DART User Manual – Zips Racing

Author: Cole Barach

Date Created: 2026.02.17

Date Updated: 2026.02.22

The DART (or Data Acquisition and Racing Telemetry System) is Zips Racing's custom data acquisition system and vehicle dashboard. This user manual details the operation and maintenance of the DART, specifically its data logging capabilities. For details about the BMS page of the DART, see the *BMS User Manual*.

General Tips

- To prevent the DART from running out of storage, it is recommended that **all data logs are deleted before a testing day begins**.
- To prevent multiple logging sessions taken in the same testing day from being confused with one another, it is recommended that **no data logs are deleted during a test day**. The DART will not run out of storage.
- To prevent important data logs from being deleted, it is recommended that data logs are **retrieved as soon as possible after a testing day ends**.
- To prevent local data logs from being overwritten, it is recommended that data logs are **copied into OneDrive immediately after being copied locally**. If this is not possible, the directory should be renamed to something unique (**especially if the DART's data logs have been deleted**).
- To make future data analysis easier, it is recommended that **all notes taken during a test day have the current session number written down with them**. This number is reported on any of the DART's driving pages.

Data Logging Basics

All data logs the DART generates are divided into "sessions". Each session begins when the DART powers on and ends when it powers off. During this time, every single CAN message received by the device is written into the log. If the data log exceeds a certain size, it will be "split" into multiple smaller files. A single logging session may generate over 10 split files. Split files are stored chronologically; split 0 is always the first, followed by split 1, and so on. There should be no discontinuity between the data in split files.

Digitally, each logging session is stored in a unique directory (`session_0`, `session_1`, etc.). Inside of this directory are each of the log's split files. When powered online, the DART will determine the current session number based on the highest existing session number. For instance, if the device has a session called `session_11`, the DART will choose `session_12` as the next session number. If no logging sessions are present on the device, the DART will start from `session_0`.

Accessing Data Logs

To retrieve one or more data logs from the DART, the `dart-cli` application (downloaded from the ZRE-CAN-Tools project, https://github.com/ZipsRacingElectric/ZRE-CAN-Tools) may be used.

In order to communicate with the DART, your PC must be connected to the DART's ethernet connector. Locate the correct cable and connect your PC before starting the `dart-cli`. Your PC may take a minute or two to recognize the DART. If the `dart-cli` says it failed to connect to the DART, close the application and try again.

Left: The DART's ethernet connector. Typically installed on the dashboard of the vehicle.
Right: An Ethernet-to-ASL cable. The right side connects to the vehicle, the left to your PC.

After connecting successfully, the `dart-cli` should give an output similar to below.

```
[cole@arch zre_cantools]$ dart-cli

Testing Connection... (Ctrl+C to Cancel)
Connected.

Enter an option:
 l - List all remote log files
 c - Copy all logs locally
 x - Delete all remote log files
 s - Open an interactive SSH connection to the DART
 j - Print the DART's system journal
 t - Test connection to the DART
 q - Quit
```

Listing Data Logs

To get a list of all the data logs on the device, or to check the amount of remaining storage, the “l” option can be used.

```
Enter an option:
 l - List all remote log files
 c - Copy all logs locally
 x - Delete all remote log files
 s - Open an interactive SSH connection to the DART
 j - Print the DART's system journal
 t - Test connection to the DART
 q - Quit
l

Remote Logs:

session_0
session_1

Filesystem     1K-blocks    Used Available Use% Mounted on
/dev/mmcblk0p2   3382356 1152728   2037488  37% /
```

The “Remote Logs” section lists all logs present on the device, including the log that is currently being written to.

Underneath, the “Filesystem” section lists the total size of the system’s storage, the amount of used storage, and the amount of available storage. Note these quantities are all in terms of 1024 byte blocks.

The “Use%” column indicates the percentage of the storage being used. Note that this quantity refers to the amount of space being used by both data logs and the operating system itself.

Copying Data Logs

To retrieve data logs, the “c” option can be used. This command will copy all of the DART’s data logs into a local directory.

```
Enter an option:
 l - List all remote log files
 c - Copy all logs locally
 x - Delete all remote log files
 s - Open an interactive SSH connection to the DART
 j - Print the DART's system journal
 t - Test connection to the DART
 q - Quit
c

Copying Logs to '/home/cole/zre//dart_2026.02.17'...

split_0.mf4
split_0.mf4

Done.
```

The destination directory is located in the user’s `Documents` directory, in a directory based on the current date. For example:

```
%USER_PROFILE%/Documents/ZRE/dart_2026.02.17/
```

It is important to note that if data logs are copied multiple times in the same day, the destination directory will be the same and the newer data logs will overwrite the older data logs. For this reason, it is recommended that data logs are not deleted in the middle of a testing day.

The DBC file (or files) associated the set of data logs is also copied over. These DBC files must be kept with the data logs, as they are the only way to decode them later (more on this in the *Analyzing Data Logs* section).

Deleting Data Logs

To prevent the DART from running out of storage, data logs must be deleted periodically. To do this, the “x” option can be used.

When entered, the CLI will ask you to confirm whether you would like to delete all the data logs. **This operation cannot be undone, so exercise caution**. Enter “y” to confirm the operation. Once this is done, all the previous logs (including the log currently being written into) will be deleted. The DART will reset and begin logging at `session_0` again.

When the DART runs out of available storage, the behavior is undefined. The current logging session is likely to be corrupted, and no future logging sessions will be saved. Because of this, it is important to delete data logs before any important testing.

Analyzing Data Logs

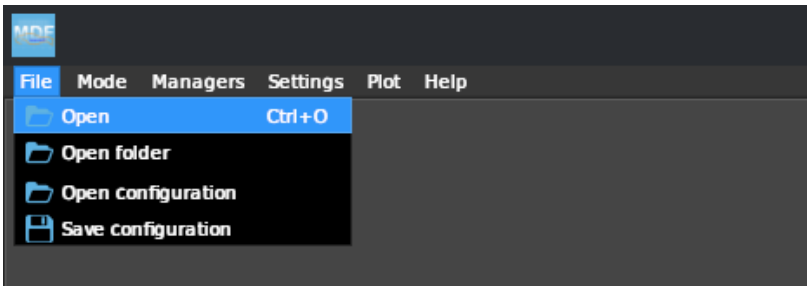
Once one or more data logs have been retrieved from the DART, they can be analyzed with the ASAMMDF-GUI software (downloaded from https://canlogger.csselectronics.com/canedge-getting-started/ce2/log-file-tools/asammdf-gui/). Note, this software doesn't actually have an installer, just extract the archive and copy it somewhere permanent (you can create a start menu or taskbar shortcut for convenience).

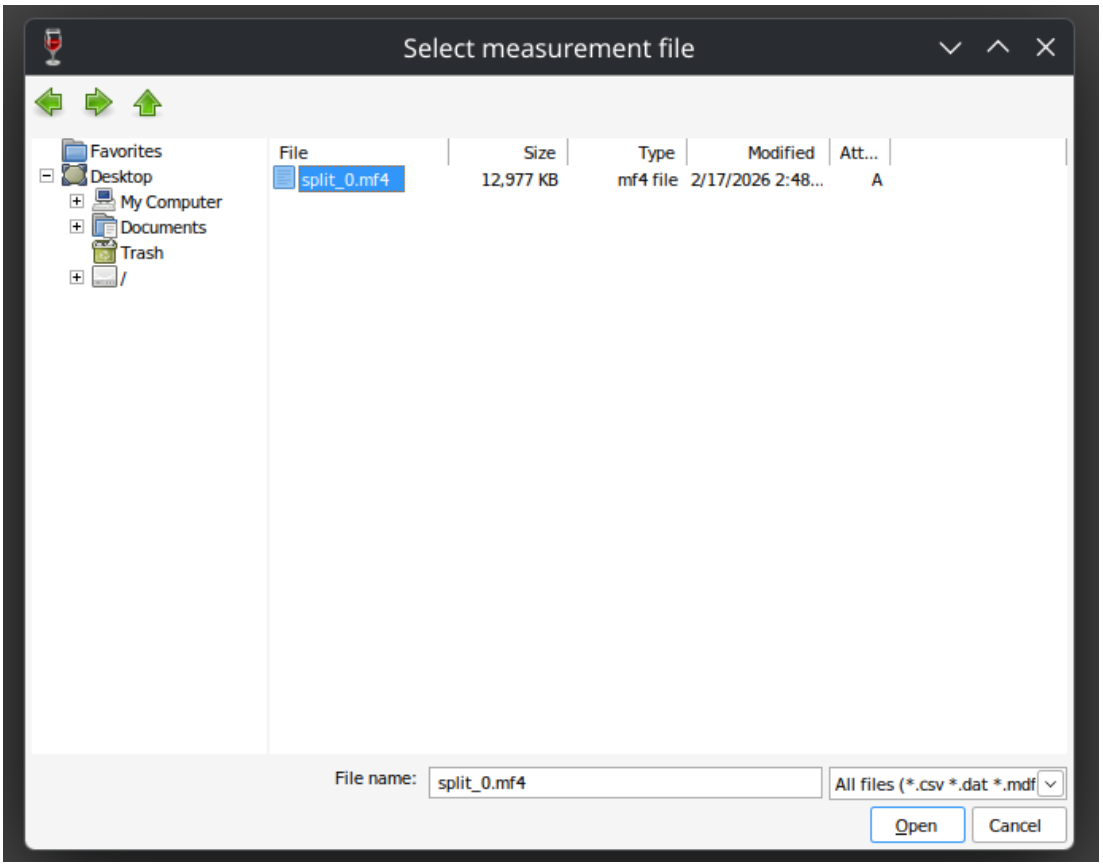
In order to analyze a data log, it must first be decoded. The "key" to decoding a raw data log is the vehicle's CAN DBC files, which "tell" analysis software how to interpret the raw CAN information. This interpretation is dependent on the current version of the vehicle's hardware & firmware, meaning not all data logs can be interpreted with the same version of a DBC file. Whenever a group of data logs are retrieved from the DART, the associated version of the vehicle's DBC files are also retrieved. These are the DBC files that must be used to interpret said data logs, not any other version.

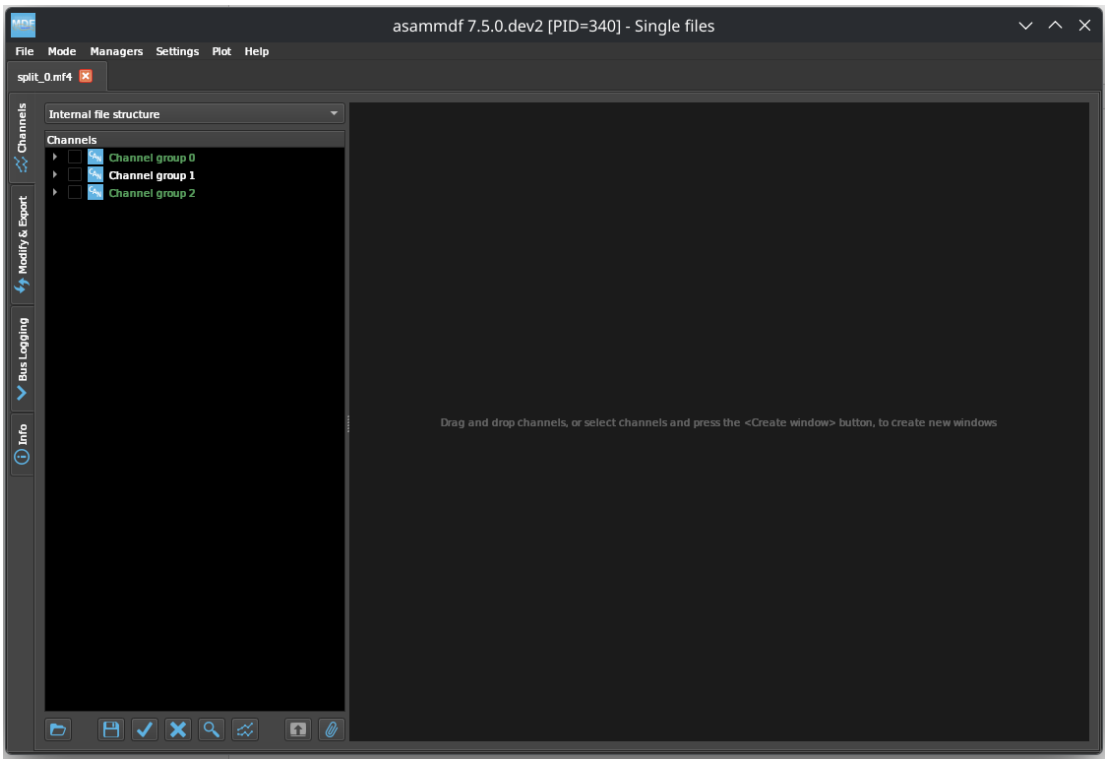
Decoding a Single Split File

For a log file that has no splits, that is, the session directory only contains a single file, the decoding process is relatively straightforward.

1. Open the log file in ASAMMDF-GUI (make sure you are in Single Files mode)

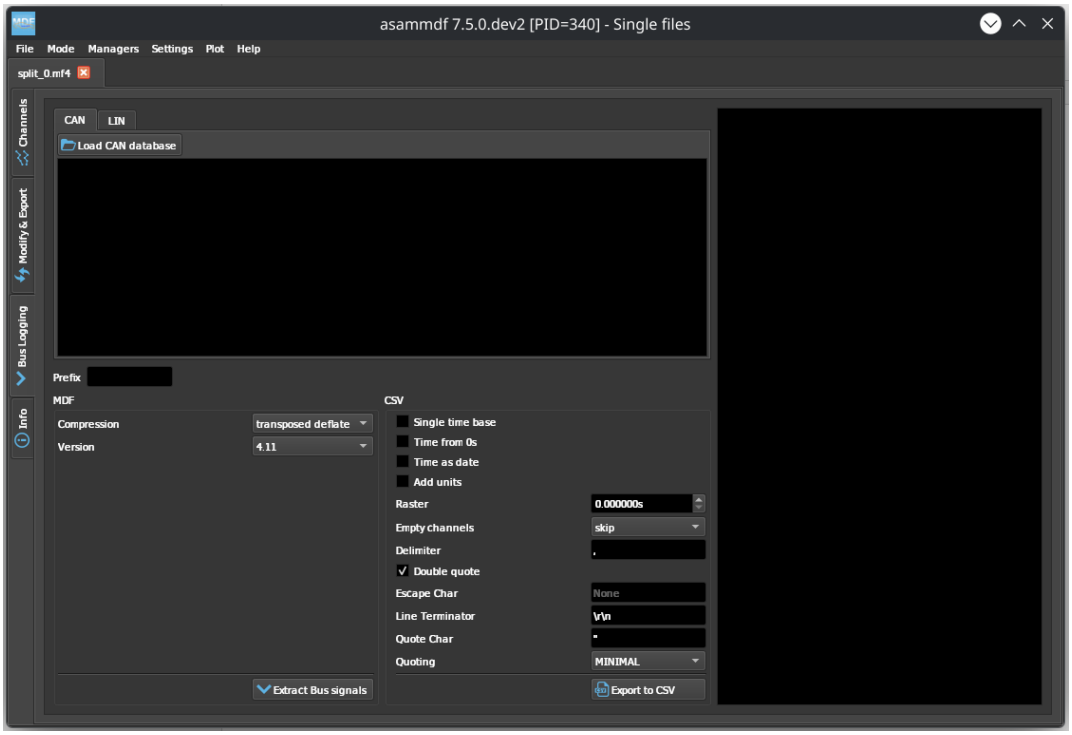




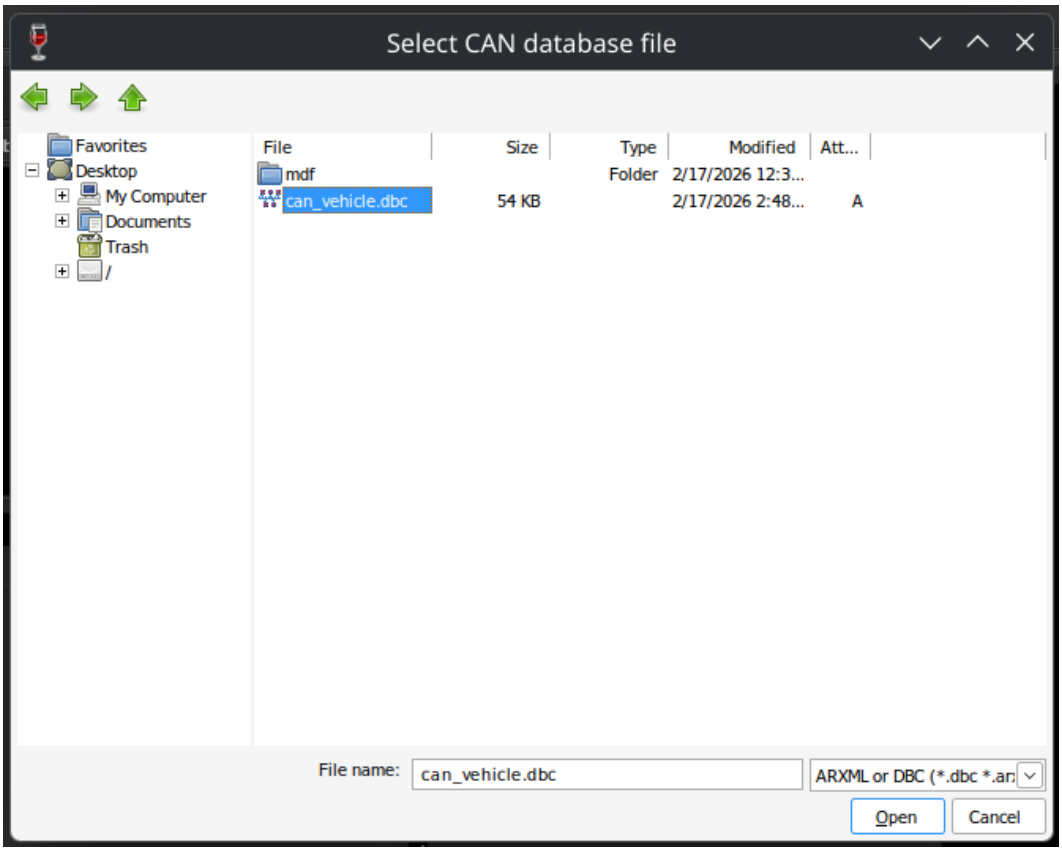


After loading, the GUI should look something like this.

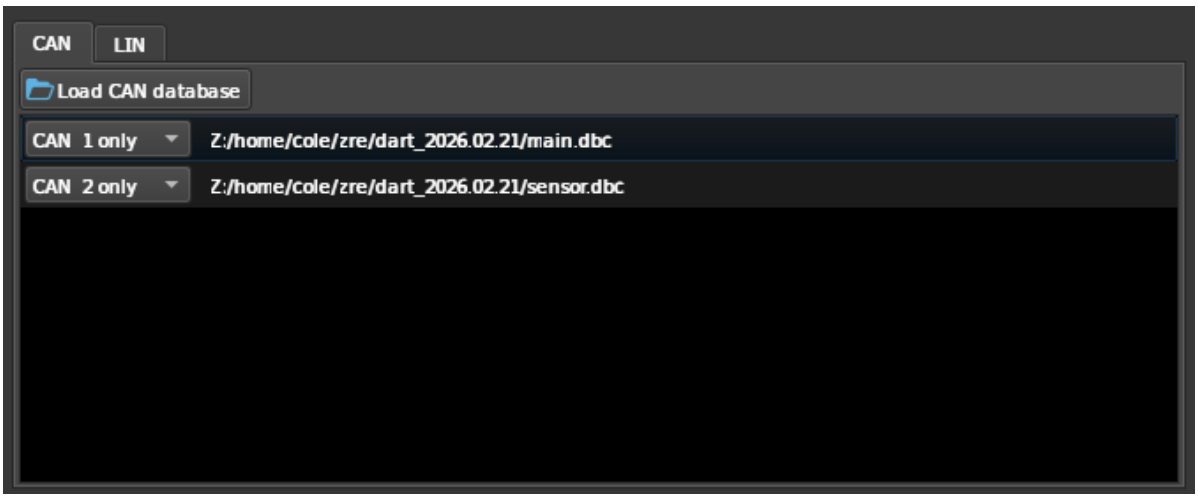
2. Navigate to the “Bus Logging” tab.



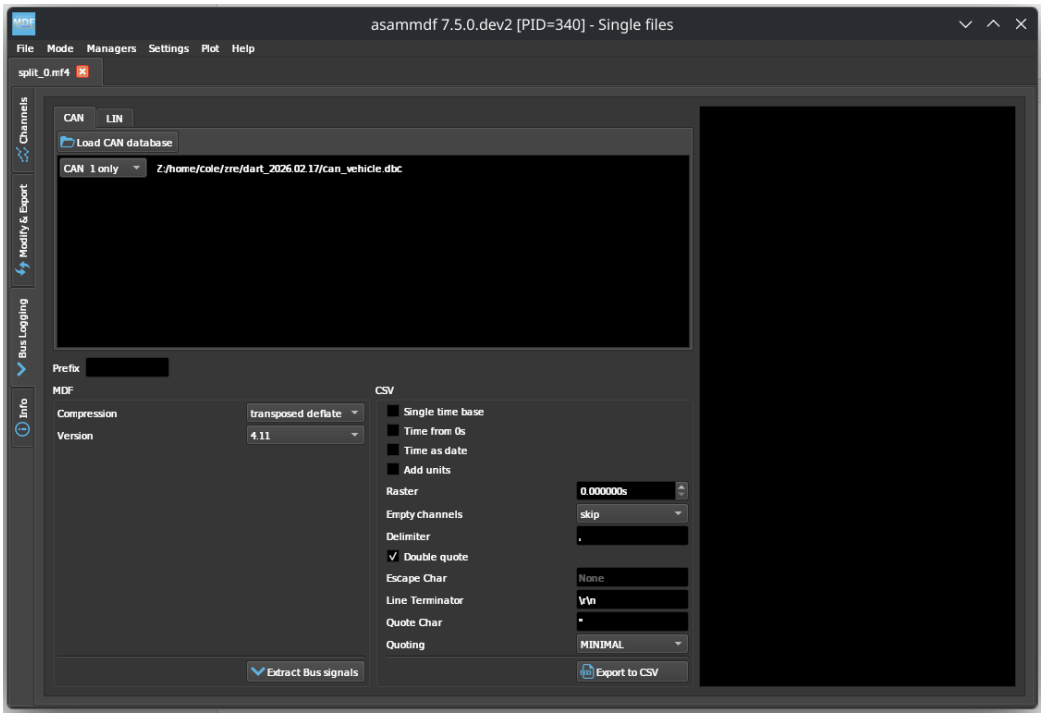
3. Click “Load CAN database” to and select the data log’s DBC file(s)

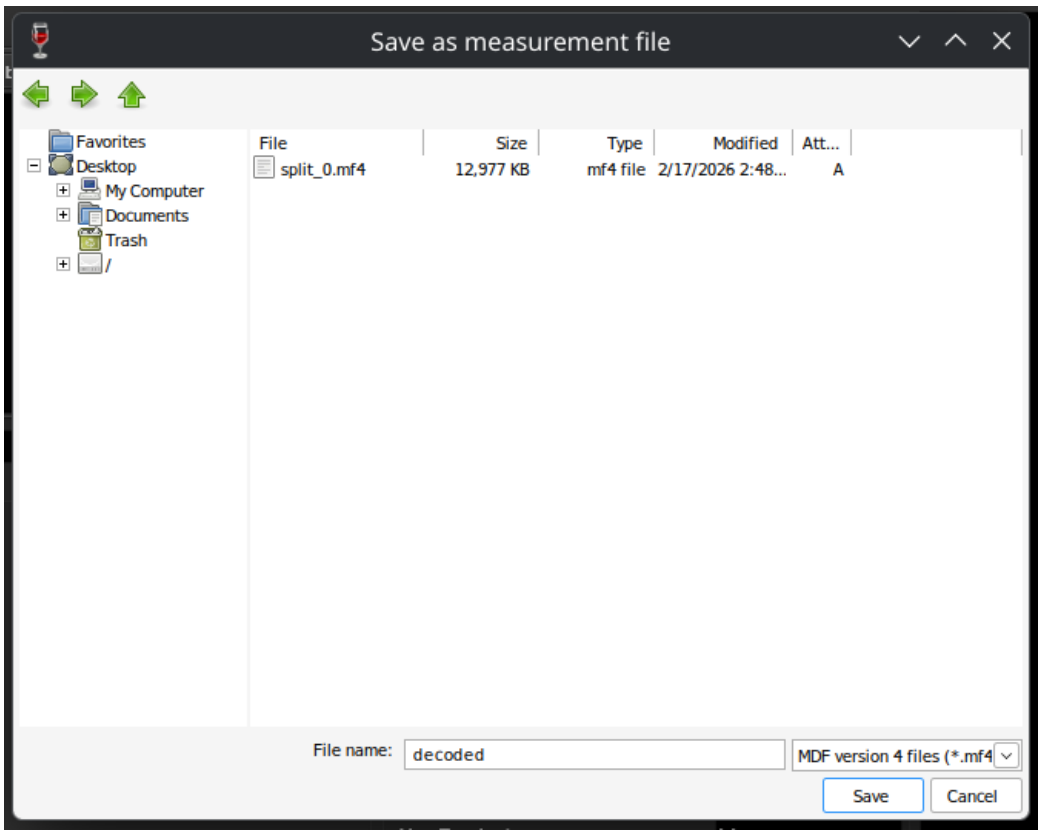


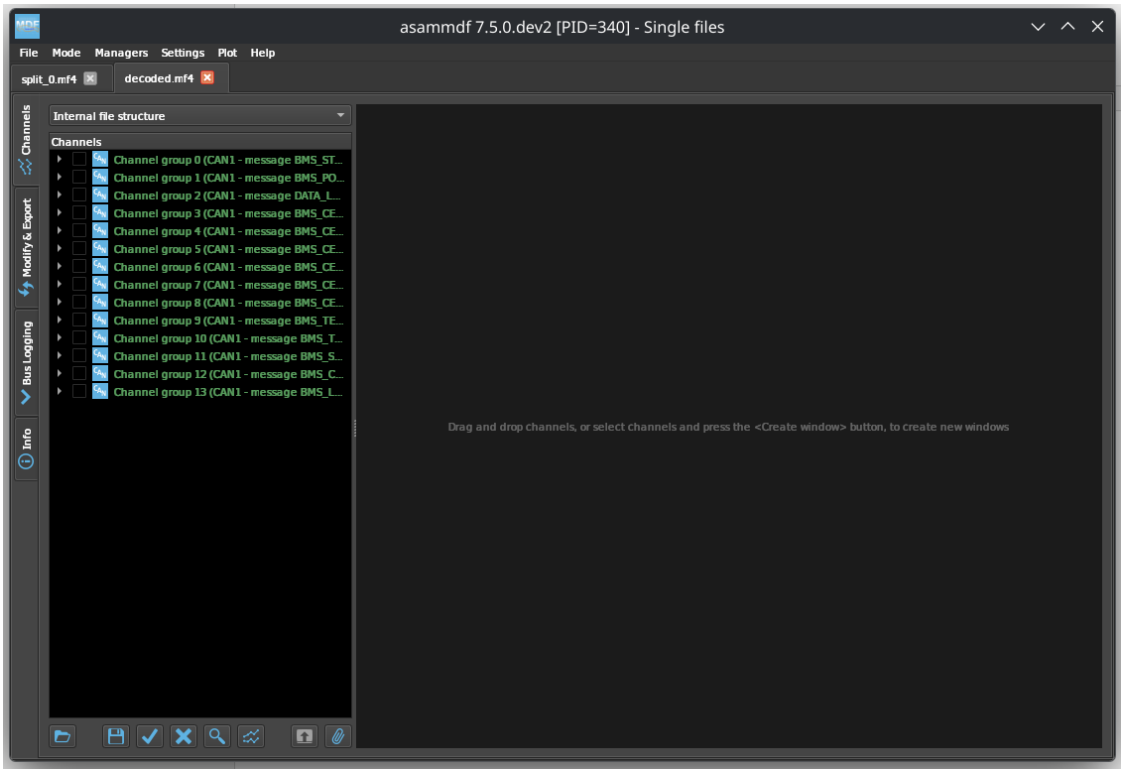
4. Ensure the correct channels are mapped to the correct DBC files. Typically, “main.dbc” is “CAN 1 only” and “sensor.dbc” is “CAN 2 only”. **Make sure these are the DBC files in the same directory as the session directory is.**



5. Click “Extract Bus signals” in the lower right corner and save the file as “decoded.mf4”. Make sure this is saved within the session’s directory.





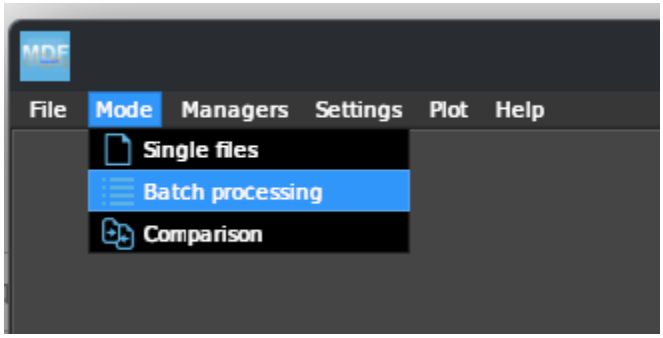


After saving, the decoded log file should open. The GUI should look something like this.

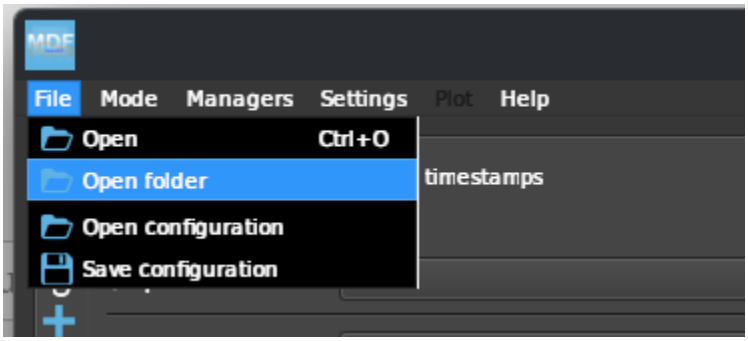
Decoding a Group of Split Files

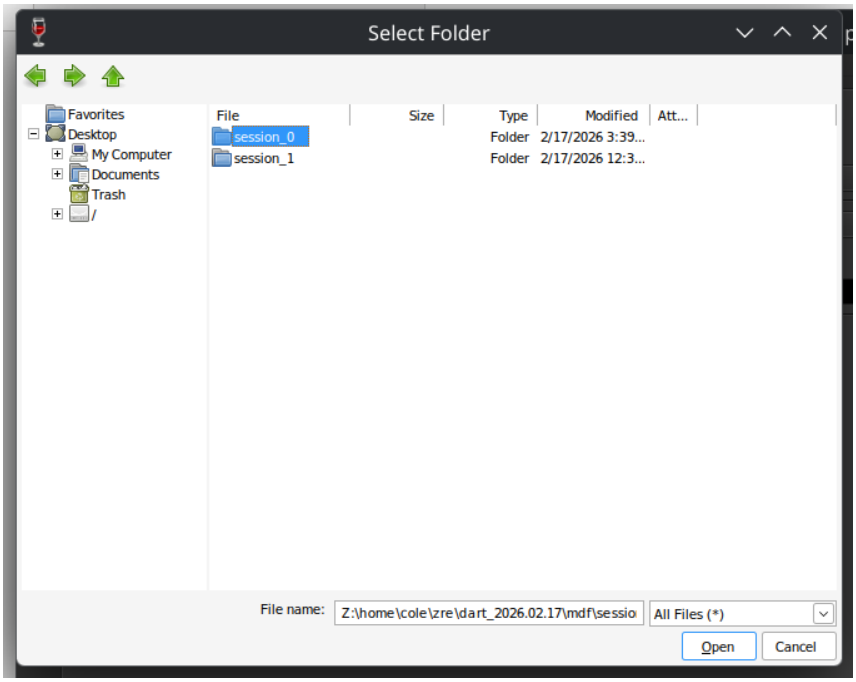
For a data log that is split into two or more split files, the split files must first be concatenated before they can be decoded.

1. Put ASAMMDF-GUI into “Batch Processing” mode.

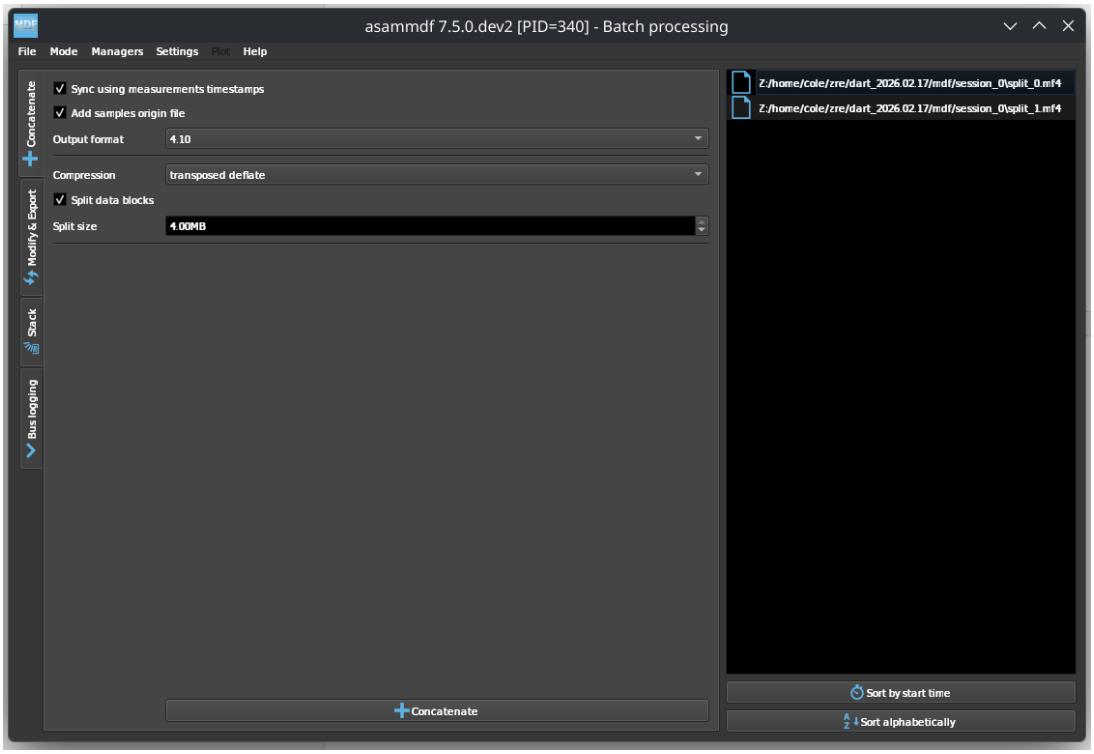


2. Select “Open Folder” and open the session’s directory.

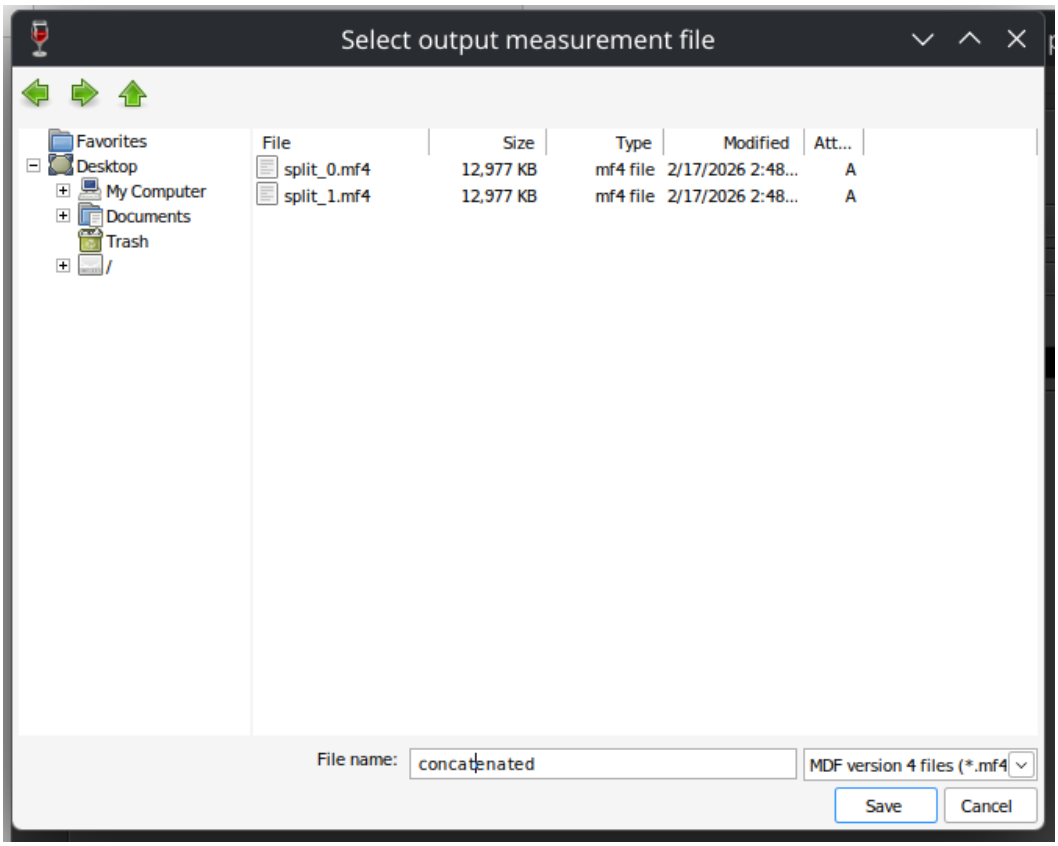




3. Click “Concatenate” in the lower section of the GUI.



4. Save the concatenated file as “concatenated.mf4”. Make sure this is saved within the session’s directory.



5. From this point, the concatenated log file may be opened and decoded as if it were a single split file (see *Decoding a Single Split File*). Make sure to set the GUI back to “Single Files” mode.