Apex II Technical Guide

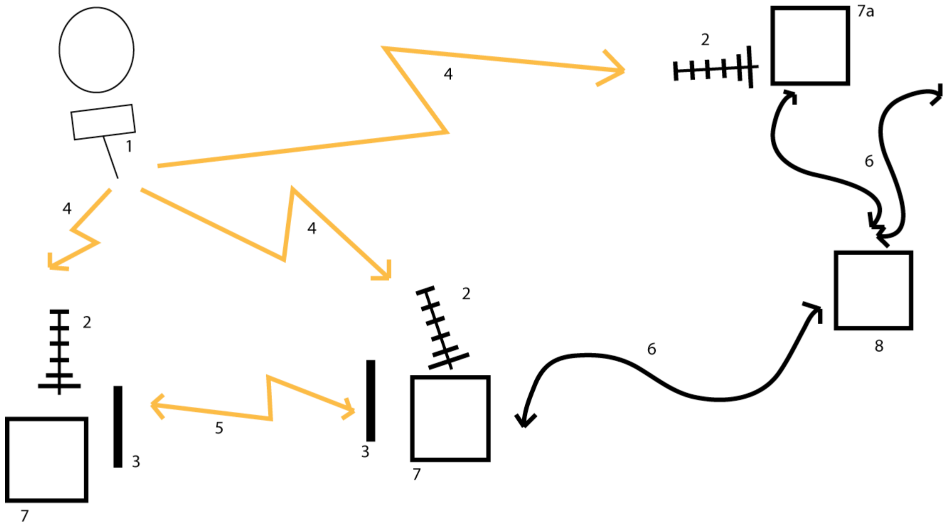
28/06/2010

Matt Brejza
Jon Sowman
Ben Oxley

Links & Resources

- Code Repository
  http://github.com/mattbrejza/APEX
- Image Gallery
  http://balloon.hexoc.com/gallery/apex-ii
- Main PIC Programming Guide
  http://balloon.hexoc.com/media/apex-programming-guide.pdf
- Uplink protocol specification
  http://balloon.hexoc.com/media/comms-proto.pdf
- Main PIC Programming Guide
  http://balloon.hexoc.com/media/pic-programming-guide.pdf

Systems Overview



1. Balloon 434 antenna
2. Base station 434 antenna (usually Yagi)
3. 144 antenna
4. 434 balloon link
5. 144 APRS link
6. Internet link (mobile broadband where appropriate)
7. Balloon tracking station
    a. Other client of the DL
8. DL (distributed listener) server

The balloon transmits on 434.075MHz (1) and is received by base stations (2). The data is processed by the tracking station computer (7). The tracking stations operated by us will echo the data on 144.8MHz, the APRS ham frequency. This allows base stations to share data, for hams to listen to the balloons progress, and data is picked up by internet connected APRS basestations (iGates) which then display the data on sites such as findu.com

Other basestations listen on the 434 band for balloon launches (7a) and are part of the distributed listener for balloons (DL). These stations report to the DL server (8), along with any of our balloon trackers (7) with internet access through mobile broadband. These base stations are also able to retrieve data from the DL server and display the data along with our own received data, and that shared through the APRS link.

Data transmission modes

FSK

Serial data is fed directly into the transmitter data input, meaning data is represented by a shift in the frequency of the carrier wave. If transmitted and received with a FM radio, the data in is the same as that going out. If the data is transmitted on FM, and received on SSB (AM radio with suppressed carrier and one suppressed sideband), the SSB radio will output tones representing the difference between the centre frequency to which the radio is tuned, and that which is being received. The signal can then be decoded in the same way as AFSK. This is the mode used by DL clients, dl-fldigi can decode it as 50 baud RTTY.

OOK

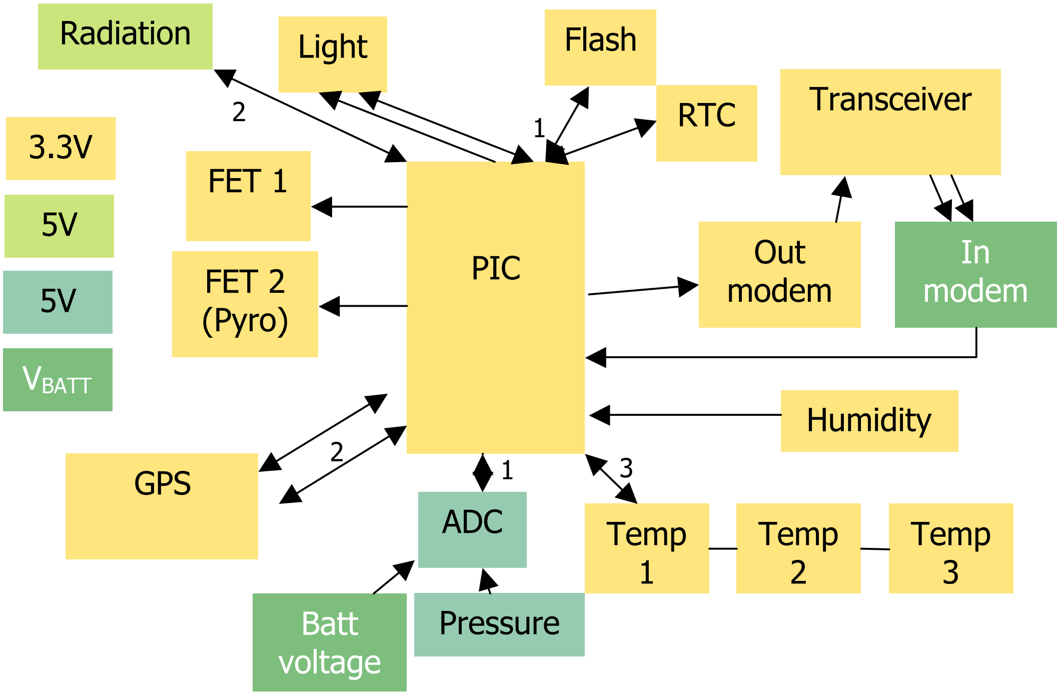
OOK – on/off keying. A '1' is represented by sending a signal, while a '0' is represented by nothing. This will be used to send data to the balloon, as the demodulator is simpler, and if a tone if sent for a '1,' then is becomes less prone to noise.

Data Encoding

The data sent from the balloon will be in the UKHAS format so we can use the DL. See here for more details. In the <custom data> part will be sensor data, as well as checksums for each individual part of the packet. The actual format of this part of the packet is undecided, and will probably be based on APEX I. The proposed format for the uplink to the balloon can be found here.

The packet will be transmitted using RTTY, which uses the FSK radio mode as explained above. To be understood by the rest of the DL, 50baud, 8bits/char, no parity, 1 stop, although this is flexible. The upload will be OOK of a similar specification.

Balloon Payload

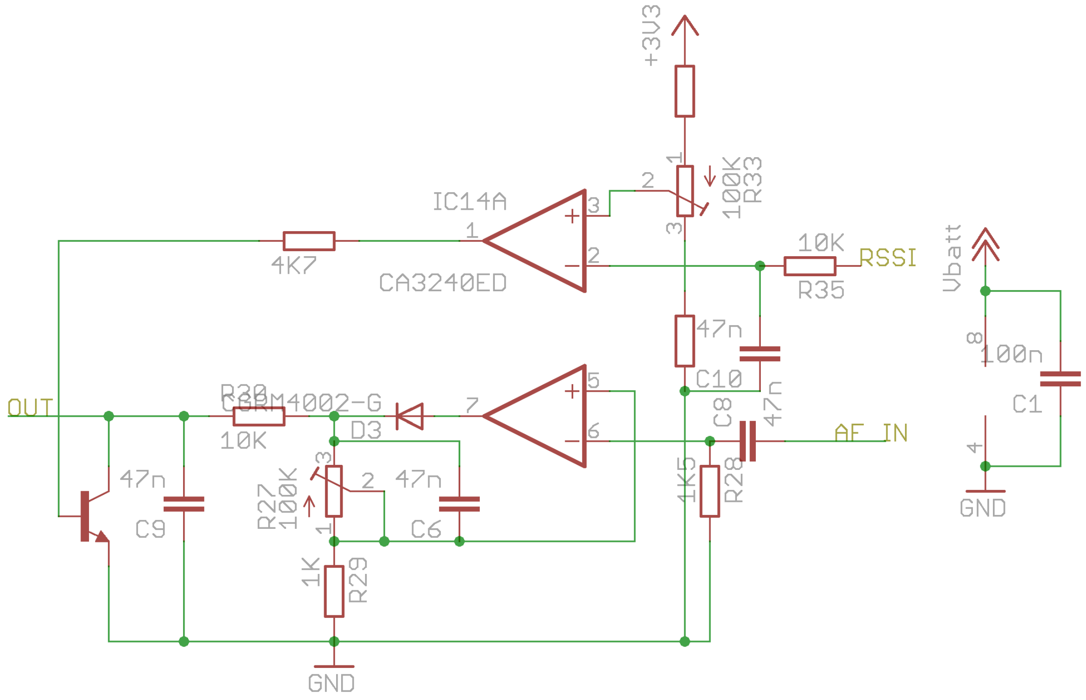


1. SPI bus (ADC shares CLK/MOSI with other SPI bus), the light sensor has extra signalling, and is not a SPI device, but shares the bus
2. Serial link
3. One wire bus

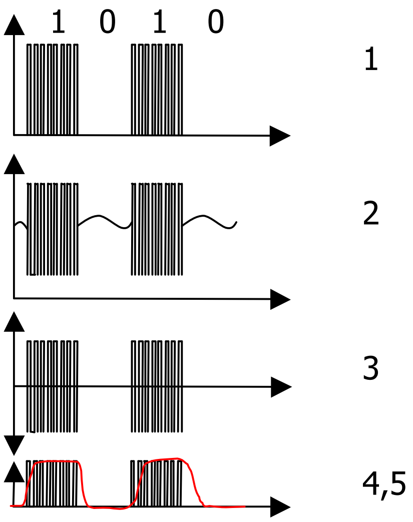
The colours of the boxes represent the voltages used to power the device and in the signalling by the device. The blue 5V devices are powered by a precision reference device

The FETs switch devices connected to the battery. The radiation units are on different PCBs to the remainder of the circuit.

In modem (demodulator) (balloon)



Opamps not necessarily CA3240, must be rail-to-rail O/P.

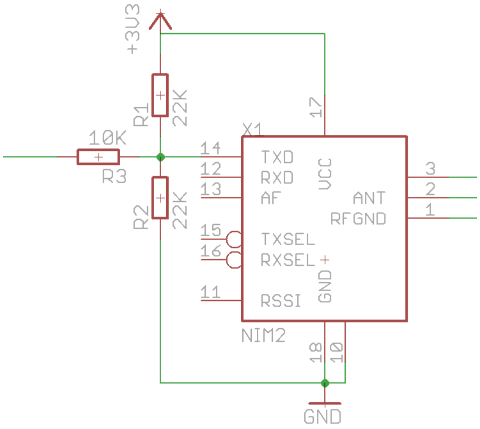


1. The transmitter modem uses a tone (~2.5kHz) as 1, nothing for 0.
2. After being transmitted and received, there is a DC component and low frequency components (~100Hz)
3. These are removed by the RC high pass filter on the I/P
4. This is then rectified (half wave), and amplified by the opamp.
5. The low pass on the opamp O/P forms a envelope detector, which passes the red waveform to the PIC

The RSSI O/P of the receiver disables the demodulator O/P when there is no signal (squelch). Initially there is a low pass filter, then the signal is then compared against the threshold voltage by the other half of the opamp. When there is not a received signal, the transistor is turned on, which holds the O/P low.

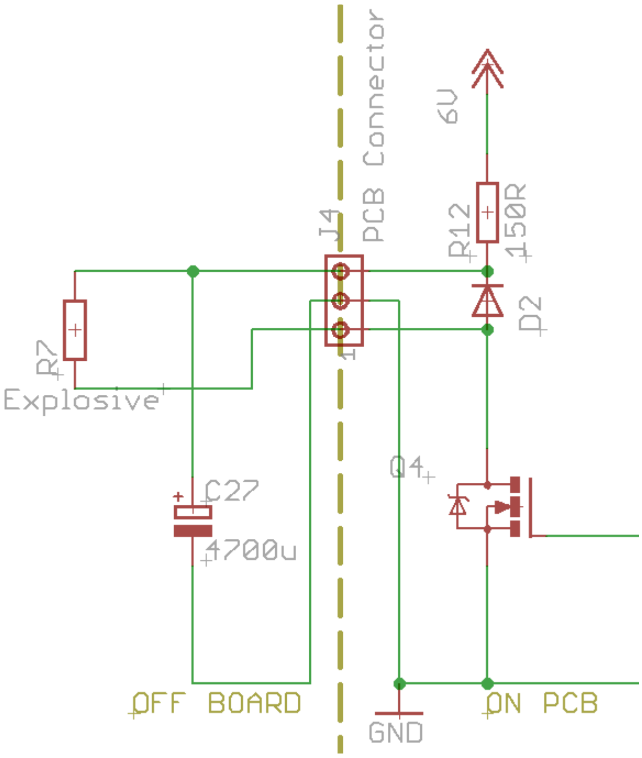
The threshold of the squelch should be set at -120 -> -123 dBm, or 0.5-0.6V, see http://www.radiometrix.co.uk/products/nim2/nim2_rssi.gif

Out modem (modulator) (balloon) - RTTY



The three resistors R1-R3 set the two voltages, V_{LO} and V_{HI}, for RTTY. Resistors R1 and R2 set the centre voltage of V_{LO} and V_{HI} and the voltage can be calculated as a potential divider, however, the input of the radio has the equivalent of a 100K resistor to GND, which needs to be taken into account. The amount the two voltages drift from the centre frequency is set by R3. On the final PCB, these resistors may be changed to two variable resistors, one to set the centre frequency, and one for the deviation.

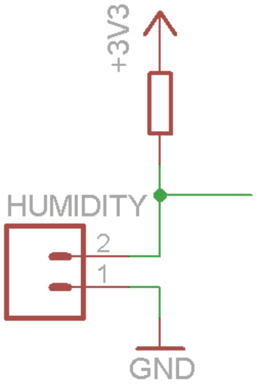
Cutdown Device



As with Apex I, although never used, the base station will have the ability to tell the payload to cut the balloon free and begin its descent (due to unexpected wind etc..). The chosen method this time is to use an explosive charge normally used to ignite rocket motors. This is still in the testing and development stage, but is something that is reported to work well. Information can be found here. When testing the device, it should be tested to see if it can be detonated from a 4700µF capacitor, as shown in the link. The FET circuitry is already present on the main board, although it can be also tested by just charging the capacitor against a 6V battery, then touching the terminals of the capacitor against the explosive's wires. The capacitor should be connected to the explosive charge off board, as shown in the diagram.

Humidity sensor

The main board has provisions for a capacitive humidity sensor, wired as shown below.



This circuit attaches to a PIC ADC pin, which also has the ability to be an output. The pin is held low, so the humidity sensor is discharged. Then the pin is turned into an analogue input, and so the capacitor can charge. The PIC either times how long the humidity sensor takes to charge to a certain level, or what the voltage is after a certain time. This value can then be translated to a value for humidity by the program on the ground.

Camera Control (CHDK) – Canon Ixus 40 (SD300)

Canon cameras are used, as their firmware can be replaced with custom firmware. We will be using CHDK as custom firmware, which allows for scripts to be run automating all camera functions. Whether the cameras are to be powered b their own batteries or a regulator on the main PCB are yet undecided.

The cameras will be running scripts to take pictures at set intervals, though the exact interval duration remains to be decided pending battery lifetime testing. See the camera test files in the Documents/ directory of the code repo.

Code is in place for the cameras to focus manually to infinity before taking pictures – something we did not implement on Apex I.

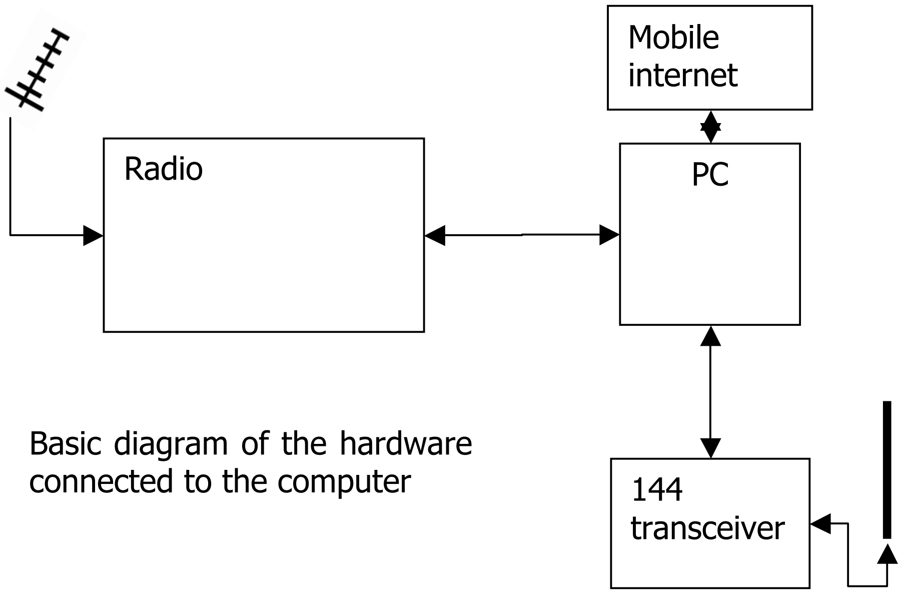
Phone

A mobile phone will be on the payload as a backup, which will text the payloads position when about to land. The phone should get no signal above 2-3km, but if possible, the radio of the phone should be turned off when in the air, either by turning off the entire phone, providing it can be turned back on through the interface, or by using AT commands to turn off just the radio. For simplicity, a phone with a serial UART is needed as they are much easier to interface with a PIC.

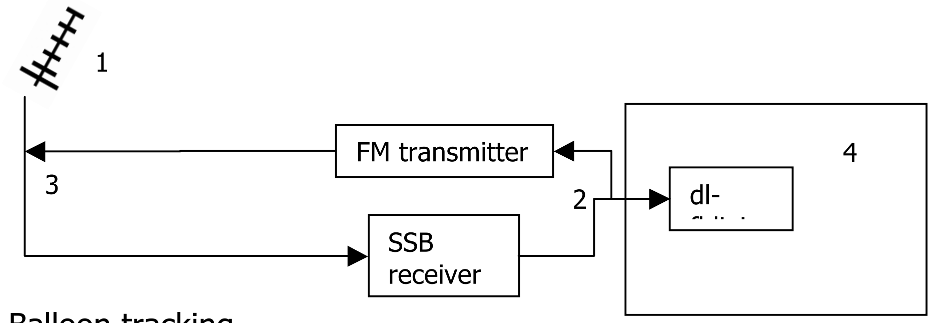
Light Sensor

A RGB light sensor will be used to measure light levels, and the levels of each of the primary colours. The sensor outputs a frequency proportional to light levels, which is scaled according to the inputs.

Balloon Tracking Station



Basic diagram of the hardware connected to the computer

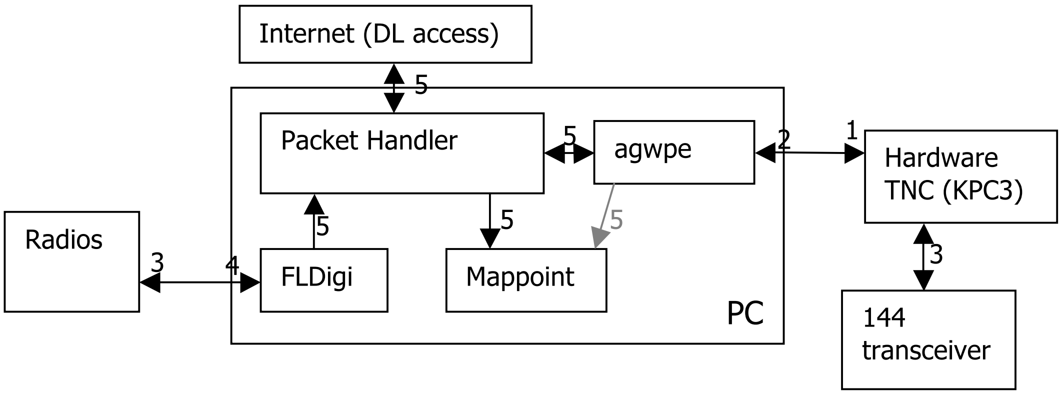


Balloon tracking radio arrangement

1. 434 antenna
2. Audio links
3. RF link (only goes to one device at a time)
4. Computer

The SSB radio receives signals from the balloon, and outputs two tones for 1 and 0, (RTTY), which get fed into the line in of the PC’s soundcard, where FLDigi processes the signal to get back the original data. (Note – if the PC only has MIC in, an attenuator, consisting of a potential divider to lower the voltage is needed).

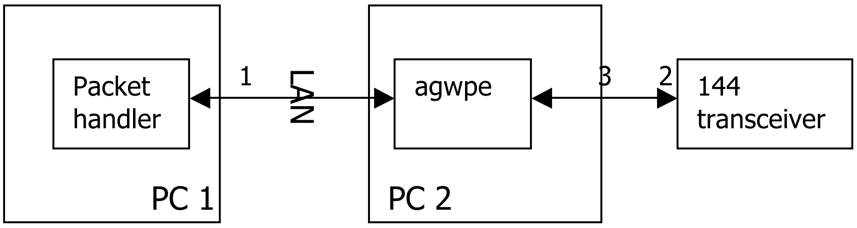
The FM transmitter is only needed to send data to the balloon. (The SSB radio maybe able to both tasks). ‘dl-fldigi’ will generate the OOK signal and send it out the soundcard which is then fed into the radio for transmission. As the FM transmitter is ground based, a higher power output can be used (up to 10W for the foundation amateur radio licence).



1. Serial link
2. PC's serial port
3. Audio link
4. PC's soundcard
5. TCP interface

This diagram shows how devices connect to the computer, and the internal structure of programs. The 'Packet handler' was known as 'APRS hander' on Apex I, and both programs will be similar, but with 'Packet handler' upgraded to accept data from different sources. Mappoint is the program that shows the balloon trace on a map, and so making it easy to see the track of the balloon. AGWPE handles the interface between various TNCs (which are modems for the APRS data format) and the other programs. Mappoint may connect directly to 'Packet handler,' or via AGWPE, as shown on the diagram.

It may be remembered from Apex I that AGWPE can directly generate tones for APRS communication through the serial port, however, inspection of the diagram shows the soundcard could be used by the SSB receiver input. However, as AGWPE communicates through a TCP interface, another PC could be used for AGWPE as shown below



1. TCP connection
2. Audio link
3. PC's Soundcard

This arrangement would work in the same way as above, with the only difference in 'Packet handler' would be the setting for the host IP of AGWPE.

Alternatively an external USB soundcard could be used, so that the PC could have two soundcards, however, whether this is possible depends on how the software (dl-fldigi and AGWPE) are configured

Antennas

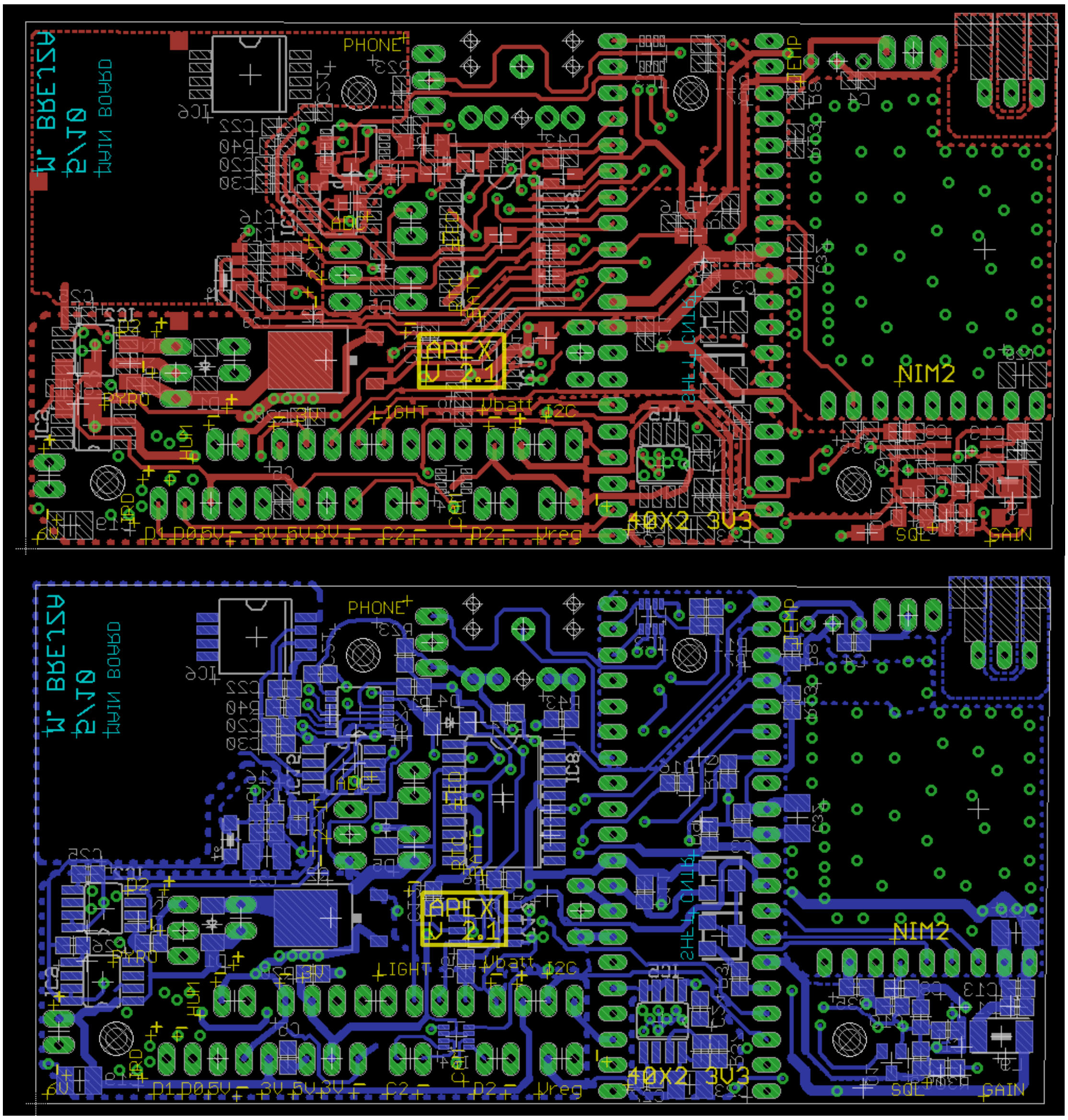
The base station will use high gain Yagi antennas for fox hunting the balloon after landing and keeping it in radio range when it is further away from us. For the rest of the flight, ¼ wave whip antennas will be used on the chase cars as they can be stuck on the roof and the car can be driven.

The Yagis in use are Diamond A430S10R 70cm 10el, of which we have two. The ¼ wave whips (hopefully with magmounts) will be borrowed.

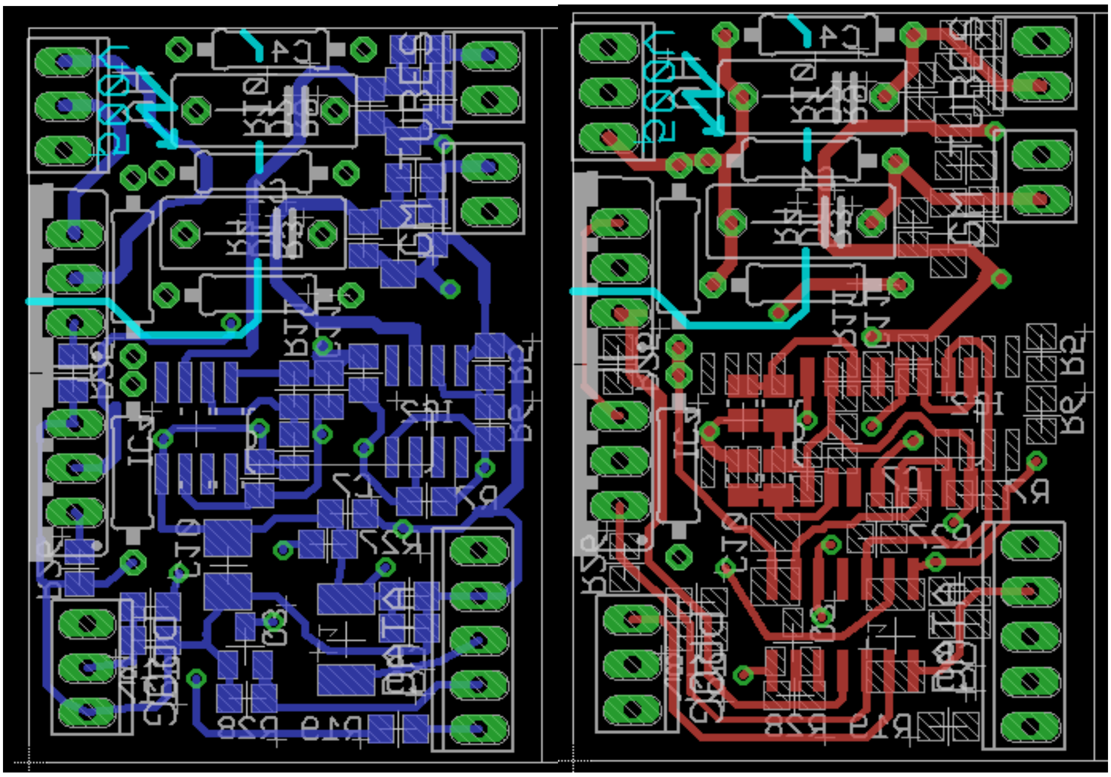
The minibus/other stationary tracking vehicles will drive to somewhere high up between the launch site and predicted landing spot, and get the Yagi up on the roof using a camera tripod as a mount. They will remain there and track whilst the chase cars get to the landing spot.

PCB - Main board (see image gallery for images of the board)

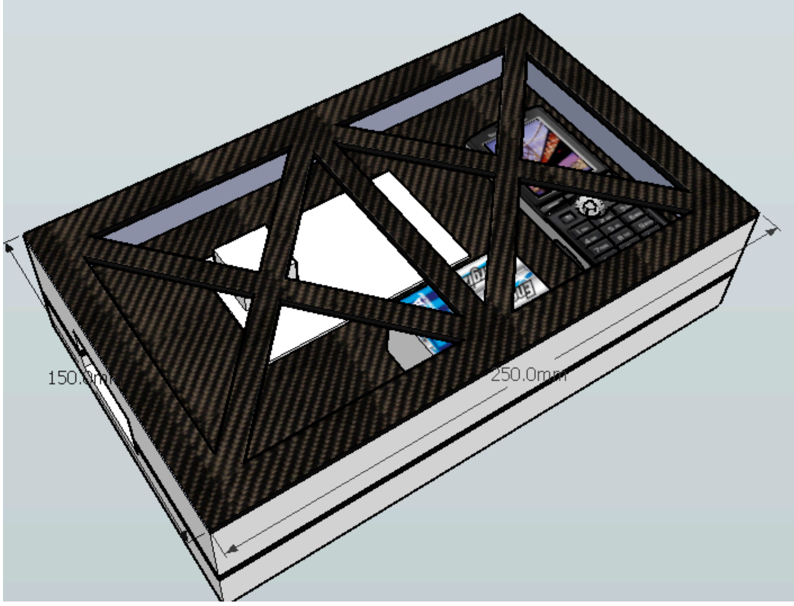
Top:

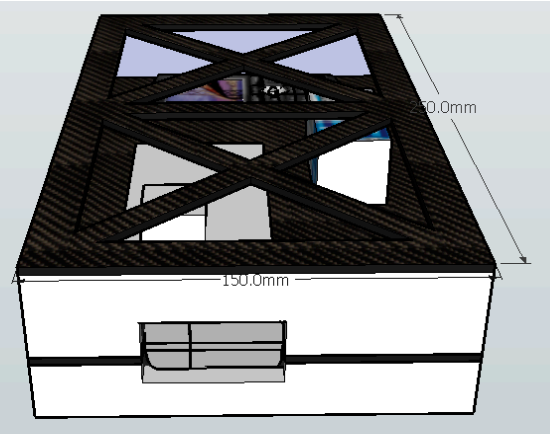


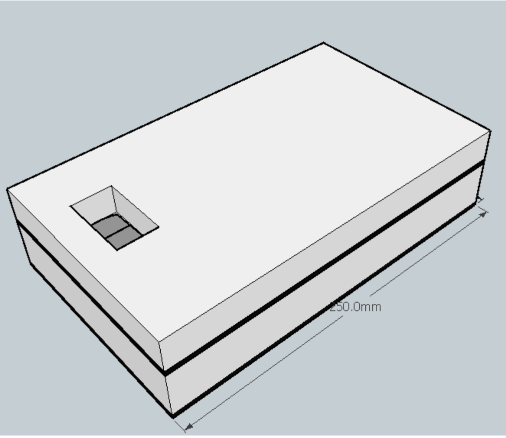
PCB – IRD



Case







The structure is made from carbon fibre, which is CNC milled to shape (ask Ben about the carbon fibre bit).

The CAD (Sketchup - .skp) files are in the code repo and are freely available for viewing.