DAQ HDF5 File Format

Adam Greig

adam@ael.co.uk

This document specifies version 2 of the Airborne DAQ HDF5 file format.

1 Overview of HDF5

HDF5 is a standardised file format for storage and organisation of datasets.

HDF5 files are comprised of objects which may be datasets or groups. Datasets contain n-dimensional homogeneous data arrays, while groups contain named links to either datasets or other groups. Objects may have many attributes, which are arbitrary key-value pairs. Each HDF5 file contains a single root group, and all other objects may be accessed by a path from this root group. Objects may exist at multiple paths without duplication of the underlying data. The format also provides support for compressing and checksumming datasets.

For further information on HDF5 refer to its website: http://hdfgroup.org/.

2 Overview of the DAQ system

During a single test run, the DAQ (data acquisition) system records data from many DAUs (data acquisition units) which each comprise many channels, which are connected to transducers in the test system, such as pressure transducers and thermocouples. Multiple channels share a common sampling timebase, and at each sampling instant, the value for that channel is recorded as raw data. This raw data is later postprocessed using configuration and calibration data to produce readings in engineering units, which are used for further analysis. The final output from a test run is a time series for each channel, consisting of the timestamps at which samples were taken, and the corresponding values in converted and calibrated engineering units. Additionally the raw channel data is available in case re-conversion is required in the future.

DAQ channels are organised into logical groups, for example all the thermocouples in the system may belong to the "temperatures" group, while all transducers on an injector may belong to the "injector" group. Channels may belong to multiple groups.

© Airborne Engineering Limited 2019 Confidential - Proprietary Information

3 Objectives

The design of this format has the following objectives:

- Archivability
  - Standard and documented format is easier to read in the future
  - Compression and integrity checking make long term storage of many datasets practical
  - Keeping all data and metadata together in a single file prevents fragmentation and improves traceability
- Usability
  - Common dataset format allows easier analysis tool reuse
  - Documented format provides easier sharing of datasets with external users
- Extensibility
  - Support future use cases in a backwards-compatible fashion

4 DAQ File Format

4.1 Version

This document specifies version 2 of the file format. Changes to file structure or changes to mandatory attributes will use higher version numbers. See the Revision History at the end of this document for changes from previous versions.

4.2 Nomenclature

For clarity, HDF5's concepts of a group is hereafter referred to as an "HDF5 group", while a group of DAQ channels is referred to as a "channel group". The HDF5 group which represents a single DAQ channel is referred to as a "Channel HDF5 Group", and the HDF5 group which represents a channel group is referred to as a "Group HDF5 Group".

4.3 File Names

Files are typically named in the following format: `YYYYMMDD-NNN-OOO.h5`, where `YYYMMDD` is the current date, `NNN` is an incrementing number representing the test number that day, and `OOO` is the output type, such as `raw` or other output type name.

This name format is not controlled by this document and may vary.

4.4 File Attributes

Each file contains the following attributes. All strings are stored with datatype `H5T_STRING` with UTF-8 encoding and variable length.

- `version`: Version of this file format, stored as an integer `H5T_STD_I64LE`. Currently 2.

© Airborne Engineering Limited 2019 Confidential - Proprietary Information

- `name`: Test run name. Identical to filename without output name or suffix. Typically in the format `YYYYMMDD-NNN`.
- `output`: Output name. Typically the `OOO` part of the filename.
- `file_datetime`: The time this file was created in UTC using the following ISO8601 format: `YYYY-MM-DDTHH:MM:SS.SSSSSSZ`, for example `2018-03-14T10:29:55.427732Z`.
- `start_datetime`: The test start time in UTC using the same format as `file_datetime`.
- `t0_datetime`: The T0 time instant in UTC using the same format as `file_datetime`. T0 is the instant from which channel timestamps are computed. The event that determines T0 is test-specific, for example a valve opening or a sequence starting.
- `end_datetime`: The test end time in UTC using the same format as `file_datetime`.
- `location`: Location of the test.
- `hostname`: The hostname of the system running the DAQ software during the test.
- `operator`: Test operator name (aka author or firing officer).
- `summary`: Description of the test run and data.
- `project`: Name of the project encompassing this test. May be blank.
- `daq_git_commit`: The full Git commit ID of the DAQ software which created this file.

Optional file attributes:

- `subconfig`: Name of the subconfiguration in use, if any.
- `atmospheric_pressure`: Ambient pressure at time of test, stored as a double precision floating point value in bar. For some test runs this may instead be present as a full data channel, potentially at low sample rate.
- `analysis_metadata`: JSON-encoded dictionary of metadata for the AEL data analysis software, such as averaging windows and x limits.
- `project_metadata`: JSON-encoded dictionary of metadata relating to this project. Typically used for further project-specific analysis.
- `changelog`: Free-form text description of any changes made to this data file compared to its as-recorded state, for example recalibrating a transducer.

4.5 File Structure

At the top level, each file contains HDF5 groups named `channels`, `groups`, `sequences`, `config`, and `raw_data`.

The `channels` top-level HDF5 group contains one child channel HDF5 group for every channel in the system. See the "Channel HDF5 Groups" section below for details on these groups.

The `groups` top-level HDF5 group contains one child HDF5 group for every channel group in the system. See the "Group HDF5 Groups" section below.

The `sequences` top-level HDF5 group contains one child dataset for every sequence file used in this test run. See the "Sequences" section below.

The `config` top-level HDF5 group contains child datasets with configuration files used to generate this data file. See the "Config" section below.

The `raw_data` top-level HDF5 group contains one child dataset for every DAU recorded by the system. See the "Raw Data Datasets" section below.

© Airborne Engineering Limited 2019 Confidential - Proprietary Information

4.6 Channel HDF5 Groups

Each channel in the DAQ system is represented by an HDF5 group. The HDF5 group is named according to the channel ID, and contains two child datasets, named `time` and `data`. The `time` dataset contains the timestamps in seconds before/after the T0 instant which correspond to the readings in the `data` dataset, and is typically a hard link to a shared `time` dataset used by many channels. Not all channels will necessarily share the same timestamps if they were sampled at different frequencies on the same timebase, and not all channels will share a timebase depending on the source of the channel. The `data` dataset contains the processed channel readings in engineering units. Both `time` and `data` are one dimensional, and have the same size in that dimension. Both datasets are stored chunked and using gzip compresssion and Fletcher32 checksums. The `time` dataset has the 64-bit floating point type `H5T_IEEE_F64LE`, and while the `data` dataset typically has the same type, its type may vary for certain channels.

The channel HDF5 group has the following attributes:

- `name`: Human readable name for the channel.
- `latex_name`: Optional LaTeX-formatted name for the channel. Include explicit `$` symbols as appropriate for maths mode.
- `units`: The units associated with this channel.
- `colour`: Optional colour specification for displaying this channel. Either X11 color names or 6-digit hex colours starting with a `#` character are supported.
- `format`: Optional format specifier for numerical display of this channel. Uses C format specifiers starting with a `%` character.

4.7 Group HDF5 Groups

Each channel group in the DAQ configuration is represented by an HDF5 group. The HDF5 group is named according to the group ID, and contain multiple child channel HDF5 groups by symbolic links to HDF5 groups in `/channels`.

The group HDF5 group has the following attributes:

- `name`: Human readable name for the group.
- `latex_name`: Optional LaTeX-formatted name for the group.

4.8 Sequences

Multiple text files may be used to set up a test run, defining timing sequences for control devices in the DAQ system. These files are stored in the `sequences` top-level HDF5 group, as plain text arrays. The dataset name is the same as the filename of the original file, and the dataset data matches the file contents. The contents of the sequence files is not controlled by this document.

4.9 Config

Any files used to configure the DAQ system for this test run are stored in the `config` top-level HDF5 group. The DAQ system is typically configured with two such files, `assets.yaml` and `config.yaml`.

For each file in the `config` HDF5 group, the following attributes must be present:

© Airborne Engineering Limited 2019 Confidential - Proprietary Information

- `path`: The original path to the configuration file.
- `sha256`: A SHA256 checksum of the file contents.
- `git_commit`: If applicable, the full Git commit ID of the version of the file in use. If not applicable, this attribute is an empty string.

Files are stored as fixed length strings with datatype `H5F_STRING` and UTF-8 encoding. The format of the contents of configuration files is not controlled by this document.

4.10 Raw Data Datasets

Some HDF5 files will contain the raw DAQ sample data. Typically these files do not also contain processed data, so the `channels` HDF5 group will be empty.

Each DAU is represented by a single dataset which contains a one dimensional array of datatype `H5T_OPAQUE`, with gzip compression and Fletcher32 checksums. Each element contains one frame of raw data, with a fixed length per DAU. The format of the raw data is not controlled by this document.

© Airborne Engineering Limited 2019 Confidential - Proprietary Information

4.11 Example Structure

```
/
|-- channels/
|   |-- p_inj/
|   |   |-- data
|   |   |-- time
|   |-- t_inj/
|       |-- data
|       |-- time
|-- groups/
|   |-- injector/
|   |   |-- p_inj -> /channels/p_inj/
|   |   |-- t_inj -> /channels/t_inj/
|   |-- pressures/
|   |   |-- p_inj -> /channels/p_inj/
|   |-- temperatures/
|       |-- t_inj -> /channels/t_inj/
|-- sequences/
|   |-- seq1003.txt
|-- raw_data/
|   |-- 1001
|   |-- 1002
|-- config/
|   |-- assets.yaml
|   |-- config.yaml
|-- changelog
```

© Airborne Engineering Limited 2019 Confidential - Proprietary Information

4.12 Revision History

4.12.1 Issue 1: File Format 1: 2018-04-10

- Original release.

4.12.2 Issue 2: File Format 2: 2018-11-01

- Version number increased to 2.
- File name description changed to accommodate an output type, which may be `raw` for datasets containing raw data, or otherwise indicates the output type.
- Added `file_datetime` top-level attribute to record the time the HDF5 file was created.
- Removed all `config_file_*` and `assets_file_*` top-level attributes. The same data now exists as attributes on the `config.yaml` and `assets.yaml` files in the `config` top-level HDF5 group.
- Added new `subconfig` and `changelog` optional top-level attributes.
- Added new `config` top-level HDF5 group.
- Moved top-level `config.yaml` and `assets.yaml` datasets into new `config` top-level HDF5 group.

© Airborne Engineering Limited 2019 Confidential - Proprietary Information

Notes:

airborne engineering

© Airborne Engineering Limited 2019.
Registered in the UK as company number 4205661.
Registered Office: Central Office, Cobweb Buildings, The Lane, Lyford, Wantage OX12 0EE, U.K.