

ARLISS大会報告・技術詳細報告書

提出日：2019年 9月 30日

文責：中理 怡恒

- チーム情報

CanSatチーム名	電気通信大学 Tadpole
CanSatチーム 代表者	中理 怡恒 iko0528@cas.lab.uec.ac.jp, 080-4324-5583
UNISEC団体名	電気通信大学 高玉研究室
UNISEC団体 学生代表	前川 佳幹 yoshimiki.maekawa@uec.ac.jp, 090-2903-5615
責任教員	高玉 圭樹 keiki@inf.uec.ac.jp, 042-443-5808
CanSatクラス	Open Class

- メンバー

役割	名前（学年）
リーダー/回路	中理怡恒（M1）
ソフトウェア	北島瑛貴（M2）
ソフトウェア	小林亮太（M2）
ハードウェア	梶原奨（M1）
ハードウェア	福本有季子（B4）
ハードウェア	金子颯汰（B2）

- CanSatの製作目的・大会参加理由

本研究室では、エンジニアとしての開発技術の向上や課題解決能力の向上、チームマネジメントなどの教育を目的としてCanSatの開発をしARLISSに参加をしている。そのため、ARLISSへ向けたEnd-to-end試験や本番の予行として貴重な機会であるため、本大会に参加をしている。

目次

ミッションについて	4
ミッションの意義と目的	4
ミッションシーケンス	4
サクセスクライテリア	7
要求項目の設定（各チームごとに項目を追加してください）	9
システム要求（安全確保のために満たすべき要求）	9
ミッション要求（ミッションを実現するためのシステム要求）	9
システム仕様	11
CanSat設計図（CAD図面や回路レイアウト図や写真など、公開可能であれば図面をここに示し、データも添付をお願いします。）	11
CanSat外観/質量/サイズ	13
CanSat内観・機構/電力	15
使用部品	19
電子系	19
動力系	20
構造系	20
製作時に使用した機材・サービス	21
プログラム・アルゴリズム	21
会計	23
試験項目設定（項目別試験、結合試験、EndtoEnd試験）	24
実施試験内容	25
工程管理、ガントチャート（スプレッドシートを推奨）	63
チーム内・審査会等	63
各担当（ハード・ソフト・全体などの進行状況・予定を記入）	63
大会結果	64
能代宇宙イベント	64
目的	64
結果	64
取得データ	64
故障原因解析・解決手段等	66
ARLISS（上記能代と同様に記載してください）	66
目的	66
結果	66

取得データ	67
故障原因解析・解決手段等	71
まとめ	73
工夫・努力した点（ハード、ソフト、マネジメント面すべて）	73
良かった点・課題点	73
チームのマネジメント等、プロジェクト全体でのよかった点、反省点	74

第1章 ミッションについて

1. ミッションの意義と目的

近年、ヘビ型ロボットが盛んに研究されており、レスキュー現場において実用化が進んでいる。ヘビ型ロボットの特徴である多関節機構の形状と姿勢の自由度の高さから、悪路に対する走破性、地面の掘削を行うなど様々な能力を1つの機構によって実現できる可能性を有している。

その一方で、ヘビ型ロボットは移動速度が遅く長距離走行に難がある。そこで、我々は精密制御が可能な多関節機構を有したCanSatに、牽引・掘削能力を持たせ、さらに従来の2輪型CanSatをかけ合わせることで、任意の地点に移動するための長距離走行能力を備えた多関節×2輪型CanSatを提案する。この多関節×2輪型CanSatは走行能力と掘削能力を両立し、任意の地点へ移動しを掘削することが可能な新しい探査機である。

本ミッションでは、多関節機構が轍に対する走破性と地面の掘削に有効であることを検証するために、未知の惑星環境に似た地形をもつアメリカのBlack Rock砂漠で実際に多関節機構を用いて轍からの脱出、地面の掘削をする。また、当大会のComeback Competitionのゴールを目指すことで、従来の2輪型CanSatがもつ長距離走行能力が失われていないことを確認する。さらに、ゴール後に追加実験として故障したCanSatを牽引可能か検証することで2輪×多関節CanSatの可能性を追及する。

2. ミッションシーケンス

Fig. 1.2.1にミッションシーケンスの概要を表す。

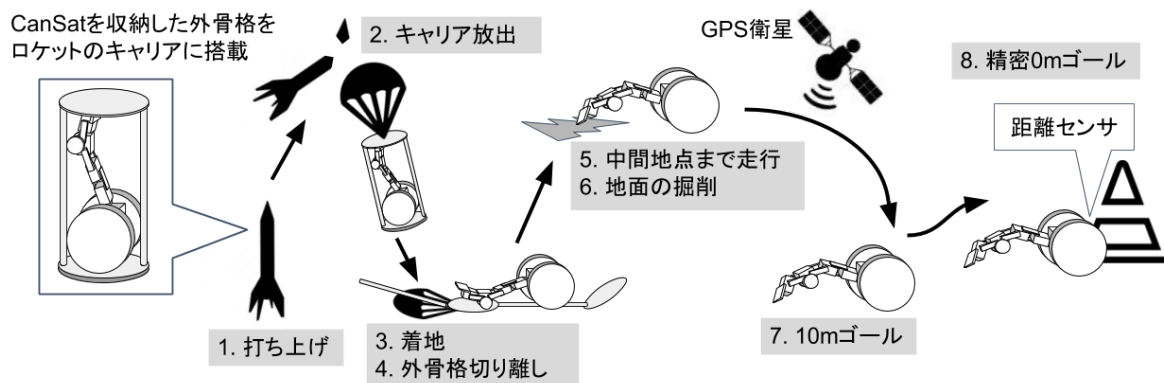


Fig. 1.2.1 ミッションシーケンスの概要図

以下にミッションシーケンスの詳細を示す。

1. 打ち上げ

CanSatを収納した外骨格をロケットに搭載し、打ち上げる。ここで、外骨格とはCanSatを衝撃から保護する専用のケースである。4.2節で詳細を述べる。

2. キャリア放出

空中でロケットのキャリアから外骨格が放出され、パラシュートで降下する。

3. 着地

- 外骨格が地上に着陸する。
4. 外骨格切り離し
外骨格の接合部分を内部からCanSatが分離することで、CanSatは外骨格から切り離される。なお、パラシュートは外骨格に固定されているため、同時にパラシュートもCanSatから切り離される。
 5. 中間地点まで走行
GPSを用いてCanSatが自己位置推定し、目標の座標と自己位置の座標の差分を用いて走行制御することで、当大会のComeback Competitionのゴール中間地点まで走行する。
 6. 地面の掘削
CanSatの多関節機構を用いて、地面の掘削を開始する。CanSatに搭載している加速度センサの値からCanSatの傾きを算出し、多関節機構の長さから掘削の深度を推定する。5cm以上の掘削を検知するまで掘削を続ける。
 7. 10mゴール
5と同様にGPSを用いて中間地点からゴール地点まで走行し、ゴールから10m以内に到達する。
 8. 精密0mゴール
GPSを用いて、自己位置推定がゴールから10m以内に到着したと判断した後、距離センサを用いた精密ゴール走行に移行する。ゴール位置の推定に成功した場合、距離センサを用いてゴールに接近することで0mゴールを達成する。

第2章 サクセスクライテリア

	内容		評価方法	
項目	走破性能	多関節	走破性能	多関節
ミニマムサクセス	ロケットから放出されたCanSatに破損なく、着地した後、CanSatを着地衝撃から保護するための外骨格（4章で説明）から脱出し走行開始する。		着地後、外骨格の切り離しに成功し、走行可能である。また、掘削するための機構が破損していない。	
ミドルサクセス	GPSセンサを用いたナビゲーションシステムにより、着地地点とゴール地点との中間地点まで走行する。	中間地点付近にて、地表から5cm以上の深さを多関節機構を用いて掘削する。	着地地点とゴール地点との中間地点まで走行することができたか。	着地地点とゴール地点との中間地点において、地表から5cm以上の深さを多関節機構を用いて掘削できたか。
フルサクセス	ミドルサクセスを達成後、ゴール地点から10m圏内に到達する。また、走行中に轍などによってスタックしてしまった場合は多関節機構を利用し脱出する。		ゴール地点から10m圏内に到達できたか。走行中にスタックしなかった場合、追加実験として投下したCanSatを用いて轍走行試験を実施し検証する。	
アドバンスドサクセス	フルサクセスを達成した後、距離セ	ゴール後、遠隔操作を想定し、ゴー	0mゴールできたか	故障CanSatをゴールから500m離れ

	ンサを用いた正確なゴール位置を推定するアルゴリズムを利用して、0mゴールをする.	ル付近に置かれた故障CanSatを500m牽引する.		た地点に牽引することができたか.
--	--	----------------------------	--	------------------

なお、走破性能、多関節の各項目ごとに評価し、全体では、どちらかの項目が成功した場合、成功したとする.

第3章 要求項目の設定（各チームごとに項目を追加してください）

1. システム要求（安全確保のために満たすべき要求）

要求番号	自己審査項目（ ARLISS打ち上げ安全基準 ）
S1	質量と容積がレギュレーションを満たすことが確認できている
S2	ロスト対策を実施しており、有効性が試験で確認できている
S3	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
S4	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が試験で確認できている
S5	打ち上げ時の振動荷重によって、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	ロケット分離時の衝撃荷重によって、安全基準を充足するための機能が損なわれていないことが試験で確認できている
S7	パラシュートが開傘する時の衝撃荷重によって、安全基準を充足するための機能が損なわれないことが試験で確認できている
S8	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
S9	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
S10	S1-S9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試験を実施できしており、今後、安全性に関わる大幅な設計変更はない

2. ミッション要求（ミッションを実現するためのシステム要求）

要求番号	自己審査項目（ ミッション実現要求項目 ）
M1	外骨格が着陸時の衝撃荷重によって、CanSatを保護することができ、ミッションを実現するための機能が損なわれないことが試験で確認できている
M2	CanSatが着地衝撃から守るための外骨格から脱出可能であることを試験で確認できている
M3	多関節機構を用いた轍に対してのロバスト性を試験で確認できている

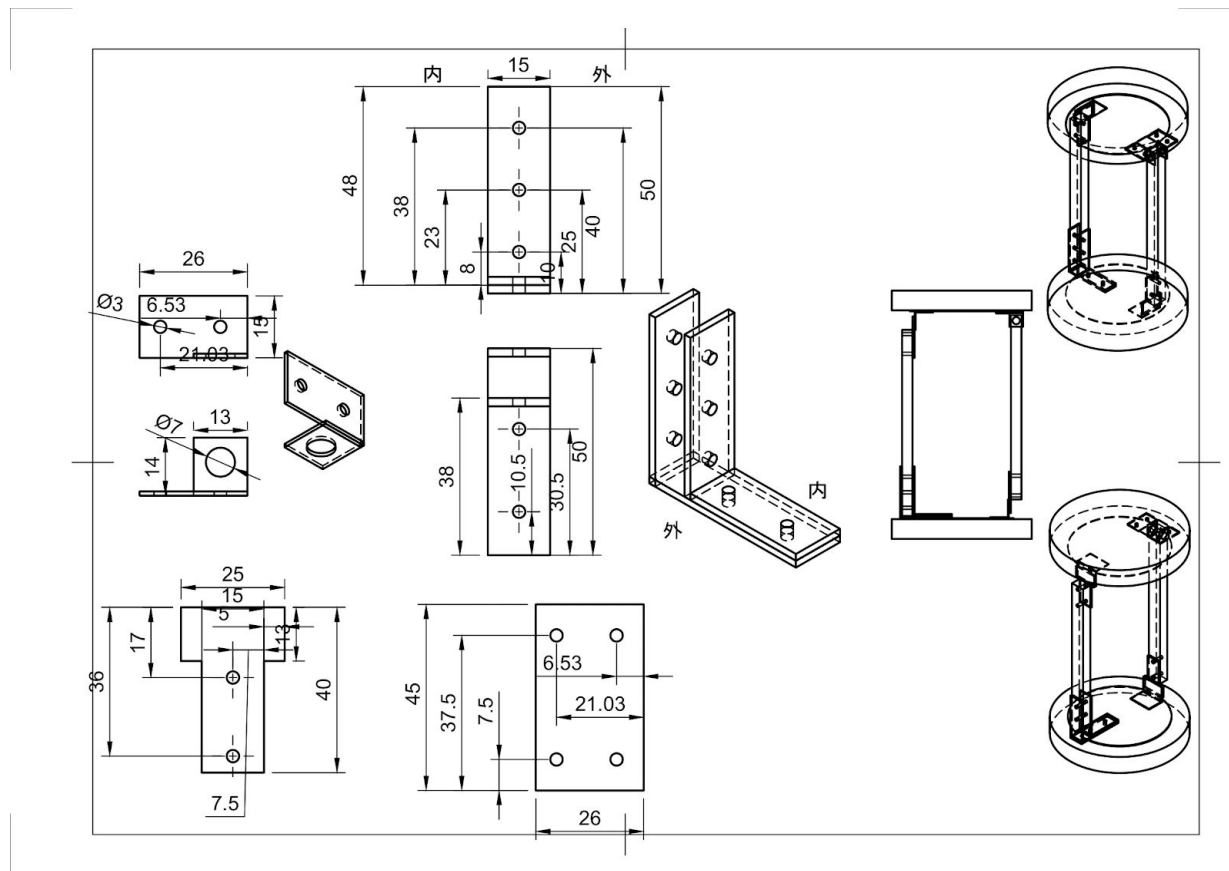
M4	CanSatが反転・横転した際に復帰できることを試験で確認できている
M5	多関節機構を用いて地面を掘削できることを試験で確認できている
M6	多関節機構を用いて地面を牽引できることを試験で確認できている
M7	ミッションを実現するために十分なバッテリーがあることを試験で確認できている
M8	着地地点からゴール地点へ走行するためのナビゲーションシステムを有することが試験を通じて確認できている
M9	ゴール地点に設置されたコーンを検知し、0mゴールすることができることを確認できている
M10	ミッション時に人間が介在しない自律制御を実施することが確認できている
M11	ミッション後、規定された制御履歴レポートを運営者へ提出する準備ができている

第4章 システム仕様

1. CanSat設計図

(1) ハード系

以下に公開可能な各モジュールの設計図を示す.

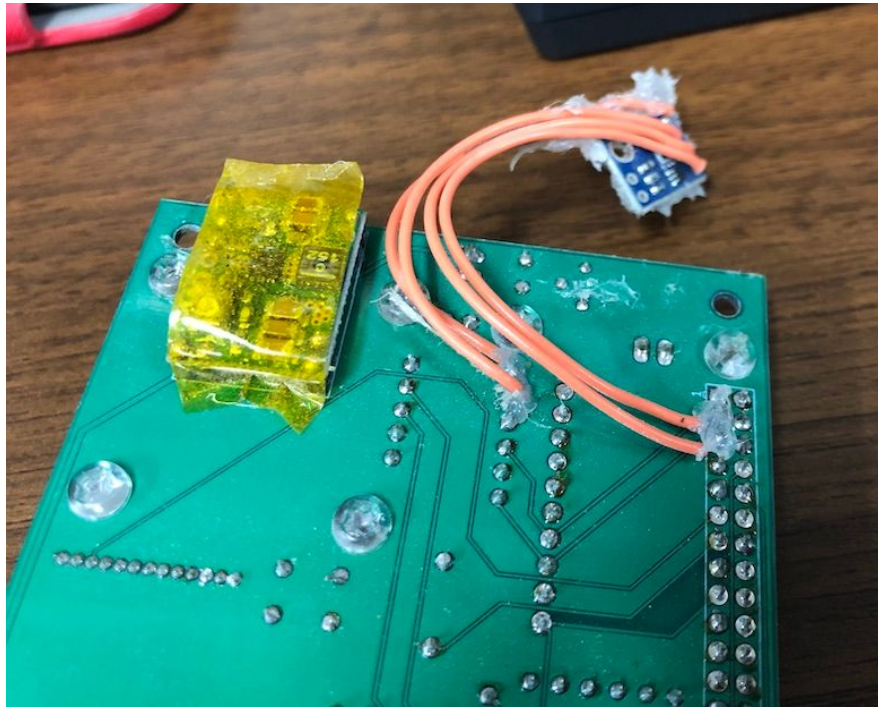


外骨格図面

(2) 回路系

以下に回路の配線と実際のレイアウト図を示す.

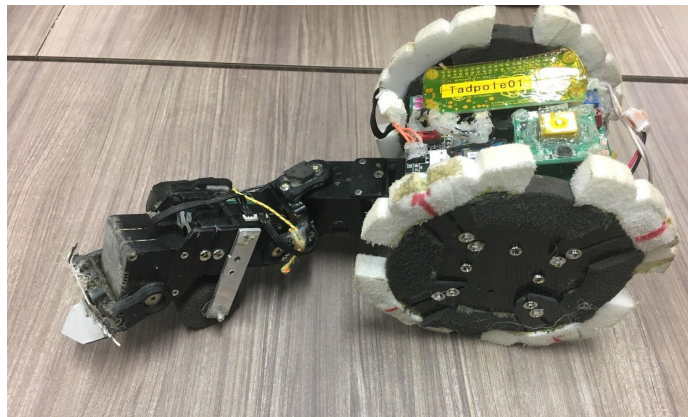
実装することが困難であったためである。よって、下の写真のように、後から実装した。



後から距離センサを実装した際の外見

2. CanSat外観/質量/サイズ

CanSatの全体像をFig. 4.2.1に示す。本CanSatは2輪と多関節機構を持つため縦長のCanSatであることから、CanSatの横幅はCanSatの直径内に収める必要がある。その実現のため、拡張タイヤと外骨格を用いている。それぞれの詳細については以下の項目(拡張タイヤ機構、外骨格構造)で述べる。



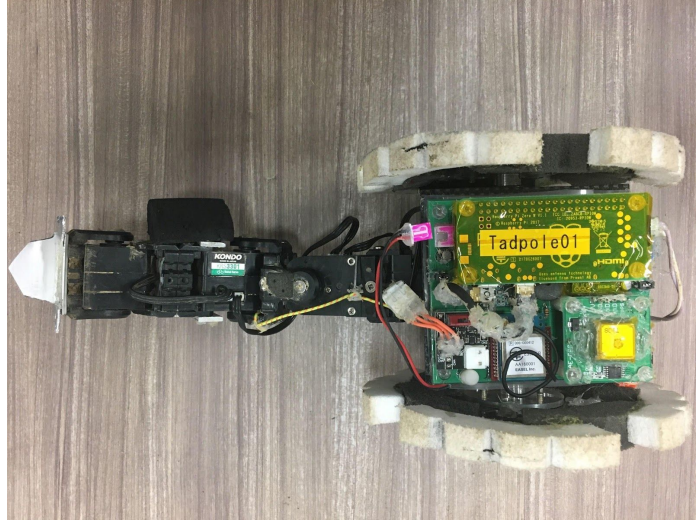


Fig. 4.2.1 CanSatの外観 側面(上図) 上面(下図)
CanSat+外骨格+パラシュートの総質量は1047gである。また、パラシュートは外骨格に付いているため切り離す必要はない。以下、表4.2.1に各機構の質量を示す。

表4.2.1 各機構の質量

CanSat	外骨格	パラシュート	総計
705g	270g	72g	1047g

キャリアにはFig4.2.2のような状態で外骨格にCanSatを収納する。さらにパラシュートの収納方法は以下の動画に示す。

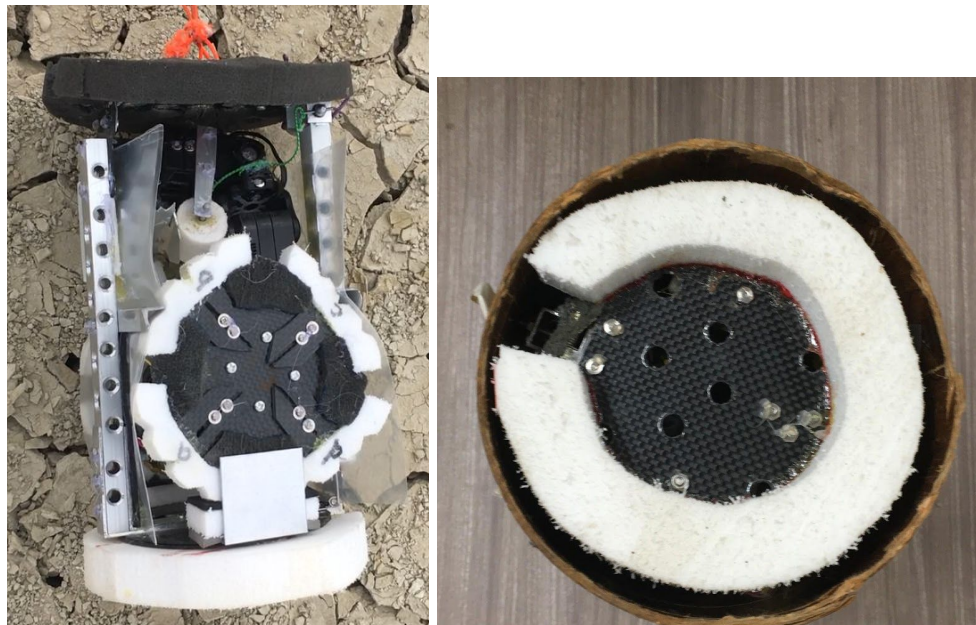


Fig 4.2.2 CanSatを収納した様子 (左：横面 右：上面)

パラシュートの収納動画：

<https://www.youtube.com/watch?v=RQcR0EbI6lk&feature=youtu.be>

3. CanSat内観・機構/電力

□ CanSat基本構造

シャーシにモータ1つ・歯車・シャフト・電池・回路基板が設置されており、電池はRaspberry Pi用の電源としてリチウムポリマー電池、モータ+サーボモータ用の電源としてエナジャイザーを使用している。モータを1つにすることでCanSatの重量とモータによる占有スペースを抑えることができる。しかし、モータ1つでは方向転換ができないため、多関節機構を左右に曲げることで方向制御を行う。詳細は以下の項目「多関節機構」で述べる。なおCanSatの横幅を最小限するためにモータからシャフトへ歯車4つを用いて駆動伝達をする。(Fig. 4.3.1)

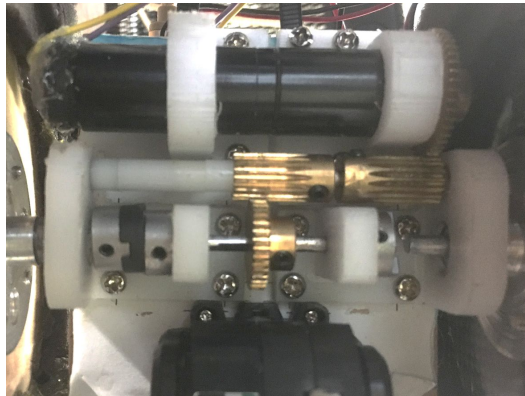


Fig. 4.3.1 モータと歯車の構成

- 多関節機構

多関節機構の外装を以下のFig. 4.3.2に示す。

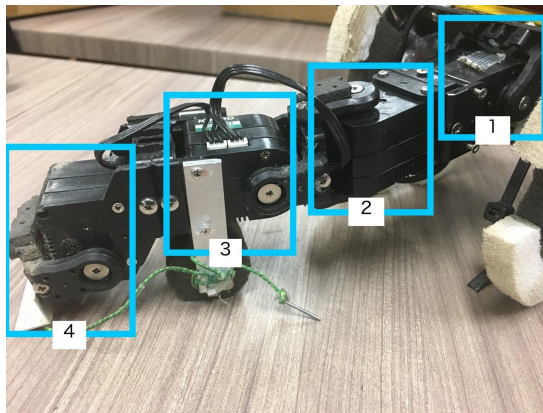


Fig. 4.3.2 多関節機構

多関節機構はCanSat本体後部に接続した4つのサーボモータから構成される。このうち本体に近いものから順に1, 3, 4番目のサーボモータは上下方向に駆動する。4番目のサーボモータでスタック時に地面を押す動作や掘削を行い、1, 3番目のサーボモータで4番目のサーボモータの地面に対する角度を調節する。また、2番目に接続されているサーボモータは左右方向に駆動する。この左右に駆動するサーボモータによって、左右の旋回機構を実現する。サーボモータを左右に曲げている様子を以下、Fig. 4.3.3に示す。サーボモータを縦方向に接続したままの状態では全長が長くなりキャリアに収納不可能なため、3, 4番目のサーボモータを内側に折り曲げることで全長を短くしキャリアへの収納を可能にする。また、多関節機構の末端にL字型のアルミ板(Fig. 4.3.4)を取り付けることで掘削を行いやすくする。

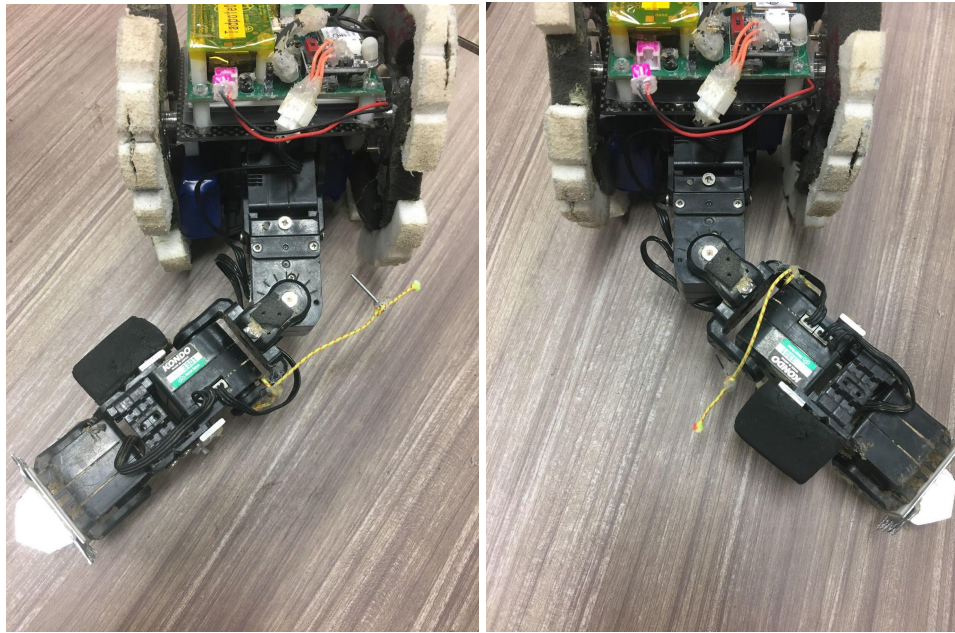


Fig. 4.3.3 サーボモータを左右に曲げた様子



Fig. 4.3.4 L字型のアルミ板

- 拡張タイヤ機構

CanSatのタイヤは、タイヤ径の大きさに比例して走破性が向上する。さらに走行時に回路や電池が地面と擦れてしまうことを防ぐためにも、タイヤ径の確保は必要である。しかし、キャリアの直径内にCanSatの横幅を収めようとする、それに合わせてタイヤ径を小さくせざるを得ない。そこで本CanSatのタイヤは、タイヤが径が収納時には100mm、走行時は125mmとなるような拡張タイヤとなっている。以下、Fig. 4.3.5に拡張タイヤの機構を示す。

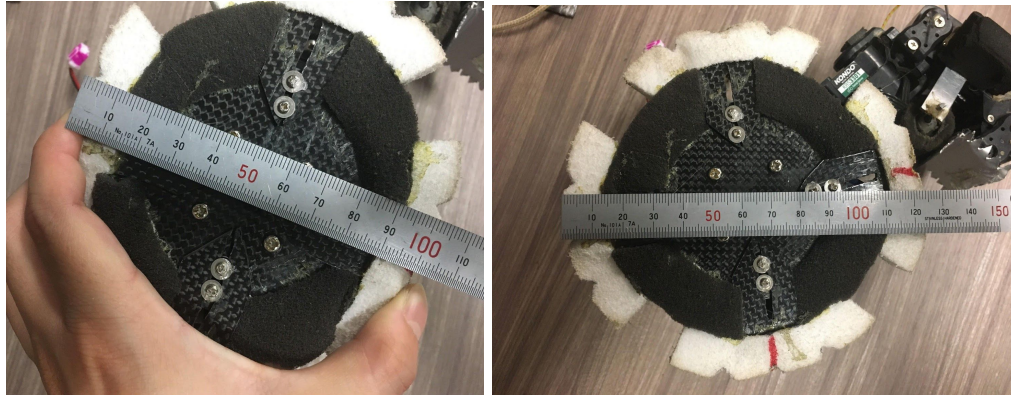


Fig.4.3.5 拡張タイヤ(右:収納時 左:走行時)

タイヤ表面の外側の白いスポンジが4つに分離しさらに切り込みが設けられていることによって、表面に凹凸の少ないタイヤよりも地面を捉えやすくなり空転を防ぐことで走破性の向上が期待される。

- 外骨格構造

外骨格は 開傘時・着地時に直接的にCanSat、特に多関節機構に力が加わることを防ぎ多関節機構の損傷を防ぐために用いられる。また上面にパラシュートを繋げるための紐がついている。Fig. 4.3.6に外骨格の外観を示す。外骨格は上下面のスポンジ+カーボンと側面の2本の支柱で構成されており、2本の支柱のうち一方は固定ピンと分離ピンで固定されている(Fig. 4.3.7)。2本の支柱の中央付近に付いている横棒は、CanSatのタイヤを固定するためであり前述の拡張タイヤをこの部分で押さえつけて収納する。底面に付いているL字型のパーツはCanSatが外骨格内で横に滑り抜け出してしまうことを防ぐものである。また片方の支柱に付いている白い板のパーツは外骨格から多関節機構が外骨格外部へ飛び出ること防ぐためのものである。着地判定後、多関節機構のサーボモータを曲げ、分離ピンを引き抜くことで、外骨格が切り開く。



Fig. 4.3.6 外骨格外観(左:閉じた状態 右:開いた状態)

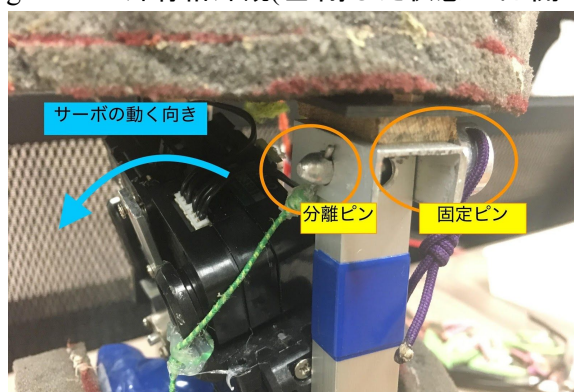


Fig. 4.3.7 外骨格脱出機構

□ システム図

システム図をFig. 4.6.1に示す。ミッションシーケンスのフェーズにおいて、GPSセンサ、9軸センサ（加速度、ジャイロ、地磁気）、気圧センサ、光センサ、距離センサから得られる値を入力としてCanSatの状態を判定し、モータ、サーボモータ、LED、ブザーを制御し行動する。

本ミッションでは、環境情報を取得する各センサ（入力）について、以下のように用いる。

- 距離センサ
キャリア放出判定・ゴール検知
- 光センサ
キャリア放出判定
- GPSセンサ
自己位置推定、ナビゲーション処理、スタック判定
- 9軸センサ
着地判定、スタック判定、ナビゲーション処理、横転・反転判定

- 気圧センサ
着地判定

また、ロスト対策の為に、GPSが取得した自己位置情報を定期的に無線モジュールLoRaを用いて定期的に発信する。

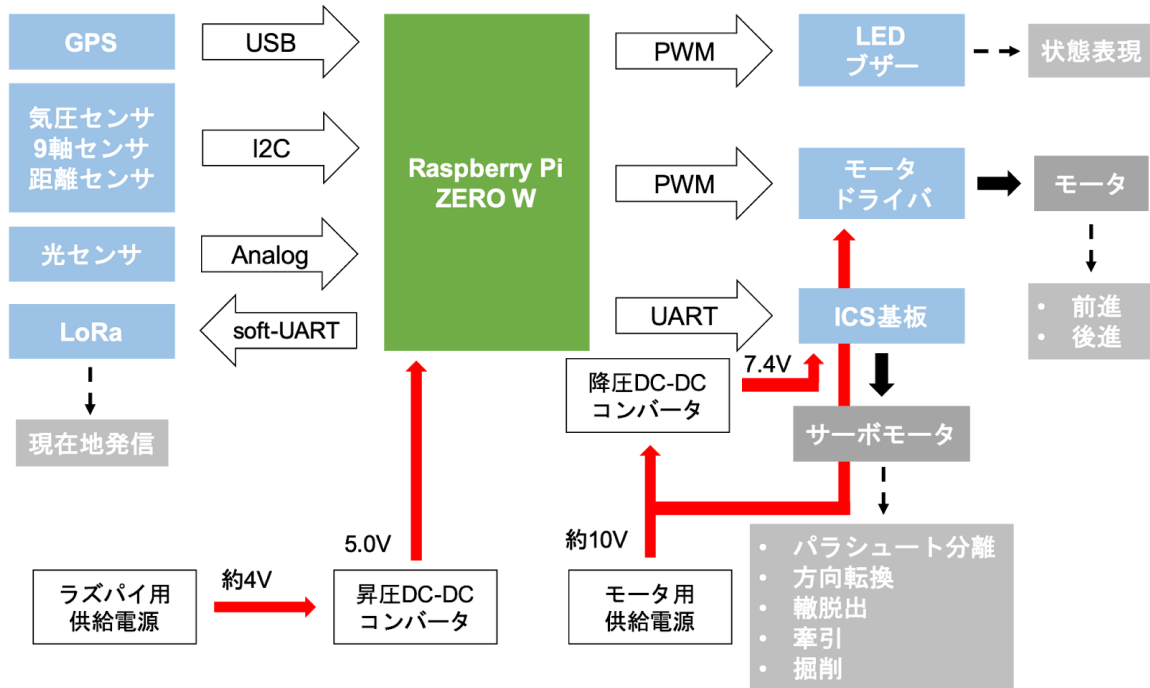


Fig. 4.3.8 システム図

□ 電力

- モータ用電池(Energizer AA: 6本)
 - 打ち上げ前: 10.57V
 - 打ち上げ後: 記録なし
- 回路用電池(リポ電池)
 - 打ち上げ前: 4.27V
 - 打ち上げ後: 記録なし

上記のように、打ち上げ後のバッテリー残量を計測していなかったため、バッテリーの使用状況は6章の17. 電力耐久試験を参照されたい。

4. 使用部品

□ 電子系

分類	名称・型番	入手先・参考情報等	備考
マイコン	Raspberry Pi zero W	http://akizukidenshi.com/catalog/g/gM-12958/	
センサ	GPSモジュール・GMS-G9, PA6H	http://akizukidenshi.com/catalog/g/gK-09991/	
信号変換	超小型USBシリアル変換モジュール・FT234X	http://akizukidenshi.com/catalog/g/gM-08461/	GPS情報を変換

センサ	気圧センサ・BME280	http://akizukidenshi.com/catalog/g/gK-09421/	
センサ	9軸センサ・Hiletgo MPU9250	https://www.amazon.co.jp/dp/B0154PM102/ref=cm_sw_em_r_mt_dp_U_Mzp.CbAD0E5HF	
センサ	レーザ距離センサ・KKHMF GY-530 VL53L0X	https://www.amazon.co.jp/dp/B07FBFPRYF/ref=cm_sw_r_cp_api_i_TQN-CbVKW33EZ	
センサ	cdsセル・I-00110	http://akizukidenshi.com/catalog/g/gI-00110/	
抵抗	可変抵抗・E104	http://akizukidenshi.com/catalog/g/gP-03283/	光検知量の調整
状態把握	LEDフルカラー 5mm・I-02476	http://akizukidenshi.com/catalog/g/gI-02476/	
状態把握	ブザー・PB04-SE12HPR	http://akizukidenshi.com/catalog/g/gP-04497/	
制御	ステッピング&DCモータードライバ・DRV8835	http://akizukidenshi.com/catalog/g/gK-09848/	モータ用
コンバータ	可変型昇降圧DC-DCコンバータモジュール(2.5V~9V)・TPS63070	https://strawberry-linux.com/catalog/items?code=16370	サーボモータ用
コンバータ	昇圧型DC-DCコンバータモジュール(5V 2.5A)・TPS61230	https://strawberry-linux.com/catalog/items?code=12060	ラズパイ電源供給用
信号変換	ICS変換基板・03121	https://kondo-robot.com/product/03121	サーボ制御用
無線モジュール	ES920LRSDK1	http://easel5.com/download/es920lr/ES920LR%E3%83%87%E3%83%BC%E3%82%BF%E3%82%B7%E3%83%BC%E3%83%88_1.03.pdf	
バッテリー	Energizer Ultimate Lithium AA Batteries	https://www.energizer.com/batteries/energizer-ultimate-lithium-batteries	モータ，サーボモータ用電源・6本使用
バッテリー	リチウムイオンポリマー電池3.7V2000mAh	https://www.marutsu.co.jp/pc/i/836419/?gclid=CjwKCAjwibzsBRA MEiwA1pHZrUlcANwJMCs8WzcSHG3qU17tmgl-1GNwGPy_wYyS7MmN-Q2eBvfz_xoCWzMQAvD_BwE	回路用電源

□ 動力系

分類	名称・型番	入手先・参考情報等	備考
タイヤ駆動用モータ	マクソンDCモータ・253297(モータ118684, ギアヘッド110323)	https://www.maxongroup.co.jp/maxon/view/service_search?query=253297	
多関節機構用サーボモータ	ICSサーボ・KRS-3301	https://kondo-robot.com/product/krs-3301-ics	
多関節機構	ICSサーボ・KRS-3304	https://kondo-robot.com/product/krs-3304-ics	ARLISSの2回目の打ち

の2番目の サーボモータ		rs-3304-ics	上げ以降に使用
-----------------	--	-----------------------------	---------

□ 構造系

分類	名称・型番	入手先・参考情報等	備考
タイヤホイール	CFRP/カーボン板 2.0×100×330mm	ABCホビー	
タイヤ用スポンジ	サンペルカ ポリエチレンフォーム	東急ハンズ新宿店	
シャーシ	CFRP/カーボン板 1.5×100×330mm	ABCホビー	
モータホルダ	POM板 25×300×300	https://www.cutpla.com/	
車軸	ジュラルミン丸棒 A2017 φ40×38L	http://www.tokukin.com/01sozai/02alum_free.html	
カップリング	アサ電子工業ASJ12-3-4	Misumi	
ベアリング	688-H-ZZ	Misumi	
ギア	S50B45B+0203 S50B32B+0303 S50B16K+0803	Misumi	
シャフト	泰豊 ステンレス丸棒 3mm×1.05m	東急ハンズ新宿店	
外骨格用スポンジ	サンペルカ ポリエチレンフォーム	東急ハンズ新宿店	
外骨格支柱	アルミ角パイプシルバー 1.2×1000×15mm	モノタロウ	
外骨格上面底面	CFRP/カーボン板 2.0×100×330mm	ABCホビー	
外骨格蝶番	uxcell スプリングロード蝶番 25×33×0.8mm	Amazon	

5. 製作時に使用した機材・サービス

分類	名称・型番	入手先・参考情報等	備考
特殊機材	3D切削機	電通大施設	
特殊機材	フライス盤	電通大施設	
特殊機材	旋盤	電通大施設	
特殊機材	ボール盤	電通大施設	
特殊機材	コンターマシン	電通大施設	
外部サービス	プリント基板	Elecrow	回路設計後、ユニバーサル基板で実装後にデバッグをして、発注

6. プログラム・アルゴリズム

使用言語: C++

Fig. 6.1にアルゴリズムのフローチャートを示す.

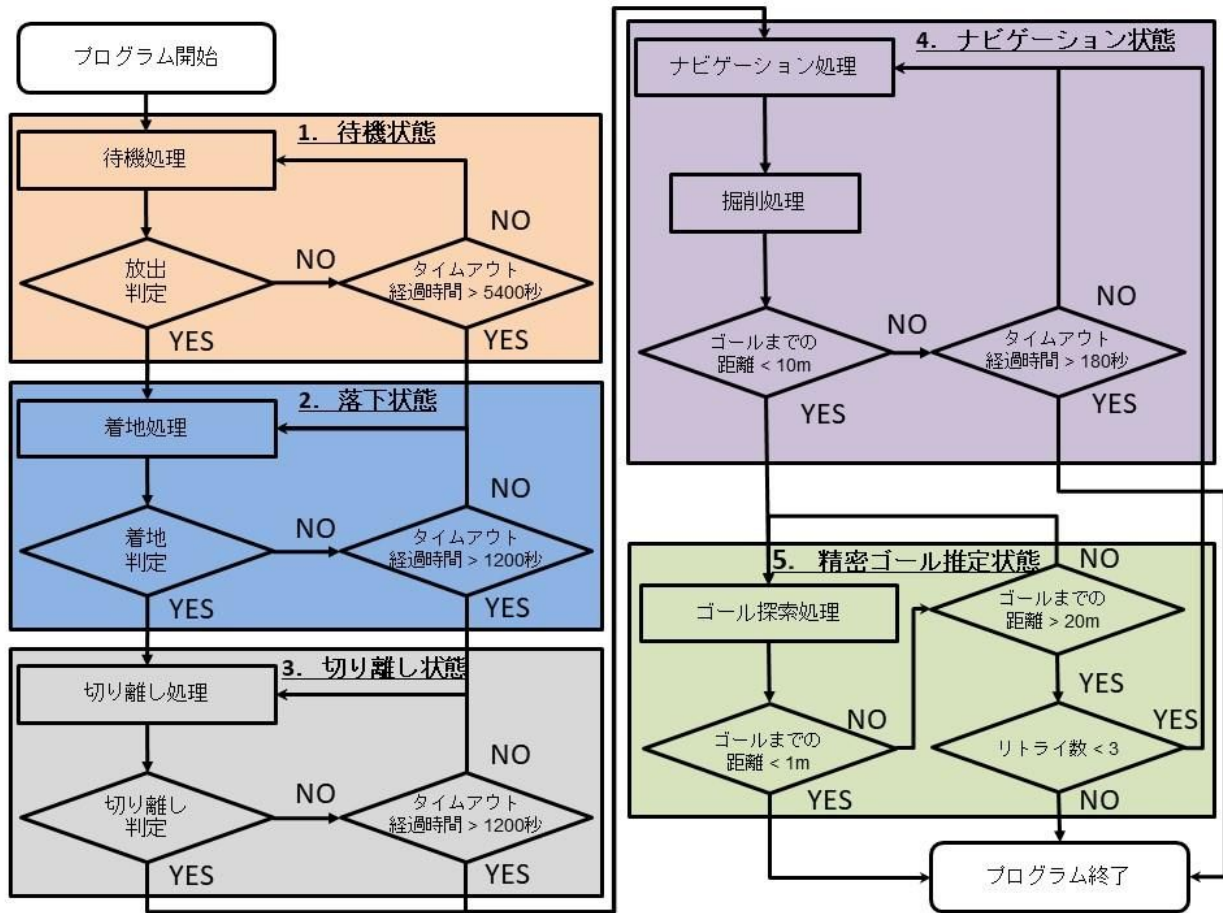


Fig. 6.1 アルゴリズムのフローチャート

アルゴリズムは以下の5つの状態を持つ.

1. 待機状態
2. 落下状態
3. 切り離し状態
4. ナビゲーション状態
5. 精密ゴール推定状態

以下に詳細を示す. なお, どの状態においても適切なタイムアウトが設定されており, センサの故障など予想外のアクシデントにより, 状態移行のための判定条件に失敗しつづけたとしても, 次の状態に移行する.

1. 待機状態

プログラムが開始するとこの状態に遷移する. この状態中, CanSatはロケット内で待機していることを想定し, キャリアから放出されているか否かの放出判定をする. 判定は光センサとCansatの進行方向に設置された距離センサの出力値が閾値を一定時間超える場合, CanSatが放出され, パラシュートが開いた状態と判定する. 放出判定後, CanSatはキャリアから放出され空中落下している状態と判断し, プログラムは次の状態である落下状態に移行する. 放出判定されなかった場合でもタイムアウト判定(待機状

態を開始してから90分)により、落下状態に移行する。

2. 落下状態

この状態中にCanSatはキャリアから放出され空中落下していることを想定し、地面に着地しているか否かの着地判定をする。監視された加速度センサと気圧センサの値の変位が十分に小さい場合に着地と判定する。着地判定後、CanSatは外骨格およびパラシュートの切り離しのため、切り離し状態に移行する。また、着地判定に失敗しつづけた場合でもタイムアウト(落下状態を開始してから20分)により、切り離し状態に移行する。

3. 切り離し状態

この状態中ではCanSatは着地している状態を想定し、外骨格をCanSatから切り離す処理をする。なお、パラシュートは外骨格に固定されている。CanSat本体に取り付けられた分離ピンをサーボモータにより引き抜くことで、外骨格との切り離しをする。引き抜き動作後、閉じた外骨格の蓋をサーボモータを伸ばすことで開く。モータの回転およびサーボモータの伸縮を行い、起き上がり動作による外骨格との接触を回避する動作を繰り返す。最後に安全な走行に向け、ゆっくりモータを動かし外骨格の位置を距離センサで特定後、物体検知ができなくなった方向に向け一定時間走行する。以上の切り離し処理の完了後、外骨格が解体されたかどうかの外骨格切り離し判定をする。折りたたんでいた3つ目のサーボを伸ばすようにコマンド出力をし、一定時間後に同サーボモータの現角度値を確認する。この時の値が先ほどのコマンド出力値と一致していた場合、CanSatは外骨格の切り離しが完了したと判断し、ゴール地点への走行のため、ナビゲーション状態に移行する。もし、外骨格切り離し判定に失敗し続けた場合、タイムアウト処理(切り離し状態を開始してから20分)により、ナビゲーション状態に移行する。

4. ナビゲーション状態

この状態中にCanSatは、ゴール地点に向けて走行することを想定する。CanSatはGPSを用いて自己位置推定し、予め入力されたゴール地点のGPS座標に向かって走行制御する。その際、直近の観測から過去10サンプルほどプールしたCanSatの自己座標の集合から求めたCanSatの方向ベクトルと、直近に観測した自己座標とゴール地点の座標から求めた目標ベクトルを用いる。つまり、方向ベクトルを目標ベクトルに近づけるための修正を方向制御用のサーボモータの回転角を調整することで行う。その際、なるべく負荷のかからない修正を実現するためPID制御する。一定距離走行後、サーボモータを使ってゴール方向を示す簡単な目印を地面に掘削する。自己座標とゴール座標の距離が10m以内であれば、CanSatはゴールしたと判断し、GPSゴール判定する。その後、0mゴールのため精密ゴール推定状態に移行する。もし、GPSゴール判定を長時間しなかった場合、タイムアウト判定(ナビゲーション状態を開始してから180分)により、強制的にゴール判定をし、プログラムを終了する。この時間は電力耐久試験により得られた時間であり、その妥当性は6章.17を参照されたい。

5. 精密ゴール状態

この状態中にCanSatは、その場で後輪機構のサーボを左右どちらかに曲げて前進して回転行動をとる。その際、CanSatの前方に取り付けられた距離センサを用いて、CanSatは0mゴールしているかを判定をする。回転行動中、徐々にサーボの角度を開くことで、段々と軌跡の円の半径を長くしていきゴールの探索範囲を広げる。もし、距離センサが反応すれば、回転をやめ、真っすぐ前進することで0mゴールを試みる。0mゴールの失敗は自己座標とゴール座標の距離が20m以上になった場合とする。0mゴールに失敗した場合、4.のナビゲーション状態に再び移行する。規定回数失敗し続けたとき、精密ゴー

ルには移行せずゴール判定をする。

7. 会計

CanSatはARLISSにおける2回の打ち上げのための2機に加え、1機の予備機を含めた3機を開発した。CanSat3機分の部品購入費の総計を以下に示す。

ハード	回路	ソフト	総計
65,851円	69,187円	15,155円	150,193円

なお、上記の費用に加えてモータ代金は1機あたり21,330円であり、2機分購入した。型番はmaxon DC motor 253297である。

第5章 試験項目設定（項目別試験、結合試験、EndtoEnd試験）

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）	実施日
V1	質量試験	S1（R1）	7/20
V2	キャリア収納試験	S1, M2（R1）	7/20
V3	通信距離試験	S2（R2）	7/16
V4	落下試験	S3（R3）	7/3
V5	静荷重試験	S4（R4）	7/7
V6	振動試験	S5（R5）	8/2
V7	分離衝撃試験	S6（R6）	8/2
V8	開傘衝撃試験	S7（R3）	7/19
V9	通信機電源ON/OFF試験	S8（R7）	8/15
V10	通信周波数変更試験	S9（R8）	8/20
V11	着地衝撃試験	M1	7/10, 20, 21
V12	外骨格脱出試験	M2	7/21
V13	走行試験	M3（R12）	7/20, ARLISS現地 にて追加で 実施

V14	反転・横転復帰試験	M4	7/20
V15	掘削性能試験	M5	7/21
V16	牽引性能試験	M6	8/21
V17	電力耐久試験	M7	8/22
V18	ナビゲーション試験	M8 (R10)	7/21, ARLISS現地 にて追加で 実施
V19	ゴール検知試験	M9	7/21
V20	End-to-end試験	S10, M1, M2, M5, M8, M9, M10 (R9 , R10)	7/21. ARLISS現地 にて追加で 実施
V21	制御履歴レポート作成試験	M11 (R11)	7/20

第6章 実施試験内容

1. 質量試験

- 目的
CanSatの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
質量計で測定を行い、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+外骨格+パラシュートの総重量は1047g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。また、Fig. 6.1.1に質量測定の結果を示す。



Fig. 6.1.1 外骨格とパラシュートの質量

- 結論

パラシュートと外骨格の質量を含めたCanSatの質量がレギュレーションを満たす.

2. キャリア収納試験

- 目的

CanSatをキャリアに収納し、その後放出可能であることを確認する.

- 試験の内容

キャリア収納を行う. メジャーを用いて高さと直径を計測し結果を写真として記録する. また, キャリアを軽く振ることでCanSatが放出可能であることを確認する.

- 結果

以下, Fig. 6.2.1, 6.2.2にキャリアに収納済みのCanSatと長さを計測した写真を載せる



Fig. 6.2.1 キャリア収納時のCanSatの様子



Fig. 6.2.2 CanSatの高さと直径（240mm, 135mm）

また表6.2.1に示すように、CanSatをキャリアに収納後、問題なく放出できた。

表6.2.1 キャリア放出実験の結果

回数	実験動画	放出判定
----	------	------

1	https://youtu.be/N0XF7uiRa2U	少し揺らすことで放出できた
2	https://youtu.be/xt5kbF2eGuY	少し揺らすことで放出できた
3	https://youtu.be/l9x6ir6e3aI	少し揺らすことで放出できた
4	https://youtu.be/S0sHlVapPas	少し揺らすことで放出できた
5	https://youtu.be/UXxarvskwq8	少し揺らすことで放出できた

- 結論

CanSatをキャリアに収納でき、その後放出可能である。

3. 通信距離試験

- 目的

ARLISS本番でCanSatのロスト対策のために、CanSatとPCの通信できる最大距離を知り、探索可能であることを確認する。

- 試験内容

障害物が少なく、見晴らしの良い多摩川の河川敷で実験をする。まず、長距離無線通信モジュールLoRa（ES920LR）を繋いだPCを地上局とし、CanSat側から地上局へCanSatの位置情報をLoRaで発信し、PC側がその位置情報を受信できることを確認する。次に、地上局の位置を固定し、CanSatを持った人が地上局から離れていき、地上局側がCanSatから位置情報を受信できなくなる地点で両者のGPS座標を取得し、2点間の距離を通信可能距離とする。なお、同団体の別チームであるSiriusuチームのCanSatの回路に搭載されたLoRaモジュールの配置が弊チームのCanSatの回路と類似しているため、合同で実験を実施した。

- 結果

以下のような結果になった。地上局と通信限界時のCanSatの位置を地図上に出力したものをFig. 6.3.1に示す。

地上局の座標： 35.642996, 139.524233
 通信が途絶えた座標： 35.633461, 139.556343
 通信可能距離： 3.1km

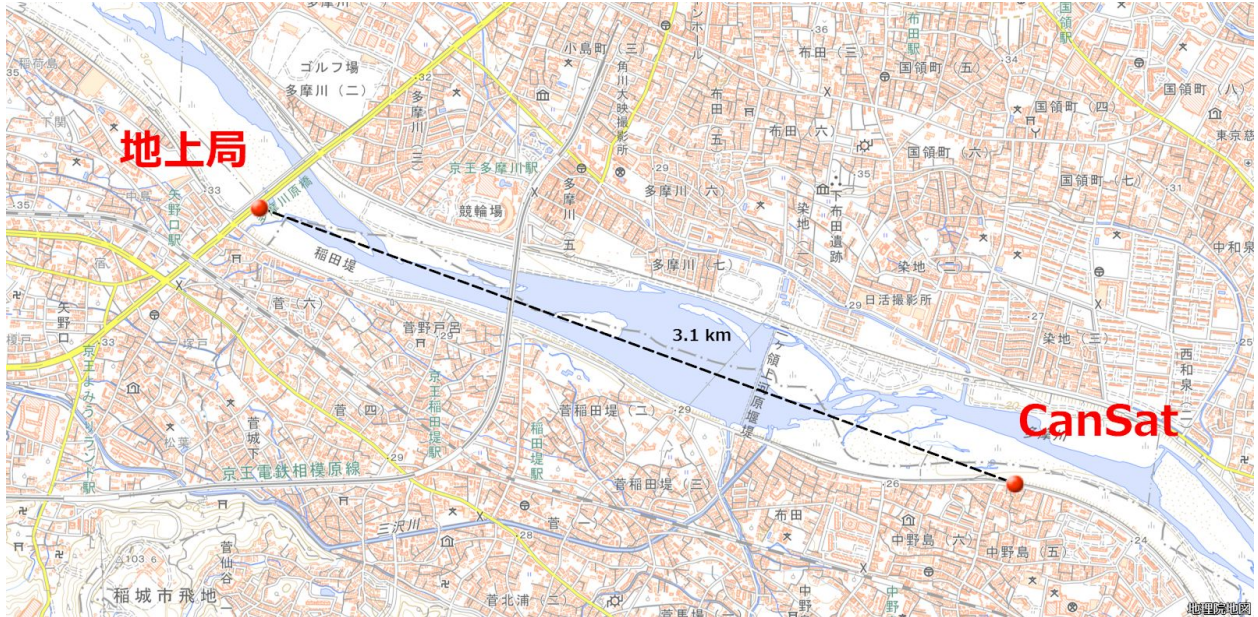


Fig. 6.3.1 地上局と通信限界地点との距離

- 結論

PCとCanSatとの通信限界距離は3.0kmであることを確認できた。つまり、3.0km以内であればCanSatのGPS座標を確認できる。ARLIS本番において、CanSatは上空約4kmの高さにおいてロケットから放出され、上空約3.0kmの高さからGPS座標を受信できると考えられる。落下試験より、パラシュートが開傘した後の終端速度は後述する4. 落下試験より最大でも8.2m/sであるため、上空約3.0kmから着地までかかる時間は約6分であり、車で時速30km程の速さ(6分で3km走行可能)でCanSatを追跡するため、着地する時間前後までにCanSatを視認することが可能であると考えられ、ロスト対策できていると言える。

4. 落下試験

- 目的

キャリアから放出されたときに、減速機構であるパラシュートが問題なく開傘し、減速可能であるかを確認する。

- 試験内容

電気通信大学西6号館7階（地面からの高さ約25m地点）からキャリアに収納した擬似外骨格とCanSat（重量および大きさを実際の外骨格およびCanSatと同等になるように作成したモックアップを使用した。本試験では、パラシュートの性能の検証を目的としているため、大きさと重量が同等であれば試験の目的を満たす。尚、着地時の衝撃については11. 着地衝撃試験で検証する。）を放出し、降下させる。減速の検証は実験動画から落下速度を算出し、落下時の速度が自由落下に比べて減速していることを確認する。

落下時の速度に関しては、Fig. 6.4.1に示すように、パラシュートをつけた状態でキャリアから放出されて落下したCanSatが、地上から約8.4m（建物3階）の高さを通り、着地までにかかった時間を動画から計測し、以下の式を用いて計算する。

$$h=8.4 \text{ m}$$

$$v=h/t \text{ m/s}$$

また，自由落下した際の速度は以下の式より，22.1m/sと求まる．

$$v=\sqrt{2gh}$$

$$h=25\text{m}, g=9.8\text{m/s}^2$$

$$\therefore v=22.1 \text{ m/s}$$



Fig. 6.4.1 地面から3階までの高さ

- 結果

キャリア放出後にパラシュートが開傘するか確認するため，10回試験を行った．結果と試験動画を表6.4.1に示す．

表6.4.1 パラシュート開傘の結果

実験回数	試験動画	結果（開傘）
1回目	https://youtu.be/lf5mwtaB-ns	成功
2回目	https://youtu.be/IA8BC40HWks	成功

3回目	https://youtu.be/h68ORXLLKE4	成功
4回目	https://youtu.be/RMP5NHVMY5c	成功
5回目	https://youtu.be/UHwB_fKcHFQ	成功
6回目	https://youtu.be/KMzp8pDWS6g	成功
7回目	https://youtu.be/nReQO0nXzIg	成功
8回目	https://youtu.be/-xO2IVJfxkI	非常階段に 当たってしまった
9回目	https://youtu.be/zBO8u6GN9uY	成功
10回目	https://youtu.be/0hFcZ41NL-s	成功

また、終端速度を計算した結果を表6.4.2に示す。各落下において動画から落下するまでにかかる時間を見積もり、この時間から速度を算出した。

表6.4.2 終端速度計算結果

実験回数	算出速度	動画該当箇所
1回目	8.4m /0.96 s $\div$ 8.8m/s	00:16:17~00:17:13
2回目	8.4m /1.02 s $\div$ 8.2m/s	00:18:08~00:19:10
3回目	8.4m /1.06 s $\div$ 7.9m/s	00:09:13~00:10:19
4回目	8.4m /1.02 s $\div$ 8.2m/s	00:13:24~00:14:26
5回目	8.4m /1.06 s $\div$ 7.9m/s	00:06:19~00:07:25
6回目	8.4m /1.10 s $\div$ 7.6m/s	00:18:00~00:19:10
7回目	8.4m /0.98 s $\div$ 8.6m/s	00:15:22~00:16:20
8回目	-	-
9回目	8.4m /1.00 s = 8.4m/s	00:14:25~00:15:25
10回目	8.4m /1.00 s = 8.4m/s	00:10:15~00:11:15

上記の結果より、終端速度の平均を算出すると $v = 8.2\text{m/s}$ であり、自由落下した際の速度である 22.1m/s に比べて減速していることを確認した。

- 結論

減速機構であるパラシュートが問題なく開傘され、減速していることを確認できた。このことから、安全基準を満たすものであると言える。

5. 静荷重試験

- 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ていないことを確認する。

- 試験内容

準静的荷重はレギュレーションを参考にし、キャリアの縦方向（高さ）に約10GをCanSatに与える。この状況を再現するために、紐をつないだ袋にロケット搭載状態を想定したCanSatを入れてハンマー投げの要領で、袋を回す。加速度はCanSatに搭載されたHiLetGo MPU9250を用いて測定する。なお、センサ値は0.1秒ごとに加速度を取得する。

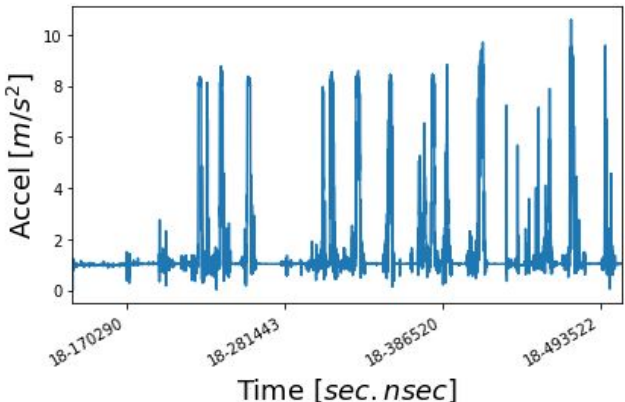
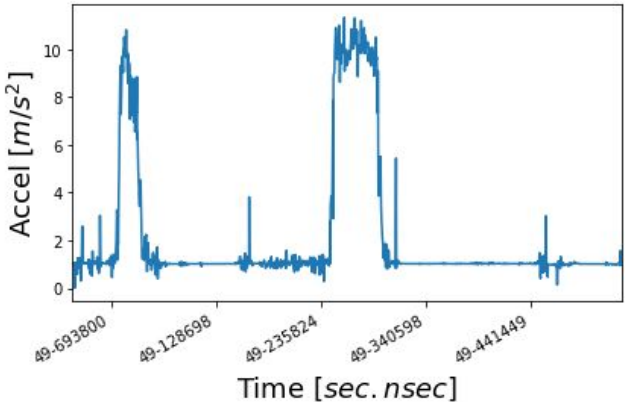
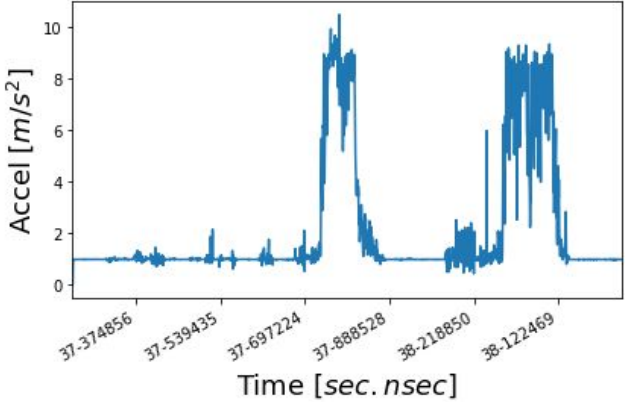
準静的荷重を加えた後、キャリアからCanSatを放出する。CanSatの機能が損なわれていないことを確認するためにCanSatに破損がないかを目視で確認し、サーボモータとモータの制御による外骨格脱出を行うことで動作系が正常に機能するとみなす。

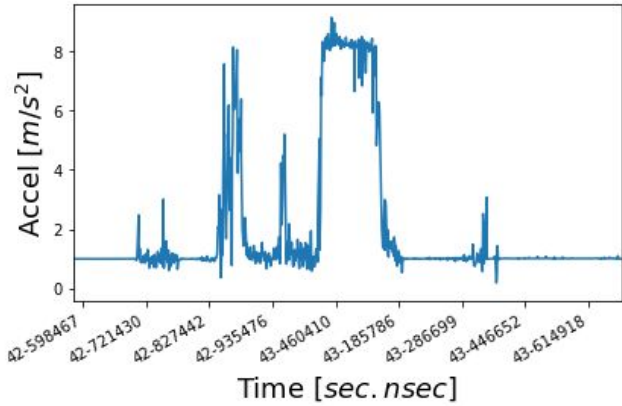
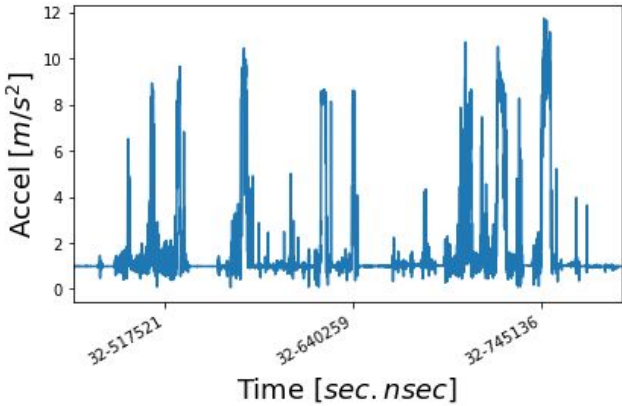
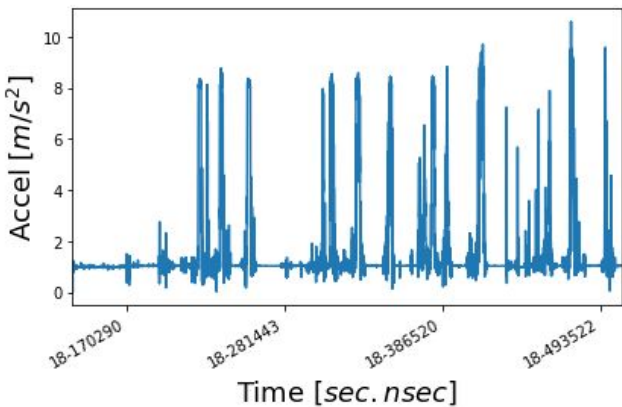
- 結果

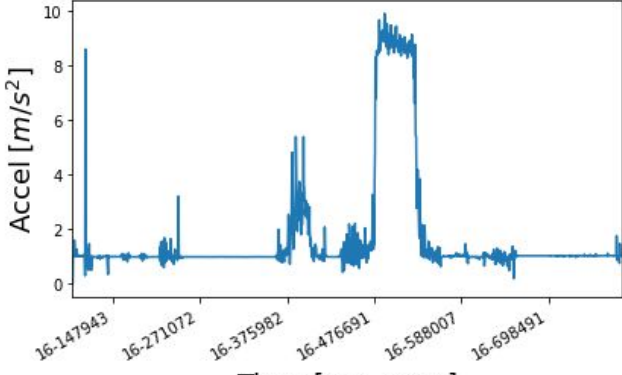
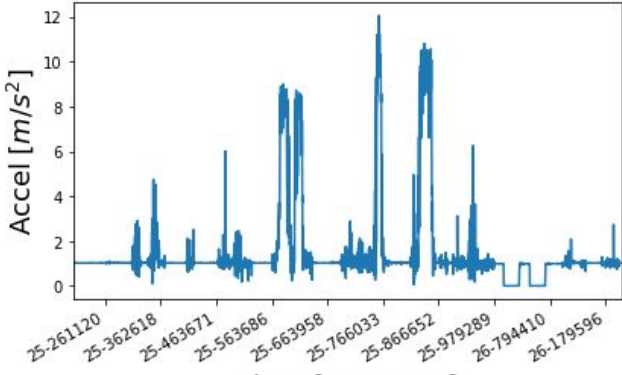
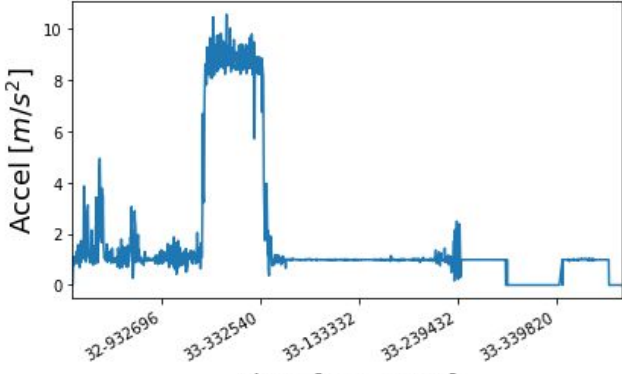
実験結果を表6.5.1に示す。図中の縦軸は加速度を、横軸は時間を表している。10回中7回において、負荷をかけた後でも、CanSatに破損は見られなかった。また、サーボモータの制御による外骨格脱出、各センサの値取得、モータが動作することから、正常に動作していることが確認できた。なお、2回目の加速度センサの不具合に関しては、センサの接触不良が原因であり、ホットボンドでしっかりと固定し直した。3回目のサーボモータのケーブルが断線したことに関しては、コネクタを抜き差ししやすくするためのホットボンドが圧迫されることによって、ケーブルが破損してしまったからである。この問題に対しては、ホットボンドをつける部分を薄くするなどの対策で解決した。7回目の試験では、CanSat収納の向きを逆さにしてしまったため、外骨格の中でCanSatが上方向にずれたが、外骨格の中でのCanSatの位置を戻したのち、切り離しに成功した。

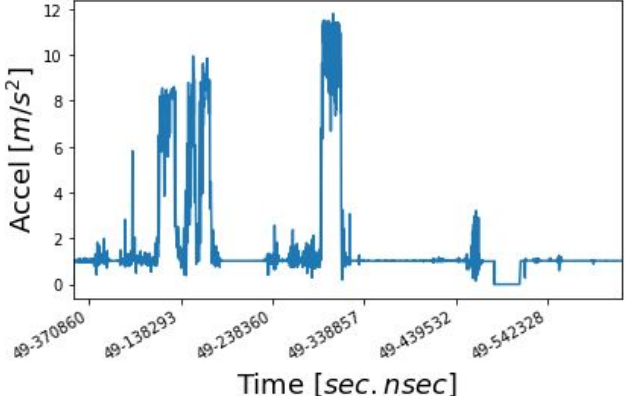
表6.5.1 静荷重試験結果

実験回数	実験動画	結果
------	------	----

1回目	https://youtu.be/tA-Ql6AkYC8	 CanSatに問題なし，切り離し成功
2回目	https://youtu.be/tL6KCKjVA9g	 加速度センサに不具合が現れる，切り離し成功
3回目	https://youtu.be/JlJtKj3j27o	 サーボのケーブルが一部破損， モータやセンサ類は問題なし

4回目	https://youtu.be/px8-EMdSu8	 Accel [m/s^2] Time [sec. nsec] CanSatに問題なし, 切り離し成功
5回目	https://youtu.be/aZmDNmGVn-4	 Accel [m/s^2] Time [sec. nsec] CanSatに問題なし, 切り離し成功
6回目	https://youtu.be/mvIc0rZUTT4	 Accel [m/s^2] Time [sec. nsec] CanSatに問題なし, 切り離し成功

7回目	https://youtu.be/3qHoZ50pWYo	 Time [sec. nsec] キャリアへの収納を逆にしてしまったため、 CanSat外骨格の中で上方向にずれた。 位置を戻したのち、切り離し成功
8回目	https://youtu.be/AKzcFzOPfDE	 Time [sec. nsec] CanSatに問題なし、切り離し成功
9回目	https://youtu.be/UM3siRyFpIU	 Time [sec. nsec] CanSatに問題なし、切り離し成功

10回目	https://youtu.be/q-O3jv2tkjQ	 CanSatに問題なし，切り離し成功
------	---	--

- 結論

CanSatに10Gの負荷をかけてもCanSatに破損はなく，プログラム上で外骨格切り離しが可能であることが確認できたため，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ていないことが示された。

6. 振動試験

- 目的

振動試験装置により，CanSatにロケット打ち上げ時相当の荷重をかけた後，CanSatに破損がなく，全センサが正しく動作することを確認する。

- 試験内容

外骨格に収納されたCanSatをキャリアに収納し，振動試験装置を用いてロケット上昇時に生じるランダムな振動を与える。振動の種類はARLISSレギュレーションの1.4項目に従い，ランダム振動15Gを加える。

試験後，外骨格脱出から走行までのシーケンスを行い，CanSatが正常に動作することを確認する。

なお，モータに関して，発注の関係上，本来使用するモータと比べて4g軽いモータを搭載している。モータの周りにスポンジなどをつけることによって，重さを同じにしている。

- 結果

キャリア収納から振動試験装置で振動を加えた後，外骨格から脱出しCanSatが正常に動作するかどうか一連の流れを撮った実験動画を以下に示す。

<https://youtu.be/5yxCbJfCGIg>

以下に動画中の各イベントの時間を示す。

0:00～ 外骨格収納

2:00～ キャリア収納

3:40～ 質量計測
 4:30～ 振動試験装置の準備
 6:40～ ランダム振動開始
 10:20～ ショック振動開始
 14:00～ キャリア放出
 14:30～ 落下判定
 15:40～ 着地判定
 15:50～ 外骨格切り離しおよび脱出
 (光, 加速度, ジャイロセンサ値・気圧センサ値の取得)
 17:40～ ナビゲーション開始
 37:40～ 各センサの値の取得
 38:15～ GPS値の取得
 38:30～ CanSatに損傷がないことを確認

上に示すように, GPS値の取得が38:15～と試験終了からしばらく時間がたっているが, これは試験場所が建物に囲まれており見晴らしの良い場所が確保できずGPSとの通信が困難だったためである. 与えたランダム振動の加速度推移をFig. 6.6.1に示す. グラフの横軸は時間(sec), 縦軸は加速度(G)を表している.

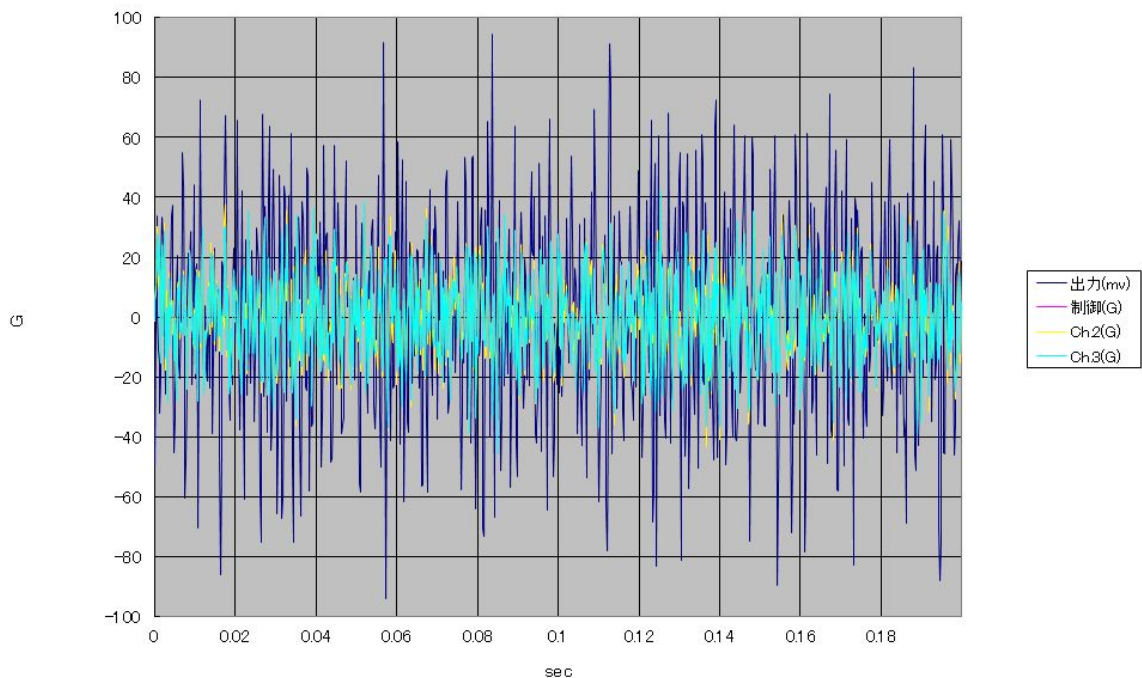


Fig. 6.6.1 ランダム振動の加速度

ランダム衝撃を加えた後、全センサ値(GPS, 大気圧, 加速度, ジャイロ, 地磁気, 光, 距離)を正しく取得し、動力系(モータ・サーボモータ)も問題なく機能することを確認できた。以下、Fig. 6.6.2に各センサ(GPSを除く)値取得時の様子を示す。また、CanSatおよび外骨格に破損がないことを確認した。

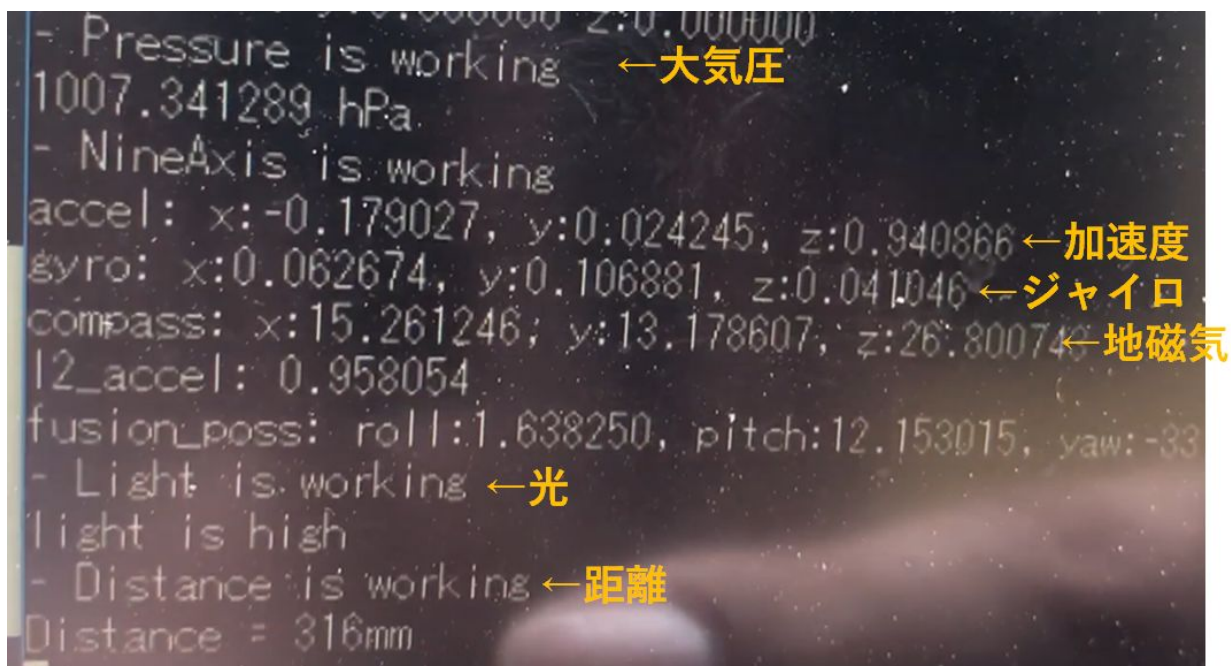


Fig. 6.6.2 各センサ値取得時の様子

- 結論

振動試験装置によりCanSatにロケット打ち上げ時相当の荷重をかけた後、CanSatや外骨格、モータ類に損傷がなく、全センサ(GPS, 光, 加速度, ジャイロ, 地磁, 気圧)が正しく動作することを確認できた。

7. 分離衝撃試験

- 目的

CanSatにロケット分離時相当の衝撃を振動試験装置によりかけた後、CanSatやモータ類に損傷がなく、全センサが正しく動作することを確認する。

- 試験内容

6. 振動試験と同様の振動試験装置を用いて40Gのショック振動を与える。試験後、外骨格脱出から走行までのシーケンスを行い、CanSatが正常に動作することを確認する。衝撃の種類はARLISSレギュレーションの1.4項目に従い、ショック振動：[目標]
 $392\text{m/s}^2 \rightarrow 392/9.8(1\text{G})=40(\text{G})$ を与える。

- 結果

本試験の実験動画は7. 振動試験と同一である。与えたショック振動の加速度推移を以下、Fig. 6.7.1に示す。グラフの横軸は時間(sec)，縦軸(m/s^2)は加速度を表している。

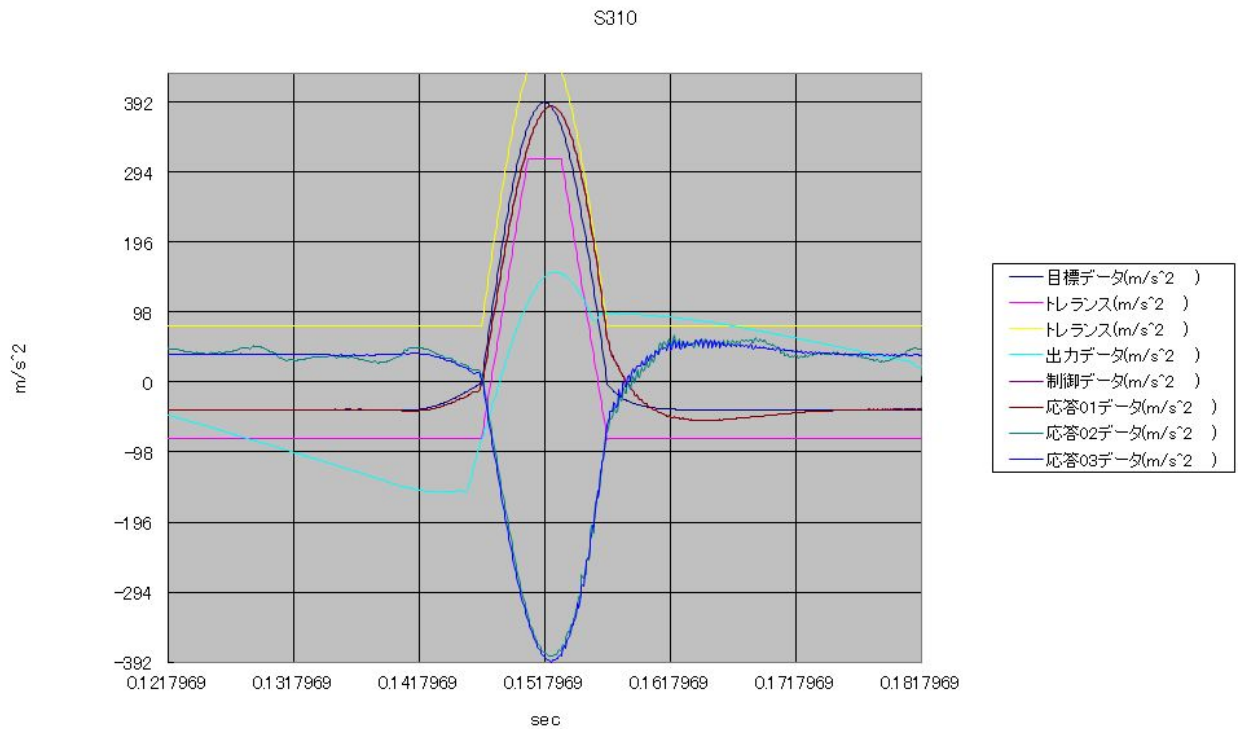


Fig. 6.7.1 ショック振動の加速度

ショック振動を与えた後、CanSatが正常に動作するかを確認した。結果は7. 振動試験と同様であるため、ここでは省略する。

- 結論

ショック振動を与えた後、一連のシーケンスを行うことができたことため、CanSatがロケット分離時の衝撃に対して耐えうることを確認した。

8. 開傘衝撃試験

- 目的

キャリアからCanSatが放出され、落下中のパラシュート開傘衝撃によって、CanSat及びパラシュート連結機構の機能が損なわれていないことを確認し、CanSatが正常に動作することも確認する。

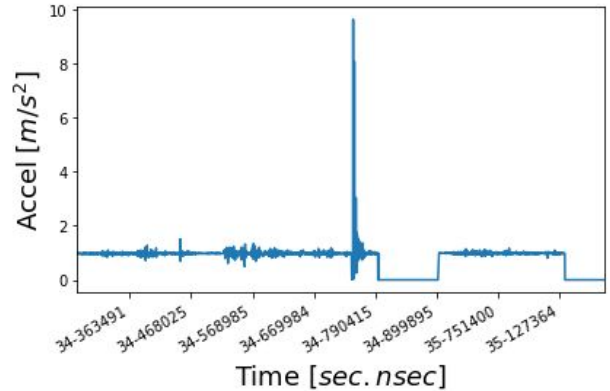
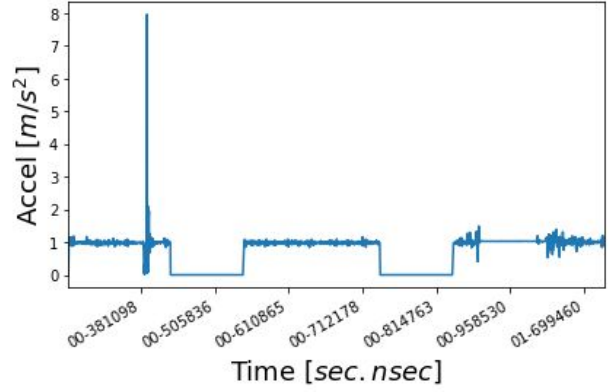
- 試験内容

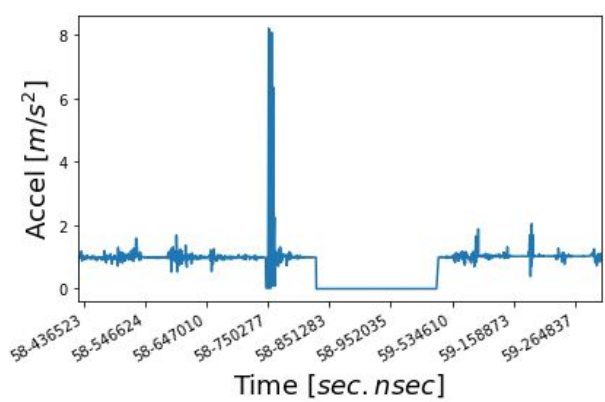
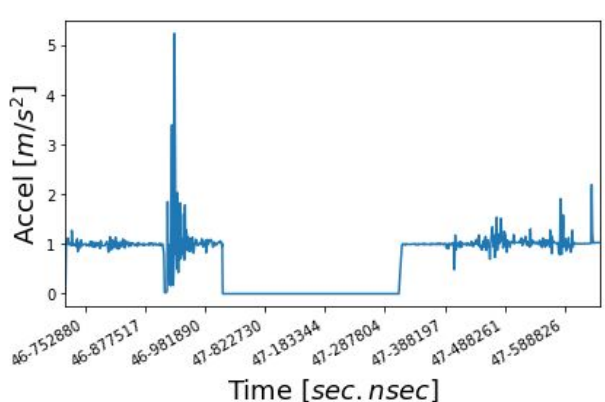
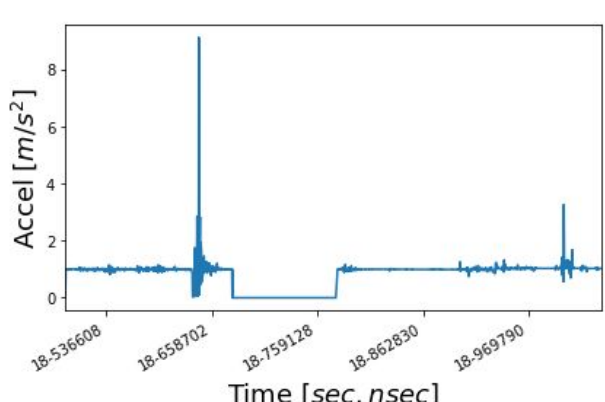
CanSatにパラシュートを取り付けた状態で、パラシュート紐の上端を固定し自由落下させることで、開傘時の衝撃を再現する。開傘衝撃を与えた後、全てのセンサの動作確認、サーボモータの動作確認、外骨格とCanSatに破損がないかを確認する。

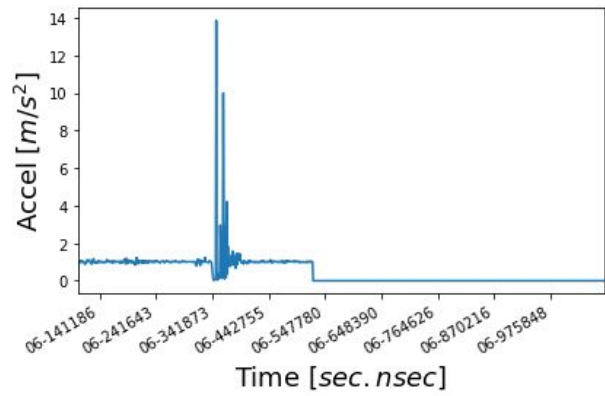
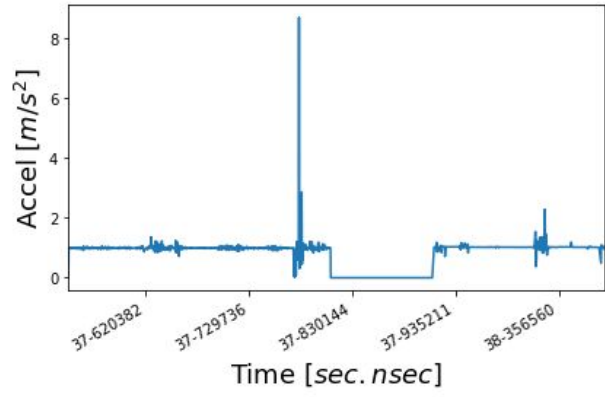
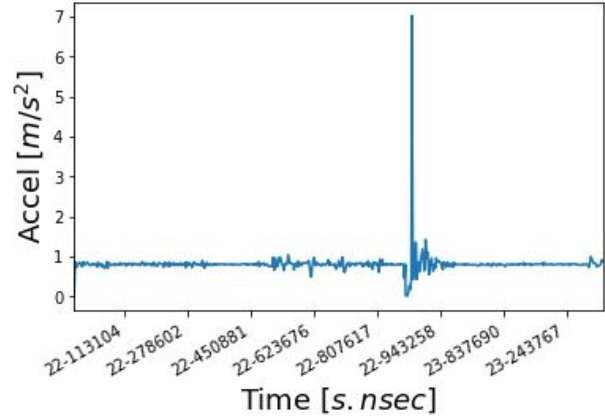
- 結果

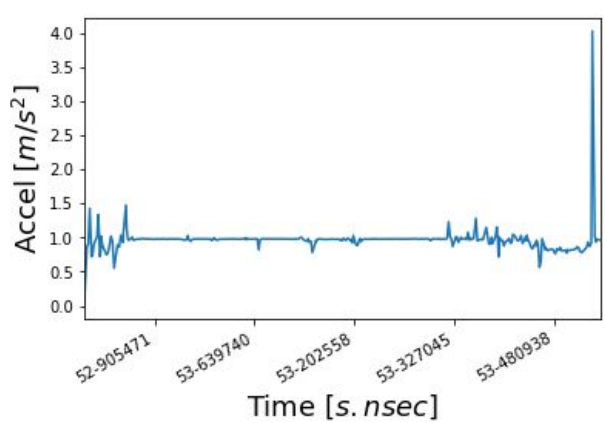
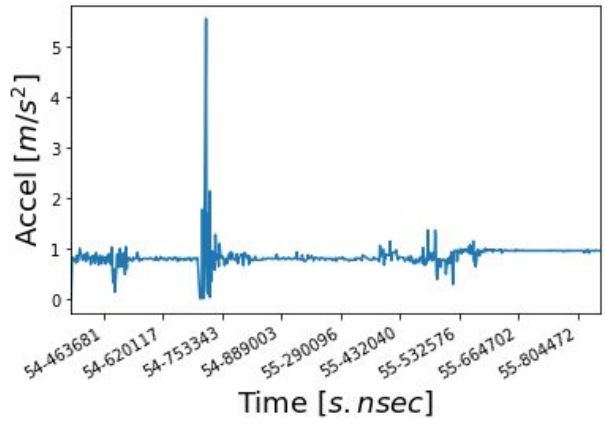
表6.8.1に実験動画とその結果を示す。「衝撃の大きさ」の列にある図は、縦軸が開傘衝撃時のCanSatにかかる衝撃の大きさを、横軸が時間を表している。実験1回目で衝撃により多関節機構が外骨格外にはみ出してしまう問題が起き、これに対処するため2回目以降では、はみ出すことを防ぐパーツを外骨格に取り付けて実験をした。(なおパーツを付け加え改変した外骨格は、放出試験等の外骨格を用いる実験の際、全て改変後の外骨格を用いて実施している。)3～5回目において、外骨格脱出用のピンを切り離すことができていなかったが、6回目以降、ピンをつないでいる紐の長さを調節することで解決した。

表6.8.1 開傘衝撃の結果

回数	実験動画	CanSatの状態	衝撃の大きさ
1	https://youtu.be/8Vu553AkBtw	外骨格内部のCanSatのシャーシ部分が回転してしまい、サーボが外骨格切り離しの動作を実行することができなかった。	
2	https://youtu.be/tSA LtI8GI	問題なし	

3	https://youtu.be/KXAiwJJm3hI	外骨格脱出用のピンを切り離すことができなかった.	 Accel [m/s^2] Time [sec. nsec]
4	https://youtu.be/cx5HwK0Uph4	外骨格脱出用のピンを切り離すことができなかった.	 Accel [m/s^2] Time [sec. nsec]
5	https://youtu.be/zwLLtxF8WHo	外骨格脱出用のピンを切り離した際に、コンクリートの地面に当たってしまい、シャーシ部分から2番目のサーボのギヤが破損してしまった。しかし、着地時は外骨格によって保護されるため、この衝撃はサーボにはかからないため問題はない.	 Accel [m/s^2] Time [sec. nsec]

6	https://youtu.be/3D-Ohbg6h3w	問題なし	 Acceleration graph for item 6. The y-axis is labeled 'Accel [m/s²]' and ranges from 0 to 14. The x-axis is labeled 'Time [sec. nsec]' and shows values from 06-141186 to 06-975848. A sharp peak is visible at approximately 06-442755, reaching an acceleration of about 14 m/s².
7	https://youtu.be/Oq0n3NkATlk	問題なし	 Acceleration graph for item 7. The y-axis is labeled 'Accel [m/s²]' and ranges from 0 to 8. The x-axis is labeled 'Time [sec. nsec]' and shows values from 37-620382 to 38-356560. A sharp peak is visible at approximately 37-830144, reaching an acceleration of about 8 m/s².
8	https://youtu.be/t4akRX3_JP4	問題なし	 Acceleration graph for item 8. The y-axis is labeled 'Accel [m/s²]' and ranges from 0 to 7. The x-axis is labeled 'Time [s. nsec]' and shows values from 22-113104 to 23-243767. A sharp peak is visible at approximately 22-943258, reaching an acceleration of about 7 m/s².

9	https://youtu.be/a3Kzgows-c	電池ボックス部分が外れた。原因はテープをしっかりと巻いていなかったからである。	 Accel [m/s^2] Time [s. nsec]
10	https://youtu.be/rH7aLlsg-Dg	問題なし	 Accel [m/s^2] Time [s. nsec]

- 結論

キャリアからCanSatが放出された際のパラシュートの開傘衝撃に対して、CanSat及びパラシュートの連結機構が損なわず、またCanSatが衝撃に耐えうる。

9. 通信機電源ON/OFF試験

- 目的

ロケットの通信機器に悪影響を与えないよう、キャリア内でWi-Fi、LoRaの電源が切れることを確認する。

- 試験内容

- Wi-Fi

CanSatに搭載されているRaspberry Pi ZERO WにWi-Fiが内蔵されているため、アクセスポイント化し、PCと通信している状態から、Wi-Fiの通信状態をOnからOffに切り替える。Wi-Fiの通信状態がOffに切り替わることを、ターミナル上で自由にOn/Offできることを確認する。

まず、ターミナルにコマンド（wifistop）を入力し、Wi-Fiの通信状態をOnからOffに切り替える。Wi-Fiの通信状態がOffに切り替わることを、ターミナル上でセンサ値ログが表示されなくなることにより確認する。次に、キャリアからCanSat

を放出させ、シーケンスをWaitingから移行させた後、Wi-Fiの通信状態がOnに切り替え、CanSatとPCが通信できていることを、センサ値やCanSatのシーケンス状態がターミナル上に表示されることにより確認する。

- LoRa

LoRaはRaspberry Pi ZERO WのGPIOピンを制御することで、スリープモード（電波の出力が0）にすることが可能である。そのため、該当するGPIOピンを制御し、LoRaのスリープモードを自由にOn/Offできることを確認する。

まず、ターミナルにコマンドを入力し、LoRaの通信状態をOnからOffに切り替える。LoRaの通信状態がOffに切り替わることを、地上局側のLoRaがCanSatからの信号を受け取れなくなることにより確認する。次に、Wi-Fiの時と同様にして、地上局側のLoRaがCanSatからの信号を受け取れることにより、LoRaの通信状態がOnに切り替わることを確認する。

- 結果

- Wi-Fi

実験動画を以下に示す。Waitingのシーケンス内で本番と同様の手順でマイコンのWifiの通信をOffにした。実際にOffにされたかをスマートフォンアプリのWifiの電波を確認するツール(Wifi Analyzer)を用いて確認した。また、Waitingシーケンスが終了した後、自動的にWifiの通信機能がOnになることも同様に確認した。

<https://youtu.be/MKzOcMvg45A>

- LoRa

実験動画を以下に示す。はじめは、CanSatのLoRaから地上局側のLoRaに向けてGPS座標を送信している状況である。CanSat側のターミナルにLoRaをスリープモードにするコマンドを入力したのち、地上局側がCanSat側からGPS座標を受信しなくなることが確認できた。次に、CanSat側のターミナルにLoRaのスリープモードを解除にするコマンドを入力したのち、地上局側がCanSat側からGPS座標を受信できることが確認できた。

<https://youtu.be/FWWk44p0BOE>

- 結論

- Wi-Fi

問題なくWi-Fiの通信状態をOnからOffに切り替えと、シーケンス移行後OffからOnへできる。

- LoRa

問題なくLoraの通信状態をOnからOffに切り替えと、シーケンス移行後OffからOnへできる。

10. 通信周波数変更試験

- 目的

通信機器の周波数が他チーム等と干渉した時に、Raspberry Pi側から発生させるLoRaの信号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

- Wi-Fi

CanSatに搭載されているRaspberry Pi ZERO Wが発信しているWi-Fiに接続し、

Raspberry Pi ZERO Wの設定ファイルを書き換えることでWi-Fiの周波数（チャンネル）を変更する。変更後、周波数に変更されていることをWi-Fi分析ソフトの「WiFi Analyzer」を用いて確認する。

- LoRa

CanSatに搭載されているLoRaモジュール（ES920LR）を2つとPCを2台用意する。はじめに、両LoRaの周波数を統一し、通信ができることを確認する。次に、片方のLoRaの周波数を変更し、通信ができないことを確認したのち、もう片方のLoRaの周波数を変更し、通信ができることを確認する。なお、LoRaの通信周波数はあらかじめ設定されているチャンネルの通信周波数のみ使うことができる。そのため、LoRaの使用チャンネルを変更することによって、通信周波数を変更する。LoRaの各チャンネルの使用周波数を表6.10.1に示す。

表6.10.1

帯域幅	チャンネル	使用周波数
125kHz以下	1~15	920.6 ~ 923.4 MHz (0.2MHz刻み)
250kHz	1~7	920.7 ~ 923.1 MHz (0.4MHz刻み)
500kHz	1~5	920.8 ~ 923.2 MHz (0.6MHz刻み)

- 結果

- Wi-Fi

周波数チャンネル変更前と後の試験結果をFig. 6.10.1, Fig. 6.10.2として下記に掲載した。図より、CH2に設定されていたWi-Fiの周波数チャンネルが設定ファイルの書き換えによってCH3の周波数チャンネルに変更されたことを確認した。これにより、自由に周波数チャンネルを変更できることが示された。

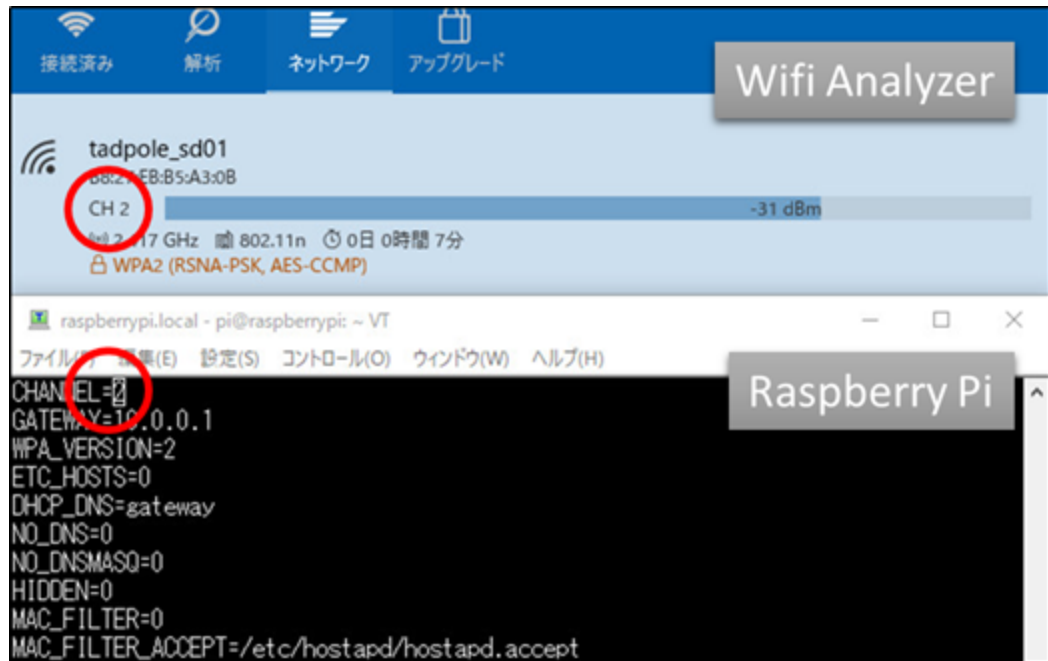


Fig. 6.10.1 周波数変更前

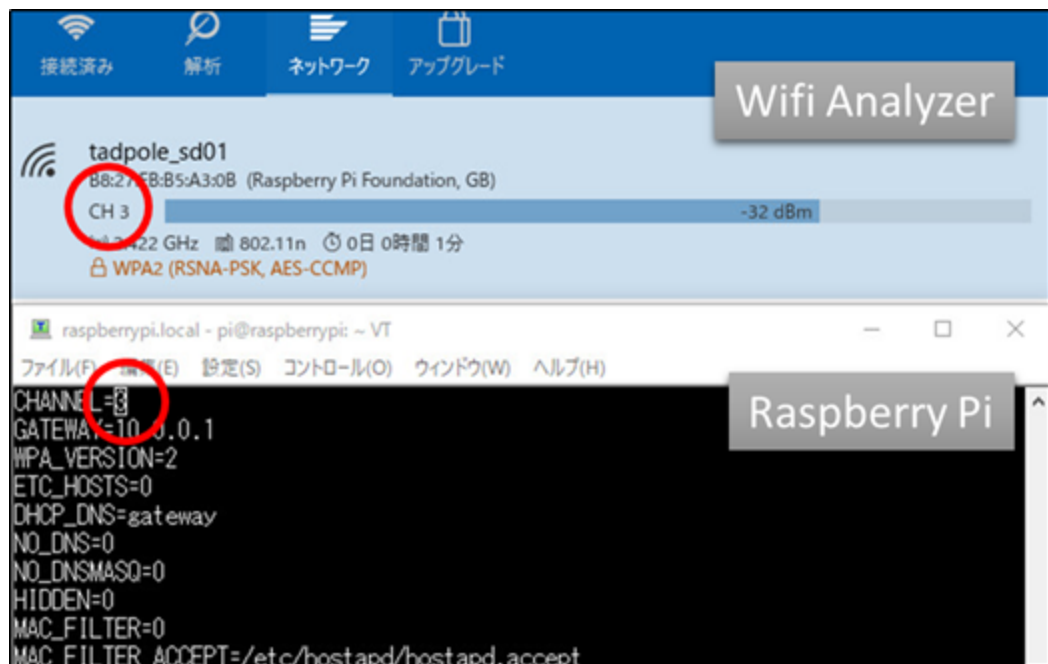


Fig. 6.10.2 周波数変更後

○ LoRa

実験動画を以下に示す. はじめは2つのLoRa (A, B) の使用周波数が同じであるため, データを互いに送受信できることが確認できた. 次に, 片方のLoRa Aの使用周波数のみを変更したのち, データの送受信ができなくなることが確認できた. この後, LoRa Bの使用周波数をLoRa Aの使用周波数に合わせることでデータの送受信ができるようになることが確認できた. よって, LoRaの通信周波数は表

6.10.1の周波数に制限はされているが、変更することが可能であることが示された

<https://youtu.be/Eo23mbuZhzU>

- 結論

無線機器の通信周波数を限定的ではあるが、変更することが可能であることが確認できた。よって、他団体やロケット関連の無線機器と干渉する場合に、通信周波数を調整することが可能であることが示された。

11. 着地衝撃試験

- 目的

CanSatがARLISS本番で想定される着地時の衝撃に耐えることができるかを確認する。

- 試験内容

ARLISS本番同様の終端速度が出るような高さからCanSatを自由落下させることで着地時の衝撃を再現する。パラシュートを用いて落下した時の終端速度は昨年度ARLISS本番で打ち上げたCanSat（高玉研究室チームAres）のログを参考に算出する。以下に算出に用いたデータシートを添付する。なお、本年度使用予定のパラシュートは昨年度使用したものと同一のため、終端速度は以下の算出値から大きく変化しないと考えられる。

https://drive.google.com/file/d/1fmW3v8m8a36b4Hb4_-EUrAHla68eWZpY/view?usp=sharing

このデータシートから想定される終端速度は、 $420\text{m}/57\text{s} \div 7.37\text{m/s}$ である。この終端速度をもとに本番で想定される着地時の衝撃を与えるために必要な高さを以下の式により求める。

$$\frac{1}{2}mv^2 = mgh \Leftrightarrow h = \frac{1}{2g}v^2$$
$$h = \frac{1}{2 \times 9.8} \times 7.37^2 \div 2.77\text{m}$$

以上より実際の落下衝撃に相当する高さは2.77mである。本試験ではこの高さより高い約3mからCanSatを自由落下させることで、本番同等の着地衝撃を与える。CanSatの着地後、CanSatの状態、センサ値の取得、動力系の動作を確認し、着地時の衝撃により問題が起きていないことを確認する。1～5回目は外骨格の底面に柔らかいスポンジがついていたが、着地後の跳ね返りが強かったため、6～8回目は底面のスポンジを硬い素材のスポンジに変えて実験を行った。

- 結果

着地衝撃試験の結果を表6.11.1にまとめた。

表6.11.1 着地衝撃試験の結果

回数	実験動画	外骨格の状態	CanSatの状態	外骨格脱出
1	https://youtu.be/ya98JN1wQzU	問題なし	問題なし	成功
2	https://youtu.be/hXntRYpeylA	問題なし	ラズパイが再起動してしまった	成功

			が、センサ類やモータ類の動作に問題は見られなかった	
3	https://youtu.be/ktgHgIfi2_E & https://youtu.be/FznkSHzdoVY	問題なし	問題なし	成功
4	https://youtu.be/M1GIgYuFQHg	問題なし	問題なし	手で補助してピンを切り離した
5	https://www.youtube.com/watch?v=55p98Bps8Is&feature=youtu.be	問題なし	問題なし	着地時にピンが抜けてしまった
6	https://youtu.be/8D5iub6h384	問題なし	多関節機構の末端部が外れてしまった	成功
7	https://youtu.be/2ODmuF4QuM	問題なし	問題なし	成功
8	https://youtu.be/FqsGx5hGTVc	問題なし	問題なし	成功

- 結論
外骨格およびCanSatが着地時の衝撃に耐え、外骨格の底面のスポンジの素材を変更した結果外骨格からの脱出可能である。なお6回目の多関節機構の末端部が外れた原因は固定が弱かったためである。

12. 外骨格脱出試験

- 目的
CanSatが着地衝撃から守るための外骨格を切り離し、脱出可能であることを確認する。
- 試験/解析内容
CanSatを外骨格に収納し、外骨格脱出シーケンスのプログラムを実行することによって、CanSatが自律的に外骨格切り離し、走行し始められることを確認する。
- 結果
試験結果を表6.12.1に示す。なお、各回の実験において、外骨格の向きはすべて異なる方向が地面を向くように配置した。

表6.12.1 外骨格脱出試験結果

回数	実験動画	結果
----	------	----

1回目	https://youtu.be/0YJCJthgjkCc	脱出して走行できた
2回目	https://youtu.be/H_2wmsqIUmk	脱出して走行できた
3回目	https://youtu.be/HjjZUYB2EZU	脱出して走行できた
4回目	https://youtu.be/00TCJdycn-I	脱出して走行できた

- 結論
CanSatが外骨格から脱出し走行可能である。

13. 走行試験

- 目的
CanSatが目的地に到達するために必要な走破性能を持つことを確認する。
- 試験内容
轍を作成した状態の砂場で走行させる。なお、轍は幅を約40cm、深さは異なる2種類(8cm, 10cm)を作成する。さらに、ARLISS現地で作成した轍、もともとできていた深い轍、複数の轍が連続している環境、坂道の4種類について追加で実験をした。
- 結果
前年度のCanSat（Aresローバ）では抜け出すことが不可能であった深さ10cmの轍を抜け出すことができた。昨年度のローバの概要と昨年度の試験結果を表6.13.1に示す。

表6.13.1 Aresローバの概要と試験動画

タイヤ径	139 mm
全幅	225 mm
高さ	135 mm
重量	1008 g
試験動画	https://youtu.be/76BcOgAAjeA

今回の試験結果を表6.13.2に示す。

表6.13.2 試験結果

深さ	回数	実験動画	結果
8cm	1回目	https://youtu.be/3P_kJC9j1yA	多関節機構を利用して轍を抜け出すことが出来た
8cm	2回目	https://youtu.be/TPpkaxQvrQ4	多関節機構を利用して轍を抜け出すことが出来た
8cm	3回目	https://youtu.be/jk2YeyF1-18	多関節機構を利用して轍を抜け出すことが出来た
8cm	4回目	https://youtu.be/iEpsQQFF1C	多関節機構を利用して轍を抜け出すこ

		U	とが出来た
8cm	5回目	https://youtu.be/JSHqR6gmjLA	多関節機構を利用して轍を抜け出すことが出来た
10cm	1回目	https://youtu.be/ftbx6un7VXE	多関節機構を利用して轍を抜け出すことが出来た
10cm	2回目	https://youtu.be/XC63HLIPed4	多関節機構を利用して轍を抜け出すことが出来た

また、ARLISS現地で追加で実施した試験結果を表6.13.3に示す。

表6.13.3

種類	実験動画	結果
自らの手で作成した轍	https://youtu.be/acUPF92NNGE	多関節機構を利用して轍を抜け出すことが出来た
もともとできていた深い轍	https://youtu.be/Bj-VoyaUkKU	多関節機構を利用して轍を抜け出すことが出来た
複数の轍が連続している環境	https://youtu.be/Jed7I37ISiA	多関節機構を利用して轍を抜け出すことが出来た
坂道	https://youtu.be/R8azUQJtIRs	多関節機構を利用して坂道を走行することができた

- 結論

CanSat が轍にはまったにも抜け出して走行可能であり、坂道も走行可能な走行性能があることが確認できた。

14. 反転・横転復帰試験

- 目的

CanSatが走行中に反転・横転をした際に、走行可能な姿勢に復帰できることを確認する。

- 試験内容

- 反転復帰

CanSatを反転させた状態から反転復帰シーケンスのプログラムを実行し、走行可能な姿勢に復帰させる。

- 横転復帰

CanSatを横転させた状態から横転復帰シーケンスのプログラムを実行し、走行可能な姿勢に復帰させる。

- 結果

反転復帰の試験結果を表6.14.1に示す。

表6.14.1 反転復帰

回数	実験動画	結果
1回目	https://youtu.be/NVawD4vZ9nI	復帰して走行できた
2回目	https://youtu.be/WrE9lTZVERo	復帰して走行できた

横転復帰の試験結果を表6.14.2に示す。

表6.14.2 横転復帰

回数	実験動画	向き	結果
1回目	https://youtu.be/JdIuH4_OnKM	右側に横転	復帰して走行できた
2回目	https://youtu.be/QvEqvKNnqkE	右側に横転	復帰して走行できた
3回目	https://youtu.be/a2nrxOq5rEo	左側に横転	復帰して走行できた
4回目	https://youtu.be/832pNIN8hpY	左側に横転	復帰して走行できた

- 結論

CanSatが走行中に想定される反転、横転から、走行可能な姿勢に復帰できる。

15. 掘削性能試験

- 目的

多関節機構を有したCanSatの掘削性能を確認する。

- 試験内容

公園にある砂場（柔らかい）と地面（堅い）の2種類について、多関節機構を制御することで掘削をする。

- 結果

表6.15.1に実験結果を示す。

表6.15.1 掘削性能試験の結果

回数	実験動画	地面の状態	深さ
1回目	https://youtu.be/sg_kNldnEVE	柔らかい	4cm
2回目	https://youtu.be/E_IVbcP-dh4	堅い	表面にある柔らかい砂の部分しか掘

			れなかった。堅い部分は数ミリ削れる程度の掘削性能であった。
--	--	--	-------------------------------

- 結論
地面の堅さに限度はあるが、多関節機構を利用した掘削が可能である。

16. 牽引性能試験

- 目的
多関節機構を有したCanSatの牽引性能を確認する。
- 試験内容
故障したと仮定したCanSatを平地で牽引する。牽引する際は、あらかじめ故障したと仮定したCanSatを多関節の末端部分に引っ掛ける状態にしたのち、前進する。
- 結果
以下に実験動画を示す。簡易的に別のCanSatを牽引することができることが確認できた。また、多関節部分を左右に動作させることで、牽引中のCanSatを引き離れたことが確認できた。
<https://youtu.be/7vdDG-P0E3o>
- 結論
多関節機構を用いた牽引が可能であることが示されたが、今後、重量の限界を調査する必要がある。

17. 電力耐久試験

- 目的
CanSatがミッションを実現するためのバッテリー容量を有しているかを確認する。
- 試験内容
充電満タンのリポバッテリーと乾電池を使用して、走行不能になるまでCanSatを走行させる。走行開始から走行不能までの時間をバッテリー使用可能時間とする。
- 結果
試験動画を以下に示す。
冒頭： <https://youtu.be/mBPkO5AVdwk>
終盤： <https://youtu.be/H8NNP16B03k>
試験開始時の回路用電源（リチウムイオンバッテリー）とモータ、サーボモータ用電源（乾電池）の電圧と、終了時の電圧を表6.17.1に示す。3時間38分経過した頃から、モータとサーボモータの動作が鈍くなり、走行不能とした。よって、3時間38分間が走行可能時間とした。

表6.17.1

	回路用電源	モータ、サーボモータ用電源
--	-------	---------------

試験開始前	4.14 V	9.65 V
試験終了時	3.87 V	5.34 V

- 結論

試験の結果から、3時間38分間の走行が可能であることがわかった。CanSatの走行時の速さは、時速1.5kmであった。よって、4km以上の走行が可能であることから、ミッション中に電力切れにならないことが示された。

※例年、ARLISSのゴール地点から1km~3km程離れた地点に着地している。

18. ナビゲーション試験

- 目的

CanSatが自律制御で目的地まで走行可能であることを確認する。

- 試験内容

ゴール地点のGPS座標とCanSatの一座標から、ゴール地点までの距離を計算し、その距離が10m以下になればゴール付近に到達したと検知することを確認する。さらに、ARLISS現地で今年度のゴール地点から約2.5km離れた地点からナビゲーション試験を追加で実施した。

- 結果

試験結果を表3.18.1に示す。10回全てにおいてゴール付近まで到達することができた。

表6.18.1 ナビゲーション試験の結果

回数	実験動画	ゴールからの距離
1回目	https://youtu.be/kSkQ95jLkZs	7.15m
2回目	https://youtu.be/D5FIIUqw1V8	5.22m
3回目	https://youtu.be/OVzaWATy3_o	7.15m
4回目	https://youtu.be/PczUujGJZQY	4.95m
5回目	https://youtu.be/DUur3qdx6zk	3.16m
6回目	https://youtu.be/SVuRwly129E	5.04m
7回目	https://youtu.be/EVoRHv_ly34	0.35m
8回目	https://youtu.be/tMTSzo1LX58	4.77m
9回目	https://youtu.be/Vx-bkpuC5ww	1.59m
10回目	https://youtu.be/qSnf2l2lWGo	2.12m

また、ARLISS現地での追加試験の動画のURLを以下に示す。ゴールからの距離は4.77mであった。

<https://youtu.be/0wUijWhuU6M>

Fig 6.18.1にARLISS現地でのナビゲーション試験でのCanSatの軌跡を示す。スタート地点からゴール地点まで直線に近い軌跡で走行していることからゴール地点までのナビゲーションが正常に行われていることがわかる。

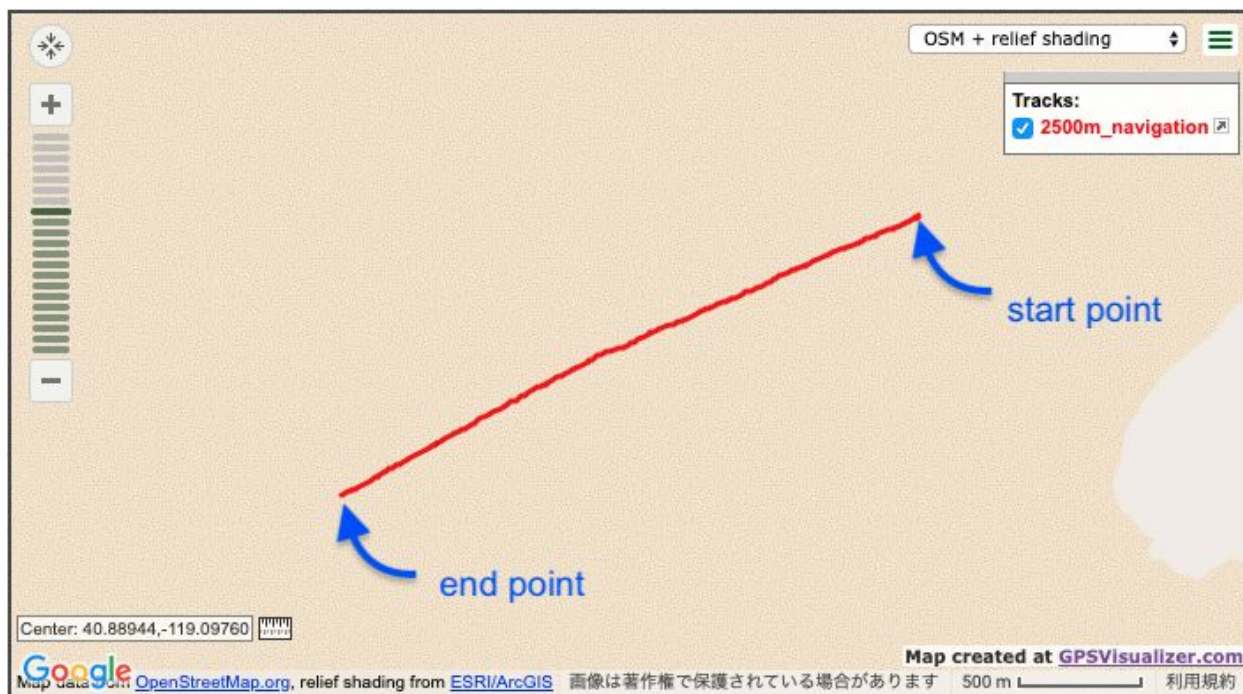


Fig 6.18.1 ARLISS現地での約2.5kmのナビゲーション試験

- 結論

GPS座標を用いてナビゲーションにより10m以内に到達可能であることが確認できた。また、現地での追加試験により、長距離でも正確に目標地点へ走行可能であることが確認できた。

19. ゴール検知試験

- 目的

CanSatが目的地に設置されたゴールコーンを検知し0mゴールが可能であることを確認する。

- 試験内容

まず、ゴールのGPS座標を設定し、その地点にゴールコーンを設置する。次に、CanSatをゴールのGPS座標から半径10m以内に設置し、精密ゴール用の自律プログラムを実行する。CanSatがゴールコーンに対して、0mゴール可能であることを確認する。

- 結果

表6.19.1に結果を示す。3回の実験のうち1回目のみ失敗した。失敗した原因はゴール間近で横転し、制御不能となったためである。

回数	実験動画	0mゴール検知
1回目	https://youtu.be/dwgUpds-fqs	失敗
2回目	https://youtu.be/trHohwyKpM0	成功
3回目	https://youtu.be/fv7yDNeBUJI	成功

- 結論

0mゴールを達成できる可能性があることが示された。しかし、ゴールコーンへの接近時に横転してしまった場合の処理が必要である。具体的には、横転復帰の動作をすることによって、再度ゴールコーンへ接近するように改善する必要がある。これについて、ARLISS現地で横転した際はナビゲーションシーケンスに戻るよう実装した。

20. End-to-end試験

- 目的

レギュレーションを満たしたCanSatが、キャリア収納からミッション終了までの全てのシーケンスを連続して自律的に制御できるか確認する。

- 試験内容

CanSatが本ミッションのサクセスクライテリアを満たし、キャリア放出後、着地判定、外骨格脱出、着地点とゴール地点との中間地点までのナビゲーション、掘削、中間地点からゴール地点までのナビゲーション、精密ゴール判定の全てのシーケンスを自律的に制御できることを確認する。

- 結果

試験結果の動画を以下に示す。

1回目： <https://youtu.be/pGm25N9coyk>

2回目： <https://youtu.be/B5k8aQAxxTY>

気球試験： https://youtu.be/PZ8J_Ewf4c4

ARLISS現地での試験： <https://youtu.be/jy8stJ0NWgw>

以下に気球試験における動画中の各イベントの時間を示す。

0:00～ 外骨格収納

4:17～ キャリア収納

6:16～ 質量計測

6:40～ 投下の準備

13:00～ 投下開始

13:05～ 投下終了
 17:35～ 放出判定
 17:48～ 着地判定
 18:11～ 外骨格切り離しおよび脱出
 19:30～ 草地のない場所に移動しナビゲーション開始
 24:38～ ゴール判定後, CanSatの確認

- 結論
 レギュレーションを満たしたCanSatがキャリア放出, CanSatの着地, GPSゴール判定シーケンスを自律的に制御可能であることを示した.

21. 制御履歴レポート作成試験

- 目的
 ミッション後, 規定の制御履歴レポートを運営へ提出する必要があることから, CanSatが保存したログデータから制御履歴レポートを作成できるかを確認する.
- 試験内容
 End-to-end試験で得られたログデータ(2回目)から, 提出用の制御履歴レポートを作成する.
- 結果
 CanSatの制御履歴を下記に示す. > から始まる文は, 入力コマンドを意味し, 制御履歴の先頭のstart arliss以降ではコマンドが入力されていない. 赤字が説明である.

CanSatの制御履歴
<pre> [2019-8-22-14-38-52] > start arliss(本番のシーケンスが実行) [2019-8-22-14-38-52] Start arliss (stopped) [2019-8-22-14-38-52] ----- [2019-8-22-14-38-52] [Arliss State] Start [2019-8-22-14-38-52] ----- [2019-8-22-14-38-52] Time --> 2019:8:22:14:38:52 [2019-8-22-14-38-52] ----- [2019-8-22-14-38-52] Arliss Mission Start [2019-8-22-14-38-52] Good Luck!!!!!!!!!!!!!! [2019-8-22-14-38-52] ----- [2019-8-22-14-38-52] NineAxis Sensor is Finished [2019-8-22-14-38-52] Distance Sensor is Finished [2019-8-22-14-38-52] Lora is Finished [2019-8-22-14-38-52] ----- [2019-8-22-14-38-52] [Waiting State] Start(CanSatが待機状態) [2019-8-22-14-38-52] ----- [2019-8-22-14-38-52] Time --> 2019:8:22:14:38:52 </pre>

[2019-8-22-14-38-52] [Testing State] Finished
 [2019-8-22-14-38-52] GPS Sensor is Ready!
 [2019-8-22-14-38-52] light sensor is Ready
 [2019-8-22-14-38-53] NineAxis Sensor is Ready!
 [2019-8-22-14-38-53] Pressure Sensor is Ready!
 [2019-8-22-14-38-53] Distance Sensor is Ready!
 [2019-8-22-14-38-55] Lora is Ready!
 [2019-8-22-14-38-55] [Waiting State] Light Check Initialize(光センサーによる放出判定の処理を実行)
 [2019-8-22-14-38-55] [Waiting State] Abort Check 2.9 / 3600(Sub) 360(Last)
 [2019-8-22-14-38-56] [Waiting State] Light Check Update 1(s) / 10(s)
 [2019-8-22-14-38-56] [Waiting State] Abort Check 3.9 / 3600(Sub) 360(Last)
 [2019-8-22-14-38-57] [Waiting State] Light Check Update 2(s) / 10(s)
 [2019-8-22-14-38-57] [Waiting State] Abort Check 5.0 / 3600(Sub) 360(Last)
 [2019-8-22-14-38-59] [Waiting State] Light Check Failed.
 [2019-8-22-14-38-59] [Waiting State] Abort Check 6.2 / 3600(Sub) 360(Last)
 [2019-8-22-14-39-0] [Waiting State] Light Check Initialize
 [2019-8-22-14-39-0] [Waiting State] Abort Check 7.2 / 3600(Sub) 360(Last)
 [2019-8-22-14-39-1] [Waiting State] Light Check Update 1(s) / 10(s)
 [2019-8-22-14-39-1] [Waiting State] Abort Check 8.2 / 3600(Sub) 360(Last)
 [2019-8-22-14-39-2] [Waiting State] Light Check Update 2(s) / 10(s)
 [2019-8-22-14-39-2] [Waiting State] Abort Check 9.3 / 3600(Sub) 360(Last)
 [2019-8-22-14-39-3] [Waiting State] Light Check Update 3(s) / 10(s)
 [2019-8-22-14-39-3] [Waiting State] Abort Check 10.3 / 3600(Sub) 360(Last)
 [2019-8-22-14-39-4] [Waiting State] Light Check Update 4(s) / 10(s)
 [2019-8-22-14-39-4] [Waiting State] Abort Check 11.3 / 3600(Sub) 360(Last)
 [2019-8-22-14-39-5] [Waiting State] Light Check Update 5(s) / 10(s)
 [2019-8-22-14-39-5] [Waiting State] Abort Check 12.4 / 3600(Sub) 360(Last)
 [2019-8-22-14-39-6] [Waiting State] Light Check Update 6(s) / 10(s)
 [2019-8-22-14-39-6] [Waiting State] Abort Check 13.5 / 3600(Sub) 360(Last)
 [2019-8-22-14-39-7] [Waiting State] Light Check Update 7(s) / 10(s)
 [2019-8-22-14-39-7] [Waiting State] Abort Check 14.5 / 3600(Sub) 360(Last)
 [2019-8-22-14-39-8] [Waiting State] Light Check Update 8(s) / 10(s)
 [2019-8-22-14-39-8] [Waiting State] Abort Check 15.5 / 3600(Sub) 360(Last)
 [2019-8-22-14-39-9] [Waiting State] Light Check Update 9(s) / 10(s)
 [2019-8-22-14-39-9] [Waiting State] Abort Check 16.6 / 3600(Sub) 360(Last)
 [2019-8-22-14-39-10] [Waiting State] Light Check Update 10(s) / 10(s)
 [2019-8-22-14-39-10] [Waiting State] Abort Check 17.6 / 3600(Sub) 360(Last)
 [2019-8-22-14-39-11] [Waiting State] Light Check Success.(放出判定が10秒間連続で成功したため、落下状態に移行)
 [2019-8-22-14-39-11] -----
 [2019-8-22-14-39-11] [Falling State] Start(CanSatが落下状態)
 [2019-8-22-14-39-11] -----
 [2019-8-22-14-39-11] Time --> 2019:8:22:14:39:11
 [2019-8-22-14-39-11] [Waiting State] Finished
 [2019-8-22-14-39-12] Gyro Check Initialize(ジャイロセンサーによる着地判定の処理を実行)
 [2019-8-22-14-39-12] Pressure Check Initialize(大気圧センサーによる着地判定の処理を実行)

行)

(動画内で確認できるが、落下を模擬するため、CanSatをわざと揺らしているため、ジャイロセンサーの着地判定が失敗し続ける。)

[2019-8-22-14-39-12] [Falling State] Duration 1.056658(s) / Sub 900(s) / Abort 1200(s)
[2019-8-22-14-39-13] Gyro Check Failed. (Gyro Diff: 4.095920 > 1.000000)
[2019-8-22-14-39-13] Pressure Check Update 1(s) / 10(s) (Pressure Diff: 0.094421 < 1.000000)
[2019-8-22-14-39-13] [Falling State] Duration 2.222985(s) / Sub 900(s) / Abort 1200(s)
[2019-8-22-14-39-14] Gyro Check Initialize
[2019-8-22-14-39-15] Pressure Check Update 2(s) / 10(s) (Pressure Diff: 0.846497 < 1.000000)
[2019-8-22-14-39-15] [Falling State] Duration 3.313007(s) / Sub 900(s) / Abort 1200(s)
[2019-8-22-14-39-16] Gyro Check Failed. (Gyro Diff: 3.931040 > 1.000000)
[2019-8-22-14-39-16] Pressure Check Update 3(s) / 10(s) (Pressure Diff: 0.845337 < 1.000000)
[2019-8-22-14-39-16] [Falling State] Duration 4.319047(s) / Sub 900(s) / Abort 1200(s)
[2019-8-22-14-39-17] Gyro Check Initialize
[2019-8-22-14-39-17] Pressure Check Update 4(s) / 10(s) (Pressure Diff: 0.027466 < 1.000000)
[2019-8-22-14-39-17] [Falling State] Duration 5.367964(s) / Sub 900(s) / Abort 1200(s)
[2019-8-22-14-39-18] Gyro Check Failed. (Gyro Diff: 1.376517 > 1.000000)
[2019-8-22-14-39-18] Pressure Check Update 5(s) / 10(s) (Pressure Diff: 0.101624 < 1.000000)
[2019-8-22-14-39-18] [Falling State] Duration 6.368411(s) / Sub 900(s) / Abort 1200(s)
[2019-8-22-14-39-19] Gyro Check Initialize
[2019-8-22-14-39-19] Pressure Check Update 6(s) / 10(s) (Pressure Diff: 0.750000 < 1.000000)
[2019-8-22-14-39-19] [Falling State] Duration 7.599688(s) / Sub 900(s) / Abort 1200(s)
[2019-8-22-14-39-20] Gyro Check Update 1(s) / 10(s) (Gyro Diff: 0.446643 < 1.000000)
[2019-8-22-14-39-20] Pressure Check Update 7(s) / 10(s) (Pressure Diff: 0.773682 < 1.000000)
[2019-8-22-14-39-20] [Falling State] Duration 8.713904(s) / Sub 900(s) / Abort 1200(s)
[2019-8-22-14-39-21] Gyro Check Failed. (Gyro Diff: 1.025190 > 1.000000)
[2019-8-22-14-39-21] Pressure Check Update 8(s) / 10(s) (Pressure Diff: 0.027466 < 1.000000)
[2019-8-22-14-39-21] [Falling State] Duration 9.773753(s) / Sub 900(s) / Abort 1200(s)
[2019-8-22-14-39-22] Gyro Check Initialize
[2019-8-22-14-39-22] Pressure Check Update 9(s) / 10(s) (Pressure Diff: 0.008240 < 1.000000)
[2019-8-22-14-39-22] [Falling State] Duration 10.806224(s) / Sub 900(s) / Abort 1200(s)
(この時点でCanSatを静止させたため、以降ではジャイロセンサーの着地判定が成功し続ける)
[2019-8-22-14-39-23] Gyro Check Update 1(s) / 10(s) (Gyro Diff: 0.029650 < 1.000000)
[2019-8-22-14-39-23] Pressure Check Update 10(s) / 10(s) (Pressure Diff: 0.118164 < 1.000000)(大気圧センサーの着地判定に10秒間成功)
[2019-8-22-14-39-23] [Falling State] Duration 11.807373(s) / Sub 900(s) / Abort 1200(s)
[2019-8-22-14-39-24] Gyro Check Update 2(s) / 10(s) (Gyro Diff: 0.006633 < 1.000000)
[2019-8-22-14-39-24] Pressure Check Success
[2019-8-22-14-39-24] [Falling State] Duration 12.872007(s) / Sub 900(s) / Abort 1200(s)
[2019-8-22-14-39-25] Gyro Check Update 3(s) / 10(s) (Gyro Diff: 0.033256 < 1.000000)
[2019-8-22-14-39-25] [Falling State] Duration 13.933354(s) / Sub 900(s) / Abort 1200(s)
[2019-8-22-14-39-26] Gyro Check Update 4(s) / 10(s) (Gyro Diff: 0.021018 < 1.000000)
[2019-8-22-14-39-26] [Falling State] Duration 15.000500(s) / Sub 900(s) / Abort 1200(s)
[2019-8-22-14-39-27] Gyro Check Update 5(s) / 10(s) (Gyro Diff: 0.002659 < 1.000000)
[2019-8-22-14-39-27] [Falling State] Duration 16.049722(s) / Sub 900(s) / Abort 1200(s)

[2019-8-22-14-39-28] Gyro Check Update 6(s) / 10(s) (Gyro Diff: 0.001754 < 1.000000)
 [2019-8-22-14-39-28] [Falling State] Duration 17.050172(s) / Sub 900(s) / Abort 1200(s)
 [2019-8-22-14-39-29] Gyro Check Update 7(s) / 10(s) (Gyro Diff: 0.001005 < 1.000000)
 [2019-8-22-14-39-29] [Falling State] Duration 18.111667(s) / Sub 900(s) / Abort 1200(s)
 [2019-8-22-14-39-30] Gyro Check Update 8(s) / 10(s) (Gyro Diff: 0.000883 < 1.000000)
 [2019-8-22-14-39-30] [Falling State] Duration 19.173814(s) / Sub 900(s) / Abort 1200(s)
 [2019-8-22-14-39-31] Gyro Check Update 9(s) / 10(s) (Gyro Diff: 0.001760 < 1.000000)
 [2019-8-22-14-39-31] [Falling State] Duration 20.226724(s) / Sub 900(s) / Abort 1200(s)
 [2019-8-22-14-39-32] Gyro Check Update 10(s) / 10(s) (Gyro Diff: 0.001300 < 1.000000)
 [2019-8-22-14-39-32] [Falling State] Duration 21.227616(s) / Sub 900(s) / Abort 1200(s)
 [2019-8-22-14-39-34] Gyro Check Success. (Gyro Diff: 0.001645 < 1.000000)
 [2019-8-22-14-39-34] [Falling State] Duration 22.517875(s) / Sub 900(s) / Abort 1200(s)(ジャ
 イロセンサーの着地判定も10秒間連続で成功したため、切り離し状態に移行)
 [2019-8-22-14-39-34] -----
 [2019-8-22-14-39-34] [Separating State] Start(CanSatが切り離し状態)
 [2019-8-22-14-39-34] -----
 [2019-8-22-14-39-34] Time --> 2019:8:22:14:39:34
 [2019-8-22-14-39-34] [Falling State] Finished
 [2019-8-22-14-39-35] [Separating State] Separating Count (1/16)
 [2019-8-22-14-39-36] [Separating State] Separating Count (2/16)
 [2019-8-22-14-39-37] [Separating State] Separating Count (3/16)
 [2019-8-22-14-39-38] [Separating State] Separating Count (4/16)
 [2019-8-22-14-39-40] [Separating State] Separating Count (5/16)
 [2019-8-22-14-39-41] [Separating State] Separating Count (6/16)
 [2019-8-22-14-39-42] [Separating State] Separating Count (7/16)
 [2019-8-22-14-39-43] [Separating State] Separating Count (8/16)
 [2019-8-22-14-39-44] [Separating State] Separating Count (9/16)
 [2019-8-22-14-39-45] [Separating State] Separating Count (10/16)
 [2019-8-22-14-39-46] [Separating State] Separating Count (11/16)
 [2019-8-22-14-39-47] [Separating State] Separating Count (12/16)
 [2019-8-22-14-39-48] [Separating State] Separating Count (13/16)
 [2019-8-22-14-39-49] [Separating State] Separating Count (14/16)
 [2019-8-22-14-39-50] [Separating State] Separating Count (15/16)
 [2019-8-22-14-39-51] [Separating State] Separating Count (16/16)
 [2019-8-22-14-39-51] [Separating State] Separating Finished
 [2019-8-22-14-39-52] [Separating State] Opening Count (1/20) (Cansatが外骨格脱出をして
 いる状態)
 [2019-8-22-14-39-53] [Separating State] Opening Count (2/20)
 [2019-8-22-14-39-54] [Separating State] Opening Count (3/20)
 [2019-8-22-14-39-55] [Separating State] Opening Count (4/20)
 [2019-8-22-14-39-56] [Separating State] Opening Count (5/20)
 [2019-8-22-14-39-57] [Separating State] Opening Count (6/20)
 [2019-8-22-14-39-58] [Separating State] Opening Count (7/20)
 [2019-8-22-14-39-59] [Separating State] Opening Count (8/20)
 [2019-8-22-14-40-0] [Separating State] Opening Count (9/20)
 [2019-8-22-14-40-1] [Separating State] Opening Count (10/20)
 [2019-8-22-14-40-3] [Separating State] Opening Count (11/20)

[2019-8-22-14-40-4] [Separating State] Opening Count (12/20)
[2019-8-22-14-40-5] [Separating State] Opening Count (13/20)
[2019-8-22-14-40-6] [Separating State] Opening Count (14/20)
[2019-8-22-14-40-7] [Separating State] Opening Count (15/20)
[2019-8-22-14-40-8] [Separating State] Opening Count (16/20)
[2019-8-22-14-40-9] [Separating State] Opening Count (17/20)
[2019-8-22-14-40-10] [Separating State] Opening Count (18/20)
[2019-8-22-14-40-11] [Separating State] Opening Count (19/20)
[2019-8-22-14-40-12] [Separating State] Opening Count (20/20)
[2019-8-22-14-40-12] [Separating State] Opening Finished
[2019-8-22-14-40-13] [Separating State] Fighting Free Count (1/16)
[2019-8-22-14-40-14] [Separating State] Fighting Free Count (2/16)
[2019-8-22-14-40-15] [Separating State] Fighting Free Count (3/16)
[2019-8-22-14-40-16] [Separating State] Fighting Free Count (4/16)
[2019-8-22-14-40-17] [Separating State] Fighting Free Count (5/16)
[2019-8-22-14-40-18] [Separating State] Fighting Free Count (6/16)
[2019-8-22-14-40-20] [Separating State] Fighting Free Count (7/16)
[2019-8-22-14-40-21] [Separating State] Fighting Free Count (8/16)
[2019-8-22-14-40-22] [Separating State] Fighting Free Count (9/16)
[2019-8-22-14-40-23] [Separating State] Fighting Free Count (10/16)
[2019-8-22-14-40-24] [Separating State] Fighting Free Count (11/16)
[2019-8-22-14-40-25] [Separating State] Fighting Free Count (12/16)
[2019-8-22-14-40-26] [Separating State] Fighting Free Count (13/16)
[2019-8-22-14-40-27] [Separating State] Fighting Free Count (14/16)
[2019-8-22-14-40-28] [Separating State] Fighting Free Count (15/16)
[2019-8-22-14-40-29] [Separating State] Fighting Free Count (16/16)
[2019-8-22-14-40-29] [Separating State] Fighting Free Finished
[2019-8-22-14-40-30] [Separating State] Getting Distance Motor Right
[2019-8-22-14-40-30] [Separating State] Getting Distance Count (1/10)
[2019-8-22-14-40-31] [Separating State] Getting Distance servo release
[2019-8-22-14-40-31] [Separating State] Getting Distance Count (2/10)
[2019-8-22-14-40-32] [Separating State] Getting Distance Motor Right
[2019-8-22-14-40-32] [Separating State] Getting Distance Count (3/10)
[2019-8-22-14-40-33] [Separating State] Getting Distance servo release
[2019-8-22-14-40-33] [Separating State] Getting Distance Count (4/10)
[2019-8-22-14-40-34] [Separating State] Getting Distance Motor Right
[2019-8-22-14-40-34] [Separating State] Getting Distance Count (5/10)
[2019-8-22-14-40-35] [Separating State] Getting Distance servo release
[2019-8-22-14-40-35] [Separating State] Getting Distance Count (6/10)
[2019-8-22-14-40-36] [Separating State] Getting Distance Motor Right
[2019-8-22-14-40-36] [Separating State] Getting Distance Count (7/10)
[2019-8-22-14-40-37] [Separating State] Getting Distance servo release
[2019-8-22-14-40-37] [Separating State] Getting Distance Count (8/10)
[2019-8-22-14-40-38] [Separating State] Getting Distance Motor Right
[2019-8-22-14-40-38] [Separating State] Getting Distance Count (9/10)
[2019-8-22-14-40-39] [Separating State] Getting Distance servo release
[2019-8-22-14-40-39] [Separating State] Getting Distance Count (10/10)

[2019-8-22-14-40-39] [Separating State] Getting Distance Finished
 [2019-8-22-14-40-40] [Separating State] armor detected
 [2019-8-22-14-40-42] [Separating State] found safe way to run !
 [2019-8-22-14-40-48] [Separating State] Finished
 [2019-8-22-14-40-48] -----
 [2019-8-22-14-40-48] [Navigating State] Start(**CanSatがナビゲーション状態**)
 [2019-8-22-14-40-48] -----
 [2019-8-22-14-40-48] Time --> 2019:8:22:14:40:48
 [2019-8-22-14-40-48] [Navi] mMidDistance = 14.942477
 [2019-8-22-14-40-48] [Navi] InitialRunWhile...
 [2019-8-22-14-40-52] [Navi] distance;30.485108 middistance;14.942477
 [2019-8-22-14-40-52] [Navi] FarGoalNavi
 [2019-8-22-14-40-52] [Navi] Speed: 0.159000
 [2019-8-22-14-40-52] [Navi] Distance: 30.485108
 [2019-8-22-14-40-52] [Navi] Goal Angle: 109.027201 Rover Angle: -89.999999 Delta Angle: 90.000000(Left)
 [2019-8-22-14-40-52] [Navi] current: 90.000000 target: 0.000000 inc: 0.200000
 [2019-8-22-14-40-52] [Navi] distance;30.655825 middistance;14.942477
 [2019-8-22-14-40-52] [Navi] FarGoalNavi
 [2019-8-22-14-40-52] [Navi] Speed: 0.159000
 [2019-8-22-14-40-52] [Navi] Distance: 30.655825
 [2019-8-22-14-40-52] [Navi] Goal Angle: 109.027201 Rover Angle: -89.999999 Delta Angle: 90.000000(Left)
 [2019-8-22-14-40-53] [Navi] current: 90.000000 target: 0.000000 inc: 0.200000
 [2019-8-22-14-40-53] [Navi] distance;30.826654 middistance;14.942477
 [2019-8-22-14-40-53] [Navi] FarGoalNavi
 [2019-8-22-14-40-53] [Navi] Speed: 0.190000
 [2019-8-22-14-40-53] [Navi] Distance: 30.826654
 [2019-8-22-14-40-53] [Navi] Goal Angle: 108.936210 Rover Angle: -89.999999 Delta Angle: 90.000000(Left)

(中略)

[2019-8-22-14-43-13] [Navi] current: 30.636391 target: 0.000000 inc: 0.068081
 [2019-8-22-14-43-14] [Navi] distance;14.701006 middistance;14.942477
 [2019-8-22-14-43-14] [Navi] MidPoint reached!
 [2019-8-22-14-43-14] [Navi] Digging
 [2019-8-22-14-43-14] -----
 [2019-8-22-14-43-14] [Digging] Start(**CanSatが掘削状態に移行**)
 [2019-8-22-14-43-14] -----
 [2019-8-22-14-43-14] Time --> 2019:8:22:14:43:14
 [2019-8-22-14-43-15] [Digging] Digging Start
 [2019-8-22-14-43-16] [Digging] IncWorm Loop Count 0 / 20
 [2019-8-22-14-43-18] [Digging] IncWorm Loop Count 1 / 20
 [2019-8-22-14-43-20] [Digging] IncWorm Loop Count 2 / 20
 [2019-8-22-14-43-22] [Digging] IncWorm Loop Count 3 / 20
 [2019-8-22-14-43-24] [Digging] IncWorm Loop Count 4 / 20


```

[2019-8-22-14-43-26] [Digging] IncWorm Loop Count 5 / 20
[2019-8-22-14-43-28] [Digging] IncWorm Loop Count 6 / 20
[2019-8-22-14-43-31] [Digging] IncWorm Loop Count 7 / 20
[2019-8-22-14-43-33] [Digging] IncWorm Loop Count 8 / 20
[2019-8-22-14-43-35] [Digging] IncWorm Loop Count 9 / 20
[2019-8-22-14-43-37] [Digging] IncWorm Loop Count 10 / 20
[2019-8-22-14-43-39] [Digging] IncWorm Loop Count 11 / 20
[2019-8-22-14-43-41] [Digging] IncWorm Loop Count 12 / 20
[2019-8-22-14-43-43] [Digging] IncWorm Loop Count 13 / 20
[2019-8-22-14-43-46] [Digging] IncWorm Loop Count 14 / 20
[2019-8-22-14-43-48] [Digging] IncWorm Loop Count 15 / 20
[2019-8-22-14-43-50] [Digging] IncWorm Loop Count 16 / 20
[2019-8-22-14-43-52] [Digging] IncWorm Loop Count 17 / 20
[2019-8-22-14-43-54] [Digging] IncWorm Loop Count 18 / 20
[2019-8-22-14-43-56] [Digging] IncWorm Loop Count 19 / 20
[2019-8-22-14-43-58] [Digging] IncWorm Loop Count 20 / 20
[2019-8-22-14-44-0] [Digging] Finished !
[2019-8-22-14-44-0] -----
[2019-8-22-14-44-0] [Digging] Finished(CanSatが掘削状態が終了)
[2019-8-22-14-44-0] -----
[2019-8-22-14-44-1] [Navi] Freeze
[2019-8-22-14-44-1] [Navi] distance;12.044681 middistance;14.942477
[2019-8-22-14-44-1] [Navi] FarGoalNavi

(中略)

[2019-8-22-14-44-23] [Navi] current: -41.660121 target: 0.000000 inc: -0.092578
[2019-8-22-14-44-23] [Navi] distance;4.618702 middistance;14.942477
[2019-8-22-14-44-23] [Navi] FarGoalNavi
[2019-8-22-14-44-23] [Navi] Speed: 0.648000
[2019-8-22-14-44-23] [Navi] Distance: 4.618702 (ゴールから4.618702mと判定)
[2019-8-22-14-44-23] [Navi] Goal Angle: 87.852448 Rover Angle: 72.894793 Delta Angle:
-14.957654(Right)
[2019-8-22-14-44-23] [Navi] current: -14.957654 target: 0.000000 inc: -0.033239
[2019-8-22-14-44-24] [Navi] Far Mode Goal(ゴール判定)
[2019-8-22-14-44-24] [Navi] Navigating Finish Point:(35.659387 139.541652)
[2019-8-22-14-44-24] Time --> 2019:8:22:14:44:24
[2019-8-22-14-44-24] [Navi] Finished(ナビゲーション状態を終了)

```

Fig. 6.21.1にCanSatの制御履歴から得られたGPS座標をプロットした。CanSatの軌跡を赤色のプロットで示す。また、赤色のピンでゴール座標を示す。スタートしてから、公園端の花壇に侵入した後、ゴールに向かう様子が軌跡から確認できる。



Fig. 6.21.1 走行軌跡

- 結論

CanSatの各シーケンス内の制御をログから詳細に読み取れることが示された。また、GPSの位置座標のログから、CanSatが方向制御しゴールに向かう様子をプロットできたことが確認できた。以上より、規定された制御履歴レポートが作成可能であることが示された。

第7章 工程管理、ガントチャート（スプレッドシートを推奨）

1. チーム内・審査会等

（チーム全体、班、各部署ごとの会議の頻度、開発メンバー以外の人を含めた審査会の有無や時期、回数を記載してください）

- MTG関連
 - 週1回チームMTGを実施し、渡米前までに25回実施した。
 - 週1回研究室内全体MTGを実施し、渡米前までに25回実施した。
- 各フェーズの大まかなスケジュール
 - ミッションや機体の構想開始・・・2019年3月
 - 設計開始・・・2019年4月
 - BBM(機能モデル)製作開始・・・2019年5月1日～5月31日
 - BBM(機能モデル)検証完了・・・2019年6月9日
 - EFM(実物大モデル)製作開始・・・2019年6月10日～6月22日
 - 実験と機体の微調整の繰り返し・・・2019年6月23日～8月20日
 - EFM(実物大モデル)検証完了・・・2019年8月22日

2. 各担当（ハード・ソフト・全体などの進行状況・予定を記入）

チーム内で用いたガントチャートへのリンクを以下に示す。

https://docs.google.com/spreadsheets/u/1/d/1coKfG6m2j3QF_WnoUUr6RsHhEojxS3lrnnP0C7Llu9E/edit#gid=1447771871

第8章 大会結果

1. 能代宇宙イベント

□ 目的

同じCanSat開発仲間との交流や技術の共有、さらに、ARLISSへ向けて開発したCanSatがレギュレーションを満たした上でのEnd-to-end試験や本番の予行として貴重な機会であるため、本大会に参加をした。

□ 結果

能代	パラシュート開傘	壊れずに着地	外骨格切り離し	走行開始	ゴールまでの中間地点まで制御走行	5cm掘削	10m以内ゴール
1回目	○	○	○	○	×	×	×
2回目	○	○	×	×	×	×	×

□ 取得データ

● 投下1回目

Fig8.1.1にプログラム起動開始(キャリア収納後)からミッションリタイヤ時までの気圧センサから取得した高度のログを示す。縦軸は高度(m), 横軸は日時を表す。Fig8.1.1の赤い丸印から、ドローンによって、実験場の地面から相対的にみて上空約40mまで上昇していることがわかる。

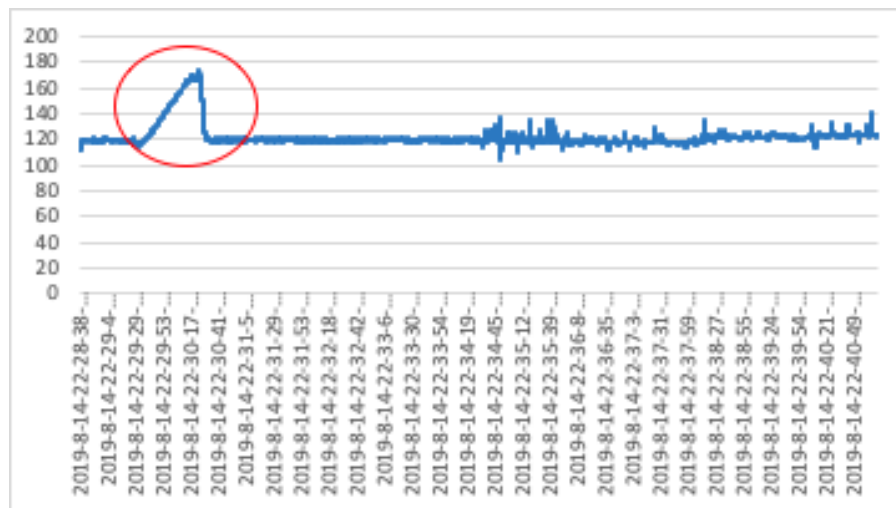


Fig8.1.1 投下1回目の高度のログ

Fig8.1.2にドローンによる上昇から着地後までの9軸センサから取得した値から算出した加速度のログを示す。縦軸が加速度(m/s^2), 横軸は日時を表す。Fig8.1.2の赤い3つの丸印から、ドローンによる上昇開始時、パラシュート開傘時、着地時に大きな加速度が生じていることがわかる。

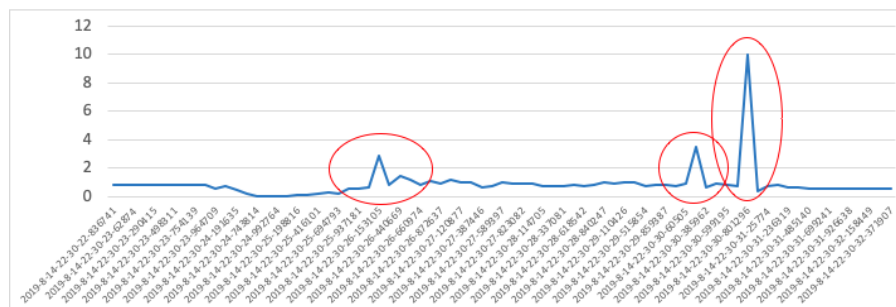


Fig8.1.2 投下1回目の加速度のログ

● 投下2回目

Fig8.1.3にプログラム起動開始(キャリア収納後)からミッションリタイヤ時までの気圧センサから取得した高度のログを示す。縦軸は高度(m), 横軸は日時を表す。図3.3.3の赤い丸印より、気球によって、実験場の地面から相対的にみて上空約30mまで上昇していることがわかる。

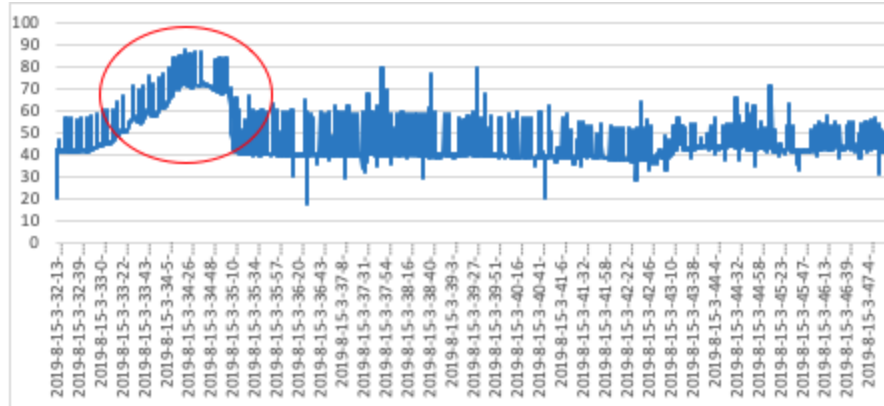


Fig8.1.3 投下2回目の高度のログ

Fig8.1.4に気球による上昇から着地後までの9軸センサから取得した値から算出した加速度のログを示す。縦軸が加速度(m/s^2)、横軸は日時を表す。図3.3.4の赤い3つの丸印から、気球による上昇開始時、パラシュート開傘時、着地時に大きな加速度が生じていることがわかる。

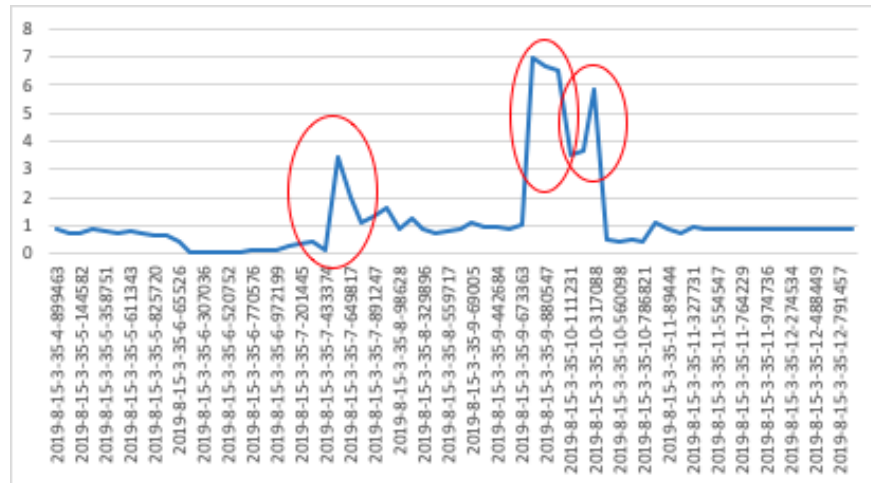


Fig8.1.4 投下2回目の加速度のログ

□ 故障原因解析・解決手段等

● 投下1回目

着地後に走行を行うことができた。しかし、走行時に横転し地面側になったタイヤが空転しその状態から抜け出すことが不可能となった。このとき、モータからの駆動力が複数のギアを介してカップリングから車軸へ伝達する部分に問題が生じていた。この問題点は、車軸をカップリングに接合するためにD字に削り面を出す工程が十分に行われていなかったことにある。この問題発覚後、車軸のD字加工を十分に行い、車軸の空転は解消された。

● 投下2回目

外骨格から脱出することができず、その後のミッションを続行することが不可能となった。着地後にCanSat並びに外骨格が倒れたが外骨格脱出時に不適切な状態であったため外骨格から脱出することができなかった。具体的には、外骨格からの脱出には、

サーボモータを駆動させてピンを引き抜く動作をするが、その際にサーボモータが駆動する方向に多量の草がありサーボモータを正常に駆動させることができなかった。

2. ARLISS

□ 目的

開発したCanSatの有効性を検証するため、ロケットに搭載し、打ち上げてから、キャリア放出判定からゴール判定およびミッション達成までのシーケンスを自律して制御できるかを検証するために参加した。

□ 結果

ARLISS	パラシュート開傘	壊れずに着地	外骨格切り離し	走行開始	ゴールまでの中間地点まで制御走行	5cm掘削	10m以内ゴール	0mゴール
1回目	○	○	×	×	×	×	×	×
2回目	○	○	○	×	×	×	×	×
3回目	○	○	○	×	×	×	×	×

□ 取得データ

● 1st Flight

Fig8.2.1にプログラム起動開始(キャリア収納後)からミッションリタイア時までの気圧センサから取得した高度のログを示す。縦軸は高度(m)、横軸は日時を表す。Fig8.2.1の赤い丸印から、ロケットによって、ブラックロック砂漠の地面から相対的にみて上空約3,000mまで上昇していることがわかる。

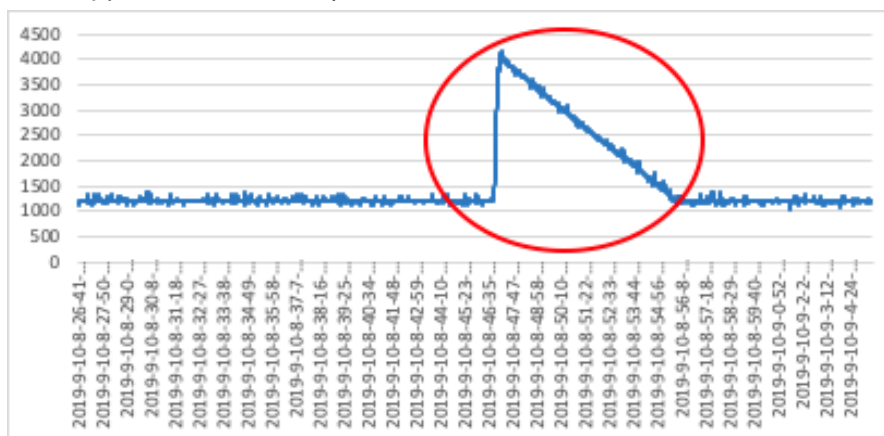


Fig8.2.1 1st Flight の高度のログ

Fig8.2.2にロケットによる上昇から着地後までの9軸センサから取得した値から算出した加速度のログに高度のログを重ね合わせたものを示す。左側の縦軸が高度(m)、右側の縦軸が加速度(m/s^2)、横軸は日時を表す。また、青線が高度のログを、オレンジ線が加速度を示している。Fig8.2.2の赤い3つの丸印から、ロケットによる上昇開始時、パラ

シュート開傘時，着地時に大きな加速度が生じていることがわかる。

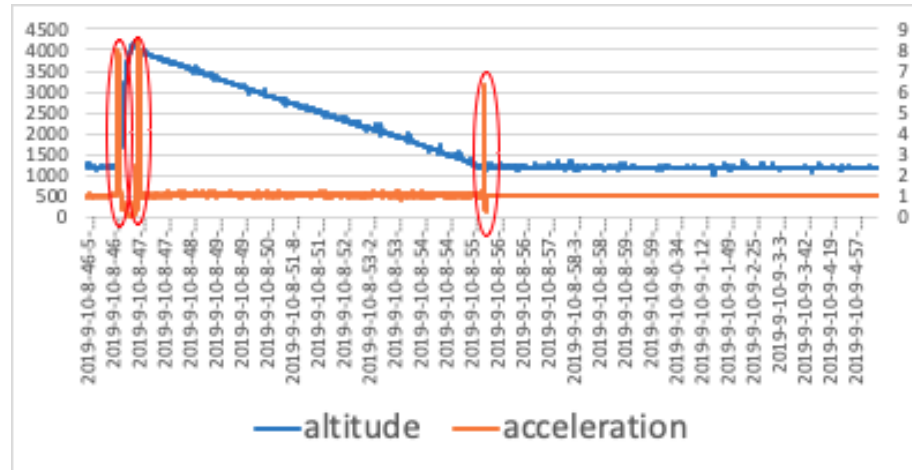


Fig8.2.2 1st Flight の加速度のログ

Fig8.2.3にロケットのキャリアから放出された時から着地までのGPSのログを元に作成した軌跡を示す。CanSatがキャリアから放出されて初めて取得したGPS座標は緯度40.882005°，西経119.11383°であり，着地点のGPS座標は緯度40.875493°，西経119.10839°であった。

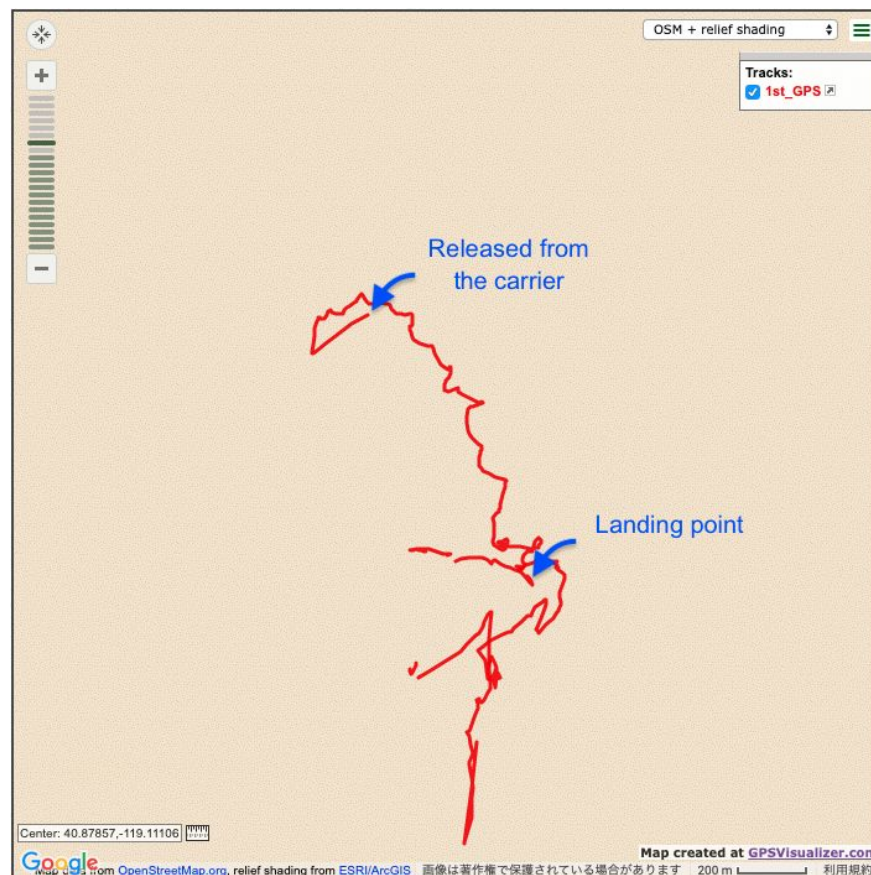


Fig8.2.3 1st Flight の落下中のGPSのログ

- 2nd Flight

Fig8.2.4にプログラム起動開始(キャリア収納後)からミッションリタイヤ時までの気圧センサから取得した高度のログを示す。縦軸は高度(m), 横軸は日時を表す。Fig8.2.4の赤い丸印から、ロケットによって、ブラックロック砂漠の地面から相対的にみて上空約3,500mまで上昇していることがわかる。

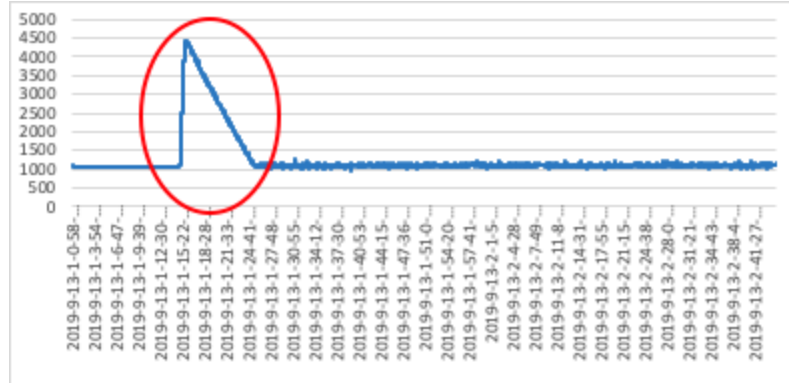


Fig8.2.4 2nd Flight の高度のログ

Fig8.2.5にロケットによる上昇から着地後までの9軸センサから取得した値から算出した加速度のログに高度のログを重ね合わせたものを示す。左側の縦軸が高度(m), 右側の縦軸が加速度(m/s^2), 横軸は日時を表す。また、青線が高度のログを、オレンジ線が加速度を示している。図4.3.5の赤い2つの丸印から、ロケットによる上昇開始時、パラシュート開傘時に大きな加速度が生じていることがわかる。また、パラシュート開傘時の衝撃によって9軸センサから取得できる値が不安定となっていることがわかる。後に、プログラムを再起動したところ9軸センサから取得できる値は安定していた。

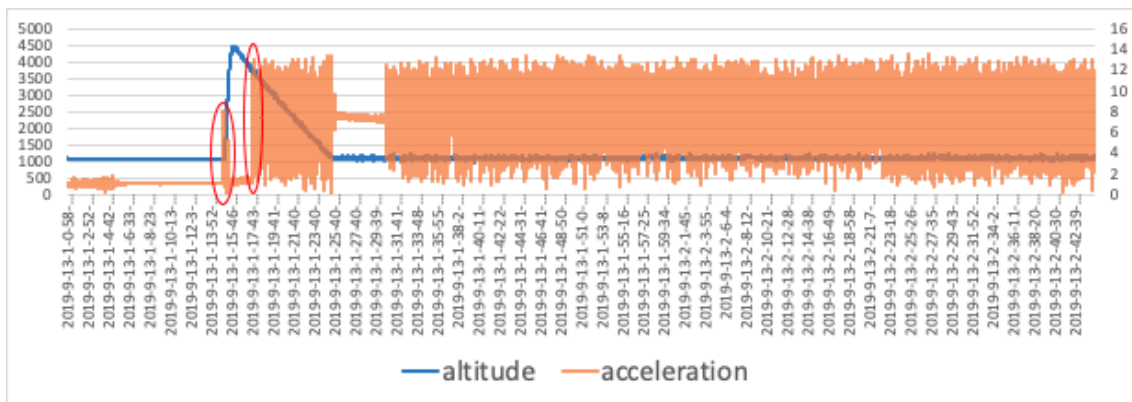


Fig8.2.5 2nd Flight の加速度と高度のログ

Fig8.2.6にロケットのキャリアから放出された時から着地までのGPSのログを元に作成した軌跡を示す。図4.3.3の地図と比較して縮尺が1/2となっていることに気がつけたい。CanSatがキャリアから放出されて初めて取得したGPS座標は緯度 40.879678°, 西経 119.10666°であり、着地点のGPS座標は緯度40.884237°, 西経 119.11292°であった。

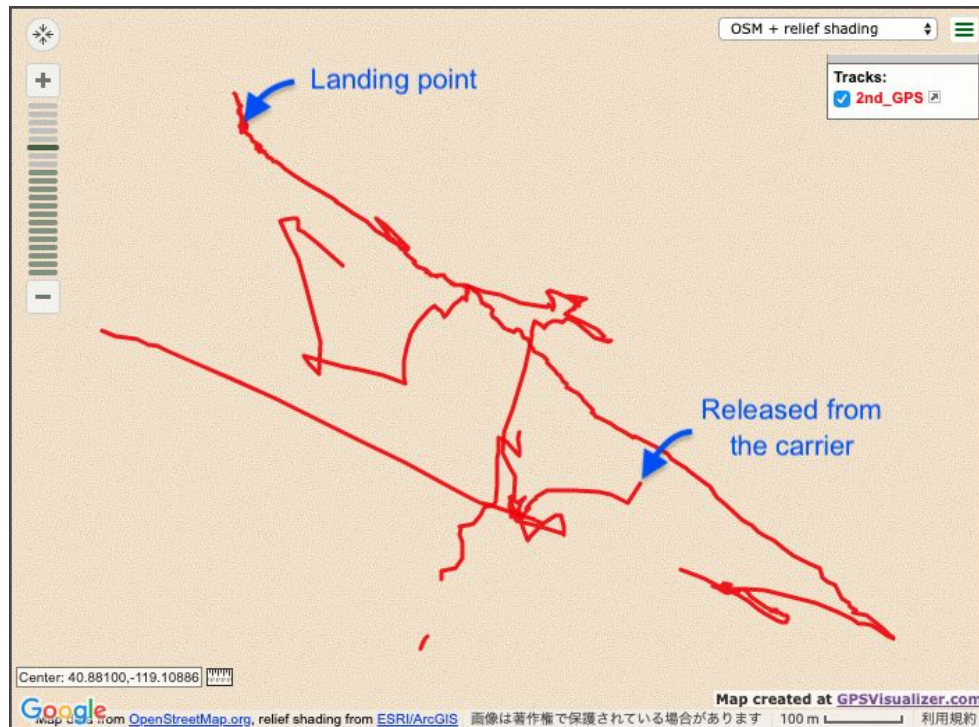


Fig8.2.6 2nd Flight の落下中のGPSのログ

● 3rd Flight

Fig8.2.7にプログラム起動開始(キャリア収納後)からミッションリタイヤ時までの気圧センサから取得した高度のログを示す。縦軸は高度(m)，横軸は日時を表す。Fig8.2.7の赤い丸印から，ロケットによって，ブラックロック砂漠の地面から相対的にみて上空約3,300mまで上昇していることがわかる。

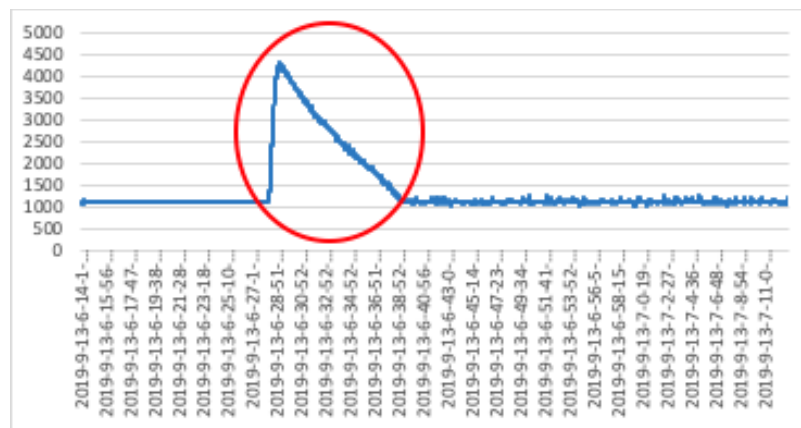


Fig8.2.7 3rd Flight の高度のログ

Fig8.2.8にロケットによる上昇から着地後までの9軸センサから取得した値から算出した加速度のログに高度のログを重ね合わせたものを示す。左側の縦軸が高度(m)，右側の縦軸が加速度(m/s^2)，横軸は日時を表す。また，青線が高度のログを，オレンジ線が加速度を示している。Fig8.2.8の赤い3つの丸印から，ロケットによる上昇開始時，パラシュート開傘時，着地時に大きな加速度が生じていることがわかる。また，着地時の衝

撃によって9軸センサから取得できる値が不安定となっていることがわかる。後に自律的にプログラムを再起動したところ、9軸センサから取得できる値は安定していた。

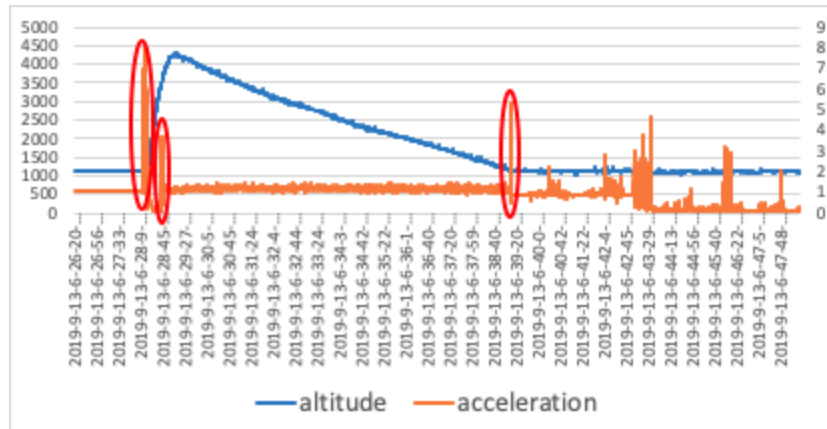


Fig8.2.8 3rd Flight の高度

Fig8.2.9にロケットのキャリアから放出された時から着地までのGPSのログを元に作成した軌跡を示す。Fig8.2.3の地図と比較して縮尺が1/2となっていることに気がつけたい。CanSatがキャリアから放出されて初めて取得したGPS座標は緯度 40.869558°, 西経 119.09528°であり、着地地点のGPS座標は緯度40.884237°, 西経 119.11292°であった。3回目の打ち上げでは、キャリアから放出された後すぐにGPS座標を受信できず、高度 3000mにおいて初めて受信した、また、軌跡が途切れているが、これは落下途中の高度 2,062mから1562mまでの間においてGPS座標を取得できなかったからである。



Fig8.2.9 3rd Flight の落下中のGPSのログ

□ 故障原因解析・解決手段等

● 1st Flight

無事に着地したが、外骨格切り離しをすることができず、外骨格の中でナビゲーションシーケンスへ移行し、外骨格脱出不可能であると考えミッションを終了した。外骨格切り離しの失敗の原因は、キャリア放出判定のタイムアウト時間の設定を誤ったというヒューマンエラーである。本来、ARLISSでのキャリア放出判定のタイムアウト時間は1時間であったが、能代において設定したタイムアウト時間、6分を修正せずにプログラムを起動した。よって、CanSatをキャリアに収納し、ロケットへ収納した後すぐにプログラムを起動したため、ロケットを立ち上げている頃に、キャリアの中で放出判定のタイムアウト時間を迎えた。その後、ロケットが打ち上がる前に、着地判定を通り抜け、外骨格切り離しのシーケンスに移行し、次にナビゲーションシーケンスへ移行したと考えられる。ロケットの中で外骨格切り離しのシーケンスを終えたため、落下中に外骨格を切り離し、CanSatが自由落下することはなかった。

このヒューマンエラーが起きた原因は、打ち上げ前に確認をしていたが、プログラム上の今回の原因に関係するタイムアウト時間の設定ファイルにおいて、使われていないタイムアウト時間の設定箇所もあり、それと、本来設定すべき箇所が混同したからである。今後、使われていない記述は削除またはコメントアウトするとともに、各設定箇所にコメント文を挿入することで誤りを未然に防ぐ必要がある。

● 2nd Flight

パラシュート切り離しまでの一連のシーケンスを問題なく実行することができた。しかし、4.3節の2nd Flightの図4.3.5の9軸センサから得られた値をもとに計算した加速度のログから分かるように、CanSatがパラシュート開傘時の衝撃を受けた時点で9軸センサの取得値に異常が発生した。これにより、CanSatが横転、反転の状態を正しく判定することができず、横転した状態で、前進、後進を繰り返した。しばらく、横転した状態で前進、後進を繰り返したため、次第にパラシュートの紐に近づき、紐がタイヤに絡まり、走行不能となった。(Fig. 8.2.10)



Fig. 8.2.10 パラシュートの紐がタイヤに絡まった様子

当初、2回目の打ち上げ直後では、場所によってキャリブレーションを実施しなければ、9軸センサが正常に動作しないという分析をしたため、プログラムに急遽変更し、場所による影響が少ない方法での横転・反転判定を実装し、同じ回路で3度目の打ち上げをした。そのため、3回目の打ち上げも同じような原因でミッション失敗した。

- **3rd Flight**

パラシュート切り離しまでの一連のシーケンスを問題なく実行することができた。しかし、4.3節の3rd Flightの図4.3.8の9軸センサから得られた値をもとに計算した加速度のログから分かるように、CanSatが着地時の衝撃を受けた時点で9軸センサの取得値に異常が発生した。これにより、CanSatが横転、反転の状態を正しく判定することができず、横転した状態で、前進、後進を繰り返した。しばらく、横転した状態で前進、後進を繰り返す途中に、CanSatの多関節機構に外骨格の蓋が徐々に干渉しはじめモータとサーボモータ共に負荷がかかり、次第に動かなくなった。(Fig. 8.2.11)この後、ナビゲーションシーケンスに移行してから30分間GPS座標の変化が100m以下であると、プログラムを再起動するというプログラムが動作した。これにより、9軸センサの異常は見られず、正しく横転判定をするようになったが、多関節機構に対して、大きな負荷が長時間加わったため、一部のサーボモータが保護モードに入り、力が伝わらなくなり、外骨格から離れることができなかった。後に、もう一度、プログラムが再起動され、保護モードが解除されたが、長時間モータとサーボモータに負荷が加わったため、駆動用バッテリーの残量がわずかとなり、走行不能となった。

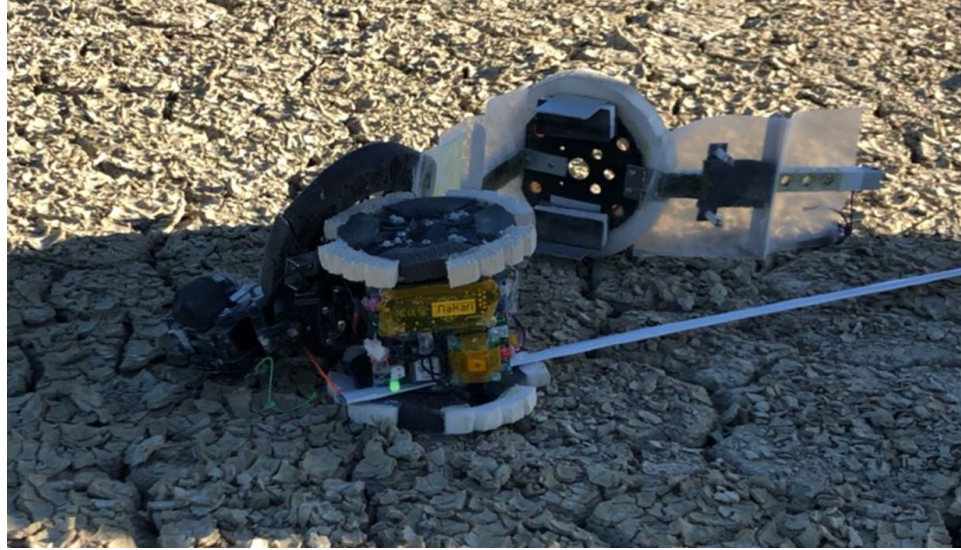


Fig. 8.2.11 外骨格の蓋と多関節機構が鑑賞している様子

2回目の打ち上げと同じく、9軸センサに衝撃が加わったことで9軸センサが取得する値に異常が見られた。2回目の打ち上げ後にしっかりログを分析することで、回路を変更、または、着地時に9軸センサを初期化するなどの対応ができた。しかし、今回できなかったのは、思い込みが原因である。打ち上げ後は、しっかりログを分析し、データをもとに判断するという基本ができていなかったため、2度も同じような失敗をした。今後、ログの分析をもとに判断すべきである。

第9章 まとめ

1. エフ・努力した点（ハード、ソフト、マネジメント面すべて）

【エフした点】

● ハード

- ギア4枚を介してモータの駆動力をタイヤに伝達するようにし機体全幅を小さく抑えられるようにした点.
- タイヤを拡張タイヤとしてタイヤ径を大きくしたことによって、キャリアに対して縦方向の収納では有るものの走行性能を確保できるようにした点.
- 外骨格の着地衝撃を吸収するためのスポンジを着地衝撃による跳ね返りを減らすために堅いスポンジを採用した点.
- タイヤの表面をギザギザにすることで、スパイクのようなアタッチメントを付けずに地面をかく力を強くした点.

● 回路

- GPS, ICS基板(サーボモータ制御用), LoRaの3つに対しての通信方式がシリアル通信であったため, Raspberry PiのTX, RXピン, USB経由, GPIOピン2つをソフトウェアのTX, RXピンとして使用することで3つのシリアル通信を可能とした点.
- Raspberry PiやGPSの下部に生じるスペース, 回路の裏面のスペースを有効に使うことで, たくさんの素子を搭載することを可能にした点.
- サーボモータを数珠つなぎにしてケーブル類の最適化をした点.
- 2×2の電池Boxと1×2の電池Boxをつなぎ合わせてL字状の電池Boxを自作することによって, 狭いスペースに6本の電池を収納できるようにした点.
- Raspberry Piの電源と動力系の電源を分けることによって, Raspberry Piの電源周りを安全にした点.
- サーボモータへの電圧変換をする際に, 昇降圧型のDC-DCコンバータを用いることによって, 常に7.4Vの供給を実現した点.

● ソフト

- 従来のPWMサーボからシリアルサーボに変更されたため, ソフトウェアのサーボ制御について1から構築した点
- サーボ制御するためのコマンド処理を隠ぺいし, ユーザがコマンドを意識せずに動作のための関数を利用するシステムを構築した点
- 複数のサーボを干渉なく同時制御するシステムを構築した点
- LoRaを制御するシステムを構築した点
- 従来のシステムではパラメータ設定もコンパイルする必要があったが, JSONで読み込み可能なシステムを構築した点
- 9軸センサの制御に外部のライブラリを用いることで, 信頼性のある値を取得可能にした点
- 距離センサのライブラリを取り込んだ点
- ナビゲーション制御において, 単純なPID制御で後輪モータの角度の変更をする
とタイヤの種類によって精度が大きく変化する問題があったが, PID制御の位置
と初期位置を交互に変更することでPID制御のパラメータ設定では難しい大きな
カーブを描く制御が可能になった. この制御により, ローカルの無駄な細かい制
御が制限され, 精度と省電力化向上につながった.
- 0mゴールを目指すため, ナビゲーション制御によるゴールに対する突入角度を
維持し, 回転を継続することでゴールがあるだろう円周を探索する制御を実装し
た

- 外骨格脱出のため、複数の脱出のシーケンスを設定し、脱出に必要な十分な行動をCanSatが実施することを可能にした
- ナビゲーションシーケンス中にGPSセンサ値の変化をみて、異常があった場合に自動再起動するシステムを構築した

【苦労した点】

- ハード
 - 外骨格とパラシュートの切り離しの成功率が低かった点
 - 外骨格が着地の衝撃により何度も開いてしまった点
- 回路
 - 限られたスペースにたくさんの素子を搭載した点.
 - 初期の頃、サーボモータのケーブルがなんども破損してしまった点.
 - ユニバーサル基板での開発時に、基板の両面に素子を配置していたため、デバックが大変だった点.
- ソフト
 - 従来用いてないセンサやサーボを利用したため、新たにハードウェア制御の処理を多く書く必要があった点
 - LoRaのデバイスのコマンド処理が不明慮だったため、LoRa制御の処理の実装に時間がかかった点
 - 外骨格脱出した後、外骨格とCanSatの距離を広くとる最適なシーケンスの模索に長時間かかった

2. 良かった点・課題点

【良かった点】

- ハード班
 - 作業量の多い中で、各人タスクをやり遂げた点
 - メンバー間のコミュニケーション、困っていることを抱え込まず共有することができた点
- 回路班
 - 回路の一部(ケーブルなど)が破損した際に、予備部品を準備していたため、すぐに修正することができていた.
 - 各コネクタに色づけをされており、誤って違う部分にコネクタを接続することがなかった.
 - 地上局用の通信基板を開発したおかげで、ソフト班が簡単に操作できるようになった.
- ソフト班
 - ソフト班の中で、片方はハードウェア制御、一方はシーケンス制御の処理を主に書くという役割分担がなされていたと思う
 - バグ自体はなかったと思われる

【課題点】

- ハード班
 - 各機体と外骨格の個体差
 - 機体、外骨格ともに細かなパーツが多く、設計図との誤差が多かったため個体差を生む原因となった。また、設計図なしに作っていたパーツもあったことが大きな原因でもある。全てのパーツがどの機体、外骨格にも使えるよう、設計図を徹底し、誤差が生まれにくいよう意識しながら製作しなくてはならない。

- また、組立時におけるネジの締め付けの強さや僅かな位置関係の差異が生じていても問題とならないような機構を採用することも必要である。必ずしも同じ人が同じ強さで同じように作業を行えるわけではないので、誰が作業を行っても同じような結果を得られるような設計が重要である。
 - モータ発注の遅れ
 - モータ発注のデッドラインをプロジェクト始動時にすぐに確認すべきであった。
 - 人員の管理
 - 各タスクに対して人という資源を有効に割り当てることができなかった。各メンバがどれだけの時間を作業にあてられるか不透明であったこと、各タスクがどれくらいの時間を要するか不明確であったことが一つの原因と考えられる。
- 回路班
 - Raspberry PiのGPIOピンについて、デフォルトの電圧考慮不足による、Raspberry Pi起動時にモータが回転してしまった点。
 - Raspberry Piに対してコマンドで、デフォルト値を確認したつもりであったが、CanSatのプログラムを一度起動していたため、各GPIOピンのデフォルト値が変更されていた。今後、各GPIOピンのデフォルト値を確認する際は、Raspberry Piを念の為に一度再起動する必要がある。
 - 反転時にケーブルなどが地面に当たる点。
 - シャーシ幅、ならびにタイヤ径の縮小の関係上、回路のケーブルや素子が地面に当たることがあり、その影響でケーブルが断線することが度々起きた。事前にタイヤ径がどれほどの大きさになるのかをハード班と連携して、回路設計をする必要があった。
 - ピンソケットを用いることで、一部のセンサが壊れた際にすぐ交換ができるようにしたが、一部のセンサが壊れるということが昨年度から引き続きあまり見られなかったため、安定性を求めるために、ピンソケットを用いずに直ではんだ付けしたほうが良かった。
- ソフト班
 - ソフト班がどちらも睡眠を重視するタイプだったため、双方とも寝ている時間が多かった。また、寝てしまうことで次の日までに実装または修正が間に合わないという自体が発生した。
 - 初期のROSの導入の試みは時間の無駄であった
 - ソフト班の中で実装について相談はあまりしなかった
 - 閾値のチェックをフライト前にならずやるべきであった

3. チームのマネジメント等、プロジェクト全体でのよかった点、反省点

【良かった点】

- 各開発班が隔たりなく、相談がよくできていた。特に、ハード面において抱えている問題を他班も親身になって考え開発を手伝っていた点が良かった。
- 授業関連の課題や論文執筆、各メンバの研究が忙しい時期に、チーム全体でカバーできていた。
- 全員が連絡をしっかりとっていた。
- 研究室内の過去の機体からヒントを得て開発をしており、研究室内の遺産を有効活用できていた。

- 度重なる試験の失敗に落ち込まずに、改善をすることができていた。

【反省点】

- 開傘衝撃試験，着地衝撃試験の失敗に伴うプロジェクトの停滞
 - 例年と異なる仕様を採用する際は，例年よりも早く試験に取り掛かり，早急に課題を見つけ取り組む必要がある。
- ログの解析がソフト班にしかできなかった。
 - ソフト班だけではなく，回路班，ハード班の人もログを解析できるようにすることで，回路面やハード面の問題に早く気づける可能性があるため，今後はソフト班以外もログ解析をできるようにするべきである。