

DOCUMENT NUMBER:	90000002
DOCUMENT REVISION:	A
DATE:	22/07/2025

Customer: INTERNAL

RADAR DATA READER

SOFTWARE USER MANUAL



REVISION RECORD



TABLE OF CONTENTS

1 Introduction: A Complete Workflow.....	4
2 Initial Setup (Mandatory Step).....	4
3 Phase 1 & 2: Flight Analysis and Segment Export (Tab 1. Flight Analyzer).....	4
4 Phase 3: Data Extraction from Segments (Tab 2. Segment Processor).....	5
5 Single .out File Analysis (Tab 3. Single OUT Processor).....	6
6 Creating and Managing Export Profiles.....	6
7 Manual Conversion (Tab 4. REC to OUT Converter).....	6
8 Troubleshooting and Frequently Asked Questions (FAQ).....	6

1 INTRODUCTION: A COMPLETE WORKFLOW

Welcome to the Radar Data Reader & Processor. This tool is designed to guide you through a complete workflow for radar data analysis, transforming raw flight recordings into structured, analysis-ready data.

The application is organized to follow a logical, three-phase path:

Phase 1: Flight Analysis: Start with an entire folder of .rec files to automatically identify and summarize the flight's significant operational segments.

Phase 2: Segment Export: Once the segments of interest are identified, this phase extracts them, creating raw data files (.out, video, etc.) for each one.

Phase 3: Data Processing: The final step is to batch-process the exported segments to extract the desired numerical values into standard formats (CSV, JSON) using custom profiles.

This manual will guide you through each phase.

2 INITIAL SETUP (MANDATORY STEP)

Before using the application, it is **critical** to specify the location of the g_reconvert.exe executable, which the application depends on.

Launch the application.

Navigate to the “4. REC to OUT Converter” tab.

Click the “g_reconverter Advanced Config...” button.

In the window that opens, the first field is “Executable Path”. Click “Browse...” and select the exact location of your g_reconvert.exe file.

Click “Save & Close”. The configuration will be saved for future use.

Without this step, the main features of the application will not work.

Guide to the Main Workflow

This section describes the most common and powerful use case for the application.

3 PHASE 1 & 2: FLIGHT ANALYSIS AND SEGMENT EXPORT (TAB 1. FLIGHT ANALYZER)

This is your command center for starting a new analysis. Here, you will identify and export the key moments of the flight.

Goal: To turn a folder full of .rec files into a list of operational segments and generate the raw data for the ones you are interested in.

Steps:

1. **Select Flight Folder:** Click “Browse...” next to “Recordings Folder” and select the folder containing the complete sequence of .rec files for the flight.
 2. **Name the Flight:** Enter a unique identifier for the flight in the “Flight Name” field (e.g., FlightTest_May25). This name will be used to create a dedicated workspace folder (flight_workspace/FlightTest_May25) where all results will be saved.
 3. **Start Preliminary Analysis:** Click “Start Flight Analysis”. The application will run g_reconvert.exe in the background to analyze all files, generate a text report, and create a full-mission video. This process may take several minutes.
 4. **View Results:** Upon completion, the “Flight Summary & Segments” table will be populated with a summary of all operational segments found, showing modes, durations, and reference files.
 5. **Export Segments of Interest:**
 - * Select one or more segments from the table that you wish to analyze in detail.
 - * Click “Export Selected Segment(s)”.
-

* The application will create a subfolder for each selected segment within the flight's workspace. Each subfolder will contain an .out file with the raw data for that segment, along with other ancillary files (segment video, GPS track, etc.).

Outcome of this phase: You now have a well-organized workspace folder with a flight summary and subfolders containing raw data ready for extraction.

4 PHASE 3: DATA EXTRACTION FROM SEGMENTS (TAB 2. SEGMENT PROCESSOR)

Now that you have your exported segments, it's time to extract the numerical data you need.

Goal: To batch-process the .out files from the exported segments to generate CSV or JSON files based on custom profiles.

Steps:

1. **Load Exported Segments:** Click "**Load Exported Segments from Current Flight**". The table will be populated with the segments found in the current flight's workspace.

* Segments ready for processing will have a "**Ready**" status (in green).

* Segments you saw in the previous tab but have not yet exported will have a "**Not Exported**" status (in gray) and will not be selectable.

2. **Configure the Output:**

* **Output Directory:** Choose a **dedicated** destination folder for your final data files (e.g., C:/Final_Analysis/FlightTest_May25).

* **Formats and Profiles:** Enable CSV and/or JSON export by checking the corresponding boxes and select an **Export Profile** for each. (See Section 6 for creating profiles).

* **Folder Structure:** Check "**Create a separate folder for each segment's analysis**" if you want each analysis file (e.g., segment_A.csv) to be saved in its own subfolder. Leave it unchecked to have all analysis files in the same directory.

3. **Select and Start:**

* Select the "Ready" segments you want to process from the table.

* Click "**Process Selected Segments**".

4. **View the Results:** The application will process each segment, generating the CSV/JSON files in the output directory you specified. You can monitor the progress from the progress bar and the console.

Outcome of this phase: You have obtained clean, structured data files, ready to be imported into MATLAB, Python, Excel, or other analysis tools.

Supporting and Advanced Features

5 SINGLE .OUT FILE ANALYSIS (TAB 3. SINGLE OUT PROCESSOR)

Use this tab when you have a single .out file you want to analyze quickly, outside of a full flight analysis workflow. The functionality is identical to Phase 3 but applied to a single file.

6 CREATING AND MANAGING EXPORT PROFILES

The Profile Editor (File -> Manage Export Profiles...) is the heart of data extraction customization.

Left Panel (Profiles): Create new profiles (e.g., "Thermal_Analysis", "Tracking_Performance") or delete old ones.

Center Panel (Available Fields): Browse the tree of all data structures and fields the application can read. Fields marked with (Enum) can be translated into human-readable text. Select a field and click >> to add it to the current profile.

Right Panel (Selected Fields for Profile):

View and reorder the fields of the selected profile using the **Up/Down** buttons.

“Translate” Column: This is a powerful feature. By clicking the checkbox in this column, you can decide whether an (Enum) field should be exported as its raw numerical value (e.g., 15) or its text representation (e.g., STT_MASTER_MODE), making your CSV files instantly readable.

7 MANUAL CONVERSION (TAB 4. REC TO OUT CONVERTER)

This tab provides a simple frontend for `g_reconvert.exe`, useful for quick conversions or for testing specific parameter combinations without going through the entire analysis workflow.

8 TROUBLESHOOTING AND FREQUENTLY ASKED QUESTIONS (FAQ)**Q: The application starts, but no functions seem to work. What should I do?**

A: You have almost certainly not set the path to `g_reconvert.exe`. Carefully follow the steps in the “Initial Setup” section.

Q: Why does exporting a segment seem to fail or crash?

A: `g_reconvert.exe` can be sensitive to its input parameters. The current export logic relies on specifying the number of .rec files to process. If the problem persists, ensure the .rec files are not corrupted.

Q: When I start an analysis, a second window of my application opens. Is this normal?

A: This is a known behavior when running an application packaged with PyInstaller. While the current code should prevent this issue using `multiprocessing.freeze_support()`, if it occurs, it indicates that the new process was not handled correctly.

Q: Can I add a data field that I don't see in the “Available Fields” tree?

A: Yes, but it requires modifying the source code. You will need to define the new C-style data structure in the `radar_data_reader/core/structures` package and teach the `struct_parser.py` how to read it from the binary file.