High Altitude Weak Signal adventures

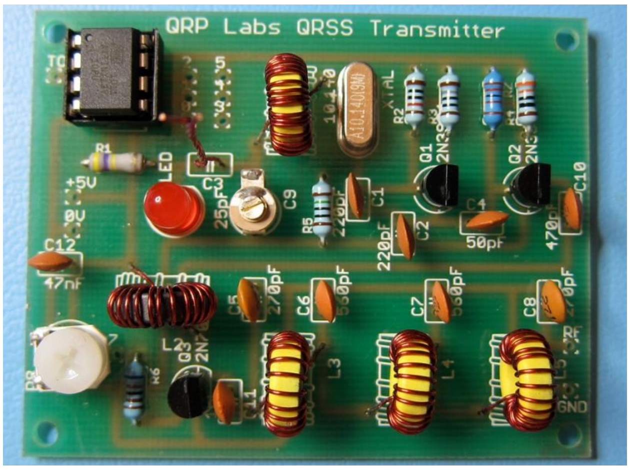
Development of telemetry over WSPR system

Hans Summers, G0UPL (QRP Labs)

This article is about the invention of a system for sending telemetry over WSPR for use on tracking long distance balloon flights for a QRP Labs tracker development; variations of the concept have since been used by other balloonists in their own developments.

Some history

I first started playing with QRSS in late 2003. Back then, there was no WSPR. There was only a very small gang known as QRSS Knights, gathered together on a mailing list, experimenting with various weak-signal mode variants of QRSS, plain slow on/off CW with typically ~5 second dit time.

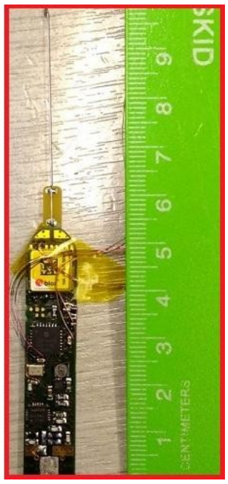


I had a lot of fun designing and operating simple transmitting beacons, and more complex message encoding beacons; even a simple receiver. You can read about some of these on my website: http://hanssummers.com. In 2010 that led to the first QRP Labs kit (pictured right), a simple low cost FSKCW callsign beacon with about 100mW power output. It was the start of a thriving kit business, which since 2016 has become my full-time job, and filling my mind every waking (and sleeping) moment.

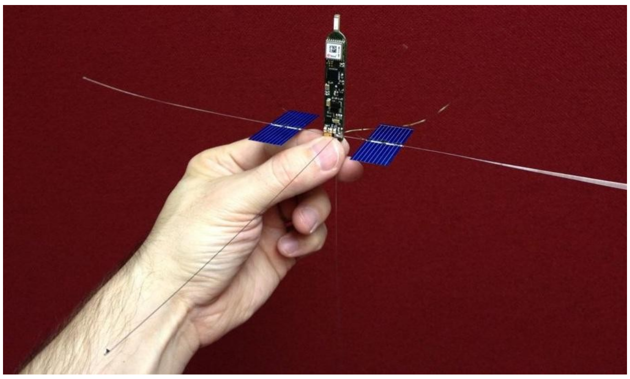
Roll along 2009 or 2010 or so, I had an idea to build an ocean-going buoy which would have some solar panels and a QRSS transmitter. It would have a bunch of sensors on board, and a GPS, and report back its location and the sensor values. Every day I used to walk to work along the river Thames in London and I'd be looking at the water and imagine my drain-pipe type of buoy bobbing along in the water (though really, practically speaking, you'd have to drop it off a boat miles off the coast, to avoid it getting stuck somewhere). I called the project "Voyager", after the two NASA space probes launched in the 70's which still send back signals from the far outer reaches of our solar system. I had always been fascinated by spacecraft exploring the solar system. To me, a buoy floating around in the North Sea would be kind of like a poor-man's personal explorer. Some of the challenges are even similar, but on a much smaller, more hobbyist-solvable scale: sensors, long distance communications, low power. I actually built a transmitter and a 40m receiver to operate the project, but never got far with it. You can read more here http://hanssummers.com/voyager.

The next thing I found really interesting was the High Altitude Balloon (HAB) launches, which often seemed to be an engineering project in US universities (right). These use weather balloons filled with helium and ascend to up to around 30-35,000m altitude, where they burst, deploy a parachute and come back to Earth. At that altitude, the sky looks black and the curvature of space is easily visible. So it's popular to take along video cameras, as well as GPS and radio equipment for communicating the altitude and position, and batteries to run all that. Since these large balloons 2-5m in diameter can carry quite a significant payload, around 500 grams, they don't have to try to miniaturize anything. The flights last a few hours, during which time they typically have line of sight radio communication with the balloon, so UHF telemetry is feasible e.g. on 70m bands. They often use integrated UHF telemetry modules for the communication. The chase is as much fun as the launch – teams enjoy driving sometimes a couple of hundred miles to hunt the balloon, to retrieve their equipment and precious photography and videos (see right).

Exciting though that all is – what REALLY made me sit up to attention, was when I came across the so-called floater or pico balloons in 2014. They're a fraction of the size, a fraction of the weight, and a fraction of the cost (see right). Leo M0XER made the news in July 2014 with his first circumnavigation. It was the 64th flight in Leo's balloon obsession that had started just over a year earlier. Leo had two more circumnavigation flights, B-63 and B66. He then appears to have retired from ballooning. Leo's flights are all detailed here: http://leobodnar.com/balloons/



A weather balloon is made from latex (rubber), which stretches and stretches as it rises through the atmosphere and the air pressure decreases. Eventually it can stretch no more and BOOM it bursts, and down it comes. Some latex balloon flights have succeeded in judging the amount of gas required at launch very precisely, and achieving an equilibrium altitude where the pressures balance but the balloon envelope doesn't burst. Then they can travel quite long distances. The sun's strong ultraviolet radiation attacks the Latex though and as far as I know, they can't last more than a few days.



Floater balloons are different. They're made from a mylar film (plastic) material. You've all seen the shiny foil balloons at parties, with helium inside. This is what is used for floater balloon flights. The mylar doesn't stretch (much). The balloon is launched only partially gas-filled; as the altitude increases the gas expands, and eventually fills the entire available volume of the balloon. If you didn't over-fill it at the launch site, then an equilibrium altitude is reached, where the balloon can just float along in the wind. The mylar film is inelastic and can withstand a certain pressure without bursting. It's a lot easier to target an equilibrium state than with a latex balloon, and the mylar film is much more UV-resistant. The flights can last days or even months.

The main risk to floater balloons is the weather. High altitude clouds deposit ice on the balloon envelope. It doesn't take many grams of ice to bring it down. We've seen many cases where as the balloon came down, weighed down by ice, the temperature warmed up, melting the ice; and in some cases even evaporating it before the balloon reached the ground, then we could observe a miraculous recovery and the flight continued. But really, if you want a long distance flight, with minimum or no risk of high altitude weather, you have to get above the clouds. 11,000m should be considered a minimum, preferably even more.

The difficulty is – to get to 11,000m you need a very light payload. Typical floater balloon payloads are around 10 grams. 10 grams isn't much, to carry a radio transmitter, power supply (solar panels), antenna, GPS receiver, and some computation capability to tie it all together. Yet, this thing is going to fly around the world. Over wide oceans, desolate landmasses devoid of many radio amateurs. We will want to know where it is, and ideally also something about its altitude, and perhaps other interesting parameters such as temperature and battery voltage.

Which brings us to the crux of the matter, and the reason we're all here: RADIO COMMUNICATION. The weather balloon crowd don't have so much of a problem because their balloons travel a couple of hundred miles at most, and the flights are all over in a few hours. Several watts of VHF or UHF will be easy to power with batteries, and they can carry ½ a kg or more; although it isn't trivial, neither is it particularly challenging. APRS is a popular tracking solution.

But for floater balloons the situation is entirely different. Not only do we need radio communications capable of spanning thousands of kilometers, but we can only carry a few grams. There's no possibility of carrying enough batteries for a flight lasting weeks, so solar cells are in order. And we certainly can't carry enough solar panels to generate several watts of RF. How can this be resolved?

Leo M0XER had used VHF modes mainly. The problem there is that as soon as your balloon flies over an ocean, or a sparsely inhabited part of the planet without lots of radio amateurs monitoring, everything goes quiet. You have no idea where the balloon is, until (if you're lucky), it files within a few hundred km of a radio amateur who happens to be monitoring whatever mode and frequency you're transmitting on. I came across various US flights using APRS http://www.aprs.org/ and they have exactly the same problem. With APRS, matters are complicated by the fact that different parts of the world use different APRS frequencies, so your balloon transmitter needs to be programmed to automatically adjust its transmission frequency depending on its location.

The solution to the communications problem must be weak signal modes on HF. Weak signal modes because we just don't have enough power capability, or enough payload carrying capability, for anything more than a few milliwatts of RF. And HF for the beyond line-of-sight range. I had recognized this back when I'd been thinking about the "Voyager" ocean buoy project, and had planned to use a telemetry system involving slanted dashes, whose presence would indicate a 1 and absence, a 0. The problem with that would have been painstakingly slow and tedious manual decoding.

```
// /  / /  /// / /
```

which means:

```
110100101001110101
```

The next great innovation as far as I was concerned, was by Andy VK3YT who launched a series of flights from Melbourne, Australia, documented on his website http://picospace.net/. Andy's "balloon phase" appears to have lasted from February 2014 to his 73rd flight which was in December 2019. Many of Andy's flights used JT9 mode and WSPR on 30m and 20m HF bands.

JT9 is a weak signal communications mode with powerful error correction. 13 characters of text are encoded into a series of 9-tones. These are transmitted typically in a 1-minute transmission (though the protocol also supports slower transmission rates for higher SNR).

Andy VK3YT developed a special telemetry protocol in which he encoded the various parameters such as location, altitude, battery voltage etc into a 13-character transmission for JT9. One important detail to mention, is that Andy didn't just take each character of the JT9 transmission and make it mean something, for example for altitude, A means 500m, B means 1000m, C means 1500m etc. Instead, he decided the resolution he wanted on each telemetry parameter, and encoded that into bits that were then reversed out into the 13 character transmission for JT9. This allowed him a lot more flexibility on data resolution.

Andy VK3YT actually needed TWO JT9 transmissions to transfer all the data he wanted. For decoding and collecting the data from tracking stations, he modified the WSJT-X software package, which is Open Source, to be able to decode his two JT9 telemetry packet styles, and upload the decoded data to his own server. Use of HF weak signal modes allowed Andy to reliably track his balloons even across the largest ocean of all, the Pacific ocean. He had several circumnavigation flights.

I followed all these flights avidly around 2014/2015. Around that time, I was also developing firmware improvements for the Ultimate3S http://qrp-labs.com/ultimate3/u3s which is a QRSS and WSPR and other weak-signal modes transmitter you probably know of, and is standalone with no PC required. I had had some requests to include JT9 in the list of supported modes and was working on that.

In late 2013 I had been discussing an ocean floater with tracker with a group of South African radio amateurs.

Then around Feb 2014 I had an email from Dave VE3KCL, which really was what kicked my balloon dreams into a whole new gear. Dave had taken an interest in high altitude “floater” balloons and thought it would be fun to launch an Ultimate3S on a balloon; he was excited to hear that I planned to implement Andy VK3YT’s JT9 balloon telemetry messages.

It was the start of a long friendship between Dave and I that has generated thousands of emails of correspondence, long IM chats, phone calls, meeting in person at Dayton hamvention when Dave came to help me man the QRP Labs booth. Dave has tremendous energy and enthusiasm which is incredibly infectious. In April 2015 I officially included JT9 in the Ultimate3S firmware, and not long after, shared with Dave a series of beta versions that included things like Andy’s JT9 telemetry for balloon tracking.

VE3KCL’s first flight

So it was that in June 2015, Dave VE3KCL launched his first balloon, carrying an Ultimate3S, transmitting WSPR and JT9 telemetry, just as Andy VK3YT did on his balloon flights. It reached an altitude of 4,233m and flew for 2.5 hours, covering 68km. The second balloon flight the following month, had a very similar flight profile. These two flights are detailed at http://qrp-labs.com/flights/s1 and http://qrp-labs.com/flights/s2 . I recall a funny discussion from around this time, where Dave and I were discussing what to call the flights. After all, Leo M0XER used the prefix “B” as in B-1, B-2, B-3 etc., presumably the B was for either Balloon or Bodnar (Leo’s surname), who knows. And Andy VK3YT used PS e.g. PS-1, PS-2 etc., clearly the PS for picospace after his website http://picospace.net. Dave suggested “S” for short, since 2½ hours was evidently as long as they could fly, and so S it was, starting with S-1, then S-2 etc.

By the third flight, Dave had uncovered the cause of the early demise – some heavy gas sitting in the bottom of his hydrogen tank meant that he was over-filling the balloons to get the calculated lift. Once properly emptied out and starting again with fresh, pure hydrogen, the flight durations increased dramatically.

Inventing a protocol for telemetry over WSPR

Meanwhile I continued observing the flights of Andy VK3YT and observed two sort-of problems with the tracking.

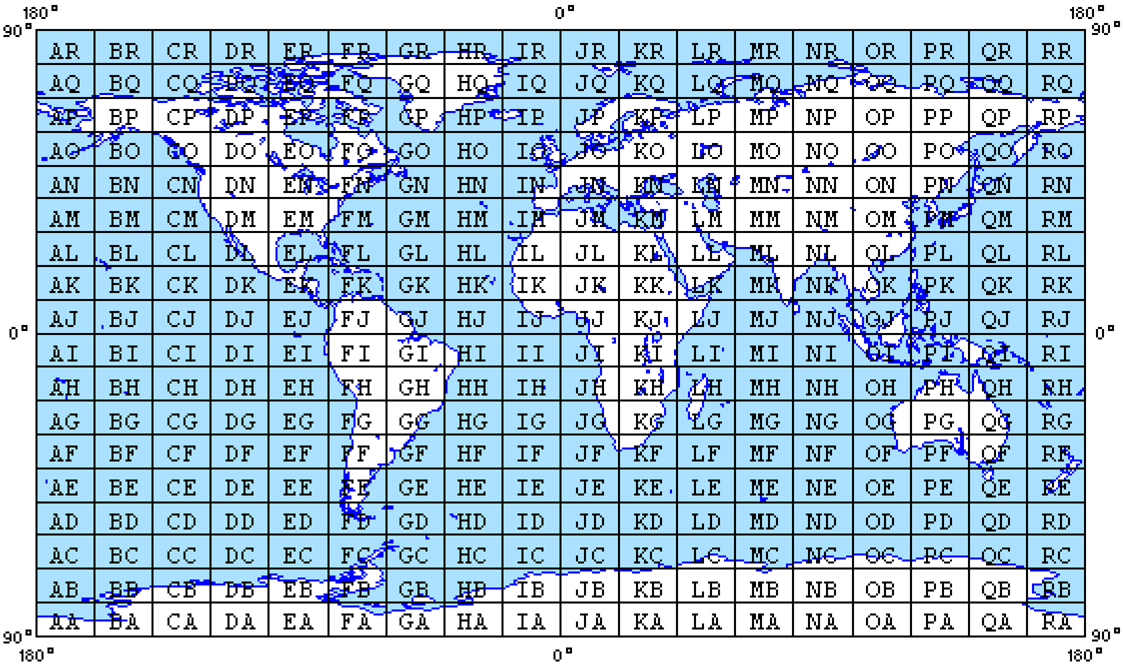
Firstly, by using JT9 for tracking, and a customized version of WSJT-X to decode it and upload the reports to his server, Andy had to recruit his own network of volunteer tracking stations to install the special WSJT-X version and listen for his balloon. This necessarily limited the number of stations globally that could receive the tiny signals from his flights, so there were at times, gaps in the tracking capability.

Secondly, Andy VK3YT was also using WSPR on his balloons. I observed that there were many occasions when no JT9 was decoded, but WSPR was – even by the same receiving stations – which eliminates any effect due to the much larger WSPR receiving network.

This is as you would theoretically expect too, since the bandwidth of JT9, data payload (13 characters) and the bit-rate, are all higher – leading to lower Signal to Noise Ratio (SNR). In other words, more power would be necessary to achieve the same reception reports as WSPR. This is completely normal.

So it appeared, theoretically and practically, that WSPR would be a better solution for tracking balloons. However, a significant problem exists with using WSPR for tracking, too. This is due the fact that unlike the free-format 13-character text message carried by JT9 (and some other JT-modes), WSPR has a fixed-format:

- 6-character station callsign
- 4-character Maidenhead grid-square
- Power in dBm (19 different levels available)



The power level is one of 19 values from 0 to 60dBm, 0, 3, 7, 10, 13, 17, 20dBm etc. The problem is that a Maidenhead grid square measures 1° latitude by 2° longitude. According to ARRL at http://www.arrl.org/grid-squares, that is “approximately 70 × 100 miles in the continental US” (110 x 160km). Not only is the location precision very coarse, but we also lack that most crucial of parameters watched by any balloonist: the altitude!

I believe Andy VK3YT had started mapping altitude levels to the 19 power levels of the WSPR transmission, so that as a backup, if JT9 was unavailable – and if he had WSPR reports – then he could at least determine approximately where his balloon was and roughly at what altitude. But really, 19 levels is again a very coarse measure, and ideally we’d like more data from the balloon including battery level, temperature and speed. So WSPR in its intended state, isn’t a great tracking tool.

The enormous advantage of WSPR is the worldwide tracking network! WSPR has been very popular, and is ever-growing in popularity, since it was first created in 2008. If it were possible to use WSPR for the tracking, then it would solve the first fundamental problem of Andy VK3YT’s HF weak signal tracking, that of recruiting volunteers to install his special software for JT9 decoding. If the limitations of the protocol could be overcome, we would end up with the PERFECT tracking system for long distance “floater” balloons:

- Long range coverage by using HF
- Weak signal mode works fine with a few milliwatts
- Powerful error correction telemetry
- Worldwide existing network of monitoring stations
- Automatic tracking

Dave VE3KCL and I discussed all this frantically for several weeks and months, with various ideas on how to use the locator field of WSPR transmissions as well as the power level, to carry some of the data we wanted. We called it “WSPR abuse”!

A big breakthrough came when I noticed that NO official amateur radio callsigns ever exist with prefixes 0, 1 or Q, and had the idea that we could make use of this. The International Telecommunications Union (ITU) simply doesn’t allocate any callsigns with these prefixes, according to their published policy. There have been some unofficial operations with callsign prefix, generally operations in states that have no official prefix of their own, or a number of troubled states (that I shall not provide examples of, for fear of somehow upsetting somebody somewhere) around the world whose existence, independence or sovereignty is disputed.

I started to design a telemetry packet, re-encoding our additional data into a second WSPR transmission, which was identified as telemetry, not a real amateur radio station, by using a callsign with 0 or Q as the first of the six characters. I also used the third callsign character, which is by the definition of the WSPR protocol, a numeric digit in the range 0-9. Each balloon telemetry would be identified by its callsign being of the format CxNxxx where:

- C is character ‘0’ (zero) or ‘Q’
- N is a number in the range 0-9
- x are characters repurposed for our telemetry

Therefore 20 different “channels” were created, allowing 20 different balloon flights to use the telemetry system, without interference with each other. Later, inclusion of the ‘1’ prefix increased the number of available channels to 30.

The four characters of the Maidenhead grid square, and the 19 levels of power, were also re-purposed to carry telemetry. A complete WSPR transmission consisting of station callsign, locator and power was therefore represented:

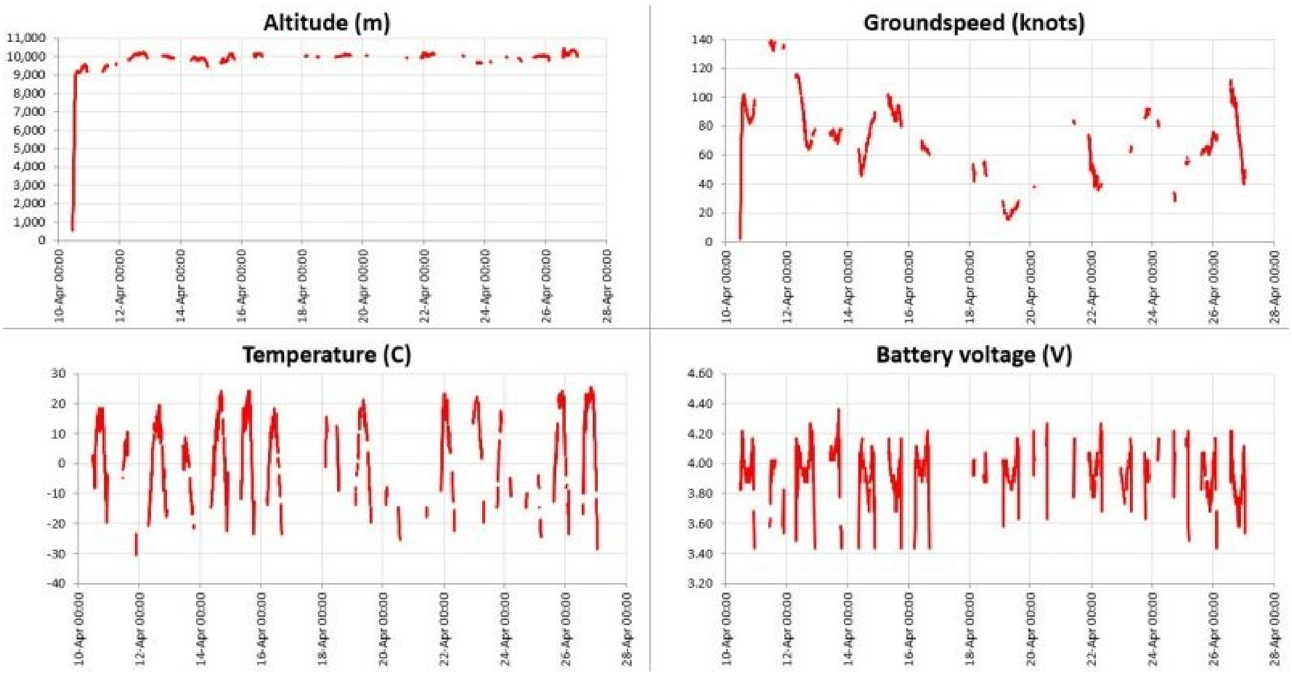
CxNxxx xxxx xx   (callsign, grid-square, power)

where x are the parts of the standard message whose bits are re-purposed to carry telemetry. A WSPR message contains only 50 bits to start with. These are encoded with forward error correction into 162 symbols, each of which is one frequency transmission (tone), spaced 1.46Hz apart. The symbols are transmitted in just under 2 minutes.

After deducting the first callsign character (always 0, 1 or Q), the 3rd (always a number), and allowing for the fact that there are wasted un-used bits in the power field, which cannot be reclaimed – in the end we have only 39 binary bits of data to carry our desired telemetry. So careful thought is required as to the required resolution of data. What I designed for the WSPR telemetry protocol was:

- Maidenhead grid subsquare 5’th and 6’th characters, which are letters in the range ‘A’ to ‘X’. In addition to the 4-character gridsquare sent in the main WSPR transmission, this gives us a location resolution of a few kilometers, which is adequate for tracking.
- Altitude: range 0 to 21,340m with a resolution of 20m. This altitude resolution of 20m is plenty fine enough!
- Temperature: -50C to +39C with a resolution of 1C (90 possible values)
- Battery voltage: 3.00 to 4.95V with a resolution of 0.05V (90 possible values)
- Ground speed in knots: 0 to 82 knots with a resolution of 2 knots
- GPS satellite status bit: 1 if we have a healthy good GPS lock
- GPS number of satellites: 1 if there are at least 8 tracked satellites

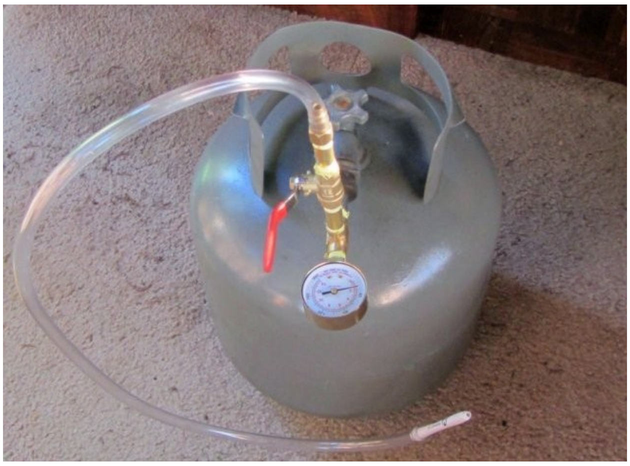
Altitude, temperature, battery voltage and ground speed “wrap” - by this I mean that for example, if the ground speed reported by the GPS is 86 knots, the telemetry will report it as 2 knots. 88 knots would be reported as 4 knots, and so on. This is important because when the balloons get in the powerful jetstream winds they can certainly exceed 84 knots and do so frequently. It is normally easy to know whether 84 needs to be added to the reported number, because of the difference in position which is easily calculated in kilometers, allowing an approximate speed to be determined, at least well enough to know whether it is above or below 84 knots.



This system is described in detail here http://qrp-labs.com/flights/s4#protocol with an example encoding/decoding spreadsheet. It has proven very reliable and robust. During many test flights with power level of about 10-20mW, communication has been possible across all the oceans and land masses of the world. Probably the longest distance achieved was long-path reception in New Zealand, when one of the test balloon flights was in Europe, a distance of 22,000km!

Dave VE3KCL S-series flight payloads

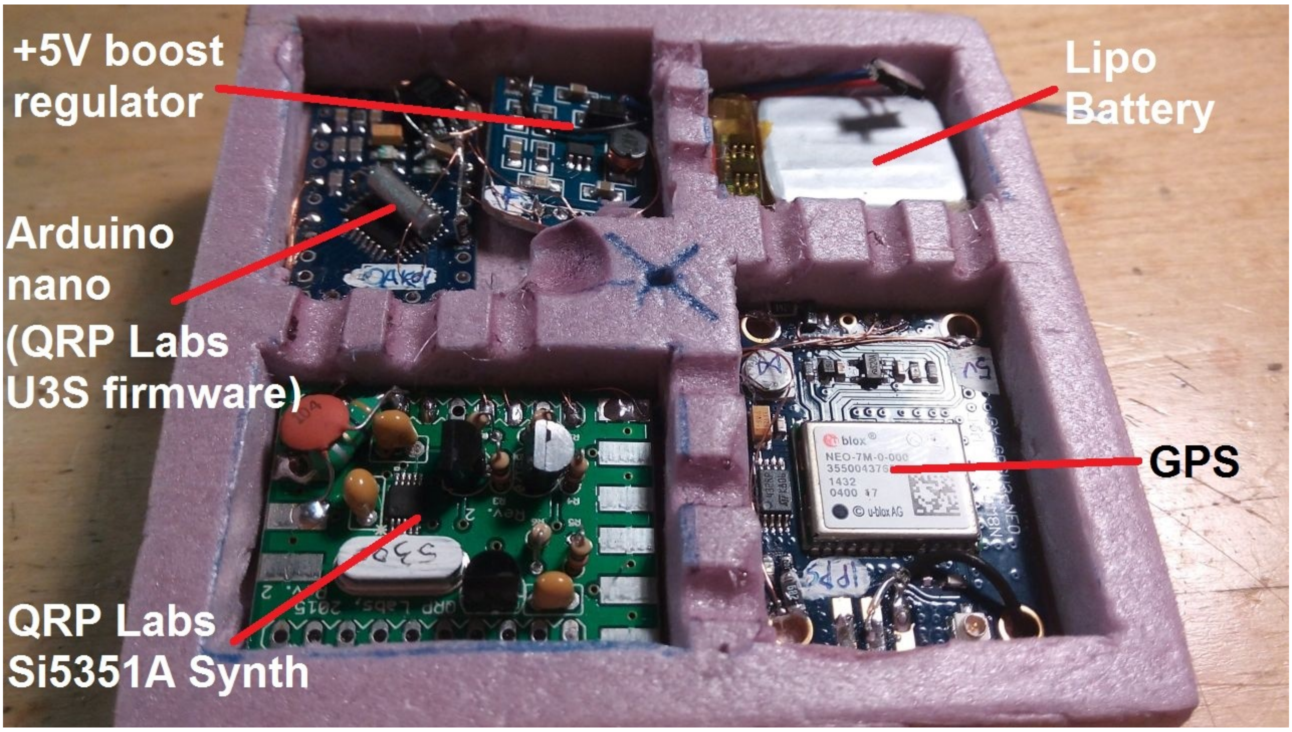
Some details are due, regarding Dave VE3KCL's balloon hardware. The early flights used inexpensive mylar "party balloons" printed with the message "Over The Hill"… which is of course, about as far as they managed to fly. Hydrogen is a bit better at lifting than helium, has slower leakage rate due to the larger hydrogen molecule, and is also more environmentally friendly (since helium is a non-renewable resource). However hydrogen is also an explosive gas so some care in handling is required, with all due attention to necessary safety measures.



Dave VE3KCL's early flights consisted of a collection of off-the-shelf hardware modules, packaged in a polystyrene enclosure. The parts of the system are:

- QRP Labs Si5351A Synth kit http://qrp-labs.com/synth
- Arduino Nano
- uBlox GPS board
- Voltage regulator module
- LiPo battery

The Arduino Nano was not used as an Arduino. It was only used because it contains the same ATmega328 processor as the Ultimate3S, but in a much smaller SMD package which saves weight. Dave flashed the 'nano's processor with the Ultimate3S firmware, the special version of the firmware that implements the WSPR telemetry system. Changing the Arduino nano's crystal to 20MHz to match the Ultimate3S was also needed. Eventually, Dave designed his own PCB to hold the necessary components, resulting in further weight savings.



Dave's S-series flights based on the Ultimate3S consisted of 26 flights in total. The 11th, in June 2016, was the first circumnavigation Dave achieved, about a year after his first 2.5-hour, 68km flight. What a scale of improvement in one year!

Tracking Spreadsheet

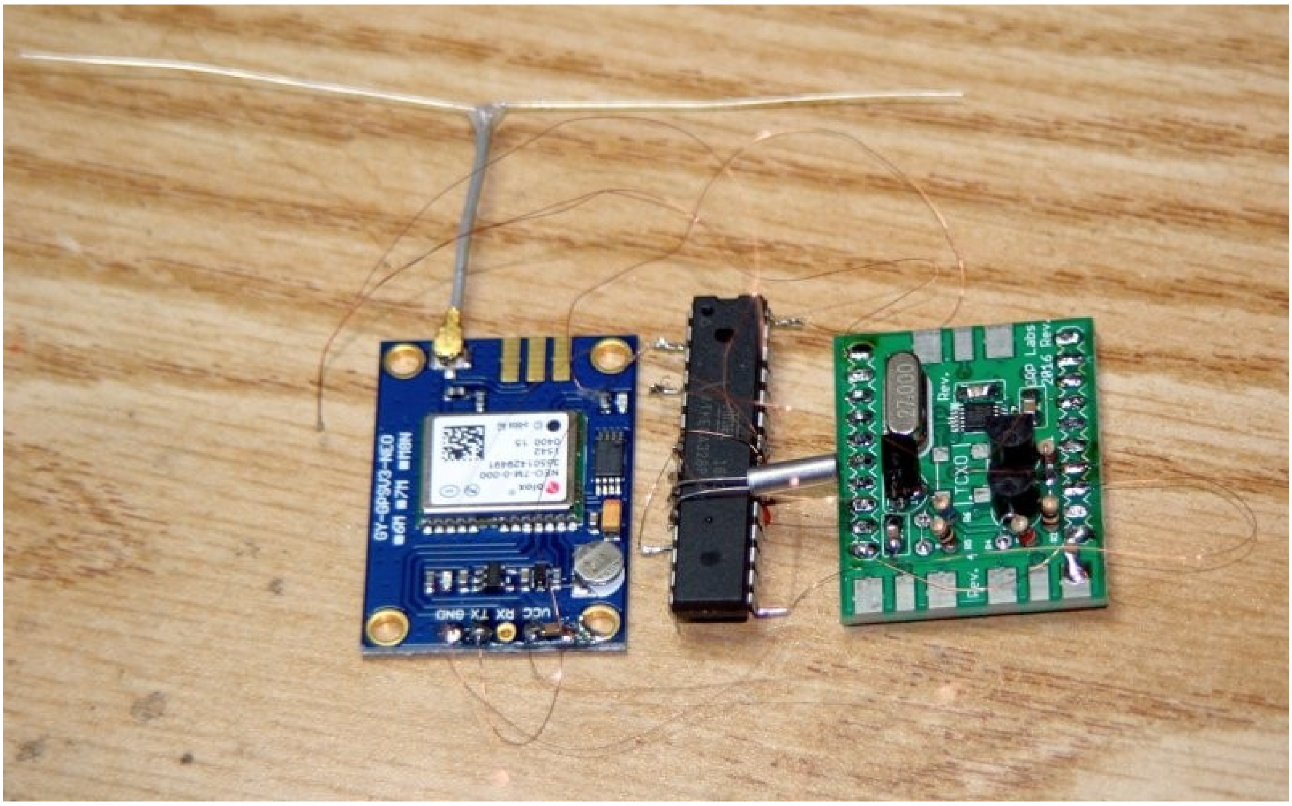
At the other end of the tracking system, was an Excel spreadsheet that I wrote. Every two minutes, the spreadsheet connected to WSPRnet and loaded the entire 5,000 last spots for 20m and 30m bands. The sheet then went through the list and parsed the spots to find any balloon telemetry spots, which were decoded and matched to the normal WSPR transmission reports in the preceding two-minute slots. The two pieces of information (4-character Maidenhead grid from the main WSPR transmission, and the 5th and 6th character grid subsquare information from the telemetry transmission) were pieced together and converted to latitude and longitude, which was written into an HTML file in Google Map format and FTPed to my website. The result was a live tracking map on the website that updated every 5 minutes, with the latest decoded information from the balloon (that was sent every 12 minutes or 10 minutes).

Balloon flights by Jim N2NXZ

Mention is due to other balloon celebrities who have also been involved with flying the special QRP Labs Ultimate3S code with the WSPR telemetry tracking system.

Prominent and indeed foremost among these is Jim N2NXZ who has his own particular style and enthusiasm. I particularly liked Jim's low-tech way of implementing a low weight version of the Ultimate3S, which was simply using an existing set-up Ultimate3S ATmega328 chip, full-size DIP28 package, and soldering the wires of the GPS and Si5351A Synth kit directly to the chip! No PCB at all for the processor.

Jim had been experimenting with balloon flights for a number of years. But then started his own series of flights with prefix U3S since they were all based on the Ultimate3S kit, running the standard Ultimate3S firmware, not the special telemetry version. Somewhat later on, Jim did also run the special version with the WSPR telemetry. We even have a photograph of Jim launching his U3S-7 flight from the roof of his house!



Jim's contribution is notable and commendable for his enthusiasm and non-traditional way of looking at things, which was always interesting to discuss, and we learned many things form these U3S-series flights too. In total Jim N2NXZ launched 33 balloons in his U3S-series, and they were always fun to watch and learn from.

Bob ZL1RS

Bob is notable not just for running several balloon flights, BB-01 to BB-05 using the special WSPR telemetry, the first time it was used in the southern hemisphere – but also for actually realizing in real life, my dream of an ocean buoy! You can read much more about this voyage here https://www.qsl.net/zl1rs/oceanfloater.html . Somewhat like my original "Voyager" plan, Bob used D-cell batteries in a drain pipe; but of course the JT9 and WSPR tracking was much more advanced than I had ever imagined back then. Bob's Ocean Floater project drifted around in the South Pacific Ocean for 407 days, reporting its position and condition continuously during that time.

Bob ZL1RS' 5th balloon launch BB-05, achieved circumnavigation with two laps around the world, finally coming down in the South Atlantic ocean half way around its third lap.

Daniel DL6OW

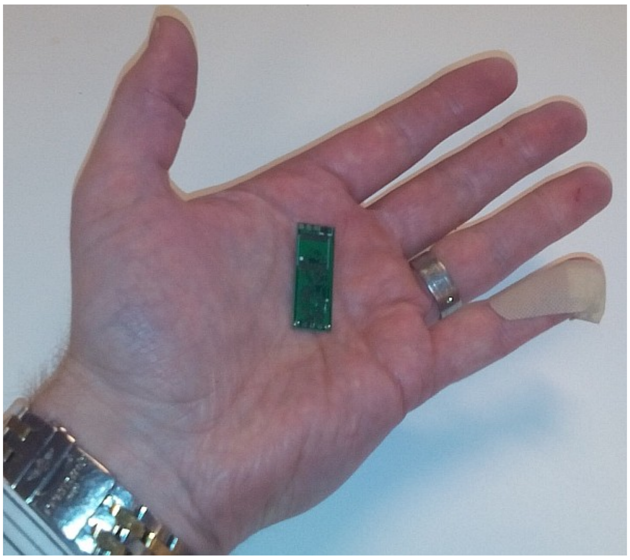
Then there is Daniel DL6OW and his friends flying the flag for Europe, with WSPR telemetry balloons called Stella-1, Stella-2 and so on up to Stella-22. Stella is the name of Daniel's daughter! Daniel's first flights, like Dave VE3KCL's, were on multiple connected off-the-shelf modules but subsequently he designed his own PCB to reduce the size and weight.

Others

Eventually the special Ultimate3S firmware found its way outside the small group noted above and several others have also flown it, or variations of it. Apologies for not listing in full all the names and details here, which would require a lot of research for which time is lacking.

QRP Labs U3B tracker development

Right from the beginning in my conversations with Dave VE3KCL, I had been considering how nice it would be to produce and market a small tracker board. There are people who can design their own PCBs and assemble the SMD components, write the firmware and so on. But for every person like that, there must be 10 or 20 times more people who don't have the resources (experience, time) to do that. An assembled board would make it possible for many more people to participate in this interesting and crazy hobby of high altitude long distance ballooning!



I called my tracker "U3B", since it was based on the Ultimate3S kit http://qrp-labs.com/ultimate3/u3s with the "B" suffix standing for Balloon. The size was 38.0 x 12.7mm, including transmitter, voltage regulator, GPS and ATmega328 processor. Components on both sides included 0402 size resistors and capacitors.

For fun and the extreme deliciously and un-neccessarily over-the-top-ness of it, the U3B did not use the standard or even a modified version of the Ultimate3S firmware. Instead I wrote a 32-bit virtual machine, that ran inside the 8-bit AVR processor (ATmega328), with a BASIC interpreter and compiler, storing the program in the 2K EEPROM of the processor. The idea behind this, was that the experimenter would be free of the constraints of the Ultimate3S system.

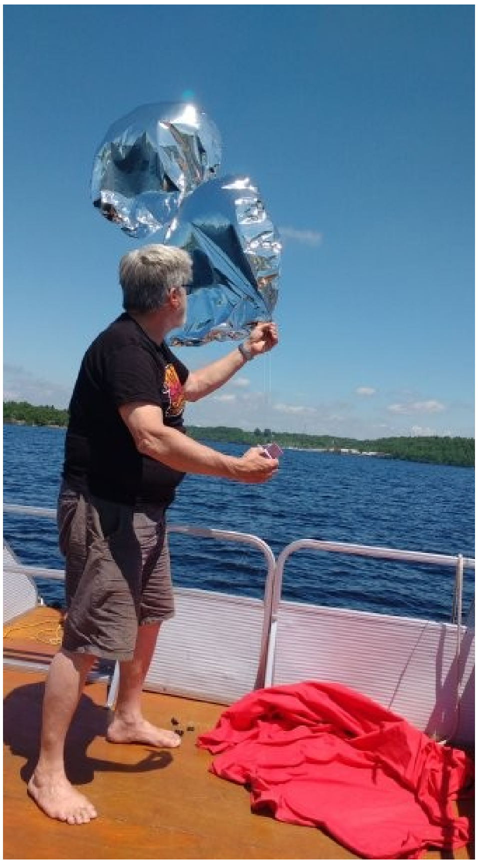
In the Ultimate3S, one configures a number of transmission slots, including specifying the mode and message to be transmitted. But in the U3B, the BASIC programming language statements included commands to transmit a message in CW, FSKCW, WSPR, or the Telemetry etc. Even JT9.

But the U3B system could be configured with a very very simple BASIC program for a really basic (pun not intended) flight. Or a more advanced user could write his program to customize the flight to his requirements. U3B had a number of I/O pins which could be read as analog or digital pins, or other sensors on the I2C bus could be read, and so on.

U3B was programmed via a 9600-baud serial port, using a USB-to-Serial converter and a terminal emulator program on a PC.

Dave VE3KCL assembled and flew the first prototype U3B on 16-Jun-2016 and so started the U3B-series of balloon flights, with U3B-1. The BASIC program listing of that first flight was:

```
  5 LET FR = 26999800
  6 GPS 300
  7 PRINT FS , FR
 10 SLEEP 10 0
 20 PRINT FS , FR
 25 PRINT BT
 30 TELE 2 10140260 10 "G0UPL"
 35 PRINT BT
 38 PRINT FS , FR
 47 TELE 2 14097130 10 "G0UPL"
 50 PRINT BT
 70 PRINT FS , FR
 75 JT9 1 10140400 "G0UPL #M6"
 80 PRINT FS , FR
 90 GPS 300
 95 PRINT FS , FR
100 GOTO 10
```



Variables FS, FR and BT are system variables holding the system crystal frequency (20MHz), Si5351A reference frequency (27MHz) and Battery voltage respectively. You can see that there are commands for transmitting telemetry, JT9, and doing the GPS calibration.

In total there were 28 U3B test flights, each one experimenting with different aspects of the hardware, the firmware, or the software (BASIC flight program), and with varying degrees of success. The U3B prototypes turned out to be rather hard to modify, since everything was so incredibly small and closely packed; so we designed a PCB twice the size, with mostly all the components on the top side, and

Dave VE3KCL and all the balloon technicalities

Throughout nearly 6 years of collaboration on the balloon flights, Dave VE3KCL and I have worked closely together. I designed hardware, PCB layouts and wrote firmware and the tracking spreadsheet. Dave assembled the units to fly. Dave may be 2 decades my senior but he’s not afraid of hand-soldering 0402-size SMD components!

Dave has endless enthusiasm and has gone deep into many aspects of flying balloons, other than just the tracker hardware, firmware and software.

Just the launch of one of these tiny balloons has many perils of its own. Zero or very low wind is necessary, since the lift is so low that any wind will blow the balloon sideways and gusts may throw it into the ground and destroy it. You also need clear skies and not just at the launch, but also forecast for the next two or three hours it will take for the balloon to ascend to its target altitude above all the clouds. You need to not tangle up the terribly fine antenna wire, nor break it, nor do anything that could break the extremely fragile glass solar panel cells. There’s a lot to think about and Dave has mastered it all, now nearly all the flights are launched successfully.

Dave even built his own cooler system to test the behaviour of the tracker in the extreme low temperatures found at 12,000m. There comes a time when the -16 or -20C in your domestic fridge freezer just doesn’t do it. That’s when it’s time to bring in Peltier effect devices. They transfer heat at the junction of two different materials, effectively sucking heat from one side to the other. It’s a rather inefficient cooling method compared to ordinary refrigeration techniques seen in consumer appliances. But by nesting three sets of Peltier-cooled chambers inside each other, with water cooling to extract the large amounts of heat generated by the outer one, Dave was able to reach temperatures below -50C, which provided very valuable test data on the required temperatures for the electronics to work properly.

Dave knows how to perform the almost unbelievable feat of slicing the incredibly fragile glass solar cells in half without breaking them.

Balloon pre-stretching and filling is another important area that Dave has investigated. The mylar film balloons are inelastic but not perfectly so. By stretching them under pressure on the ground, Dave has been able to expand the volume and achieve higher float altitudes.

Dave VE3KCL has launched from a moving boat; dodged dogs having their morning walk on the launch beach, and even successfully and repeatably persuaded his better half to provide photography services for the documentary record!

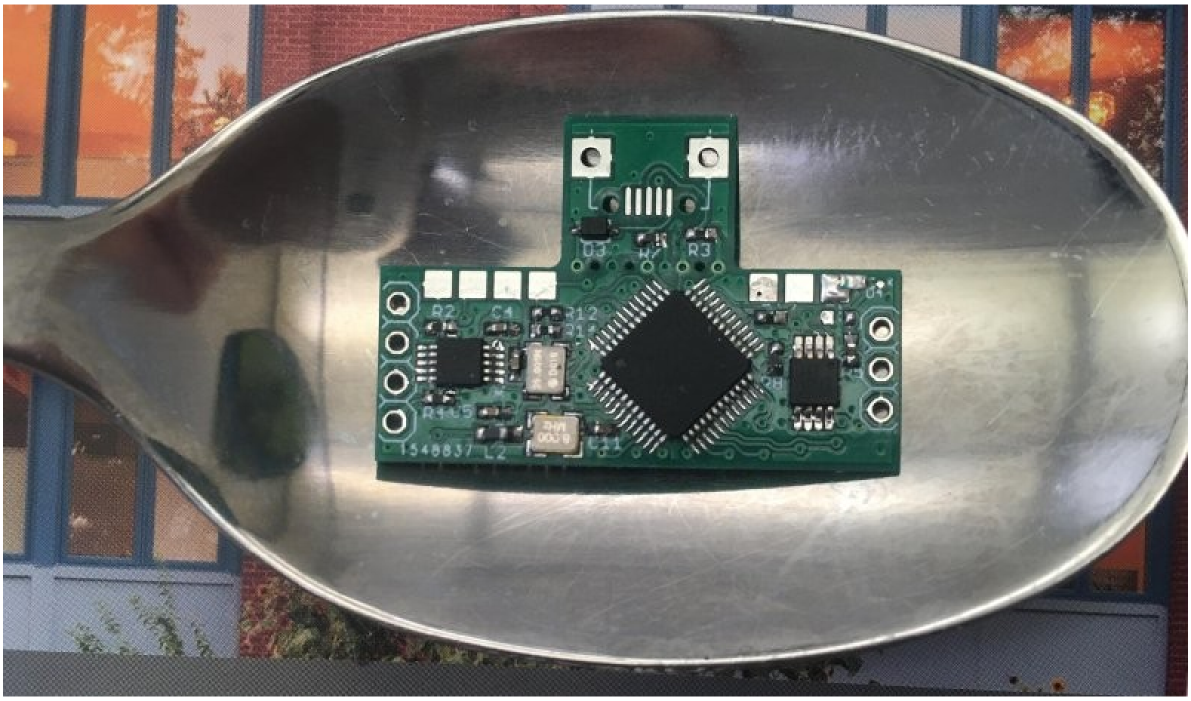
Above all Dave has an enduring, quietly understated enthusiasm that is infectious and inescapable; he has been endlessly patient with me while waiting for me to be distracted in all the other QRP Labs projects I've got lost in over the intervening years, a great help at Dayton hamvention… just an awesome guy, ya can't love Dave enough!

<u>U4B – at last…</u>

Then as my room-mate at the Dayton hamvention in May 2019, we got to discussing how to finally finish up the U3B project, Dave said if I allocate a whole week to it I could finish it (yeah-right!) and somehow during those discussions, it mutated into the U4B. "4" because now I have changed the processor to a 32-bit STM32 ARM processor. These have very much more raw power, peripherals and functions compared to an ATmega328 AVR processor, and yet have a very low price.

The U4B will be the final tracker, the one that that enters production. The first prototypes were on a double-sized board, with components on one side only, just like the U3B. There is a micro-USB connector on a protruding section of the PCB that is designed to snap off before flight, for weight-saving. Like the U3B, the U4B tracker weighs around 1.5 grams, it is a 33.0 x 12.7mm PCB with GPS, processor, Si5351A (synthesizer/transmitter), voltage regulator and 128K EEPROM chip.

The final board is pictured below.



The firmware contains a customized operating system called QDOS (QRP Labs Disk Operating System) with a command prompt, FAT12 file system on the 128K EEPROM, file manager, full-screen editor application, BASIC compiler and integrated debugger. All of this is hosted on the device itself, there is no special software running on a PC downloading code into the U4B. The only thing needed is a terminal emulator, for example on Linux we use PuTTY. The U4B presents itself to the PC as a Virtual COM port serial device.

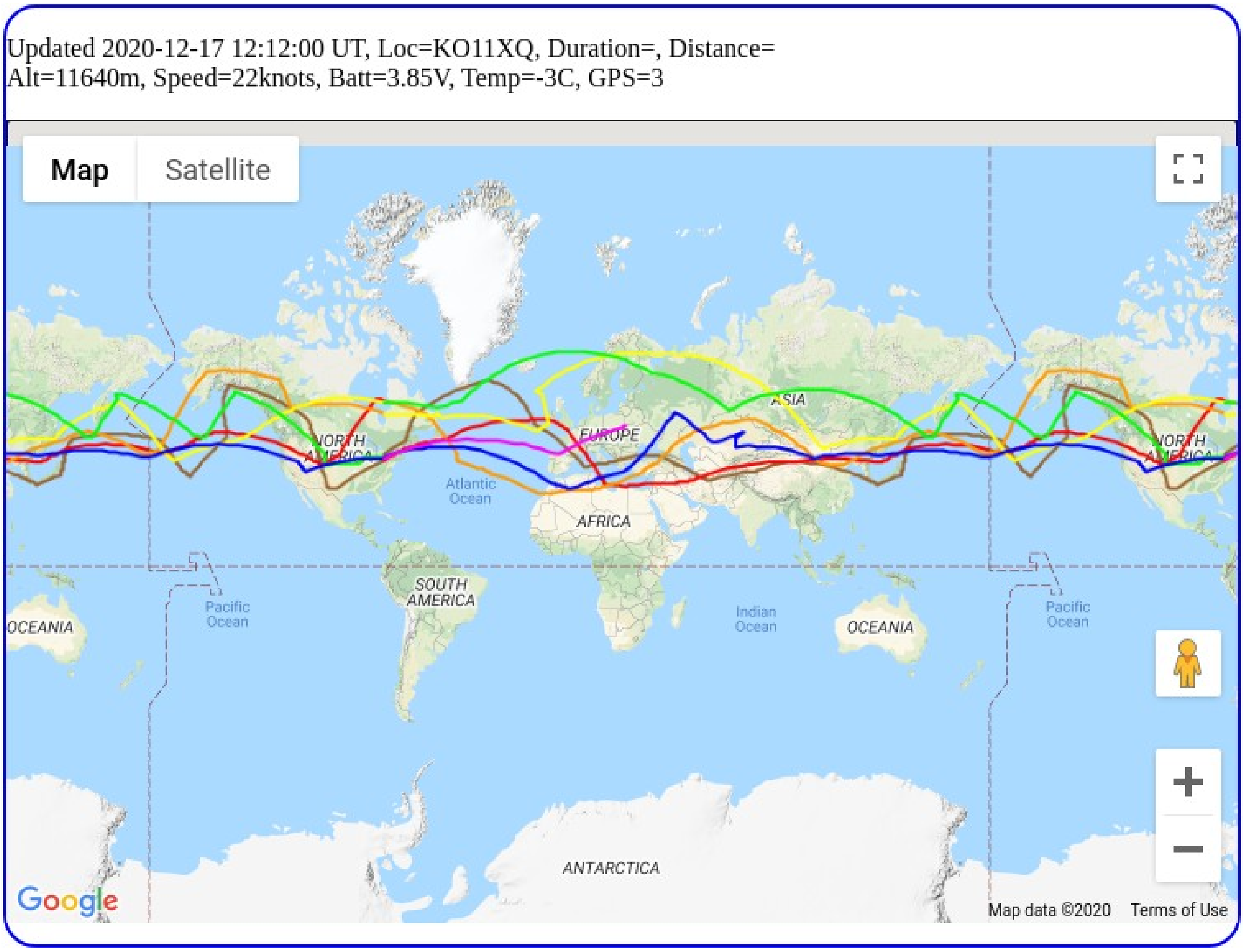
The BASIC programs have similar capabilities to the U3B. Which is to say, almost all the modes the Ultimate3S is capable of, as well as the WSPR telemetry type.

Modifications to the telemetry protocol allow up to 600 telemetry channels, and additional data fields for your own sensor data (8 general purpose analog/digital I/O pins, I2C bus sensors, and more).

At time of writing (17-Dec-2020) there have been 13 test flights of the U4B. The latest one U4B-13 uses an actual real-size final PCB layout that will be used for the production batch. It is currently on its first lap around the world, having traveled from Toronto Canada, to China so far.

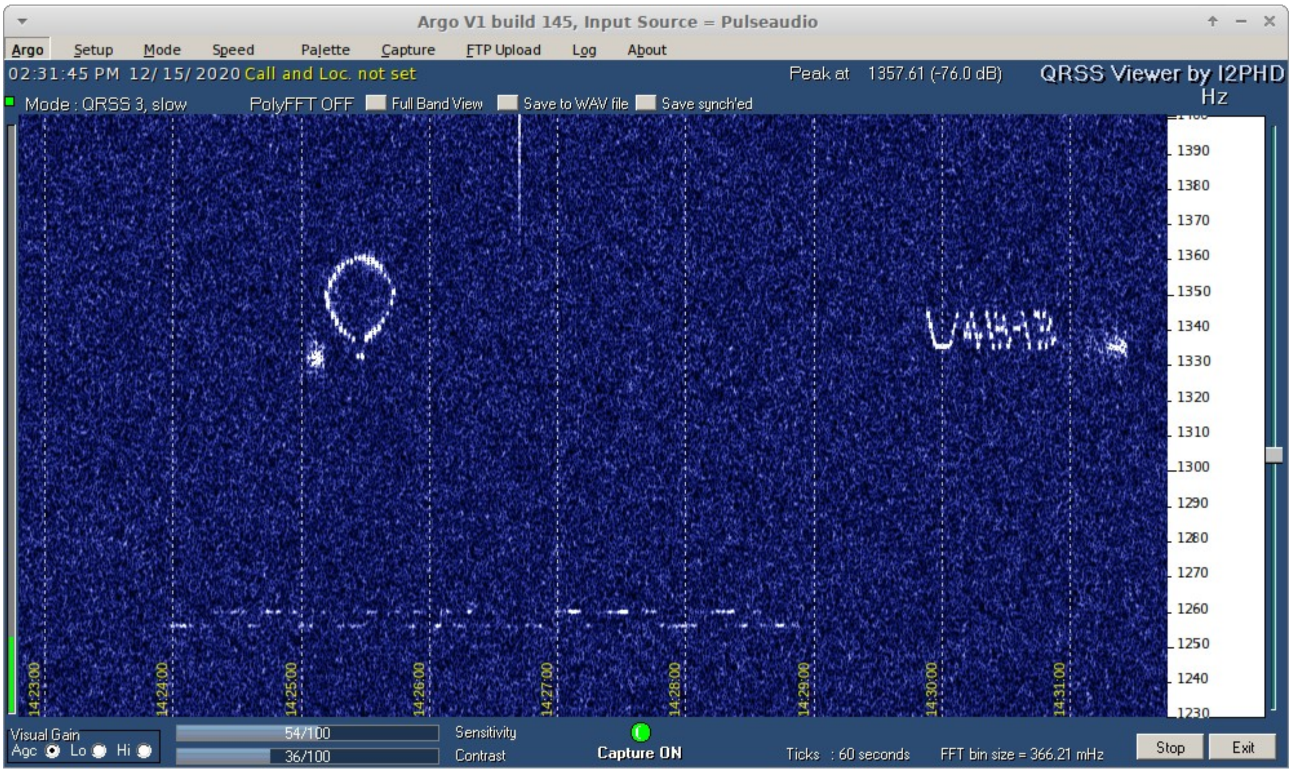
The U4B-12 flight was launched in September and has been up now for almost three months; it is on its 7th lap around the world!

On the map below, the colours indicate the lap of our planet, according to the standard resistor colour code. In its current position over Austria the line ending is purple, indicating the 7th lap.



To return to more traditional QRSS for a moment – many of the flights have included, as part of their transmission schedule, traditional QRSS modes such as FSKCW, and Slow-Helshreiber (shifted tones that produce character patterns). More recent flights include a charming balloon glyph.

The screenshot below was recorded by me (TA4/G0UPL) on 15-Dec-2020 from my QTH in South West Turkey while U4B-13 was crossing Northern Iraq. It shows the balloon glyph and the “U4B13” text. Note that this text is produced by a series of custom shifts; text can also be produced using slow-Helshreiber which uses the original font from the Hellshreiber machines developed by Rudolf Hell in the 1920’s.



One wonders what Rudolf Hellshreiber, or Samuel Morse, would have made of all this. 1.5 gram transmitters circling the globe, transmitting milliwatts?

Weak signal modes are what makes it all possible!

Acknowledgments

My sincere thanks are due to all the radio amateurs involved in this adventure, particularly Dave VE3KCL but also including Jim N2NXZ, Daniel DL6OW and Bob ZL1RS; as well as the giants on whose shoulders we stand, Leo M0XER and Andy VK3YT. The WSPR tracking protocol development described here have made tracking floater balloons worldwide possible. The protocol has since been used and modified by other balloon enthusiasts too, including Alan W7QO, Michael KD2EAT and Bill WB8ELK.

Further reading:

http://qrp-labs.com/flights for details of all the QRP Labs test flights (67 flights in the S, U3B and U4B series launched by VE3KCL; as well as the N2NXZ, DL6OW and ZL1RS flights)
http://qrp-labs.com/circumnavigators.html “The Circumnavigators”: a page devoted to the early floater balloon circumnavigation flights, starting with M0XER in July 2014, which I believe to be the first circumnavigation flight.
https://groups.io/g/picoballoon a discussion group devoted to floater (pico) ballooning.