Kyutech
Kyushu Institute of Technology
国立大学法人
九州工業大学

La SEINE

BIRDS-X Project

APRS Student Mission Ground Terminal

VOLCANO MONITORING

Volcano harm

- About 1500 active volcanos in the world
- There is volcano disaster it has victims every 10~30 years (only in Japan)
- Eruption affect to space (gas and particle generated by eruption causes unusual currents of extreme winds)

By monitoring volcano

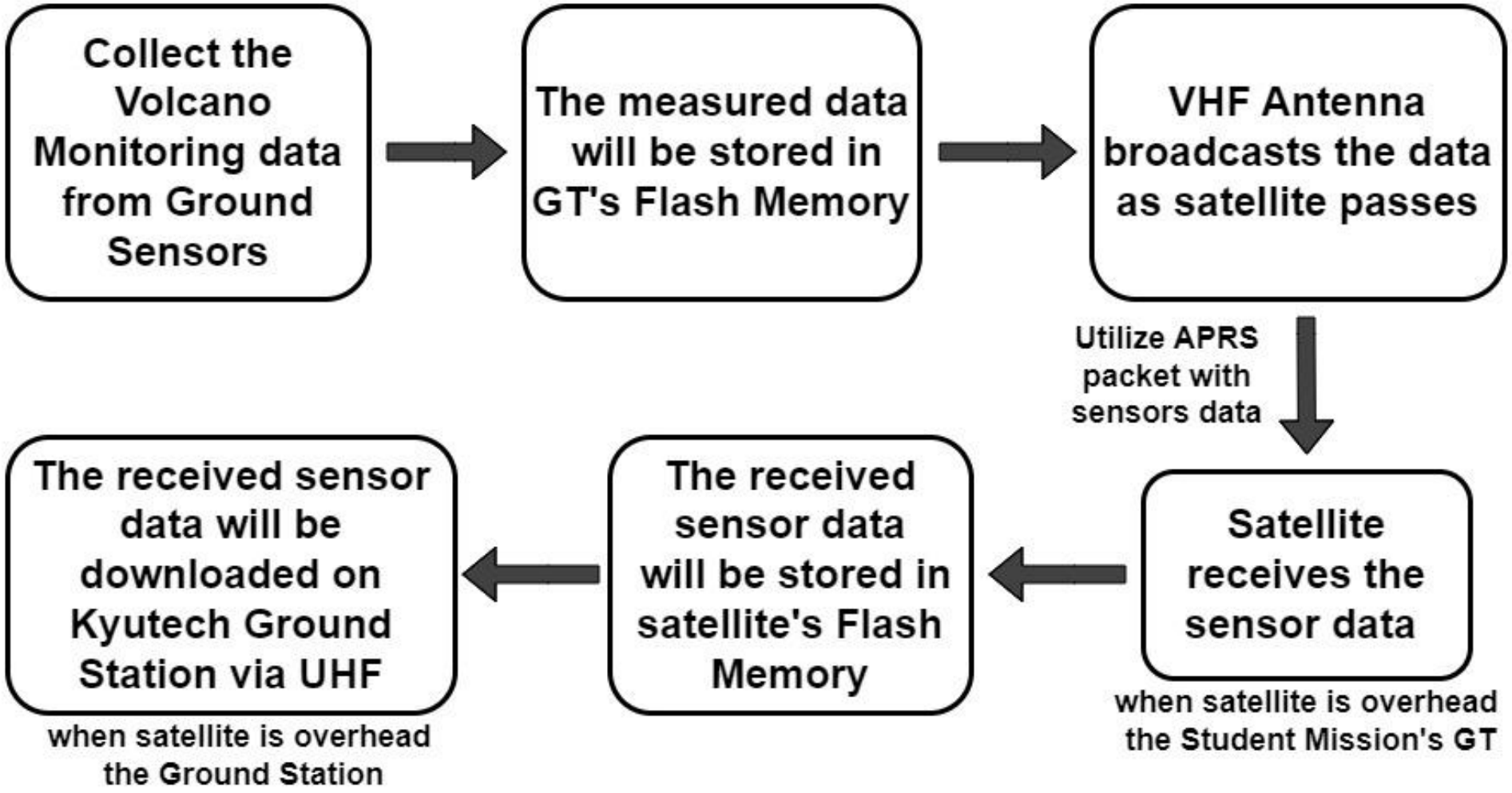
- Predict volcano eruption
- Avoid disaster
- Understand earth activity

https://www.carsensor.net/contents/article_images/_65319/eruption_bousai_01.jpg

VOLCANO MONITORING

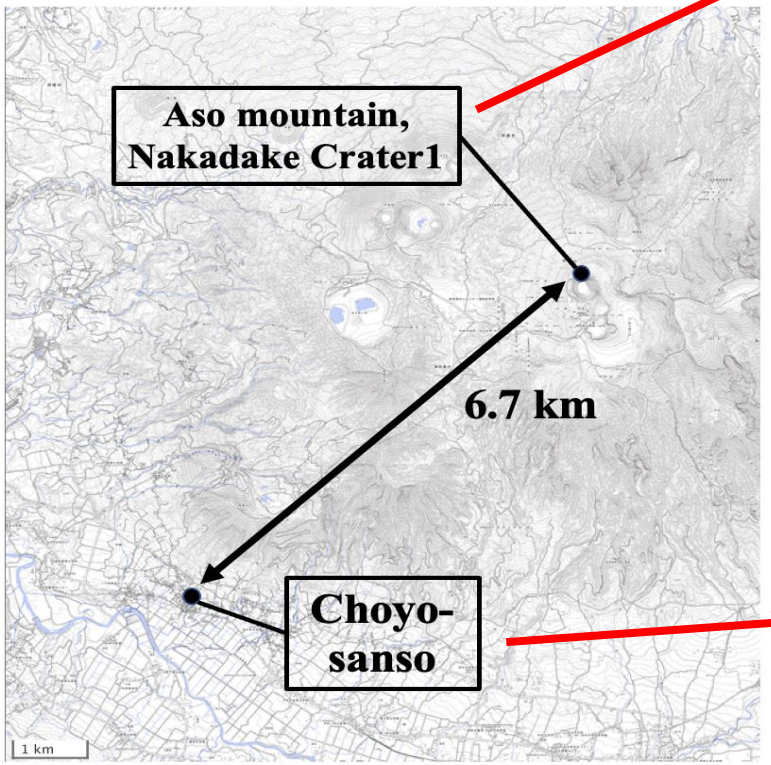
- **Problem:** Difficulty in gathering the data from a remote area
  - No internet connection
  - Hard situation to observe by human
  - Difficult to get data continuously
- **Solution:** Use satellite as a platform to collect data from sensors for volcano monitoring
- **Mission objectives:**
  - To operate the satellite to store and forward data acquired from the remote area to the ground station.
  - To monitor the volcano activity and collect the data through our satellite by using APRS

MISSION SCENARIO



EXPECTED LOCATION

- “Choyo-Sanso” Kyutech Retreat Facility near the Aso mountain
- Active volcano
- 6.7 km from the Nakadake Crater 1



Sensors for volcano monitoring:

1. Gas sensor

DF Robot SEN0470 (SO2)

(measures gas concentration of SO2)

Specification

- Detects: Sulfur Dioxide (SO2)
- Temperature: -20 to 50 ℃
- Detection range: 0 - 20 ppml
- Protocols: I2C, UART

2. Weather windmeter

DS-15901

(measures speed and direction of wind in the vicinity of volcano)

Specification

- Detects: Wind speed & direction
- Including:
  - Wind vane
  - Anemometer

3. Thermal sensor

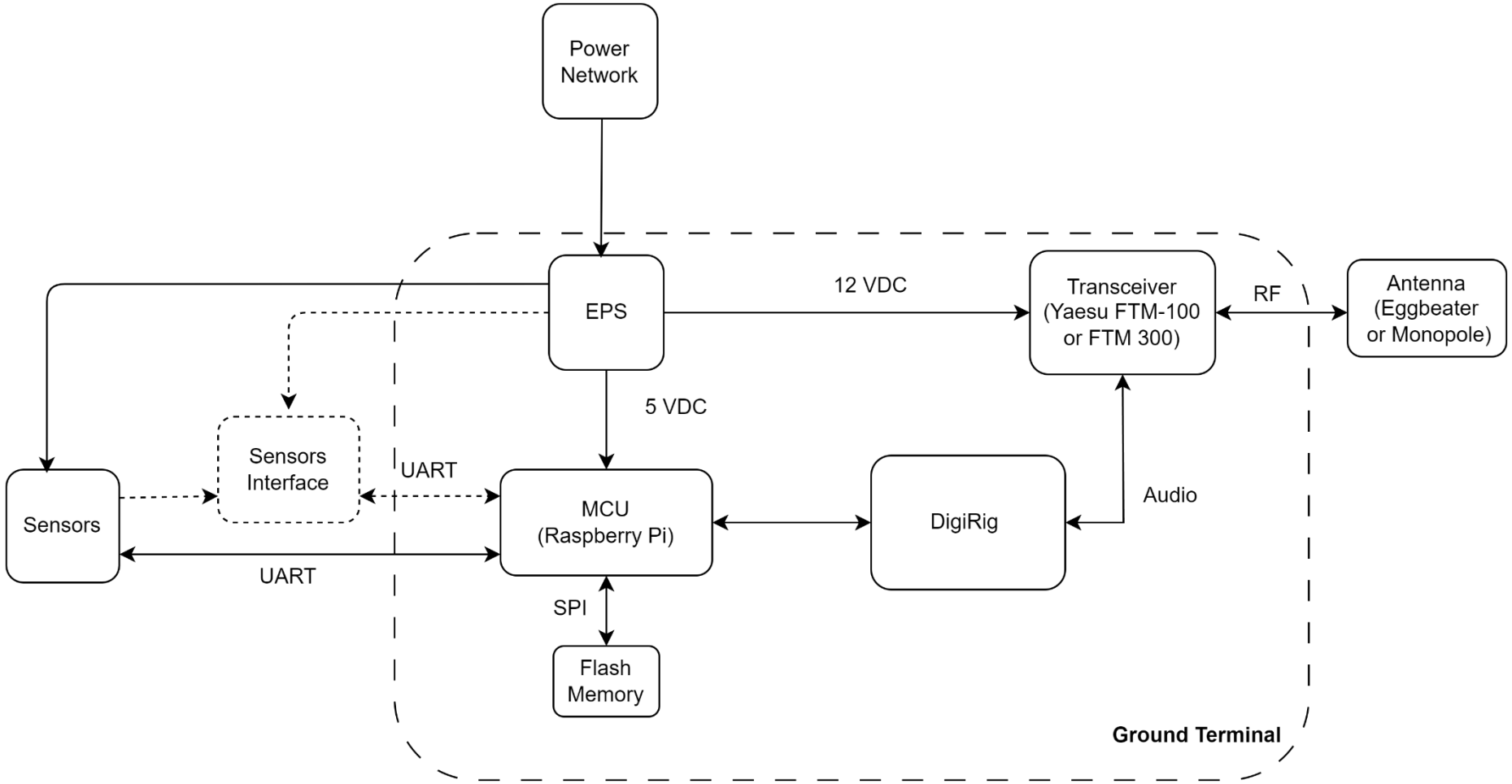
MLX90614ESF-ACK-000-SP

(measures change in temperature of volcanic vents)

Specification

- Detects : Temperature change
- Sensing temperature (Local): -40℃ to 85℃
- Sensing temperature (Remote): -70℃ to 380℃
- Temperature resolution: 0.02℃

Ground Sensor Terminal Block Diagram



Ground Sensor Terminal (GST)

Eggbeater Antenna 145 MHz

Tripod Stand

Wind Sensor

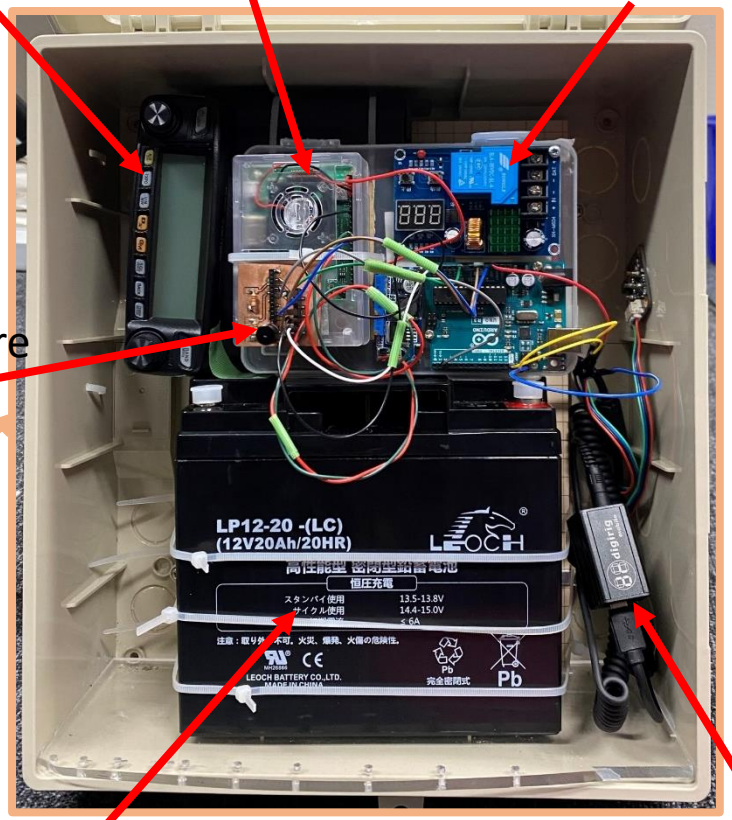
Temperature Sensor

Electronics Box

Raspberry pi

Radio Transceiver

Battery Charge Controller



Front View

Car Battery [12V 20Ah]

Digirig

Left Side View

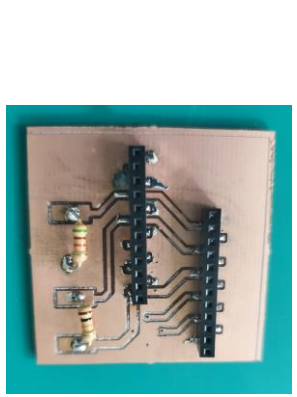
SO2 Sensor

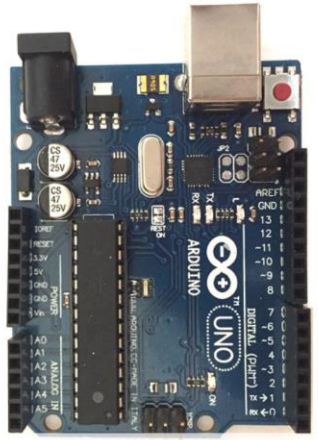
Components of the GST

Microcomputer Unit (MCU): Raspberry Pi 3B

RF Transceiver: Yaesu FTM 100-DH

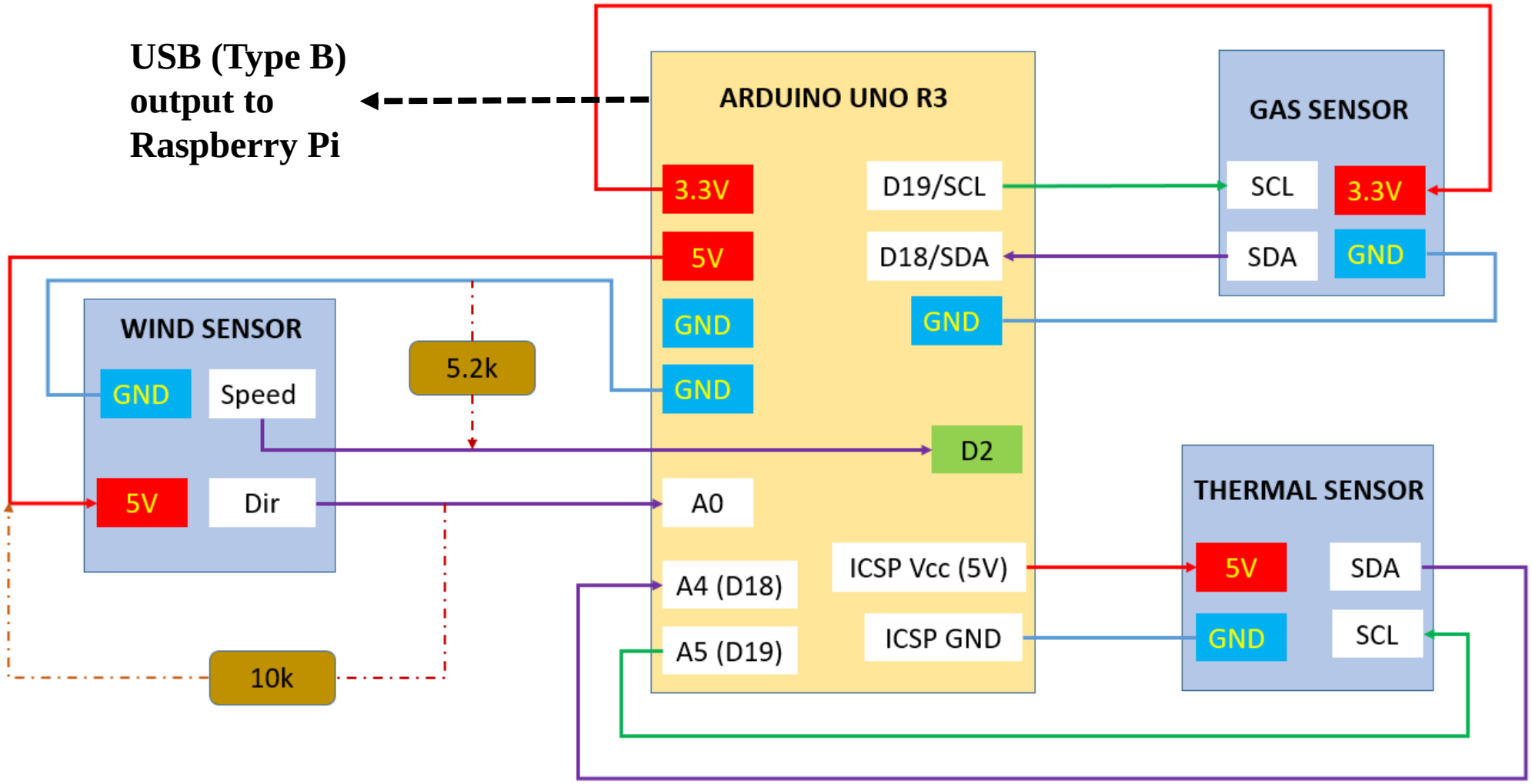
Antenna:
Eggbeater antenna





Sensor board and Sensor Interface (Arduino UNO R3)

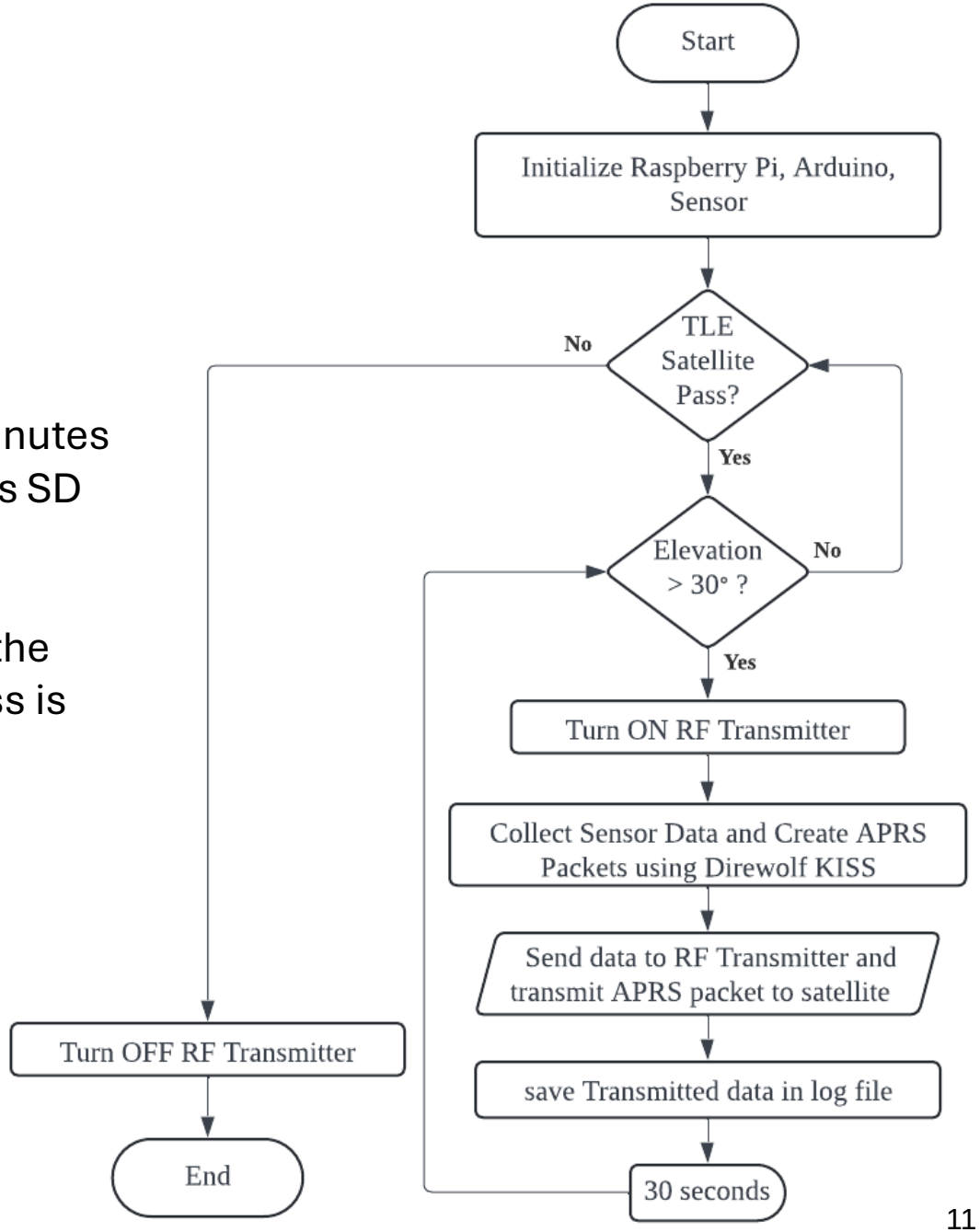
Digirig mobile



Sensors – Sensor Interface connection Block Diagram

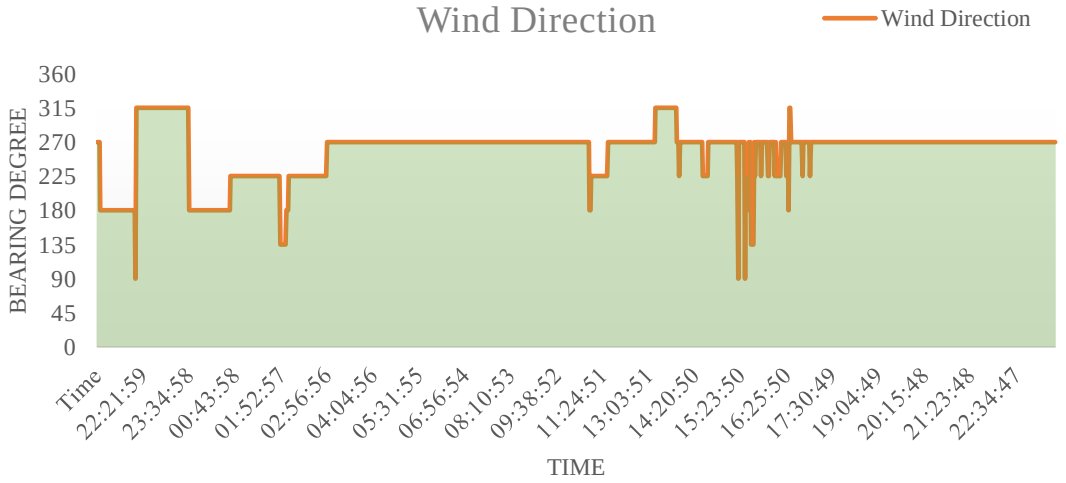
Operation Flowchart

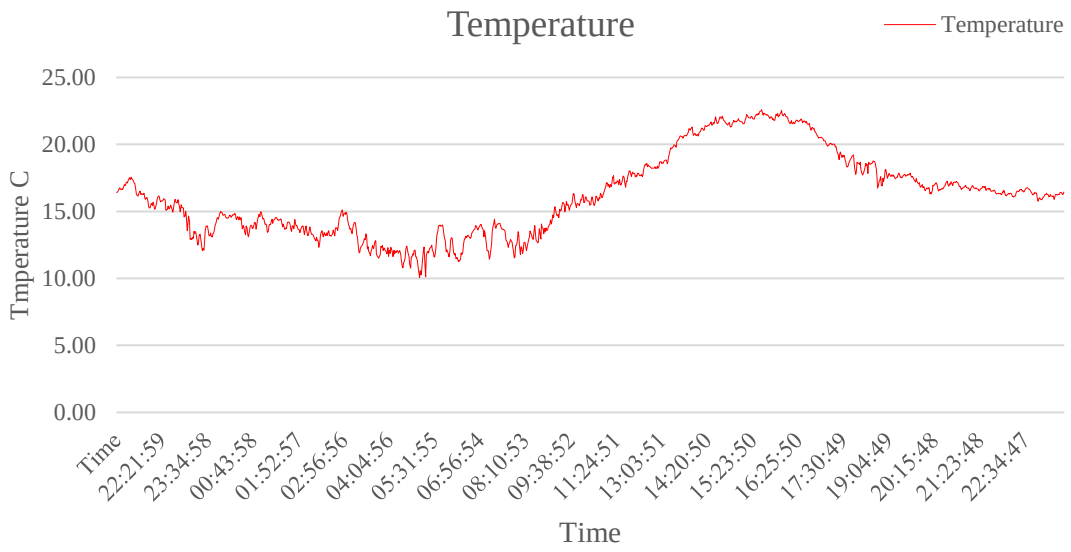
- Raspberry Pi, Arduino, Sensors and DC/DC buck converter will be turned on all the time
  - Sensors will capture Volcano data every two minutes
  - Volcano data will be stored in the Raspberry Pi's SD card
- YAESU transceiver will be turned on around 10 mins before a satellite pass, and be turned off right after the satellite pass (assume that the duration of each pass is approximately 10 minutes)
  - The 15W transmission mode will be chosen as default TX mode
  - The transmitting time is **2 minutes**
  - Each transmission will be **30 seconds apart**

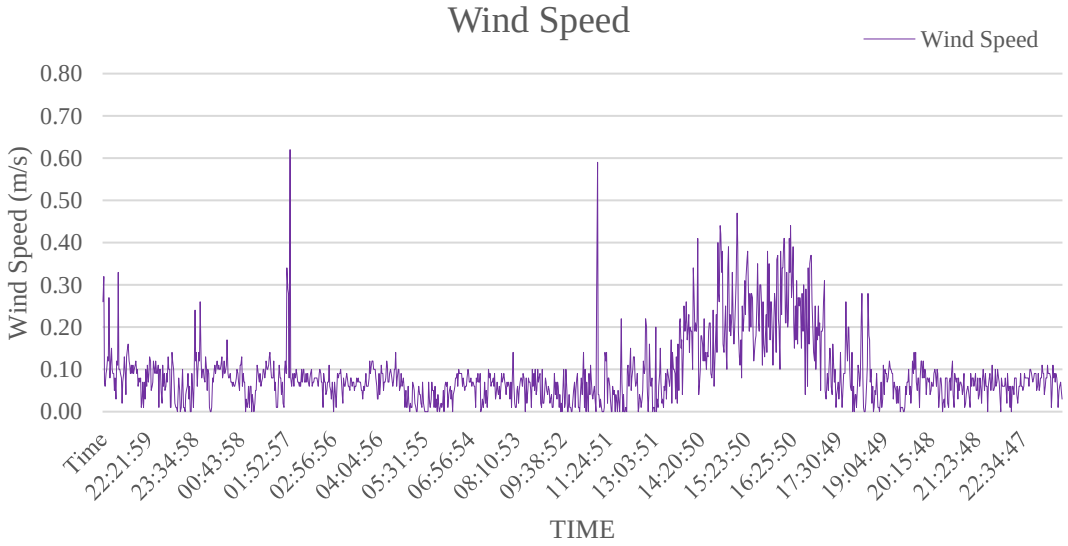


Data Collection Test

Duration: ***26 hours 24 mins***



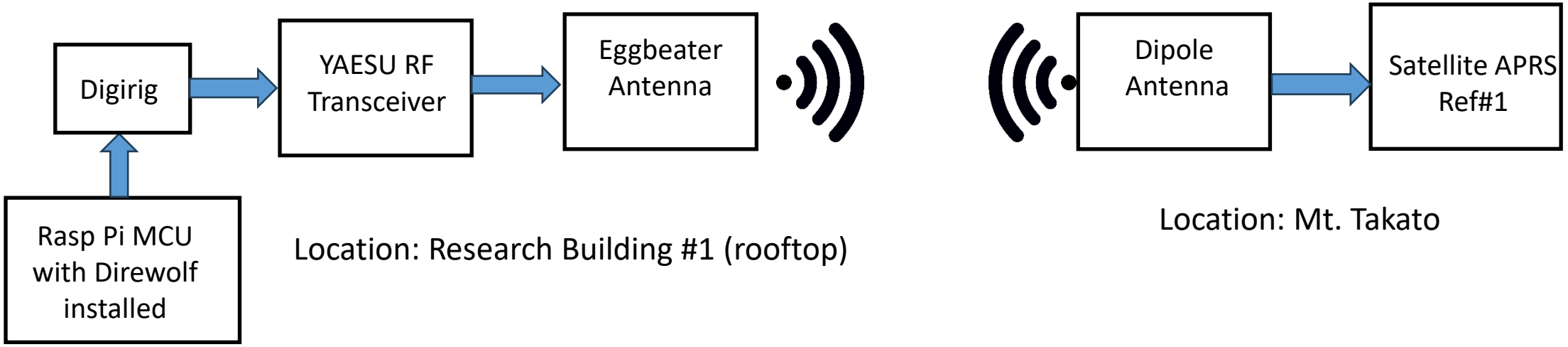




Summary of the data results

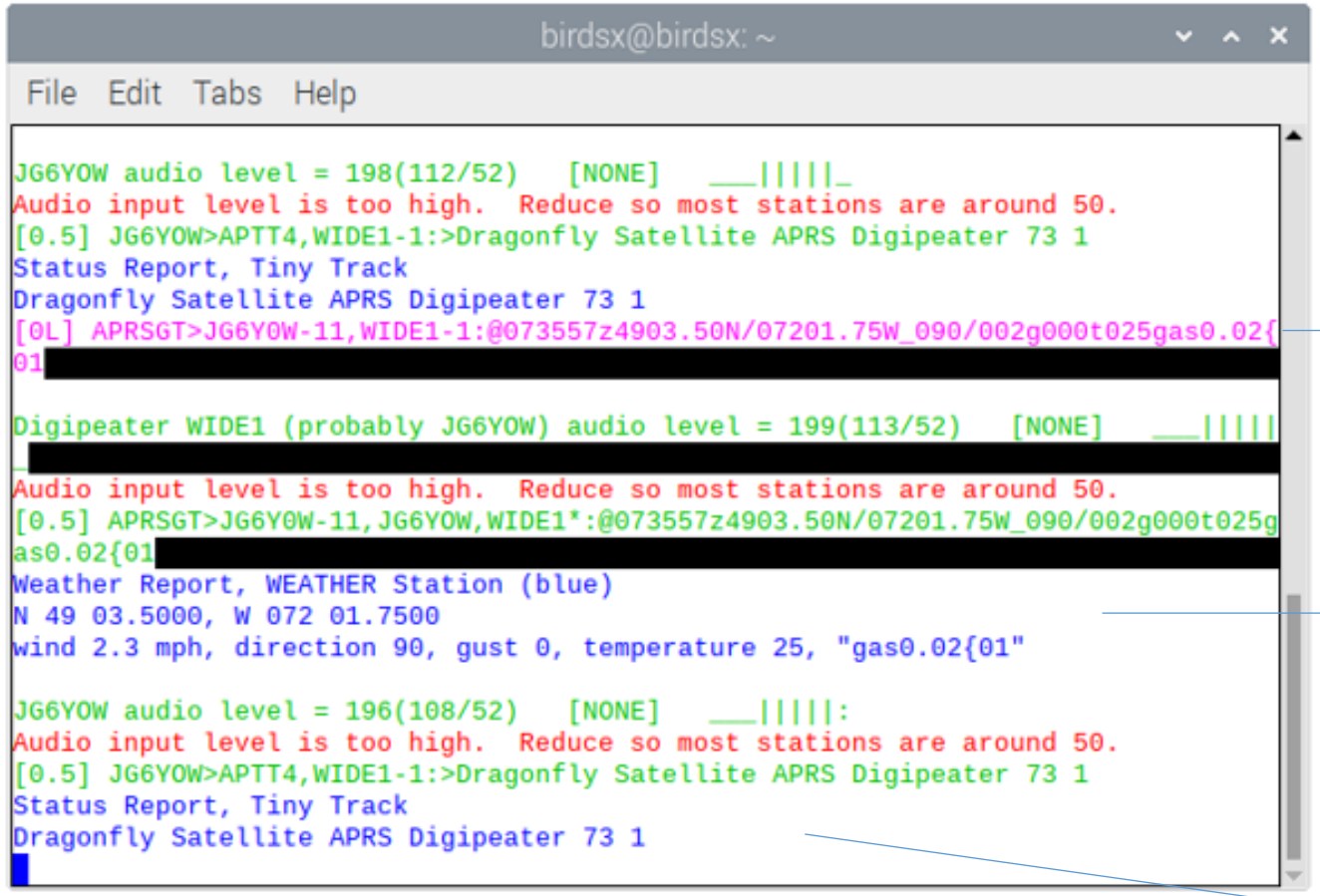
Sensor data	Max	Min	Avg
Temperature	22.59 ºC	9.93 ºC	16.07 ºC
Wind Speed	0.62 m/s	0 m/s	0.07 m/s
	Most dominant wd		**Duration**
Wind direction	South West (270º)		18h 48m

RF Test (Long range)



- An APRS message was transmitted from GST to satellite's APRS REF#1 at VHF
  - Transmitted APRS message from APRS GST: APRSGT>JG6Y0W-11,WIDE1-1:@073557z4903.50N/07201.75W_090/002g000t025gas0.02{01
- Satellite received this message, then digipeated back to GST

RF Test (Long range)



The transmitted APRS signal

The content of the decoded APRS signal

The digipeated message from satellite

Link Budget Analysis (VHF)

Parameters	Unit	El	El	El	El	El	El	El	El	El
		0	*10*	*20*	*30*	*40*	*50*	*60*	*75*	*90*
Frequency	MHz	145.825	145.825	145.825	145.825	145.825	145.825	145.825	145.825	145.825
At the Ground Station Terminal side										
EIRP (**5W**)	dBm	20	26	32	36	38	39	41.5	41.6	41.62
EIRP (**15W**)	dBm	24.76	30.76	36.76	40.76	42.76	43.76	46.26	46.36	46.38
EIRP (**50W**)	dBm	30	36	42	46	48	49	51.5	51.6	51.62
TOTAL LOSS	dB	151.88	147.84	144.54	142.05	140.21	138.86	137.88	137.00	136.72
At the Satellite Side										
Rcv power at TRX (for **5W** TX case)	dBm	-134.28	-124.24	-114.94	-108.45	-104.61	-102.26	-98.78	-97.8	-97.5
Rcv power at TRX (for **15W** TX case)	dBm	-129.52	-119.48	-110.18	-103.69	-99.85	-97.5	-94.02	-93.04	-92.74
Rcv power at TRX (for **50W** TX case)	dBm	-124.28	-114.24	-104.94	-98.45	-94.61	-92.26	-88.78	-87.8	-87.5
Required power at TRX	dBm	-105	-105	-105	-105	-105	-105	-105	-105	-105
LINK MARGIN										
*** for 5W transmission scenario**	dB	***-29.28***	***-19.24***	***-9.94***	***-3.45***	***0.39***	***2.74***	***6.22***	***7.20***	***7.50***
*** for 15W transmission scenario**	dB	***-24.52***	***-14.48***	***-5.18***	***1.31***	***5.15***	***7.50***	***10.98***	***11.96***	***12.26***
*** for 50W transmission scenario**	dB	***-19.28***	***-9.24***	***0.06***	***6.55***	***10.39***	***12.74***	***16.22***	***17.20***	***17.50***

Good values

Power Budget Analysis

Components		Current consumption (A)	Operating Voltage (V)	Power consumption (W)	Duty Cycle (h)	Energy per day (Wh)
Raspberry Pi, Arduino and Sensors		0.68	5	3.4	24	81.6
RF Transceiver (TX mode)	**5W**	2.24	12.8	28.67	0.033	0.06
	15W	4.12	12.8	52.74	0.033	0.12
	50W	9	12.8	115.2	0.033	0.26
(RX mode)		0.58	12.8	7.42	0.33	2.45

Total energy consumption per day: **~84.3 Wh**

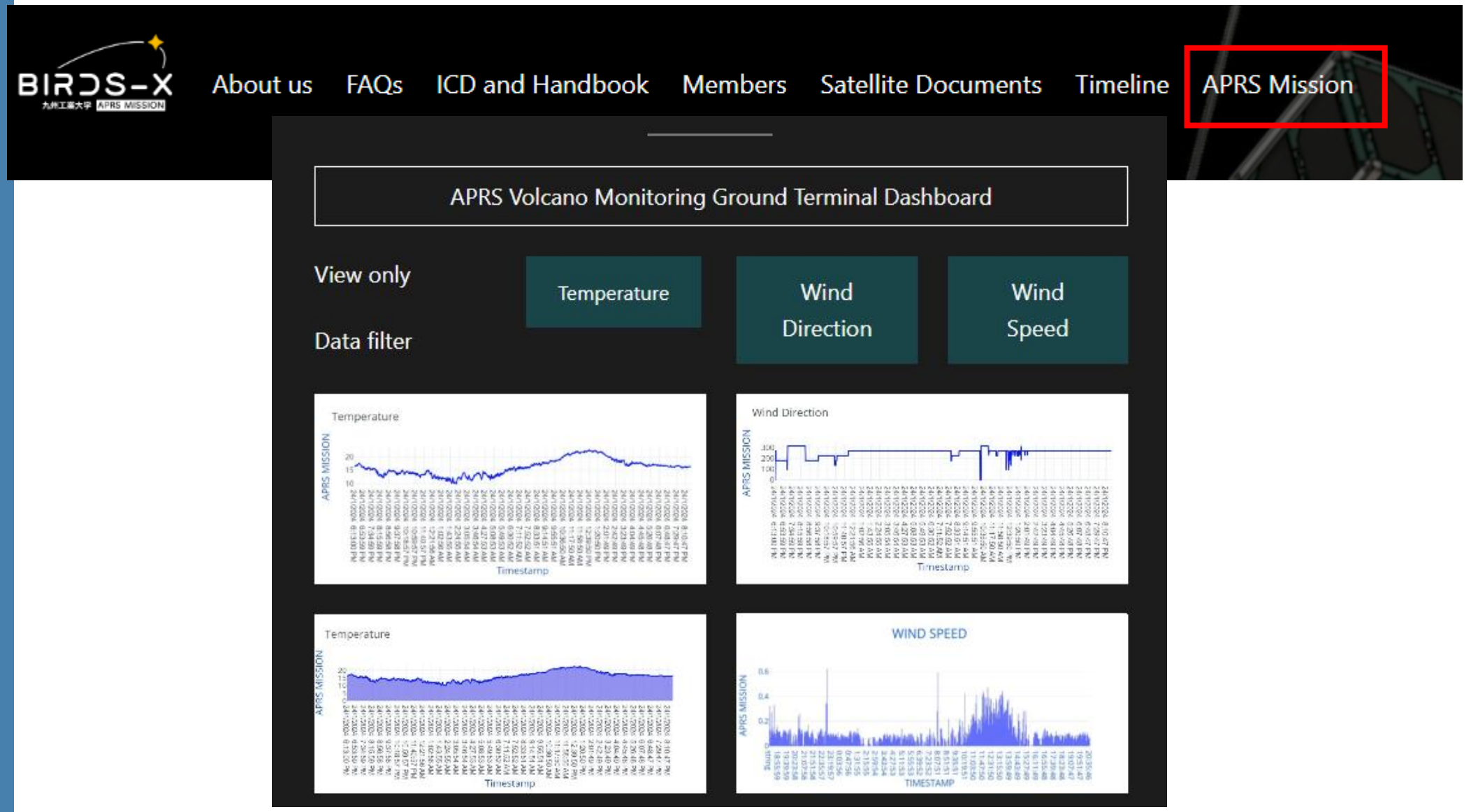
Data Budget Analysis

Data Budget in a day when there is a satellite pass

	Parameters	Values	Units	Notes
A	Mean time per satellite pass	476.6	s	STK access simulation
B	Effective time for data collection	238.3	s	Assume that is only 50% of A
C	Mean duration of each access with good communication	120	s	Analysis from the AER result, with respect to the condition El ≥ 33º
D	Maximum amount of data that can be collected for each 30 seconds	67	bytes	Assume that the whole length of an APRS frame must be used
E	Maximum amount of data that can be collected when considering B	532	bytes	= (B/30)*D; corresponds to ***7 APRS packets***
F	Maximum amount of data that can be collected when considering C	268	bytes	= (C/30)*D; corresponds to ***4 APRS packets***

Data Dashboard

https://birds-x.birds-project.com/



Appendix

APRS Data Format

- Formatted the output sensor data in APRS format:
  - ➢ Chosen format: ***Complete Weather report*** (with normal Lat/Long, timestamp)

Complete Weather Report Format — with Lat/Long position and Timestamp									
/ or @	*Time DHM / HMS*	*Lat*	*Sym Table ID*	*Long*	*Symbol Code*	*Wind Directn/ Speed*	*Weather Data*	*APRS Software* S	*WX Unit* uuuu
1	7	8	1	9	1	7	n	1	2-4

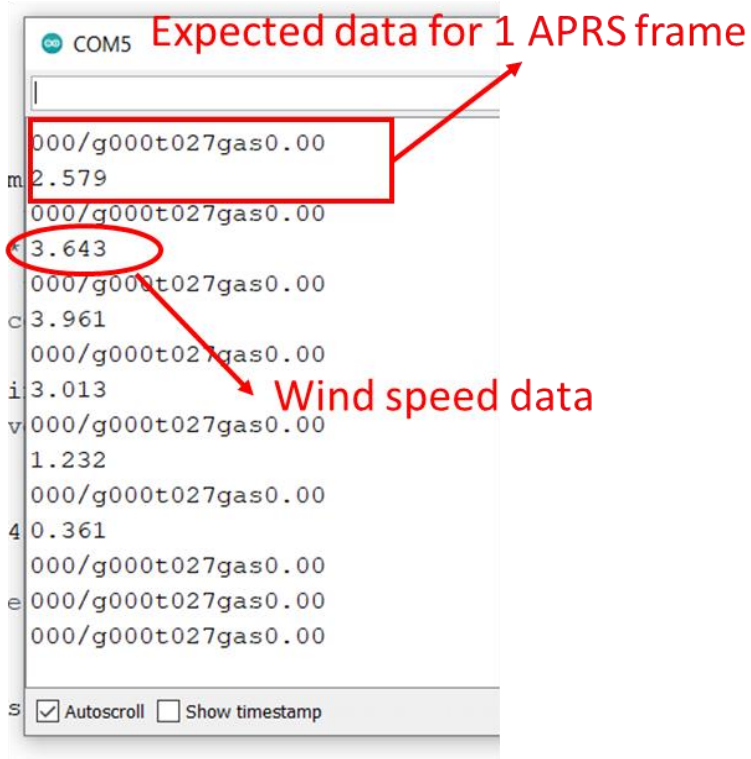
The results appear in the Direwolf's decode window:

```
APRSGT>JG6YOW-11,WIDE1-1:@092345z4903.50N/07201.75W_090/002g000t025gas0.02
Weather Report, WEATHER Station (blue)
N 49 03.5000, W 072 01.7500
wind 2.3 mph, direction 90, gust 0, temperature 25, "gas0.02"
```

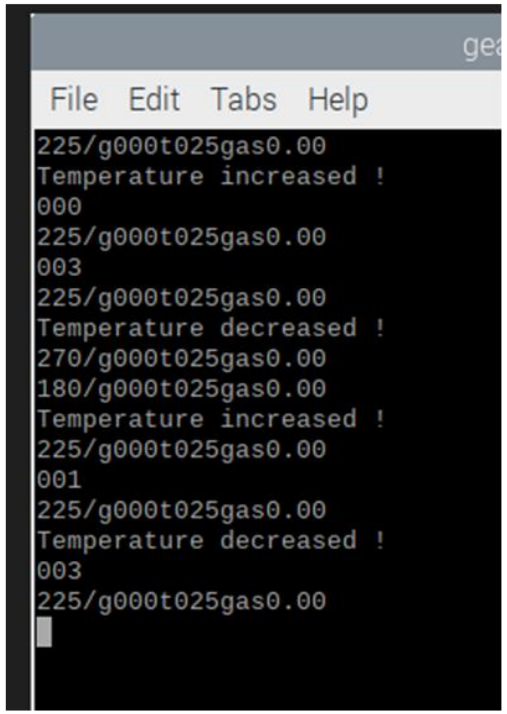
sio

APRSGT>JG6YOW-11,WIDE1-1:@092345z4903.50N/07201.75W_090/002g000t025**gas0.02**

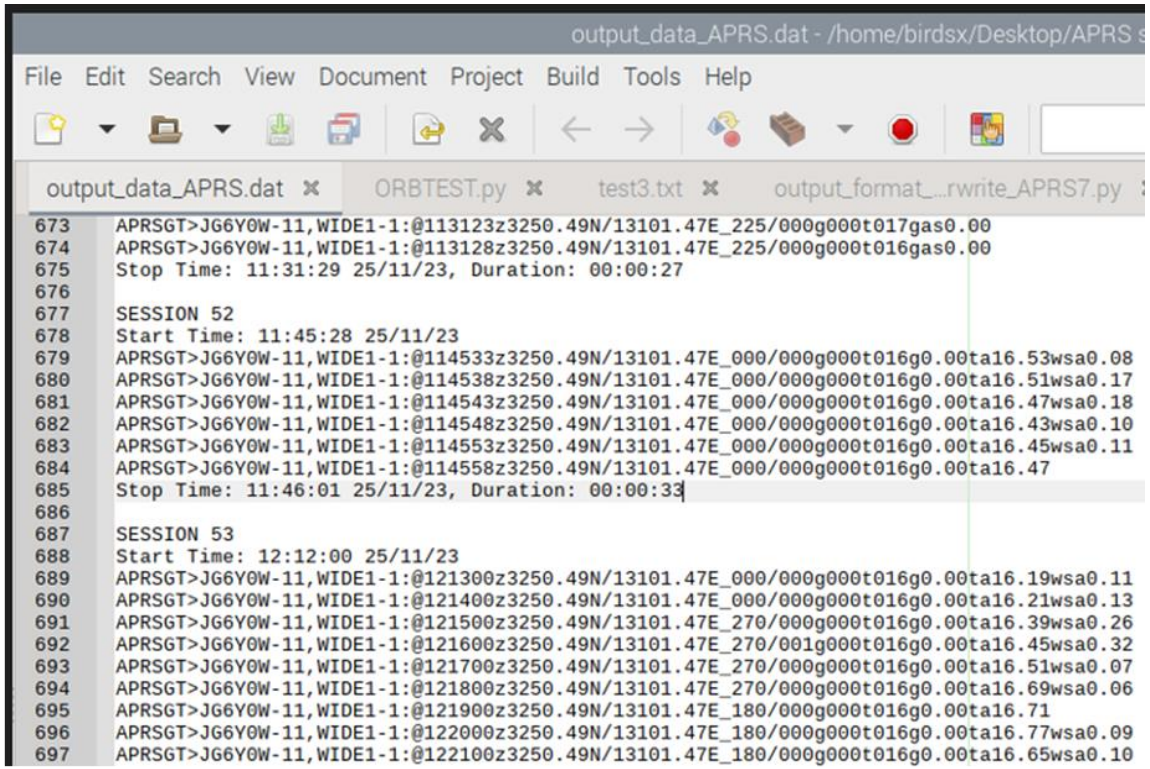
GST Software



Sensor data collection by Arduino



Sensor data received from Arduino by Rasp Pi



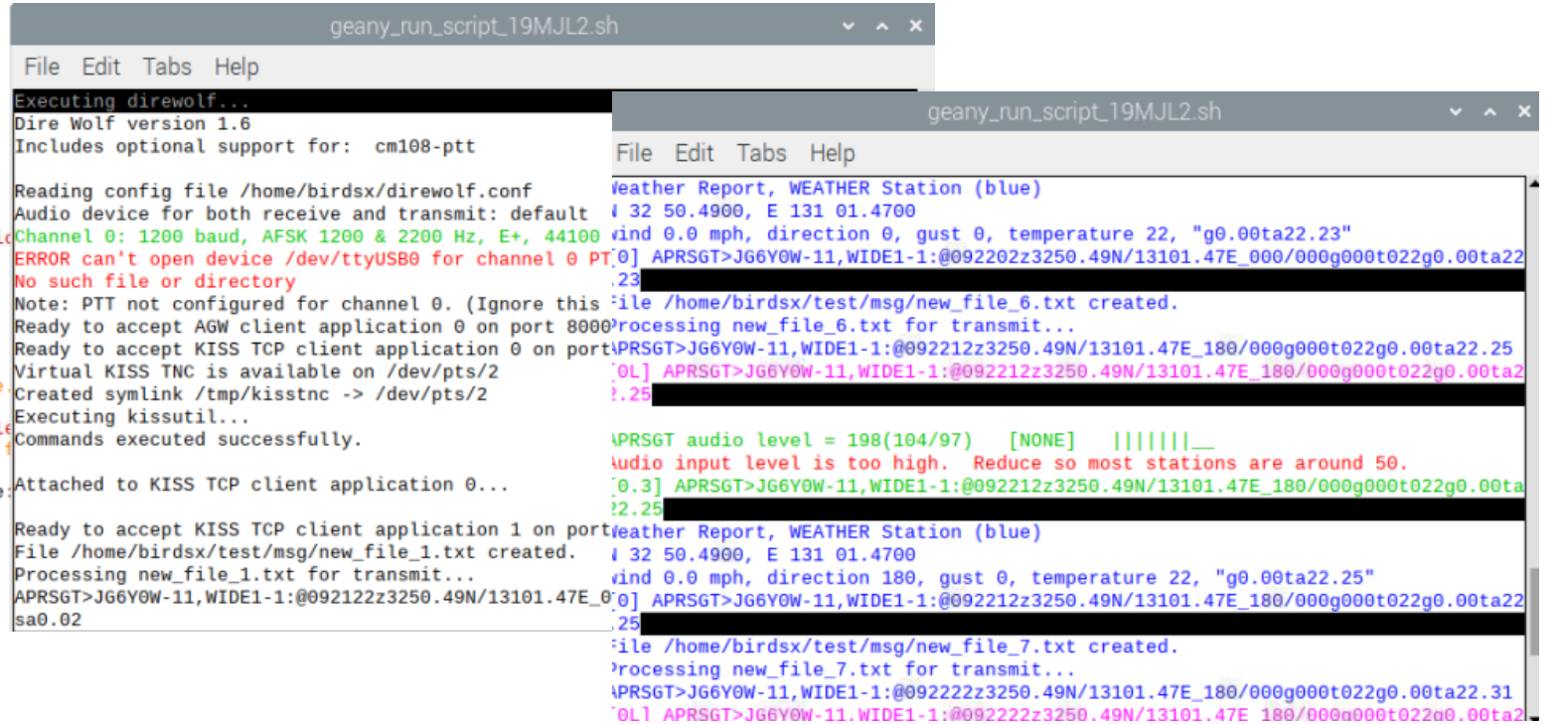
The APRS-formatted data stored in log file

GST Software

aprs_output.txt | output_data.dat | orbtest.log | elevation_log.txt | ORBTEST

```
1  APRSGT>JG6Y0W-11,WIDE1-1:@092122z3250.49N/13101.47E_090/000g000t022g0.00ta22.15wsa0.02
2  APRSGT>JG6Y0W-11,WIDE1-1:@092132z3250.49N/13101.47E_090/002g000t022g0.00ta22.15wsa1.05
3  APRSGT>JG6Y0W-11,WIDE1-1:@092142z3250.49N/13101.47E_090/002g000t022g0.00ta22.15wsa1.22
4  APRSGT>JG6Y0W-11,WIDE1-1:@092152z3250.49N/13101.47E_090/000g000t022g0.00ta22.21
5  APRSGT>JG6Y0W-11,WIDE1-1:@092202z3250.49N/13101.47E_000/000g000t022g0.00ta22.23
6  APRSGT>JG6Y0W-11,WIDE1-1:@092212z3250.49N/13101.47E_180/000g000t022g0.00ta22.25
7  APRSGT>JG6Y0W-11,WIDE1-1:@092222z3250.49N/13101.47E_180/000g000t022g0.00ta22.31
8  APRSGT>JG6Y0W-11,WIDE1-1:@092232z3250.49N/13101.47E_180/000g000t022g0.00ta22.35
9  APRSGT>JG6Y0W-11,WIDE1-1:@092242z3250.49N/13101.47E_180/003g000t022g0.00ta22.37wsa1.52
10 APRSGT>JG6Y0W-11,WIDE1-1:@092252z3250.49N/13101.47E_180/002g000t022g0.00ta22.43wsa1.25
11 APRSGT>JG6Y0W-11,WIDE1-1:@092302z3250.49N/13101.47E_090/001g000t022g0.00ta22.51wsa0.37
12
```

The APRS output file that extracts data from the log file



Direwolf reads the content in APRS output file and transmit to transceiver by KISS protocol and digital mode interface (Digirig)

GST Software

1. The collected sensor data appeared in the Arduino IDE monitoring screen;
2. The collected data appeared on the Raspberry Pi's screen after running the Python Code
3. In the log file:

- Each sensor data collection session will be recorded;
- The start time, stop time and duration of each session will be recorded;
- The avg data after 3 hours have been recorded to APRS output file, as well as the data trend, and anomaly (if it has)

4. In the output file:

- Each row in each session is a complete APRS data frame (containing the mentioned avg data, data trend…) which is formatted in the Complete Weather Data report format.