

Table of Contents

1	Information on this document.....	2
1.1	Content	2
1.2	About changing the content.....	2
1.3	Disclaimer	2
2	Conanair automatic mode component and operation features	2
2.1	Component	2
2.2	Operation features	3
2.3	Restrictions	4
2.4	conanair identification	4
3	File-Pass labels	5
4	File Settings.....	5
4.1	Common rules about the description of the setting file	5
4.2	Reflection of Settings	6
4.3	Connection key files.....	6
4.4	ID/Tag association file.....	6
4.5	Measurement Schedule Setting File.....	7
4.5.1	Repetition	8
4.5.2	Handling of schedules with short time intervals.....	8
4.5.3	Constrains on Default settings	8
4.5.4	Schedule management exceptions	9
4.6	APP_ROOT Configuration file	9
5	Measurement result file.....	11
5.1	Communication channels for measurement results	11
5.2	Folder Tree	11
5.3	Data format	11
5.3.1	Summary information.....	12
5.3.2	Real data	14
	History of revisions.....	15

1 Information on this document

1.1 Content

This document contains information that is essential to know when creating an application that uses conanair operated by an automatic system. The majority of this document is about data communication, its applications, and conanair. Finally, it includes necessary background information to help users better understand and grasp what they are reading.

1.2 About changing the content

conanair has been in the process of developing and improving characteristics of their products. This document is a snapshot of the latest features they have created.

Every feature mentioned in this document may or may not be present at the time of release. After the first release, features and characteristics may undergo further changes due to updates.

1.3 Disclaimer

As mentioned in the previous section, the content mentioned here might change before the release of conanair's product. If a decision must be made based on the content of this document, please contact us to make sure there are not any changes. If you decide based on this document without any inquiries or confirmation of any changes, and when any damages are happened because of its changes, we don't take any responsibilities about it. Also, because of our business judgement, we might suspend the part of the product's function or the entire product or the release. In the case of happening the damage because of suspending the release, we don't take any responsibilities except a particular contract.

2 conanair automatic mode component and operation features

2.1 Components

conanair automatic mode consists of the following components.

- conanair
 - It's a Wi-Fi monitoring vibration sensor.
 - A form ware to support certain automatic mode needs to be equipped.
- Base station
 - It's an application group for receiving a data from conanair and conducting an operation instruction.

- It can run on a dedicated computer, or a server that already purposed for another task.
- It can be operated on Windows or Linux operating system.
- It must run continuously all the time.
- It must run in an environment that can adjust the time accordingly (「conanair network」 reference)
- conanair Network
 - Wi-Fi network is needed to connect to conanair.
 - You can connect multiple conanair to the same network (coverage of directed_broadcast_packet). In most cases, the maximum number of conanair are decided by network limitations.
 - The base station should join to Wi-Fi network or connect to using a transmissible LAN cable.
 - It is possible to have multiple base stations connected to the same network, but usually configured such that there is only one base station communicating at a given point in time, such as active/standby.
 - By setting up a mechanism for multiple base stations to share data, a configuration that allows multiple base stations to communicate at a given point in time can be achieved.
 - ✧ This is a logical possibility, no concrete design etc. is prepared.
 - Sharing a network for other purposes and constructing an exclusive network are both possible.
 - ✧ It's determined according to required security level, etc.
 - It is easiest to build an isolated, dedicated network for high security, but this requires special measures for base station time synchronization.
 - ✧ An example of the measure is to run the base station on a PC with two network interfaces, one connected to conanair network and the other to a network that can communicate with a NTP server.
 - ✧ The other example is to equip the computer on which the base station operates with an intrinsically accurate clock (Time Source), such as a radio clock, but this is generally expensive.

2.2 Operation features

To achieve long-term battery operation, conanair spends most of the time in a deep sleep state, returns at a predetermined time, performs measurements, and transmits the results to the base station.

- During Deep-Sleep, any other operations except forced reset or reset by battery are inactive.
- Deep-Sleep is only available around an hour at a time, the device will repeatedly return to the Deep-Sleep state until the specified time is reached and then immediately transition to the Deep-Sleep state.
- Because of the above operation, even if you reset forcibly, it's activating Deep-Sleep mode right away. To interrupt Deep-Sleep state is only possible by removing the battery in person.
- The timer that controls automatic return from Deep-Sleep has an error of up to several percent of the time.
 - Most of this is an individual difference, which can be compensated for by calibration process.
 - After correcting for individual differences, the remaining error is expected to be up to 0.5%.
 - ✧ A measurement frequency should be determined so deviations in measurement time due to errors are within an acceptable range.
 - When transmitting the results, the next measurement timing is revised by a base station clock, so errors do not accumulate.

2.3 Restrictions

Due to the characteristics described in the previous section, changes to the measurement schedule by remote control are not reflected until measurements are taken according to the previously changed schedule. To reflect the measurement schedule right away, you need to perform a reset physically, such as removing and installing the batteries.

2.4 Conanair identification

conanair identifies each hardware with a unique ID and a Tag name that can be set voluntarily.

- It is determined when manufacturing ID hardware. It cannot be changed.
- Tag name can be set voluntarily by the user.
 - It is not possible to set the same Tag name for more than one conanair at a time.
 - If you replace the conanair hardware, you can retain your previous Tag name.
- One of the configuration files defines the correspondence between ID and Tag name.
 - It is the user's responsibility to maintain the correspondence between the proper ID and Tag name.
- Settings and data are identified by Tag name, unless they can't be identified without ID.

3 About file-pass label

From now on, file-pass will be denoted as follows.

- Use / (slash) as pass separator
- The top pass in the tree for a conanair running on a computer and base station is represented as "CAIR_ROOT".
 - The specific location is the CAIR_ROOT subfolder in the folder where you installed the base station application.
- The ID of conanair is expressed as "CAIR_ID"
- The Tag name of conanair is represented as "CAIR_TAG"
- The top pass in the tree about conanair on the computer where the application is running is represented as "APP_ROOT".
 - The specific location is described in one of the base station configuration files.
- Distinguishing or not distinguishing capital letters or small letters of file-pass are depended on OS.
 - Naming is based on the premise of no distinction and determined to avoid duplication.
 - Distinguished OS must be as described in this document to work properly.

4 Setting file

4.1 Common rules about the description of the setting file

- Record: One line is one record. A record cannot span multiple lines.
- Encoding should be UTF-8, with or without BOM.
- Newline characters: CR (0x0d), LF (0x0a), or CRLF (0x0d 0x0a)
- Invalid line: If any line is present that cannot be interpreted as a valid setting, all settings in the file are treated as errors.
- Blank lines: It doesn't affect the settings. Any voluntary number of blank lines can describe voluntary points.
- Delimiter: comma
- Leading and trailing whitespace: It ignores leading, trailing spaces and tabs in each field after being separated by a comma.
- Comments: The # sign on each line, and anything to the right of it, is treated as a comment.
- Extra fields: If more than the specified number of fields are specified, the extra fields are treated as comments.

4.2 Reflection of Settings

If you change any settings, the change is not immediately reflected.

- Reflection by schedules: It reads the contents of the current configuration file every 15 and 45 minutes.
 - The contents of files that are exclusively opened by other processes, or files that have errors won't be reflected.
 - If any of the above events occur, a message will be displayed or recorded in a log.
- Reflection by restart: When to immediately reflect the new settings, the process of base station needs will restart.
 - If the restart takes a long time (more than 10 seconds), the results sent by conanair may be missed.
 - Restarting the base station process will fail if there is even one unreadable or error-ridden file.

4.3 Connection key file

- Explanation
 - This means handling conanair by applications is not a principal.
 - The key file for the base station is used for mutual authentication when conanair and the base station first start communicating.
 - It is created during the initial setup operation by conanair and is copied to a designated location by the user.
 - ✧ At this time, the conanair side key is created and stored in non-volatile memory.
 - It is possible to re-create it and invalidate the previously created key file.
 - Without a valid key file, the conanair and the base station cannot communicate.
- Configuration file name and location: CAIR_ROOT/connection keys/CAIR_ID.key
- Description rule: The description rule is not disclosed because the file is automatically generated.
- Sample of contents: No sample is available because the file is automatically generated.

4.4 ID/Tag association file

- Explanation
 - This file is intended to be edited by the user using a text editor etc., but the application that uses conanair can also provide an editing function.
 - If there is a valid key file, conanair and the base station can communicate, but if there is no valid description, the received data will not be saved.
- Configuration file name and location: CAIR_ROOT/tags.conf
- Description rules
 - Field 1: Tag name

- Field 2: ID
- Duplicate records:
 - If the same Tag name or ID appears more than twice, an error will occur.
- Sample contents:

```
# Sample tags file
P-123, cna_fa3c5e
```

4.5 Measurement Schedule Setting File

- Explanation
 - This is intended to be edited by the user with a text editor or a similar program, but the application that uses conanair can also provide editing functions.
- Configuration file name and location: CAIR_ROOT/schedule.conf
- Description rule
 - Field 1: Tag name
 - Field 2: Schedule type (sample reference)
 - Field 3~12: Schedule (sample reference)
 - The time zone for the time and date will follow the settings of the computer on which the base station is running.
- Duplicate record:
 - If the same Tag name appears more than twice, the last record will be valid.
- Order of fields
 - Fields do not have to be in time order.
- Multiple fields are the same value.
 - The field with that value is considered to have been written only once.
- Sample content:

```
# Sample schedule file
*, daily, 09:30, 10:30, 11:30, 12:30, 13:30, 14:30, 15:30, 16:30, 17:30, 18:30
P-123, daily, 09:30, 13:30, , , , , , ,
P-456, weekly, Mon 09:30, Wed 09:30, Fri 09:30, , , , , ,
C-567, monthly, Mon1 09:30, Mon2 09:30, Mon3 09:30, Mon4 09:30, , , , ,
C-657, monthly, 05 09:30, 15 09:30, 25 09:30, , , , , ,
P-234, daily, 09:00,+01:30*5, , , , , , ,
```

- Remarks:
 - ✧ Tag name "*" is the default setting (used when there is no individual setting by Tag name).
 - ✧ Time is based on 24-hour system.
 - ✧ "Mon1 09:30" is 9:30 on the first Monday.

- ✧ "05 09:30" is 9:30 on the 5th.
- ✧ xth day format and day format can be mixed (deprecated)
- ✧ "hh:mm, +hh:mm*n" is a repeat specification (see below for details)

4.5.1 Repetition

- Explanation
 - Instead of listing specific times, "start time, + time interval * number of repetitions" can be specified.
 - However, if the start time has already passed, the measurement will be performed at the next timing that matches the "start time, + time interval * number of repetitions".
 - Due to the restrictions described in the next section, "Handling of schedules with short time intervals," the time interval must be a multiple of 30 minutes.
 - If the schedule type is "weekly", the start time must include the day of the week and the time interval must include the number of days. Note that the repeat specification cannot be used "monthly".
 - If the schedule type is "daily", the schedule will not repeat after 23:59 and will follow the next day's schedule.
 - Similarly, if the schedule type is "weekly", it will not repeat after 23:59 on Sunday.
 - You cannot mix this with the method that lists specific times.
- About wildcards for the number of repetitions
 - Instead of a concrete number, the wildcard "?" can be specified.
 - If "?" is specified as the number of repetitions, it will continue to repeat within the range of schedule type.

4.5.2 Handling of schedules with short time intervals

- Explanation
 - Conanair does not assume repeated measurements at short time intervals.
 - Specifically, it is assumed that there will be an interval of at least 30 minutes by the next measurement.
 - If the set schedule is not zero or 30 minutes every hour, the entire line is ignored as an invalid setting.
 - However, it is possible for a timer error to result in an actual measurement interval of less than 30 minutes.

4.5.3 Constrains on Default settings

- Background
 - Default settings may be applied to many sensors.

- In this case, if no measures are taken, many sensors will be communicating at the same time, which may not be handled completely.
- Limitations
 - The default setting limits the specification of time and time intervals to multiples of 30 minutes.
 - That is, only 00 or 30 can be specified, and any others will be an error.

4.5.4 Schedule management exceptions

- All sensors are assigned an offset time that is a multiple of ± 10 seconds, depending on the order in which they are listed in the ID/Tag association file.
 - The offset of ID/Tag described at the beginning is zero.
 - The second offset is +10 seconds.
 - The third offset is -10 seconds.
 - The fourth offset is +20 seconds...
- The nth offset [second] follows the below formula.
 - If n is odd: $-10 \times (n - 1)/2$
 - If n is even: $+10 \times n/2$
- Regardless of the individual setting or default setting, it is controlled to be shifted by the offset time assigned to each sensor by the timing.

4.6 APP_ROOT Configuration file

- Explanation
 - This is intended for users to edit with a text editor, etc., but it is also possible for applications that use conanair to provide editing functions.
 - This file contains the following settings.
 - ✧ Pass of APP_ROOT (location)
 - ✧ A number of generations for the measurement data to keep.
 - ✧ Data synchronization intervals
 - ✧ Others (hidden settings)
- Configuration file name and location: CAIR_ROOT/app_root.conf
- Description rules
 - Field 1: Type of setting [APP_ROOT|DATA_GEN|DATA_SYNC]
 - Field 2: Configuration value (see sample)
- Duplicate records:
 - If the same configuration type appears more than twice, the last record is valid.
- Sample contents:

Sample app_root file

APP_ROOT, ¥¥server¥share¥somefolder¥subfolder¥
--

DATA_GEN, 3

DATA_SYNC, 10

➤ Remarks:

- ✧ The path of APP_ROOT should be whatever can open the file in read/write mode with using the concatenated string of filenames after it.
- ✧ The number in DATA_GEN is the number of generations to be retained.
- ✧ The number of DATA_SYNC is the minimum interval to process writing data under APP_ROOT, in minutes.
- ✧ Writing data under APP_ROOT is performed when more than DATA_SYNC minutes have elapsed since the last time data was received.

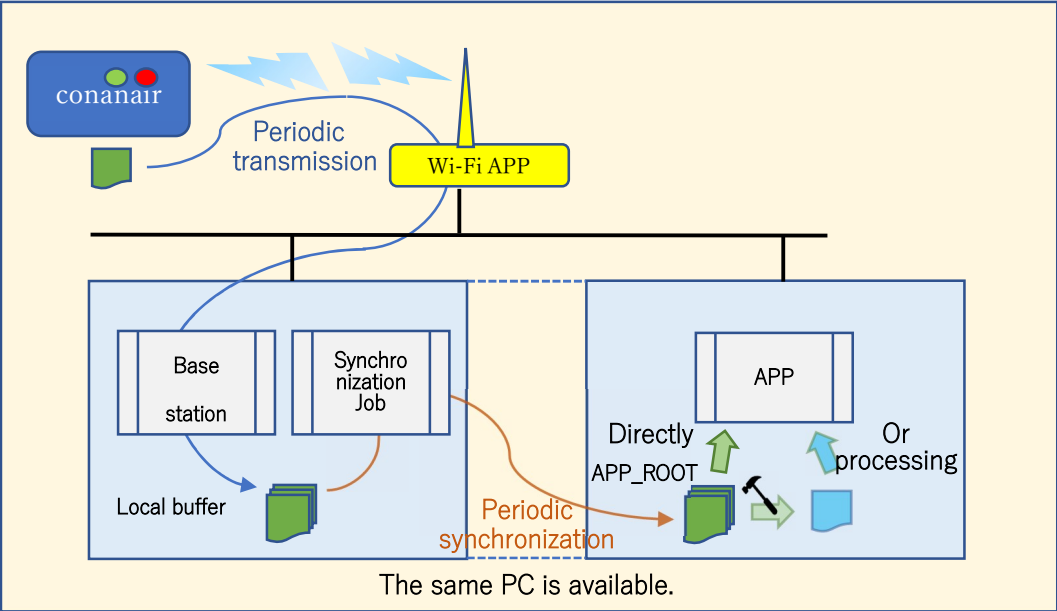
● Assumed measurement data saving process:

To minimize the risk of losing received measurement data, the base station application performs buffer duplication as follows (see next chapter).

- The process to receive measurement data makes files in a different location once and in a location in storage on the computer where the base station is running.
 - ✧ This location is called the local buffer.
- A process that synchronizes the data differently from the one that receives the measurement data is started for each DATA_SYNC setting value,
 - ✧ For each tag, the number of measured value files remaining in the local buffer is checked, and if it exceeds the DATA_GEN setting value, the old files are deleted.
 - ✧ It synchronizes the difference between APP_ROOT and the local buffer.
 - ✧ The synchronization process copies the created files and deletes the lost files.

5 Measurement result file

5.1 Communication channels for measurement results

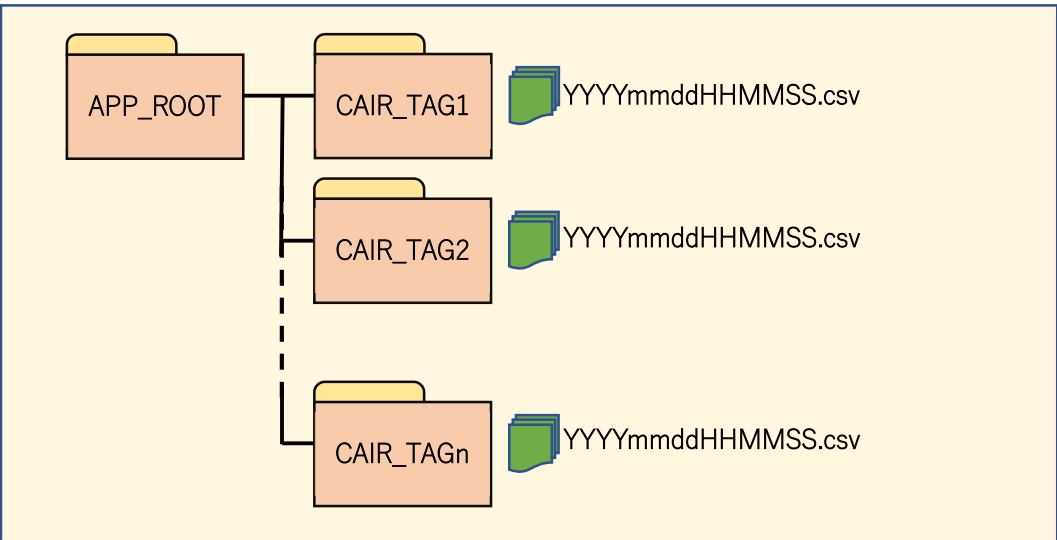


If the data needs to be processed beforehand for the application to read it, please prepare the processing process on the application side.

5.2 Folder Tree

The folder tree under APP_ROOT is shown below.

The local buffer has the same structure, but access from the app is prohibited.



5.3 Data format

Within a single file, the first half contains summary information, and the second half contains actual data.

Both are in a comma-separated format, but the number of columns differs.

All numbers may contain a small number and should be interpreted as real numbers.

The encoding and newline characters are as follows.

- Encoding: UTF-8 with BOM
- Newline character: CR+LF (0x0d 0x0a: Windows style)

5.3.1 The summary information

The summary information section has 2 lines, forming a Key-Value pair.

The number of rows is decided for each version, but to get flexibility changes, the end of the summary section is indicated by a special key.

Although the real data described in the next section is compressed irreversibly for transmission, this summary information is calculated based on the uncompressed values, so it differs from what is calculated from the real data.

No.	Key	Value explanation
1	ID	ID=CAIR_ID 【characters】 of conanair
2	Tag	Tag associated with the ID Tag= CAIR_TAG 【characters】
3	Fileame	File name (measurement date) 【characters】
4	ODR	Nominal sample rate[samples/sec] 【All subsequent values are numeric】
5	N_Samples	Number of samples (XYZ 1 set is counted as one)
6	Act_ms	Actual measurement time[milliseconds] ● Actual sample rate=N_Samples x 1000 ÷ Act_ms
7	Vdd3V3	Supply voltage at the time of measurement[V]
8	RAccAveX	Average acceleration in each axis, including gravity [m/s ²] ● Sensor's mounting position to the vertical direction can be determined.
9	RAccAveY	
10	RAccAveZ	
11	RAccPkXP	Peak value of acceleration including gravity for each axis [sensor readout]. ● P at the end is the positive side peak value. ● N at the end is the peak value on the minus side. ● The minimum value that the sensor can measure is -4096 and the maximum value is 4094, so if a value is obtained within these measurements, it can be assumed to be over scaled (it may be slightly unscaled).
12	RAccPkXN	
13	RAccPkYP	
14	RAccPkYN	
15	RAccPkZP	
16	RAccPkZN	

(表の続き)

No.	Key	Value の説明
17	HAccPkXP	Peak acceleration in each axis and in removed 3D vector gravity [m/s ²]. <ul style="list-style-type: none"> ● High-pass filter of Fc=10Hz is applied to remove gravity acceleration ● P at the end is the positive peak value ● N at the end is the peak value on the minus side ● For 3D vectors, there is no minus side because it is an absolute value (magnitude).
18	HAccPkXN	
19	HAccPkYP	
20	HAccPkYN	
21	HAccPkZP	
22	HAccPkZN	RMS value of acceleration in each axis and in removed 3D vector gravity [m/s ²]. <ul style="list-style-type: none"> ● A high-pass filter of FC=10Hz is applied to remove gravity acceleration.
23	HAccPk3D	
24	HAccRmsX	
25	HAccRmsY	
26	HAccRmsZ	
27	HAccRms3D	Peak value of velocity [mm/s] calculated from the acceleration excluding gravity for each axis and 3D vector. <ul style="list-style-type: none"> ● P at the end is the peak value on the positive side ● N at the end is the peak value on the minus side ● For 3D vectors, there is no negative side because it is an absolute value (magnitude).
28	VelPkXP	
29	VelPkXN	
30	VelPkYP	
31	VelPkYN	
32	VelPkZP	RMS value of velocity calculated from the acceleration in each axis and 3D vector excluding gravity [mm/s].
33	VelPkZN	
34	VelPk3D	
35	VelRmsX	
36	VelRmsY	
37	VelRmsZ	Transmission strength of the connected Wi-Fi access point [dBm].
38	VelRms3D	
39	AP_RSSI	Temperature of the internal substrate [°C]
40	Dev_Tmp	
41	END Summary	End of summary section. Values are present but not meaningful. (To align the number of columns)

5.3.2 Real data

There is no key in each row of real data. It contains the following columns of data only.

There is no header row, so the number of rows is equal to N_Samples in the summary information.

No.	データ ID	説明
1	RAccX	Acceleration[m/s ²] including gravity in each axis. ● It can determine the position change of the sensor to the vertical direction.
2	RAccY	
3	RAccZ	
4	HAccX	Acceleration in each axis and 3D vector without gravity [m/s ²] ● Pure vibration component, unaffected by position changes ● A high-pass filter of Fc=10Hz is applied to remove gravity acceleration.
5	HAccY	
6	HAccZ	
7	HAcc3D	
8	VelX	Velocity [mm/s] calculated from the acceleration in each axis and in removed 3D vectors velocity.
9	VelY	
10	VelZ	
11	Vel3D	

Revision History

2019-4-12 0.0.3 → 0.0.4

Deleted the description of pre-release documents from "1.1 Description".

2021-3-18 0.0.4 → 2.0.0

Two items were added to 5.3.1 Summary Information by upgrading to V2
(AP_RSSI、 Dev_Tmp) were added.

Conclusion