

Open Drone ID

Message Specification

Draft Specification Version 0.61.1

Protocol version 0

November 13, 2018

Contents

1	Introduction.....	5
2	Related Documents	5
3	Block Message Definitions.....	5
	General	5
	Message Header.....	5
	Basic ID Message	6
	Location Message.....	7
	Authentication Message	8
	Self ID Message	8
	Operator Message.....	9
	Enumerated Field Definitions.....	10
4	JSON Representation of Messages.....	11
	Basic ID (Message Type 0) JSON Representation.....	11
	Location (Message Type 1) JSON Representation.....	11
	Authentication (Message Type 2) JSON Representation	11
	Self ID (Message Type 3) JSON Representation	12
	Operator (Message Type 4) JSON Representation.....	12
5	Compliance and Interoperability.....	12

Update History

Version	Date	Changes	Author
0.54	3/1/2018	Started Change Control (initial version baseline)	G. Cox, Jan S.
0.55	4/3/2018	<ul style="list-style-type: none"> * Changed all Messages Modes S field from 28 bit, to 32 bit UniqueID * Base ID Message: Moved Sub-Type up to after under Drone Type * Location Message: Changed final reserved field to 4 bits since the Unique ID went from 24 to 32 bits. 	G. Cox
0.57	4/24/2018	Added Additional Packet Detail Tables, Moved Message Definitions to end	G. Cox
0.58	4/25/2018	Location Message: Moved "Reserved" up to after "Status"	G. Cox
0.59.0	8/15/2018	<p>**Major field updates**</p> <p>* Note: We are not incrementing protocol version until this specification goes from draft to final and after a change is made to the final version.</p> <ol style="list-style-type: none"> 1. General: <ol style="list-style-type: none"> 1.1. Format of Messages documentation changed to show Offset/Len in 1st 2 columns and more clearly display bitfields 1.2. Clarified that multi-byte IDs shall be expressed in Network Byte Order, and multi-byte numerical values shall be expressed in little endian. 1.3. Added DRAFT watermark 2. Basic ID Message: <ol style="list-style-type: none"> 2.1. Shortened Make/Model to 8 bytes and provided a structured format to supply make/model. 2.2. Added "Flags" byte of which bit 0 represents Weight Multiplier 2.3. Increased Gross weight from 14 to 16 bits. This (and moving multiplier to Flag byte) is to divide the message fields along even 8bit boundaries so that it may more easily be parsed/processed. 3. Location Message: (lots of refactoring to split fields and offsets on even byte boundaries) <ol style="list-style-type: none"> 3.1. Move Multipliers (V-Speed, H-Speed into common byte with Status) 3.2. Increased speed fields from 6 to 8 bits 3.3. Increased Lat/Lon fields from 26 to 32 bits. Not only does this provide 10⁷ precision, but it also allows for an easy/common 32bit Int processing. 3.4. Increased MSL and AGL Altitude fields to 16bit and removed multipliers. 3.5. Added Horizontal/Vertical precision confidence values 3.6. Removed all references to "heading" fields since this can be derived from the speed vector components. 3.7. Removed operator location (this is how we made all the space) and moved to a separate dedicated Operator Message. 3.8. Allocated 4 bytes as reserved 3.9. Moved enumerated status list to "Enumerated Field Definitions" 4. Added Operator Message that includes lat, lon, source. 	G. Cox
0.60.0	8/23/2018	<ol style="list-style-type: none"> 1. Extracted Data to a stand-alone document 2. Corrected Location Flags to be NS, EW Multipliers (removed Vertical Mult). 	G. Cox
0.61.0	11/9/2018	<ol style="list-style-type: none"> 1. Major Updates to consensus agreed field list 	G. Cox

		<ol style="list-style-type: none"> 2. Header: Removed UniqueID (moved to BasicID). Unique HW address will need to be used as a “session key”. Message size has gone from 21 to 25 bytes. 3. Basic ID: Removed: Flags, Weight, Reg ID, Make/Model Added: Single Byte of ID Type + Drone Type Added: General 24 Byte ID Field (of ID type) 4. Location Message: Replaced AGL with Geodetic Alt Added Age of Data (in 10ms) 5. Authentication Message: Added Auth Type/Page Number Auth Type will drive Format of Message 6. Self ID Message: Increased to 25 bytes (see item 2) 7. Added Formation info to Operator message 8. Added Horizontal and Vertical Accuracies to Enumerated Field Definitions 	
0.61.1	11/12/2018	<p>Corrected Type for Location/Vector Message Reserved Field size (should say “8 bytes reserved” instead of 3)</p> <p>Removed UniqueID from JSON messages</p>	G. Cox

1 Introduction

On December 19th 2017 the Federal Aviation Administration (FAA) published the UAS Remote Tracking & ID ARC Report¹ to update the public about the latest results from the Aviation Rulemaking Committee (ARC) chartered by the FAA.

This specification is designed to meet such needs expressed in the ARC Report.

This document is part of a group of documents to meet the total solution requirements of Remote ID.

This document is currently in *DRAFT* and is under a standardization process within the ASTM F38 Remote ID Workgroup. The outcome of this collaboration will most certainly result in many changes as a part of this process.

2 Related Documents

Open Drone ID - Bluetooth Broadcast Specification

Open Drone ID - WiFi Broadcast Specification

Open Drone ID - Network Access Specification

3 Block Message Definitions

General

The “Block” messages are intended to be packed into lightweight direct broadcast packets that are designed to fit into WiFi or Bluetooth “Beacon Advertisements”.

Each 26 byte message shall begin with a 5 byte header followed by 21 bytes of data. Numerical (such as the Unique ID) shall be expressed in Network Byte Order (MSB First). Other numerical values that represent a magnitude (such as Lat, Lon, Weight, Alt, etc.) shall be expressed as “little endian” (LSB First).

Message Header

The message header includes the Message Type and Protocol Version and must be sent in each message.

Header (1 byte)		Message (25 bytes)
Message Type (4bits)	Protocol Version (4bits)	Message Fields based on Message Type
0x1-0xF	0x0	< Message Data >

Figure 1 - Message Format

¹ <https://www.faa.gov/news/updates/?newsId=89404>

Basic ID Message

Message Type: 0x0, Static

The BasicID message includes the Drone Type and the Unique ID. This Unique ID would default to the Manufacturers Serial number expressed in the ANSI/CTA-2063 Physical Serial Number format. If a Civil Aviation Authority (CAA) has issued ID for the UAV, then that ID or a UTM Assigned ID must be sent in this message. It is sent at a low frequency. This message is **mandatory**.

Offset (Byte)	Len (Bytes)	Data Field	Details	Limitations	Example
1	1	ID Type, Drone Type	<div style="text-align: right; margin-right: 20px;">[0 0 0 0 0 0 0]</div> <div style="display: flex; align-items: center;"> <div style="border-left: 1px solid black; width: 100px; height: 100px; margin-right: 10px;"></div> <div> <u>ID Type</u> 0: None 1: Serial Number (ANSI/CTA-2063) 2: CAA Assigned ID 3: UTM Assigned ID </div> </div> <div style="display: flex; align-items: center; margin-top: 10px;"> <div style="border-left: 1px solid black; width: 100px; height: 100px; margin-right: 10px;"></div> <div> <u>UAV Type</u> VTOL, fixed wing, hybrid, etc. (See <i>Figure 6</i> below for more details.) </div> </div>	Up to 15 ID types Up to 15 Drone types	
2	24	Drone ID	Drone ID within the format of ID Type	Max. 24 Bytes	

Figure 2 - Basic ID Message

Location Message

Message Type: 0x1, Dynamic

The Location/Vector message provides the location, altitude, direction and speed of the drone.

This message is **mandatory**.

Note: Speed is expressed in North(+)/South(-) and East(+)/West(-) components. This will easily allow for computing bearing and resultant speed. If either speed component magnitude is over 127 m/s, the corresponding multiplier bit must be set and the speed component shall be expressed 10s of units.

Offset (Byte)	Len (Bytes)	Data Field	Details	Limitations	Example
1	1	Status, Flags	Bits [7..0] [0 0 0 0] [0 0 0 0] Status: _____ Optional: 0 = Undeclared Flags: _____ Reserved _____ NS-Speed Multiplier _____ EW-Speed Multiplier _____ 0 = x1, 1 = x10	0..15 statuses Multiplier enables speeds up to +- 1270. Only use when speed exceeds 127 m/s.	
2	1	Speed North/South	Speed in m/s (+N, -S)	Up to +-1270 m/s	5 m/s
3	1	Speed East/West	Speed in m/s (+E, -W) *Note, Heading/Track is derived as a vector of these 2 magnitudes.	Up to +-1270 m/s	2 m/s
4	1	Vertical Speed	Vertical Speed m/s (+ up, - down)	Up to +-127 m/s (25k ft/min)	0 m/s
5	4	Latitude	Latitude of drone	Int signed deg*10 ⁷	-48123987
9	4	Longitude	Longitude of drone	Int signed deg*10 ⁷ (11mm precision)	11989298
13	2	Altitude (MSL)	Pressure Altitude (Ref 29.92inhg)	+32767m (107503ft) 16bit Signed Int	
15	2	Geodetic Altitude	WGS84-HAE	+32767m (107503ft) 16bit Signed Int	65m
17	1	Horizontal/Vertical Precision	Bits [7..0] [0 0 0 0] [0 0 0 0] Vertical _____ Horizontal _____ Vertical: Extended ADS-B GVA Horizontal: Extended ADS-B NACp	See Accuracy enumeration tables below.	
18	1	Data Age	Max Potential Age of data * 10ms	0-255: up to 2.5s	
19	8	Reserved	8 bytes reserved for future use		

Figure 3 - Location Message

Authentication Message

Message Type: 0x2, Static

An Authentication message can provide an authentication token to prove the authenticity of the identity of the aircraft sending the messages. There is still some work to be done to get the exact contents down of each authentication type.

Offset (Byte)	Length (Bytes)	Data Field	Details	Limitations	Example
1	8	AuthType, Data Page	Bits [7..0] [0 0 0 0] [0 0 0 0] Auth Type ——— [0 0 0 0] 0: None 1: CA Signed 2: SHA-1 Secret 3: HW Addr Data Page ——— [0 0 0 0] Start at Page 0.	Up to 15 Types, Up to 15 Pages	
2	24 Bytes	Authentication Data	Opaque Authentication Data Type 0: No Data Type 1: 64 Byte Public Key 32 Byte Auth Token (4 pages) Type 2: HMAC_SHA1(K ID+TimeStamp) Clear(ID+TimeStamp) Type 3: HW Addr/MAC Address	Up to 15 Pages * 24 Bytes	

Figure 4 - Authentication Message

Self ID Message

Message Type: 0x3, Static

The Self-ID message is an opportunity for the Drone Pilot to (**optionally**) declare their identity and purpose of the flight. This can serve the purpose of putting people at ease if concerns exist as to why a drone is flying in a particular area. For Example: A Realtor may want to declare that they are taking photos of a client's house to put the neighbors at ease.

This message is **optional**

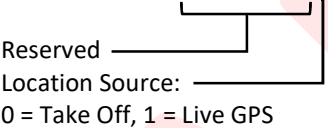
Offset (Byte)	Length (Bytes)	Data Field	Details	Limitations	Example
1	25	Operation Description	Text of Operator and Purpose	25 Bytes	DronesRus: Survey

Figure 5 - Self ID Message

Operator Message

Message Type: 0x4, Static (slow update, but does change)

The Operator Message represents information about the ground control station (GCS). At this revision, it only contains the location. If the mechanism used for determining the location of the GCS is “takeoff location”, then the operator must remain near (within 20m) the takeoff location and may use the Location Source of 0. Otherwise, if the operator is “roving”, then Live GPS (or other live update mechanism) must be used to ensure the operator location remains accurate. Since this value generally does not change at the same rate of a drone, the update frequency shall be the same rate as static messages. If a group of aircraft is being represented, the number of aircraft and radius of flight area shall be expressed in this message.

Offset (Byte)	Length (Bytes)	Data Field	Details	Limitations	Example
1	4	Latitude	Latitude of operator	Int signed deg*10 ⁷	-48123987
5	4	Longitude	Longitude of operator	Int signed deg*10 ⁷ (11mm precision)	11989298
9	1	Flags	Bits [7..0] [0 0 0 0 0 0 0 0] 		
10	2	Formation Count	Number of Aircraft in Formation (default 1)	Up to 65,000	
12	1	Formation Radius	Radius of Cylindrical area of Formation * 10m	Up to 2.5km	
13	14	Reserved	Reserved for future use	14 bytes	

Enumerated Field Definitions

In the data structures above, some fields are enumerated values. The table below assigns meaning to those enumerations. This can be updated as needed.

Field Name	Details	Length (bits)	Notes
Drone Type	0: None 1: Fixed Wing Powered 2: Rotorcraft/Multirotor 3: LTA (Lighter than Air) Powered 4: LTA Unpowered (Balloon) 5: VTOL 6: Free Fall/Parachute 7: Rocket 8: Glider 9: Other 10-15: Reserved	8	Up to 255 Types
Status	0: Undeclared 1: Ground, 2: Airborne (manual control) 3: Returning Home 4: Automated Mission 5: Emergency 6-15: Reserved	4	Up to 16 Statuses
Horizontal Accuracy	0 ≥ 18.52 km (10NM) Unknown accuracy 1 < 18.52 km (10NM) 2 < 7.408 km (4NM) 3 < 3.704 km (2NM) 4 $< 1\,852$ m (1NM) 5 < 926 m (0.5NM) 6 < 555.6 m (0.3NM) 7 < 185.2 m (0.1NM) 8 < 92.6 m (0.05NM) 9 < 30 m 10 < 10 m 11 < 3 m 12 < 1 m 13-15 Reserved	4	This is the same NACp enumeration from ADS-B. Value 12 was added for a more complete range for UAVs.
Vertical Accuracy	0 >150 m Unknown 1 <150 m 2 <45 m 3 <25 m 4 <10 m 5 <3 m 6 <1 m 7 Reserved	4	This is the same GVA enumeration from ADS-B. Values 4-6 were added for UAVs.

Figure 6 - Enumerated Field Definitions

4 JSON Representation of Messages

When transmitting Open Drone ID messages over a network to a Web Service, a JSON representation shall be required. Below are the JSON example representations of the above messages. *Note that multipliers are not used in these messages.

Basic ID (Message Type 0) JSON Representation

```
{
  "MessageType": 0,
  "Version": <UInt8>,
  "IDType": <UInt8>,
  "DroneType": <UInt8>,
  "DroneID": "string(24)",
}
```

Location (Message Type 1) JSON Representation

```
{
  "MessageType": 1,
  "Version": <UInt8>,
  "Status": <UInt8>, // Optional, Default 0
  "SpeedNS": <Int32 (+-m/s)>,
  "SpeedEW": <Int32 (+-m/s)>,
  "SpeedVertical": <Int32 (+-m/s)>,
  "Latitude": <float>,
  "Longitude": <float>,
  "AltitudeMSL": <Int32 (+-meters)>,
  "AltitudeGeo": <Int32 (+-meters)>,
  "HAccuracy": <Int8>,
  "VAccuracy": <Int8>
}
```

Authentication (Message Type 2) JSON Representation

```
{
  "MessageType": 2,
  "Version": <UInt8>,
  "AuthType": <UInt8>,
  "AuthToken": "string(96)"
}
```

Self ID (Message Type 3) JSON Representation

```
{  
  "MessageType": 2,  
  "Version": UInt8,  
  "Description": string(25)  
}
```

Operator (Message Type 4) JSON Representation

```
{  
  "MessageType": 4,  
  "Version": <UInt8>,  
  "LocationSource": <UInt8>,  
  "Latitude": <float>,  
  "Longitude": <float>,  
  "FormationCount": <UInt16>,  
  "FormationRadius": <UInt16>  
}
```

5 Compliance and Interoperability

As of this version, compliance can be “self-certified” using the following means:

1. Every “shall”, “must” and any other logical directive in this document must be implemented.
(See IETF RFC2119 for adopted definitions of imperatives: <https://www.ietf.org/rfc/rfc2119.txt>)
2. Interoperability shall be verified against “known working” clients for both BLE 4 and Bluetooth 5 Extended Advertising receivers.
3. Hardware/RF/Signal compliance TBD.
4. If a system is not compliant with this spec, then it may not claim, advertise or display references to “Open Drone ID”.