AEOLUS
Communications

This document describes the protocol used to communicate with AEOLUS-SENSE with firmware version 3.8 and above. The same protocol is used to communicate with the older, now discontinued, AEOLUS sensor boxes. Throughout this document we will use the term AEOLUS to refer to both AEOLUS and AEOLUS-SENSE sensor boxes.

ATTENTION: A-EFIS and AEOLUS/AEOLUS-SENSE are not certified aviation instruments. Do not rely on them as your only navigation aid. Failure to comply to this warning may result in property damage, serious injury or death. You assume total responsibility and risk associated with using this application.

Contents

1 Short description of communications **3**

2 AEOLUS WiFi connection **3**

3 Testing the connection **3**

4 Packet and message format for data broadcasted by AEOLUS **4**

4.1 Steps to construct an AEOLUS message 5

4.2 Steps to disassemble an AEOLUS message 5

4.3 CRC calculation 6

5 Coordinate system convention **6**

6 Message IDs **6**

7 Description of individual messages **7**

7.1 AEOLUSINFO 7

7.2 PRESSURE 8

7.3 GYRO 9

7.4 ACCEL 9

7.5 MAGNET 9

7.6 TEMP 10

7.7 GPS 10

7.8 ATTITUDE 11

7.9 COMPASS 12

7.10 WIND 12

7.11 CALIBINFO 13

7.12 DEBUGINFO 13

7.13 LOG 13

7.14 TOAST 13

7.15 COMMAND 13

7.16 SETTIME 14

8 Additional information **14**

1 Short description of communications

2 AEOLUS WiFi connection

AEOLUS creates a wifi network with SSID "AEOLUS_XXXXXXXXX", where X is a unique alphanumeric string (the MAC address of the wireless network card). AEOLUS' wifi uses WiFi Protected Access with Pre-Shared Key (WPA-PSK) with TKIP (Temporal Key Integrity Protocol) for encryption. The pre-shared key (passphrase) is "AEOLUS01".

AEOLUS creates sublet 192.168.1.0 with network mask 255.255.255.0. The IP of AEOLUS itself is 192.168.1.1 and all connected devices are assigned IP addresses in the range 192.168.1.70 - 192.168.1.200.

The broadcast address of the subnet is 192.168.1.255

All communications between AEOLUS and other devices take place on this wifi network.

3 Testing the connection

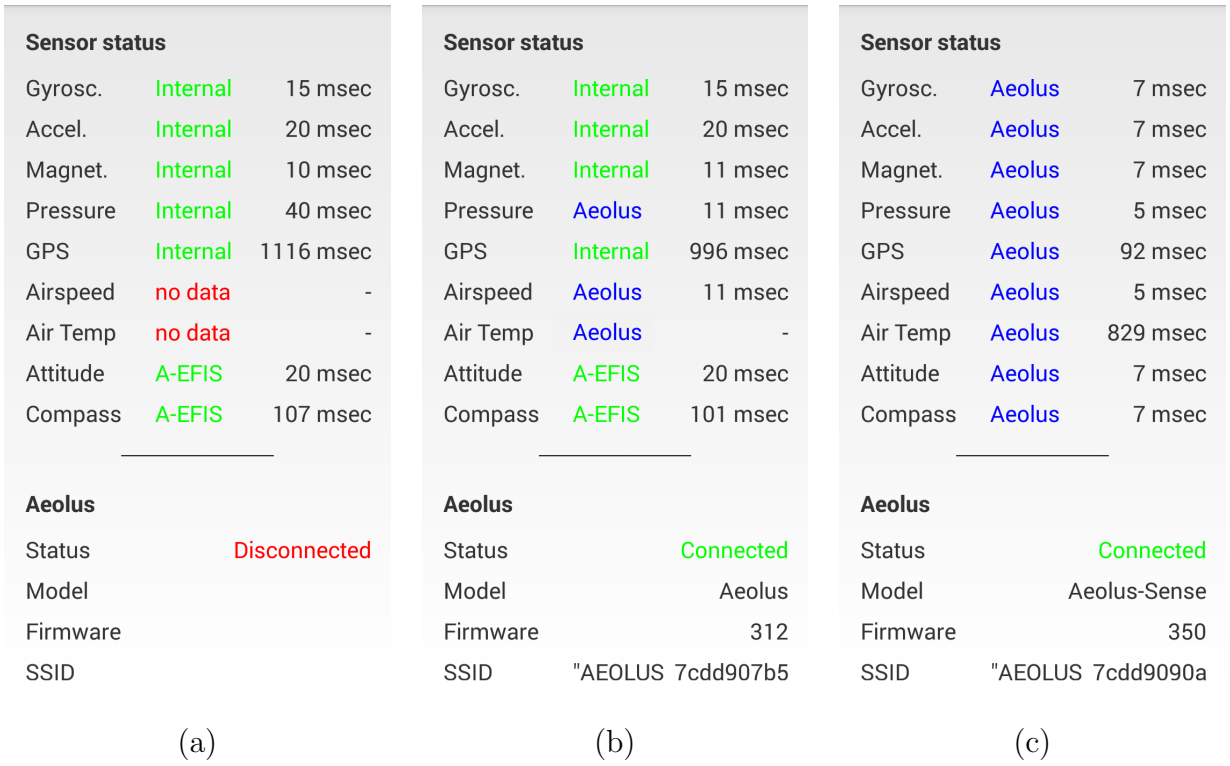
You can test the wifi connection to AEOLUS using an IOS or Android device. For this purpose, you should download the "A-EFIS free' app which can be found on Google Play market and on Apple's app store.

After you start A-EFIS free, swipe right to find the "sensor status" screen (Figure 1) .

The Sensor status screen displays information about the availability of the supported sensors and other sensor-related information. If your smartphone/tablet is connected to an AEOLUS device, you should be able to see something like Fig. 1b or Fig. 1c. Towards the bottom of the screen, you will also find information about AEOLUS model and firmware version.

Each row indicates a different type of information transmitted by AEOLUS. The center column indicates the source of the information and should be "AEOLUS" for information transmitted by AEOLUS. The right column indicates the average interval between two consecutive receptions of this particular type of information.

For sensors that are not available on the sensor box (i.e. in the case of the AEOLUS sensor box) or in case of some failure (communication, power, etc), A-EFIS substitutes the missing sensors with the internal sensors of the mobile device (if available). In this case, the display sensor status screen looks like Fig. 1a.



(a) (b) (c)

Figure 1: Sensor status screen. (a) A-EFIS working with internal sensors only, (b) A-EFIS connected to an AEOLUS device, (c) A-EFIS connected to an AEOLUS-SENSE device.

4 Packet and message format for data broadcasted by AEOLUS

All communications between AEOLUS and other devices take place as UDP packets, broadcasted on the 192.168.1.0 subnet (broadcast address 192.168.1.255) AEOLUS broadcasts data on port 5001 and listens for broadcasted data on port 5002

Each UDP packet may contain one of more AEOLUS "messages". Each message contains a different type of information.

The format of the AEOLUS messages roughly follows the GDL90 format.

Size (bytes)	Description
1	Flag byte (0x7E)
1	message ID
N	message data
2	CRC
1	Flag byte (0x7E)

More specifically, an AEOLUS message comprises:

- A Flag Byte (0x7E) indicating the beginning of the message.

- An one-byte Message ID, which specifies the type of the message that follows. The size of the message also depends on this ID.

- The message data. The length on the message data, N, depends on their type.

- A 16-bit CRC. The least significant byte is first.

- A Flag Byte (0x7E) indicating the end of the message.

Before transmitting, the portion of the message between the two flag bytes is escaped using the escape character 0x7D. That is, wherever a Control-Escape character (0x7D) or a flag byte character (0x7E) is found in-between the two flag bytes, a Control-Escape character (0x7D) is inserted, followed by the original byte, XOR'ed with the value 0x20.

4.1 Steps to construct an AEOLUS message

The steps to construct a message for transmission are as follows:

- Assemble the message comprising the Message ID byte and the message data.

- If necessary, calculate the CRC and append it at the end of the message (with the least significant byte first). **Depending on the contents of the packet, a CRC may not be considered necessary. In this case two zero characters are appended instead of the CRC.**

- Find all the Control-Escape (0x7D) and the Flag-Byte (0x7E) characters within the resulting message (comprising the ID, the data and the CRC) and escape each of them them by inserting an extra Control-Escape (0x7D) character just before and by XOR-ing the original character with 0x20.

- Add two Flag-Byte characters, one at the beginning, and one at the end of the message.

Two or more messages can be sent on the same UDP packet. In these case, as expected, there will be two consecutive Flag-Bytes in-between the two messages: One marking the end of the previous message and another one marking the beginning of the next.

4.2 Steps to disassemble an AEOLUS message

The process to disassemble an AEOLUS UDP packet on the receiving device is the following:

- Locate the first Flag-Byte (0x7E) character.

- Read all characters until the next Flag-Byte character. The stream in-between the two Flag-Bytes is the AEOLUS message.

- Look for all Control-Escape (0x7D) characters within the message and discard each one of them, XOR'ing their next character with 0x20.

- Check if the last two bytes (the CRC) are non-zero. If they are non-zero, calculate the CRC of the remaining message (i.e. the part of the message comprising the ID and the message data only) and check it against the received CRC. If they are zero then the message contains no CRC, so no CRC check is necessary.

4.3 CRC calculation

The CRC used is a 16 bit CRC-CCITT. To accelerate performance a 256-element lookup table can be generated during the initialization of the program. The following code illustrates how to calculate the CRC. In the example below, the function "crcInit()" calculates "crclut", an unsigned 16-bit integer array of 256 elements. at the beginning of the program. The function "crcCompute(unsigned char *, unsigned int)" computes the CRC at runtime using the lookup table "crclut".

```
unsigned short crclut[256]; // 256 x 2 byte

void crcInit() {
  for (unsigned short i=0; i<256; i++) {
      unsigned short crc = (i<<8); // two byte unsigned int
      for (unsigned short ibit = 0; ibit < 8; ibit++)
          crc = (crc << 1) ^ ((crc & 0x8000) ? 0x1021 : 0);
      crclut[i] = crc;
  }
}

unsigned short crcCompute(unsigned char *aData, unsigned int aDataLen) {
  unsigned short crc = 0; // two byte unsigned int
  for (int i=0; i<aDataLen; i++)
      crc = crclut[crc>>8] ^ (crc<<8) ^ aData[i];
  return crc;
}
```

5 Coordinate system convention

Throughout this document and for the communication needs of AEOLUS we assume the NED coordinate system (North, East, Down). That is, the X axis is towards the direction of flight (north), the Y-Axis is towards the right of the pilot (east), and the Z-Axis is downwards (down).

The NED coordinate system is a right-handed system, hence the convention for angular quantities is that of the right-hand rule. More specifically looking towards the axis of rotation, positive rotations are the clockwise rotations.

The gravity vector is (0,0,G) and the acceleration measured by the accelerometers due to gravity is (0,0,-G).

6 Message IDs

The following table contains the list of message IDs sent/received by AEOLUS

ID (decimal)	Name	Size (bytes)	Description
51	AEOLUSINFO	32	Information about AEOLUS version and firmware
61	PRESSURE	32	Pressure sensors readings and airspeed
62	GYRO	32	Gyroscope readings
63	ACCEL	40	Accelerometer readings
64	MAGNET	32	Magnetometer readings
65	TEMP	32	Outside air temperature
66	GPS	32	GPS readings
71	ATTITUDE	52	Attitude information
72	COMPASS	80	Compass information
73	WIND	32	Wind information
75	CALIBINFO	32	Calibration status (internal use)
76	DEBUGINFO	208	Debug information (internal use)
81	LOG	varies	Log (internal use)
82	TOAST	varies	Messages for the user
11	COMMAND	varies	Text commands to AEOLUS (internal use)
12	SETTIME	32	Set AEOLUS time (internal use)

7 Description of individual messages

7.1 AEOLUSINFO

The AEOLUSINFO message contains information about the AEOLUS model and the firmware version. Additionally it contains the mac address if the wifi adapter of the AEOLUS device transmitting the packet.

Byte	Type	Size (bytes)	Content
ID	char	1	51 (decimal)
0	char	1	AEOLUS Model (83:AEOLUS SENSE, 65: AEOLUS)
1	char	1	Firmware version Large (e.g. 3 for v3.8.0)
2	char	1	Firmware version Middle (e.g. 8 for v3.8.0)
3	char	1	Firmware version Small (e.g. 0 for v3.8.0)
4-7			Reserved
8	char	1	AEOLUS MAC Address byte 0
9	char	1	AEOLUS MAC Address byte 1
10	char	1	AEOLUS MAC Address byte 2
11	char	1	AEOLUS MAC Address byte 3
12	char	1	AEOLUS MAC Address byte 4
13	char	1	AEOLUS MAC Address byte 5
14-15			Reserved
16-32			Reserved

7.2 PRESSURE

The PRESSURE message contains the pressures from the Pitot and the static inputs of AEOLUS as well as the current airspeed in m/sec.

Byte	Type	Size (bytes)	Content
ID	char	1	61 (decimal)
0	char	1	Data valid (1:valid, 0:not valid)
1-7			Reserved
8-15	double	8	Timestamp (msec since boot)
16-19	float	4	Static pressure (mbar)
20-23	float	4	Pitot pressure (mbar)
24-27	float	4	Airspeed (m/sec)
28-32			Reserved

7.3 GYRO

The GYRO packet contains the angular speeds are measured by the three gyroscopes of the device.

Byte	Type	Size (bytes)	Content
ID	char	1	62 (decimal)
0	char	1	Data valid (1:valid, 0:not valid)
1-7			Reserved
8-15	double	8	Timestamp (msec since boot)
16-19	float	4	Gyro X (rad/sec)
20-23	float	4	Gyro Y (rad/sec)
24-27	float	4	Gyro Z (rad/sec)
28-32			Reserved

7.4 ACCEL

The ACCEL packet contains the accelerations measured by the three accelerometers of the device.

Byte	Type	Size (bytes)	Content
ID	char	1	63 (decimal)
0	char	1	Data valid (1:valid, 0:not valid)
1-7			Reserved
8-15	double	8	Timestamp (msec since boot)
16-19	float	4	Accelerometer X (m/sec^2)
20-23	float	4	Accelerometer Y (m/sec^2)
24-27	float	4	Accelerometer Z (m/sec^2)
28-32			Reserved

7.5 MAGNET

The MAGNET packet contains the readings of the three magnetometers of the device.

Byte	Type	Size (bytes)	Content
ID	char	1	64 (decimal)
0	char	1	Data valid (1:valid, 0:not valid)
1-7			Reserved
8-15	double	8	Timestamp (msec since boot)
16-19	float	4	Magnetometer X (Gauss)
20-23	float	4	Magnetometer Y (Gauss)
24-27	float	4	Magnetometer Z (Gauss)
28-32			Reserved

7.6 TEMP

Contains the temperature as measured by the Outsize Air Temperature (OAT) probe.

Byte	Type	Size (bytes)	Content
ID	char	1	65 (decimal)
0	char	1	Data valid (1:valid, 0:not valid)
1-7			Reserved
8-15	double	8	Timestamp (msec since boot)
16-19	float	4	Temperature (degrees Celcius)
20-32			Reserved

7.7 GPS

This packet contains the readings of the GPS receiver.

Byte	Type	Size (bytes)	Content
ID	char	1	66 (decimal)
0	char	1	Data valid (1:valid, 0:not valid)
1	char	1	Has fix (1:has fix, 0:does not have fix)
2	char	1	Number of satellites
3-7			Reserved
8-15	double	8	Timestamp (msec since boot)
16-19	float	4	Latitude (degres)
20-23	float	4	Longitude (degres)
24-27	float	4	Altitude (m)
28-31	float	4	Ground Speed (m/sec)

7.8 ATTITUDE

This message gives the attitude of the airplane in Euler angles and in quaternions. It also gives the indications for the turn coordinator and the slip ball.

For the Euler angles, the convention is r-p'-y" (roll, then pitch under the rotated coordinate system and then yaw under the rotated-rotated coordinate system).

Byte	Type	Size (bytes)	Content
ID	char	1	71 (decimal)
0	char	1	Data valid (1:valid, 0:not valid)
1-7			Reserved
8-15	double	8	Timestamp (msec since boot)
16-19	float	4	Attitude: Roll (rad)
20-23	float	4	Attitude: Pitch (rad)
24-27	float	4	Attitude: Yaw (rad)
28-31	float	4	Slip ball indication
32-35	float	4	Turn coordinator indication
36-39	float	4	Attitude: Quaternion qx
40-43	float	4	Attitude: Quaternion qy
44-47	float	4	Attitude: Quaternion qz
48-51	float	4	Attitude: Quaternion qw

7.9 COMPASS

This message contains the compass heading. It is also used to broadcast the calibration status while the compass is being calibrated,

Byte	Type	Size (bytes)	Content
ID	char	1	72 (decimal)
0	char	1	Data valid (1:valid, 0:not valid)
1	char	1	Calibration Status (1:No calibration, 4:Calibrating, 2:Just finished, 3:Calibration ok, -1: Too few data, -2: Field weak -3: Field distorted, -4:Error very high)
2	char	1	Has value (1:has, 0:hasn't)
3	char	1	Calibration % (0..100)
4	char	1	Calibration % remaining time
5-7			Reserved
8-15	double	8	Timestamp (msec since boot)
16-19	float	4	Compass heading (degrees)
20-31			Reserved
32-35	float	4	Soft calibration params
36-39	float	4	Soft calibration params
40-43	float	4	Soft calibration params
44-47	float	4	Hard iron calibration
48-51	float	4	Hard iron calibration
52-55	float	4	Hard iron calibration
56-59	float	4	Hard iron calibration
60-63	float	4	Hard iron calibration
64-67	float	4	Hard iron calibration
68-71	float	4	Hard iron calibration
72-75	float	4	Hard iron calibration
76-79	float	4	Hard iron calibration

7.10 WIND

The WIND message contains the wind vector.

Byte	Type	Size (bytes)	Content
ID	char	1	73 (decimal)
0	char	1	Data valid (1:valid, 0:not valid)
1-7			Reserved
8-15	double	8	Timestamp (msec since boot)
16-19	float	4	Wind magnitude (m/sec)
20-23	float	4	Wind direction (degrees)
23-32			Reserved

7.11 CALIBINFO

This message is for internal use.

7.12 DEBUGINFO

This message is for internal use.

7.13 LOG

This message is for internal use.

7.14 TOAST

This packet contains a message that AEOLUS would like to send to the user. E.g. The message "AEOLUS on" displayed on A-EFIS when AEOLUS boots.

Byte	Type	Size (bytes)	Content
ID	char	1	82 (decimal)
0-(N-1)	string	N	A null terminated string

7.15 COMMAND

This packet contains a service command, sent to AEOLUS in the form of the string. Within A-AFIS a user can send such commands to AEOLUS by selecting "Service" and then typing a command that starts with "a ". E.g. "a reboot" can be used to reboot AEOLUS, "a pause", "a resume" can be used to pause/resume AEOLUS, "a c pressure" can be used to calibrate the pressure sensors, etc.

These commands are mostly reserved for internal use.

Byte	Type	Size (bytes)	Content
ID	char	1	82 (decimal)
0-(N-1)	string	N	A null terminated string that contains the command

7.16 SETTIME

A-EFIS periodically sends a SETTIME message to set the time of AEOLUS sensor box. AEOLUS needs to know the time only for logging and debugging purposes so this message is for internal use only.

Byte	Type	Size (bytes)	Content
ID	char	1	12 (decimal)
0	char	1	Current year minus 1900
1	char	1	Current month (Jan=0, Dec=11)
2	char	1	Current day 1..31
3	char	1	Current hour 0..23
4	char	1	Current minute 0..59
5	char	1	Current second 0..59
7-31		26	Reserved

8 Additional information

For more information regarding AEOLUS' communications, please contact:

rnd@talosavionics.com

For general questions & feedback, please contact:

info@talosavionics.com