
2. Renamed these parameters or functions:
 - a. trfInvalidDeviceHandle to trfDeviceHandleIsInvalid
 - b. trfDisposeJSON() to trfDisposeString()
3. Removed these parameters or functions:
 - a. In trfBasebandFrame struct members: uID, bDiscontinuity
 - b. trfInitAPI() and trfShutdownAPI() - not required as .dll/.so load and unload perform the required operations.
4. trfGetVersion() returns ‘char’ instead of ‘const char’.
5. Updated the Reference Manual with a new architecture diagram and much added details.
6. Significant improvement to the overall code stability and robustness, especially with IQ/spectrum frame handling, finite/stream capture, JSON handling, file related handling, and improved memory efficiency and garbage collection.

7. Improved RBW and sweep frequencies calculation improved on-the-fly parameter change handling.
8. Enhanced frame capture usage with added sequence number and JSON info tied to the frame to allow processors to communicate with user-code frame-synchronously.
9. Improved existing examples and comments.
10. Improved UDP discovery handling.

4 Fixed Defects

The following issue(s) are fixed between versions 1.1.1 and 1.4.0:

1. Fixed a linux socket issue that could lead to SIGSEGV.
2. Fixed spectral inversion flag setting when handling I14 data conversion to complex which led to occasional spectral distortion.
3. Fixed spectrum power calibration to improve absolute accuracy.
4. Fixed spectrum with occasional segments not properly aligned.
5. Fixed finite capture still attached when trfExhausted reported.
6. Fixed incorrect spectrum data with 20MHz span.
7. Fixed an exhausted stream attached to a receiver would not report 'detached'.
8. Fixed buffer handling that could result in bloated/runaway buffer consumption in large span, narrow RBW sweep captures.
9. Fixed reference source (TRFPllSource) handling.
10. Fixed trfExamineDevices() handling of invalid IP address.
11. Fixed trfGetParameters() handling of nullptr for pWindowType.
12. Fixed stream reading from a file still indicating 'attached' after completion.

5 Known Issues and Limitations

The following are the known limitations or other issues present in version libtrf version 1.4.0:

1. R55xx/R57xx - Between 50MHz and 30MHz centre frequencies, it is not possible to stream a full 40MHz IFBW as IQ data.
2. Current implementation does not support below 100MHz of IFBW.
3. In the current implementation IQ data with IFBW below 1MHz and FCentre below 50MHz exhibit sinc-type spectral distortion.

6 Contact Us

ThinkRF Support website provides online documents for resolving technical issues with ThinkRF products at <https://thinkrf.com/resources>.

For all customers who hold a valid end-user license, ThinkRF provides technical assistance 9 AM to 5 PM Eastern Time, Monday to Friday. Contact us at <https://support.thinkrf.com/> or by calling **+1.613.369.5104**.