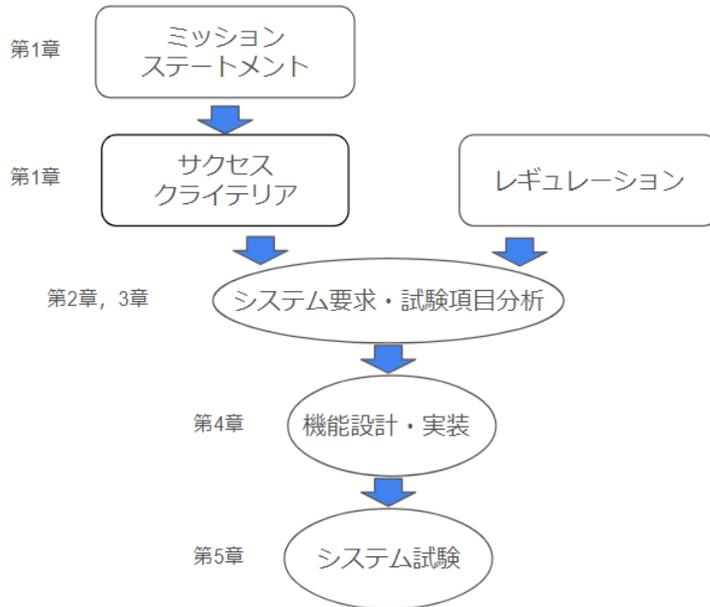


ARLISS2024大会報告書

提出日:2024年10月19日



開発審査書構成_ARLISS2024

- 第1章 ミッション定義
 - 第1.1節 ミッションステートメント
 - 第1.2節 ミッション内容
 - 第1.3節 サクセスクライテリア
- 第2章 システム要求
 - 第2.1節 レギュレーションを満たすためのシステム要求
 - 第2.2節 ミッションを達成するためのシステム要求
- 第3章 システム試験項目の設定
 - 第3.1節 レギュレーションを満たすためのシステム試験項目
 - 第3.2節 ミッションを達成するためのシステム試験項目
- 第4章 システム仕様
 - 第4.1節 機体概観
 - 第4.2節 機体機構
 - 第4.3節 搭載機器
- 第5章 システム試験
 - 第5.1節 レギュレーションを満たすためのシステム試験
 - 第5.2節 ミッションを達成するためのシステム試験
- 第6章 工程管理
- 第7章 責任教員による確認

チーム情報

CanSatチーム名	東京大学 Flying Cactus
CanSatチーム代表者情報	原島寛之 09090686494
UNISEC団体名	東京大学中須賀船瀬五十里研究室
UNISEC団体学生代表	中島彬
責任教員	中須賀真一 nakasuka@space.t.u-tokyo.ac.jp,03-5841-6590
CanSatクラス	Open Class

メンバー

役割	氏名(学年)
PM/機構	原島寛之(学部3年)
サブPM/制御	熊谷洗希(学部3年)

機構班長	生田篤史(学部3年)
機構/会計	栗田真希(学部3年)
機構	高木健人(学部3年)
電装班長	大屋孝輔(学部3年)
電装/会計	伊井和尊(学部3年)
電装	杉本慎之助(学部3年)
制御班長	白倉達啓(学部3年)
制御	白坂翠萌(学部3年)
制御	田村紘一(学部3年)

CanSatの製作目的・大会参加理由

実際にものづくりに取り組む機会の少ない弊学科の学生にとって貴重な実践の機会として、宇宙開発・航空機開発に近い開発に適切な難易度で取り組めると考えたため参加した。一連の開発を通して、工学的な知識技能やマネジメント能力を高めること、振動や荷重条件として実際の衛星状況に近い条件で実験を行うことで、実際の宇宙開発の一端に触れることを目的としている。

目次

第1章 ミッション定義	3
第1.1節 ミッションステートメント	3
第1.2節 ミッション内容	3
第1.3節 サクセスクライテリア	3
第2章 システム要求	3
第2.1節 レギュレーションを満たすためのシステム要求	3
第2.2節 ミッションを達成するためのシステム要求	4
第3章 システム試験項目の設定	5
第3.1節 レギュレーションを満たすためのシステム試験項目	5
第3.2節 ミッションを達成するためのシステム試験項目	5
第4章 システム仕様	6

第4.1節 機体概観	6
第4.2節 機体機構	6
第4.3節 搭載機器	6
第5章 システム試験	6
第5.1節 レギュレーションを満たすためのシステム試験	6
第5.2節 ミッションを達成するためのシステム試験	7
第6章 工程管理	7
第7章 責任教員による確認	7

第1章 ミッション定義

第1.1節 ミッションステートメント

フライバックによるゴールへの0mアプローチ

JAXAの月面探査機「SLIM」が月面に降り立ったことは記憶に新しい。「降りたいところに降りる」というキャッチフレーズは今後の宇宙探査機において非常に重要な指標になるであろう。このことは人工衛星を模した今回のcansat大会に於いて、まず念頭に置かなければいけないことだと考えた。東京大学チームは過去にフライバック0mを達成したことがないため、我々はマルチコプターによる史上初のフライバック0m達成を目指す。画像処理技術を用いることで、正確に0mで着陸することができると思った。過去の実績から大きな課題としては着陸後の再離陸の難しさ(例:着地後パラシュートの処理,着地姿勢)が挙げられる。我々は本ミッションに於いてこの技術的解決も目標としている。

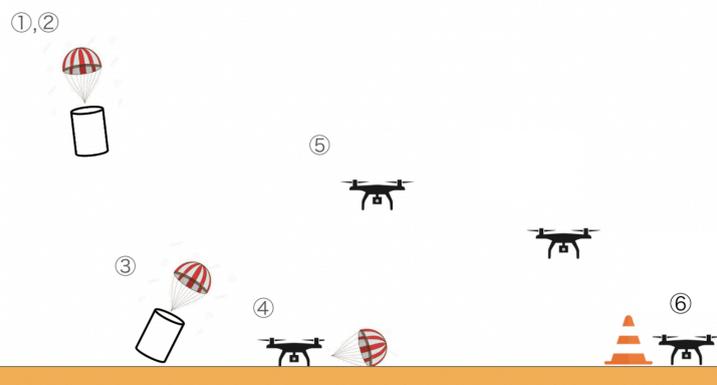
第1.2節 ミッション内容

ミッションアイテムは大きく分けて以下のようになると想定している

1. キャリアへの収納まで
2. キャリアから放出
3. パラシュート展開・着地
4. 離陸
5. 飛行
6. 着陸

各状態でのミッションアイテムをいかに詳しくまとめる。

1. キャリアへの収納まで
 - 1.1 主電源ON
 - 1.2 センサキャリブレーション
 - 1.3 GPSの通信開始
 - 1.4 ロジックの電源ON
 - 1.5 通信off
 - 1.6 キャリアへの収納
 - 1.7 ロケットへの搭載
2. キャリアからの放出
 - 2.1 光センサの値と気圧センサの値からキャリアから放出されたことを判定
 - 2.2 放出シーケンス終了を地上局へ送信
 - 2.3 地上局へのGPS情報のダウンリンク開始
3. パラシュート展開・着地
 - 3.1 パラシュートが風により展開する
 - 3.2 終端速度にて着地
 - 3.3 気圧センサ・ジャイロセンサの値から機体の静止を推定し,着地を判定
 - 3.4 着地シーケンス終了を地上局へ送信
4. 離陸
 - 4.1 機体が離陸体制に移行する.
 - 4.2 Take-off モードの起動,機体が離陸する
 - 4.3 離陸シーケンス終了を地上局へ送信
5. 飛行
 - 5.1 GPSを元にゴール地点を目指して飛行開始する
 - 5.2 GPS上でゴール地点に到着したら高度を10m程度にあげてホバリングする



6. 着陸

- 6.1 カメラで目標物を認識する
- 6.2 目標物に接近しつつ,高度を落とす
- 6.3 目標物近くに着陸する
- 6.4 着陸シーケンス終了を地上局に送信

第1.3節 サクセスクライテリア

画像航法を除く各シーケンスを,個別では実現可能なことをMinimum successとした.

Full successは0mゴール達成と, 0mゴールに向けた最終的な位置調整 (画像航法) とした.

以下の表にサクセスクライテリアと, その検証方法を示した.

サクセスクライテリア	内容	検証方法
Minimum success	放出・落下判定ができる	目視, 制御ログによる確認
	パラシュートが展開する	目視による
	機体が落下衝撃に耐えられる	目視による, ミッション終了後に機体を確認
	機体がパラシュートに干渉することなく飛び立つ	目視による
	飛行・制御ログが残せる	制御ログの確認
	機体が自律飛行できる	目視による
	Full success	0mゴール達成
機体とゴールの距離, 位置関係を測定できる		制御・飛行ログと目視による
Extra success	機体が損傷しないでミッションを終了する	目視で確認 (ミッション前後で写真を記録)

第2章 システム要求

第2.1節 レギュレーションを満たすためのシステム要求

番号	レギュレーションを満たすためのシステム要求
R1	CanSatは規定のサイズと質量以内でなければならない.
R2	CanSatはロケットから放出後, 位置が特定できなければならない.
R3	CanSatは規定の終端速度の範囲内で降下しなければならない.
R4	CanSatは打上げ時, パラシュート開傘時の衝撃を受けた後もその機能を維持しなければならない.
R5	CanSatはロケット搭載時に無線の送波を停止しなければならない.

R6	CanSatに搭載する全ての無線機は要求に応じて周波数の変更を行えなければならない。
R7	CanSatはロケットに搭載後メンテナンスなしにミッションを維持しなければならない。
R8	CanSatはロケットに損傷を負わせる可能性のある機構や物質を搭載してはいけない。
R9	CanSatは制御されることなく着地しなければならない。

番号	レギュレーションを満たすためのシステム要求 (Comeback Competition)
CR1	CanSatは完全に自律的に制御されなければならない。
CR2	チームはレギュレーションで指定されたコントロールレコードを提出しなければならない。

番号	レギュレーションを満たすためのシステム要求 (UAS)
UR1	CanSatはFAA認証を取得しなければならない。
UR2	CanSatは対地高度400ftを超えて飛行してはならない。

第2.2節 ミッションを達成するためのシステム要求

各シーケンスを成功させるために必要な実験を洗い出した。これを全てクリアすることで,Minimum successをクリアできることになる。

番号	ミッションを達成するためのシステム要求
M1	パラシュートが開傘することが試験で確認できている
M2	パラシュートの開傘衝撃に耐えられることが試験で確認できている
M3	着地時の衝撃によって本体が構造・回路共に損傷しないことが確認できている
M4	着地後, 離陸姿勢に移行できる
M5	着地後, 自律飛行モードに移行できる
M6	GPSを用いた飛行ができる
M7	航続距離が5km以上ある
M8	地上局-機体間通信ができる

M9	ゴール付近でカメラによる撮影ができる
M10	画像からコーン（ゴール）を認識できる
M11	画像からコーンと機体の位置関係を推定できる
M12	画像航法によってゴール0mでミッションを終了することが試験等により確認できている

第3章 システム試験項目の設定

第3.1節 レギュレーションを満たすためのシステム試験項目

番号	試験項目名	対応するシステム要求番号	実施予定日
RV1	質量試験	R1	7/9
RV2	機体収納試験	R1,R8	7/5
RV3	GPS取得試験	R2	7/19
RV4	パラシュート投下試験	R3,R4,R9	7/7
RV5	準静的荷重,振動,分離衝撃試験	R4	7/3
RV6	無線機接続・切断試験	R5	7/10
RV7	周波数変更試験	R6	7/10
RV8	自律飛行試験	R7,R9	6/28

番号	試験項目名	対応するシステム要求番号	実施予定日
CRV1	自律飛行試験	CR1	6/28
CRV2	End to End試験	CR1	7/10
CRV3	制御ログ取得試験	CR2	7/10

番号	試験項目名	対応するシステム要求番号	実施予定日
FRV1	FAAへの機体登録	UR1	8/2
FRV2	高度制御試験	UR1,2	7/10

第3.2節 ミッションを達成するためのシステム試験項目

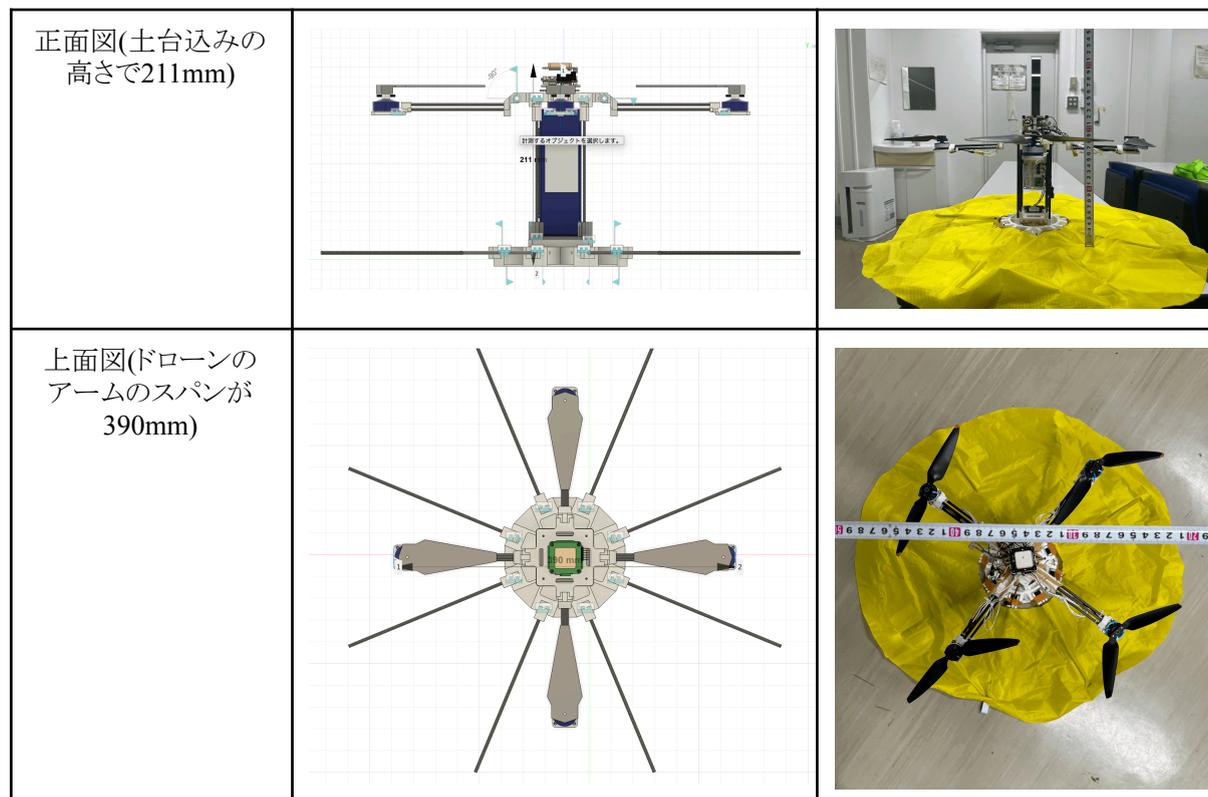
番号	試験項目名	対応するシステム要求番号	実施予定日
MV1	機体放出試験	M1,M2	7/11
MV2	着地衝撃試験	M3	7/10
MV3	機体起き上がり試験	M4	7/10
MV4	自律飛行試験	M5,M6,M8	6/28
MV5	航続距離試験	M7	8/1
MV5	画像認識試験	M10,M11	7/9
MV6	画像航法試験	M9,M10,M11,M12	7/9
CRV2	End to End試験	M1~M12	7/10

第4章 システム仕様

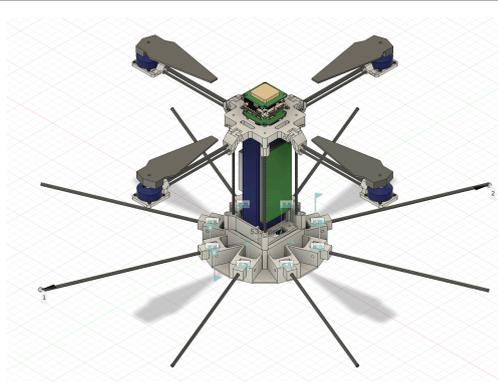
第4.1節 機体概観

- 機体外観(CAD図,写真)
寸法は設計上のものを示す。

展開時

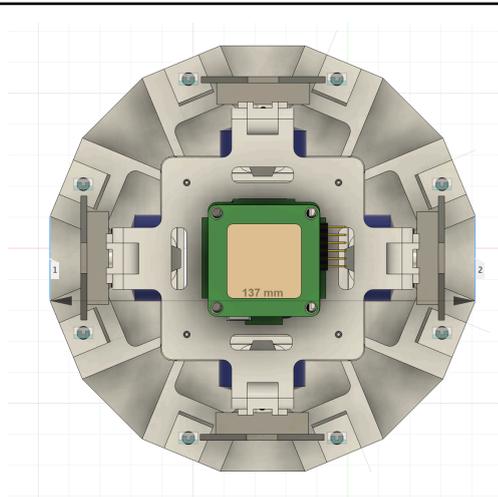


鳥瞰図(土台部分の
アームのスペンが
533mm)

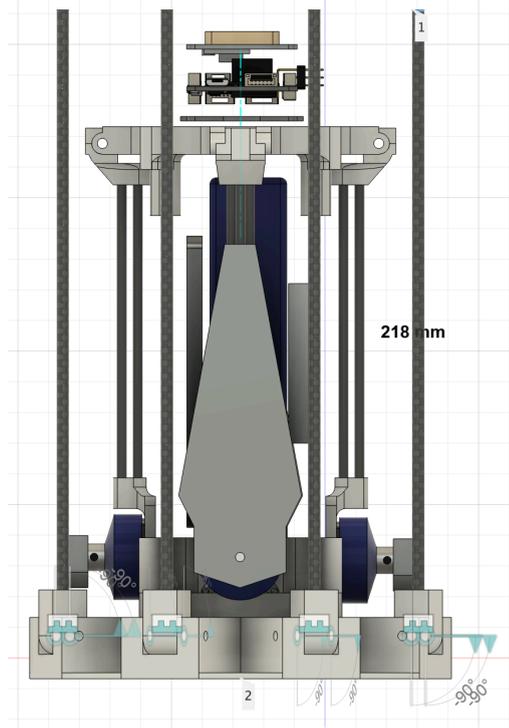


収納時(ケースに収納した状態)

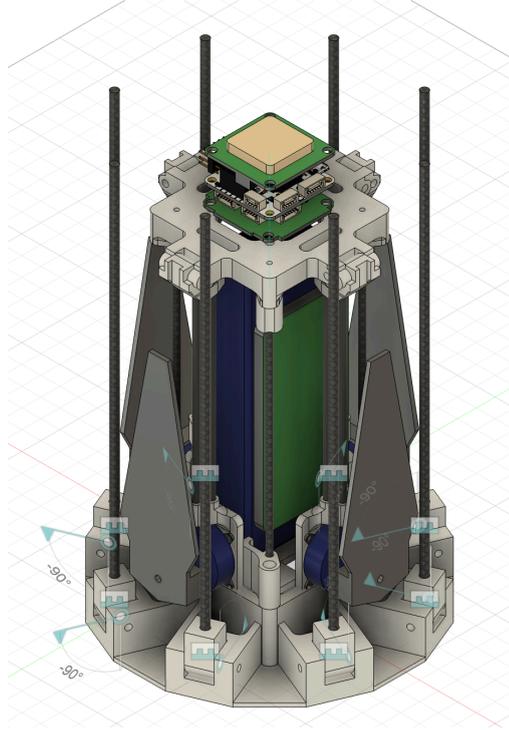
上面図(土台部分の
直径が137mm)



側面図(収納時の高さが218mm)



鳥瞰図



- 質量・寸法など
現状では設計が確定していない部分があったり,材質の変更を試みている部分があるため,今後いくつかの変化が予想される.

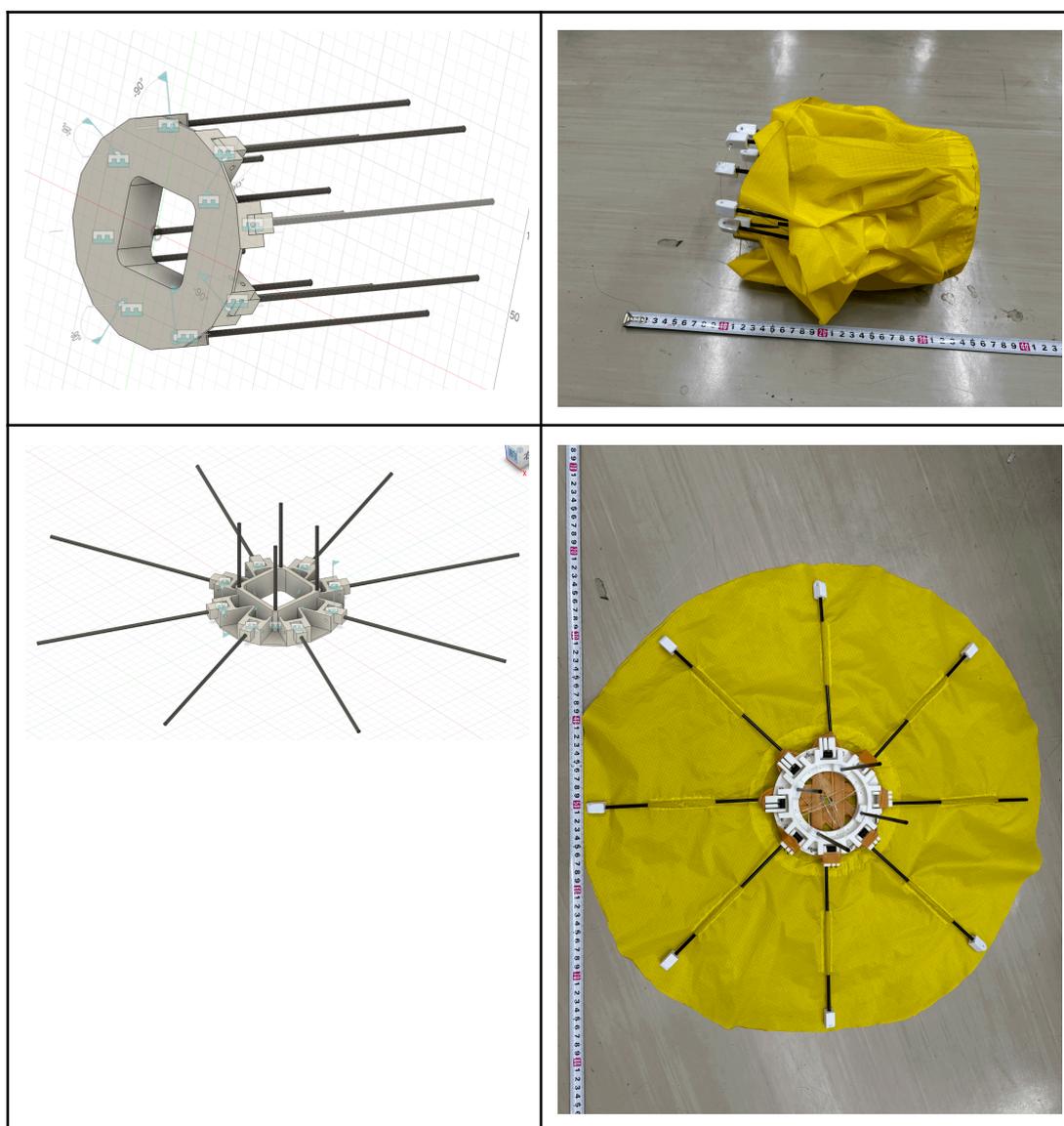
状態	収納時	飛行時
----	-----	-----

第4.2節 機体機構

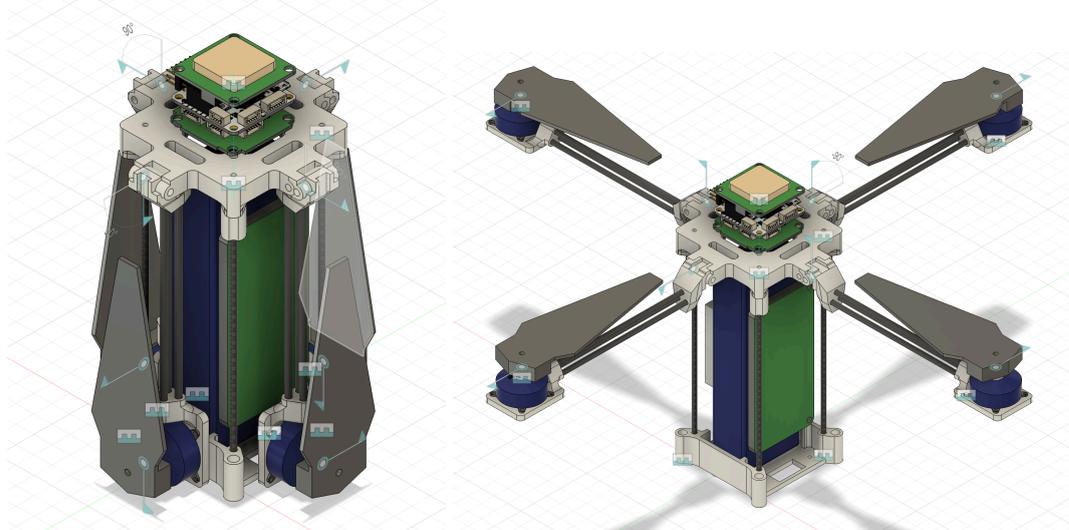
- 本機体はパラシュートが取り付けられ、着陸後の起き上がりを行う土台部分とその上に乗るドローン部分に分けられる。土台部分とドローン部分はともに収納時は足とアームをそれぞれテグスにより拘束され格納した状態であるが、ニクロム線によりテグスを溶断し、ゴム(現在バネに変更することを検討している)の張力によって展開される。

□ 土台部分

8本の足の根本にゴムをかけ、足を曲げて格納可能な状態にすると強い張力がかかるようになっていいる。上方にカーボンパイプの支柱が伸びており、この支柱をドローンの穴に通すことで発射台としての役割を果たす。[起き上がり試験動画](#)



- ドローン部分
土台部分と同様に収納時はテグスによりアームが拘束され溶断されるとかけられたゴムの張力により展開される。プロペラとして水平方向に折りたたむことができるプロペラを採用しており、モーターを回すと遠心力により自動的に羽が展開され離陸可能になる。



詳しい搭載機器については次節で紹介することにする。

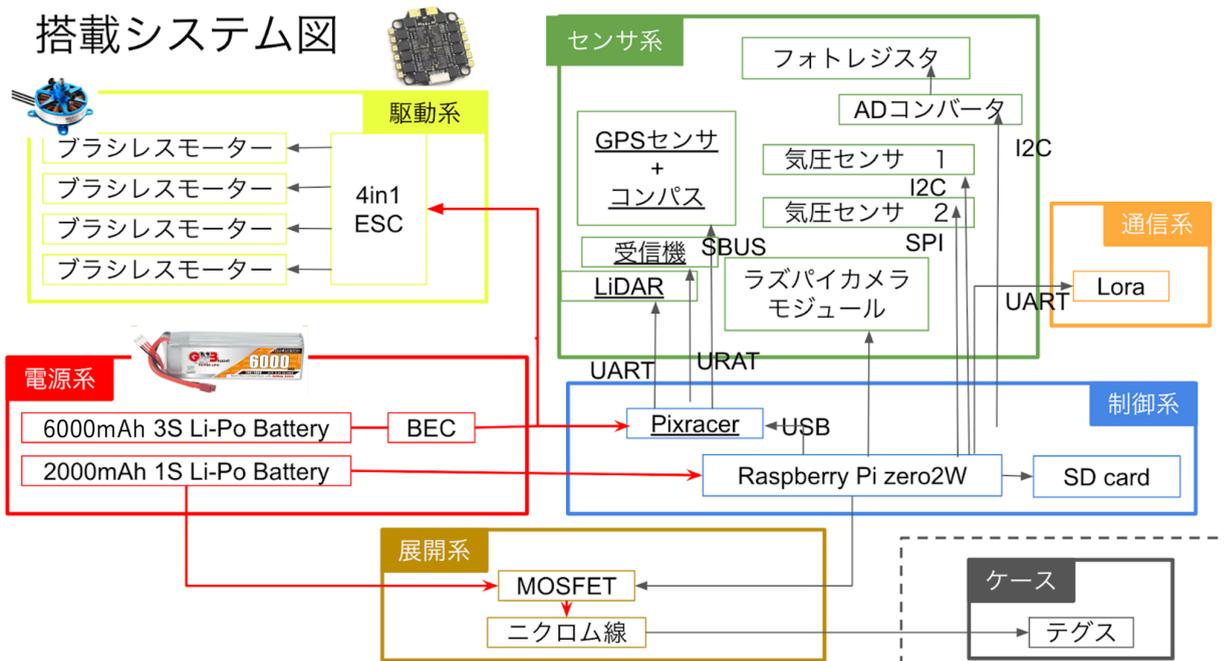
第4.3節 搭載機器

- 搭載機器は以下の通りである。

機器名	概要/選定理由	データシート
Raspberry Pi zero2W	マイクロコンピュータ/ 小型で、市販のマイクロコンピュータの中でシングルコアでなくて、CPUの性能がよかったため採用。画像処理や飛行時の非同期処理をすることを見越した。	https://datasheets.raspberrypi.com/rpizero2/raspberry-pi-zero-2-w-product-brief.pdf
pixracer	フライトコントローラ/ 機体の形状に合わせて、従来の横長のものではなく、正方形のフライトコントローラを採用した。	https://docs.px4.io/main/en/light_controller/pixracer.html
Raspberry Pi カメラモジュール V3	カメラモジュール/ Raspberry Piように作られたものである。	https://www.raspberrypi.com/documentation/accessories/camera.html
IRLU3410PBF	NchパワーMOSFET/ 120Aまでの大電流に対応している。ニクロム線によるテグス溶断を行うために、電流を多く流せるものを選定した。	https://akizukidenshi.com/goods/IRLU3410PBF_IRLR3410PBF.pdf
MPL115A2,SCP1000	気圧センサ/ I2C通信のものとSPI通信のもの2種類を併用することで冗長性をとる。できるだけ細かく値が取れるものを選定した。	https://www.nxp.jp/docs/en/data-sheet/MPL115A2.pdf

MCP3425	16-bit アナログ-デジタルコンバータ/ i2c通信で繋がり2本の配線で通信 できる利便性より選定した.	https://ww1.microchip.com/downloads/en/DeviceDoc/22072b.pdf
TPS61230A	5V/6A 高効率昇圧型コンバータ/ 大電流を供給可能なものが 必要であるため定格2.5Aを 詠うこのモジュールに決定した.	https://strawberry-linux.com/pub/slvscz5a.pdf
TLM922S-P01A	Loraモジュール/ 日本,アメリカ両方で使用 することを考慮し,技適認 証とFCC認証両方に対応 しているものを使用す る.両方に対応する物 の中で,データシート 上で通信距離が10km 程度であったこのモ ジュールに決めた.	https://akizukidenshi.com/goods/affix/tlm922s.pdf
MI527/MI5527	光センサ/ 昨年動作している実績 や,コスト面,先輩から の引き継ぎによる入手 性からこのcdsセルに 決定した.	https://akizukidenshi.com/goods/affix/MI5_series_jp.pdf
TF-mini-S	Lidar/ 短い距離での精度が 高い.(0.1~6mの範囲 にあるときに±6cm)	https://www.mouser.jp/datasheet/2/1099/Benewake_10152020_TFmini_S-1954051.pdf
NEO-M8	GPSモジュール/ Pixracerに対して標準 のGPSである.	https://content.u-blox.com/sites/default/files/NEO-M8-FW3_DataSheet_UBX-15031086.pdf
GNB60003S70AHV	Lipohvバッテリー/ 航続距離を伸ばすため に,容量の多いものを 採用.重量との兼ね合 いもあり,この大きさ のものに落ち着いた. このバッテリーの使 用によって,ecal という航続距離計算 サイトでは11kmの航 続距離を実現できる 見込みである.(詳し くは後述の試験MV5 にて記載)	GNB-GAONENG-6000mah-3S-11-4V_1601113104756.html

- システム図は以下の通りである



第5章 システム試験

第5.1節 レギュレーションを満たすためのシステム試験

RV1, MV1 質量試験

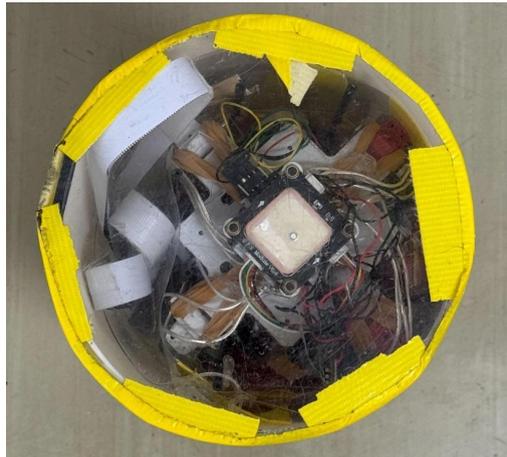
- 目的
 - CanSatがキャリア格納時にレギュレーションを満たすことを確認する。
- 試験内容
 - CanSat本体とパラシュートの合計質量を計測する。
- 試験結果
 - 以下のように、キャリアに格納時に載せる全ての部品の合計重量は1046gとなった。秤の誤差は±3gである。



RV2 機体収納試験

- 目的
 - CanSatがキャリア格納時にレギュレーションを満たすことを確認する。
- 試験内容
 - CanSat本体とパラシュートの両方を同時にキャリアに収納する。
- 試験結果
 - CanSat本体とパラシュートの両方ともキャリアに収まることが確認できた。

- また,CanSatの収納前後でキャリアは損傷していない.
以下写真



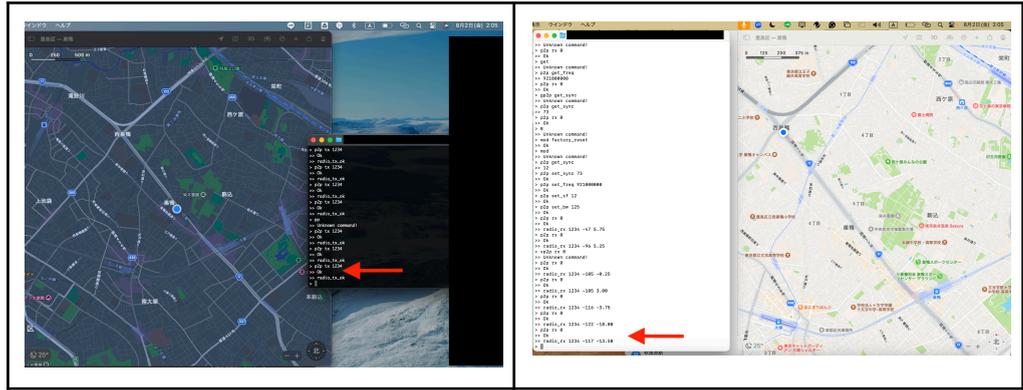
RV3 GPS取得試験

- 目的
 - CanSatがGPSを取得できることを確認する
- 試験内容
 - GPSから得られた位置情報を地上局に送信する
 - 東京大学工学部7号館横でGPSを取得した.
 - Loraの性能テストとして,おおよそ1kmの区間での無線の送受信試験も行った
- 試験結果
 - GPSの値は,以下のように出力される.また,この情報はログとしてcsvファイルに残す.

```
(Cansat2024A) Cansat2024A2@rosberry1:~/Cansat2024A/test_code $ python3 GPS.py
/home/Cansat2024A2/Cansat2024A/log/2024-08-04_logger/000.csv already exists
/home/Cansat2024A2/Cansat2024A/log/2024-08-04_logger/001.csv already exists
/home/Cansat2024A2/Cansat2024A/log/2024-08-04_logger/002.csv already exists
2024-08-04 22:36:18.248, [INFO] Start getGPS.
2024-08-04 22:36:18.268, [INFO] Position: [latitude_deg: 35.715418299999996, longitude_deg: 139.760471, absolute_altitude_m: 35.487008326416016, relative_altitude_m: -0.012000000104308128]
2024-08-04 22:36:18.269, [INFO] In air: False
2024-08-04 22:36:18.291, [INFO] Gps info: GpsInfo: [num_satellites: 9, fix_type: FIX_3D]
2024-08-04 22:36:18.432, [INFO] Position: [latitude_deg: 35.715418199999995, longitude_deg: 139.76047119999998, absolute_altitude_m: 35.47200012207031, relative_altitude_m: -0.027000000700354576]
2024-08-04 22:36:18.487, [INFO] Gps info: GpsInfo: [num_satellites: 9, fix_type: FIX_3D]
2024-08-04 22:36:18.532, [INFO] Position: [latitude_deg: 35.715418199999995, longitude_deg: 139.7604713, absolute_altitude_m: 35.4630012512207, relative_altitude_m: -0.036000002176569534]
2024-08-04 22:36:18.633, [INFO] Position: [latitude_deg: 35.715418199999995, longitude_deg: 139.7604713, absolute_altitude_m: 35.45600183105469, relative_altitude_m: -0.044000000348687172]
2024-08-04 22:36:18.687, [INFO] Gps info: GpsInfo: [num_satellites: 9, fix_type: FIX_3D]
2024-08-04 22:36:18.731, [INFO] Position: [latitude_deg: 35.715418199999995, longitude_deg: 139.7604714, absolute_altitude_m: 35.448001861572266, relative_altitude_m: -0.05100000277161598]
2024-08-04 22:36:18.765, [INFO] In air: False
2024-08-04 22:36:18.831, [INFO] Position: [latitude_deg: 35.715418199999995, longitude_deg: 139.7604715, absolute_altitude_m: 35.439002990722656, relative_altitude_m: -0.06000000238418579]
2024-08-04 22:36:18.886, [INFO] Gps info: GpsInfo: [num_satellites: 9, fix_type: FIX_3D]
2024-08-04 22:36:18.931, [INFO] Position: [latitude_deg: 35.715418199999995, longitude_deg: 139.7604716, absolute_altitude_m: 35.43100357055664, relative_altitude_m: -0.06800000369548798]
2024-08-04 22:36:11.31, [INFO] Position: [latitude_deg: 35.715418199999995, longitude_deg: 139.7604716, absolute_altitude_m: 35.42399978637695, relative_altitude_m: -0.07600000298823224]
2024-08-04 22:36:11.92, [INFO] Gps info: GpsInfo: [num_satellites: 9, fix_type: FIX_3D]
2024-08-04 22:36:11.131, [INFO] Position: [latitude_deg: 35.715418199999995, longitude_deg: 139.76047169999998, absolute_altitude_m: 35.41699981689453, relative_altitude_m: -0.0820000022649765]
2024-08-04 22:36:11.266, [INFO] In air: False
2024-08-04 22:36:11.291, [INFO] Gps info: GpsInfo: [num_satellites: 9, fix_type: FIX_3D]
2024-08-04 22:36:11.337, [INFO] Position: [latitude_deg: 35.7154181, longitude_deg: 139.7604718, absolute_altitude_m: 35.402000427246094, relative_altitude_m: -0.09700000286102295]
2024-08-04 22:36:11.437, [INFO] Position: [latitude_deg: 35.7154181, longitude_deg: 139.7604719, absolute_altitude_m: 35.39500045776367, relative_altitude_m: -0.10400000214576721]
2024-08-04 22:36:11.491, [INFO] Gps info: GpsInfo: [num_satellites: 9, fix_type: FIX_3D]
2024-08-04 22:36:11.537, [INFO] Position: [latitude_deg: 35.715418, longitude_deg: 139.7604719, absolute_altitude_m: 35.388999938964844, relative_altitude_m: -0.110000000685453415]
2024-08-04 22:36:11.631, [INFO] Position: [latitude_deg: 35.715418, longitude_deg: 139.7604719, absolute_altitude_m: 35.384002685546875, relative_altitude_m: -0.11500000208616257]
2024-08-04 22:36:11.692, [INFO] Gps info: GpsInfo: [num_satellites: 9, fix_type: FIX_3D]
2024-08-04 22:36:11.731, [INFO] Position: [latitude_deg: 35.7154179, longitude_deg: 139.760472, absolute_altitude_m: 35.38199996948242, relative_altitude_m: -0.117000000613927841]
2024-08-04 22:36:11.766, [INFO] In air: False
2024-08-04 22:36:11.831, [INFO] Position: [latitude_deg: 35.7154179, longitude_deg: 139.760472, absolute_altitude_m: 35.38100051879883, relative_altitude_m: -0.11800000016583633]
2024-08-04 22:36:11.891, [INFO] Gps info: GpsInfo: [num_satellites: 9, fix_type: FIX_3D]
2024-08-04 22:36:11.934, [INFO] Position: [latitude_deg: 35.7154179, longitude_deg: 139.7604721, absolute_altitude_m: 35.38199996948242, relative_altitude_m: -0.117000000613927841]
2024-08-04 22:36:12.31, [INFO] Position: [latitude_deg: 35.7154178, longitude_deg: 139.76047219999998, absolute_altitude_m: 35.38300032486328, relative_altitude_m: -0.11600000041272849]
2024-08-04 22:36:12.91, [INFO] Gps info: GpsInfo: [num_satellites: 9, fix_type: FIX_3D]
2024-08-04 22:36:12.131, [INFO] Position: [latitude_deg: 35.7154178, longitude_deg: 139.76047219999998, absolute_altitude_m: 35.38500213623847, relative_altitude_m: -0.11400000751818524]
2024-08-04 22:36:12.231, [INFO] Position: [latitude_deg: 35.7154178, longitude_deg: 139.7604723, absolute_altitude_m: 35.38500213623847, relative_altitude_m: -0.11400000751818524]
2024-08-04 22:36:12.264, [INFO] In air: False
```

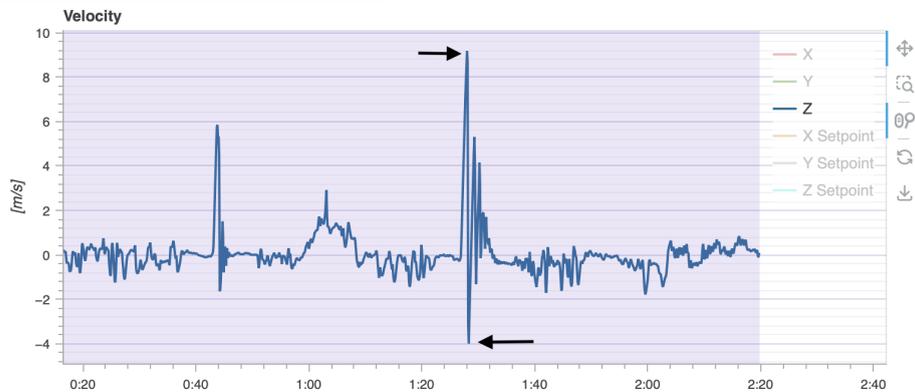
- 以下の表のように1kmからの送受信に成功している.
送信局,受信局ともに青点でその時の位置を示している.

送信局	受信局
-----	-----



RV4 パラシュート投下試験

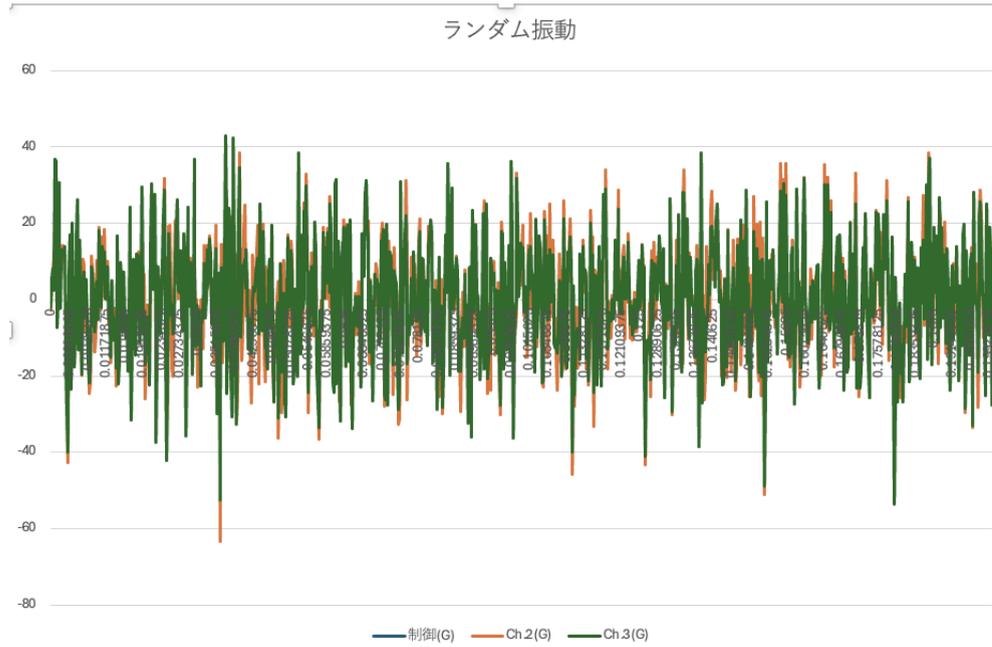
- 目的
 - パラシュートが開傘する衝撃でCanSatが損傷せずに機能し、また、パラシュートによって充分減速されることが確認できる
- 試験内容
 - パラシュートを高所から投下し、パラシュートの開傘衝撃に耐えることを目視で確認、また落下速度がレギュレーションの4~6m/sであることをフライトコントローラで速度を確認する
 - 50Gの開傘衝撃に耐えることを、落下中に急に減速させることで50Gを生み出し、確認する
- 試験結果
 - 減速は以下の動画のようにして確認された。また、その時の終端速度は4.2m/sになった。
[RV4パラシュート投下試験_減速.mov](#)
 - 開傘衝撃試験の結果は以下のようである。動画のように実験を行い50Gを再現した。載せていたセンサによると、12m/sの変化が0.2S以下で起きているので600m/s=61G以上の変化が起きているとわかった。(矢印で示しているところの値を用いる)
[RV4パラシュート投下試験.MOV](#)



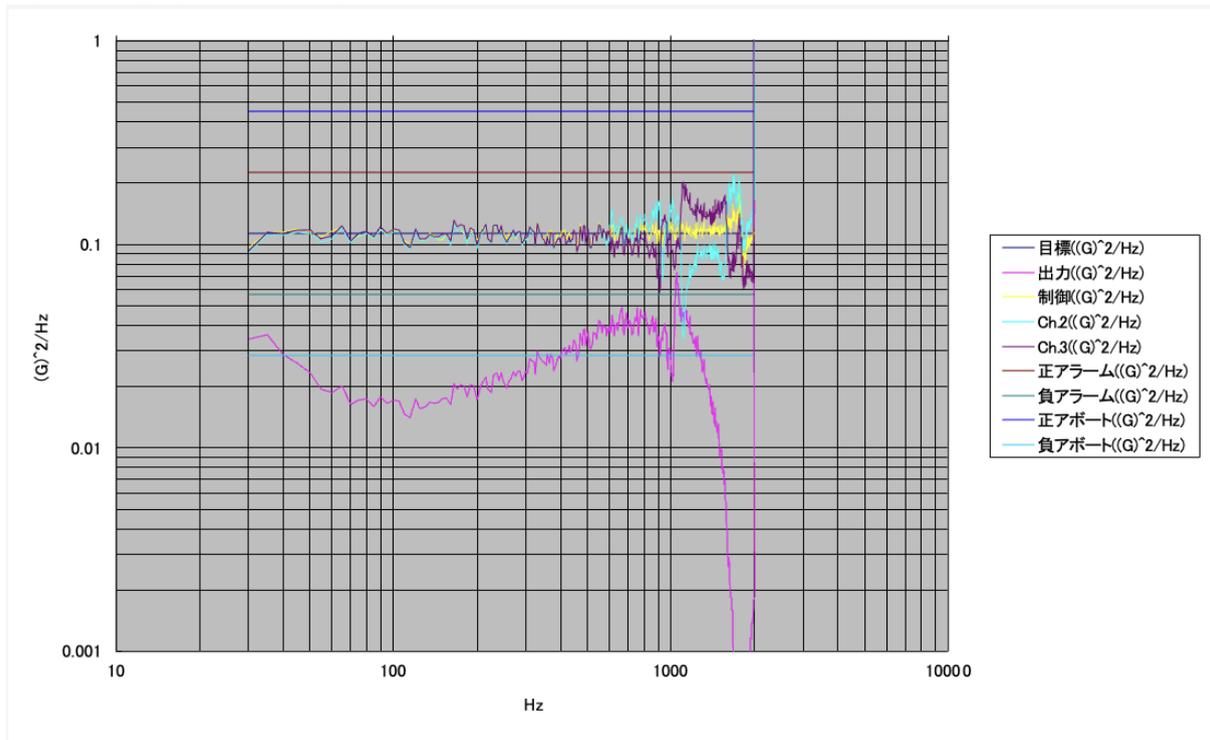
RV5 準静的荷重,振動,分離衝撃試験

- 目的
 - CanSatは打上げ時の衝撃を受けた後もその機能を維持することを確認する。
- 試験内容
 - 正弦波振動30Hz~2000Hzまで15G、または同等のランダム振動に機体が耐えられることを確認する。
 - ロケット打ち上げ時の鉛直方向にかかる10Gに10秒間CanSatが耐えることを確認する。
 - ロケットから分離するときに鉛直方向にかかる撃荷重40GにCanSatが耐えることを確認する。
- 試験結果
 - ランダム振動試験の結果、機体は構造・回路ともに損傷しなかった。加えた振動の様子は以下

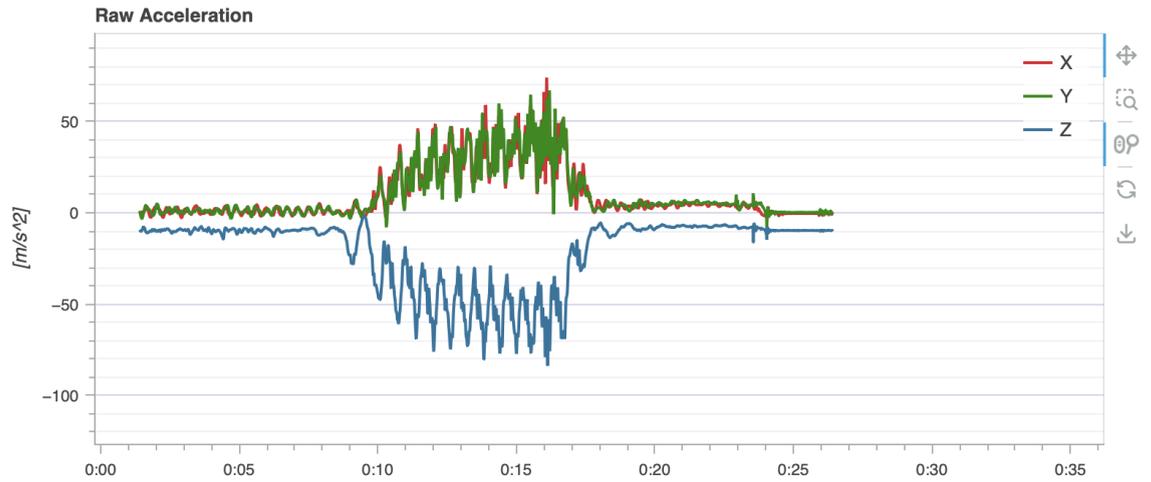
のグラフで示す。



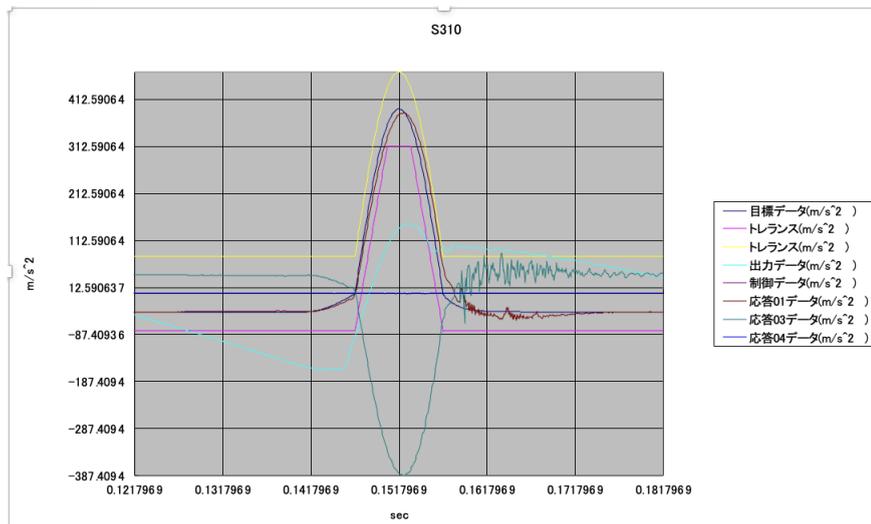
応答PSDは以下のグラフのようであった。



- 10Gの荷重を加えた結果,機体は構造・回路ともに損傷しなかった.加えた荷重の様子は以下のグラフで示す.荷重の加え方は,CanSatを円運動させることで,垂直方向と向心力の向きを一致させることで実現した.
実験動画:[10G10s.mp4](#)



- 衝撃荷重40Gを加えた結果,機体は構造・回路ともに損傷しなかった.加えた振動の様子は以下のグラフで示す.



RV6 無線機接続・切断試験

- 目的
 - 無線の送信/停止を自在に行えることを確認する
- 試験内容
 - CanSatに搭載されているRaspberry Piからの制御によって,LoRa(無線機)からの信号の送信/停止を行う.それを地上局(PC)で受信する.
- 試験結果
 - Loraの仕様上,以下のように送信を行わない限りは,電波の送波が行われない.
以下では「1234」という文字列を送っている.実際にはRV3で得られるようなGPS座標を送る.長い文字列も送ることができると確認済みである.

```

> p2p tx 1234
>> Ok
>> radio_tx_ok
> p2p tx 1234
>> Ok
>> radio_tx_ok
> p2p tx 1234
>> Ok
>> radio_tx_ok
>

```

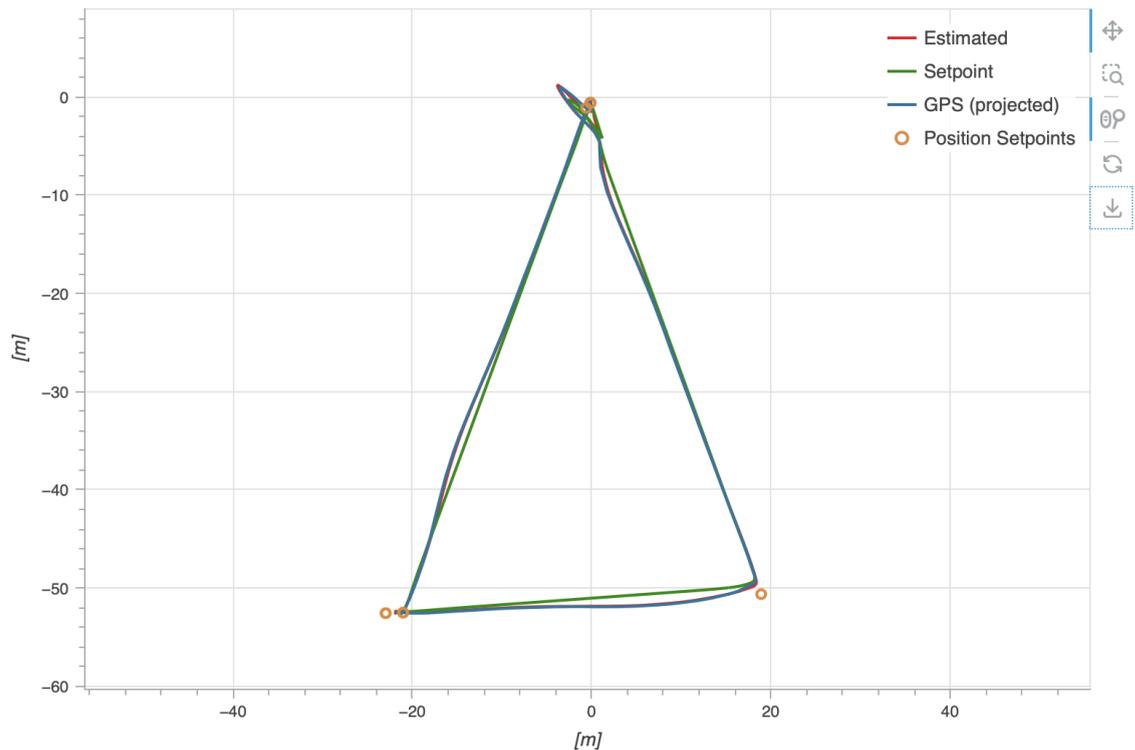
RV7 周波数変更試験

- 目的
 - 無線の周波数を変更することができることを確認する
- 試験内容
 - CanSatに搭載されているRaspberryPiからの制御によって,LoRa(無線機)が送波する周波数を変更する.
- 試験結果
 - 921MHzと922MHzで試験を行なった.同様の方法で,862MHz~932Mhzの範囲で周波数を変更可能である.
以下の写真のように設定し,送受信が可能である.

周波数	921MHz	922MHz
送信側	<pre> TLM9228 < > Unknown command! > Ok > p2p set_freq 921000000 > Ok > p2p tx 1234567890 > Ok > radio_tx_ok </pre>	<pre> TLM9228 < > p2p set_freq 922000000 > Ok > p2p tx 1234567890 > Ok > radio_tx_ok </pre>
受信側	<pre> TLM9228 < > p2p set_freq 922000000 > Ok > p2p tx 1234567890 > radio_tx_ok </pre>	<pre> TLM9228 < > Ok > p2p set_freq 922000000 > Ok > p2p rx 0 > Ok > radio_rx 1234567890 -93 5.75 </pre>

RV8,CRV1,MV4 自律飛行試験

- 目的
 - CanSatが事前に入力した飛行経路を元に自律的な飛行を行うことができることを確認する.
- 試験内容
 - 事前にCanSatに飛行経路を入力し,離陸から,飛行,着陸までを一連で行う.
- 試験結果
 - 事前に2点のwaypointを指定してCanSatを飛行させた.機体の外観は異なるが,中に載せているマイコンは一緒である.指示した飛行経路をCanSatが飛行していることを確認できた.試験動画のリンクは以下
[自律飛行試験.mov](#)
 - 飛行経路のログは以下のようにになっている.Positin Setpointsが事前に入力したwaypointになっており,そこを経由するような飛行ができたことが示された.



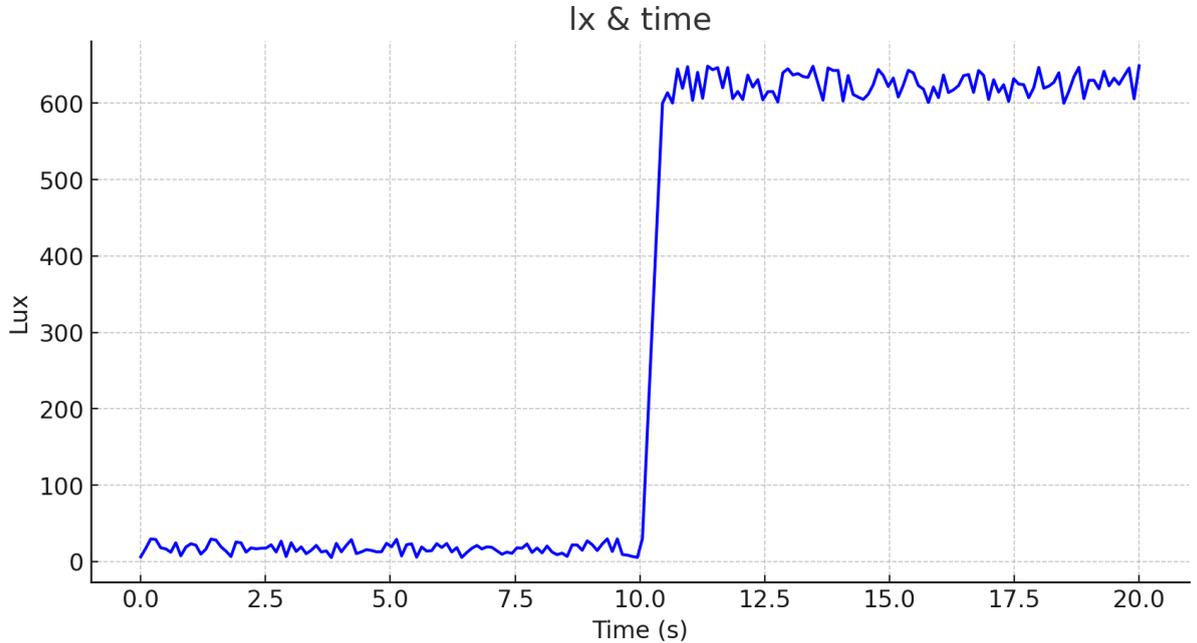
CRV2 End to End 試験

- 目的
 - CanSatが完全に自律した制御によってミッションを完遂することができるかどうか確認する。
- 試験内容
 - 機体が落下してから飛行開始されるまでの試験を行った。全てを通して行うのは、飛行試験場に行ける日程の都合上、8/8を予定している。
 - 能代宇宙イベントにおいて60mからのCansat投下を行い、画像航法を除くすべてのシーケンスを通して行った。
- 試験結果
 - 以下の動画のように、落下した状態からの起き上がり、飛行開始が問題なく行えている。ただし、起き上がりから飛行開始を一連の実験として行うことができてはいない。(飛行の試験を先に行ってしまう、テントの天井にぶつかった拍子で落下し機体のアームが折れてしまったため、折れたアームを除いて起き上がり試験を行っている。動画はわかりやすいようにシーケンス通りに進めている。)
 - [E2E.mp4](#)
 - 以下の動画に定常飛行/着陸の様子を示す
[縦型定常飛行.MOV](#)
 - 能代宇宙イベントにおいては、GPSのみでゴールから2.1m地点に着陸し、missionの終了が宣言された。
すべてのシーケンスがスムーズに行われているようであった。
 - 以下に動画を示す。
風が強く、かなり煽られているようだったが姿勢制御などもよく行えているようである。
 - [CRV2 E2E.mov](#)

CRV3 制御ログ取得試験

- 目的
 - 提出できるような制御ログを取得できることを確認する。
- 試験内容
 - 各シーケンスにおいてどのおよな処理が行われているかの制御情報を記録する。
- 試験結果

- 飛行ログ(例)は以下リンク先のようになる.
https://review.px4.io/plot_app?log=390f24fb-6668-4655-9043-9c3da6ca2243
- その他制御履歴について
 光センサの値と気圧センサの値を示す.
 光センサ(プロット)
 光センサの大きな変化を見て放出判定とする.



気圧センサ:

1秒ごとに気圧の値をとり、直近10回の値を移動平均を取る
 3回連続で移動平均の差が0.1以下の場合、Droneの着地判定を返す.

```
(Cansat2024A) Cansat2024A2@raspberrypi2:~/Cansat2024A/test_code $ python3 pressu
reasync.py
Current Average Pressure: 1003.32 hPa
Current Average Pressure: 1003.40 hPa
Current Average Pressure: 1003.32 hPa
Current Average Pressure: 1003.23 hPa
Drone Landed!
```

FRV1 FAAへの機体登録

- 目的
 - FAA認証を取得する
- 試験内容
 - FAAに機体を登録する
- 試験結果
 いかの写真のように機体登録は終了した.
 実際に機体に登録番号「FA3LF3YH97」も記載する予定である.

1. Safety Guidance 2. Inventory 3. Payment 4. Review & Pay 5. Confirmation

Congratulations!

Your registration number is **FA3LF3YH97**

Your registration certificate has been sent to the email address you provided in your registration.

You can also [print your certificate](#).

Be sure to have your registration certificate available when you fly your UAS.



- All drones must be marked on the outside with a registration number.
- You can mark all drones that you own and fly exclusively under [The Exception for Recreational Flyers](#) with the same registration number.
- Recreational flyers must operate in accordance with community-based aero-modeling club safety guidelines and within the programming of a nationwide community-based aero-modeling organization.

To fly under [The Exception for Recreational Flyers](#), you must:

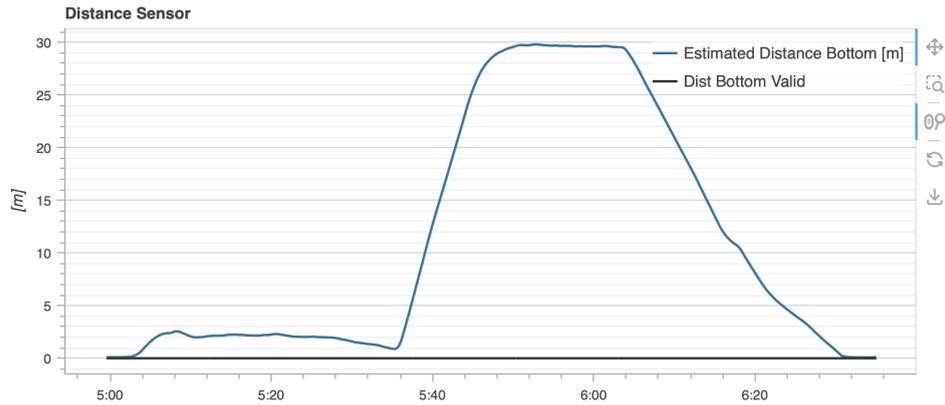
- Have a current registration
- Fly only for recreational purposes
- Follow the safety guidelines of a community based organization
- Keep your drone within your visual line of sight
- Give way and do not interfere with any manned aircraft
- Fly at or below 400' in controlled airspace and only with prior authorization
- Fly at or below 400' in uncontrolled airspace
- Comply with all airspace restrictions

FRV2 高度制御試験

- 目的
 - FAAルールに抵触しないように高度を制御できるか確認
- 試験内容
 - 事前にCanSatに飛行高度を指定し,その通りの挙動を示すか目視/ログで確認する.
- 試験結果
 - おおよそ,事前に指定した高度通り飛行した.設定した最高高度よりは高くないので,FAAルールに抵触することはない.以下高度の飛行ログである.setpointが事前に入力した高度,赤の線はGPSを元にした高度を表す.



また,機体に取り付けているLiDARから得られた高度のログも示す.



第5.2節 ミッションを達成するためのシステム試験

MV2 着地衝撃試験

- 目的
 - 終端速度4~6m/sでの落下衝撃によってCanSatが構造的,回路的に損傷しないことを確認する
- 試験内容
 - 高さ1.25mからCanSatを自由落下させる.
- 試験結果
 - 落下前後で機体に損傷はみられなかった.

[MV2落下1.25m.mov](#)



MV3 機体起き上がり試験

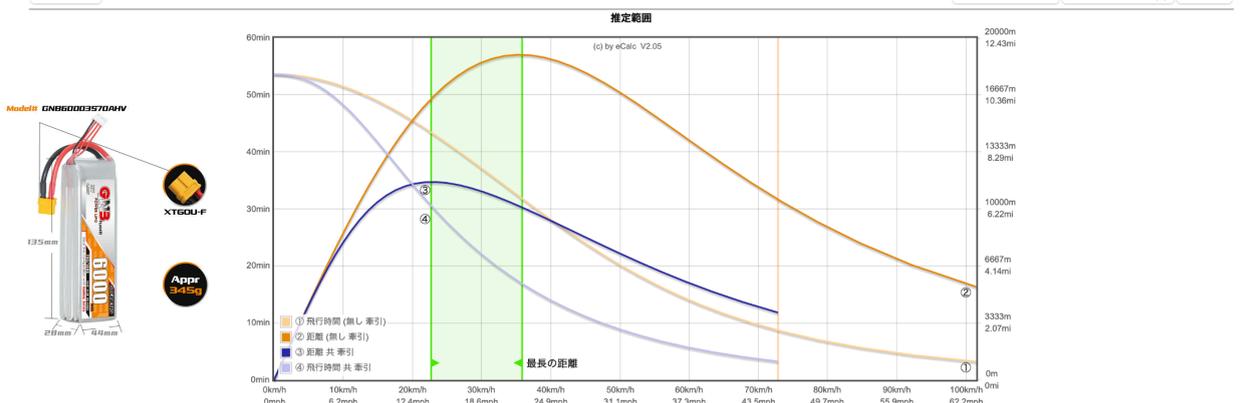
- 目的
 - パラシュートによる着地後,機体が飛行姿勢に移行することを確認する
- 試験内容

- パラシュートで落下した状態から、ニクロム線によるテグスの溶断によってCanSatが飛行姿勢に移行することを確認する。
- 試験結果
 - 以下の動画のように、CanSatが離陸姿勢に移行できることが確認できた。
[MV3機体起き上がり試験.mov](#)

MV5 航続距離試験

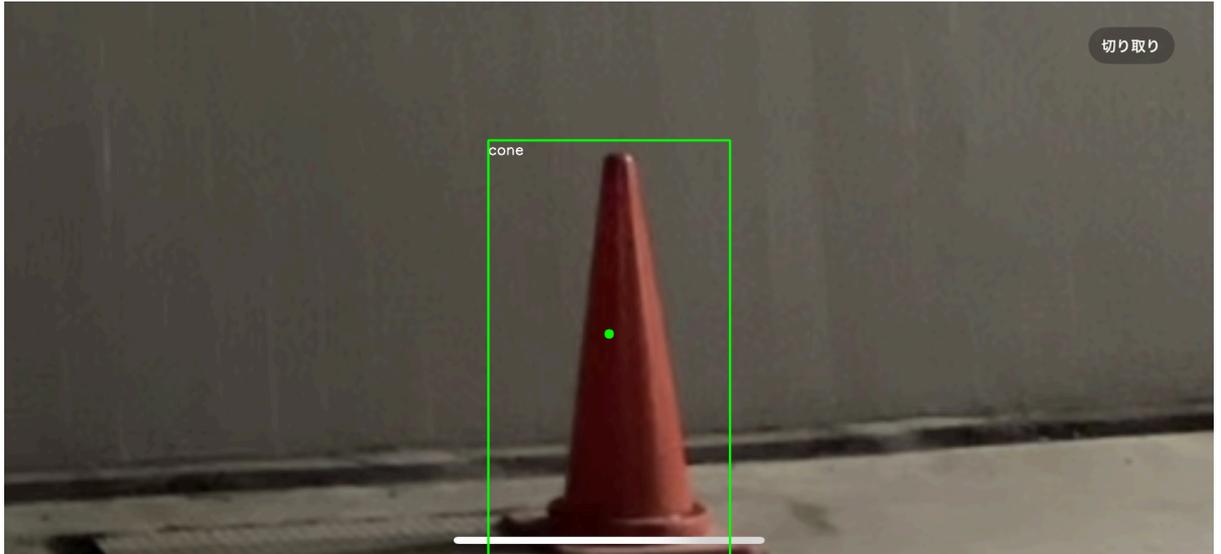
- 目的
 - どのくらいCanSatが流されるかわからないため、航続距離が少なくとも5km以上あることを確認する。
- 試験内容
 - 実際に飛行させることのできる場所がなく、折り返し飛行では本番の環境にも近くないこと、Lipoバッテリーを使い切ることが危険な可能性がある(途中落下の可能性)ため、ecalcという航続距離計算ソフトを用いた。
- 試験結果
 - 以下写真のように11km程度の飛行理論上可能である。

バッテリー	モーター@最大効率	モーター@最大出力	モーター@ホバリング	全駆動装置	マルチヘリコプター
放電容量: 20.84 C	電流: 9.21 A	電流: 31.25 A	電流: 1.43 A	駆動装置重量: 911 g	全備重量: 650 g
電圧: 11.22 V	電圧: 11.77 V	電圧: 11.03 V	電圧: 12.04 V	32.1 oz	追加のペイロード: 3488 g
定格電圧: 11.40 V	回転数: 17698 rpm	回転数: 13201 rpm	回転数: 4463 rpm	7.4 : 1	123 oz
エネルギー: 68.4 Wh	電圧: 108.4 V	電力: 344.8 W	推力(対数): 13 %	電流 @ ホバリング: 5.70 A	最大デルタ: 81 °
合計容量: 6000 mAh	発生推力: 93.4 W	発生推力: 250.6 W	推力(linear): 27 %	消費電力 @ ホバリング: 68.9 W	最高速度: 107 km/h
使用される容量: 5100 mAh	変換効率: 86.1 %	電源重量: 2122.1 W/kg	電力: 17.2 W	発生推力 @ ホバリング: 54.5 W	66.5 mph
最小飛行時間: 2.4 min		変換効率: 72.7 %	発生推力: 13.6 W	変換効率 @ ホバリング: 79.1 %	11572 m
混合飛行時間: 18.4 min		962.6 W/lb	電源重量: 106.0 W/kg	電流 @ 最大: 125.01 A	推奨範囲: 7.19 mi
ホバリング飛行時間: 53.6 min		変換効率: 83 °C	48.1 W/lb	消費電力 @ 最大: 1510.6 W	上昇の最大速度: 19.0 m/s
重量: 528 g		181 °F	変換効率: 79.4 %	発生推力 @ 最大: 1002.3 W	3740 ft/min
18.6 oz			推奨温度: 27 °C	変換効率 @ 最大: 66.3 %	15.34 dm²
		電力計測定	81 °F		237.77 in²
		電流: 125 A	比推力: 9.47 g/W		ローターの失敗:
		電圧: 11.22 V	0.33 oz/W		
		電力: 1402.5 W			



MV6 画像認識試験

- 目的
 - CanSatが画像処理によりコーンを認識できるか確認する
- 試験内容
 - CanSatにコーンの画像を学習させ、コーンが映った写真に対して、コーンの位置を特定させる
 - 試験結果
 - 以下の写真のように、画像内のコーンを認識し、四角で囲うことができた。また、長方形の中心位置に点を打つこともできた。



MV7 画像航法試験

- 目的
 - 認識したコーンの画像を元に,CanSatがコーンに近づいて行くことができるか確認する
- 試験内容
 - CanSatをコーンの近くから離陸させ,画像認識を開始,画像処理を行いながらコーンに近づくようにする
- 試験結果
 - 本審査書の提出に試験が間に合わなかった.

第6章 工程管理

7月後半に大学の試験期間が設定されているため,7/10までに一回目のEnd-to-End試験を行うことを目標にスケジュールを立てた.End-to-End試験の実施は制御班が行う予定である.テスト終了後改めてデバッグを行う.能代での共同実験に参加する都合上,8月末を目安に再びEnd-to-End試験を行い,それ以外の期間はデバッグに当てる予定である.

班	7/1	7/2	7/3	7/4	7/5	7/6	7/7	7/8	7/9	7/10	7/11-29
1機造	起き上がり機構印刷/試験										テスト期間
1機造	振動試験										
1機造	カーボン素材発注										
1機造			落下衝撃試験								
1機造			パラシュート投下試験								
1機造			開傘衝撃試験								
1機造					修正等						
1機造							本番機印刷/組み立て				
1機造									予備日		
1機造											
1機造											
2回路			ニクロム線溶断試験								
2回路				本番基板発注							
2回路						LoRaによるロスト対策試験					
2回路									予備日		
2回路											
3制御	コーディング										
3制御				シーケンス毎のテスト							
3制御						デバッグ					
3制御									End_to_End①		
3制御											
3制御											
3制御											
3制御											



第7章 大会結果報告

第7.1節 目的

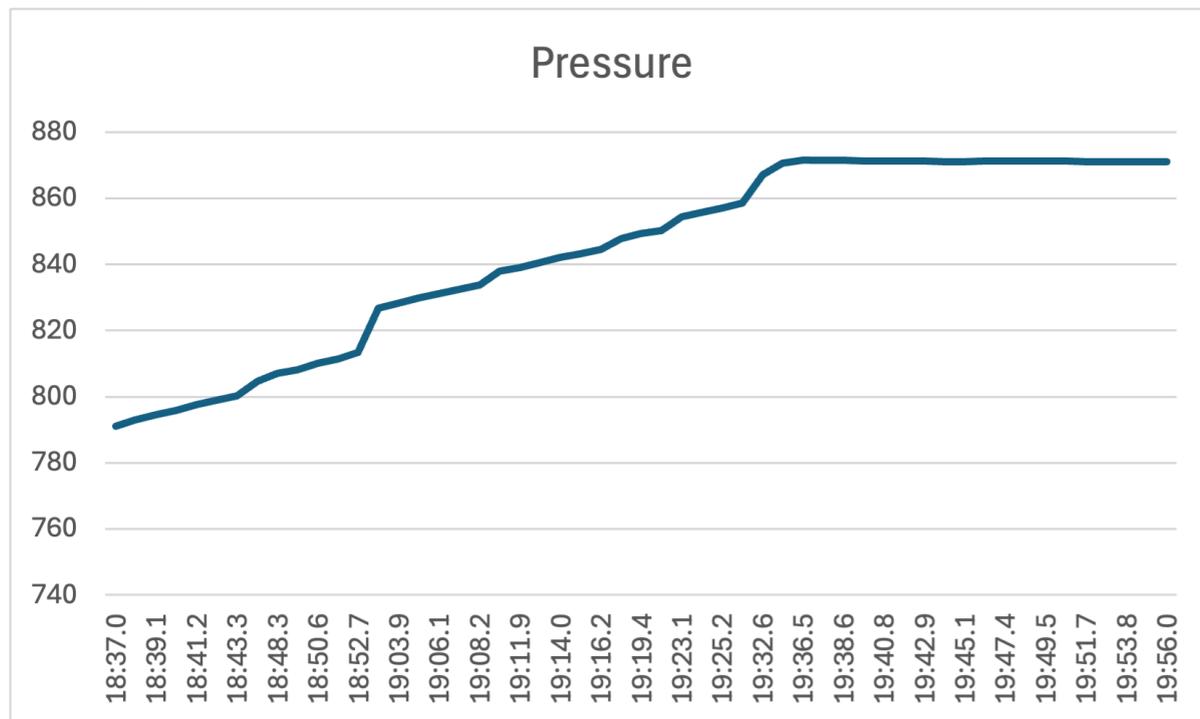
実際にものづくりに取り組む機会の少ない弊学科の学生にとって貴重な実践の機会として、宇宙開発・航空機開発に近い開発に適切な難易度で取り組めると考えたため参加した。一連の開発を通して、工学的な知識技能やマネジメント能力を高めること、振動や荷重条件として実際の衛星状況に近い条件で実験を行うことで、実際の宇宙開発の一端に触れることを目的としている。

第7.2節 結果

・一度目の投下試験

着陸後、展開機構が想定通り作動せず離陸姿勢に体制を帰ることができなかった。風によってCanSatが横転してしまったタイミングで棄権となった。

落下時の気圧ログより、機体の落下が高速でかなり大きな落下衝撃がCanSatに加わったと推定できる。以下ロケットから打ち上げられ、光によって放出判定をされてから着陸が判定されるまでの気圧の変化のグラフである。



これを元に速度を概算すると、8.5m/sであった。

1度目のログは以下の通りであった

<https://drive.google.com/file/d/1DwBuC6DmMaahfqNyXyRY1mPsh32TkTVy/view?usp=sharing>

・二度目の投下試験

一度目でうまく行かなかった離陸姿勢への移行まではスムーズに行われていた。しかし、時間が経過しても機体が離陸しなかった。その間、LoRa通信によって、地上局にはGPSによる位置座標と、飛行開始の指示を出したことは送られてきていた。

飛行開始が制御的に送られることがないと判断し、棄権となった。

その後の飛行試験において墜落してしまった影響でログを読み出すことができなくなってしまった。

第7.3節 考察

サクセスクライテリアの達成度は以下の表の通りである。

サクセスクライテリア	内容	検証方法	成否
Minimum success	放出・落下判定ができる	目視,制御ログによる確認	成功
	パラシュートが展開する	目視による	成功
	機体が落下衝撃に耐えられる	目視による,ミッション終了後に機体を確認	成功

	機体がパラシュートに干渉することなく飛び立つ	目視による	失敗
	飛行・制御ログが残せる	制御ログの確認	成功
	機体が自律飛行できる	目視による	成功
Full success	0mゴール達成	計測による	失敗
	機体とゴールの距離,位置関係を測定できる	制御・飛行ログと目視による	失敗
Extra success	機体が損傷しないでミッションを終了する	目視で確認(ミッション前後で写真を記録)	失敗

1度目の投下では上空での放出後に光センサで放出判定、気圧センサで地上への着地は判定できていたが、raspberrypi側から送信指示が出ていたにも拘らず、地上局がわでLora通信が受信できず、途絶えてしまっていた。

機体側のLoraモジュールからの通信が途絶えた理由については以下のように考察した。

当日の投下後に起動確認した際に問題なく動いたこと、ラズパイ内部に光センサと気圧センサのログは残っていたこと、またLoraの仕様として電源が落ちた際に初期設定から行う必要があることの3点が判明した。このため、解散衝撃によるLoraの瞬断が原因で、GPSデータが地上局に届いていなかったと考えられる。

2度目の投下で飛行を開始しなかった理由については以下のように考察した。

1度目の投下の反省を生かし、地上への着陸後にGPS取得機能を再起動するシーケンス、arm指示が行われるまでarmをループする機能を追加した。

事前の試験にて、各シーケンスごとの動作確認、着地判定からの飛行までの通しでのシーケンスは確認できていたため、問題ないと考えていた。しかし、実際にはGPSの取得→Lora通信というシーケンスにかかる時間におよそ0.5～3秒のズレがあり、事前の飛行試験ではたまたま0.5秒以下の誤差であったため、その後のarm指示、飛行まで問題なく行っていた。しかし、本番では

1. 上空からの長期間の落下で繰り返し周期の誤差が重なっていたこと
2. 上空でLoraが瞬断していなかったため、再起動→GPSの取得→Lora通信というシーケンスになり、さらに時間がかかったこと

この2点が原因で、GPSを取得するシーケンス指示とarm指示のタイミングがpixracer内で重なってしまい、arm指示は行われたがarm指示できていない状態になってしまったと考察した。前述の通り、armの繰り返し処理はGPS取得→arm指示がpixracerに行われるまでarmを繰り返すというシーケンスであったため、armが行われず、飛行が開始しなかったと考えられる。

事前の試験の回数が少なかったため、機体の完成のタイミングをもう少し早め、より多くの試験を行うべきであった。また、プロペラを取り付けなくても動作が確認できる範囲のミスでもあったため、効率的な試験のやり方が他にあったと考えられる。

第8章 まとめ

第8.1章 工夫点・努力した点

機構

- 航続距離延長のための工夫
 - 特殊素材を使った究極の軽量化
 - チタンネジやジュラルミンナットを使用し、ステンレスネジに比べておよそ半分の重量になった。

- lidarとGPSを用いたホバリング性能の高さ
- フィルタリングにより、オーバーシュート・アンダーシュートの大幅な低減を実現

第8.2章 課題点

機構

- 軽量化を最優先したため機体の剛性が不足していた。
- 二つのコンセプトの機体を同時に開発していた時期があったため開発のための時間が圧迫されてしまった。
- 手加工の部品の精度を確保することができなかった。
- 機体の開発ペースを急ぎすぎてフィードバックが疎かになっていたように感じた。
- タスクの数と人数のバランスが微妙に噛み合わず手が空いてしまうことがあった。
- 軽量化の代償として機体のメンテナンス性が悪かった。
- 飛行ロボットコンテストやNHKの学生ロボットコンテストなど他の開発を経験した人の先入観が裏目に出ることがあった。
 - 飛行機よりも剛性や加工精度がかなり求められる。
 - 屋外飛行する機体としては採用するべきでない部品があった(屋内飛行機用モーター、ペラゴム)
 - ロボコンよりも失敗による損害が大きく、同じペースでの開発はできない。

電装

- 能代の基盤で基本設計は完成していたが、より薄く、洗練した基板を発注したので本番用基板が2枚とも完成したのはアメリカに行ってから
- 各自できるタスクを増やしていきながら役割分担の漸次強化
- 大きな問題は大体班の間で起きる
 - 動かないloro、コンパチでないHAWKとRACER
- コンパチ(嘘)を維持したまま軽量、小型部品に変更
 - 予算、コンパチではない問題
- 最初から覚悟はしていたが、予算が莫大に(電装だけで35万)
 - PX4対応とは書いてあるがソフトが動作するだけで、制御定数、センサの特性は全く異なる
- 耐久性
 - 箱で覆われているHAWKに対して、すべての部品がむき出しのRACERは衝撃に弱いように感じた。
- モーターの選定
 - 電費のみを考えて選んでしまったため、取り付けで様々な問題が発生し、振動という形で跳ね返ってきた

制御

- 開発方針について
 - 安定飛行が可能な状態の機体を残してプログラムの実行を行わなかったこと
 - pixracerとpixhawkが似ているという前提で制御を行っていたこと
 - 画像認識を使うにも関わらずpython中心で開発したこと。
 - 制御由来の飛行挙動の不安定化(PIDの発散)に注目するのが遅れたこと
 - シミュレーションの不足
 - ログのシステムを公式のものではなく、刷新するべきだった
- 役割分担・スケジュール等について
 - 実機を触る人とコードを書く人を完全に分けてしまったこと
 - 途中まで構造・電装にUIの使い方や制御の仕組みを明確に伝達できていなかったこと
 - 7月あたりまでのデバッグ作業分担が偏っていたこと
 - 日本で飛行テストに必要な時間を確保できなかったこと
- 知識面
 - 制御班で作成した資料が多すぎたため、重要なものを整理しておくべきだった

- 非同期処理への試験不足
- Pixracerに付属しているモジュールのデータが不鮮明であったこと

マネジメント

- スケジュール
 - かなり厳しいスケジュールの中で、2種類の案から一種類に絞るタイミングが幾度か後ろ倒しになってしまった
 - 8月末に日本で飛行試験をたくさんしてから行く予定であったが、想定よりも回数が減ってしまい、色々な条件を試すことができていなかった
- タスク分配
 - 誰がどれほどの仕事量を持っていて、手が空いているのが誰かなどを全て把握している人間がいなかった

第8.3章 今後の展望

同じメンバーでARLISSに出場することはおそらくないが、来年参加するであろう学科の後輩が0mゴールを達成できるよう、我々のチームとして得られた知見を集積し引き継いでいきたい。