FlightSaver file format *(ver 1.04)*

FlightSaver data files consist of a sequence of records:

Record 0
Record 1
Record 2
Record 3
…
Last record

The number of bytes in each record is a multiple of 64. The first byte of each record identifies the record type. There are five record types:

- ID='blank': Power on
- ID='B': Bookmark
- ID='F': Fuel-flow data
- ID='P': Pressure data
- ID='U': UBG-16 engine analyzer data
- ID='G' GPS data

History

Date	File rev	Changes
16-Mar-03	1.00	Initial version
26-Mar-03	1.01	Added a byte to record type 'U' which indicates the record length
30-Mar-03	1.02	Changed the GPS data record to a highly compressed form
22-Apr-03	1.03	Fixed time stamp of type 'U' record to agree with the beginning of the record instead of the end. Deleted '85' frame type from 'G' records and added types '80' thru '87'.
20-Aug-03	1.04	Added record type 'P' (Pressure data, or "Pitot-static")

Power-on record (record identifier = 0x20 = ‘blank’)

A 64 byte power-on record is inserted every time the FlightSaver is turned on. (So of course, record 0 must be of this type.) All bytes are ASCII characters except the last six bytes. The record looks like this:

```
Byte  0         1         2         3         4         5         6
  #   0123456789012345678901234567890123456789012345678901234567890123
    " FlightSaver 1.03  FFU1  16-Mar-03 15:17:33  13.67v  Time:ymdhms"
```

Byte 0	Power-on record identifier (‘blank’)
Bytes 13-16	File type version number (changed whenever file format is modified)
Byte 22	Fuel flow unit (see table below)
Bytes 25-33	Current date (dd-mon-yr)
Bytes 35-42	Current time (hh:mm:ss)
Bytes 45-50	Current system voltage
Bytes 58-63	Current date/time (redundant – same info as bytes 25-42)
	Byte 58: year – 2000(binary – from 3 to 255)
	Byte 59: month (binary – from 1 to 12)
	Byte 60: day (binary – from 1 to 31)
	Byte 61: hour (binary – from 0 to 23)
	Byte 62: minute (binary – from 0 to 59)
	Byte 63: second (binary - from 0 to 59)

Unit code	Fuel flow unit
'1'	.01 gal/hr
'2'	.1 gal/hr
'3'	.1 lb/hr
'4'	.1 liter/hr
'5'	.1 kg/hr

Fuel flow unit codes (byte 22)

Book-mark record (record identifier = 0x4D = ‘B’)

A 64 byte book-mark record is inserted every time the “Mark” button is pressed. As with the power-on record, all bytes are ASCII characters except the last six. The record looks like this:

```
Byte  0         1         2         3         4         5         6
  #   0123456789012345678901234567890123456789012345678901234567890123
    "BA+++++++++++++++++++++++ 16-Mar-03 15:17:33  13.67v  Time:ymdhms"
```

Byte 0	Bookmark record identifier (‘B’)
Byte 1	Bookmark identifier (ASCII A thru Z)
Bytes 25-63	Same as for power-on record

Fuel flow data record (record identifier = 0x46 = ‘F’)

This is a 128-byte record containing 1 minute of fuel flow data (60 samples)

Byte	Contents
0	`‘F’`
1	`month (1-12)`
2	`day  (1-31)`
3	`hour (0-23)`
4	`minute (0-59)`
5	`second (0-59)`
6	`lsb(gal(t0))`
7	`msb(gal(t0))`
8	`lsb(gph(t0))`
9	`msb(gph(t0))`
10	`lsb(gph(t0+1))`
11	`msb(gph(t0+1))`
12	`lsb(gph(t0+2))`
13	`msb(gph(t0+2))`
14	`lsb(gph(t0+3))`
15	`msb(gph(t0+3))`
.	.
.	.
.	.
126	`lsb(gph(t0+59))`
127	`msb(gph(t0+59))`

where:

`t0`	is the time given by bytes 1-5
`t0+1`	is one second later
`gph`	is the fuel flow in the specified units (see unit codes above)
`gal`	is the fuel remaining (e.g. in .01 gallon units for unit code 1)
`lsb()`	indicates the least significant byte
`msb()`	indicates the most significant byte

Note: Bytes 1 and 2 are somewhat redundant since the date is also included in the power-on record

UBG-16 Engine Analyzer data record (record identifier = 0x55 = 'U')

This record contains two minutes of data (24 samples at a .2 Hz sample rate for 16 data channels) – 384 values in all. The data is stored in units of °F. The record length is equal to N*64 bytes where N can range from 1 to 7 (and depends on how many bits are needed to represent each data value). The form of the record is as follows:

Byte 0	'U'
Byte 1	N (from 1 to 7), where length of record = 64*N
Byte 2	Reserved (0)
Bytes 3/4/5	TimeStamp: Hour(0-23)/Minutes(0-59)/Seconds(0-59)
Channel 1 (EGT 1)	2 to 26 bytes (see tables below)
Channel 2 (CHT 1)	2 to 26 bytes
Channel 3 (EGT 2)	2 to 26 bytes
Channel 4 (CHT 2)	2 to 26 bytes
Channel 5 (EGT 3)	2 to 26 bytes
Channel 6 (CHT 3)	2 to 26 bytes
Channel 7 (EGT 4)	2 to 26 bytes
Channel 8 (CHT 4)	2 to 26 bytes
Channel 9 (EGT 5)	2 to 26 bytes
Channel 10 (CHT 4)	2 to 26 bytes
Channel 11 (EGT 6)	2 to 26 bytes
Channel 12 (CHT 6)	2 to 26 bytes
Channel 13 (Oil T)	2 to 26 bytes
Channel 14 (OAT)	2 to 26 bytes
Channel 15 (VAC)	2 to 26 bytes
Channel 16 (----)	2 to 26 bytes
Zero padding	0 to 63 bytes (padding so that record length = 64*N bytes)

There are 15 ways to encode the 24 samples for each channel depending on the degree of compression possible. Note that each channel may be encoded using a different encoding type. The encoding types are:

Encoding type	Bits per sample	res (Resolution in °F)	Bytes per channel
0	0	1	2
1	1	1	5
2	2	1	8
3	4	1	14
4	8	1	26
5	0	2	2
6	1	2	5
7	2	2	8
8	4	2	14
9	8	2	26
10	0	4	2
11	1	4	5
12	2	4	8
13	4	4	14
14	8	4	26
15	←——— Reserved ———→		

For all 15 encoding types, the first two bytes are in this form:

← Byte 1 →								← Byte 0 →							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
Encoding type				0	V_{min} (-1024 to +1023)										

The 24 samples have the value: res • (V_{min} + V_i)

where: res equals 1,2, or 4 °F according to the above table

V_i (i = 0 to 23) are non-negative integers as defined below

For encoding types {0,5, 10} all V_i are zero (i.e. all 24 samples have the same value) and the two bytes above are the only bytes stored for the channel.

For encoding types {1,6,11} the 24 values V_0 - V_{23} (which can be 0 or 1) are packed into the next 3 bytes:

bit →	7	6	5	4	3	2	1	0
Byte 2	V_7	V_6	V_5	V_4	V_3	V_2	V_1	V_0
Byte 3	V_{15}	V_{14}	V_{13}	V_{12}	V_{11}	V_{10}	V_9	V_8
Byte 4	V_{23}	V_{22}	V_{21}	V_{20}	V_{19}	V_{18}	V_{17}	V_{16}

For encoding types {2,7,12} the 24 values V_0 - V_{23} (which can be 0 thru 3) are packed into the next 6 bytes:

bits →	7:6	5:4	3:2	1:0
Byte 2	V_3	V_2	V_1	V_0
Byte 3	V_7	V_6	V_5	V_4
Byte 4	V_{11}	V_{10}	V_9	V_8
Byte 5	V_{15}	V_{14}	V_{13}	V_{12}
Byte 6	V_{19}	V_{18}	V_{17}	V_{16}
Byte 7	V_{23}	V_{22}	V_{21}	V_{20}

For encoding types {3,8,13} the 24 values V_0 - V_{23} (which can be 0 thru 15) are packed into the next 12 bytes:

bits →	7:4	3:0
Byte 2	V_1	V_0
Byte 3	V_3	V_2
Byte 4	V_5	V_4
Byte 5	V_7	V_6
Byte 6	V_9	V_8
Byte 7	V_{11}	V_{10}
Byte 8	V_{13}	V_{12}
Byte 9	V_{15}	V_{14}
Byte 10	V_{17}	V_{16}
Byte 11	V_{19}	V_{18}
Byte 12	V_{21}	V_{20}
Byte 13	V_{23}	V_{22}

For encoding types {4,9,14} the 24 values V_0 - V_{23} (which can be 0 thru 255) are placed in the next 24 bytes:

bits →	7:0
Byte 2	V_0
Byte 3	V_1
Byte 4	V_2
…	…
…	…
…	…
Byte 25	V_{23}

GPS data record (record identifier = 0x47 = 'G')

This is a fixed size record of 256 bytes.

Byte	Contents
0	`'G'`
1	`'G' (future use)`
2	`Sample period (Δt) in seconds (1-255)`
3	`hour (0-23)`
4	`minute (0-59)`
5	`second (0-59)`
6	`0 (future use)`
7	`0 (future use)`
8	`GPS data frame 0 (time t0)`
9	.
10	.
	`GPS data frame 1 (time t0+Δt)`
	.
	`GPS data frame 2 (time t0+2Δt)`
	.
	.
	.
	.
	.
	.
	`GPS data frame N (time t0+nΔt)`
	.
254	.
255	`0x80 (nop)`

Each GPS data frame consists of a single position sample. Note that 0x80 is a null data frame used as filler to pad the GPS record out to 256 bytes if necessary. There are three data frame types as shown below:

Data frame type 8F (15 bytes):

Byte	Contents
0	0x8F (identifies the frame as the 15 byte format)
1	GPS derived time: Hours (0-23)
2	GPS derived time: Minutes (0-59)
3	GPS derived time: Seconds (0-59)
4	Latitude: [Sddddddd] S = 0/1 (north/south), d = degrees (0-89)
5	Latitude: [mmmmmmmm] LSB - minutes in hundredths (0-5999)
6	Latitude: [000mmmmm] MSB
7	Longitude: [dddddddd] d = degrees (0-179)
8	Longitude: [mmmmmmmm] LSB - minutes in hundredths (0-5999)
9	Longitude: [E00mmmmm] MSB - E = 0/1 (west/east)
10	Altitude (gps derived): LSB (in meters) –32768 if unavailable
11	Altitude (gps derived): MSB
12	Magnetic variation: LSB (in sixteenths of a degree) East/West=+/-
13	Magnetic variation: MSB
14	Estimated accuracy - in sixteenths of a nm (255 if unavailable)

Data frame types 80-87 (1 to 5 bytes):

Byte	Contents
0	Frame type: 0x80-87 = binary: 10000sap
1	eT: Latitude correction (-1.28 to +1.27 minutes)
2	eG: Longitude correction (-1.28 to +1.27 minutes)
3	eA: Altitude correction (-128 to +127 meters)
4	eS: Time correction – GPS derived (-128 to +127 seconds)

The position represented by this data frame can be computed as follows:

Latitude = Tp + (Tp-Tpp) + eT
Longitude = Gp + (Gp-Gpp) + eG
Altitude = Last specified altitude + eA

where: Tp/Gp is the lat/long of the preceding sample
Tpp/Gpp is the lat/long of the sample before that

If the p bit is 0, then bytes 1 and 2 are combined into a single byte as follows:

Contents
[ttttgggg] : tttt = Latitude correction Et (-.07 to +.07 minutes) gggg = Longitude correction Eg (-.07 to +.07 minutes)

If the a bit is 0, then byte 3 is omitted and the altitude is assumed to be the same as for the previous frame. Normally the GPS derived time will be Δt seconds later than the previous position sample. However occasionally a small correction will be inserted by setting s=1 and supplying the time correction in byte 4. If the s bit is 0, the byte 4 is omitted. If s=a=p=0 then byte 0 is omitted and the frame consists of just a single byte. Frame types 88-8E are reserved for future use. No data is remembered from previous records, which means that the first data frame is always type 8F. Otherwise type 8F is only encountered during rare lat/long glitches. Unless you are moving very slowly, the second data frame will have p=1 (since Tpp and Gpp are initialized to Tp and Gp respectively.)

Pressure data (or Pitot-static) record (record identifier = 0x50 = 'P')

This is a 128-byte record containing 5 minutes of altimetry data (60 samples)

Byte	Contents
0	`'P'`
1	`month (1-12)`
2	`day  (1-31)`
3	`hour (0-23)`
4	`minute (0-59)`
5	`second (0-59)`
6	`P.alt(t0), lsb`
7	`P.alt(t0), msb`
8	`CAS(t0), lsb`
9	`CAS(t0), msb`
10	`ΔP.alt(t0+5)`
11	`ΔCAS(t0+5)`
12	`ΔP.alt(t0+10)`
13	`ΔCAS(t0+10)`
14	`ΔP.alt(t0+15)`
15	`ΔCAS(t0+15)`
.	.
.	.
.	.
126	`ΔP.alt(t0+295)`
127	`ΔCAS(t0+295)`

where:	`t0`	is the time given by bytes 1-5
	`t0+5`	is five seconds later
	`P.alt`	is the pressure altitude (in units of 4 feet)
	`ΔP.alt`	is the pressure altitude change from the previous sample (in units of 4 feet)
	`CAS`	is the calibrated airspeed (in units of .2 knots)
	`ΔCAS`	is the calibrated airspeed change from the previous sample (in units of .2 knots)
	`lsb`	indicates the least significant byte
	`msb`	indicates the most significant byte

Note: Bytes 1 and 2 are somewhat redundant since the date is also included in the power-on record