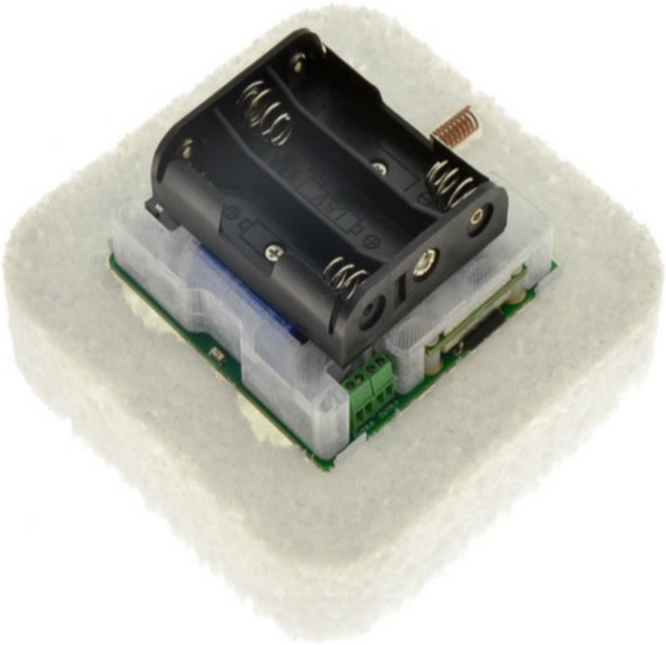
User Guide

HAB PILOT

Satellite Balloon Flight Computer

Balloon
Ascent
Technologies

The sky's not the limit; it's where we start.



Overview

The *HAB Pilot* is a stand-alone balloon flight computer that:

- Controls other Loonatec HAB Modules such as the HAB Ballaster & HAB Venter
- Command & telemetry via the Iridium satellite network for world-wide coverage
- Built-in flight termination: can activate both squibs (pyrotechnic) and burn-wires (thermal)
- Monitors flight conditions: GPS, pressure, accelerometer, magnetometer, and clock sensors and can respond autonomously, sending pre-programmed commands

Rev. 5 — Jan 2024

loonatec.com

Pilot Operating Specifications

Parameter	Min	Typical	Max	Unit
Operating: Temperature	-80	-	60	°C
Operating: Time (w/ 3x L91 batteries)	10	20	80	hours
Operating: Heater Set Point	-60	-10	60	°C
Operating: Internal Power	3x Li/FeS2 AA Batteries			
Operating: External Power	2-5 VDC, < 0.5 amps			
Sensor: Pressure	8	-	1200	mBar
Sensor: Temperature	-70	-	120	°C
Sensor: GPS Altitude	-	-	50,000	meters
Sensor: Accelerometer	-	-	16	g’s
Sensor: Gyroscope	-	-	500	°/sec
Mechanical: Mass (no Enclosure or Batteries)		75		g
Mechanical: Mass (incl. Batteries)		120		g
Mechanical: Mass (incl. Enclosure & Batteries)		140		g
Mechanical: Size in Foam Enclosure	112 x 112 x 78 mm			
Mechanical: Size without Foam Enclosure	70 x 70 x 35 mm			

Table of Contents

1 Pre-Flight Checklist

Perform the following steps up to 24 hours in advance of launch:

1) Move Power Switch to *OFF* position
2) Insert batteries – 3x L92 batteries
3) Rig into flight train so that Iridium antenna will face up

Perform the following steps 30-60 min before launch

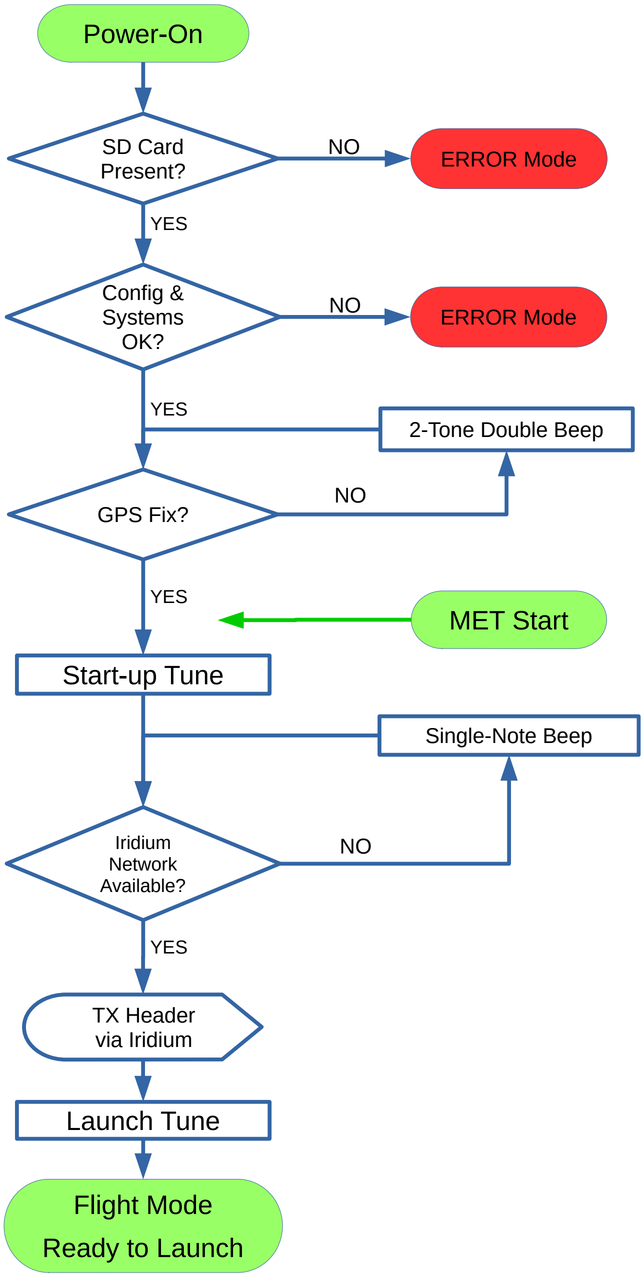
1) Go to the launch area with an unobstructed view of the sky
2) Move Power Switch to *ON* position
3) Place HAB Pilot upright
4) Listen for the 2-tone double beep indicating waiting for a GPS fix
5) Turn on any other HAB Modules that are controlled by the HAB Pilot
6) Listen for the Start-up tune to indicate GPS fix has been acquired

   NOTE TIME → this is MET = 0

7) Verify that the HAB Modules' blue BLE LED changes from 2-4 Hz to a slow flash rate depending on Flight.cfg setting
8) Wait for data header to be transmitted via Iridium SBD

   DO NOT LAUNCH until message sent

9) Listen for Launch Tune to play, followed by silence
10) Verify that header data has been received by the Mission Control software
11) Verify HAB Pilot's green Status LED is flashing at 4 Hz.
12) Unit is now ready to be flown—place in foam enclosure and seal with tape



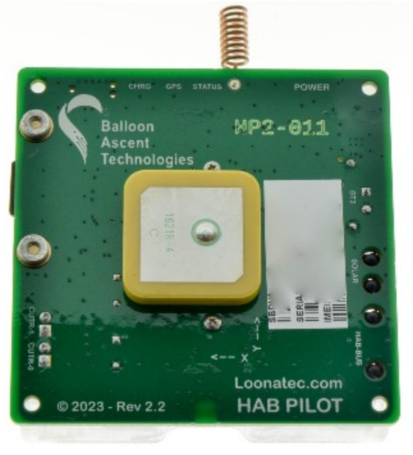
2 Hardware

Overview

The HAB Pilot is a small, stand-alone flight computer. It can be powered from a combination of primary lithium AA batteries, solar, or external power. It can be wired directly to control burn-wires or pyrotechnic squibs. It also offers and opt-isolated *HAB Bus* connection so that *HAB Modules* can be connected directly.

Antennas

There are 3 antennas on the *HAB Pilot* oriented so that when it is placed in a standard orientation all three have a clear view.



In the center top of the *HAB Pilot* is the Iridium modem’s ceramic patch antenna. It should face upward for best satellite coverage. Ideally the *HAB Pilot* should be rigged so that it remains facing upward after balloon separation, otherwise Iridium communication will be greatly degraded.

Sticking out of the side of the *HAB Pilot* is the LoRa spring antenna. In this orientation it has good coverage up and down the flight train where associated *HAB Modules* will likely be arranged.

Lastly there is a small GPS chip antenna that shares a similar view of the sky with the Iridium antenna. It has reasonable 360° coverage so that even if the *HAB Pilot* inverts after balloon separation it can continue to receive GPS position.

Foam Enclosure

Loonatec offers a custom two-part EPP foam enclosure for the *HAB Pilot*. This foam enclosure provides both thermal insulation as well as mechanical protection during landing.

The enclosure uses some small snaps to hold the halves together. While these work reasonably well an additional wrap of tape around the entire enclosure before launch is still recommended.

The larger bottom section’s snaps usually release first. This exposes the batteries, power switch, and micro-SD slot. If the *HAB Pilot* needs to be removed from the smaller top section then gently pull back on two of the plastic clips on one side while lifted it out of the foam.

3 Interface

Physical

Iridium SBD

The following ASCII formatted commands are permitted:

SleepMode: Allows the Iridium SBD modem to go into a low-power mode when not actively transmitting. Results in up-link commands not being received until a down-link data message is sent.

AwakeMode: Keeps the Iridium SBD modem in a powered-on mode to receive Ring Alerts. Should result in up-link commands being received promptly but uses more battery power.

Report #: Changes the *HAB Pilot's* down-link data message transmission rate. Number argument is in seconds. It is suggested to use a minimum report rate of 300 seconds (5 minutes) for optimal Iridium performance.

Shutdown: Turn the *HAB Pilot* OFF. If the Power Switch is still in the 'ON' position then the HAB Pilot will restart.

LoCFS #: Sets the Loss-of-Comm-Failsafe timer to n seconds and starts counting down. A value of "-1" will disable the LoCFS timer. When the timer expires the Terminate command, if configured, will be activated.

Command Name: Any command in the Flight.cfg file can be activated by sending its name. Example: sending "PwrOnTest" (see Command Section)will cause the "Command: PwrOnTest{}" in the Flight.cfg file to be activated again, regardless of any "Conditions{}" set.

HAB Module: A command can be relayed to any HAB Module that is configured in the Flight.cfg file (see Configuration Section)by sending the module's name, a space, and the desired command. Example: sending "**`!cutr1 sd=Y`**" will shutdown the module with the name **`!cutr1`**.

micro-SD Card

The *HAB Pilot* creates a new directory each time it is powered on. Inside of this directory are the following files:

- Flight.CSV – Contains comma-separated data recorded at up to 2 Hz
- Flight.KML – Contains position data in a format that programs like Google Earth use
- Flight.LOG – Contains a time-stamped system log for post-flight analysis

Power Switch

ON Position: This will turn the *HAB Pilot* on when batteries are present. It enables a backup super-capacitor that guards against short power losses of a couple of seconds. It ensures that the *HAB Pilot* turns back on after any power interruptions or the Iridium SBD *shutdown* command is issued

Off Position: This will not turn off the HAB Pilot. It does disable the backup super-capacitor so momentary power losses will cause it turn off. It means that the HAB Pilot will remain off if the Iridium SBD *shutdown* command is issued.

Shutdown: Move the power switch to the OFF position and either issue the Iridium SBD command or remove one of the batteries.

Audio Tunes

GPS Wait: the HAB Pilot will play a 2-tone beep briefly once every two seconds to indicate that GPS position fix has not yet been obtained and should not be launched.

GPS Lock: Once the HAB Pilot acquires GPS lock the intro to Bach's Toccata and Fugue in D Minor will play to indicate a successful self-check power-on sequence.

Data Header Wait: the HAB Pilot will play a 1-tone beep briefly once every two seconds to indicate that the data header has not yet been transmitted and that the HAB Pilot should not be launched yet.

Launch Tune: Once the HAB Pilot has successfully transmitted the data header message the intro to Beethoven's Für Elise will play to indicate that the HAB Pilot is ready to be launched.

ERROR Tune: If an error is detected during the power-on sequence, a continuous rapid 2-tone sound will be played. The HAB Pilot is unusable while this rapid continuous beeping is sounding.

Command Activation Tone: User selectable single note sequence can be configured to play in the *Actions* section of a *Command* in the Flight.cfg file.

STATUS LED

The *HAB Pilot* has a green Status LED. Possible blink patterns are:

- While acquiring GPS lock it will will alternate at 1 Hz
- In normal operations the status LED will flash at 4 Hz
- If there is an error, the STATUS LED will flash rapidly, indicating that the *HAB Pilot* is unusable until the error is corrected

GPS LED

The HAB Pilot has a blue GPS LED. It will flash at 1 Hz when GPS time is acquired.

CHRG LED

The HAB Pilot has an orange CHRG LED. It will turn on when the two large super-capacitors are being charged, and off once they reach >90% of their target charge. These super-capacitors are used by the Iridium SBD modem and burn-wire/squib cutter module. They take about 2 minutes to fully charge if the HAB Pilot has not been used for a couple of weeks.

Error Mode

The *HAB Pilot* will enter Error Mode if any of the following problems are encountered:

- micro-SD Card Problem
    - A no-card-present failure will be indicated by Error Mode and no Flight.log file
    - An incorrect formatting or no available space failure will be indicated by Error Mode and the problem indicated in the Flight.log.
- Cutter Continuity Problem—One or more of the enabled cutters failed a continuity test
- Data Source Problem—An issue was detected with an onboard sensor

Error Mode is indicated by rapid one-tone beeping that is in sync with green Status LED blinks.

Check the Flight.LOG file for an indication of which subsystem failed. If the Flight.LOG does not exist then there is likely an issue with the micro-SD card itself.

4 Programming

The *HAB Pilot* is programmed via the “config/Flight.cfg” file. A default Flight.cfg will be created on a blank SD card that is inserted into the *HAB Pilot* when it is turned on. The unit will enter Error Mode when this occurs to alert the user that a default Flight.cfg has been created and needs to be verified.

The following is an explanation of the various options in that file.

Configuration Section

This section is used to configure the *HAB Pilot* and its subsystems. The following is an example of the default settings for a v2.2 HAB Pilot.

```
Version: 1
# -- Misc Settings
#GPSLowPwr:
#EnableDebugInfo:
HeaterEnableTemp: -10 #-20 for long flights
LoCFS: -1
# -- LoRa Radio Settings
CommDevice: Lora, 915, 125, 9, 7, 18, 2, 8
# -- Wired HAB Module Settings
# CommDevice: HabBus, 115200
# -- HAB Module Settings
HAB: !cutr1, LoRa, A, 120, 600
# HAB: !cutr2, LoRa, B, 30, 100
# HAB: !blst1, HabBus, C, 300, 1000
# HAB: !blst2, HabBus, D, 400, 2000
# HAB: !vntr1, LoRa, E, 30, 100
# -- Iridium Settings
CommDevice: Iridium, 1, 19200, 300
CommData: Timestamp, System, Position, IMU, Ambient, !cutr1
CommBinary: true
CommHeader: true
# -- SD Data Settings
SDCardData: Timestamp, System, Position, IMU, Ambient, !cutr1
SDCardRate: 1.0
# -- Enable Squib/Burn-wire
CutterControl:
```

Note: any line that begins with a ‘#’ character is considered a comment and ignored. The lines are colored green for ease of identification in this document but may not be in your text editor.

- Version → A user configurable version number for tracking *Flight.cfg* changes in the *Flight.log* files.
- # → Any line that begins with a '#' character is considered a comment and ignored. Likewise, any text that comes after a '#' is also considered a comment and ignored. The comments are colored green for ease of identification in this document but may not be in your text editor.
- GPSLowPwr → Enable low power mode on the uBlox GPS module
  Ideal for long-duration flights where battery life is important
  - Reduces HAB Pilot's power consumption by about 20%
  - Limits GPS to 1 Hz solution instead of the normal 2 Hz
  - Disables SBAS, which reduces accuracy and removes an integrity check
  - Can result in lost GPS position fixes in the dataset for a couple of seconds at a time
- EnableDebugInfo → Enable to record extra information in the "Flight.log" file
  - SD Card information
  - Memory used per Command selected
  - Available free memory
  - Time to GPS fix
- HeaterEnableTemp → Temperature at which the HAB Pilot's battery heaters turn on
  - **-10** → Heater set point, in degrees Celsius
- LoCFS ➝ Configures the Loss-of-Comm-Failsafe timer subsystem
  - **300** ➝ LoCFS timer period:
    -1 →timer not activated upon boot, requires an incoming command to start
    300 → timer starts at boot with a value of 300 seconds (5 minutes)
    Accepts values of 1 to 32,000 seconds
- CommDevice → The settings to enable LoRa-based HAB Module communication
  - **LoRa** → Use the LoRa radio communication link
  - **915** → LoRa frequency, in MHz valid range is 862.0 to 1020.0 MHz
  - **125** → LoRa bandwidth, in kHz
    Valid options are: 7.8, 10.4, 15.6, 20.8, 31.25, 41.7, 62.5, 125, 250, or 500 kHz
  - **9** → LoRa Spread Factor valid range is 6 to 12
  - **7** → LoRa CR valid range is 5 to 8
  - **18** → LoRa Sync word, in decimal
    Default is 18 (0x12), 52 (0x34) is reserved for LoRaWAN

  - **2** → LoRa Transmit Power, in mW valid range is -3 to 15
  - **8** → LoRa Preamble valid range is 6 to ?
- CommDevice → The settings to enable wire-based HAB Module communication
  - **HabBus** → Use the opto-isolated serial communication link
  - **115200** → Serial baud rate valid range is 1200 to 115200 baud
    Note: must match the HAB Module(s) baud rate, set with the **`!iSU`** command.
- HAB → The settings for a specific HAB Module
  - **!cutr1** → Module name, must start with '!' and be unique
  - **LoRa** → Communication method, either **LoRa** or **HabBus**
    This communication method must already be declared by a CommDevice call
  - **A** → Module's identification letter, set with the **`!iSI`** command
  - **120** → Interval at which to poll module at, in seconds
  - **600** → Value to set the module's WDT to, in seconds
    The value is optional, if not specified then the module's WDT is not enabled
    If used, normally a little greater than three times the polling interval
- CommDevice → The Iridium SBD serial connection settings. Arguments:
  - **Iridium** → Name of the port type [not user selectable]
  - **19200** → Baud rate [default is 19200]
  - **300** → Output interval in seconds, user selectable (300 to 1800 are common)
- CommData → The collection of data sets to be transmitted via the Iridium SBD modem. Arguments:
  - **Timestamp** → GPS timestamp and *HAB Pilot's* Mission-Elapsed-Timer (MET)
  - **Position** → Longitude, Latitude, Altitude, Heading, Ground Speed, Vertical Speed
  - **IMU** → Acceleration in XYZ, Aggregate acceleration, Gyroscope in XYZ, and RPM
  - **Ambient** → Pressure, rate of pressure change, and humidity
  - **System** → System data such as Battery Voltage, Status flags, and Cutter info
  - **GPS** → Either the raw NMEA GGA (ASCII) or the UBX-NAV-PVT message (binary)
- CommBinary → Whether to transmit binary or ASCII data over the Iridium SBD modem
  - Arguments: true or false
- CommHeader → Whether to transmit an ASCII header at power-on to aid in decoding the subsequent binary or ASCII data

  - Arguments: true or false

- SDCardData → The collection of data containers to be logged to the SD card's *Flight.csv* file. See SDCardData above for container options.

- SDCardRate → Record every x.x seconds in 0.5 second intervals

  - Example: a setting of "0.5" results in 2 Hz Flight.csv data

- CutterControl → Enables the cutter control subsystem to activate squib(s) or burn-wire(s). No arguments, configuration occurs in the *Action* section of each programmed *Command*.

Command Section

The *HAB Pilot* can be configured to have a reasonable number of commands. The actual allowable number of commands is dependent on their complexity—RAM usage. For instance, the *HAB Pilot* supports at about six 128 waypoint geofence commands or 25 simple ones using just a Compare trigger.

The *HAB* Pilot will enter into the *Error Mode* and report the issue in *Flight.log* if too many commands are attempted. For a better idea of memory usage, usually the limiting factor, add `EnableDebugInfo:` to the configuration section of the Flight.cfg file and review the Flight.log file after powering on the *HAB Pilot* for ~30 seconds.

Each of these commands is acted upon independently by the conditions set and results in the specified actions being executed. The name, **PwrOnTest** in the below example, is user configurable and used to identify the command activation in the Flight.log and Flight.kml files as well as activate it via the Iridium communication link.

The **PwrOnTest** command, shown below, will result in the following:

1) Activate once, 30 seconds after GPS fix is acquired
2) Play a G5 note twice of duration one second, separated by one second delays
3) Sends the command **`!iSH`** (query heater set point) to all of the **`!cutr`** HAB Modules
4) Logs all of the **`!cutr`** HAB Module replies in the Flight.log file so each of their heater set points is recorded for the flight

```
Command: PwrOnTest
{
  Conditions:
  {
    Trigger:
    {
       Compare: Timestamp.MET   = 30.000000
    }
  }
  Actions:
  {
    PlayTone: 784, 1.00, 1.00, 2
    HAB: !cutr*, SH
  }
}
```

The **Terminate** command, shown below, will result in the following:

1) Activate when “Terminate” is received via the Iridium SBD modem, no other trigger conditions
2) Play a B5 note twice of duration one second, separated by quarter second delays
3) Send the **`!iCT`** (start terminate sequence) to the **`!cutr1`** HAB Module
4) Pause for 1 seconds
5) Activate the burn-wire attached to Cutter #0 at 3 watts for 5 seconds
6) Pause until the super-caps recharge to 5 volts
7) Play a B5 note twice of duration one second, separated by quarter second delays
8) Pause for 1 seconds
9) Activate the burn-wire attached to Cutter #1 at 3 watts for 5 seconds
10) Command finishes, can be called again via another Iridium SBD terminate command

```
Command: Terminate
{
  Actions:
  {
    PlayTone: 1047, 0.25, 0.25, 2
    HAB: !cutr1, CT=Y
    Pause: 1.00
    CutLine: 0, 1, 3.0, 5.00, false
    CutterRecharge: 5.00
    PlayTone: 1047, 0.25, 0.25, 2
    Pause: 1.00
    CutLine: 1, 1, 3.0, 5.00, false
  }
}
```

Note: the **Terminate** command name is unique. It is used by the Loss-of-Comm-Failsafe (LoCFS) timer. If one is not specified then the LoCFS timer will do nothing when it expires.

Conditions

This is a list of conditions that must be met for the command to be acted upon. Activate options include:

Arm → Conditions that must be met before the Trigger conditions are compared. Can permit a Command to be activated multiple times instead of the normal single activation. Not required.

Trigger → Conditions that must be met before the actions can be executed.

Disarm → Conditions that disable the command. Not required.

AndCollection → Group of conditions that are grouped together with a logical AND operator

OrCollection → Group of conditions that are grouped together with a logical OR operator

Example: The following example causes a triangular geofence boundary to be active, from one minute until ten minutes after GPS fix acquisition as long as the pressure is less than 800 hPa for 5 seconds.

```
Conditions:
{
  Arm:
  {
     AndCollection:
     {
        Compare: Pressure.Pressure < 800, 5
        Compare: Timestamp.MET = 60
     }
  }
  Trigger:
  {
        GeoFence:
        {
          Remain: inside
          Altitude: -500, 2233
          Waypoint: -70.636642, 41.585285
          Waypoint: -70.624188, 41.579674
          Waypoint: -70.625283, 41.589149
        }
  }
  Disarm:
  {
    Compare: Timestamp.MET >= 600
  }
}
```

Actions

This section holds the commands that the *HAB Pilot* will execute when the *Conditions* are satisfied. The commands are executed sequentially.

Options include:

- CutLine → Activates a cutter output. Requires the CutterControl subsystem to be enabled in the Configuration Section. The five arguments:
  - Cutter number, 0 or 1
  - Cutter Condition: only activate if the selected cutter passes its continuity test
  - Power level, 0.1 to 5.0 watts nominally
  - Duration, 0.1 to 32,000 seconds
  - Remove from continuity test after activation, true or false
- PlayTone → Causes an audible tone to be played. The *Command's Action* execution will pause while the tone is played. The arguments:
  - Frequency, in Hz
  - Duration on in seconds
  - Duration off in seconds
  - Number of iterations through the on/off cycle
- Pause → Causes the *Action* execution to be paused
  - Argument: Seconds to delay
- CutterRecharge → Causes the *Action* execution to be paused until the super-cap can recharge to a specified voltage
  - Argument: Target super-cap charge level, in volts, to delay until reached
  - Range: 0 to 5.1 volts
- HAB → Send a specific command to one or more HAB Modules
  - HAB Module(s) name as specified in the Flight.cfg settings
    '*' name wildcard that can be used to send the command to multiple HAB Modules
  - HAB Module command, see that HAB Module's User Guide for possible commands

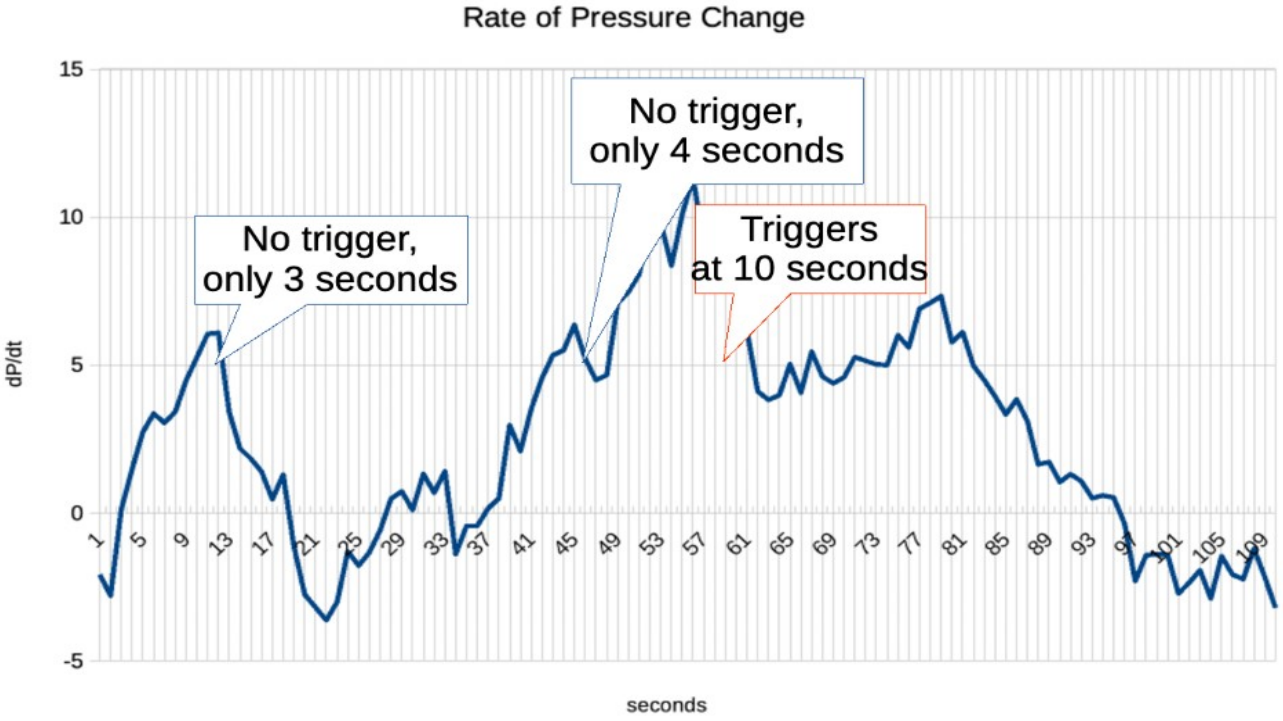
The following is a list of possible logical conditions for Arm, Trigger, and Disarm:

Compare:

This command compares an operational value against a preset value using “<”, “>”, “=”, “<=”, “>=”. Some possible operational values:

- Timestamp.MET
- System.BoardTemperature
- System.BatteryVoltage
- System.CapVoltage
- System.SystemStatus
- IMU.AccX/Y/Z
- IMU.AccAgg
- IMU.GyroX/Y/Z
- IMU.GyroAgg
- IMU.GyroFltr
- Position.Longitude
- Position.Latitude
- Position.Altitude
- Position.Heading
- Position.GroundSpeed
- Position.VerticalSpeed
- Pressure.Pressure
- Pressure.PressureRate
- Pressure.Humidity
- HABModule.Batt_dV
- HABModule.Temp_C
- HABModule.Status

A debounce parameter can be added to the Compare command. This parameter will require the comparison to be valid for a specified number of seconds, tested at 2 Hz. Example: when Pressure Rate goes above 5 Pa/s for at least 10 seconds: `Compare: Pressure.PressureRate > 5, 10`



GeoFence:

This condition compares the *HAB Pilot's* current position against a polygon geofence.

```
GeoFence: 7
{
  Remain: inside
  Altitude: -500, 2233
  Waypoint: -70.636642, 41.585285
  Waypoint: -70.624188, 41.579674
  Waypoint: -70.625283, 41.589149
  Waypoint: -70.615079, 41.586535
  Waypoint: -70.613184, 41.591485
  Waypoint: -70.629393, 41.596487
  Waypoint: -70.638358, 41.595474
}
```

Arguments include:

- GeoFence: **7** → Max number of waypoints expected for this Geofence.

  If no number is supplied then the *HAB Pilot* will accept up to 12 waypoints. If this is set below the actual number of waypoints provided in the Flight.cfg file then an error will be generated in the Flight.log.

  Note: there is no need to manually close the polygon. If the first waypoint is not the last then the *HAB Pilot* will automatically include it to close the polygon. This repeated waypoint does not need to be included in the expected number of waypoints as long as it isn't listed as a user-entered waypoint.

- **Remain** → "inside" triggers if the *HAB Pilot* goes outside of the geofence region, while "outside" will trigger if the *HAB Pilot* enters into the geofence region.
  Defaults to "inside" if this argument is missing.

- **Altitude** → Minimum and maximum altitudes permitted over the entire geofence.
  Defaults to -1,000 to 50,000 meters if this argument is missing.

- **Waypoint** → Each GeoFence needs at least three and not more than 128 waypoints. The *HAB Pilot* will repeat the first waypoint to close the polygon automatically if needed.

5 Log Files

All SD-card data files are buffered at 2 Hz but only written every ten seconds.

The Flight.CSV Data File

Data

- DateTime GPS date in YYYY/MM/DD HH:MM:SS format
- MET Internal second timer, starts counting from 0 at power-on
- Status A bit-mapped data flag
    - Bit-0 Heater Flag, the battery heater is active
    - Bit-1 Iridium power-save mode
    - Bit-2 GPS Power Save Mode active
- LoCFS Loss-of-Comm-Failsafe timer, in seconds
- CutC Cutter continuity bit-mapped flag (updated every 60 seconds)
    - Bit-0 Cutter #0: 0=Good continuity, 1=No Continuity
    - Bit-1 Cutter #1: 0=Good continuity, 1=No Continuity
    - Bit-2 Cutter #2: 0=Good continuity, 1=No Continuity
    - Bit-3 Cutter #3: 0=Good continuity, 1=No Continuity
- Measurements
    - IntT The Pilot's internal board temperature, in °C
    - BatV The Pilot's battery voltage
    - CapV The super-cap voltage
    - Pressure Atmospheric pressure, in hPa (millibar)
    - PressureRate Pressure Rate of change, in Pa per second
    - Humidity Ambient humidity, in percent
    - NumSat Number of GPS Satellites currently used by the GPS module
    - Lon /Lat GPS longitude in degree decimal
    - Alt GPS altitude in meters above Ellipsoid
    - GndS GPS ground speed in m/s
    - VerS GPS vertical rate in m/s
    - HDG GPS heading in degrees
    - AccX/Y/Z Accelerometer readings in indicated axis, in g's

MaxAcc	Max Aggregate Acceleration of all 3 axis, in g’s, since last report
GyroX/Y/Z	Magnetometer readings in indicated axis, in degrees
GyroAgg	Instantanous aggregate Gyroscope reading, in °/sec
GyroFilt	Filtered aggregate Gyroscope reading, in °/sec
AvgGyro	Filtered aggregate Gyroscope reading, in °/sec

Notes:

- Unless otherwise noted, each data value is updated for every entry.
- Blue text indicates data logged only in ASCII format

The Flight.KML Data File

This file allows rapid plotting of the flight path in applications such as Google Earth.

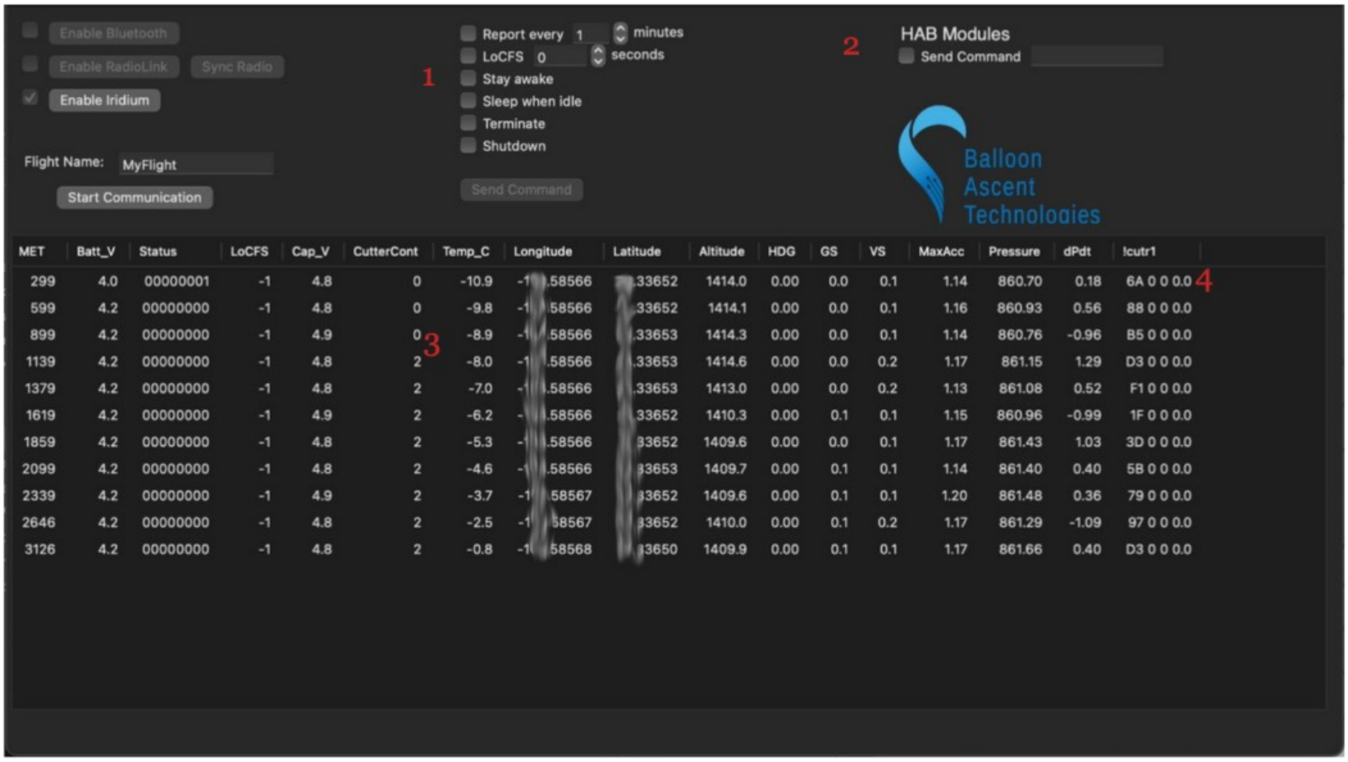
Note: it not time-stamped. Please use the values from the Flight.CSV file for data that includes a timestamp

The Flight.LOG File

This file provides a log of the system operations, commands sent and received, as well as any issues encountered and errors generated.

6 Mission Control Software

Iridium SBD data is parsed and displayed in Loonatec's Mission Control software. This software runs on MacOS, Windows, and Linux. It uses a simple interface to display relevant data and send commands. It automatically creates and updates a CSV data file for later reference and a KML file that can be used as a "Network Link" to provide realtime plotting in Google Earth Pro desktop.



□ Enable Bluetooth
□ Enable RadioLink
☑ Enable Iridium

Flight Name: MyFlight

□ Report every 1 minutes
□ LoCFS 0 seconds

MET	Batt_V	Status	LoCFS	Cap_V	CutterCont	Temp_C	Longitude	Latitude	Altitude	HDG	GS	VS	MaxAcc	Pressure	dPdt	!cutr1
299	4.0	00000001	-1	4.8	0	-10.9	-1[redacted].58566	[redacted].33652	1414.0	0.00	0.0	0.1	1.14	860.70	0.18	6A 0 0 0.0
599	4.2	00000000	-1	4.8	0	-9.8	-1[redacted].58566	[redacted].33652	1414.1	0.00	0.0	0.1	1.16	860.93	0.56	88 0 0 0.0
899	4.2	00000000	-1	4.9	0	-8.9	-1[redacted].58566	[redacted].33653	1414.3	0.00	0.0	0.1	1.14	860.76	-0.96	B5 0 0 0.0
1139	4.2	00000000	-1	4.8	2	-8.0	-1[redacted].58566	[redacted].33653	1414.6	0.00	0.0	0.2	1.17	861.15	1.29	D3 0 0 0.0
1379	4.2	00000000	-1	4.8	2	-7.0	-1[redacted].58566	[redacted].33653	1413.0	0.00	0.0	0.2	1.13	861.08	0.52	F1 0 0 0.0
1619	4.2	00000000	-1	4.9	2	-6.2	-[redacted].58566	[redacted].33652	1410.3	0.00	0.1	0.1	1.15	860.96	-0.99	1F 0 0 0.0
1859	4.2	00000000	-1	4.8	2	-5.3	-[redacted].58566	[redacted]33652	1409.6	0.00	0.0	0.1	1.17	861.43	1.03	3D 0 0 0.0
2099	4.2	00000000	-1	4.8	2	-4.6	-[redacted].58566	[redacted]33653	1409.7	0.00	0.1	0.1	1.14	861.40	0.40	5B 0 0 0.0
2339	4.2	00000000	-1	4.9	2	-3.7	-1[redacted]58567	[redacted]33652	1409.6	0.00	0.1	0.1	1.20	861.48	0.36	79 0 0 0.0
2646	4.2	00000000	-1	4.8	2	-2.5	-1[redacted]58567	[redacted]33652	1410.0	0.00	0.1	0.2	1.17	861.29	-1.09	97 0 0 0.0
3126	4.2	00000000	-1	4.8	2	-0.8	-1[redacted]58568	[redacted]33650	1409.9	0.00	0.1	0.1	1.17	861.66	0.40	D3 0 0 0.0

1. A selection of preconfigured commands that can be sent to the HAB Pilot
2. A free-form command that can be sent to the HAB Pilot, normally used to send commands directly to a HAB Module
3. The pyrotechnic squib was activated at this point, which changes the reported cutter continuity indicating that it properly fired
4. A line of data from the HAB Pilot. The !cutr1 HAB Module is currently off in the above example.
    1. The first two characters are a TX/RX counter in HEX. Note how the TX counter keeps going up as the HAB Pilot continues to send commands to it, and the RX counter keeps going down as the expected command acknowledgements are never received.
    2. The next field is the HAB Module's status byte (0 in this example)
    3. Followed by the HAB Module's temperature (0 in this example)
    4. And finally the HAB Module's battery voltage (0.0 in this example)

7 Troubleshooting

Device Will Not Turn On

The *HAB Pilot* should turn on when the power switch is moved to the *ON* position. If it doesn't, check the following:

- Check the battery orientation
- Install new batteries. The Energizer L91 batteries should not be stored in hot conditions for extended periods of time (months). In that condition they self-discharge more quickly.

Device Can Not Acquire GPS Fix

The *HAB Pilot* normally acquires GPS fix in less than 5 minutes. If it is not able to acquire a GPS fix within 20 minutes then please check:

- That it has a clear and unobstructed view of most of the sky down to the horizon
- That there are GPS satellites overhead and visible to the Pilot. https://in-the-sky.org/satmap_worldmap.php

Note: a super-capacitor backup power source was added for the GPS module in HAB Pilot version 2.2. It can provide about 10 hours of stand-by power for the GPS module which enables warm starts (ephemeris and almanac data are stored, and time is maintained). This can greatly reduce the time to first fix if the unit is tested shortly before a launch.

Device Can Not Transmit the Data Header Message

The *HAB Pilot* normally transmits the Data Header Message in the Power-On Sequence within 5 minutes. If it still hasn't sent the message after 20 minutes then please check:

- That it has a clear and unobstructed view of most of the sky down to the horizon
- That the Iridium SBD antenna is aimed upward (batteries will be down, on the bottom)
- That there are Iridium satellites overhead: https://www.groundcontrol.com/Iridium_Coverage_Map.htm

8 Consumables

Batteries

The *HAB Pilot* has been designed to use primary lithium batteries in the 'AA' size. The Lithium/Iron Disulfide (Li/FeS2) chemistry has proven to work well, even at the low temperatures experienced on balloon flights. Possible suppliers of these batteries include:

- Energizer Ultimate Lithium L91 AA Battery
- Varta Ultra Lithium 6106 (FR14505) AA Battery
    - Note: the 6106 is rated at 18% less capacity than the L91 battery (2.9Ah vs 3.5Ah)

While the *HAB Pilot* will operate on standard alkaline batteries, it is strongly recommend against using them due to their poor performance at cold.

External Power

External power can be supplied to the *HAB Pilot* via a 2-pin Molex SL connector. Power requirements are 2 to 5.5 VDC, < 0.5 amps from either additional batteries or a solar panel. If this power source is 0.5 VDC higher than the battery voltage then the batteries won't be used—they are essentially diode-or'd together.

Memory Card

Any SD/SDHD micro-SD card should work with the *HAB Pilot*. We recommend using the SD Memory Card Formatter (https://www.sdcard.org/downloads/formatter/) from the SD Association to format new cards. Using the OS formatting utility (Mac OSX or Windows) can cause problems.

Note: some cards can operate slower than expected. If this happens they will generate a "WARNING - RunLoop Timing Overflow" message in the Flight.LOG file. This is not a problem as long as it only happens occasionally, once a minute or less frequently.

9 Appendix A – Binary Data

This section is intended for users that are not using Loonatec's Mission Control software.

Overview

Binary telemetry packets from the HAB Pilot are dependent on the Flight.cfg settings and the firmware version. Each data source (ex Timestamp, System, Position, etc) requested in the *Configuration* section of the Flight.cfg file by **CommData:** adds a prescribed set of parameters to the telemetry packet. The user has control over which data sources are included in the telemetry packet but not what parameters each data source adds to the packet.

To ascertain what parameters are being sent by the HAB Pilot an ASCII comma-separated values (CSV) header can be sent at start-up. To enable this add **CommHeader: true** to the *Configuration* section of the Flight.cfg file. The first telemetry packet for a new flight from the HAB Pilot will look something like the following:

```
1,MET,BatV,Ssts,LoCFS,CapV,CutC,IntT,!vntr1
```

This header also serves to delineate multiple HAB Pilot flights in same email address, as long as it is always enabled. Software can parse through the Iridium SBD messages in reverse order until it finds the most recent header. Any newer binary messages can then be associated with that flight and decoded based on that header.

Standard Messages

Datasource	**HAB Pilot**			
Comment	The first value in the CSV header is a message format identifier. Currently only a value of ‘1’ is to be expected from HAB Pilots.			
Name	*Type*	*Scale*	*Unit*	*Description*
1	-	-	version	Binary message format

Datasource	**Timestamp**			
Comment	This datasource contains time data			
Name	*Type*	*Scale*	*Unit*	*Description*
MET	U4	1	Seconds	Mission Elapsed Time—time since start of flight

Datasource	**System**			
Comment	This datasource contains data pertaining to the HAB Pilot’s operation			
Name	*Type*	*Scale*	*Unit*	*Description*
IntT	I2	1e-2	°C	Board temperature
BatV	I2	1e-2	volts	Battery voltage
CapV	I2	1e-2	volts	Super-capacitor voltage
CutC	X1	-	-	Line continuity bit-0: CUTR-0 → 0=good, 1=fail bit-1 CUTR-1 → 0=good, 1=fail
SSts	X1	-	-	System Status bit-0: Heater active bit-1: Iridium active bit-2: GPS power save mode enabled bit-3: Heartbeat active
LoCFS	I2	1	seconds	Loss of Comm Failsafe Timer • Disabled = -1 • Values >0 indicate seconds remaining before activation

Datasource	**Pressure**			
Comment	This datasource contains atmospheric data			
Name	*Type*	*Scale*	*Unit*	*Description*
Pres	I4	1e-3	hPa	Barometric pressure
dPdt	I2	1e-2	Pa/s	Change in barometric pressure

Datasource **IMU**

Comment This datasource contains inertial measurement data

Name	*Type*	*Scale*	*Unit*	*Description*
MaxAcc	U2	1e-3	g's	Max aggregate acceleration since last report
AvgGyr	U1	1	°/s	Average combined gyro rate

Datasource **Position**

Comment This datasource contains GPS data

Name	*Type*	*Scale*	*Unit*	*Description*
Lon	I4	1e-7	deg	Longitude
Lat	I4	1e-7	deg	Latitude
Alt	I4	1e-3	meters	Altitude
Hdg	I4	1e-5	deg	Heading of motion (2-D)
GndS	I4	1e-3	m/s	Ground speed (2-D)
HorS	I4	1e-3	m/s	Horizontal speed (2-D) Note: same as Ground Speed, only one will be included
VrtS	I4	1e-3	m/s	Vertical speed (ascent rate)

Datasource **Inputs**

Comment This datasource contains external inputs to the HAB Pilot

Name	*Type*	*Scale*	*Unit*	*Description*
-	-	-	-	Not currently available on the HAB Pilot

Datasource **HAB Module**

Comment This datasource contains data from a specified HAB Module.
Multiple HAB Module datasources are possible.
Any field that starts with '!' in the CSV header is a HAB Module.
Device identifier follows the '!' and is user specified in the Flight.cfg file.

Name	*Type*	*Scale*	*Unit*	*Description*
!xxxx	X4	-	-	HAB Module identified by xxxx.
The following is how the device's four bytes are to be interpreted:				
-	X1	-	-	Communication Counters

				bit-0...bit-3: RX count Increments for each successful receipt Decrements for each missed receipt bit-4...bit7: TX count Increments on each comm attempt
-	X1	-	-	Device's status bit-0: Heater active bit-1: Limit switch 1 bit-2: Limit switch 2 bit-3: WDT active bit-4: Actuator warning bit-5: Actuator active bit-6: Device error bit-7: Charging super-capacitor
-	I1	1	°C	Device's temperature
-	U1	1e-1	volts	Device's battery voltage

Data Types

Name	*Type*	*Size (Bytes)*	*Range*
U1	Unsigned 8-bit integer	1	0...2^{8}-1
I1	Signed 8-bit integer, two's complement	1	-2^{7}...2^{7}-1
X1	8-bit bitfield	1	n/a
U2	Unsigned little-endian 16-bit integer	2	0...2^{16}-1
I2	Signed little-endian 16-bit integer, two's complement	2	-215…2^{15}-1
X2	16-bit little-endian bitfield	2	n/a
U4	Unsigned little-endian 32-bit integer	4	0...2^{32}-1
I4	Signed little-endian 32-bit integer, two's complement	4	-2^{31}...2^{31}-1
X4	32-bit little-endian bitfield	4	n/a

10 Safety Precautions and Recommendations

Regulatory

Always adhere to the regulations governing high-altitude ballooning applicable in the country of operation. In the United States, high-altitude balloon flights are governed by “FAA Part 101”.

Always adhere to the regulations for high explosives if pyrotechnic squibs are being used. Some countries restrict ownership, transportation, and use of these devices.

Touch Hazard

Burn-wires are fragile and produce high temperatures when activated. Never press-down on and try to avoid touching the burn-wires.

Falling Payload Hazard

Always clear the launch area prior to launch. While unlikely, balloon or line failure after release may result in a falling payload impacting the ground before the parachute fully deploys.

The owner shall be liable for any damages resulting from any use of the *HAB Pilot* and other related materials, and shall defend, hold harmless and indemnify Balloon Ascent Technologies LLC, officers, employees and agents, against any and all claims, suits, actions, costs, counsel fees, expenses, damages, judgments and decrees, by reason of any person or property being injured or damaged directly or indirectly by use of the *HAB Pilot* or activities arising therefrom.