Tel +41 31 974 16 66 · Fax +41 31 974 16 65 · admin@netline.ch · http://www.netline.ch

GPS Logger project

Progress Report 2007

Marc Andre, André-Netline

At the CIA Conference in 2007 it was decided in the Online Contest Working Group to define a logger device for ballooning competitions. André-Netline was assigned to build a prototype until the next conference.

We had lots of discussions after the CIA conference with competitors, officials and other persons concerned. During the summer we built an early prototype and were able to test it during the Worlds Test-Event in Hofkirchen in autumn. The focus was to test the general setup and the GPS modules. This report shows the progress of the project during this year and summarizes the feature list of the logger.

GPS-Logger Features

We had several conversations about the functions and shape of the logger with competitors and officials during events this year. We concluded on the following specification:

a) **Logger handling**
The logger should be owned by the pilot or at least handled by the pilot during a whole event (including charging, …). The goal should be that every pilot owns his own logger. Because this is currently not possible (we will loose too many competitors who don't want to buy a logger), loggers should also be available for rent (e.g. from the NAC or CIA).

b) **Data interfaces**
The pilot doesn't need to bring the whole logger back for readout after the flight. He just returns a memory chip (expectedly a SD Card) with the logger track on it. This will make the readout process much easier and faster.
In addition, the logger has a real-time USB (serial) output with NMEA data. That way, the pilot can connect the logger to his PC and use it as his flight navigation device. This will eliminate most discussions about precision between the pilots and the event GPS.

c) **User interface**
The user interface must be very simple. It allows the pilot just to do what is necessary during a flight.

d) **Security**
Security for the loggers should be introduced in a moderate way: It has to be ensured that the logger data (which is stored on the memory card in a PC-readable format) cannot be modified manually. This can be accomplished with digital signatures used in modern encryption algorithms. Details on the digital signature will be specified during development. We don't need physical security seals.

e) **FlyOn Gloals**
We had an agreement, that re-introducing FON tasks in logger competitions is a big advantage. The logger will have a keypad and a display that allows the pilot to enter goals during the flight. The goals are saved together with a timestamp that ensures the goal was entered before a specific task.

f) **Accuracy / Logger errors**
Several tasks at the Europeans this year (especially the maximum distances tasks) have shown that with good conditions the pilots sometimes fly better than the measurement accuracy. 5sec interval at 30kph results in 40m distance between two track points. The logger interval must be decreased to 1s with still having enough logger time for at least 5h operation (at least 20'000 logger points).
With the current loggers (especially the Garmin eTrex series), many events face the problem of bad tracks (spikes, off track recordings).This problem can be reduced with newer and better GPS chips. (See also the technical discussion bellow)

g) **Logger button**
I heard a lot of voices complaining that the current system with automatically scoring the best track point in a given task removes the obligation from the pilot to make a decision. (When do I

drop the marker? Will I get closer later on?). The logger should have several buttons allowing the drop of virtual markers. The loggers can still be used without buttons if necessary.

h) **Live Tracking**
Live tracking is not the main focus of the logger. But because it is of increasing interest to provide live coverage (WAG) the logger will implement at least an output for GPRS-Modems.

i) **Logger configuration (Task definition)**
In early Logger specifications we talked about integrating a solution, that the logger "knows" about the tasks to be flown. We removed this point because we found, that it can produce lots of errors and misunderstandings. It is much more complicated if for example a task is altered or cancelled in a supplementary briefing. We now think the best way is to configure a logger for a specific event (coordinate formats, …) but not for a specific flight.

User Interface / Handling

The logger will be mounted at a pole between basket and burner. At this position GPS coverage should be good and the display is clearly visible. It is easy for the pilot to operate the device and enter data.
The user interface should be very simple. It doesn't allow the pilot to (accidentally) alter important settings. All settings are configured in a file that is stored on the memory chip. The display is a simple multi-line text-display. The user has a num-pad and some additional keys.
The following actions are possible:

- **Idle-Screen**
  In idle screen the pilot can see his position and altitude in the competition's format (UTM or Lat/Long). The display also shows the current time and the number of satellites in use.
- **Marker drop**
  A virtual marker drop can be done with a combination of a key and the number of the marker to drop. In that way no accidental marker drop is possible. The markers have numbers to make scoring easier (similar to marker colors). The logger shows if a marker has already been dropped. A secondary drop is still possible to solve issues where the pilot accidentally dropped markers before the flight.
- **Declare goal**
  A goal can be declared by entering the marker coordinates. The logger helps the user to enter the data in a properly format but in general any format is accepted. It is important to reduce problems that might delay or confuse the pilot when entering data to the logger.

Technical concept

System build-up

Based on the discussions during the CIA conference and the feature list I evaluated a platform to build the logger on. Because the quantity to build is pretty small it doesn't make sense to build a completely new system. The system should be modular. We need at least the following modules:

- Keypad and buttons
- Display
- GPS receiver
- Memory card interface
- Processor
- USB interface

As the system must be battery powered for at least 5 to 10h power consumption is also an important factor.

Processor

The main topics of the processor are:

- GPS handling
- storing the track on the memory card
- handling the user interface
- Converting the coordinates to a user friendly format (UTM)
- Calculating the digital signature

I decided to work on the basis of an AVR32 microcontroller. This controller is brand new and builds on the newest processor concepts. Its power consumption is very low but it still has enough computing power to convert the coordinates and calculate the digital signature.

GPS Module
Nowadays a lot of different GPS sensors are available. Most GPS modules have 12 to 20 channel receivers. They evaluate the position based on the received signal and in some regions they include the correction signals (WAAS and EGNOS). SIRF III is the most popular GPS receiver design and has a very good review. Some other companies have their own GPS receivers like u-blox with ANTARIS 4 and u-blox 5.

As long as the GPS coverage is good (most conditions in hot air balloons) most GPS show a similar good position. The differences start when the reception gets bad. In these situations the GPS need to make assumptions. The approaches for that (called dead reckoning) are different between the manufacturers. Because most GPS are built to operate in 2D (cars, handy) they sometimes handle it badly for our application. Most Garmin GPS do dead reckoning by assuming a smooth straight on motion. We often observed errors in tracks introduced by this procedure: The position is assumed by the GPS to move straight on after loosing coverage and suddenly jumps back to the correct position after retaining coverage.
A good dead reckoning works with acceleration sensor data as input for improving the GPS track. There are GPS modules available with this technique (uBlox LEA-4R, SirfDiRect) but the logger would be much more expensive (at least +50%).

So far we only tested devices without dead reckoning (neither with bad straight on motion nor with acceleration sensors). We implemented 3 different GPS modules for testing (u-blox LEA-4H, Fastrax uPatch100 and Falcom FSA01). In general these modules showed a much better track than the current event loggers. We only had problems in cases when reception was explicitly bad (logger between two gas tanks on the ground of the basket). We couldn't finish the evaluation of the GPS modules yet. We will decide which GPS module will be implemented later on. In order to keep the electronic design easy and the PCB production cheap I suggest using a GPS module with integrated antenna.

Altitude measuring
Barometric altitude is important for current GPS loggers because barometric altitude is in general much better than GPS altitude. Even tough the discussion is not completely finished I suggest using GPS altitude for the new loggers. The main arguments are:

- As the new logger can be used as an input device for the navigation and it shows the recorded altitude on the display the pilot knows on which altitude he is tracked.
- Handling barometric altitude with displaying the correct altitude to the pilot is difficult. Currently altitude is recorded with QNH 1013.5 and corrected to the real QNH during scoring. This wouldn't be possible anymore because we would need to show the real altitude to the pilot during flight. Entering the QNH in the logger is also not an option because it is source to lots of errors and complaints. (What happens if the pilot enters the wrong QNH?)

The main disadvantage of using GPS altitude is that the altitude between two balloons may be different allowing a balloon to fly a bit higher/lower than another balloon.

Cost estimation / Production

It is still difficult to estimate the total cost of a device. Producing 100 devices is still a very small quantity in hardware business. The main costs aren't the electronic devices but the chassis, the keyboard, the display and of course the cost of manufacturing.
The cost can be heavily influenced by choosing cheaper chassis and keypads. I think it is in the interests of all to have a good quality product that holds for a longer time.
The batch size is also an important factor. I am sure that the costs of a device we will be bellow a 1000 Euro mark.

We (Andre-Netline) cannot produce the logger ourselves. We are specialised in developing the electronics and software and we can build prototypes. We need a production partner that can build the devices in close cooperation with us.

We are currently evaluating several possible partners for the production. I still hope that maybe we can find a partner that sponsors at least some costs of the production. Suggestions are welcome. Another solution could be to build the loggers overseas.

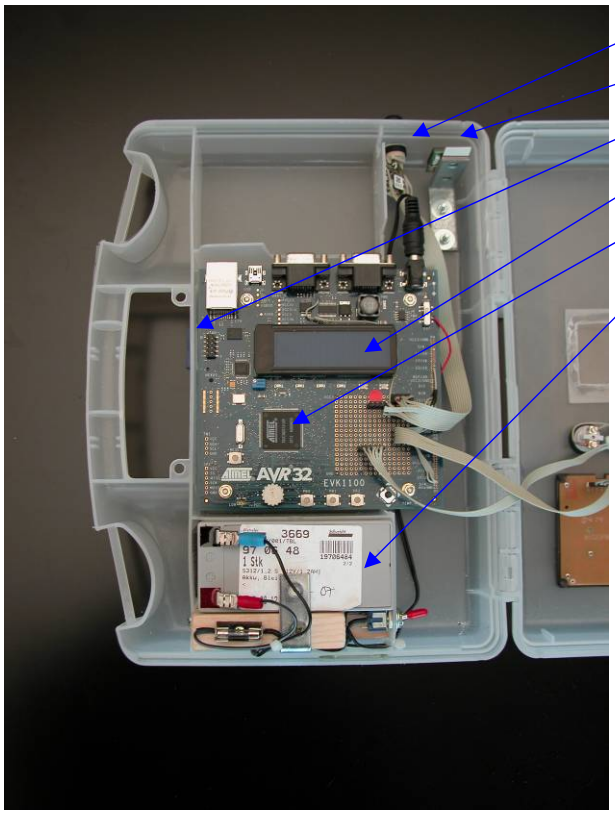
The logger costs can be defined as soon as a partner has been selected and a quote was made.

State of development

The following topics have been done so far:

1. Evaluation of concept and modules
2. Implementation of an early prototype based on a AVR32 evaluation board and several different GPS modules. (See pictures bellow)
3. Development of a simple logging device in order to test the GPS modules. The device already stores the data on a SD-card with FAT file system.
4. Testing of the GPS modules in car and hot air balloons.
5. Begin of the development of the real logger firmware with implementation of a real-time operating system.
6. Concepts for user-interface

The prototype logger is built in a prototype chassis. The display is visible trough a window.
The keypad and the additional keys are clearly visible.
The SD-Card is accessible from outside (on the left, not visible)



Next steps:

1. Development of the firmware and implementation of the user interface
2. Evaluation of production partner
3. Design of custom electronics based on the prototype → Production prototype
4. Design of final product
5. Implementation in scoring software
6. Tests, tests, tests…

Expenses for development (in CHF)

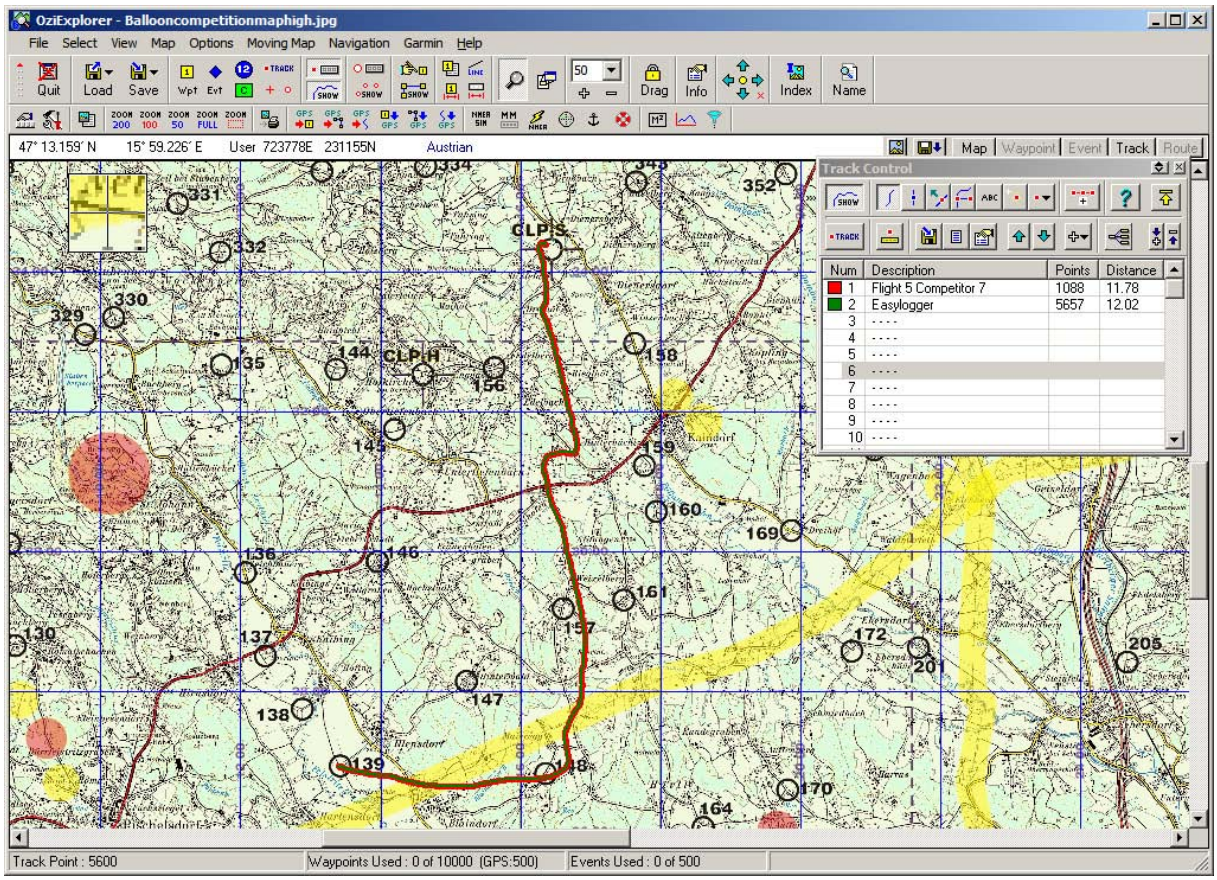
ATJTAGICE (JTAG programmer for AVR32)	420
ATEVK110 (AVR32 development board)	180
Falcom FSA01 (SIRF III GPS Module)	90
Fastrax uPatch100 (SIRF III GPS Module)	160
uBlox AEK-4H (GPS Module evaluation)	290
Misc material	100
Total	**1240**

Test results

In total we could do 10 flights with the u-blox GPS. The SIRF III GPS could only be tested on the ground so far.
In general situations the u-blox GPS behaved as expected. But in situation where the GPS coverage was explicitly bad (GPS on the ground of the basked jammed between two gas tanks) it also produced bad tracks.

I plan to do more test flights in the winter with the Sirf III chips before I can definitively decide on the GPS modules. It is hard to compare them in detail, because we have just one logger and thus I can just test one device in a flight. Our focus is to select the modul which produces a very good and robust GPS track and finds the satellites fast.

The screen shot shows a recorded track during the Worlds Test-Event in Hofkirchen this year. The red track is the eTrex (event) GPS and the green line on top of the red track is the evaluation logger.


OziExplorer - Balloncompetitionmaphigh.jpg
File Select View Map Options Moving Map Navigation Garmin Help
Quit Load Save Drag Info Index Name
47° 13.159' N 15° 59.226' E User 723778E 231155N Austrian
Map Waypoint Event Track Route
Track Control
Num Description Points Distance
1 Flight 5 Competitor 7 1088 11.78
2 Easylogger 5657 12.02
3 ----
4 ----
5 ----
6 ----
7 ----
8 ----
9 ----
10 ----
CLPiS
CLPiH
Track Point : 5600
Waypoints Used : 0 of 10000 (GPS:500)
Events Used : 0 of 500