

ARLISS2022 報告書

提出日:2022年 11月 17日

- チーム情報

CanSatチーム名	東京大学 Hummingbird
CanSatチーム 代表者情報	王方成 houseiwang0310@g.ecc.u-tokyo.ac.jp, 070-4317-0502
UNISEC団体名	東京大学中須賀船瀬研究室
UNISEC団体 学生代表	中村 介
責任教員	中須賀 真一 nakasuka@space.t.u-tokyo.ac.jp
CanSatクラス	Open Class

- メンバー

役割	名前(学年)
PM	王方成
会計・制御班長	平野皓己
構造班長	天川海音
回路班長	加計道成
サブPM	長谷川尚也
サブPM	久島明洋

- CanSatの製作目的・大会参加理由

フライバックでの0mという未だ誰も実現したことのない記録を達成するため。ドローンによるフライバックは未だ例が少なく、0mを達成するにあたり、ホバリングできるという特徴は非常に有利であるため。

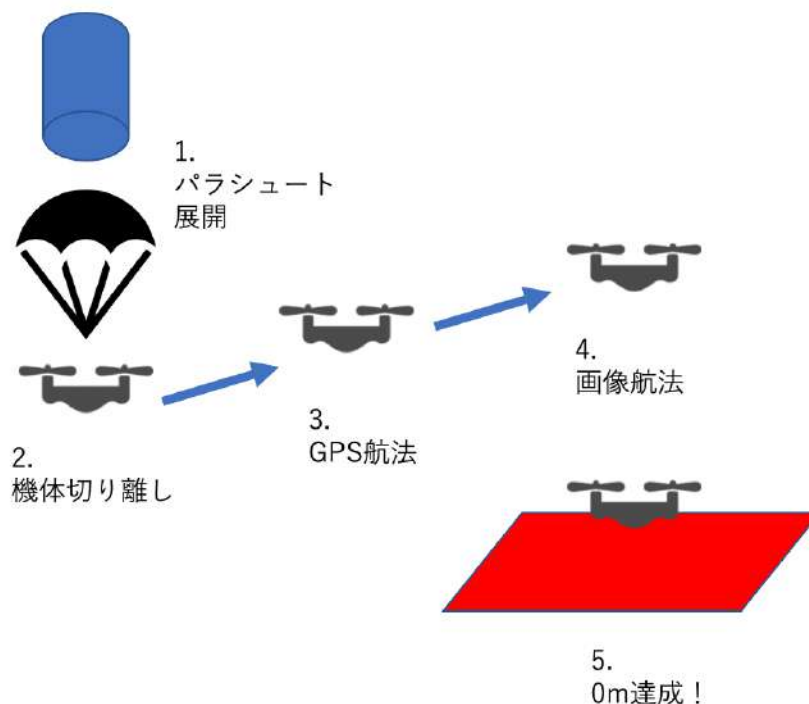
目次

第1章	ミッションステートメント(簡潔に文章でミッション定義)
第2章	サクセスクライテリア
第3章	要求項目の設定
第4章	システム仕様
第5章	試験項目設定
第6章	実施試験の内容
第7章	工程管理、ガントチャート作成(スプレッドシートを推奨)
第8章	責任教員の所感
第9章	大会結果報告
第10章	まとめ

第1章 ミッションステートメント

マルチコプターによりフライバック史上初の0mを達成する。

CanSatのランバックでは複数のチームが0mを達成しているが、フライバックでの0mはいまだ達成されていない。東京大学のチームは昨年、0mを達成するための新たなアプローチとして、マルチコプターによるフライバックを提案し、有効な手法であることを実証した。しかし、機体の展開から自律飛行へのスムーズな移行や、0m達成に必須となる画像航法を駆使して正確にターゲットに着陸する技術は発展途上である。特に、長時間・長距離のフライト時の電力不足は深刻な課題である。本ミッションの目標はこれらの技術的課題を解決し、フライバックでの5m以内を達成することである。また、さらに高い目標として、画像航法を駆使して史上初の0mを達成することである。



第2章 サクセスクライテリア

ミニマムサクセス	<ul style="list-style-type: none">1. パラシュートが展開/機能する2. ケースから機体を放出できる3. 機体とケースを分離できる4. 機体のアームを展開できる5. GPSなどで自己位置を推定できる6. ログが取得できる
フルサクセス	<ul style="list-style-type: none">1. 適切な高度でパラシュートの抵抗を変更できる2. GPSなどによる航法を用いて、ゴール地点から5m以内に到達できる3. 適切な終了判定、終了処理ができる4. 機体が損傷せずに終了できる

アドバンスドサクセス	<ol style="list-style-type: none">1. 撮影画像から機体に対する目標地点の位置が推定できる2. 画像航法を用いて、ゴール地点に着陸できる
------------	--

第3章 要求項目の設定

3.1 システム要求(安全確保, レギュレーションのための要求)

要求番号	システム要求項目 (ARLISS打ち上げ安全基準)
S1	投下する機体・ケース・付属物の総質量が1050g以下である
S2	機体のサイズが直径146mm以下、高さ240mm以下である
S3	機体は円筒形のキャリアから容易に放出できる
S4	打ち上げ時の準静的荷重(10G以上)によって、安全基準を充足するための機能が損なわれていないことが試験で確認できている
S5	打ち上げ時の振動荷重(30Hzから2000Hz・15G以上)によって、安全基準を充足するための機能が損なわれていないことが試験で確認できている
S6	ロケット分離時(パラシュート開傘時)の衝撃荷重(40G以上)によって、安全基準を充足するための機能が損なわれていないことが試験で確認できている
S7	地表近くで危険な速度で落下させないための減速機構を有し、その性能が試験で確認できている
S8	ロストへの対策として、機体・ケースの落下位置を、GPS等を用いて確認することができ、有効性が試験で確認できている
S9	機体地上局間通信ができる
S10	打ち上げ時の無線機の電源OFFの規定を遵守できることが確認できている (FCC認証かつ100mW以下の機器はOFFしなくて良い。また、スマートフォンを用いる場合はFCC認証かつソフトウェアまたはハードウェアスイッチでoffにできること)
S11	無線のチャンネル調整に応じる意思があり、また実際に調整ができることを確認できている
S12	S1-10の充足を確認した設計のCanSatによって、ロケットへの装填からミッションの開始、打ち上げ後の回収までを模擬したEnd-to-end試験を実施できており、今後、安全性に関わる大幅な設計変更はない
S13	空中で安定した姿勢を維持できる
S14	ミッション前後も含め安全に細心の注意を払ってリチウムポリマー電池を取り扱う
S15	飛行中に高度と電池残量をモニターし、ミッション継続が困難と判断された場合にはスロットルを低下させて軟着陸する
S16	飛行中に緊急事態が発生した際には、地上局から強制着陸指示を無線で送り、安全に軟着陸する

S17	CanSatの収納・投下準備を5分以内に終わらせる
-----	---------------------------

3.2 ミッション要求

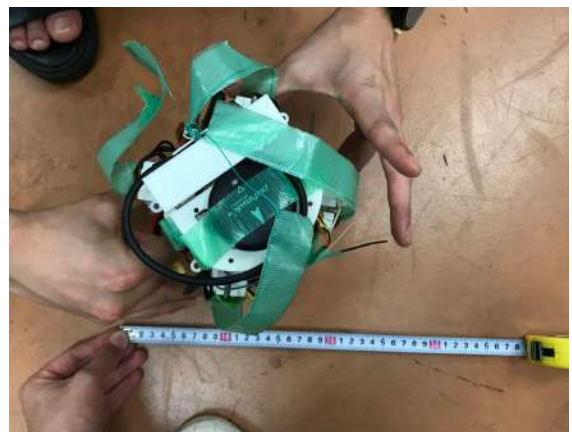
番号	ミッション要求項目
M1	モータを用いて、飛行に十分な推力を5分程度継続的に得て、機体を制御できる
M2	機体放出直後に、機体安定用のメインパラシュートを展開できる
M3	パラシュートと機体が絡まることなくケースを展開できる
M4	ケースと機体の間に十分な距離を確保したタイミングで、機体の腕を展開できる
M5	腕を展開し姿勢が安定した後、機体をケースから切り離すことができる
M6	ミッション時に人間が介入しない自律制御を実施することを確認出来ている
M7	GPS航法により、カメラでゴール地点が撮影できる位置まで機体を自律的に誘導できる
M8	地表の画像を撮影できる
M9	撮影画像から、ゴール地点と機体の相対位置を推測できる
M10	画像航法を用いて、ゴール地点近くに機体を誘導し、軟着陸できる
M11	適切な終了判定、終了処理を行える
M12	ミッション後、規定された制御履歴レポートを運営及び審査員へ提出しログ・取得したデータについて説明が可能な状態である

第4章 システム仕様

4.1 機体外観

直径[mm]	収納時:210 展開時:570
高さ[mm]	収納時:280 展開時:180
質量[g]	機体:749 ケース:151 計:900

次の写真のように腕を開閉する構造とした。腕を閉じて、ケースに入れ、ケースから出ると腕が展開するようにした。



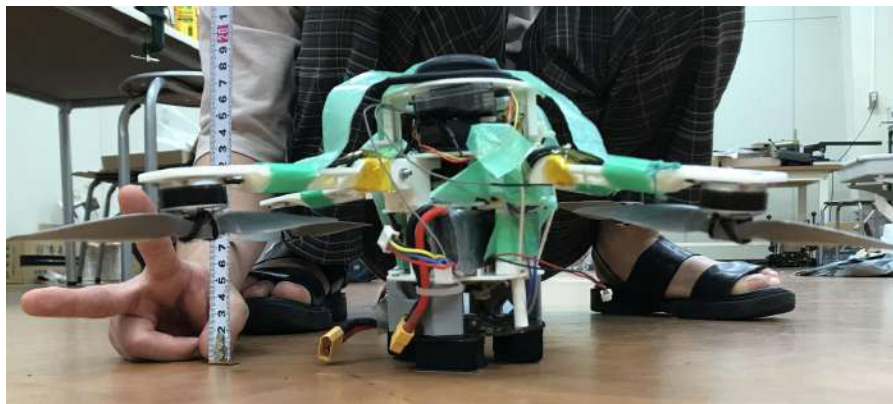
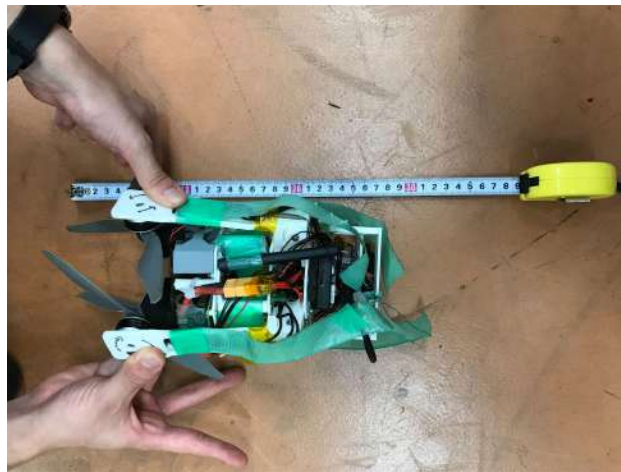
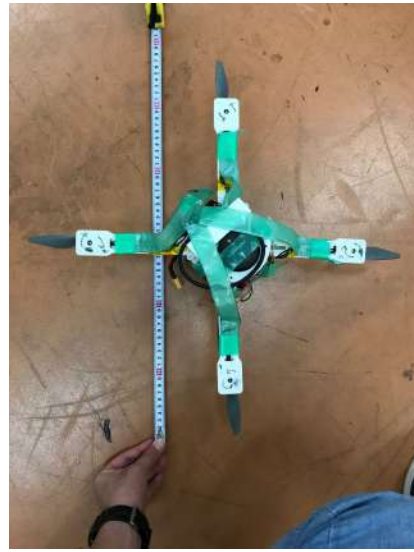


図3.1.1. 機体の外観

4.1 機体の内観・機構

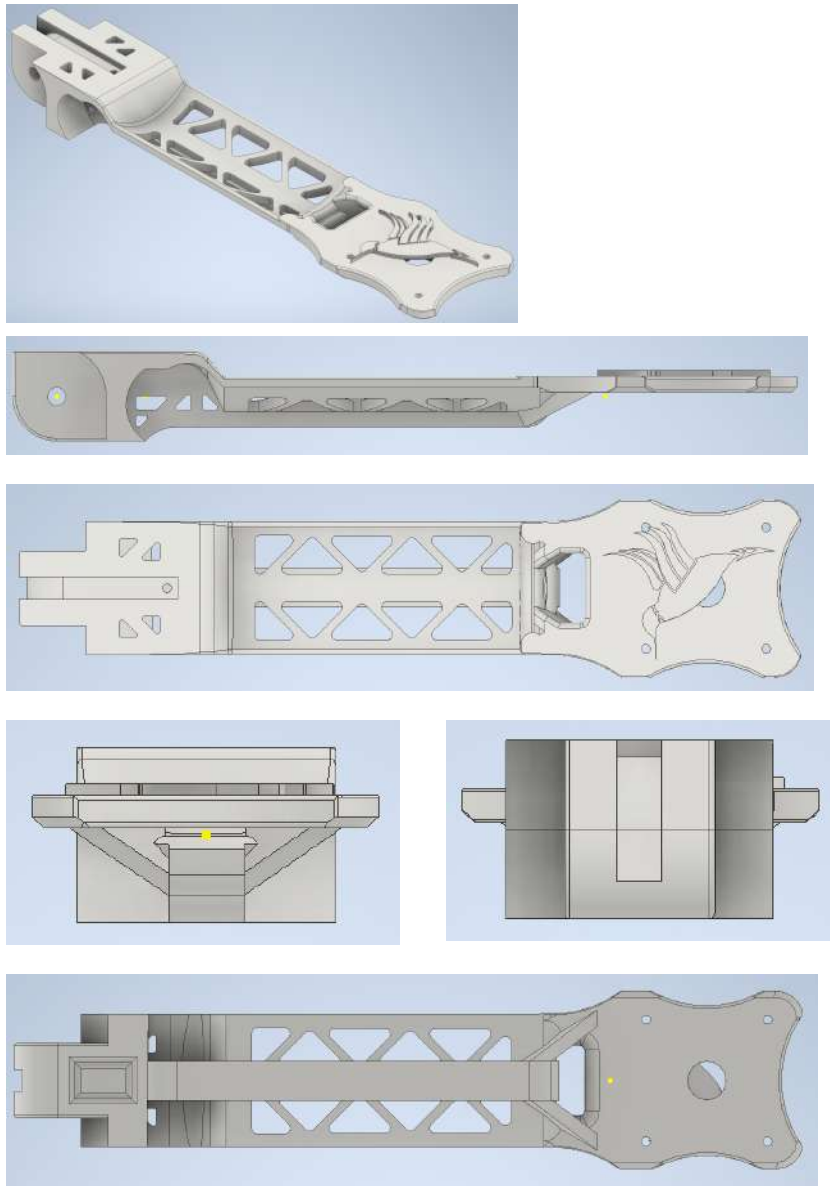


図3.2.1. 機体の腕部分

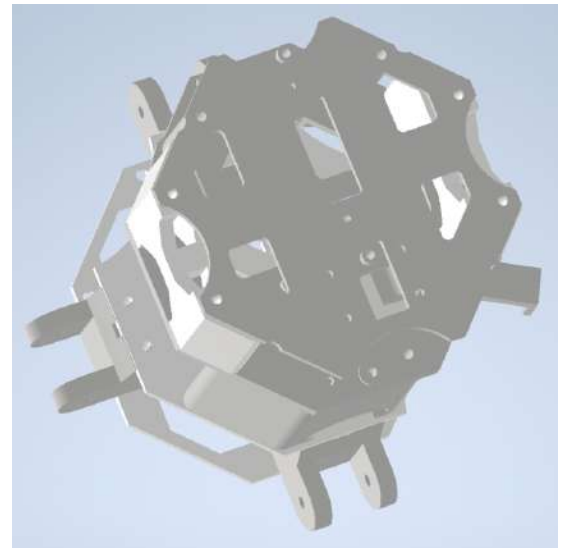
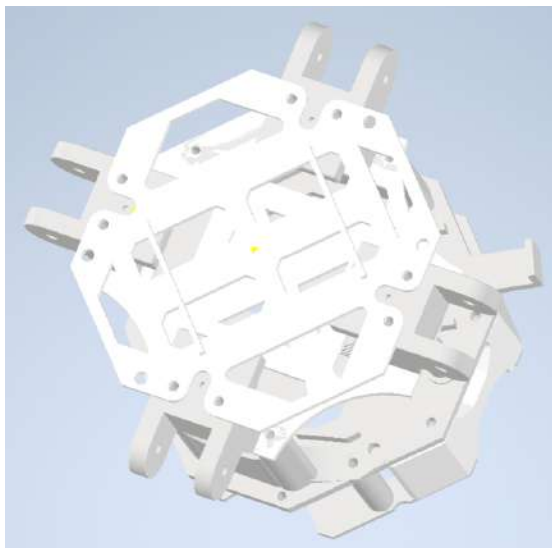
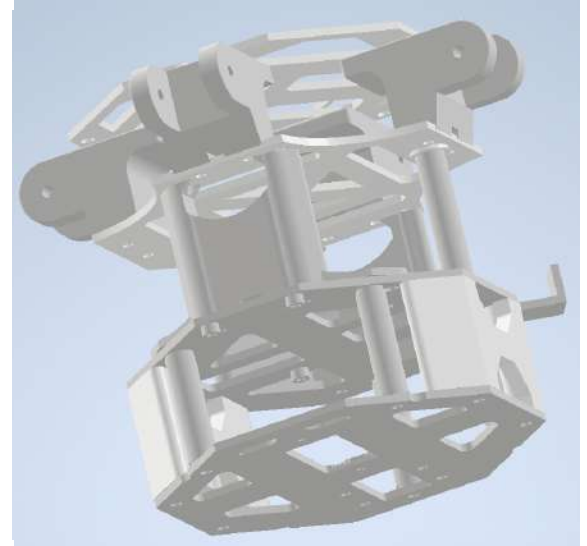
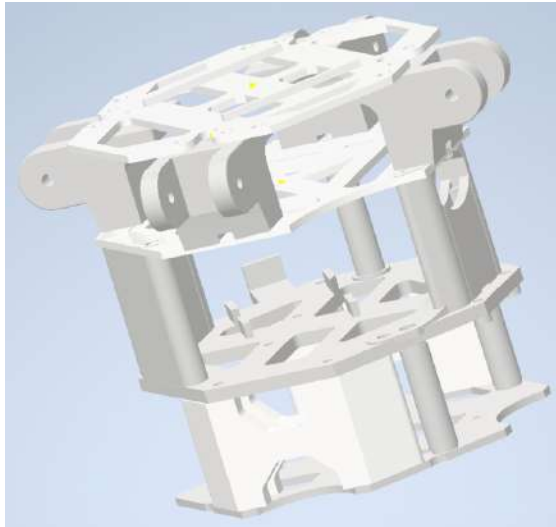
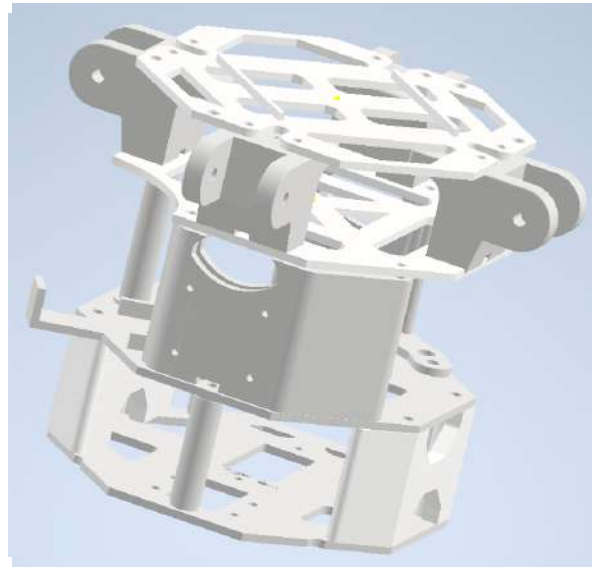
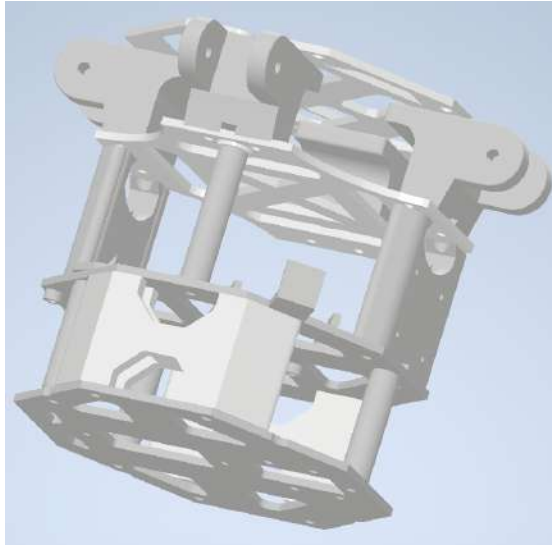


図3.2.2. 機体の胴体部分

機体の構造は4枚のボード、9本の支柱、4本の腕及びその付け根からなる。それぞれの部位は肉抜きをすることで軽量化している。特に、腕については折りたたむ機能と軽量化、生じる応力への耐久性を考慮して断面をY字形に近い構造とした。また、支柱も軽量化と耐久性を兼ねてスパーサーと中空円筒を交互に配置する仕様とした。底に下向きに配置する距離センサは機体の中に入れる仕様としたため、距離センサ用の穴を底部のボードに開けた。収納時にプロペラが干渉しないように底部のボードの辺を抉った。

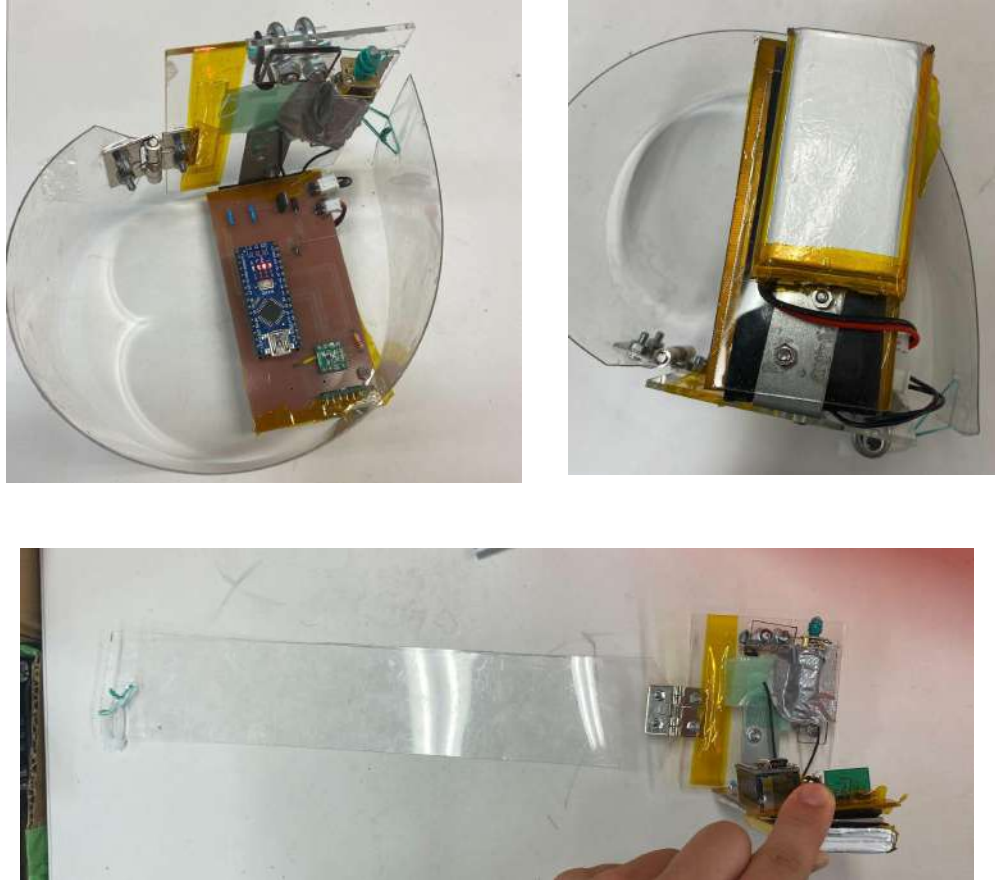


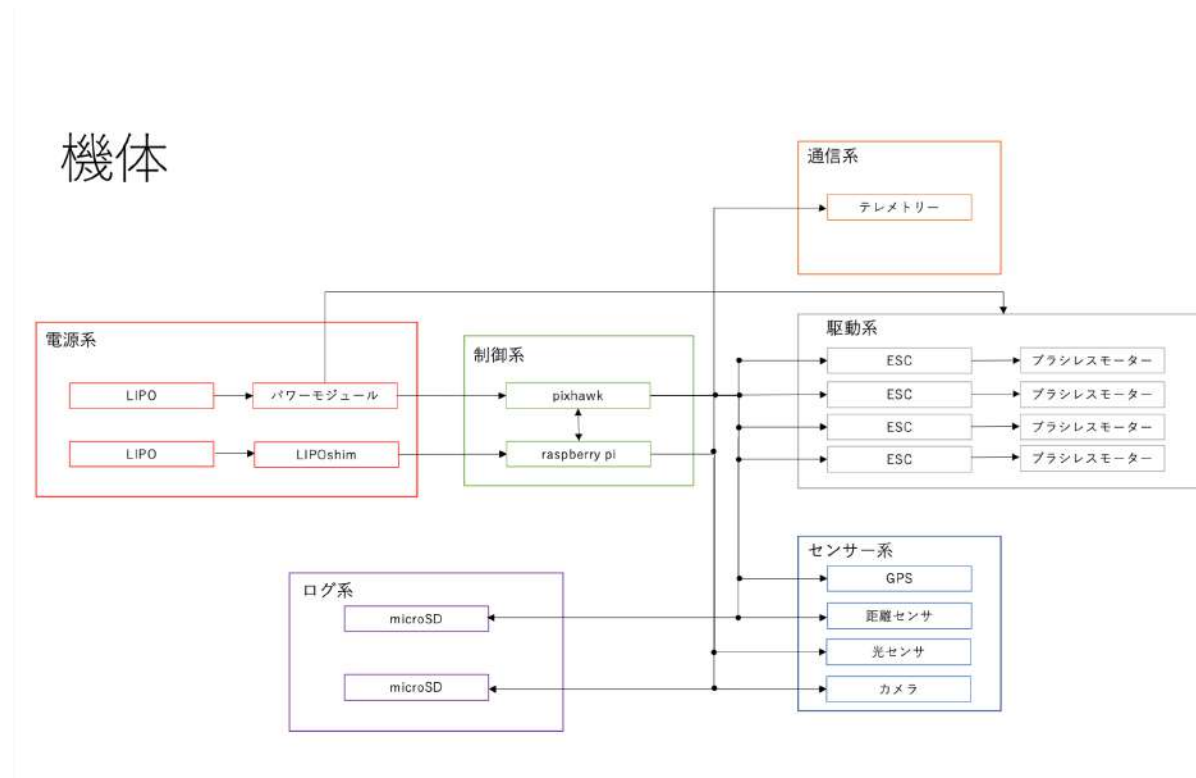
図3.2.3. ケースの構造

ケースの構造は、基本的に2つの平面からなる。1つはケース用の回路およびパラシュートが接続してある平面、もう1つはドローンを抱え込む腕およびドローンを吊るす部分が接続されている平面である。ケースはこの2つの平面をL字金具で接続したものになっている。ドローンを抱え込んでいる腕を解放する際は、パラシュートが展開することによって生じる物理的な力を利用している。これは介在する部品を可能な限り減らし、より確実な作動を実現するための機構である。また、腕を解放した後、ドローンを安定化させるため、数秒間ドローンはケースを介してパラシュートに吊るされた状態となる。これを解放する際は、センサー類の検知によってモーターを作動させることで、ドローンを切り離す機構としている。この二つの異なる動作において、基本的に同じ機構を採用しているのは、不具合の多様化を避けるためである。片方の機構で生じた問題はもう片方でも生じうるため、試

験する上で相乗効果を得やすく、コストを抑えられる上、より信頼性のある機構の実現に役立つ。

基盤表面は不燃性のカプトンテープで覆うことにより、接触部での発火を防いでいる。

○回路の特徴

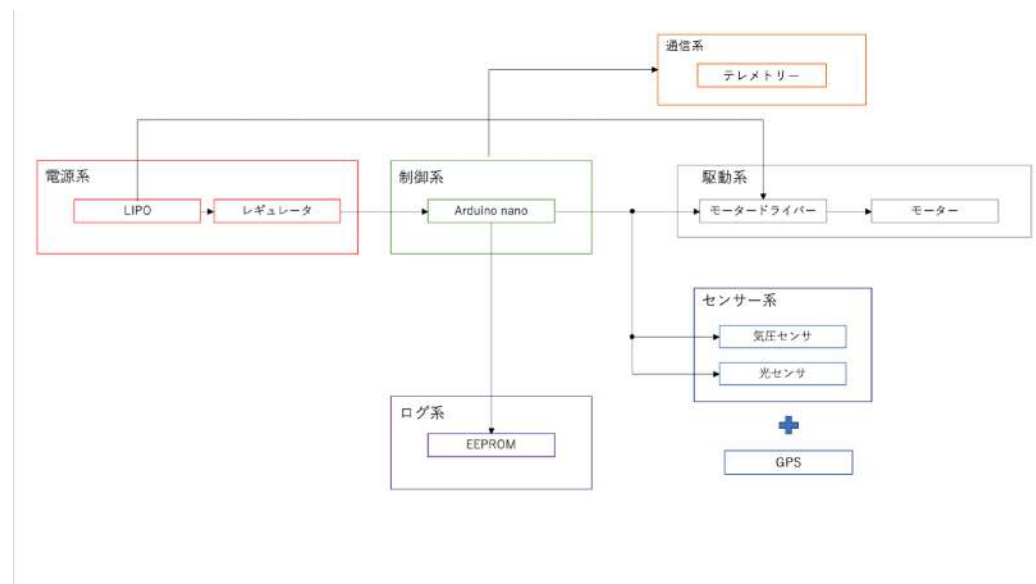


機体の航法、制御はフライトコントローラーであるPixhawkが担う。また、機体の行き先をPixhawkに指示する誘導をRaspberry piが担う。Pixhawkには、モータの制御を行うESC、GPSセンサ、距離センサが繋がれている。また、アームが展開されたことを認識する光センサと、ゴール地点付近で地面を撮影するカメラはRaspberry piに接続されている。部品表は以下の通りである。

pixhawk4	https://cdn.sparkfun.com/assets/d/d/9/9/3/Pixhawk4-DataSheet.pdf
raspberry pi zero 2 W	https://datasheets.raspberrypi.com/rpizero2/raspberry-pi-zero-2-w-product-brief.pdf
テレメトリー	https://easel5.com/service/products-information/products/wireless-module/es920lr2/
ESC	http://www.gforce-hobby.jp/products/G0211.html
ブラシレスモーター	https://www.little-bellanca.com/SHOP/21364.html
GPS	http://www.holybro.com/product/pixhawk-4-gps-module/

距離センサ	https://www.sgbotic.com/products/datasheets/sensors/DE-LiDAR%20TFmini%20Datasheet-V1.7-EN.pdf
光センサ	https://akizukidenshi.com/download/ds/macron/MI5_series_ip.pdf
カメラ	https://docs.rs-online.com/3b9b/0900766b814db308.pdf
3s LIPO 1800mAh	https://www.amazon.co.jp/2パックLipoバッテリー3S-1800mAh-SkylarkM4-FPV250ミニシュレッダー200INDY250-Nighthawk-250RCヘリ飛行機UAVドローンFPV/dp/B07H7GSQMQ?th=1

ケース



ケースに関しては、消費電力削減のために機体で用いたRaspberry piではなく、Arduinoを用いることとした。機体が放出されたことの認識については、気圧センサと光センサの二つを用いることで冗長性を持たせている。また、展開のアクチュエータはモータ駆動となっている。部品表は以下の通りである。

1s LIPO	https://www.li-polymer-battery.com/wp-content/uploads/2021/03/LP606090-3.7V-4000mAh-without-protection-circuit-and-wires-Datasheet.pdf
arduino nano	https://www.arduino.cc/en/uploads/Main/ArduinoNanoManual23.pdf
気圧センサ	https://akizukidenshi.com/download/ds/bosch/BST-BME280_DS001-10.pdf

モーター	https://www.sengoku.co.jp/mod/sgk_cart/detail.php?code=EEHD-0SDW
光センサ	https://akizukidenshi.com/download/ds/macron/MI5_series_jp.pdf
テレメトリ	https://easel5.com/service/products-information/products/wireless-module/es920lr2/
GPS	https://akizukidenshi.com/catalog/g/gK-09991/

○使用電源について

ケース、本体共にリポバッテリーを用いる。リポバッテリーは機体の内部に取り付け、落下時に衝撃ができるだけ加わらないように設計されている。また、機体に衝撃緩衝用のスポンジがつけられている。

持ち運び、保管についてはリポガード(下にurlを示した)を用い、また万が一に備えて塩を携帯するものとする。また、過放電防止のため、必ず残量を確認するようにする。

https://www.amazon.co.jp/最強防災-LiPo-Guard-リポバッテリー-セーフティバッグ/dp/B00D77M7WW/ref=sr_1_5_mod_primary_new?keywords=リポガード&qid=1659085881&sbo=RZvfv%2F%2FHxDF%2BO5021pAnSA%3D%3D&sr=8-5

○ミッションシーケンス

ミッションシーケンスのリンクは以下の通り。

<https://drive.google.com/file/d/1KmxLbAi5wnleChj0Q8fSPLNg4LziRTfG/view?usp=sharing>

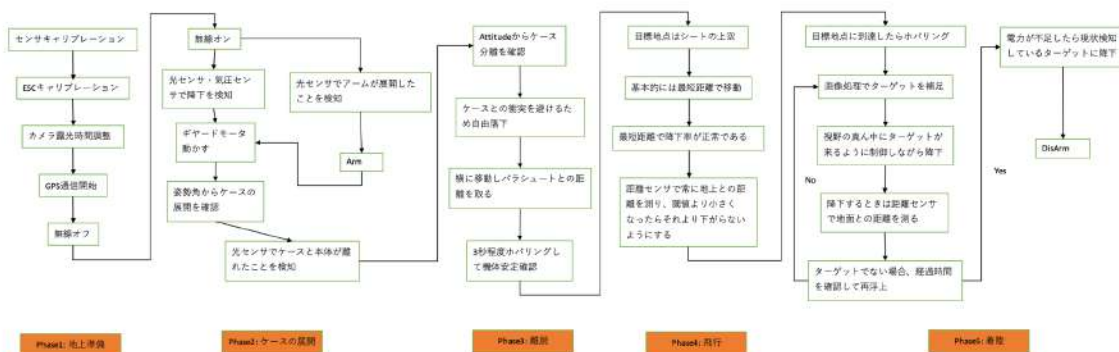


図5.1.1. ミッションシーケンス

○アルゴリズム

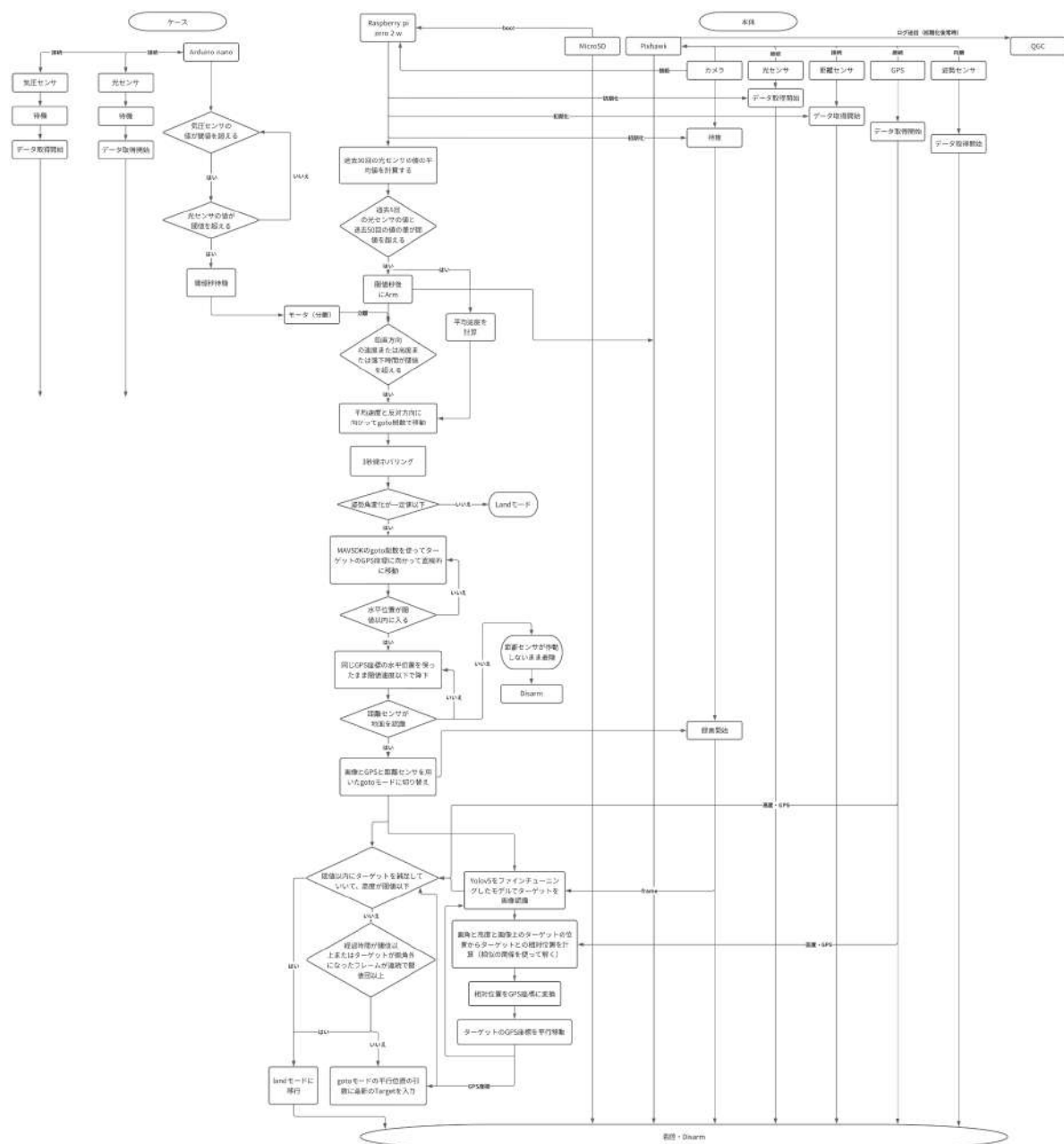


図5.1.2. アルゴリズム

上のフローチャートは下のリンクでより詳しく見ることができる。

https://drive.google.com/file/d/1GWV_IsY-SaTt4MdFmg-kbmtaUujkxFU/view?usp=sharing

アルゴリズムの特記事項

- 去年のチームで実証されたこととして、MAVSDKのgotoモードは他のモードに比べて安定しているということがある。GPSによる誘導はアルゴリズムの根幹を占めるモードであるため、信頼性の高いモードを使用する。
- フェイルセーフ設計として用意されているのは次の点である。
 - 分離後に姿勢が安定しなかった場合は緊急着陸するようにプログラムする。
 - 自由落下を止めるタイミングは高度と落下速度、自由落下時間の3つのデータを使って判定することで、確実に減速し危険な落下をしないようにしている。
 - ターゲットを連続して補足できなかった場合は画像航法を中断し、GPSのみを使った制御を用いて着陸する。
 - 時間がかかりすぎてしまった場合も途中で探索をやめて安全に着陸する。
 - 距離センサが作動しないまま着陸した場合は、外部から強制的にプロペラの回転を止められるように割り込み処理を用意する。
 - ログはPX4が自動で作成するulogファイルだけでなく、ファイル保存がうまくいかなかったときのためのリアルタイムでの自作ログを実装する。(jsonファイルにリアルタイムで姿勢やモードなどのデータを書き出し、後にmatplotlibなどで可視化できるようになっている)
- ケースに本体のArm状況を伝える手段がないが、Armとその後の緊急着陸の信頼性を試験回数によって高める。無線通信によってケースと本体でデータをやり取りする方法も検討されたが、無線通信自体に信頼性が容易に担保できない上、システムが複雑になることによるリスクの方がオーバーライドすると考えられたために却下した。
- 分離後にパラシュートに巻き込まれることを防ぐための対策は次のとおりである。
 - 分離直前の速度の平均速度を計算し、局所的な風向きを計算することで、パラシュートが流れる向きを予測し、それと反対方向に退避するようにする。
 - 抵抗はパラシュートよりも本体の方が小さいため、自由落下することによって、ケースとの距離をとる。
- gotoモードを使うために、画像認識によるターゲットの座標をGPS座標に変換する。これにより、最後まで信頼性の高いモードを使い続けることができる。これは実験でも実現可能性が確認されている。この方法のデメリットとして、gpsの誤差に影響されるということがあるため、gpsを用いずに、速度や相対位置を使った航法も実験して精度を確かめたい。(例えば、画像を上下左右4つに区切り、その中でターゲットの中心がどこにあるかで縦横の移動量を操作する方法が考えられている。)
- ターゲットを認識するために使う機械学習モデルはエッジデバイスでの推論に適したYolov5モデルをターゲットの画像でファインチューニングしたモデルである。これにより、ターゲットの画像上の位置を、様々な条件で柔軟に認識することができる。推論速度を高速化するために、Pytorchで学習したモデルをtf-liteモデルに変換する。これにより、より高頻度でターゲットの位置を修正することができる。

第5章 試験項目設定

番号	検証項目名	対応する自己審査項目の 要求番号(複数可)	実施予定日
V1	質量試験	S1	6/29
V2	機体の収納・放出試験	S2, S3, S17	6/29
V3	準静的荷重試験	S4	7/8
V4	振動試験	S5	7/5
V5	分離(パラシュート開傘)衝撃試験	S6	7/5
V6	パラシュート投下試験	S7, M2	6/19
V7	着地衝撃試験	S7	6/15
V8	GPSデータダウンリンク試験	S8, S9	7/6
V9	無線チャンネル変更試験	S10, S11	7/16
V10	End-to-end試験	S12,S13, S14,S15, S16, M1, M6, M7, M8, M9, M10, M11	7/9
V11	制御履歴レポート作成試験	M12	7/9
V12	機体ケース間分離・展開試験	M3, M4, M5	7/6

第6章 実施試験の内容

v1. 質量試験

- 目的
CanSatとパラシュートを合わせた質量が規定質量である1050g以下を満たすことを確認する。
- 試験内容
CanSatとパラシュートを質量計で計測をし、レギュレーションに記載されている質量(1050g)以下であることを確認する。
- 結果
CanSat(758g)とパラシュート(261g)の総重量は1019gであり、レギュレーションの1050g以下であることを確認した。図7.1.1に質量測定結果の結果の様子を示す。

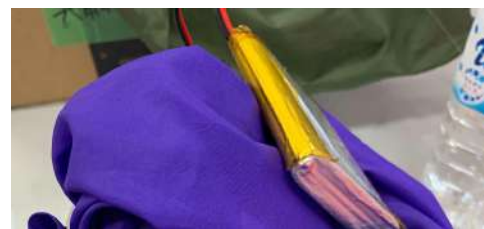


図7.1.1 パラシュートとCanSatの質量

- 考察
パラシュートの質量を含めたCanSatの総重量が、レギュレーションを満たしていることがわかった。今後、開発を進めていく中で、必要な部品が増加する可能性があることを念頭に置き、必要な強度を保ちながら構造部分やパラシュートの軽量化を継続していく必要がある。

v2. 機体の収納・放出試験

- 目的
機体、パラシュート、ケースを含めた全ての機構がキャリア内に収納でき、自重のみで放出できることを確認する。
- 試験内容
機体をキャリアに収納する。また、キャリアの下に敷いた板を取り外し、外力を加えない状況で機体が放出されるかどうかを目視により判断し、映像の記録を行う。
- 結果
機体の収納を撮影した動画は下記のリンクの通りである。
<https://youtu.be/9yC2coBD9Xc>
表7.2.1に示すように、CanSatを収納後、自重で放出することが4回行った実験全てで確認できた。

表7.2.1 キャリア放出実験の結果

	実験動画	放出判定
	https://www.youtube.com/watch?v=7UcdPmFz-KU	自重で放出できた
	https://www.youtube.com/watch?v=pqxWqwkseO8	自重で放出できた
	https://www.youtube.com/watch?v=bNR8K4np-CY	自重で放出できた
	https://www.youtube.com/watch?v=R4Mm1Npijrc	自重で放出できた

ケースが片側しか空いていない場合についてもケースの収納試験を行った。その様子は以下のリンクの動画で見ることができる。

<https://www.youtube.com/watch?v=cpaBoifj7QU>

- 考察
全ての部品がキャリアの中から放出されていることが分かる。また、パラシュートが

ケーシングされた機体よりも少し遅れて落下していることが分かる。ケーシングされた機体とパラシュートの間に距離が生じていることを踏まえると、パラシュートが空気を受けて開き、それと連動してケースが展開、安定した状態でのケースと機体の分離を行うという今後のシーケンスに問題のない放出結果となった。

v3. 準静的荷重試験

- 目的
打ち上げ時の準静的荷重によって、安全基準を充足するための機能が損なわれないことを確認する。
- 試験内容
機体を袋に収納する。袋にロープを繋ぎ、水平方向に振り回す。即ち、CanSatを収納したケースを等速円運動させて10Gの荷重を加えた際に、機体が耐えられるかどうかを検証する。下に半径を測定した写真を載せる。



図7.3.1. 半径の測定

- 結果
実験の様子を撮影した動画は下記のリンクの通りである。

<https://www.youtube.com/watch?v=HeCdNsZy1b8>

$f=0.84[\text{s}]$ の周期で、半径 $r=1.8[\text{m}]$ の等速円運動を行ったところ、

$$\omega = 2\pi/f = 7.48[\text{/s}]$$

$$v = r\omega = 7.48 \times 1.8 = 13.464[\text{m/s}]$$

$$a = v^2/r = 100.71[\text{m/s}^2]$$

であり、 $10G = 98[\text{m/s}^2]$ よりも大きい静荷重をかけることができた。試験後、pixhawkや各種センサは問題なく作動し、機体本体にも損傷は見られなかった。

- 考察

準静的荷重に耐えうる機体であることが確認できた。

v4. 振動試験

目的

ロケットによる打ち上げの際、打ち出しにかかる振動荷重に機体が耐えられることを確認する。

- 試験内容

正弦波振動30Hzから2000Hz・15G以上に相当するランダム振動を与える試験を行い、機体が耐えられるかどうかを検証する。

- 結果

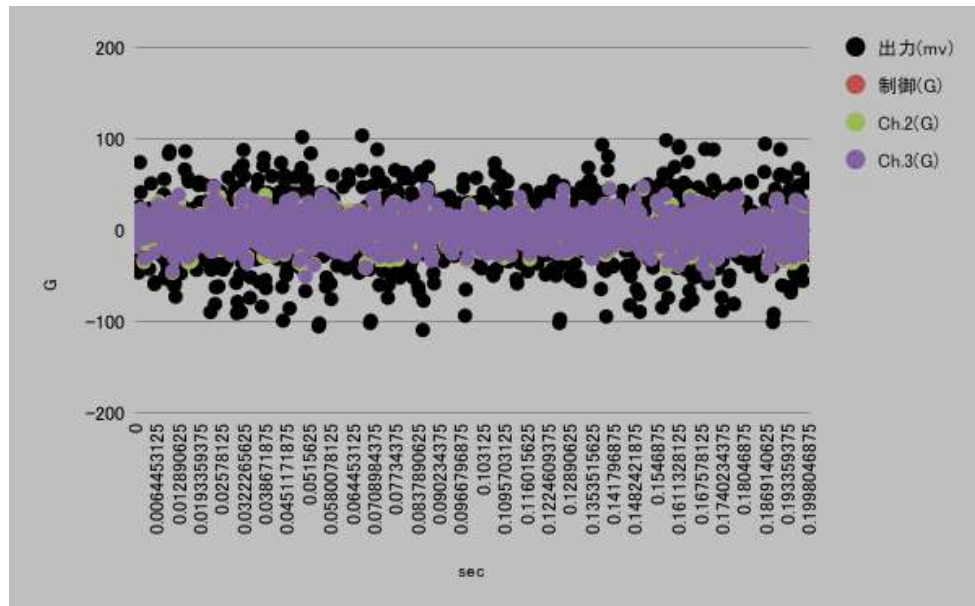


図7.4.1. ランダム振動試験のグラフ

ランダム振動試験を実施した。詳細な結果は以下のリンク

https://docs.google.com/spreadsheets/d/1_qXtPwNr6XXI7nx6xqxZRH0lvA8xKsnZ/edit?usp=sharing&oid=105218032502245381253&rtpof=true&sd=true
https://docs.google.com/spreadsheets/d/1xwllQ0aZBBk3yWSe_QGqPwUth1qHf5uL/edit?usp=sharing&oid=105218032502245381253&rtpof=true&sd=true
<https://docs.google.com/spreadsheets/d/1ApvWuXt3DjHmZkLoPhcKyleGYiZS-tzX/edit?usp=sharing&oid=105218032502245381253&rtpof=true&sd=true>

図7.4.1から分かるようにランダムな振幅が15G以上になっている。

機体に損傷は無かった(上の質量試験の写真)。GPSやモーター等の動作不良が起きている可能性がある。

- 考察

GPS等の動作不良もカバーを付けることで防止できると考えられる。

v5. 分離(パラシュート開傘)衝撃試験

- 目的

機体及びケースがロケットからの分離衝撃(40G程度)やパラシュートの開傘衝撃(50G程度)に耐えられることを確認する。

- 試験内容

$mgh = \frac{1}{2}mv^2$ より、0.2 [s]で速度が0 [m/s] に減少すると仮定し、 $h = 5.10$ [m] から自由落下させて(50Gの衝撃に相当)衝撃に耐えられることを確認する。

- 結果

今回の試験の動画は以下の通り

<https://www.youtube.com/watch?v=9n8EBviCSKE>

動画より、十分な衝撃荷重がかかっていることが確認できる。

また、以下の写真より、ケースに損傷がなく、正常に機能することが確認できる。



図7.6.1. 試験後のケース

v6. パラシュート投下試験

- 目的

パラシュートの開傘、減速が可能であることを証明する。

- 試験内容

高い場所から落下が可能な松戸市新葛飾橋から落下させ、その様子を高角カメラで捉えた。荷重は実際のCanSatの重量に合わせて900gとした。

- 結果

動画は以下のリンクの通りである。

<https://youtu.be/Ogr4uxLjtuY>

- 考察

新葛飾橋の橋桁最下面の地面からの高さは8mであり、その半分の4mの高さから地面に達するまでの時間が動画から0.75秒のため、この区間の速度は5.3m/s程度である。これを終端速度として推算したパラシュートの抵抗係数は0.314であり、逆にこの抵抗係数のパラシュートが完全に開いた状態で8mの高さから自由落下させる計

算をすると、4mでの速度は5.13m/s、8mでの速度は5.29m/sで実測値とほぼ一致するため、この値を終端速度として扱っていいと言える。今後パラシュートを変更した場合は都度試験を行うこととする。

v7. 着地衝撃試験

目標

パラシュートにより落下した際の衝撃に耐え、ミッション遂行の機能を失っていないことを示す。

- 試験内容

V6で終端速度が5.3m/sと判明しているため、推力を突然失い2.5m以上の高さから落下した飛行試験の結果を分析する。

- 結果

飛行試験のログと落下した高さは図7.8.1の通りである。

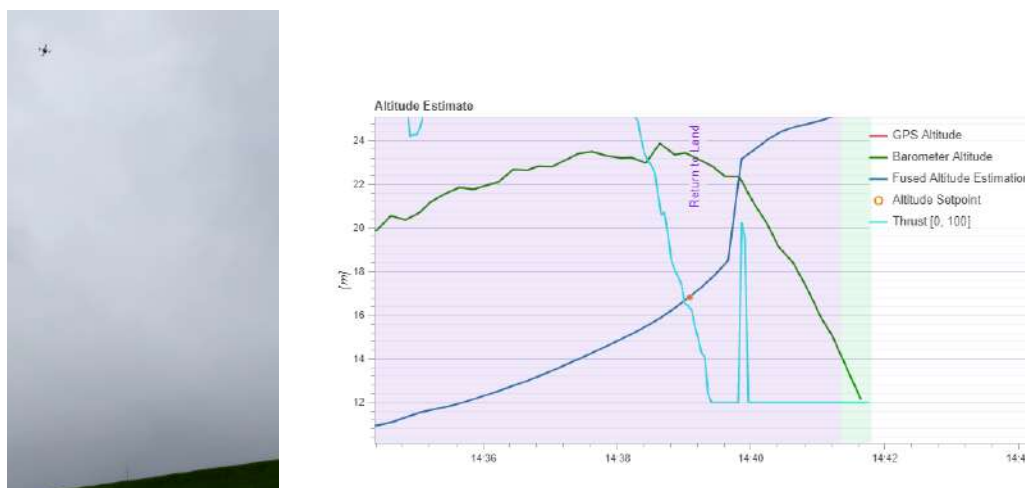


図7.7.1. 飛行試験のログ及び落下高さ

- 考察

落下後も機体に損傷はみられなかった。実際に下の動画のように飛行試験を続行している。バッテリーも問題なかった。

https://www.youtube.com/watch?v=DuNM_hYEvsU

同様に図7.8.2のようにログもとれておりセンサーが正しく動作し続けている。

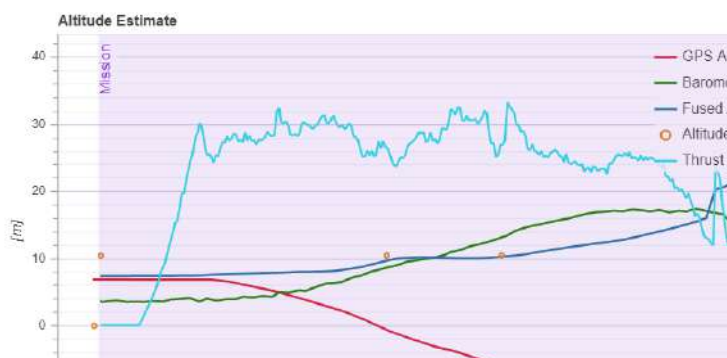


図7.7.2. 落下後の飛行試験のログ

v8. GPSデータダウンリンク試験

- 目的
機体および機体を収納するケースが取得したGPSの位置情報を地上局に送信できること、通信可能な距離を確認する。
- 試験内容
開けた場所である河川敷において、GPSと地上局との距離を変えながら、通信可能な距離範囲を検証する。
- 結果
以下の写真に示す場所で、GPSダウンリンクが可能であることを確認した。



- 考察
1.6km程度の距離であれば十分にGPSダウンリンクが可能であり、障害物の存在を考慮すれば、4km程度の通信距離が必要であるARLISSでも十分に対応ができると考えられる。

v9. 無線チャンネル変更試験

- 目的
無線のチャンネル調整が必要となった場合に、チャンネルを変更できることを確認する。また、ロケット収納時に無線の電源が切れることを確認する
- 試験内容
通信モジュールにおいて周波数を変更できることを検証する。また、モジュール自体の電源を切れることを検証する

- 結果
Tera Termを通じて、テレメトリの周波数を変更できることを確認した。また、モジュール自体の電源を、MOS-FET回路を用いて制御できることを確認した
- 考察
レギュレーションを満たしている

v10. End-to-End試験

- 目的
CanSat投下からパラシュート展開、ケース展開、ケース分離、自律飛行、終了判定・処理、データ取り出しの一連の動作ができることを確認する。
- 試験内容
一般の橋梁などからでは十分な高さを確保できず、試験飛行にて上記の一連動作を確認することは難しいため、機体放出(1)、機体側のアームと飛行開始、GPSによる誘導(2)、それぞれで動作を確認する。
 1. キャリアから放出
 2. 機体の切り離し
 3. 機体が揺れながらアーム->飛行開始
- 結果
 1. <https://www.youtube.com/watch?v=oNik6HtFAu0>
 2. https://www.youtube.com/watch?v=pGDyxmh_cgQ
 3. <https://youtu.be/1lxTdIMDtmQ>
- 考察
 - a. ミニマムサクセスを十分に達成している

v11. 制御履歴レポート作成試験

- 目標
制御履歴レポートが作成できることを確認する。また、バックアップのグラフが作成できることを確認する。
- 試験内容
空中Arm試験を行ったときのログによってレポートを作成した。試験動画は以下のリンクにある。
https://drive.google.com/file/d/1C_uY_bx0y3_aDjJh2zx_i3aHccD-z3ji/view?usp=sharing
- 結果
試験飛行中の制御履歴はdisarm後に自動的にulogファイルが保存され、Flight Review (<https://logs.px4.io/>)というサイトで可視化することができる。
以下のサイトまたは、その下のGoogle Driveのリンクからulogファイルによるログが確認できる。
https://logs.px4.io/plot_app?log=54b5fd4e-c3cd-4ea8-b24b-b67dabf2f1dd
<https://drive.google.com/file/d/14LGwHhBzNPZliTF1bY1d22X-fr0FzVWw/view?usp=sharing>
そのバックアップとして、リアルタイムでも制御履歴をjsonファイルに書き出し、それを取り出すことによって、以下のようなグラフを作成することができることを確認している。下のグラフは、上のリンクと全く同じ飛行試験中にリアルタイムで記録したものを可視化したものである。1段目左がロール角、1段目右がヨー角、2段目左がlidar（距離センサ）による高度、2段目右がピッチ角のグラフである。lidarのグラフを見る

と、60cm程度の高さからスタートし(最初の変数の初期値の影響で0mになっている)、3m程度の高さまで上がった後、着陸している様子が確認できる。他にも、目標地点と自分のその相対位置やアクチュエータの制御目標なども可視化することができる。このときに使ったjsonファイルへのリンクは以下の通り。

https://drive.google.com/file/d/1O9smAiqp7G-NI7-gLeaXyl_Ab-h3sL8p/view?usp=sharing

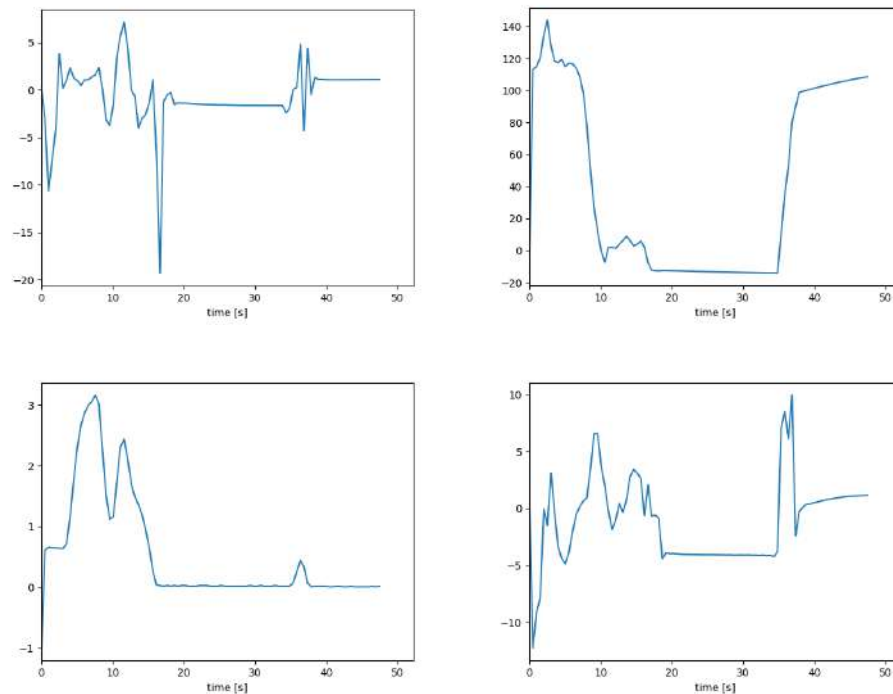


図7.12.1. バックアップ制御履歴のグラフ

下のグラフはulogファイルを可視化した時のz方向の位置のログである。このグラフがz方向を下にとっていることに注意すると、上の2段目の左のグラフと比較すると、確かに値がほとんど一致していることがわかる。

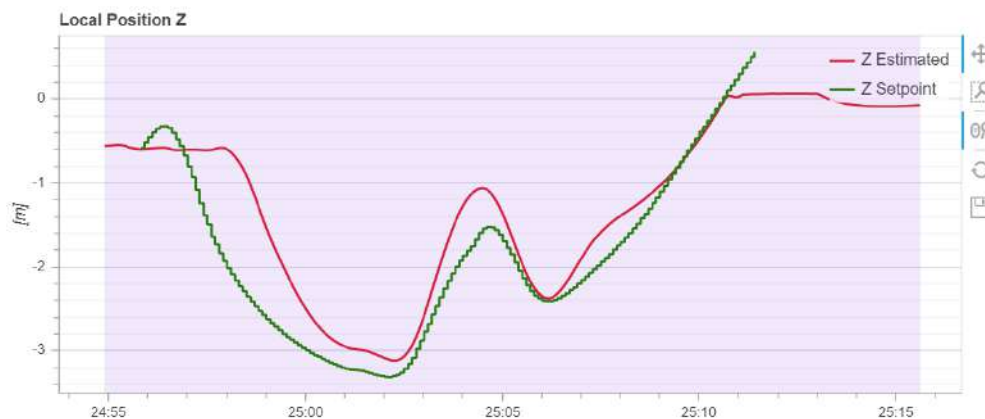


図7.12.2. ulogファイルによるz方向の位置のグラフ

v12. 機体ケース間分離・展開試験

- 目的
CanSatがキャリアから分離した後に、確実にパラシュートが展開できること、ケースが展開し機体が飛行可能な状況まで展開できること、安定した姿勢で機体とケースを分離できることを確認する。
- 試験内容
キャリアを開けてCanSatを投下し、パラシュートの開傘、ケースおよび機体の展開、機体とケースの分離を行う一連の動作ができることを検証する。具体的には、
 - ①キャリア解放後、自重でキャリアから出てくる
 - ②CanSatを投下後、パラシュートを展開できる
 - ③パラシュートを展開後、ピン抜き構造が作動する
 - ④ピン抜き構造の作動によりドローンが適切に展開する
 - ⑤ギヤードモーターの作動によりピンが抜かれ、ドローンが落下するの5項目の確認である。
- 結果
以下の動画に結果を示す。

パラシュートの展開から切り離し
<https://www.youtube.com/watch?v=oNlk6HtFAu0>

機体の展開から切り離しまで
https://www.youtube.com/watch?v=pGDyxmh_cgQ
- 考察
必要な機構は十分に達成されている

第7章 ガントチャート(工程管理)

ガントチャートは次のリンクからアクセスできる。

https://docs.google.com/spreadsheets/d/15CyWA0-ynnlkIYTDdLQfq_JJk97_E2tuMnrCl6aL-cU/edit?usp=sharing

第8章 責任教員の所感

東京大学の3年生の2チーム合同で指導し、同じペースでの開発を見てきているので、両方のチームに共通のコメントを行う。

PDR、CDRを私が参加し、研究室のARLISSのOBなども参加して実施し、所定の試験の予定を確認、また目標を実現するためのアドバイスを与えてきた。その後、7月29日に日大の二和グラウンドをお借りして、気球により高度50～60mまで上げて落下させ、各種の試験を実施した。個々の試験は別々に実施して（サブシステム試験）確認を行ったが、最終エンドtoエンド試験はまだ実施していない。それぞれの試験項目の合成により、所定の目標に向けたフライトができるであろうと考える。

また、安全のための試験はサブシステム試験により確認している。

プロジェクトマネジメント的にも、チームリーダーのもとで、しっかりと役割分担をして検討を続けており、competitionではいい成果が出ることを期待している。

第9章 大会結果報告

12.1. 目的

ロケットから放出された後、自律飛行を利用してターゲットに近づくという予定されたミッションが正常に実行されることを確認する。

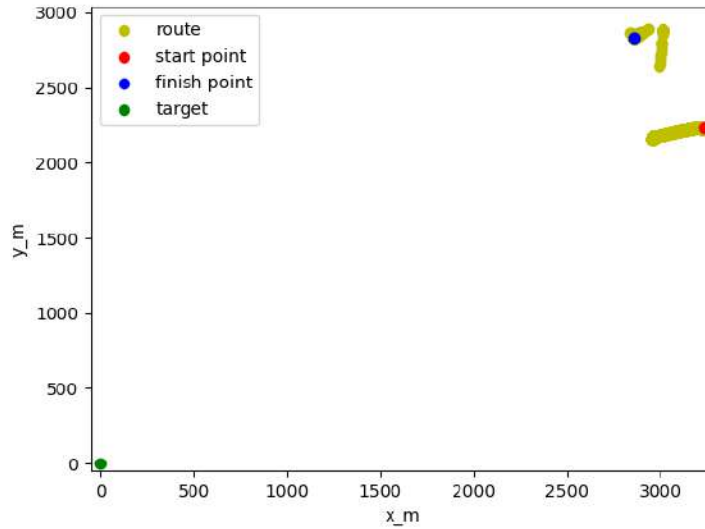
12.2. 結果

1回目の投下

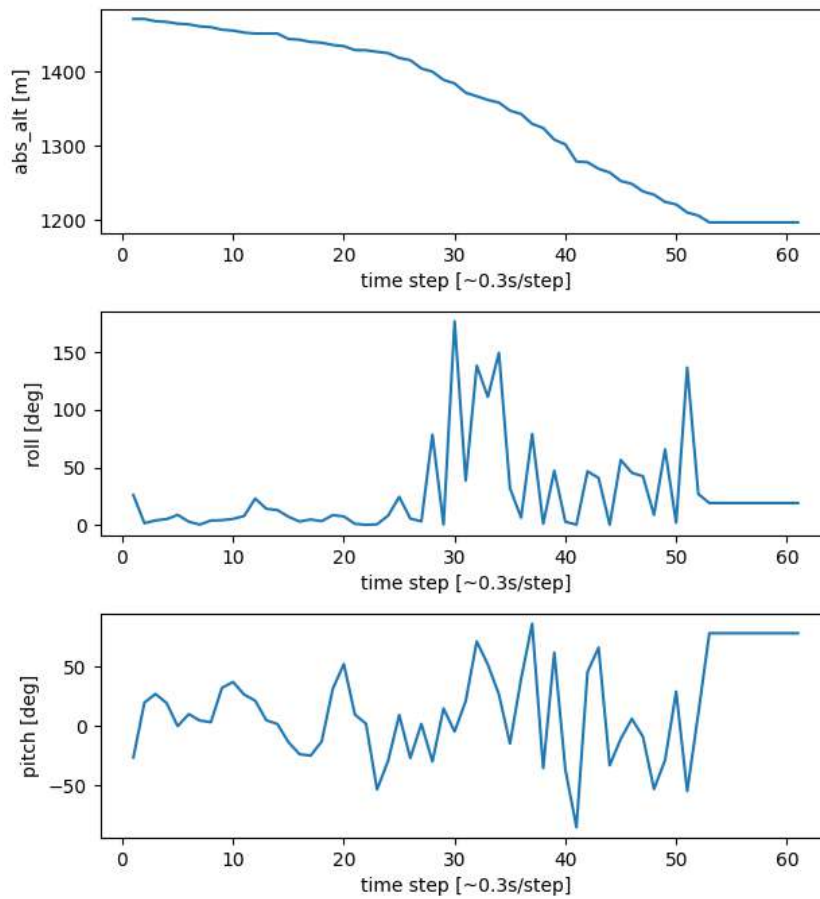
2段パラシュートの展開、パラシュートと機体の分離は成功したものの、機体は自由落下してしまった。機体を回収後、打ち上げ直前にコードが止まってしまったというログを確認した。コードが回っていなかったため、テレメトリを一度も受信ができず、機体の発見にやや時間がかかった。原因としては、打ち上げ時の衝撃で制御になっていたコンピュータへの給電が接触不良を起こし電源が切れてしまったか、地上局と機体の間のSSH接続が切断された際にコードの実行が止まってしまったことが考えられる。この失敗を受けて、マイコンとバッテリーのコネクタ付近の固定を補強し、SSH接続が切断されてもコードが実行されるようにソフトウェアも改善した。2回目の投下では同じような失敗は起こらなかった。

2回目の投下

2段パラシュートの展開と機体の分離は成功し、機体のコードも止まらずに動き続けていたものの、自由落下している状態から復帰する途中で姿勢を崩し墜落してしまった。テレメトリに関しては、loroを使ってGPS座標が地上局で受信でき、高度3000mにいる機体の位置をリアルタイムで知ることに成功した。墜落した原因としては、落下中に機体が振り子状態になり、分離した際に投げ飛ばされた形になり墜落してしまったこと、パラシュートと機体をつなぐ紐がプロペラに当たってプロペラの回転を妨げ、バランスを崩したこと、自由落下した状態から復帰するのに自重よりも大きな荷重が腕にかかった際に折れてしまったこと等が考えられるが、落下時に一部のログが消えてしまったため、原因を特定することはできなかった。



2回目の投下の位置の推移(原点はターゲット)



2回目の投下(アーム展開後)の高度とロール・ピッチ角の推移

アーム展開後5秒ほどで分離させていたこと、高度の勾配が急変している部分があることを考えると、分離直後に姿勢を大きく崩して墜落していることがわかる。

12.3. 考察

結果で示したデータからサクセスクリテリアの達成度の評価をしてください。達成できなかった項目についてなぜ達成できなかったのかを開発計画などの準備期間も含めて原因究明をしてください。原因が究明出来たら、それを解決するためには何が必要だったかを具体的に示してください。

第10章 まとめ

12.4. 工夫・努力した点(ハード, ソフト, マネジメント面すべて)

2段パラシュートにすることにより消費電力を大幅に削減した。

回路構造とともにいたるところで軽量化及び機能の維持といった工夫を凝らす努力をした。

日報を行うことにより、メンバーの進捗確認とタスク割り振りを習慣化した。

Githubを使って管理することにより、動くコードと動かないコードを分けるように努力した。

制御に使うコードをモジュール化することで、細かい単位でソフトウェアをテストできるようにした。

試験用のチェックリストを作成することで、忘れ物やケアレスミスによる失敗を防止した。また、デバッグをよりスムーズにした。

シミュレーションを使うことにより、ドローンを飛ばすことなくコードをデバッグできるようにした。

12.5. 課題点

空中で切り離されたところからの復帰をどうするか

電力問題をどうやって解決するか

どうやって本体を壊すことなく試験するか、また壊れにくい機体を作るか

12.6. 今後の展望

空中から復帰するためのアルゴリズムを改善する。

市販のオートパイロットに依存しない航法誘導制御技術を開発することによって、整備性を高める

期限を決めた開発(例:この日までにできなかったらその技術は諦める)を心がける。

長期的な計画を立て逆算したタスク振り・目標設定を行う。

この資料を見る後輩たちに向けたCanSat開発における注意点やアドバイスをお願いします。