BasicAirData – Air Data Computer

Communication Protocol - Common Message Set

Attention, this is a work in progress version
Draft Version – 2017 01 04

ID	Field String	Field Name	Description
0	HBQ	HEARTBEAT_REQ	Check if the system is present and responding
1	HBA	HEARTBEAT_ASSERT	Answer about the system presence
2	TMS	TIME_SET	Set date and time of the ADC
3	TMQ	TIME_REQ	Ask for date and time
4	TMA	TIME_ASSERT	The Datetime
5	STS	STATUS_SET	***NOT IMPLEMENTED***
6	STQ	STATUS_REQ	Ask for ADC configuration status
7	STA	STATUS_ASSERT	The Status
8	DTS	DATA_SET	***NOT IMPLEMENTED***
9	DTQ	DATA_REQ	Ask for the ADC sensors data
10	DTA	DATA_ASSERT	The Sensors data
11	SFS	STATUS_FREQ_SET	Set the ADC to send the Status at the specified frequency
12	SFQ	STATUS_FREQ_REQ	Ask for the ADC Status Frequency
13	SFA	STATUS_FREQ_ASSERT	The Status Frequency
14	DFS	DATA_FREQ_SET	Set the ADC to send the Data at the specified frequency
15	DFQ	DATA_FREQ_REQ	Ask for the ADC Data Frequency
16	DFA	DATA_FREQ_ASSERT	The Data Frequency
17	LGD	LOG_FILE_DELETE	Delete the log file
18	LGQ	LOG_FILE_REQ	Ask for the SD log file
19	LGA	LOG_FILE_ASSERT	A line of the log file

Every message string will begin with a “$” character and will end with a “\n” (newline).

# 0 – HBQ – HEARBEAT_REQ

The heartbeat message ask if a system is present and responding.
The remote system will respond with a HEARTBEAT_ASSERT (#1).

0	Field String	$HBQ
1	description	The description of the requesting device
2	firmware_version	The version of the communication protocol

Example:

```
$HBQ,StatusVisualizer,1
```

# 1 – HBA – HEARBEAT_ASSERT

The heartbeat message answer if a system is present and responding.
This message is typically the answer of a HEARTBEAT_REQ (#0).

0	Field String	$HBA
1	description	The description of the responding device
2	firmware_version	The version of the communication protocol

Example:

```
$HBA,Amaranth,1
```

# 2 – TMS – TIME_SET

The request to set the remote datetime.
The remote system will respond with a TIME_ASSERT (#4).

0	Field String	$TMS
1	year	Year (4 digits format)
2	month	Month (from 01 to 12)
3	day	The day of the month, starting from 01
4	hour	Hour, expressed in the 24 hours format (from 00 to 23)
5	minutes	Minutes (from 00 to 59)
6	seconds	Seconds (from 00 to 59)
7	millis	Milliseconds (000 to 999)

Example:

```
$TMS,2016,01,24,13,33,50,000
```

# 3 – TMQ – TIME_REQ

Ask for the remote datetime.
The remote system will respond with a TIME_ASSERT (#4).

0	Field String	$TMQ

Example:

```
$TMQ
```

# 4 – TMA – TIME_ASSERT

The message contains the current datetime of the device.

0	Field String	$TMA
1	year	Year (4 digits format)
2	month	Month (from 01 to 12)
3	day	The day of the month, starting from 01
4	hour	Hour, expressed in the 24 hours format (from 00 to 23)
5	minutes	Minutes (from 00 to 59)
6	seconds	Seconds (from 00 to 59)
7	millis	Milliseconds (000 to 999)

Example:

```
$TMA,2016,01,24,13,33,50,000
```

# 5 – STS – STATUS_SET

The request to set the status of the remote device.
The remote system will respond with a STATUS_ASSERT (#7).

0	Field String	$STS
1		

Example:
`$STS,_______________`

# 6 – STQ – STATUS_REQ

Ask for the remote status.
The remote system will respond with a STATUS_ASSERT (#7).

0	Field String	$STQ

Example:
`$STQ`

# 7 – STA – STATUS_ASSERT

The message contains the current status of the device.

0	Field String	$STA
1	SD card present	1 Present. 0 Not present. E=error code
2	Deltap sensor	1 Present. 0 Not present. E=error code
3	Absolute pressure sensor	1 Present. 0 Not present. E=error code
4	External Temperaure sensor	1 Present. 0 Not present. E=error code
5	Deltap sensor temperature	1 Present. 0 Not present. E=error code
6	Absolute pressure sensor temperature	1 Present. 0 Not present. E=error code
7	Real time clock battery	1 Present. 0 Not present.
8	Error/Warning	That should be a code handled directly by firmware. Exception handling is non implemented. “SDLOW” This is a warning

Example:

```
$STA,1,1,1,1,1,1,1,1,,
```

All the sensors, SD card, real time clock battery are present and operational, no errors are reported.

```
$STA,1,1,0,1,1,1,0,SDLOW
```

All the sensors except external temperature, and SD card are present. SD card space shortage is reported.

# 8 – DTS – DATA_SET

The request to set the data of the remote device. Configuration data will be saved to the SD card,
The remote system will respond with a DATA_ASSERT (#10).

0	Field String	$DTS
1	SD card present	1 Present. 0 Not present.
2	Deltap sensor	1 Present. 0 Not present.
3	Absolute pressure sensor	1 Present. 0 Not present.
4	External temperaure sensor	1 Present. 0 Not present.
5	Deltap sensor temperature	1 Present. 0 Not present.
6	Absolute pressure sensor temperature	1 Present. 0 Not present.
7	Real time clock battery	1 Present. 0 Not present.

Example:

```
$DTS,1,1,1,1,1,1,1
Full configuration installed
$DTS,1,1,0,1,1,1,1
Everything installed but the absolute pressure sensor.
```

# 9 – DTQ – DATA_REQ

Ask for the remote data. Selects the required data with the fields value. If a 0 is sent at the position i then the ADC will send a #10 reply with the field i not valued. If a 1 is sent at the position i then the ADC will send a #10 reply with the field i valued.

The remote system will respond with a DATA_ASSERT (#10).

i	Field String	$DTQ
1-24	Select	0 if data field not required. 1 if data field required

Example:

```
Request all the data fields of #10
$DTQ,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
Request all the data fields except Deltap and Ext Temperature
$DTQ,1,0,1,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
```

# 10 – DTA – DATA_ASSERT

Data lengths shown in the table below are variable and conservative. The ADC will respond to a #9 message with a DTA – DATA_ASSERT that contains only the data fields that have been marked with 1 within the #9 message.

i	Field String	$DTA
1	Timestamp see msg #4	18 Chars
2	Deltap [Sensor units,	6 Chars, integer. Sensor dependent.

	counts]	
3	Absolute Pressure [Sensor units, counts]	6 Chars, integer. Sensor dependent
4	Ext Temperature [Sensor units, counts]	5 Chars, integer. Sensor dependent
5	Temp deltap [Sensor units, counts]	5 Chars, integer. Sensor dependent
6	Temp absolute [Sensor units, counts]	5 Chars, integer. Sensor dependent
7	Deltap [Pa]	8 Chars, 2 decimal digit
8	Absolute Pressure [Pa]	8 Chars, 1 decimal digit
9	Ext Temperature [K]	5 Chars, 1 decimal digit
10	Temp deltap [K]	5 Chars, 1 decimal digit
11	Temp absolute [K]	5 Chars, 1 decimal digit
12	IAS [m/s]	6 Chars, 2 decimal digit
13	TAS [m/s]	6 Chars, 2 decimal digit
14	Altitude [m]	7 Chars, 2 decimal digit
15	OAT [K]	5 Chars, 1 decimal digit
16	Relative time micro millis [s*10^-6]	5 Chars, integer
17	Uncertainty IAS [m/s]	3 Chars, 1 decimal digit
18	Uncertainty TAS [m/s]	3 Chars, 1 decimal digit
19	Uncertainty Altitude [m]	3 Chars, 1 decimal digit
20	Uncertainty OAT [K]	3 Chars, 1 decimal digit
21	Air Density [kg/m^3]	8 Chars,6 decimal digit
22	Air Viscosity[Pas*10^6]	8 Chars,6 decimal digit
23	Re	8 Chars,1 decimal digit
24	c factor	6 Chars,4 decimal digit

Example:

```
$DTA,12,3,33,1,1,2013,6608,******,******,*****,*****,*****,472.60,100926.1,15.0,
18.3,18.6,27.77,27.77,63.1,15.0,1244,0.4,0.7,1.1,0.3,1.225000,18.396057,15081.1,
0.9977
```

# 11 – SFS – STATUS_FREQ_SET

The request to set the frequency rate of the remote status messages.
The remote system will respond with a STATUS_FREQ_ASSERT (#13).

0	Field String	$SFS
1	frequency	The requested frequency rate (messages per second)

Example: (set the frequency to 2 Hz)

```
$SFS,2
```

# 12 – SFQ – STATUS_FREQ_REQ

Query the frequency rate of the remote status messages.
The remote system will respond with a STATUS_FREQ_ASSERT (#13).

0	Field String	$SFQ

Example:

```
$SFQ
```

# 13 – SFA – STATUS_FREQ_ASSERT

The message contains the current frequency rate of the status messages.

0	Field String	$SFA
1	frequency	The current frequency rate (messages per second)

Example: (2 Hz)

```
$SFA,2
```

# 14 – DFS – DATA_FREQ_SET

The request to set the frequency rate of the remote data messages.
The remote system will respond with a DATA_FREQ_ASSERT (#16).

0	Field String	$DFS
1	frequency	The requested frequency rate (messages per second)

Example: (set the frequency to 20 Hz)

```
$DFS,20
```

# 15 – DFQ – DATA_FREQ_REQ

Query the frequency rate of the remote data messages.
The remote system will respond with a DATA_FREQ_ASSERT (#16).

0	Field String	$DFQ

Example:

```
$DFQ
```

# 16 – DFA – DATA_FREQ_ASSERT

The message contains the current frequency rate of the data messages.

0	Field String	$DFA
1	frequency	The current frequency rate (messages per second)

Example: (20 Hz)

```
$DFA,20
```