PWR AEROSPACE

ORION APP4HAB LAUNCH SUMMARY

High Altitude Balloon project
Designed and executed by PWr Aerospace
Wroclaw University of Science and Technology

21.04.2018

PWR AEROSPACE

Contents

Introduction

ORION is a high altitude balloon program developed by students of Wroclaw University of Science and Technology. Its main goal is to provide basic skills and knowledge needed to design big HAB projects in the future.

PWr Aerospace performed three HAB projects:

- ORION ONE
- ORION TWO
- ORION APP4HAB

ORION APP4HAB is aiming to provide an easy way to perform a high altitude flight for people with no skills in electronics and telecommunication. The goal is to design an application on Android operating smartphone that will utilize the possibilities this mobile computer provides.

This document covers the design and post-flight analysis of first experimental APP4HAB mission, which launched 21.04.2018

PWR AEROSPACE

Mission summary

	Expected	Achieved
Altitude	28 – 31km	26 682m
Temperature inside	4°C	2.8°C
Flight duration	2h	2h
Landing position relative to launch site	95km ESE	86km ESE

Onboard devices:

– Samsung Galaxy S8
– Data unit based on Arduino Mega
– Outside sensors board
– Xiaoyi 4k camera
– VHF FM transmitter
– 7.4V 3.9Ah 2s Li-Po battery
– 3.7V 1.8Ah 1s Li-Po battery
– Vaisala RS41 radiosonde with custom software as backup location tracker

Notable time events:

Day of launch: 21.04.2018 (time zone UTC +2)

Start of launch preparations:	8:33
Launch:	10:35
Balloon burst:	12:08
Landing:	12:35
Payload pickup:	12:38

Launch position:	Wroclaw Sky Club Airfield, 51.2048 N 16.9934 E
Landing position:	51.028691 N 18.200041 E

Balloon used:	CPR-1000
Gas used:	Helium
Gas volume:	3.3 m^3
Parachute used:	Klima GmbH 70cm
Payload mass:	1.8 kg

Launch weather conditions:

Temperature:	23°C
Barometric pressure:	1004 hPa
Wind speed:	0 m/s
Cloud layer:	None
Visibility:	Excellent

Launch details

We launched during very favorable weather conditions, with no wind and clear skies. We used CPR-1000 balloon filled with helium. We used 20l helium tank, which gave us around 2.3kg of net lift. The parachute we used was Klima GmbH 70cm.

Using bigger balloon and the same amount of helium we expected the burst altitude to be higher than in our previous flight, in which we used 800g balloon and reached 27 544m. Unfortunately it did not happen, as achieved altitude was about 900m lower. The most possible cause of that was difference between manufacturers of those two balloons – Hwoyee used in previous one and CPR used in this.

APP4HAB system

The main goal of the project was to develop an android app that will allow people to easily collect, store and process data gathered during high altitude flight. Basic functionality of our app includes:

- collecting data regarding atmospheric and device conditions
- taking pictures during flight
- sending position of the balloon to the ground team

Android phones are not designed for high altitude flights and we had to solve a few interesting problems during planning and development.

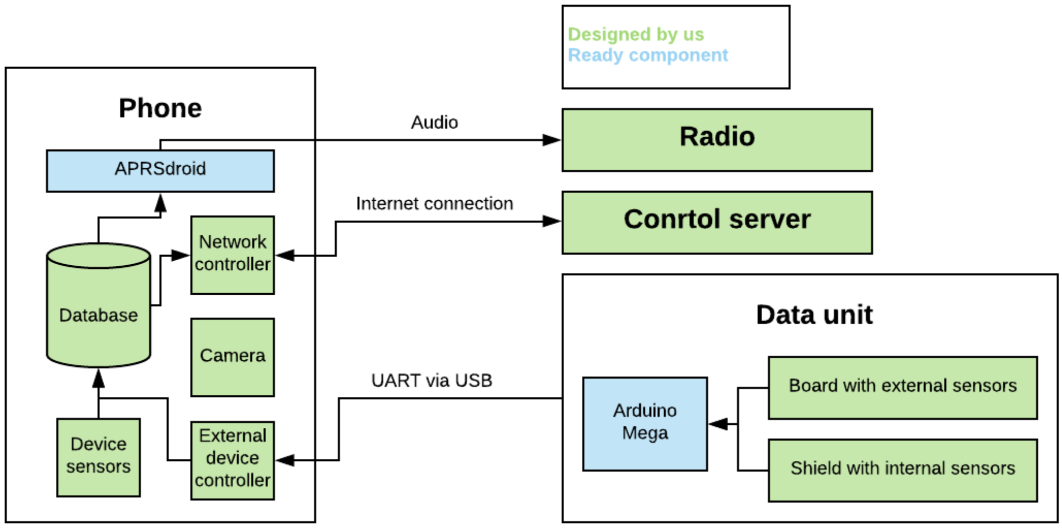
While first two of the above functionalities can be described as regular phone activities and can be easily implemented, third one is hard to achieve during high altitude balloon flight because of no GSM connection during most of the flight. For that we decided to use an open source app called APRSdroid (https://aprsdroid.org) to prepare APRS packets with balloon position and transfer it to the onboard radio using AFSK (Audio FSK). This allowed us to connect the radio and smartphone using audio cable and transmitting phone position using APRS.

Using device sensors only the collected data would be very limited to only a few interesting parameters that phone sensors can provide. We would not be able to monitor outside temperature, light intensity, air temperature inside the capsule, etc. We decided to design a board that will log all those and even more parameters and send them to the phone. Hence the phone should allow communication with external devices providing more data to collect. The easiest way to connect this kind of device to the phone is to use the USB port as virtual serial port.

Because of using the USB port to connect to the external device, we could not be using it to charge the phone during flight, the only source of power for the phone during whole flight would be its internal battery. To reduce the power consumption of the app we decided to allow switching on and off parts of app functionalities. For that we decided to use a server which the phone will connect to using internet. Thanks to that we could monitor the app behavior before start and receive data collected during flight, as long as the phone keeps internet connection.

All schemas and source code will be released as open source together with extensive documentation we are currently working on.

The overall architecture of the system is presented on the schema below:



APP4HAB Android app

The application fulfilling all mentioned above requirements was developed from scratch by members of our team. We have created a 5 students team of both experienced and starting Android programmers who designed, implemented and tested the app.

Among the technologies used for the app were:

- *Kotlin* as the programming language
- *Retrofit* for HTTP communication with control server
- *Dagger* for dependency injection
- *Anko* for asynchronous tasks management
- *SQLite* and *Room* as database system and access
- *USBserial* for communication with data unit

The phone used in project was Samsung S8 provided by our partner, Tieto Poland.

The app has no user interface as it was controlled remotely by the control server, but we plan to add it before opening the app source.

The app was developed following Scrum methodology and using Trello as project management application. The team development process was supervised by dr Zbigniew Szpunar, our Team Project teacher, as the app was also our final project.

The app worked flawlessly during most of the flight. The only issue we have encountered was shutting the app off half hour before landing and turning back on after landing. The cause of that is under investigation.

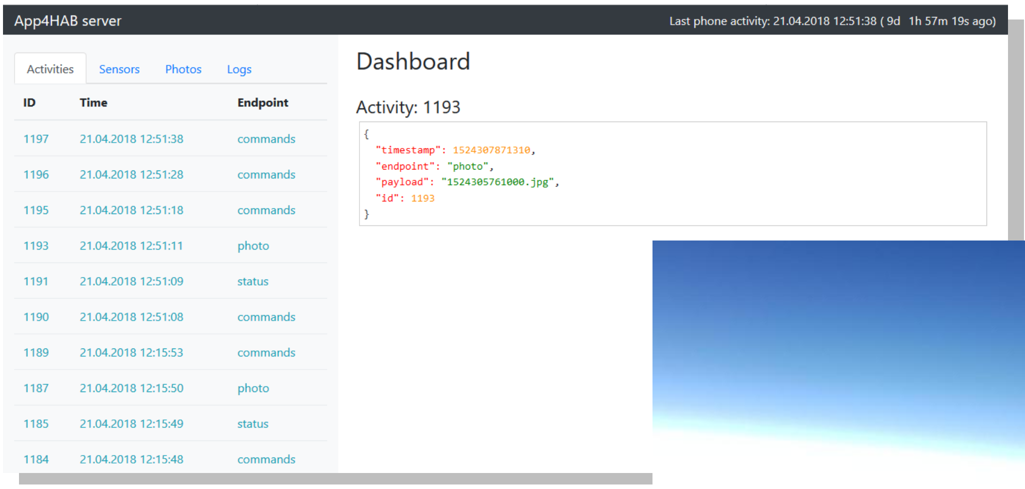
PWR AEROSPACE

APP4HAB control server

The main purpose of the control server is to provide remote access to the app, allowing turning on most of the app functionalities right before start, saving battery power during launch preparations. Additionally, we can check the system health and recorded data before and right after start, when the phone still has access to the internet.

It's a simple REST API with user interface, allowing sending commands to the phone and receive its health status.

We expected the phone to keep connection with the server to the altitude of around 3km, however the last packet received by the server during ascend was send at the altitude of 7030m, although above 2km the connection could not be considered stable. After burst the phone managed to connect to the server and upload last taken photo at the altitude of 13500m! The photo quality is low, because the phone camera had trouble picking correct picture parameters during very dynamic falling back to the surface.



Control server user interface

Photo received directly from 13km

APP4HAB Data unit

For collecting data that phone sensors could not provide we have designed a separate data unit based on Arduino Mega. It can be split into four main parts:

- Arduino Mega as the controller
- custom Arduino shield with sensors placed inside the capsule
- external board placed outside the capsule
- LoRa transmitter

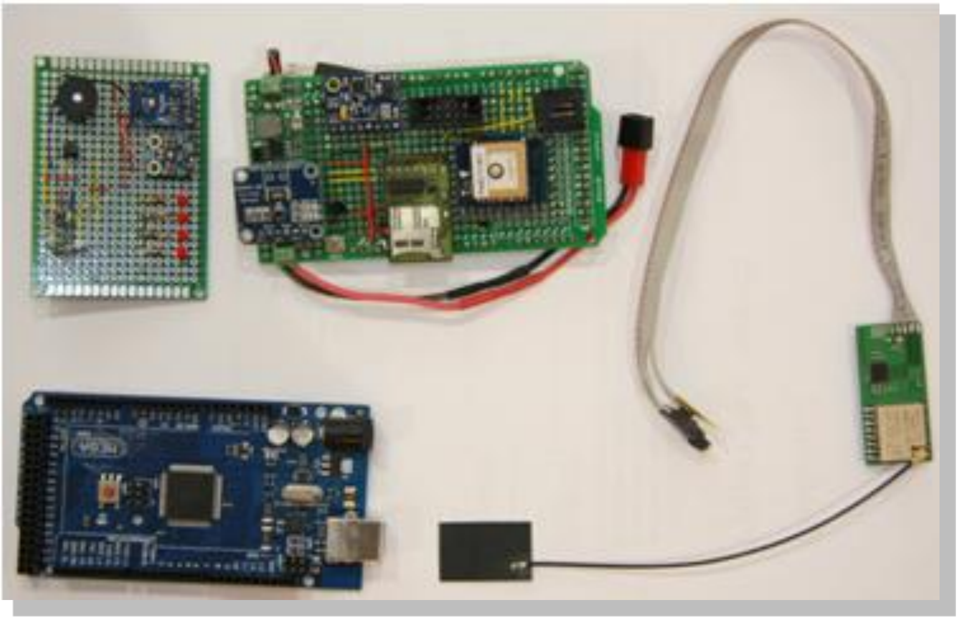
The data unit is designed to be a standalone project, which could be used without phone connected to it. It is powered by its own battery, records collected data on the onboard microSD card and sends it to our mobile receiver via LoRa radio at 70cm band. Moreover every second it sends all sensors readings to the phone using UART following the NMEA 0183 standard.

The data unit monitors:

- position using GPS
- air temperature inside the capsule
- atmospheric air temperature
- atmospheric pressure
- battery voltage
- current usage
- acceleration
- UV light intensity
- visible light intensity

The LoRa transmitter was designed and built by one of our team members. It can be used in standalone mode or with external master board as UART "extension cord". This module needs only 4μA of power in sleep mode.

We also used 4 LEDs and piezo transducer to monitor the board status without need to connect a computer to it. LED's provide information about any onboard errors, and buzzer is used to help us find the device after landing.

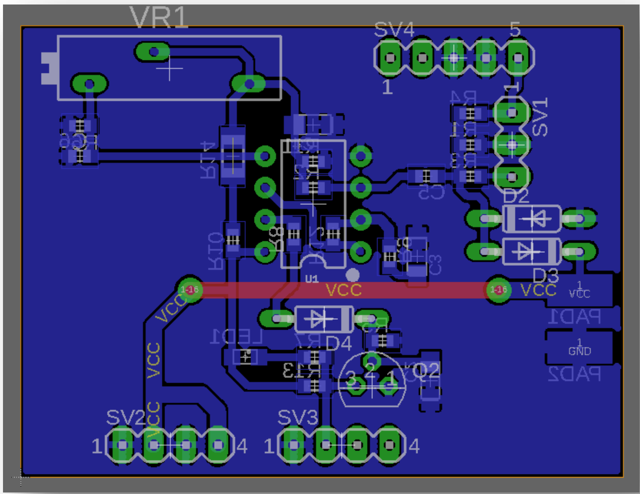


Arduino Mega board, our shield, external board and LoRa transmitter with microstrip antenna

APP4HAB Radio

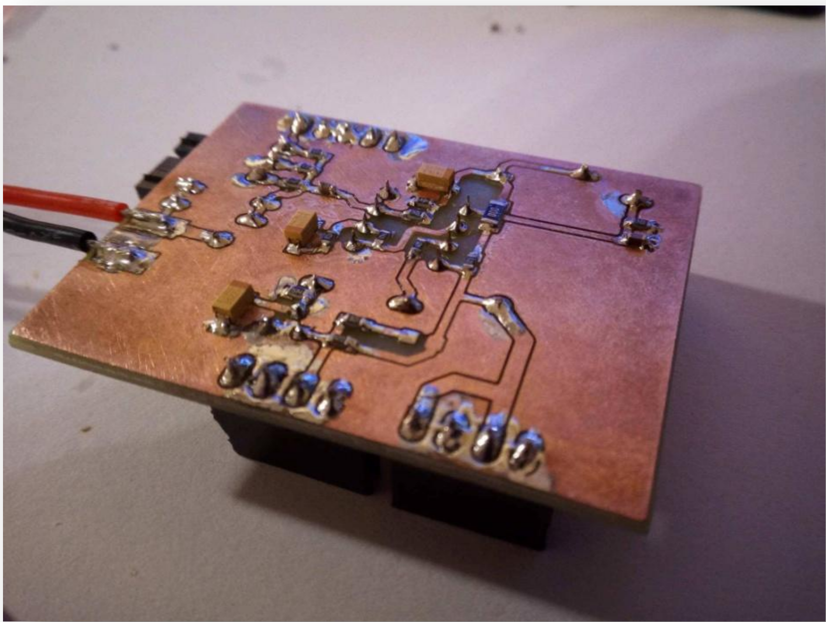
Our radio board is based on Dorji DRA818V module with our custom audio detection circuit. Radio module provides 0.5W of constant output power at 2m band to achieve the highest possible range without any special power supply system.

Radio board design



Audio detection board is used to control PTT in radio module. Without this improvement Dorji radio can't be used with simple smatphone to transmit APRS data.

PCB was designed in Autodesk Eagle and made on CNC milling machine. After that we soldered every part to the PCB and we made "sandwich" from these two boards.



Ready radio board

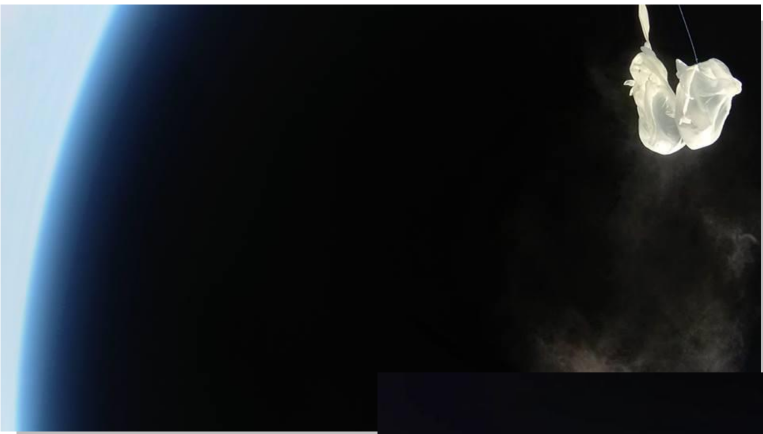
Camera

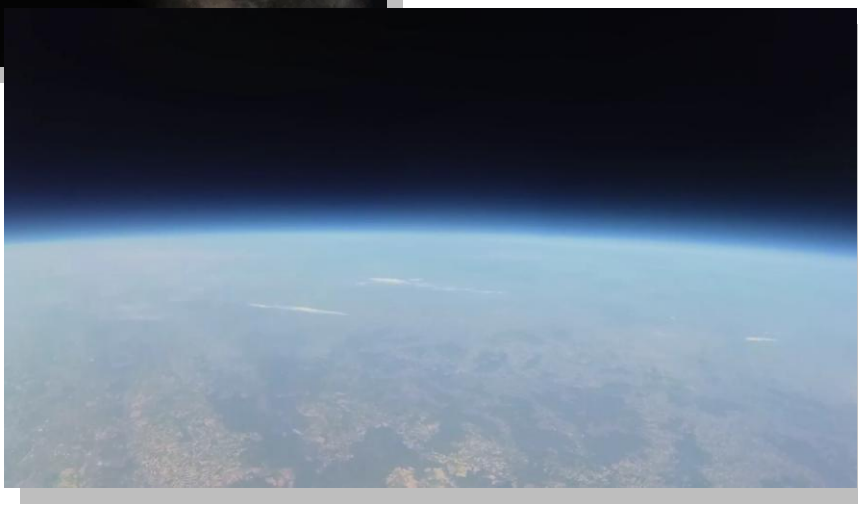
Additionally to the phone camera, flight was also recorded by Xiaoyi 4K Action camera. It was placed on the opposite side of the capsule looking at the horizon. It was set to record a full HD 1080p 60fps video.

We have reduced the field of view to minimize distortions of the image we have seen in our previous flight. Despite that, the effect of concave earth was still present, just to a lesser extent.

The camera was powered by its own battery, but also it was plugged to on board power supply providing 5V to charge the device during whole flight.

We managed to capture 152 minutes of video, covering start, burst, landing and payload pickup.





Vaisala RS41

As the backup GPS tracker we used Vaisala RS41 radiosonde. This device is used by Institute Of Meteorology And Water Management to conduct daily atmosphere sounding and preparing weather forecasts. IMWM sends two of them daily, each one of them contains sheet of paper with information about their use and disposal instructions for finder – IMWM does not require sending them back, as all information collected by radiosonde is transmitted to ground station during flight.

We managed to capture a few of them, and reprogram them to use open ISM band instead of IMWM band. It was sending its position via APRS using 432.500MHz frequency in 70cm radioamateur band.

More about Vaisala RS41 radiosonde can be found in producer datasheet:

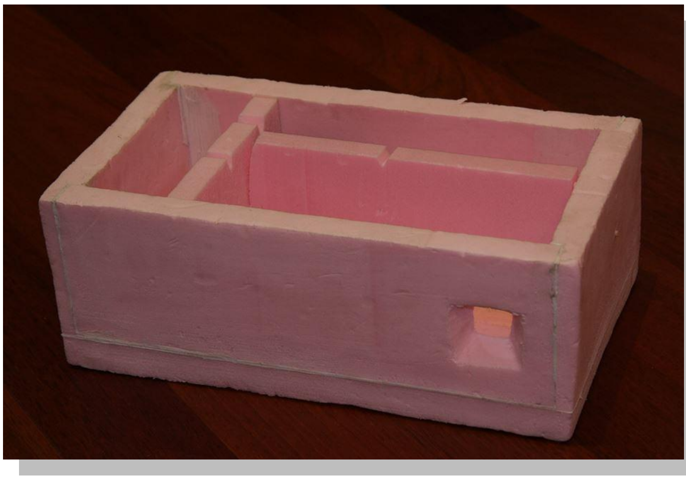
http://www.vaisala.com/Vaisala%20Documents/Brochures%20and%20Datasheets/WEA-MET-RS41-Datasheet-B211321EN.pdf.

Capsule

We reused the design from the previous flight, as it was field-tested and proved its reliability.

The capsule was made of extruded Styrofoam slabs glued together in the shape of cuboid of size 16x27x12cm. To reduce mass of glue we decided to join slabs using Mortise and Tenon Joint. This greatly increased join surface area and allowed us to use less glue. Using extruded Styrofoam as construction material worked great, as it's very easy to process and provides very good thermal isolation properties. In previous flight we have decided to protect the brittle material from strings used to tie the payload train from damage by placing them in small plastic tubes. This was unnecessary and we decided to not do this in this flight.

All components were held in place by very strong tape. We decided to not use Velcro this time, because it allows components to move a bit, which we could not tolerate in this mission, as we had much less space inside the capsule.



Afterword

This project could not become possible without support from our university – Wroclaw University of Science and Technology, and our supervisor - Associate Professor Paweł Kabacik.

Students most involved in project:

- Grzegorz Kowalik
- Jakub Pal
- Łukasz Skwarszczow
- Przemysław Materna
- Łukasz Burcon
- Damian Niemiec
- Bartek Jakubczak
- Artur Sawicki
- Marek Figlarz
- Hubert Hodowaniec
- Jakub Filipowicz
- Jakub Wojtylak
- Karol Pałac
- Miłosz Chlebowski

We want to thank our sponsors and partners, who helped us to make this project successful, especially Tieto Poland, TME Transfer Multisort Elektronik, YI Technologies, Fundacja MANUS and Wroclaw Sky Club.