Document status

Published status	Confidentiality status
☐ Draft	☐ Internal
☒ Released	☒ NDA
☐ Canceled	☐ Public

Version control

Ver.	Date	Summary of changes
7.00	2015-06-11	Completely rewritten. See Chapter 5 for more information.
7.01	2015-06-12	Minor layout corrections.
7.02	2015-07-01	Note added under PFLAE.
7.03	2015-07-29	Corrected <AlarmLevel> and <AlarmType> under PFLAU.
7.04	2015-09-14	Corrected <IDType> under PFLAA.
7.05	2015-03-16	Added description of error 0x43 to PFLAE.
7.06	2016-05-18	Changed 0x43 error description.
7.07	2016-09-09	Increased max range to other aircraft in PFLAA.
7.08	2017-03-08	Changed PFLAE sentence description.
7.09	2017-08-24	Document name changed. Added changes for protocol version 8. Clarified aircraft types and <GroundSpeed> in PFLAA sentence. Added Chapters 4 and 5. Editorial changes.
7.10	2018-07-11	Added PFLAJ documentation. Editorial changes.
7.11	2019-07-08	Added protocol version 4 changes. New protocol version 9 features introduced. New PFLAN sentence.
7.12	2019-07-31	The PFLAE sentence can now be requested on PowerFLARM. New error code 0x94. Editorial changes.
7.13	2020-03-04	Added <TisbAdsrClientStatus> field to PFLAJ. Removed length limit of <Info> field in PFLAQ.
7.14	2021-05-10	Added PFLAF for traffic and alarm simulation. Added new statistics in PFLAN output, changed semantic of RFTOP. Editorial changes.
7.15	2021-07-07	Clarified when command termination is sent on PFLAE.
7.16	2021-10-07	Included description of PFLAL messages
7.17	2022-08-16	Added details to stealth mode

Scope and summary

This document provides information about how to communicate over the serial interface with any FLARM device. The main use case is when designing FLARM Compatible displays or other devices.

The sentence composition is described, as well as all applicable sentences and commands.

This document should be read in combination with the following documents:

- FTD-014 Configuration Specification
- FTD-013 FLARM Compatible Certification Specification

These documents can be acquired by developers from this page.

Table of contents

1 Overview

This document provides information about how to communicate over the serial interface with any FLARM device. It is applicable to any Classic FLARM- or PowerFLARM-based device, including devices designed and manufactured under license from FLARM Technology Ltd. The main use case is when designing a FLARM Compatible display or any other device which consumes FLARM data.

Note that this document, as well as the interface described herein, is proprietary and copyright protected. FLARM is an internationally registered trademark and cannot be used without a license.

The document assumes you are familiar with serial communication and the basics of the NMEA 0183 version 2.0 protocol[1]. The NMEA 0183 standard is a copyright-protected document. The copyright is owned by the National Marine Electronics Association, Inc., 7 Riggs Avenue, Severna Park, MD 21146, USA. The NMEA 0183 standard may be purchased from NMEA online at www.nmea.org. NMEA® is a registered trademark of the National Marine Electronics Association Inc. The NMEA Director agreed to this document on March 9, 2005.

The most recent version of this document and test data streams are available to developers from FLARM Technology. You can also subscribe on our website to our newsletter in order to receive the latest news and firmware updates.

There is no formal requirement to have an agreement with FLARM Technology before implementing the data port protocol. However, we would very much appreciate being informed of any implementations on info@flarm.com. That way, we can inform you of any changes to the protocol and other important issues.

For manufacturers who would like to market their product as "FLARM Compatible", there is a FLARM Compatible certification program. Equipment which fulfills the certification specification can claim to be FLARM Compatible and use the FLARM Compatible logo. Contact us for more information and to receive the certification specification document.

The document applies to PowerFLARM- and Classic FLARM-based devices with firmware version 7.04 and later.

Suggestions to improve this document may be sent to info@flarm.com.

[1] The current version of the NMEA 0183 standard is 4.10.

2 Working Principle

FLARM utilizes position and movement information obtained from an integrated GPS/GNSS receiver and optionally from an embedded barometric sensor. The future flight path of the aircraft is predicted and transmitted over a low-power short-range radio channel as a very short digital message once per second. These messages are received by other FLARM devices within range and then compared with the own predicted flight path. At the same time, the own flight path is compared with fixed obstacles in the optionally installed obstacle database (power lines, antennas, cable cars, etc.). If a collision risk is calculated, FLARM warns the pilot of the chronologically closest collision risk.

GPS and collision data are transmitted as serial data to be used in external applications (e.g. displays). Several manufacturers of avionics and PDA software as well as ground-based applications use FLARM data in their applications.

The range is subject to the antenna installation in the aircraft. For e.g. PowerFLARM Core with external antennas, the range is normally more than 10 km. For Classic FLARM-based devices, the range is typically 3–5 km. The effective range can be verified with an online tool[2].

Collision warnings and the three alarm levels are issued depending on the forecasted time to impact, not a geometrical distance. The first alarm level is usually issued at 18–13 seconds, the second level at 12–9 seconds, and the third level at 8–0 seconds prior to the predicted impact. Each warning lasts as long as the alarm level is applicable. Depending on changes to the forecast, alarm levels might change or disappear. Warnings are highly selective, i.e. they are only issued when there is imminent danger. The warning sensitivity can be configured. As an additional feature, the user can be informed about other aircraft in the vicinity, even when not posing any risk of collision. This traffic information is limited to a configurable horizontal and vertical distance. FLARM is designed to handle up to 50 aircraft in range and will experience graceful performance degradation with additional aircraft in range.

To work properly, FLARM must have a 3D GPS fix. Furthermore, the antenna installation must facilitate a usable range.

FLARM uses for the radio communication between devices a proprietary, patented, and copyright protected protocol in regionally different frequency bands. The radio communication is protected against unauthorized access. The design is protected by several patents. The radio communication protocol is not public. Any non-licensed use, dissemination, copying, implementation, or reverse engineering or

[2] http://flarm.com/support/tools-software/flarm-range-analyzer/

decompilation of the FLARM radio communication protocol, the FLARM hardware and software, or parts thereof, is forbidden by law and will be prosecuted. FLARM is an internationally registered trademark and cannot be used without a license, except as provided in the document "FTD-011 FLARM Logo Usage and Branding Guidelines". Technical specifications are subject to change at any time without notice.

3 Design Philosophy

FLARM cannot warn reliably in all situations. FLARM does not issue resolution advisories. FLARM can only warn of aircraft which are equipped with FLARM (or SSR transponders and ADS-B Out equipment with specific FLARM devices) and of obstacles stored in its database. The use of FLARM does not allow a change of flight tactics or pilot behavior. It is the sole responsibility of the pilot in command to decide upon the use of FLARM. FLARM Technology Ltd cannot be held liable under any circumstances.

When designing a user interface for a display using the serial data provided on the data ports from FLARM, try to be simple and clear when presenting any data to the pilot. Use a presentation form that is suitable for immediate comprehension of danger level. We do not, in general, recommend the design of TCAS-style displays, as many light aircraft pilots are not trained to use this kind of display. Do not give any resolution advisories and ensure that your user interface cannot be misinterpreted as such.

For visualization of collision warnings, we recommend a 12-segment bearing indicator and a 4-segment vertical angle indicator. This applies also to raster displays (LCD, dot matrix, etc.), where the original LED display can be simulated or similar. Aural warnings should consist of beeps with dynamic beep frequency and amplitude, depending on the alert level. Alternatively, simple voice alerts ("traffic 3 o'clock, above") can be used.

Traffic information (non-threat traffic) should only be displayed on maps oriented heading-up or track-up. If displaying traffic information on maps oriented heading-up, be aware that the relative bearing received from FLARM is relative to true track. In such cases, the information must be corrected for variation, deviation, and wind. Especially on LED displays, it must be ensured that traffic information is not mixed up with collision warnings.

Obstacle warnings and Alert Zone warnings should be generic (i.e. without any reference to position or bearing) and not be mixed up with collision warnings. Do not plot any obstacles on a moving map based on warning information provided by FLARM.

When selecting colors for warnings on an LED display, ensure that CS 23.1322/22.1322 is complied with. An aircraft collision warning (alarm level 1–3) should be considered as a warning light. An obstacle collision warning should be considered as a warning light (needs to be distinguished from an Alert Zone warning; see `PFLAU` sentence). An Alert Zone warning should be considered as a caution light.

Alerts should make the pilot look outside the cockpit, not on a display. Therefore, whenever a warning is given, show only the warning in a conspicuous manner. At the same time, immediately stop displaying anything that is not relevant, e.g. surrounding aircraft that do not represent a danger.

Errors and degraded functionality should be shown to the user in a clear manner.

Make it clear to the users that they should under no circumstances rely on FLARM, FLARM's serial output, or your application using data coming from FLARM.

Users should not be disincentivized from using FLARM. Especially in competitions, pilots might not want to be followed by other aircraft. Some special features in FLARM (e.g. a configurable stealth mode that cannot be changed in-flight, and which prevents received data from being shown on displays and PDAs) ensure that FLARM information cannot be abused to gain a competitive advantage.

A FLARM display is the user interface of an installed FLARM system. The FLARM device itself often does not have an exposed user interface, controls, operating instructions, or similar. The display is, in most cases, the only means by which the user can interact with the FLARM system (in both directions). The display therefore normally needs to support all aspects of FLARM. More information and guidance is available in "FTD-013 FLARM Compatible Certification Specification".

4 Device Categories

FLARM devices can differ in terms of number and types of interfaces. The specification can normally be found in the installation manual of each device.

Not all configuration settings and sentences are supported by all devices. Two main categories of devices exist: Classic FLARM- and PowerFLARM-based devices. The device category can be determined by enquiring `$PFLAC,R,DEVTYPE` (described in FTD-014 Configuration Specification). The existing device types, and the category to which they belong, are listed in the table below.

Device Category	Device Type (`DEVTYPE`)
PowerFLARM-based	`PowerFLARM-Core`
	`PowerFLARM-Portable`
	`PowerFLARM-AM`
Classic FLARM-based	`Flarm04`
	`Flarm05`
	`Flarm06`
	`Flarm-IGC05`
	`Flarm-IGC06`
	`FLYTEC`
	`LX_IGC08`
	`LX06`
	`LX06_FR`
	`LXV`
	`LXV_FM`
	`LXV_FM_IGC`
	`OZ06`
	`OZ_IGC`
	`TRXFLM`
	`SOMAX`

4.1 Device features

Device features should be used to determine if a particular feature is available, indicating that a specific set of configuration items can be applied.

Features can be requested using the configuration information item `CAP`:

Feature	`CAP` Information Value
Audio Output	AUD
Alert Zone Generator	AZN
Pressure Sensor	BARO
Battery Compartment	BAT
Second Data Port	DP2
Engine Noise Level Sensor	ENL
Ground Station Device	GND
IGC Approved Recorder	IGC
Obstacle Database Installed	OBST
Antenna Diversity (Second Antenna)	RFB
SD Card Slot	SD
Garmin TIS Protocol Support	TIS
Integrated User Interface	UI
USB Slot	USB
SSR/ADS-B Module	XPDR

Example:

```
> $PFLAC,R,CAP

$PFLAC,A,CAP,OBST;IGC;SD;BARO*
```

FLARM device has an obstacle database installed, IGC approved recorder, SD card slot, and barometric sensor.

5 Serial Communication Ports Description

A bidirectional serial interface (RS-232) is used to communicate with the FLARM devices. 8 data bits, no parity, no handshake, and 1 stop bit for the RS-232 settings are used.

The FLARM communication interface is flexible and highly configurable. It supports the NMEA and FLARM proprietary protocols and is truly multi-port and multi-protocol capable. Each protocol (FLARM and NMEA) can be assigned to several ports at the same time with individual settings (baud rate, message configuration, etc.) for each port, with limitations as specified. The number of supported ports depends on the specific device.

Note: The extension port present on some Classic FLARM-based devices supports only the `PFLAU` sentence, hence it is not further mentioned in this document.

To enable specific sentences on a port, the FLARM proprietary and/or NMEA protocol must be enabled on that port using the configuration setting `NMEAOUT`.

The baud rates can be configured individually for each serial port using configuration setting `BAUD`. A connecting device may thus not assume that a particular baud rate is set. It is strongly suggested to implement automatic baud rate detection.

More information regarding the BAUD and `NMEAOUT` configuration settings can be found in FTD-014 Configuration Specification.

FLARM may interrupt RF communication and collision warnings for some seconds upon receiving NMEA commands. Therefore, sending sentences to FLARM should be avoided during flight.

6 Protocol Version Changes

The data port protocol version must normally be configured by the connecting product/display; see the section on setting `NMEAOUT` in FLARM Configuration Specification. Alternatively, the installation manual must specify the version that must be configured in the FLARM device.

The sections below specify changes and additions introduced with each protocol version.

6.1 Protocol version 4 and higher

- Traffic advisory output in PFLAU sentence.

6.2 Protocol version 6 and higher

- Non-directional targets are output (`PFLAA` sentence)

6.3 Protocol version 7 and higher

- Alert Zone alarms output (`PFLAU` sentence)
- Introduced Alert Zone information sentence `PFLAO`
- Error description added to `PFLAE` sentence

6.4 Protocol version 8 and higher

- "No track" setting added to `PFLAA` sentence
- Unique Obstacle ID sent in `PFLAU` sentence

6.5 Protocol version 9 and higher

- Source and RSSI added to `PFLAA` sentence

7 Sentence Composition

This document assumes you are familiar with the full NMEA 0183 version 2.0 specification. FLARM allows input and output.

Sentences consist of NMEA 0183 standard `GPRMC`, `GPGGA`, and `GPTXT` sentences and NMEA 0183-style proprietary sentences that start with `PFLA`, in addition to Garmin®'s proprietary sentence `PGRMZ`[3]. `FLA` has been officially assigned by NMEA as the FLARM manufacturer code.

All commands must start with "`$`" (0x24) and end with `<CR><LF>` (0x0D0A). All sentences from FLARM start with "`$`" (0x24) and end with the checksum delimiter "`*`" (0x2A), followed by two NMEA 0183 standard checksum characters and `<CR><LF>` (0x0D0A). The checksum is the two-digit hexadecimal representation of XOR of ASCII codes of all characters between, but not including, the `$` and `*`. For matters of simplicity, this document does usually not mention these characters although they must be provided in commands to FLARM and are part of the sentences sent by FLARM.

Fields are delimited with a comma (0x2C), even when a field is "empty". When a field is "omitted", it, however, means that the comma is absent. The field length is variable. Commands must only consist of valid ASCII characters.

The sentences are not case sensitive except where explicitly stated.

The maximum number of characters in a sentence is 83, consisting of a maximum of 80 characters between the starting delimiter "`$`" and the terminating delimiter `<CR><LF>`.

Sentences sent by FLARM not following this syntax must be ignored by the receiving device without further consequences. Design the application fault-tolerant.

The value types and ranges for each field in the sentences are specified separately. Values outside the given range should be ignored. The following naming convention is used:

- Integer: number which is written without a fractional component
- Fixed point: real number with a fixed number of digits after the radix point (dot)

[3] See 'Garmin Proprietary NMEA 0183 Sentence TECHNICAL SPECIFICATION', part number 190-00684-00, revision B, April 2006. Garmin® is a registered trademark of Garmin Ltd.

- Decimal: number which has ten as its base (e.g. 123)
- Hexadecimal: integer number which has sixteen as its base (e.g. AF)
- String: a sequence of ASCII characters except for NULL
- Floating point: real number with an arbitrary number of digits (within limits as specified) after the radix point (dot)

8 Sentences

All sentences are described in this chapter.

The typical sentence order during normal operation is as follows: `GPRMC`, `(PGRMZ)`, `GPGGA`, `{PFLAA}`$_n$, `PFLAU`. However, which sentences are actually sent depend on the `NMEAOUT` setting. `PGRMZ` is available on all devices with a pressure sensor. `n` is an integer >=0, dependent on the chosen baud rate and the number of aircraft within range. When implementing a sentence parser, do not expect the above order to be maintained at all times; also, don't observe a specific pattern and expect it to stay unchanged in all hardware versions and in the future, as the implementation might vary from hardware to hardware, and might be changed in the future.

Carefully read the description and usage given for each sentence on all subsequent pages.

8.1 PFLAU – Heartbeat, status, and basic alarms

Syntax:

```
PFLAU,<RX>,<TX>,<GPS>,<Power>,<AlarmLevel>,<RelativeBearing>,
<AlarmType>,<RelativeVertical>,<RelativeDistance>[,<ID>]
```

Description:

Heartbeat message; output once per second. Consumers should use this message to detect the presence (and absence) of a compatible data stream.

The sentence summarizes the most relevant status information from the last one-second interval: RF status (RX, TX), power state, and the most important current threat, either traffic, an obstacle, or an alert zone. Consumers with limited resources (e.g. with respect to display capabilities or computational resources) can thus use `PFLAU` to display basic safety information. Other consumers shall also use `PFLAA` for extended information.

On devices with SSR/ADS-B Module, non-directional targets are output if enabled (`PCASPFLAU` configuration setting).

For data port version >=7, Alert Zone alarms are available; see the `<AlarmLevel>` and `<AlarmType>` fields.

For data port version >= 4, traffic advisory notifications (INFO alarms) are available, see the `<AlarmType>` field.

Input / Output:

Only sent by FLARM

Availability:

Depending on configuration: `PFLAC,,NMEAOUT`

Periodicity:

Sent once every second (1.8 s at maximum)

Parameters:

<RX>	Decimal integer value. Range: from 0 to 99. Number of devices with unique IDs currently received regardless of the horizontal or vertical separation. Because the processing might be based on extrapolated historical data, `<Rx>` might be lower than the number of aircraft in range, i.e. there might be other traffic around (even if the number is zero). Do not expect to receive `<Rx> PFLAA` sentences, because the number of aircraft being processed might be higher or lower.
<TX>	Decimal integer value. Range: from 0 to 1. Transmission status: 1 for OK and 0 for no transmission.
<GPS>	Decimal integer value. Range: from 0 to 2. GPS status: 0 = no GPS reception 1 = 3d-fix on ground, i.e. not airborne 2 = 3d-fix when airborne If `<GPS>` goes to 0, FLARM will not work. Nevertheless, wait for some seconds to issue any warnings.

<Power>	Decimal integer value. Range: from 0 to 1. Power status: 1 for OK and 0 for under- or over-voltage.
<AlarmLevel>	Decimal integer value. Range: from 0 to 3. Alarm level as assessed by FLARM: 0 = no alarm (also used for no-alarm traffic information) 1 = aircraft or obstacle alarm, 13-18 seconds to impact, Alert Zone alarm, or traffic advisory (<AlarmType> = 4) 2 = aircraft or obstacle alarm, 9-12 seconds to impact 3 = aircraft or obstacle alarm, 0-8 seconds to impact **Note:** Alert Zone: If inside the zone, alarm level is 1 for 4 seconds, then 0 for 12 seconds, then repeats.
<RelativeBearing>	Decimal integer value. Range: -180 to 180. Relative bearing in degrees from true ground track to the intruder's position. Positive values are clockwise. 0° indicates that the object is exactly ahead. The field is empty for non-directional targets or when no aircraft are within range. For obstacle alarm and Alert Zone alarm, this field is 0.

<AlarmType>	Hexadecimal value. Range: from 0 to FF. Type of alarm as assessed by FLARM 0 = no aircraft within range or no-alarm traffic information 2 = aircraft alarm 3 = obstacle/Alert Zone alarm (if data port version < 7, otherwise only obstacle alarms are indicated by <AlarmType> = 3) 4 = traffic advisory (sent once each time an aircraft enters within distance 1.5 km and vertical distance 300 m from own ship; when data port version >=4) xx = Alert Zone alarm (see comment below) When data port version >=7, the type of Alert Zone is sent as <AlarmType> in the range 10..FF. Refer to the <ZoneType> parameter in the PFLAO sentence for a description.
<RelativeVertical>	Decimal integer value. Range: from -32768 to 32767. Relative vertical separation in meters above own position. Negative values indicate that the other aircraft or obstacle is lower. The field is empty when no aircraft are within range. For Alert Zone and obstacle warnings, this field is 0.
<RelativeDistance>	Decimal integer value. Range: from 0 to 2147483647. Relative horizontal distance in meters to the target or obstacle. For non-directional targets, this value is estimated based on signal strength. The field is empty when no aircraft are within range and no alarms are generated. For Alert Zone, this field is 0.

<ID>	The field is omitted for protocol version < 4. 6-digit hexadecimal value (e.g. "5A77B1") as configured in the target's `PFLAC,,ID`. The interpretation is only delivered in `<ID-Type>` in the `PFLAA` sentence (if received for the same aircraft). The `<ID>` field is the ICAO 24-bit address for Mode-S targets and a FLARM-generated ID for Mode-C targets. The ID for Mode-C targets may change at any time. The field is empty when no aircraft are within range and no alarms are generated. For obstacles, in PowerFLARM-based devices with data port version >= 8, this field is set to the obstacle ID unique for the obstacle and the specific database version installed. In other cases, it is set to FFFFFF. In case of Alert Zone warning, the FLARM ID of the Alert Zone station is output.

Example:

```
$PFLAU,3,1,2,1,2,-30,2,-32,755*
```

FLARM is working properly and currently receives 3 other aircraft. The most dangerous of these aircraft is at 11 o'clock, position 32m below and 755m away. It is a level 2 alarm.

Example:

```
$PFLAU,2,1,1,1,0,,0,,,*
```

FLARM is working properly and receives two other aircraft. They are both out of range.

Example:

```
$PFLAU,2,1,2,1,1,-45,2,50,75,1A304C*
```

FLARM is working properly and receives two other aircraft. The most dangerous of these aircraft has the ID "1A304C" and it is at 9 o'clock, position 50m below and 75m away. Level 1 alarm.

Example:

```
$PFLAU,2,1,2,1,1,0,41,0,0,A25703*
```

FLARM is working properly and receives the Alert Zone signal. The Alert Zone station ID is A25703 and it is a skydiver drop zone.

8.2 PFLAA – Data on other proximate aircraft

Syntax:

```
PFLAA,<AlarmLevel>,<RelativeNorth>,<RelativeEast>,
<RelativeVertical>,<IDType>,<ID>,<Track>,<TurnRate>,<GroundSpeed>,
<ClimbRate>,<AcftType>[,<NoTrack>][,<Source>,<RSSI>]
```

Description:

Data on other proximate aircraft, intended for connected devices with sufficient CPU performance. This sentence should be treated with utmost flexibility and tolerance on a best effort base. Individual parameters may be empty. The sentence is only sent when port baud rate is 19.2k or higher. In case of serial port congestion or high CPU load, this sentence may be omitted for several objects independent of the alarm level. On devices with SSR/ADS-B Module, ADS-B and non-directional targets are output as well (transponder Mode-C/S only from protocol version 6 and higher).

Obstacle information is not delivered with this sentence.

Note that in case of many targets within range, individual targets, including the most dangerous one, might not be delivered every second, not regularly, or not at all, due to less strict priority handling for the `PFLAA` sentence. **Always use** `PFLAU` **as primary alarm source.** Usually, but not always, the last `PFLAA` sentence is the one causing the `PFLAU` content. The other `PFLAA` sentences are not ordered. Do not expect to receive `PFLAU <Rx>` times `PFLAA` sentences, because the number of aircraft being processed might be higher or lower. `PFLAA` sentences can be based on extrapolated historical data. `PFLAA` sentences are limited to other aircraft with a horizontal and vertical distance less than the configured range. On Classic FLARM-based devices, the vertical distance is always 500 m. Non-moving aircraft are suppressed.

Input / Output:

Only sent by FLARM

Availability:

Depending on configuration: `PFLAC,,NMEAOUT` and `PFLAC,,BAUD` (only when baud rate >= 19.2kB).

Periodicity:

Sent when available. Can be sent several times per second with information on several (but not always all) surrounding targets.

Parameters:

<AlarmLevel>	Decimal integer value. Range: from 0 to 3. Alarm level as assessed by FLARM: 0 = no alarm (also used for no-alarm traffic information) 1 = alarm, 13-18 seconds to impact 2 = alarm, 9-12 seconds to impact 3 = alarm, 0-8 seconds to impact
<RelativeNorth>	Decimal integer value. Range: from -20000000 to 20000000. Relative position in meters true north from own position. If `<RelativeEast>` is empty, `<RelativeNorth>` represents the estimated distance to a target with unknown bearing (transponder Mode-C/S).
<RelativeEast>	Decimal integer value. Range: from -20000000 to 20000000. Relative position in meters true east from own position. The field is empty for non-directional targets.
<RelativeVertical>	Decimal integer value. Range: from -32768 to 32767. Relative vertical separation in meters above own position. Negative values indicate that the other aircraft is lower. Some distance-dependent random noise is applied to altitude data if stealth mode is activated either on the target or own aircraft and no alarm is present at this time.

<IDType>	Decimal integer value. Range: from 0 to 2. Defines the interpretation of the following <ID> field. 0 = random ID, configured or if stealth mode is activated either on the target or own aircraft 1 = official ICAO 24-bit aircraft address 2 = fixed FLARM ID (chosen by FLARM) The field is empty if no identification is known (e.g. transponder Mode-C).
<ID>	6-digit hexadecimal value (e.g. "5A77B1") as configured in the target's PFLAC,,ID sentence. The interpretation is delivered in <ID-Type>. The field is empty if no identification is known (e.g. Transponder Mode-C). Random ID will be sent if stealth mode is activated either on the target or own aircraft and no alarm is present at this time.
<Track>	Decimal integer value. Range: from 0 to 359. The target's true ground track in degrees. The value 0 indicates a true north track. This field is empty if stealth mode is activated either on the target or own aircraft and for non-directional targets.
<TurnRate>	Currently this field is empty.
<GroundSpeed>	Decimal integer value. Range: from 0 to 32767. When the aircraft is considered moving, the target's ground speed in m/s, forced to > 0. When the aircraft is considered on the ground, the field is forced to 0. This field is empty if stealth mode is activated either on the target or own aircraft and for non-directional targets.

<ClimbRate>	Decimal fixed-point number with one digit after the radix point (dot). Range: from -32.7 to 32.7. The target's climb rate in m/s. Positive values indicate a climbing aircraft. This field is empty if stealth mode is activated either on the target or own aircraft and for non-directional targets.
<AcftType>	Hexadecimal value. Range: from 0 to F. Aircraft types: 0 = (reserved) 1 = glider/motor glider (turbo, self-launch, jet) / TMG 2 = tow plane/tug plane 3 = helicopter/gyrocopter/rotorcraft 4 = skydiver, parachute (Do not use for drop plane!) 5 = drop plane for skydivers 6 = hang glider (hard) 7 = paraglider (soft) 8 = aircraft with reciprocating engine(s) 9 = aircraft with jet/turboprop engine(s) A = unknown B = balloon (hot, gas, weather, static) C = airship, blimp, zeppelin D = unmanned aerial vehicle (UAV, RPAS, drone) E = (reserved) F = static obstacle

<NoTrack>	Field is omitted if data port version <8. The target's configured no track setting. Decimal integer value. Range: from 0 to 1. 0 = no track option not set 1 = no track option set Targets with "no track" enabled express their intention to remain private. Data from these targets may thus not be persisted in any way (e.g. in a database). If the data is transmitted to a third-party system (e.g. a server), then the implementer must make sure the third-party system also respects this rule. Such targets will furthermore be suppressed from `$PFLAA` output if ownship does not move, unless the target is closer than 200m horizontally and 100m vertically.

<Source>	The field is omitted if data port version <9. Data source of the $PFLAA sentence: 0 = FLARM 1 = ADS-B 3 = ADS-R (rebroadcasting of UAT ADS-B to 1090 MHz) 4 = TIS-B (broadcast of location of non-ADS-B equipped aircraft) 6 = Mode-S (non-directional targets) If the same target is received from multiple sources, the following precedence applies: FLARM > ADS-B > ADS-R > TIS-B > Mode-S. For ADS-R and TIS-B, no alarm is computed (`AlarmLevel` = 0). Note: ADS-R and TIS-B position reports may be inaccurate due to a low update rate and/or extrapolation. Use for indicative display only. Note: ADS-R and TIS-B targets are sent by the ANSP only to so-called ADS-R and TIS-B clients. If ownship is not set up to act as such a client, this information may not be sent. Refer to DO-338 for details.
<RSSI>	The field is omitted if data port version <9. Signal level of the received target in dBm (example: "-71.2"). Empty if unknown. This field can be used to help assess the quality of the radio link. It depends on the installation of the sending station, the installation of the receiving station and the distance.

Example:

```
$PFLAA,0,-1234,1234,220,2,DD8F12,180,,30,-1.4,1*
```

There is a glider in the south-east direction, 1.7km away (1.2km south, 1.2km east), 220m higher flying on south track with a ground speed of 30m/s in a slight left turn with 4.5°/s turning rate, sinking with 1.4m/s. Its ID is a static FLARM-ID "DD8F12". There is no danger.

Note: The behavior for stealth targets is as follows:

	Target is away more than 2km / 300m vertical	Target is closer than 2km / 300m vertical	Target is nearby / collision alarm detected
Target ID	Not available	Anonymous ID	Anonymous ID
Relative position	Not available	Available	Available
Relative altitude	Not available	Available with noise	Available
Climb rate	Not available	Not available	Available
Track	Not available	Not available	Available
Speed	Not available	Not available	Available

8.3 PFLAE – Self-test result and errors codes

Syntax:

`PFLAE,<QueryType>,<Severity>,<ErrorCode>[,<Message>]`

Description:

Self-test results after startup and error information during operation. Always watch for this sentence. Inform the user when functionality is not available due to errors.

In data port version >= 7, the `$PFLAE` sentence contains a textual description of the error in `<message>`, with the sole exception of `DEVTYPE` = `Flarm04`, where this field is omitted.

Connected devices should display that an error in FLARM is present ("ERROR"), the error code, and the message if present. This should be done also for error codes that are not listed below.

Note: Usually, the FLARM system can issue the `PFLAE` sentence at any time during operation, and it is always sent at the end of the start-up. If there is no error during the start-up, sentence `PFLAE,A,0,0` is sent once for acknowledgment. If an error occurred, the error sentence is sent once per second for 30 seconds. If there are more errors, the same procedure is then repeated for the other errors.

Note: The message can also be requested via the `$PFLAE,R` command. In that case, all errors are returned one by one, with one sentence per error. The terminating `$PFLAE,A` sentence is used to denote the end of the command.

Input / Output:

Bidirectional, i.e. can also be requested

Availability:

Always available, no configuration

Periodicity:

Sent once after startup and completion of self-test, when errors occur, and when requested

Parameters:

<QueryType>	R = request FLARM to send status and error codes; other parameters should then be omitted A = FLARM sends status (requested and spontaneous)
<Severity>	Decimal integer value. Range: from 0 to 3. 0 = no error, i.e. normal operation. Disregard other parameters. 1 = information only, i.e. normal operation 2 = functionality may be reduced 3 = fatal problem, device will not work
<ErrorCode>	Hexadecimal value. Range: from 0 to FFF. Error codes: 11 = Firmware expired (requires valid GPS information, i.e. will not be available in the first minute or so after power-on) 12 = Firmware update error 21 = Power (e.g. voltage < 8V) 22 = UI error 23 = Audio error 24 = ADC error

	25 = SD card error 26 = USB error 27 = LED error 28 = EEPROM error 29 = General hardware error 2A = Transponder receiver Mode-C/S/ADS-B unserviceable 2B = EEPROM error 2C = GPIO error 31 = GPS communication 32 = Configuration of GPS module 33 = GPS antenna 41 = RF communication 42 = Another FLARM device with the same Radio ID is being received. Alarms are suppressed for the relevant device. 43 = Wrong ICAO 24-bit address or radio ID 51 = Communication 61 = Flash memory 71 = Pressure sensor 81 = Obstacle database (e.g. incorrect file type) 82 = Obstacle database expired. 91 = Flight recorder 93 = Engine-noise recording not possible 94 = Range analyzer

	A1 = Configuration error, e.g. while reading flarmcfg.txt from SD/USB. B1 = Invalid obstacle database license (e.g. wrong serial number) B2 = Invalid IGC feature license B3 = Invalid AUD feature license B4 = Invalid ENL feature license B5 = Invalid RFB feature license B6 = Invalid TIS feature license 100 = Generic error 101 = Flash File System error 110 = Failure updating firmware of external display 120 = Device is operated outside the designated region. The device does not work. F1 = Other
<Message>	Field is omitted if data port version <7 or if `DEVTYPE` = `Flarm04`. String. Maximum 40 ASCII characters. Textual description of the error in English. The field may be empty.

Example:

Requesting a list of all errors:

```
> $PFLAE,R

$PFLAE,A,0,0
$PFLAE,A
```

FLARM is asked on its status and returns that there is no problem.

Example:

Automatically issued error (data port version < 7):

```
$PFLAE,A,2,81*
```

FLARM reports during self-test after startup that there is a problem with the obstacle database (e.g. missing or corrupt) but that FLARM will continue to work with reduced functionality.

Example:

Automatically issued error (data port version >= 7):

```
$PFLAE,A,3,11,Software expiry*
```

FLARM reports error that the software has expired. FLARM will not work, fatal problem.

Example:

Startup sequence on Classic FLARM:

```
FLARM

Hardware v2.00, Software v5.00

Performing Selftest...

o.k. 16 Mbit FLASH memory

o.k. Obstacles

o.k. Logging Init

o.k. RF subsystem

o.k. Pressure subsystem

o.k. UART subsystem

o.k. GPS subsystem connection

$PFLAE,A,0,0
```

FLARM reports successful self-test after startup

8.4 PFLAV – Version information

Syntax:

`PFLAV,<QueryType>,<HwVersion>,<SwVersion>,<ObstVersion>`

Description:

Version information after startup. Allow at least 20 s after power-up. Version information should be passed to the user.

Input / Output:

Bidirectional, i.e. can also be requested

Availability:

Always available, no configuration

Periodicity:

Sent once after startup and completion of self-test and when requested

Parameters:

<QueryType>	R = request FLARM to send version; other parameters should then be omitted A = FLARM sends version (requested and spontaneous)
<HwVersion>	Decimal fixed point with two digits after radix point (dot) and one before. Range: from 0.00 to 9.99.
<SwVersion>	Decimal floating-point value. Maximum two digits before radix point and maximum 4 digits after.
<ObstVersion>	Up to 18 ASCII characters (any character, no special structure); the field is empty when no obstacle database is present.

Example:

```
> $PFLAV,R

$PFLAV,A,2.00,5.00,alps20110221_*
```

FLARM is asked on its versions and returns that it has h/w version 2.00, s/w version 5.00 and an obstacle database named 'alps20110221_'.

Example:

```
$PFLAV,A,2.00,5.00,*
```

FLARM reports that it has h/w version 2.00, s/w version 5.00, but that there is no obstacle database present.

8.5 PFLAR – Reset

Syntax:

```
PFLAR,<Value>
```

Description:

Sends a reset command to FLARM, followed by the reset without any read-back.

Input / Output:

Only sent to FLARM

Availability:

Available when on ground, no configuration

Periodicity:

Not applicable

Parameters:

<Value>	Decimal integer value. One of the following values: 0 = Reboot the device, all settings will be retained 33 = Shut down the device into power save mode (except when `DEVTYPE = Flarm04`). Shuts down the FLARM hardware. FLARM is then not operational. To come back to normal operation, use `$PFLAR,0` or `$PFLAR,99`. 99 = Reboot the device, all settings will reset to default values, user configuration is lost

Example:

```
$PFLAR,0
```

FLARM is asked to reboot and reboots.

8.6 GPRMC – NMEA minimum recommended GPS navigation data

Syntax et al.:

See NMEA-spec

Description:

Recommended minimum data. See the official NMEA 0183 specification. Currently, FLARM does not deliver magnetic variation. Note that the time is UTC, not GPS time.

Input / Output:

Only sent by FLARM

Availability:

Depending on configuration: `PFLAC,,NMEAOUT`

Periodicity:

Sent once per second

8.7 GPGGA – NMEA GPS 3D-fix data

Syntax et al.:

See NMEA-spec

Description:

GPS fix data. See the official NMEA 0183 specification. Geoid separation (undulation) and MSL altitude are calculated by the GPS, not measured by the pressure transducer. Note that the time is UTC, not GPS time.

Input / Output:

Only sent by FLARM

Availability:

Depending on configuration: `PFLAC,,NMEAOUT`

Periodicity:

Sent once per second

8.8 GPGSA – NMEA active satellites and DOP

Syntax et al.:

See NMEA-spec

Description:

Active satellites and DOP, see the official NMEA 0183 specification.

Input / Output:

Only sent by FLARM

Availability:

Depending on configuration: `PFLAC,,NMEAOUT`

Periodicity:

On Classic FLARM-based devices, the sentence is sent only on change of satellite configuration. On PowerFLARM-based devices, the sentence is sent every second.

8.9 GPTXT – NMEA text data (ignore)

Syntax et al.:

See NMEA-spec

Description:

Other text coming from GPS or CPU. To be ignored.

Input / Output:

Only sent by FLARM

Availability:

Depending on configuration: `PFLAC,,NMEAOUT`

Periodicity:

Sent only when required

8.10 PGRMZ – Garmin's barometric altitude

Syntax:

Treat the following three versions as identical although FLARM currently only delivers the last one:

```
PGRMZ,<Value>,F,3

PGRMZ,<Value>,F

PGRMZ,<Value>,F,2
```

Description:

Gives the barometric altitude in feet (1 ft = 0.3028 m) and can be negative.

Input / Output:

Only sent by FLARM devices with a pressure sensor

Availability:

Only on devices with a pressure sensor.

Depending on configuration: `PFLAC,,NMEAOUT` (treated as a FLARM proprietary sentence).

Periodicity:

Sent once per second. The sentence is not delivered when no pressure sensor is present.

8.11 PFLAS – Debugging information

Syntax:

```
> PFLAS,R
```

Description:

Request debugging information in human-readable form. Answer consists of multiple lines of internal variables and other information. Do not parse this

information as the structure and content are subject to frequent changes. Not available when `DEVTYPE = Flarm04`.

Input / Output:

Only sent to FLARM

Availability:

Available when on ground

Periodicity:

Not applicable

Parameters:

None

Example Classic FLARM-IGC06:

```
www.flarm.com
HW Flarm-IGC06, #0811810354, 0xDD822F, SW v6.01/1.23 Build 7571
libc 1.8.0
2015/03/19, 16:22:56
Stored configuration settings:
-----------------------------------------
Serial# / IGC#          0811810354 / 123
ID          2     14516783  0xDD822F

Voltage                       12.4V/ 12.4V
Err/Uptime        1/   286
Stack start=0x0d20, size=992, max used=625, unused=367, top-SP=426
Pressure/Temp                 969.2/28.3C
OSCCAL/GPS conf                     187/3
Frequency / 0TX                     100/1
Debug/lastGPS                       0/  0
Volume                                3
Aircraft Type                       0x1
Privacy Flag                          0
No Track Flag                         0
NMEA out/in                         1/0
Baudrate/Range              19200/ 3000m
FFS/in flight threshold           0/2m/s
FLASH FS status/entries             1/1
Obstacle status                       3
  DB name
  creation date
Logging config/status/int           1/1/4

-----------------------------------------
```

FLARM is asked to give debugging information and does so.

Example PowerFLARM Core Pure:

```
(c) 2011 FLARM Technology GmbH - All Rights Reserved.
Vendor: FLARM Technology GmbH (R), Name: Core
Serial: FLAPFD11E-002217, IGC: 9ON
HW 1.1, SW 6.00, Build 7558, Date 12.03.2015
Uptime: 0 days, 00:00:54
Date/Time (UTC): 2015/03/19, 16:23:32

System Configuration:
  freq                = 100               (Selected frequency)
  cflags              = 0                 (CFLAGS)
  ui                  = 0                 (UI)
  thre                = 0                 (Inflight Threshold)
  range               = 65535             (Range)

  vrange              = 500               (Vertical Range)
  aznradius           = -1                (Alert zone radius)
  aznbottom           = -1000             (Alert zone bottom)
  azntop              = 4000              (Alert zone top)
  aznoffset           = 0                 (Alert zone offset)
  aznbearing          = 0                 (Alert zone bearing)
  azntype             = 41                (Alert zone obstacle type)
  aznfrom             = PERM              (Alert zone active from)
  aznuntil            = OFF               (Alert zone active until)

  logint              = 4                 (Logging interval)
  vol                 = 3                 (Buzzer volume)

  pilot               = undefined         (Pilot)
  copil               = undefined         (Copilot)
  gliderid            = undefined         (Aircraft callsign)
  glidertype          = undefined         (Aircraft type)
  compid              = undefined         (Comp. sig.)
  compclass           = undefined         (Comp. class)

  priv                = 0                 (Stealth Mode)
  notrack             = 0                 (No track flag)
  acft                = 12                (Aircraft type)

  nmeaout             = 1                 (NMEA Output)
  nmeaout1            = 1                 (NMEA Output 1)
  nmeaout2            = 61                (NMEA Output 2)
  baud                = 2                 (Baudrate)
  baud1               = 2                 (Baudrate 1)
  baud2               = 5                 (Baudrate 2)

  ID                  = FFFFFF            (Radio ID)

  pcasrange           = 7408              (PCAS Range)
  adsbrange           = 65535             (ADSB Range)
  pcasvrange          = 610               (PCAS Vertical Range)
  adsbvrange          = 65535             (ADSB Vertical Range)
  xpdr                = 0                 (Transponder)
  rssi_threshold      = -1                (Mode C suppression threshold)
  rssi_bandwidth      = -1                (Mode C suppression bandwidth)
  modec               = 1                 (Mode C processing enabled)
```

```
modesalt          = 1                    (Use Mode S altitude)
ownmodec          = 0                    (Own mode C suppression method)
pcaspflau         = 0                    (Output PCAS targets as $PFLAU)
pcaspflau1        = 0                    (Output PCAS targets as $PFLAU on 1)
pcaspflau2        = 0                    (Output PCAS targets as $PFLAU on 2)
pcasbeep          = 1                    (Beep on PCAS alarms)
brightness        = 100                  (LCD Brightness)
pcascalibration   = 30                   (PCAS calibration)
batterytype       = 0                    (Battery Type)
audioout          = 0                    (Audio Out Enabled)
audiovolume       = 100                  (Audio Volume)

swver             = 6.00                 (Software version)
flarmver          = 1.20                 (FLARM software version)
hwrev             = 1.1                  (Hardware revision)
devtype           = PowerFLARM-Core      (Device type)
obstname          = alps_20150115_       (Obstacle database name)
obstdate          = 23.01.2015           (Obstacle database date)
igcser            = 9ON                  (IGC Serial)
cap               = OBST;RFB;TIS;IGC;ENL;AZN;DLED;USBH;DP2;AUD  (Device capabilities)
region            = EUR                  (Device region)
build             = 7558                 (Build designator)
radioid           = 2,DF08A9             (Radio ID with type)
```

8.12 PFLAQ – Operations progress information

Syntax:

`PFLAQ,<Operation>,<Info>,<Progress>`

Description:

Progress information for operations which take significant time. Not all values between 0 and 100 will occur. An operation may terminate prematurely. In this case, a `$PFLAE` will usually be output.

Input / Output:

Sent by FLARM only

Availability:

Available

Parameters:

<Operation>	String. Maximum 10 ASCII characters identifying operation in progress: IGC = IGC files download FW = Firmware update (only PowerFLARM-based devices) OBST = Obstacle database update DUMP = Diagnostic dump RESTORE = Restore file system (only PowerFLARM-based devices) SCAN = Internal consistency check
<Info>	String. Complementary info, e.g. the file name currently being processed. Only sent from PowerFLARM-based devices but may be empty. In Classic FLARM-based devices, this field is omitted.
<Progress>	Decimal integer value. Range: from 0 to 100. Indicates the percentage of completion.

Example PowerFLARM:

```
$PFLAQ,OBST,,10*
```

FLARM is reading obstacle license.

Example PowerFLARM:

```
$PFLAQ,IGC,2A8GJ7K1.IGC,55*

$PFLAQ,IGC,2A8GJ7K1.IGC,65*
```

FLARM is saving IGC file on storage medium.

Example on Classic FLARM:

```
$PFLAQ,IGC,25*

$PFLAQ,IGC,60*
```

FLARM is asked to save IGC files to attached medium storage. The device reads out IGC files indicating progress and finally success.

8.13 PFLAO – Alert Zone information

Syntax:

`PFLAO,<AlarmLevel>,<Inside>,<Latitude>,<Longitude>,<Radius>,`
`<Bottom>,<Top>,<ActivityLimit>,<ID>,<ID-Type>,<ZoneType>`

Description:

A vertical cylinder representing an active Alert Zone. For more information, see Alert Zone Specification document.

Input / Output:

Only sent by FLARM

Availability:

Depending on configuration: `PFLAC,,NMEAOUT` and `PFLAC,,BAUD` (only when baud rate >= 19.2kB).

Available only when data port version is >= 7.

Not available when `DEVTYPE` = `Flarm04`.

Periodicity:

Sent, when applicable, once per second. Can be sent several times per second with information on several (but not always all) surrounding Alert Zones.

Parameters:

<AlarmLevel>	See `$PFLAA`. For Alert Zones, the alarm level is 1 upon entry into the zone and it is only active every 16 seconds for 4 seconds while inside. Otherwise, the alarm level remains zero even while flying inside the zone.
<Inside>	Decimal integer value. Range: from 0 to 1. This value is 1 if the zone is active and we are within the horizontal and vertical limits, zero otherwise.

<Latitude>	Decimal integer value. Range: from -900000000 to 900000000. Latitude of the center of the cylinder in WGS84, degrees times 10^7.
<Longitude>	Decimal integer value. Range: from -1800000000 to 1800000000. Longitude of the center of the cylinder in WGS84, degrees times 10^7.
<Radius>	Decimal integer value. Range: from 0 to 2000. Radius of the cylinder in meters.
<Bottom>	Decimal integer value. Range: from -1000 to 6000. Bottom of the cylinder in meters above the WGS84 ellipsoid (not above MSL).
<Top>	Decimal integer value. Range: from 0 to 6000. Top of the cylinder in meters above the WGS84 ellipsoid (not above MSL).
<ActivityLimit>	Decimal integer value. Range: from 0 to 4294967295. End of activity in seconds since 00:00 Jan 1, 1970, UTC. A value of 0 indicates a zone without a set end time.
<ID>	See `$PFLAA`
<ID-Type>	See `$PFLAA`

<ZoneType>	Hexadecimal value. Range: 10 to FF. Zone types: 41 = Skydiver drop zone 42 = Aerodrome traffic zone 43 = Military firing area 44 = Kite flying zone 45 = Winch launching area 46 = RC flying area 47 = UAS flying area 48 = Aerobatic box 7E = Generic danger area 7F = Generic prohibited area Any other value from 10 to FF, inclusive, means an Alert Zone alarm that has been specified in the latest version of this document. Such Alert Zones should be treated as "Other" Alert Zone.

Example:

```
$PFLAO,1,1,471122335,85577812,2000,100,4550,1432832400,DF4738,2,41*
```

A skydiver drop zone centered at latitude 47.1122335 degrees North, longitude 8.5577812 degrees East, radius 2 km, topping at 4550 m above WGS84 with a FLARM-chosen ID of DF4738. We are currently inside the zone and the corresponding alarm level is 1. The zone will be active until 28 May 2015 @ 17:00:00.

8.14 PFLAI – IGC files readout or trigger an IGC pilot's event

Syntax:

```
PFLAI,<Value>
```

Description:

IGC files readout or trigger an IGC pilot event.

Input / Output:

Can only be requested

Availability:

Available when on ground, no configuration.

Not available on Classic FLARM-based devices.

Periodicity:

Sent when requested

Parameters:

<Value>	One of the following strings: IGCREADOUT = If a storage medium is connected and the device is not currently in flight (according to IGC criteria), reads out all flight's records as IGC files. PILOTEVENT = Trigger an IGC pilot's event.

Example:

```
$PFLAI,IGCREADOUT*
```

```
$PFLAI,IGCREADOUT,ERROR,<error>*
```

FLARM is asked to save IGC files to attached medium storage but it does not succeed. <error> is either IO (problem with storage medium) or INFLIGHT (during flight).

Example:

```
$PFLAI,IGCREADOUT*

$PFLAQ,IGC,,*

[...]

$PFLAI,IGCREADOUT,OK*
```

FLARM is asked to save IGC files to attached medium storage. The device reads out IGC files indicating progress and finally success.

Example:

```
$PFLAI,PILOTEVENT*

$PFLAI,PILOTEVENT,OK*
```

FLARM is asked to trigger pilot event and acknowledges that pilot event has been triggered. IGC logging will continue at 1s interval for 30 seconds.

8.15 PFLAC – Device configuration

Syntax:

```
PFLAC,<QueryType>,<ConfigurationItem>,<Value>
```

Description:

Configuration read-out information and settings. Settings cannot be changed when the device is moving to prevent misconfiguration during flight, unless otherwise stated. Applications should verify the acknowledge sentence of any configuration change. Settings are stored by FLARM and are reloaded at power-up, unless otherwise stated. Default values are underlined. Factory preset values can differ from default values. Note that other devices might listen to FLARM as well, therefore only send the minimum required as you might configure settings necessary for other applications. If the command is not understood or the parameters are out of range, FLARM answers with `PFLAC,A,ERROR`

Input / Output:

Bidirectional, i.e. can also be requested

Availability:

Available when on ground, no configuration

Periodicity:

Sent when requested

Parameters:

<QueryType>	R = request to send content of <ConfigurationItem>; parameter <Value> should then be omitted S = request to set <ConfigurationItem> to <Value> A = FLARM answers request or setting with current content of <ConfigurationItem>
<ConfigurationItem>	See Configuration Specification document.
<Value>	See Configuration Specification document.

Example:

```
$PFLAC,HELLO,GLIDER_PILOTS

$PFLAC,A,ERROR*
```

FLARM is asked a configuration it does not understand and returns an error.

8.16 PFLAJ – Flight and IGC recording state information

Syntax:

```
$PFLAJ,<QueryType>,<FlightState>,<FlightRecorderState>,[<TisbAdsrClientStatus>]
```

Description:

Status information for flight, IGC recording states, and TIS-B/ADS-R client status. The message is automatically sent whenever one of the indicated values changes. The status can also be queried at any time, e.g. by clients that connect while the device is already running.

Input / Output:

Bidirectional, i.e. can also be requested

Availability:

Available only when data port version is >= 8.

Not available on Classic FLARM-based devices.

Periodicity:

Sent whenever a state change occurs, or on request.

Parameters:

<QueryType>	R = Request to get status; other parameters should be omitted A = FLARM answer to request or status update.
<FlightState>	Flight state: 0 = On ground 1 = In flight
<FlightRecorderState>	IGC Flight recorder state 0 = OFF 1 = Recording 2 = Barometric recording only (no GPS)
<TisbAdsrClientStatus>	Empty if not available. Only relevant for aircraft flying in the USA. Determines whether ownship is being serviced as a TIS-B/ADS-R client. 0 = TIS-B/ADS-R unavailable 1 = Receiving TIS-B/ADS-R service See DO-317B Appendix H for details.

Example:

```
$PFLAJ,A,1,1,0*20
```

FLARM reports the device is in flight, flight recording is active, and TIS-B/ADS-R is unavailable.

```
> $PFLAJ,R
```

```
$PFLAJ,A,1,1,0*20
```

Answer to query as above.

8.17 PFLAN – Continuous radio range estimation

The device continuously records range information during operation when new aircraft appear and disappear. This data is aggregated into RF range statistics that can be read by data port consumers. The statistics provide a similar set of information as the web-based range analyzer[4], using IGC records. Compared to the latter, continuous range analysis is not constrained by the data volume of the IGC file, thus it provides a clearer picture after less flight time.

Range information is stored between flights, so it is straightforward to retrieve overall statistics for multiple flights. The continuous estimator can be deliberately reset (`$PFLAN,S,RESET`), e.g. when an installation change is to be validated.

Syntax:

```
$PFLAN,<QueryType>,<ConfigurationItem>
```

Description:

Request the device to output or reset the range statistics.

Input / Output:

Bidirectional, i.e. can also be requested

Availability:

Not available on Classic FLARM-based devices

Periodicity:

Sent when requested

Parameters:

<QueryType>	R = Request to get range statistics S = Request to reset range statistics A = FLARM answer to request

[4] https://flarm.com/support/tools-software/flarm-range-analyzer/

<ConfigurationItem>	RANGE = Obtain the range statistics RESET = Reset the range statistics (only with <QueryType> = S).

Answer:

Several different reply types are returned for the various statistical values, as defined below.

$PFLAN,A,RANGE,<StatisticType>,<Channel>,<Value1>,<Value2>,…,<ValueN>

<StatisticType>	The type of statistic being returned in the value fields. Currently, the following types are returned. New types may be added in the future. For the horizontal sector-based values, the sectors (value fields) start at the longitudinal axis (0° relative bearing) and move in a clockwise direction seen from above. Currently, there are 20 sectors of 18° each, but this may change in the future. **RFTOP** = mean range in meters per sector and channel. **RFCNT** = number of data points used to compute the mean range for each sector and channel. **RFDEV** = standard deviation of the data points in each sector per channel. **Note:** Each sector should be checked to have a sufficiently large RFCNT value to be significant.
<Channel>	Name of the radio channel. Can be A or B.
<Value1>,<Value2>,…,<ValueN>	Value or set of values for the given <StatisticType>. Normally per sector as defined under <StatisticType>. An empty field implies that the value could not be reliably computed for the relevant sector.

$PFLAN,A,RANGE,STATS,<NumberOfPointsTop>

<NumberOfPointsTop>	Number of points used to compute the average range for all horizontal sectors and both channels.

```
$PFLAN,A,RANGE,TIMESPAN,<TimestampStart>,<TimestampEnd>
```

<TimestampStart>	Timestamp, seconds since Jan 1, 1970 00:00:00 (UNIX Epoch), UTC Timestamp of the first RF packet used to compute range statistics.
<TimestampEnd>	Timestamp of the last RF packet used to compute range statistics.

Example:

```
> $PFLAN,R,RANGE

$PFLAN,A,RANGE,RFTOP,A,5600,4800,3600,2400,1200,1200,…*

$PFLAN,A,RANGE,RFCNT,A,54,121,65,41,87,98,…*

$PFLAN,A,RANGE,RFDEV,A,1200,900,1450,700,1100,1400,…*

$PFLAN,A,RANGE,RFTOP,B,4800,3600,2400,1200,1200,1200,…*

$PFLAN,A,RANGE,RFCNT,B,51,95,27,49,42,111,…*

$PFLAN,A,RANGE,RFDEV,B,1800,1100,1500,900,1200,1300,…*

$PFLAN,A,RANGE,STATS,5000*

$PFLAN,A,RANGE,TIMESPAN,1562000000,1563000000*

$PFLAN,A,RANGE*


> $PFLAN,S,RESET

$PFLAN,A,RESET*
```

8.18 PFLAF – Simulated traffic and alarms

Syntax:

```
$PFLAF,<QueryType>,<ScenarioNumber>
```

Description:

Run a predefined scenario, outputting simulated data on the data ports. This can be used to verify an installation and to test displays. The different scenarios

simulate traffic, collision warnings, as well as other alerts. Visual and aural warnings are generated by the connected system components as in a real situation. Ownship position is also simulated for the duration of the scenario. Configuration settings (e.g. range) have no effect on the scenarios.

The currently active scenario, if any, can also be requested.

The scenario will run for a predetermined amount of time and various synthetic data will be injected in the following sentences:

`$PFLAA`, `$PFLAO`, `$PFLAU`, `$PGRMZ`, `$GPGGA`, and `$GPRMC`.

Only one scenario can be active at any time. The simulation is not available while in flight.

Input / Output:

Input only.

Availability:

Not available on Classic FLARM-based devices.

Parameters:

<QueryType>	S = Request to run a scenario R = Request the currently running scenario A = FLARM answer to request
<ScenarioNumber>	An integer representing the scenario to run (see table below). Only for `QueryType` "S".

Available scenarios:

1	A single **FLARM-equipped aircraft** with ID 123456 in a collision trajectory with 0° relative bearing. Starts far away with no warning and goes through all alarm levels until collision. Lasts 30 seconds.
2	A single **ADS-B-equipped aircraft** with ID 123456 in a collision trajectory with 270° relative bearing. Starts far away with no warning and goes through all alarm levels until collision. Lasts 30 seconds.
3	A single **non-directional aircraft** (equipped with a Mode-S transponder) with ID 123456 in a collision trajectory. Starts far away with no warning and goes through all alarm levels until collision. Lasts 30 seconds.

4	A **fixed obstacle** from an installed obstacle database with a valid license with ID 123456. Starts far away with no warning and goes through all alarm levels until collision. Lasts 30 seconds.
5	An **Alert Zone**, i.e. a dynamic airspace in the form of a cylinder where special vigilance is required (e.g. active skydiving activity) with ID 123456. Ownship flies towards the zone for 4 seconds before entering, crosses for 24 seconds, and continues on the other side for 2 seconds. Lasts 30 seconds.
6	A **mixed scenario**, where multiple traffic types and objects are combined. A FLARM-equipped aircraft (ID 123456), an ADS-B-equipped aircraft (ID 123457), a non-directional aircraft (ID 123458), a fixed obstacle (ID 123459), and an alert zone (ID 123460) are all in the vicinity, none of which are generating a warning. Lasts 30 seconds.

Answer:

In the nominal case, where the scenario was successfully started:

```
$PFLAF,A,<ScenarioNumber>
```

<ScenarioNumber>	The scenario number that was started.

In case the scenario could not be started:

```
$PFLAF,A,ERROR,<ErrorType>
```

<ErrorType>	The error that occurred. The scenario was not started. COMMAND = invalid command received UNKNOWNSCENARIO = the scenario number does not match any existing scenario INPROGRESS = a scenario is already running INFLIGHT = the device is in flight

Example:

```
> $PFLAF,S,1

$PFLAF,A,1*

> $PFLAF,R
```

`$PFLAF,A,*` (in case no scenario is running)

`$PFLAF,A,1*` (if scenario 1 is running)

8.19 PFLAL – Debug message

Syntax:

`$PFLAL,<DebugMessage>`

Description:

Debug message that is periodically emitted. The content of the debug message is for debug purposes only, and the content and periodicity can change between firmware versions.

System integrators are asked to store this message on some externally accessible storage medium. It can e.g. be included in IGC files. Having access to these messages aids in resolving problems in customer installations.

Input / Output:

Output only.

Availability:

Not available on Classic FLARM-based devices.

Example:

`$PFLAL,12224002NbWFCFcMN?lknsqrbser;NAKELu[*`

`$PFLAL,122242GPS 7 39*`