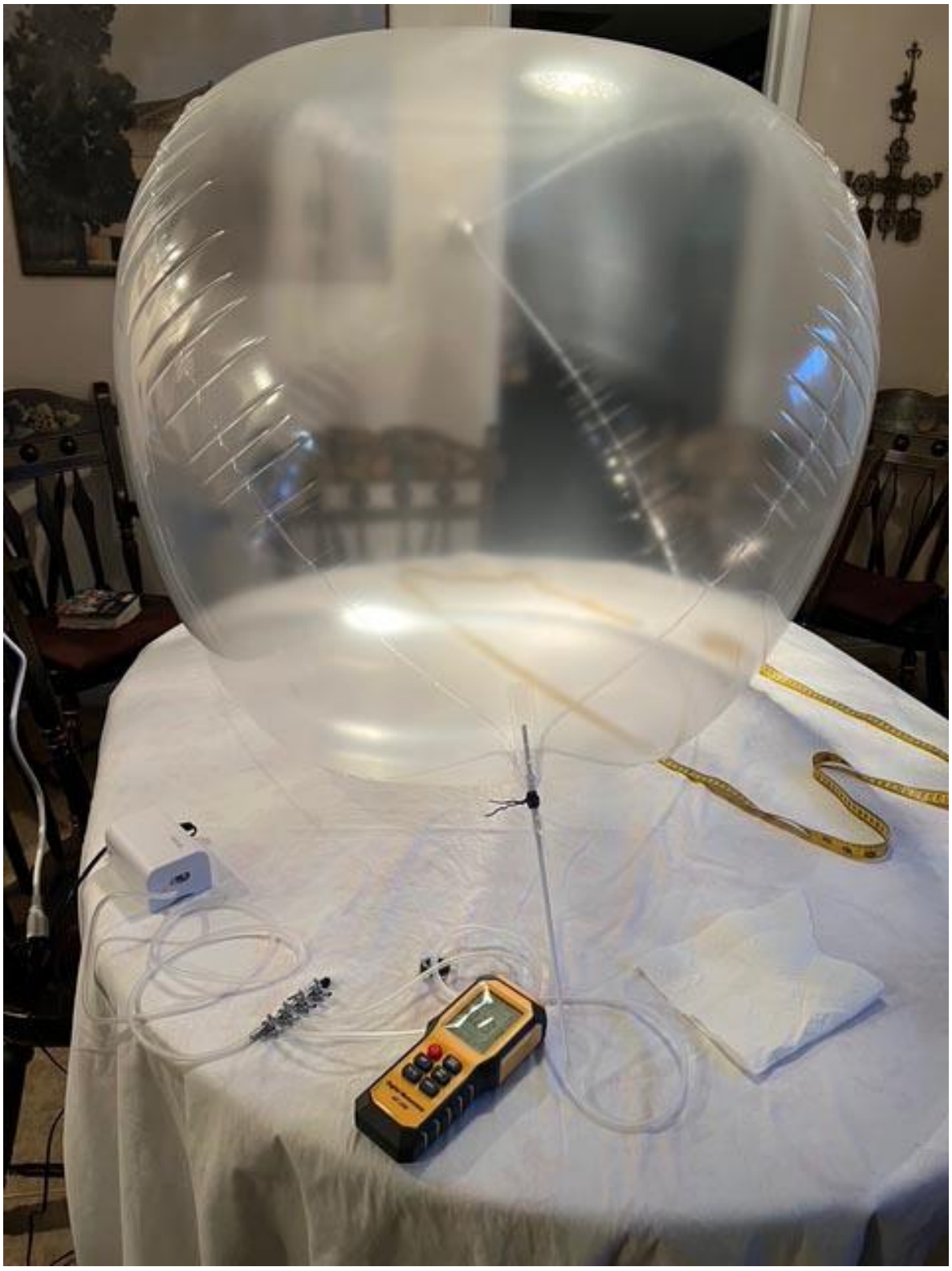
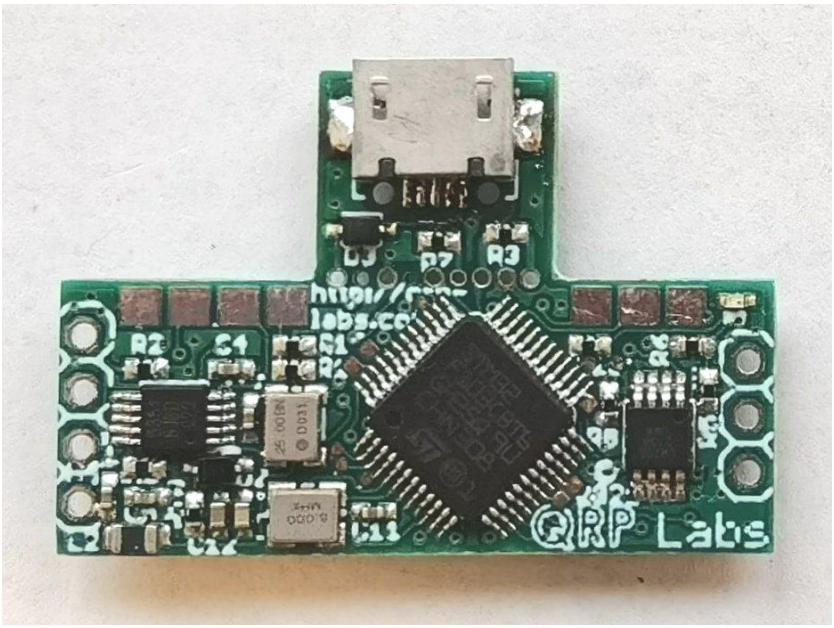
TALARC ALP40-I Balloon Construction

- Chinese 60 balloon - The balloon needs to be stretched (starts at a circumference of ~101"), so we slowly pressure fill (<0.6 psi) it up to between 101-126" in circumference using an aquarium air pump and pressure meter. The balloon for this flight was stretched to about 110" to adequately lift the 16-gram payload with 5 grams of float to reach an altitude of 43,000'. Hydrogen gas was used to fill the balloon, because it has more lift than Helium.



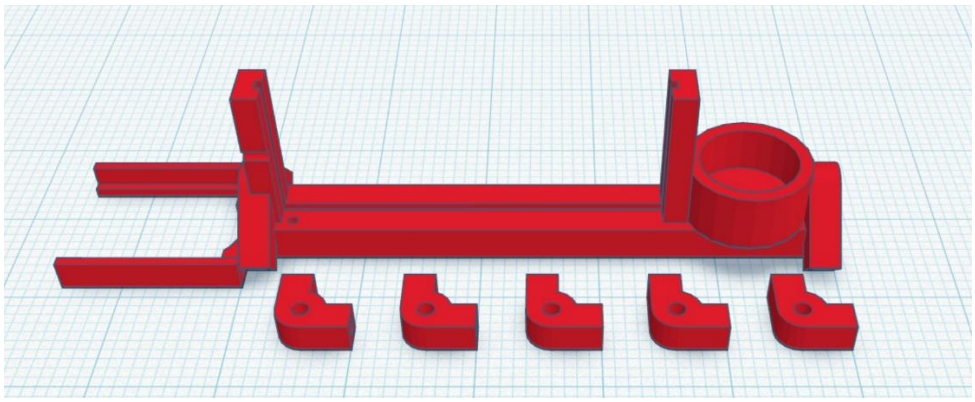
The following components were used in the construction of ALP40-I:

- QRP-Labs U4B – Electronics product description page with links to other technical documents.

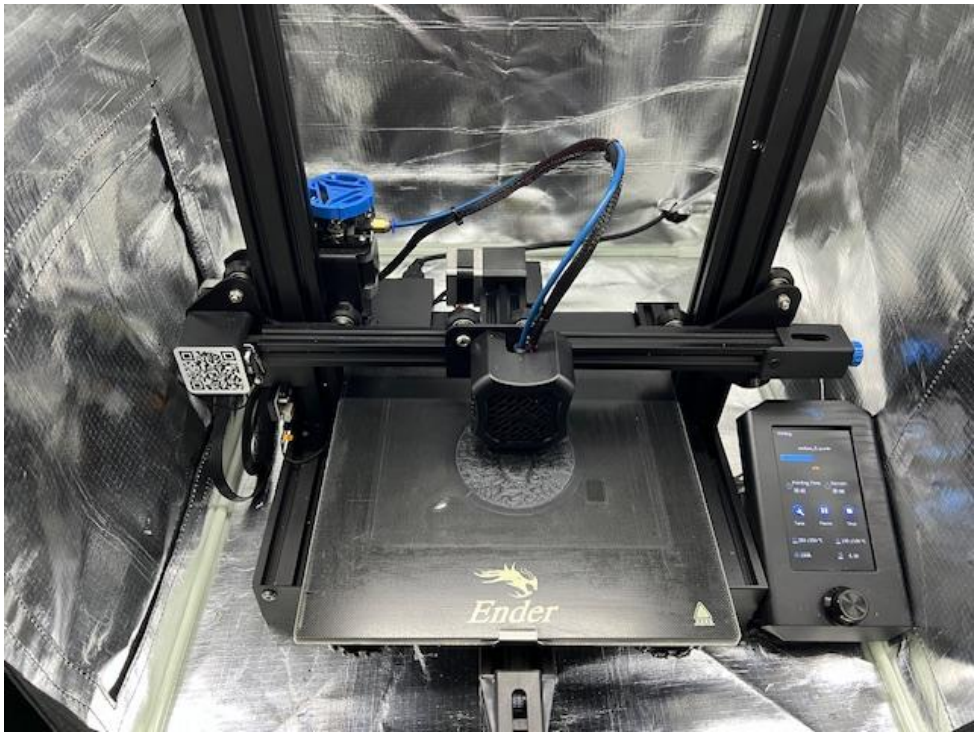


- Solar Panels - There are lots of different sizes, power, voltage and currents available. You pick the one that works best for your project. We picked a small, efficient and light one from Digi Key, the AnySolar Ltd SM141K07TF.

- Frame - Component parts were designed using Tinkercad and 3D printed using a Creality Ender 3 v2. The components were connected using 1mm carbon fiber rods and tiny drops of super glue.



Tinker CAD design photo above and Creality Ender 3v2 3D printer below



- Supercapacitor - I selected a 40 Farad 4.0 Volt capacitor for power storage. Having one of these stabilizes the voltage/current supplied to the U4B. A 3.9 Volt 0.5 watt Zener diode is added across the capacitor to act as a shunt regulator to keep it within voltage limits.

- Two 34 gauge copper wires are each cut to 8’ 4” in length, which are used for construction of the HF 10m (28.126 MHz) dipole antenna on this balloon. Fishing line is used to support one leg the copper wire going up to the balloon for connection of the payload (some builders don't use fishing line).

Photo of the antenna construction process

The frame is constructed using 3D printed ABS plastic parts and carbon fiber rods

The solar panels are mounted on ABS plastic and carbon fiber rod frame with super glue. The Schottky blocking diodes are added to the positive side of the solar panels and a straight connection is made on the negative side. Fine gauge insulated wires are added to the positive and negative side of the panels for connection to the supercapacitor, Zener diode and circuit board on the top side of the payload frame.

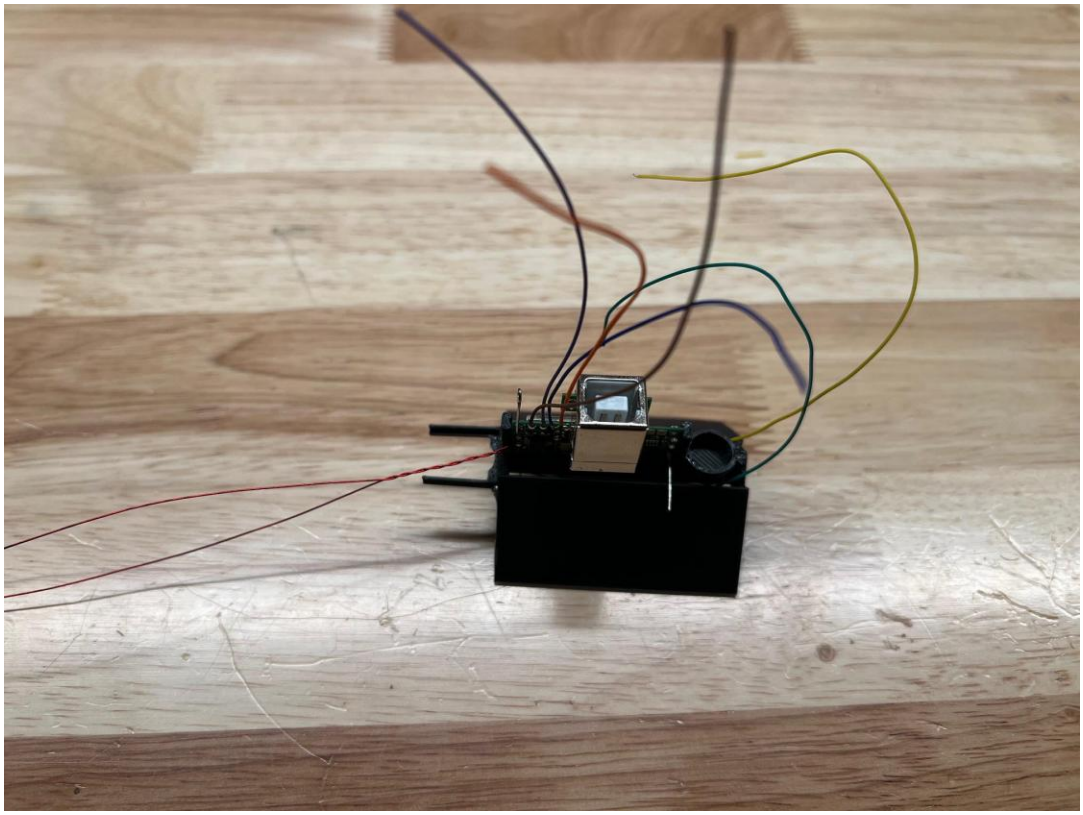
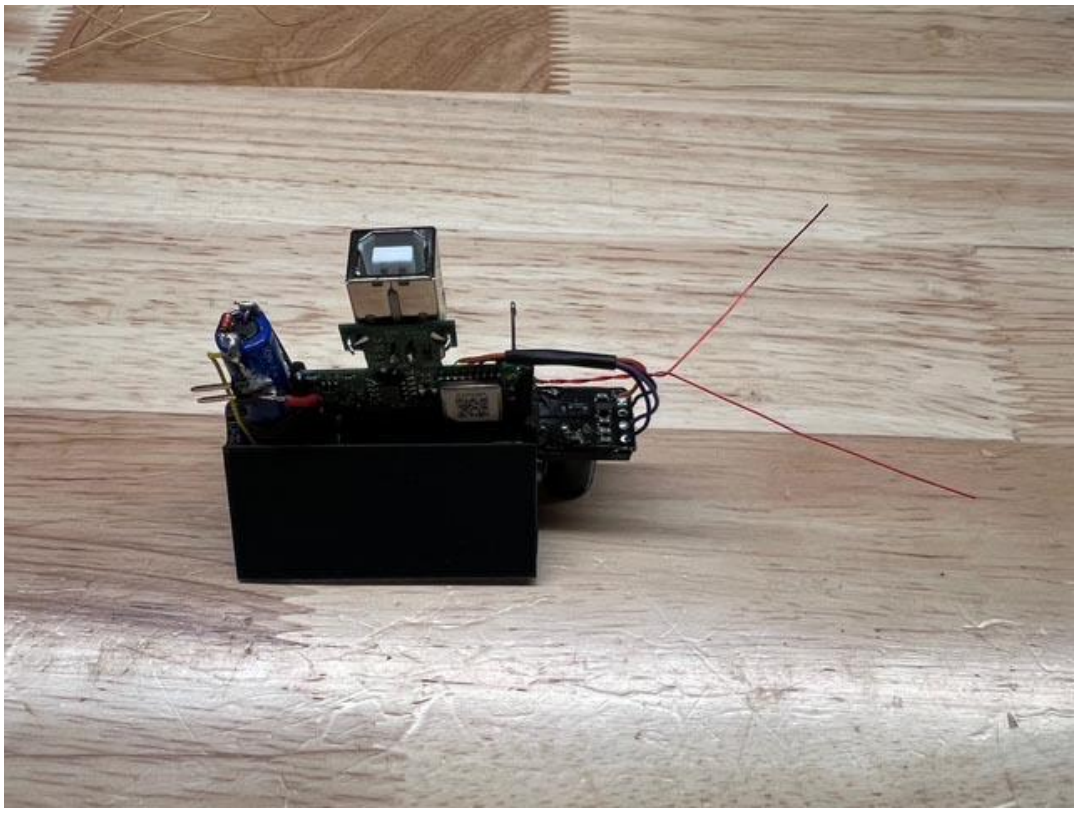


Photo of the payload prior to wiring being completed

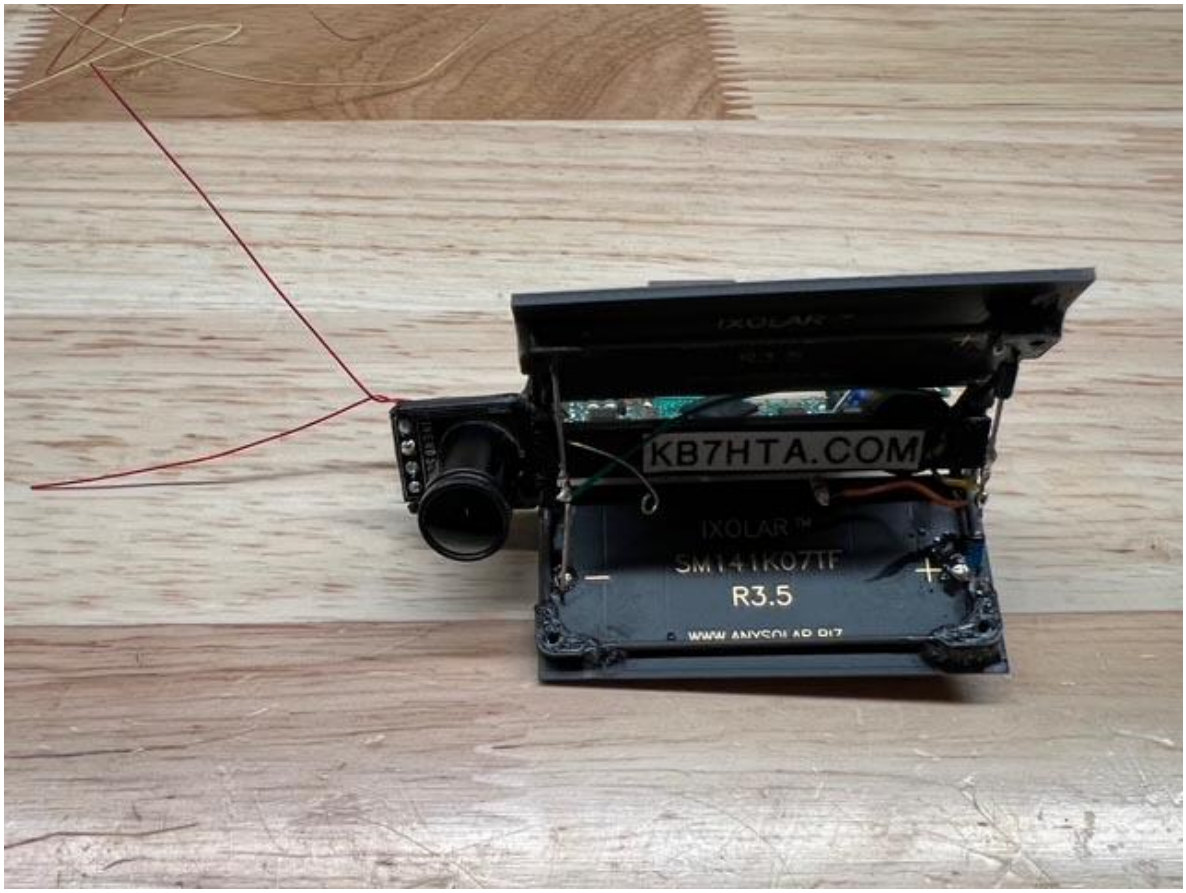
The U4B Tracker PCB is added along with the super capacitor, Zener diode, GPS antenna, Jumper and Infrared (IR) sensor.



The completed payload ready to test

The GPS antenna is the copper wires on the right. The short wire sticking up on the right is the connection point for one leg of the HF dipole antenna. The other leg is out of view under the solar panels. A jumper has been added to select flight mode. The USB connector is removed before the flight.

Bottom view of the payload showing the solar panels and TBP-I2C-H04 IR sensor



Finally, the payload is tested using flight software to make sure everything is working as expected before the flight…Test, Test and Test

TALARC ALP40-I extended telemetry is decoded from "Big Number" as follows:

Assume Big Number = 384299

Solar Panel Voltage (6 bits)						Battery Temperature (7 bits)							IR Sensor Temperature (7 bits)						
0	1	0	1	1	1	0	1	1	1	0	1	0	0	1	0	1	0	1	1
23						58							43						
384299																			

IR Sensor Temperature (Kelvin)=MOD(BigNumber,128) + 200 = 243K
Battery Temperature (Kelvin) = MOD(INT(BigNumber / 128),128) + 200 = 258K
Solar Panel Voltage (Volts) = INT(INT(BigNumber / 128) / 128) = 23 → Lookup in Table = 3.95 Volts
Note: Temperature range 200K (-73°C or -99°F) to 327K (54°C or 129°F)

Value	Volts	Value	Volts
0	2.80	32	4.40
1	2.85	33	4.45
2	2.90	34	4.50
3	2.95	35	4.55
4	3.00	36	4.60
5	3.05	37	4.65
6	3.10	38	4.70
7	3.15	39	4.75
8	3.20	40	4.80
9	3.25	41	4.85
10	3.30	42	4.90
11	3.35	43	4.95
12	3.40	44	5.00
13	3.45	45	5.05
14	3.50	46	5.10
15	3.55	47	5.15
16	3.60	48	5.20
17	3.65	49	5.25
18	3.70	50	5.30
19	3.75	51	5.35
20	3.80	52	5.40
21	3.85	53	5.45
22	3.90	54	5.50
23	3.95	55	5.55
24	4.00	56	5.60
25	4.05	57	5.65
26	4.10	58	5.70
27	4.15	59	5.75
28	4.20	60	5.80
29	4.25	61	5.85
30	4.30	62	5.90
31	4.35	63	5.95

TALARC Balloon AP40-I Flight Code

Configuration

KB7HTA-16 – 10 Meters – CH 477 – WSPR 28.126180 – CW 28.022000 – Start Minute 08/00/02 – 1st & 3rd Q -3

Variables

Letter	Initial	Description
F	0	If this flag is true (1), then the next TELE sends the external temperature vs PCB temperature.
M	4300	Below this altitude in meters, send a CW report with latitude, longitude and battery voltage
H	3900	Voltage set point for sending daytime CW
Y	0	This variable holds the result of reading the solar panel voltage after the diodes but before the Lithium battery charging circuit. There is a 6.8K Ohm divider network to cut the voltage in half to keep the voltage within the voltage range of the ADC. The full scale 12- bit ADC reading would be 4096 representing 3.3 Volts. The software multiplies this number by 2 to restore the full range of voltages 0 – 6.6, lost due to the voltage divider.
P	0	This holds the 29.180 Bit Big Number
Z	0	
A	?	Temporary Variable

B	?	Temporary Variable
C	?	Temporary Variable
D	?	Temporary Variable

```
LET HP = 1
LET P = 0
LET Z = 0
LET O = 273
LET F = 0
LET H = 3900
LET M = 4300
GPS 300
RUN “QLOCK”
10 GPS 300
OUT 19 0
SLEEP 10 7
OUT 19 1
RUN “QIR”
RUN “PACK”
TELE
TELEX 0 P
RUN “QCW”
RUN “QDWN”
GOTO 10
END
~
```

QIR – Determine If the IR Object or IR Sensor Temperature Needs to be Read and Substituted for U4B Temperature. A value of 0 K was added to ‘O’ and ‘Z’ the value and stored in the system variable TT to take care of round error between the encoding done by PACK and the encoding done on system variable.

```
IF F > 0
   RUN “RIRO”
   LET TT = O
   LET F = 0
ELSE
   RUN “RIRS”
   LET TT = Z
   LET F = 1
ENDIF
~
```

QCW – This program is run to determine, depending on battery voltage, whether to send a CW transmission.

```
IF BT > H
   CW 0 28022000 12 0 “CQ KB7HTA KB7HTA BALLOON #M6 #AT #BT”
ENDIF
~
```

FLASH – This program is used to send the number of light flashes specified by the variable A.

```
FOR X = 1 TO A
OUT 19 0
DELAY 1000
OUT 19 1
DELAY 1000
NEXT
~
```

QLOCK – This program determines if the balloon has a GPS lock. If there is a lock it turns off the LED, otherwise it flashes the LED 10 times and leaves it lit.

```
IF GL = 0
LET A = 10
RUN “FLASH”
OUT 19 1
ELSE
OUT 19 0
ENDIF
~
```

QDWN – This program determines if the balloon is possibly going down. It sends CW Latitude Longitude and battery voltage if the balloon goes below the value of variable M, which is usually 4300 meters or 14,000’.

```
IF AT < M
CW 0 146585000 12 0 “KB7HTA #LT #LN #LT #LN #BT”
ENDIF
~
```

RIRO – Read the IR Sensor Object Temperature

```
LET D = 0
LET O = 0
FOR X = 1 TO 5
   LET D = I2CR 117 7 3
   LET D = D % 65536
   LET O = O + D
NEXT
LET O = 2 * O / 500
~
```

RIRS – Read the IR Sensor PCB Temperature

```
LET D = 0
LET Z = 0
FOR X = 1 TO 5
   LET D = I2CR 117 6 3
   LET D = D % 65536
```

```
    LET Z = Z + D
NEXT
LET Z = 2 * Z / 500
~
```

RSPV – Read the solar panel voltage after the diodes.

```
FOR X = 1 TO 5
LET Y = Y + INA 7
DELAY 10
NEXT
LET A = 0
LET Y = Y / 5
LET Y = Y * 1000 / 4096
LET Y = Y * 3300 / 10000
LET Y = Y * 2
IF Y > 280
    LET Y = ( Y – 280 ) / 5
ELSE
    LET Y = 0
ENDIF
~
```

PACK – Package telemetry, IR sensor Object temperature (K), IR Sensor PCB temperature (K) and solar panel voltage index (0 – 63) representing 2.80 to 5.95 Volts with a resolution of .05 Volts.

```
LET P = 0
RUN “RSPV”
LET P = ( P + Y ) * 128
RUN “RIRS”
LET P = ( P + ( Z – 200 ) ) * 128
RUN “RIRO”
LET P = P + ( O – 200 )
~
```

Balloon Mass Analysis

3D ABS Print:	3.66 grams	
IR Temperature Sensor:	2.36 grams	
U4B Tracker:	1.80 grams	
Solar Panels x 2:	4.60 grams	
Miscellaneous:	2.34 grams	
Payload Mass:	14.60 grams	
Dipole Antenna Top:	0.80 grams	--- #34 gauge copper wire 8’ 4” with 6# fishing line and silicone
Dipole Antenna Bot:	0.50 grams	--- #34 gauge copper wire 8’ 4”
Total Payload:	15.90 grams	
Float:	5.50 grams	
Neck Weight:	21.40 grams	