優秀賞　賞金 10 万円

服は内　-衣類内気象モニタリング-

サレジオ工業高等専門学校

服は内

―衣服内モニタリング―

水口葵，　大庭康平，　島田茜，　渡邊真瑠弥
指導教員 吉田 将司
サレジオ工業高等専門学校 機械電子工学科 情報通信工学研究室

１．実施目的

夏場における健康被害でよく話題になるのは日焼け、熱中症、光化学スモッグである。これらの現象は目には見えず、すぐに被害が出るものが少ないため症状が悪化するまで身体の異変に気付くことがない。そのため毎年、夏場になるとこれらの健康被害が社会問題となっている。

そこで、昨年は野外環境での危険度の可視化を目的とした、ウェザーマルチメータの製作を行った。温湿度と紫外線量から日焼け、熱中症、不快指数に対する危険度をどこでも簡易的に測定できるシステムを製作し、その性能を評価した。図１はウェザーマルチメータを示す。

このウェザーマルチメータのメリットとして温湿度、紫外線を測定し、熱中症を予防する事ができるが、サイズが大きい事と手に持っていないと警告表示が確認できないものだった。

また、昨年のシステムでは外気の測定であり人体の変化に対応してなかったため、今回は個々の人体の変化に直接関係する指標である衣服内気候に注目した。図２は衣服内気候の測定イメージを示す。今回測定する衣服内気候とは、人体と衣服の微小な空間に生じる気候のことである[1)]。また衣服の開発において機能性や快適性の向上のため研究されている。

図1　ウェザーマルチメータ

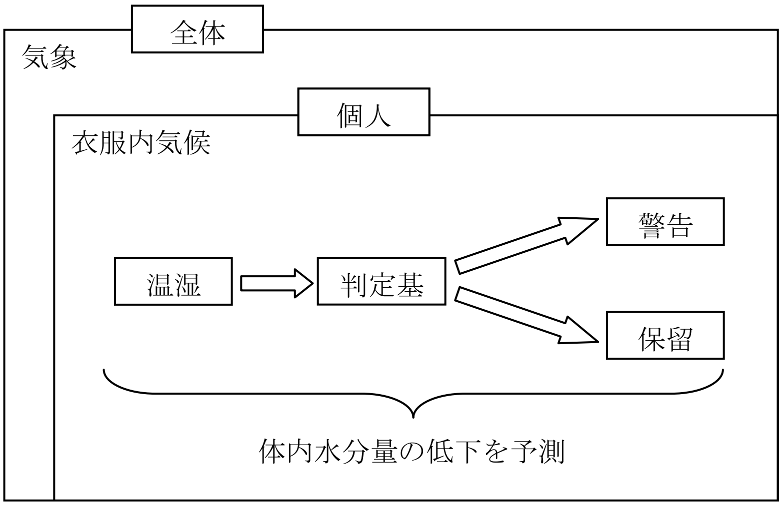


図２　衣服内気候のイメージ

本研究では屋内や屋外での活動において、衣服内温湿度の測定から体調の変化を可視化し、危険度を知らせる事で体調管理を行う事を目的とした。

図３は申請当初の機器のイメージを示す。外気の温湿度による人体への影響は個人差があるため、衣服内の温湿度を測定する事によって個々の体調変化を測定することが可能となる。特に幼児や高齢者など自身の体調の変化に気づきにくい被験者に有効である。そのため熱中症に陥る前に衣服内温湿度の変化から被験者に対し警告をだす機器を製作した。図４に申請当初の機器の装着方法を示す。

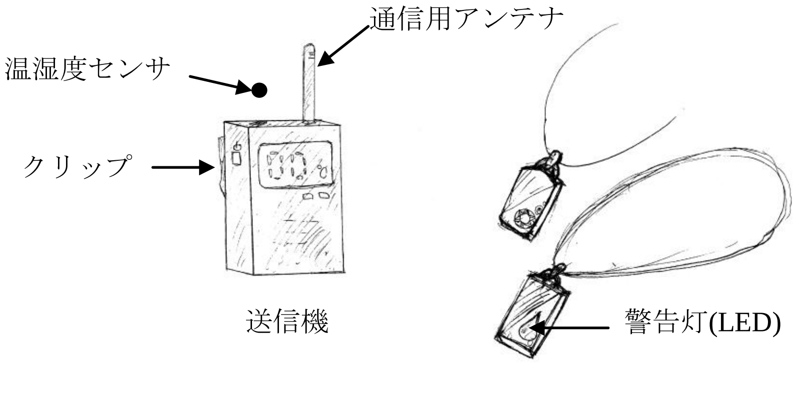


図３　機器のイメージ

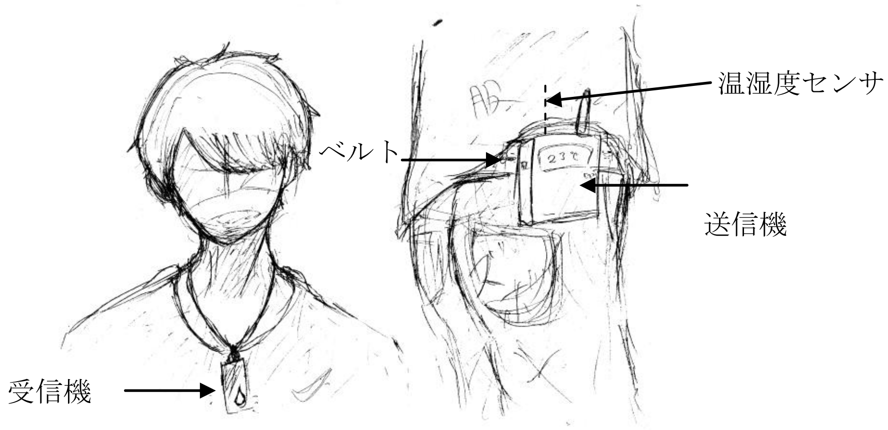


図４　使用例

この機器は皮膚温、ガルバニック皮膚応答（GSR）センサを用いた腕時計型の測定器や就寝中の衣服内気温の測定を行うデータロガーなどの従来品に比べ精度は劣るがコストを抑える事が可能となる。また、スマートフォンのアプリ等に比べ信頼性が高く安全に利用することができる。

２．測定原理

図５は測定方法のイメージを示す。本システムでは送信機のセンサを用いて衣服内温湿度を求め、その測定値から安全か危険かを判定する。判定基準は衣服内温湿度の快適値 [2)]から算出した。この快適値の領域は非線形であったが 2[℃]刻みで温度に対する湿度基準値を設定した。図６は危険値の判定基準を示す。判定は 29[℃]の時 67[%]以上、30[℃]で 72[%]以上、31[℃]で 76[%]以上、32～33[℃]で 78[%]以上、34[℃]で 75[%]以上を危険値とした。そして判定結果を受信機に送信し、受信機は危険値であればＬＥＤが赤色で点滅、安全値であれば緑色に点灯する。

図７はセンサの貼り付け方法を示す。センサは尾骶骨から 10cm の背中の位置に図７の様にテープで張り付ける。センサはプラスチックのカバー（3mm 厚）で覆い皮膚と衣服に触れない様にすることで衣服内気候の測定を行う。

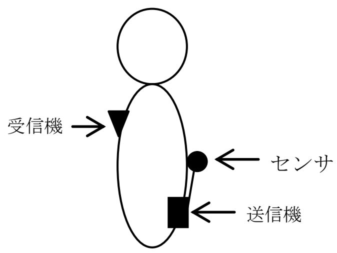


図５　イメージ

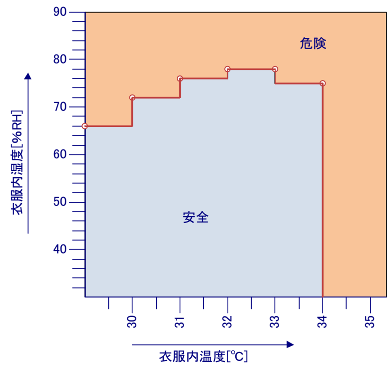


図６　判定基準

図７　センサの貼り付け

３．システム構成

　システム構成図を図８に示す。以下送信機と受信機の構成について説明する。送信機は検出部、演算部、通信部の３つの機能からなるシステムを試作した。

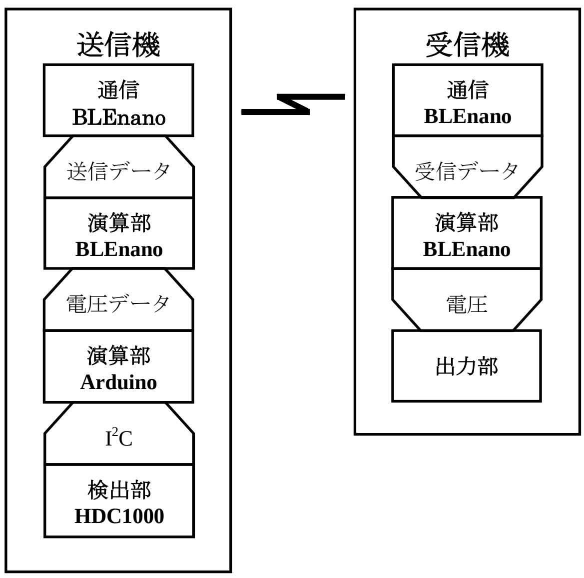


図８　本システムのブロック図

３－１　送信機の構成

図９は送信機の状態遷移図を示す。待機状態から計測を開始し危険度の判別を行いその結果を送信する。

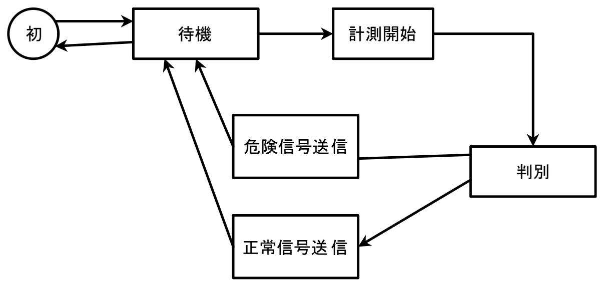


図９　送信機状態遷移

図１０は送信機の具体化したフローチャートを示す。プログラムが起動したらまず計測、判定を行い図６の判定基準から判定を行い信号の送信を行う。

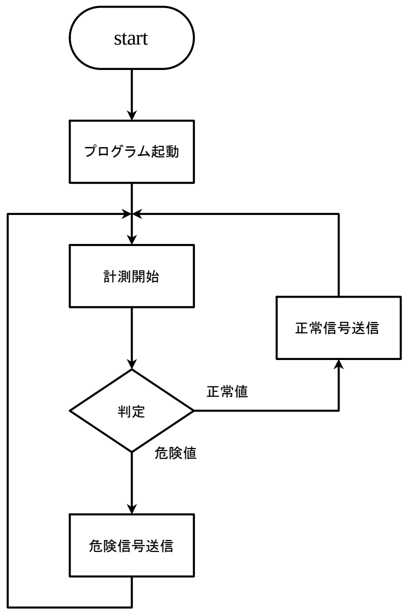


図１０　送信機フローチャート

送信機の検出部には温湿度センサ HDC1000 を使用した。温湿度センサ HDC1000 を図１１に示す。気温の測定範囲が-20[℃]～85[℃]、湿度は相対湿度で 0[%]～100[%]であり I^2C による出力である。

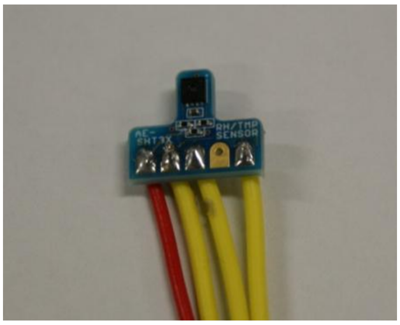


図１１　HDC1000

演算部には Arduino nano を使用した。センサから I^2C で取得した値を図６の判定基準において正常値か危険値か判別し、通信部へデジタル出力で判定結果を出力する。

通信部には BLEnano を使用した。図１２に BLEnano を示す。BLEnano は ARM Cortex-M0 と Bluetooth の一体型モジュール（Nordic nRF51822）を搭載しているためマイコンの機能と通信の機能の両機能を兼ね備えている。そのため小型化が実現した。図１０のように判定した結果から危険信号と正常信号を BLE（Bluetooth Low Energy）規格によって送信する。

図１３、図１４に送信機とケースを示す。図１３の基板を図１４のケースに格納した。

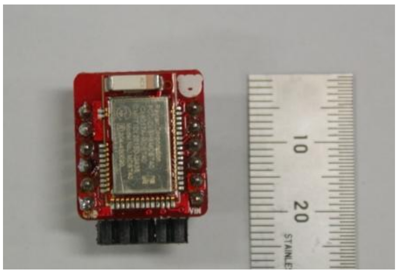


図１２　BLEnano

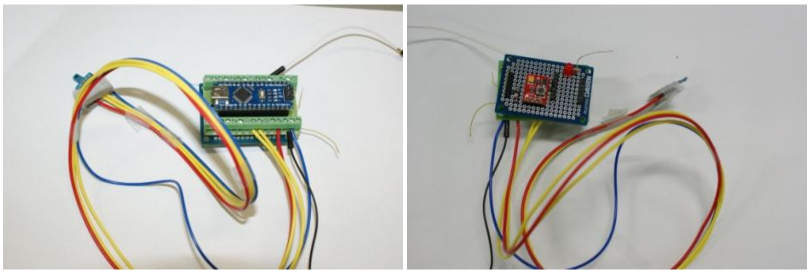


図１３　送信機基板

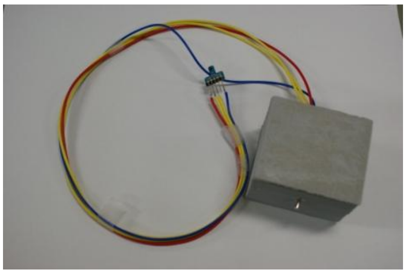


図１４　送信機のケース

３－２　受信機の構成

受信機の状態遷移図を図１５、具体化したフローチャートを図１６に示す。受信機の通信部、演算部には BLEnano を使用した。通信部では BLE で判定結果を受信する。演算部では結果から図１５、図１６の様にデジタル出力で正常値なら LED を緑点灯、危険値なら赤点滅させる。

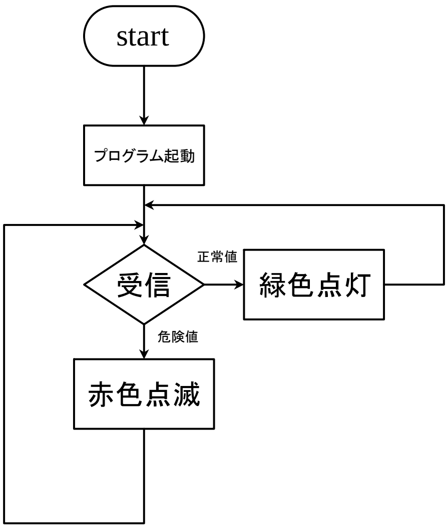


図１５　受信機状態遷移

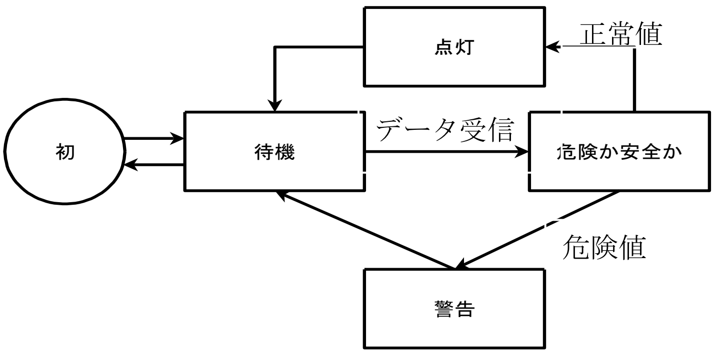


図１６　受信機アルゴリズム

図１７、図１８に受信機とケースを示す。図１７の基板を図１８のケースに格納した。

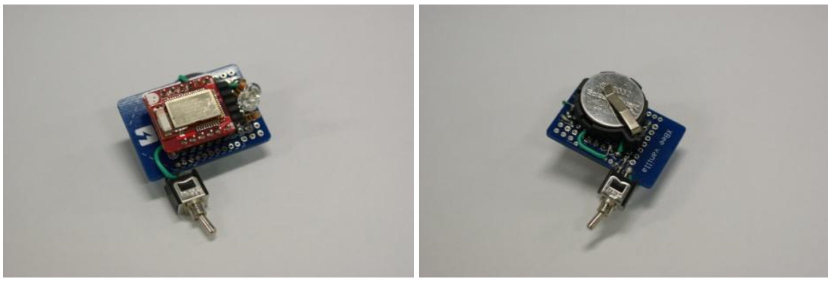


図１７　受信機基板

図１８　受信機のケース

４．実験

４－１　動作チェック

まず、製作した機器の動作確認を実施した。今回の判定基準で危険値となる状態のときに受信機の LED が緑から赤とならなければならない。送信機の温湿度センサに息を吹きかけることで温湿度が上昇し、危険値に達する。図１９は試験結果の写真を示す。試験した結果、息を吹きかけない状態で受信機の LED が緑点灯、息を吹きかけた状態で赤点滅することを確認した。次に判定精度を確認するため息を吹きかけ 1 秒間隔で温湿度及び危険度判定結果を記録した。図２１はその結果を示す。危険度判定精度は 100[%]であった。

図１９　受信機の表示

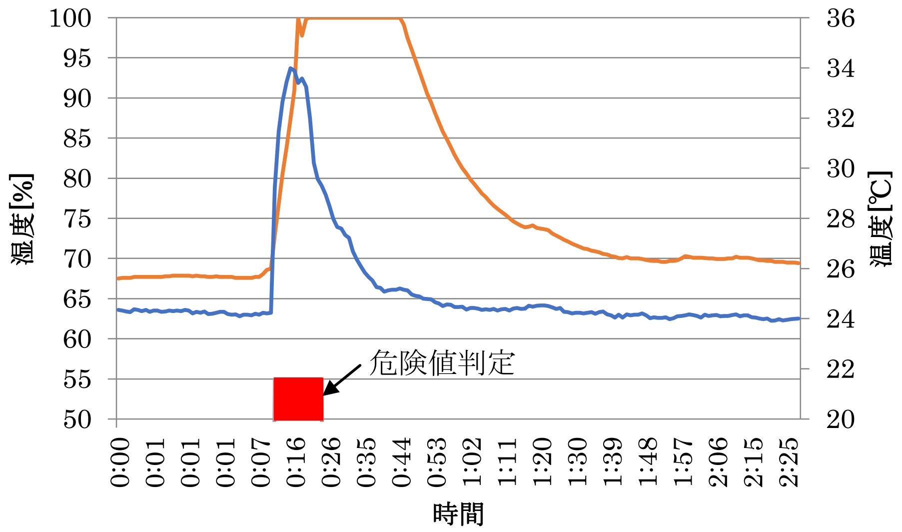


図２０　判定試験

４－２　センサの測定精度の検証

温湿度センサ HDC1000 と熱中症暑さ指数計 SK-150GT の比較を行った。実験中の写真を図２１に示す。図２１の様にセンサの高さを揃えて屋内で 5 時間測定した。

測定結果を図２２に示す。衣服内(HDC1000)のサンプリング数が 17749 点、外気(SK-150GT)のサンプリング数が 18002 点であった。総サンプリング平均は衣服内気温が 23.27[℃]、外気温が 23.27[℃]、衣服内湿度が 55.13[%]、外気湿度が 45.32[%]となった。この総サンプリング平均から求めた HDC1000 の誤差率は気温が 0.008603 [%]、湿度は 21.65[%]であった。

温度は誤差率が 1[%]未満であり誤差はほとんど無いが、湿度は 21.65[%]ある。しかし、標準偏差は衣服内が 0.4834[%]、外気が 0.4837[%]であり、小数点以下 3 桁まで偏差が等しいことから絶対誤差の 9.811[%]はバイアス成分であるといえる。よって衣服内温湿度の測定に用いるセンサの HDC1000 はバイアスを考慮することで使用可能となる。

図２１　センサ精度の確認

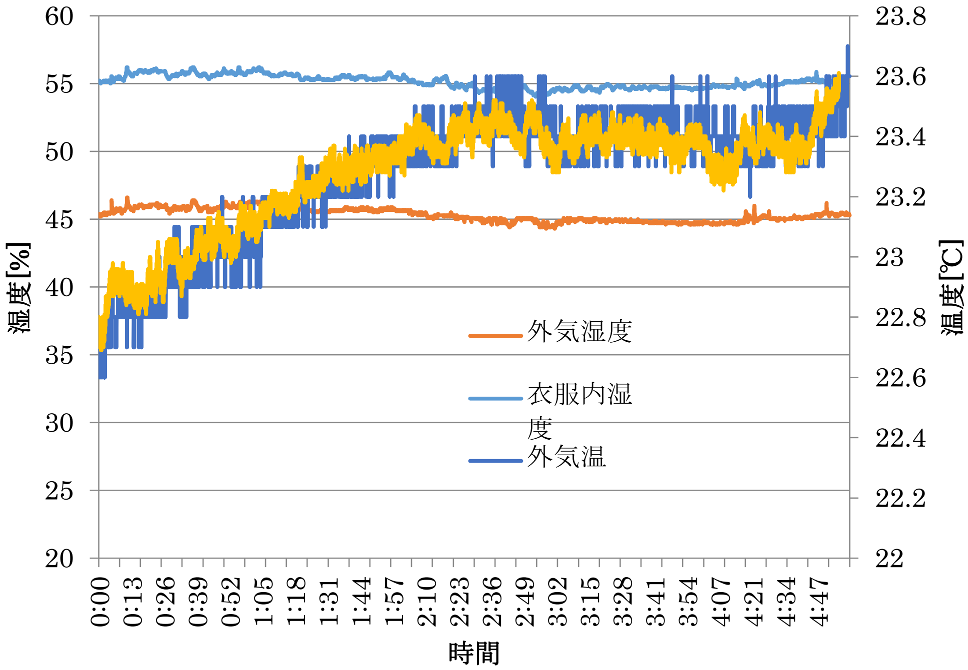


図２２　HDC1000と熱中症暑さ指数計の比較

４－３　運動時の衣服内気候

図２３は運動の様子を示す。運動ではベルトによって後輪と負荷のモータを接続した自転車を漕いで行った。図２４は実験時の服装を示す。a)が薄着の場合で、ｂ）が厚着の場合である。気温の低い屋内でも運動を行う事で熱中症になる可能性がある。そこで今回は比較的外気温湿度の変化が少ない屋内で実験を行った。図２４に示した服装の 2 条件で行った。

図２５に薄着での実験結果を示す。薄着の実験は最初の 5 分間休憩した後に 15 分間運動し、更に 1 時間休憩した。

図２５が示す様に最初の休憩時は外気から衣服内にセンサを入れた事により衣服内の温度は上昇し、湿度が下降した。最大値と最小値の差は温度 5.69[℃]、湿度 22.71[%]であった。運動開始直前は外気温度に対し衣服内温度が 7.38[℃]高く、外気湿度に対し衣服内湿度は 31.5[%]低い状態であった。

運動を開始すると衣服内温度は直前の 27.48[℃]から最大値 27.74[℃]となり衣服内温度の上昇は 0.26[℃]と微小であった。一方、衣服内湿度は直前の 51.6[%]から最大値 78.6[%]ととなり 27[%]上昇した。衣服内温度は 27[℃]から 28[℃]まで上昇した。

その後休憩をすると衣服内温度は 1 分後に最大値である 29.04[℃]まで上昇した、衣服内湿度は 2 分後に最大値である 91.53[%]まで上昇した。よって衣服内温湿度は運動後も上昇する事が分かった。これは運動の影響で発汗と皮膚温の上昇が起きた事によるものであると考えられる。次に衣服内気温は 2 分後に運動開始直前の 27.48[℃]まで回復したが衣服内湿度は運動開始直前の 51.6[%]まで回復するのに 50 分を要した。湿度の回復に時間がかかった要因に衣服の乾燥時間も考えられる。この時外気の平均気温 19.71[℃]、平均湿度 65.27[%]であった。標準偏差は気温 0.21[℃]、湿度 0.86[%]であった。また、衣服内の標準偏差は気温 0.81[℃]、湿度 10.71[%]であった。記録を厚着と同条件の休憩 5 分運動 13 分とすると衣服内の標準偏差は気温 1.06[℃]、湿度 8.46[%]であった。

図２６に厚着での実験結果を示す。図２６が示すように厚着をすると運動開始前の段階で衣服内気温は外気温と比べ 12.91[℃]高かった。休憩時間の平均衣服内温度が 30.03[℃]、平均衣服湿度が 49.61[%]であった。最小値は 27.71[℃]、43.46[%]であった。13 分の運動で最大値が 34.36[℃]、97.34[%]となり、最大 6.65[℃]、53.88[%]上昇した。この時外気の平均気温 18.83[℃]、平均湿度 66.14[%]であった。標準偏差は気温 0.057 [℃]、湿度 0.25[%]であった。また、衣服内の標準偏差は気温 1.73[℃]、湿度 19.59[%]であった。

衣服内温湿度の上昇量が高くなり、80[%]に達するまで薄着で 8 分、厚着で 9 分かかった。90[%]に達するまでに薄着の場合 19 分、厚着の場合 11 分かかった。

よって厚着は薄着に比べ 90[%]になるまでに要する時間が短く、標準偏差も薄着に比べ厚着は大きくなる事が分かった。以上の結果から衣服内温湿度は外気温湿度の変移とは異なる事が明らかとなった。

図２３　運動実験

a)薄着　　　　　　　　　　　　b)厚着

図２４　服装

図２５　薄着の場合

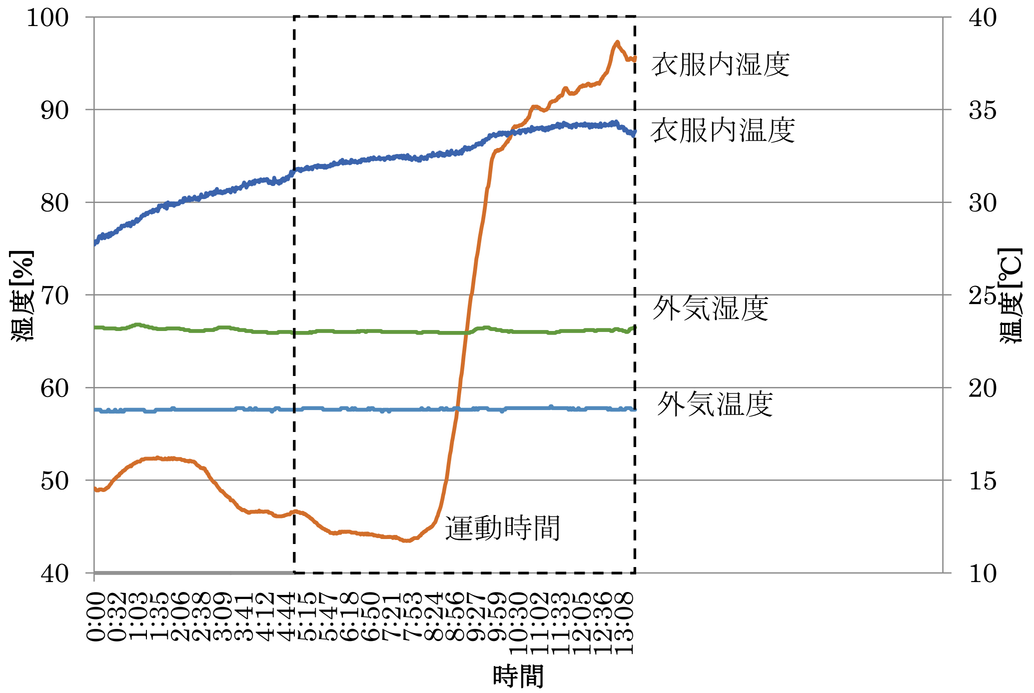


図２６　厚着の場合

５．結論

本研究は、熱中症の防止をするため衣服内温湿度を測定し、測定値からの危険度判定によって被験者に警告を出す機器の製作を行った。まず受信機を製作し警告が出ることを確認した。次に機器を用いて判定値の検討を行った。

その結果、衣服内温湿度は外気温湿度の変移とは異なる事が明らかとなった。よって、外気から熱中症を予測するよりも衣服内温湿度から予測する方が適しているといえる。また、服装によってベースとなる運動をしていない状態での衣服内温度が大きく変わる事が分かった。この結果をもとに実験を更に行い、危険度判定の精度を向上させるシステムを完成させたい。

６．今後の課題

スマートフォンアプリを開発し、衣服内温湿度及び危険度判定結果を BLE からスマートフォン経由でクラウドにデータを出力するシステムを構築したい。

７．参考文献

[1] 多井吉之助、田多井恭子：“最新被服衛生学”光生館　pp12 1960

[2] 原田隆司、土田和義衣服の快適性と衣服内気候-衣服内の水分/熱移動現象のとらえ方-”SEN-I GAKKAISHI(繊 維 と 工 業)　繊維学会誌 Vol.46, No.3　pp98 1990

優秀賞　賞金 10 万円

2017 Examine Project

大気汚染物質観測衛星「PENTA」

香川高等専門学校高松キャンパス

平成 29 年 10 月 21 日

第 6 回気象文化大賞「高校・高専『気象観測機器コンテスト』」

コンテスト 2 次審査　実証実験報告書

2017 Examine Project
大気汚染物質観測衛星「PENTA」

香川高等専門学校　宇宙開発研究部
気象観測機器コンテスト　A 班

参加学生
電気情報工学科 4 年　大西　哲
電気情報工学科 4 年　新谷　悠真
機械電子工学科 4 年　前川　雄壱

指導教員
電気情報工学科　山本　雅史

目次

要旨 1

1 導入 1

2 プロジェクトの概要 2

2.1 成層圏気球について 2

2.2 過去の高高度気象観測プロジェクト 2

① 第 2 回気象観測機器コンテスト「高層気象観測バルーン」 3

② 第 3 回気象観測機器コンテスト「Cloud Examine」 3

③ 第 4 回気象観測機器コンテスト「ガッスン from Examine Project」 4

④ 第 5 回気象観測機器コンテスト

「Stratosphere Examine Satellite(成層圏観測衛星)『SES』」 4

2.3 今回の高高度気象観測プロジェクト 5

3 ミッション 5

① 上空の気象の測定・考察 5

② 大気汚染物質の測定 5

③ 機体の状態をリアルタイムで観測 5

④ 安定したデータの取得 6

⑤ 衛星の低コスト化 6

⑥ 上空から地上の撮影 6

4 使用機器 7

① Arduino Uno Rev3 7

② mbed LPC1768 7

③ Raspberry Pi 2 Model B+ 8

④ XBee PRO ZB S2B モジュール 8

⑤ Arduino 用ワイヤレスシールド 9

⑥ GPS・ジャイロ・加速度センサ 9

⑦ Grove BME280 9

⑧ Grove Dust sensor 9

5 機器の概要 (ワイヤレスセンサネットワーク) 10

5.1 センサノード (Arduino) 11

5.2 センサノード (mbed) 13

5.3 ゲートウェイ 15

5.4 XBee による通信 17

5.5 Wvdial について......18

6. 機器の概要 (衛星外装)......19

6.1 キャリア......19

6.2 パラシュート......19

6.3 バルーン......20

7. 放球の概要......21

7.1 CUSF Landing Predictort について......21

7.2 放球場所の選定......22

7.3 落下予想地点......22

7.4 係留実験......23

7.4.1 係留実験の概要......23

7.4.2 当日の実証実験......23

8. 実験結果......24

9. 考察......27

9.1 気温と湿度の関係......27

9.2.1 気温と高度の関係......28

9.2.2 逆転層について......29

9.3 PM2.5 濃度･高度の関係......30

9.4 加速度･高度の関係......30

9.5 カメラ......31

10 評価......32

10.1 ミッション別の評価......32

① 「上空の気象の測定・考察」について......32

②「大気汚染物質の測定」について......32

③「気体の状態をリアルタイムで測定」について......32

④「安定したデータの取得」について......32

⑤「衛星の低コスト化」について......33

⑥「上空から地上の撮影」について......33

10.2 前回との比較......33

10.2.1 mbed の使用......33

10.2.2 空撮機構......33

10.2.3 データ送信......33

10.3 総合的な評価......34

10.4 次回へ向けて......34

11. 感想 **35**

12. 付録 **37**

12.1 機器の動作フローチャート 37

12.2 Wvdial の設定方法 40

13. 参考資料 **43**

要旨

大気汚染物質観測衛星「PENTA」は，Examine Project と題して気象観測機器コンテストで出展してきた成層圏気球の 5 作品目である．今回は，大気汚染物質 (PM2.5) の測定をメインとした 6 つのミッションを設定し，その達成を目的に製作した．機器は主に，2 種のセンサノードとゲートウェイからなり，機器の特性を生かすために，それぞれに別のマイコンを用いて制御している．係留実験では，上空の正確な気象データをとることができ，興味深い考察もできた．機器完成時期と昨年までの経験から，放球実験こそ断念したが，実用性は確かめることができた．また，昨年までの機器・システムと比較して，安定性・利便性・汎用性も格段に上がり有用なものを製作することができた．

1 導入

私たちが生活している地球は，地上からの高度によって気象状況が大きく変化するが，高高度 (上空 3 万メートル程度) における気象状況は, 一般にあまり知られていない. しかし, 世界中の気象に関わる企業・団体が多くの気象観測気球を打ち上げていることから分かるように，高高度の気象状況は私たちの生活圏の気象状況を予想することに大いに貢献している．また，オゾン層破壊や温室効果ガスによる地球温暖化現象などに代表される環境問題の実情を正しく知るためにも，高高度の気象状況を測定することは極めて重要である．

そこで私たちは，「バルーンを用いて高高度の気象を測定し，地上での気象との相違点，類似点を見つけ考察すること」，「地上と上空での大気汚染物質 (PM2.5) の分布濃度の違いを測定し，現在の地球環境を知ることで，環境問題の正しい知識を持つこと」を目標として大気汚染物質観測衛星「PENTA」を製作した．

2 プロジェクトの概要

2.1 成層圏気球について

私たちは，気圧・気温・湿度等の気象要素を測定するセンサを搭載し，測定した情報を送信するための無線送信機を備えた気象観測器 (ラジオゾンデ) を製作してきた．またその機器をバルーンによって打ち上げ，上空でバルーンが割れると，パラシュートによる減速効果を受けながら落下する，成層圏気球の打ち上げ実験をおこなってきた．

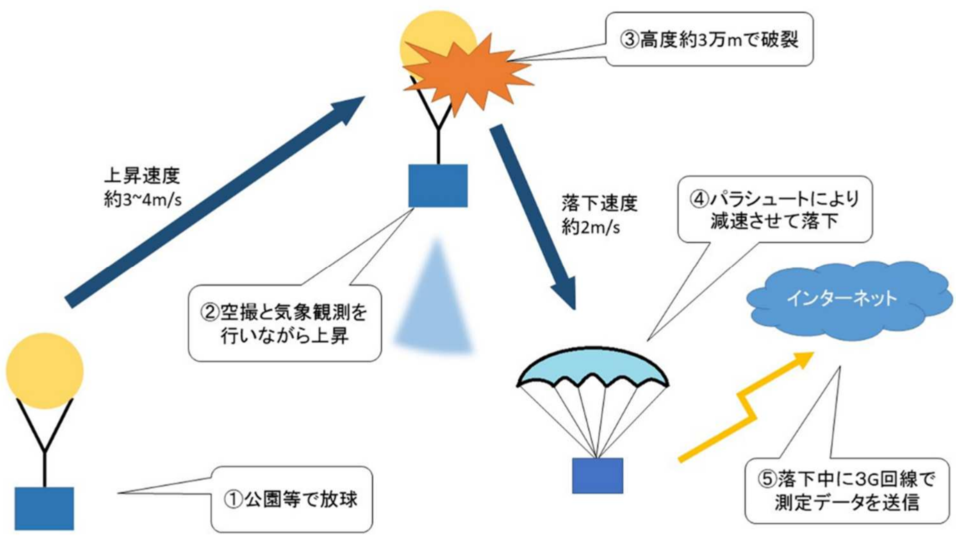


図 2-1 成層圏気球の概要図

2.2 過去の高高度気象観測プロジェクト

これまで私たちは本コンテストにおいて，過去 4 度にわたり成層圏気球を「高高度気象観測プロジェクト (Examine Project)」と名付けて出展した．過去 4 回の出場を通して，技術を進歩させ，またその年の環境問題・気象状況に合った機器を製作してきた．

後に示すのは，過去の Examine Project の概要である．

① 第 2 回気象観測機器コンテスト「高層気象観測バルーン」

第 2 回のコンテストには上級生が初めて出場し，高高度の気象観測の知識も少ない中，バルーンを 3 つ用いて校舎の屋上から釣竿で衛星を係留し大気安定度を測定した.

だが，低高度の測定だったため十分な高度のデータが得られず実用性には欠けた.

図 2-1 高層気象観測バルーンによる気象観測の様子

② 第 3 回気象観測機器コンテスト「Cloud Examine」

第 3 回のコンテストでは，3000 L のヘリウムガス用バルーンを用いて高度 3 万メートル程度の気象観測ができる装置を製作した. 測定した気象データは WEB サーバにアップするシステムを導入し，気圧・温度・湿度から成る 3 種類の基本的な気象データをインターネット上でリアルタイム閲覧することができた.

しかし，台風の影響で放球ができず係留実験のみとなってしまったため十分な高度のデータを取得することができなかった.

図 2-2 Cloud Examine による係留実験の様子

③ 第 4 回気象観測機器コンテスト「ガッスン from Examine Project」

第 4 回はセンサのプログラムをさらに安定的なものに変更，新たなセンサを追加し，第 3 回よりも詳しい気象データを得ることができた．また，第 3 回にはできなかった放球も成功した．

しかし，機器を回収することができず一部のデータを取得することができなかった．

図 2-3 ガッスン from Examine Project の放球準備

④ 第 5 回気象観測機器コンテスト「Stratosphere Examine Satellite(成層圏観測衛星)『SES』」

第 5 回ではすべてのセンサのデータを WEB 上にアップロードすることに成功した．さらに既製品をなるべく使わないようにし，低コスト化も実現した．スマートフォンを利用して写真を撮り，高高度の画像も取得することができた．

しかし，放球後に機器をロストしてしまい気象データの内容とその取得場所を結びつけることができなかった．また，ヘリウムガスが不足していたことと，当日の風が強かったことが原因で，気球は 6000 m 付近までしか浮上せず，十分なデータを取得することができなかった．

図 2-4 Stratosphere Examine Satellite(成層圏観測衛星)「SES」による上空写真

2.3 今回の高高度気象観測プロジェクト

今年の「大気汚染物質観測衛星『PENTA』」は Examine Project の 5 作品目となる．過去の技術を活かしつつ，新しいことにも挑戦している機器を製作できた．

今回の Examine Project では，地上から上空にかけての気圧・気温・湿度・PM2.5 の観測と空撮を行う．

「PENTA」(ペンタ，ギリシャ語で「5」の意味) という名前には，「気温・気圧・湿度・PM2.5・空撮という 5 つの気象状況を測定する」という意味と，「Examine Project の 5 作品目である」という 2 つの意味が掛けられている．

3 ミッション

私たちは 1 章で示した，「バルーンを用いて高高度の気象を測定し，地上での気象との相違点，類似点を見つけ考察すること」，「地上と上空での大気汚染物質 (PM2.5) の分布濃度の違いを測定し，現在の地球環境を知ることで，環境問題の正しい知識を持つこと」を達成するために，以下の 6 項目を今回の具体的なミッションとして設定した．このミッションを設定するにあたっては，データ取得だけでなく，地球環境や安全面にも十分配慮した．

① 上空の気象の測定・考察

温湿度センサや気圧センサを用いて，基本的な気象情報である気温・気圧・湿度が高度によってどのように変化するかを測定する．また，得られたデータから，地上と上空の気象の違いを検証するとともに，上空の気象状況から地上の気象状況を予想する方法について考察する．

② 大気汚染物質の測定

ダストセンサを用いて地上と上空の大気汚染物質の一種，微小粒子状物質 (PM2.5) の分布密度を測定する．測定したデータから，PM2.5 の鉛直濃度分布を算出し，大気汚染物質による環境問題の実情を知る．また，温湿度と PM2.5 の関連性を調べ，PM2.5 への対策方法について考察する．

③ 機体の状態をリアルタイムで観測

気象データを測定する機器系統とは分離させた機器を使用して，機体の状態 (機体の角度，加速度，位置情報，高度) を取得する．これによりリアルタイムで衛星の状態を観測し，気象データと位置情報・高度情報を結びつけるとともに，次回以降の気球の打ち上げ時の参考にする．

④ 安定したデータの取得

気象データを取得する機器，気体の状態を取得する機器と，それらのデータを一元的に管理・転送する機器 (ゲートウェイ) による WSN (Wireless Sensor Network：ワイヤレスセンサネットワーク) を構築する．

ゲートウェイに送られた各種データは，CSV ファイル等のデータ編集が容易なファイルに変換する．その後，SMTP (Simple Mail Transfer Protocol) により 3G 回線を用いて気球外部に送信する無線システムを構築する．

⑤ 衛星の低コスト化

昨年同様，既製品の測定器を使用せず，種々のセンサをマイコンで制御することにより，低コストかつ安定した測定を実現する．また，センサだけでなく WSN のゲートウェイもシングルボードコンピュータを用いて設計し，柔軟かつ安価なシステムを構築する．

⑥ 上空から地上の撮影

カメラのタイムラプス機能を用いて上空から地上を撮影する．カメラは WSN で使用したゲートウェイを利用し，写真データの送信には，気象データを SMTP により送信する通信回線と同じものを利用することでコストダウンを図る．

4 使用機器

以下に，今回使用したマイコン，周辺機器及びセンサの名称と写真，概要を示す．また，センサ類については仕様も併せて示す．

① Arduino Uno Rev3

Arduinoは，イタリアで開発された教育用マイコン機器である．この機器を用いて，気象データを取得するセンサ (温湿度センサ・気圧センサ・ダストセンサ) の制御を行なった．

Arduinoを採用した理由として，サンプルコードが豊富なため，開発が容易であること．ダストセンサのような利用者の少ないセンサについても，リファレンスやライブラリが豊富に用意されていること．mbedなどのマイコンに比べて，安価であることなどが挙げられる．

図 4-1 Arduinoの外観

② mbed LPC1768

mbedはARM社の開発したマイコン機器である．この機器を用いて，衛星の状態を取得するセンサ (GPS・加速度センサ・ジャイロセンサ) の制御を行った．

mbedを採用した理由として，マイナーなセンサのリファレンスこそ少ないものの，GPSやジャイロ・加速度センサといった基本的なセンサに関しては，Arduinoよりも安定して利用できること．香川高専では，UNISECの副理事長である日本大学の宮崎康之教授主催のHEPTAハンズオン講座により，mbedを用いた小型模擬衛星の開発を行っているので，この技術を応用して衛星バスに採用できることなどが挙げられる．

図 4-2 mbedの外観

③ Raspberry Pi 2 Model B+

Raspberry Pi はラズベリーパイ財団の開発したシングルボードコンピュータである．この機器を用いて，Arduino や mbed から送られたデータを CSV ファイルに編集し，SMTP によりメールで送信する，WSN のゲートウェイとして動作させる．

Raspberry Pi はシングルボードコンピュータであるため，基本的な GUI 環境が整備されている．また SMTP やシリアル通信関係のモジュールが豊富なプログラミング言語 Python の実行環境が揃っているため，WSN のゲートウェイとして採用した．

図 4-3 Raspberry Pi の外観

④ XBee PRO ZB S2B モジュール

下図の機器は，Raspberry Pi と Arduino，mbed 間の通信モジュールである．これを用いて Arduino，mbed から取得したデータを Raspberry Pi に送信する．

図 4-4 XBee の外観

⑤ Arduino 用ワイヤレスシールド

下図の機器は Arduino に搭載する XBee 用のシールドである．今回は Raspberry Pi と通信するために搭載した.

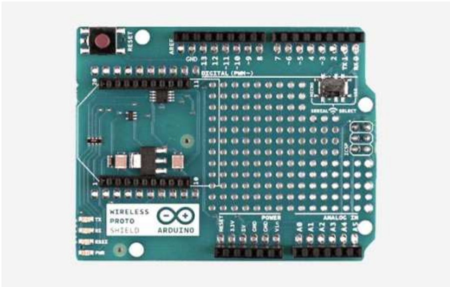


図 4-5 Arduino ワイヤレスシールドの外観

⑥ GPS・ジャイロ・加速度センサ

GPS・ジャイロ・加速度センサは，mbed で位置情報と機体の速度情報を測定するために利用した．GPS は高度 18,000m，速度 515m/秒まで測定することができる．位置精度は±5m である．ジャイロセンサは機体の角速度 (回転の速度) を±150deg/ s 〜±4800deg/s まで測定することができる. 加速度センサは 0.8〜1000Hz の範囲で機体の加速度を測定することができ，感度は 100±10%である.

⑦ Grove BME280

この機器は温度・湿度・気圧を測定することができるセンサである．サンプルコードに気圧から高度を算出するプログラムが掲載されていたので参考にした．通信形式は I^2C である．温度・湿度・気圧を同時に測定することができ，また精度が他のセンサに比べて良いので採用した.

表 4-1 Grove BME280 の測定レンジと誤差

	レンジ	誤差
温度	-40〜85℃	±1℃
湿度	―	±3%
気圧	300-1100hPa	±1.0hPa

⑧ Grove Dust sensor

このセンサは，ダストの濃度を測定することにより環境中の空気の質を表すセンサである．空気中の粒子状物質レベル (PM レベル) は，所与の時間単位での低パルス占有時間 (LPO 時間) をカウントすることによって 1 m^3 に何個の粒子が存在するかを測定することができる．この LPO 時間は PM 濃度に比例する.

5 機器の概要 (ワイヤレスセンサネットワーク)

本章では，私たちが製作した機器の，データを測定し，そのデータをメールで送信するワイヤレスセンサネットワーク (WSN) 部について解説する．

本機器のワイヤレスセンサネットワークは，気象データを測定する 2 台のセンサノードと，センサノードからのデータを受信しメールで送信する 1 台のゲートウェイから構成される． (図 5-1)

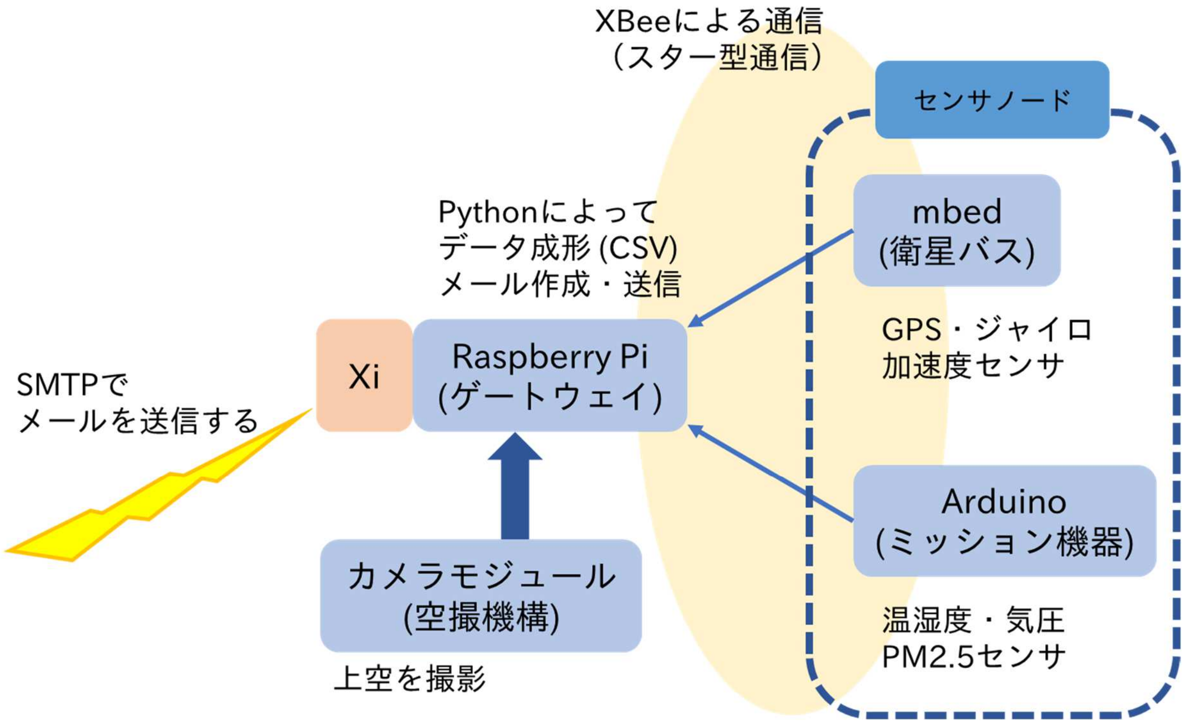


図 5-1 ワイヤレスセンサネットワークの接続図

2 台のセンサノードは，それぞれ Arduino と mbed により制御されており，ゲートウェイは Raspberry Pi により制御されている．センサノードとゲートウェイ間の通信には XBee を使用し，スター型通信と呼ばれる 1 対 n の通信技術を利用している．また空撮機構であるカメラモジュールは Raspberry Pi に USB 接続 されている．

センサノードから送られたデータは，Raspberry Pi で CSV ファイルに編集・保存されたのち，SMTP によりメールで送信される．メールの送信はインターネットを介して行われるため，インターネットの接続には NTT docomo の Xi を利用している．

5.1 センサノード (Arduino)

センサノード (Arduino) の役割は，「気温・気圧・湿度・PM2.5 の測定を行い，取得した気圧データから高度を算出する」,「取得・算出したデータを文字列に変換しゲートウェイに送信する」の 2 点である．Arduino の制御に関するフローチャートは 12 章に示す.

センサはベースシールドに接続，XBee は XBee シールドに接続し，それぞれのシールドと Arduino を接続することで構成される． (図 5-2)

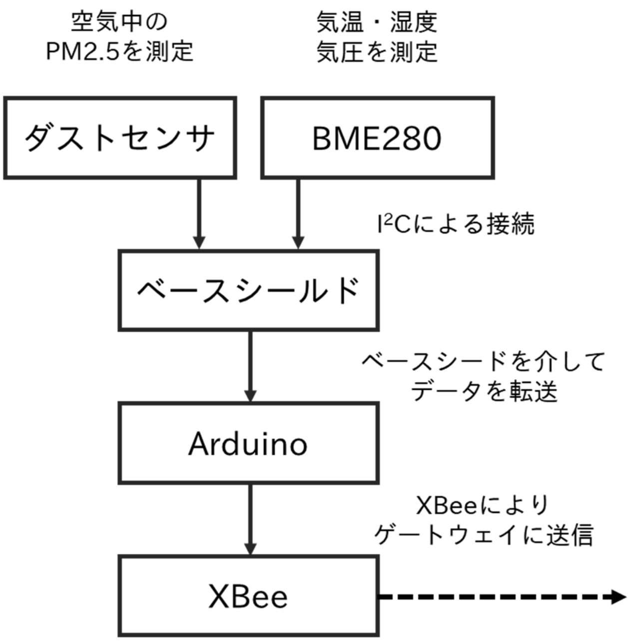


図 5-2 センサノード (Arduino) の概要

データの取得は，60 秒に 1 回行う．これは，Arduino やその他センサに負荷がかからず，またゲートウェイで処理落ちせずに処理できるように設定した時間である．

取得したデータは，気圧・気温・湿度・高度・大気汚染物質濃度の順番に図 5-3 のようなカンマ区切りの文字列に変換され，XBee によりゲートウェイに送信される．

図 5-3 センサノード (Arduino) のデータ送信フォーマット

センサで取得したデータは基本的に double 型 (実数型) や int 型 (整数型) などの数値型であるが，数値型データでは，XBee で数回に分けて送信する必要があり，また数値型データはゲートウェイで CSV ファイルを作成する際に負荷がかかる．そこで一度文字列型に変換してすべてのデータをカンマで区切り，上図のように 1 つの長い文字列型変数に格納した後，XBee でゲートウェイに送信する方式をとっている．データの先頭はヘッダ固有文字列ではじまり，これにより受信側のゲートウェイはデータが送られてきたことを検知する．センサノード (Arduino) のヘッダ固有文字列は「aaaaa」である．データの末端は Line Feed (LF) とよばれる改行コードによって締められ，これにより受信側のゲートウェイはデータの送信が終了したことを検知する．

また，この文字列への変換の際に気圧から高度の算出も行っている．気圧を p をとする時の高度 h を求める式は，以下のように表せる．

$$h = \frac{1 - \left(\frac{p}{1.01325\times10^{5}}\right)^{\frac{1}{5.25588}}}{2.25577\times10^{-5}}$$

例えば，ヘッダ固有文字列が「aaaaa」，気圧が 1000 hPa，気温が 25.0 ℃，湿度が 45.5 %，PM2.5 濃度が 20 μg/m^3 の時，センサノードからゲートウェイに送られるデータは，「aaaaa,1013.25,25.0,45.5,110.88,20.0 (LF)」となる．

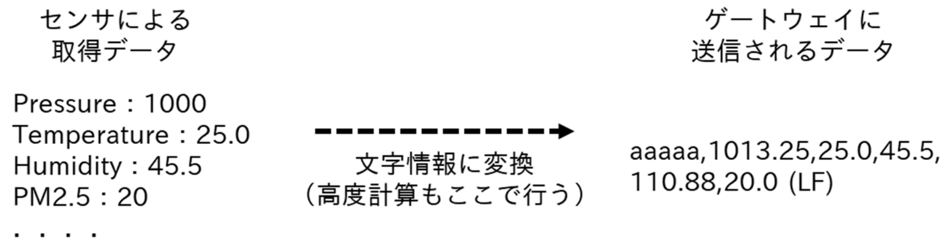


図 5-4 Arduino による文字列変換

昨年度は，これらのデータ列を Carriage Return (CR) によって区切って送信していたが，今年度のゲートウェイは Linux に大きく依存していることから，CR によるデータ分割は不適切であると判断し，カンマと LF によってデータ分割を行った．

5.2 センサノード (mbed)

センサノード (mbed) の役割は,「機体の状態 (角度・加速度・位置情報・高度) を観測する」,「取得したデータを文字列に変換しゲートウェイに送信する」の 2 点である．mbed の制御に関するフローチャートは，12 章に示してある．

センサノード (mbed) には,「HEPTA ハンズオン・トレーニング」にて，日本大学の宮崎教授より譲り受けた，HEPTA-Sat を使用している．HEPTA-Sat は，Sensor Board，C&DH (Command & Data handling) Board，EPS (Electrical Power Supply) & Comm. Board の 3 つの Board から構成されており，センサ系，通信系の各コンポーネントが各インターフェイス仕様に基づき，C&DH 系であるオンボードコンピュータ (mbed) に接続されている．今回は，センサ系より，3 軸ジャイロ，3 軸加速度，GPS を，通信系より，Xbee を使用する．

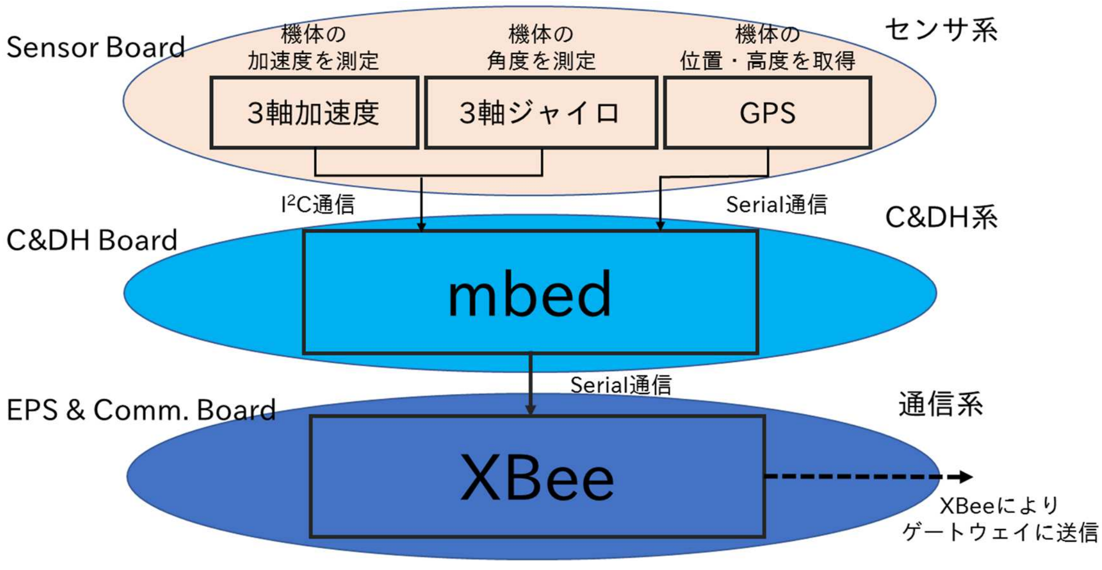


図 5-5 センサノード (mbed) の概要

データの取得は，Arduino と同じく 60 秒に 1 回行う．

取得したデータは，緯度・経度・高度・加速度 (x・y・z 軸) ・ジャイロ (x・y・z 軸) の順番で図 5-6 のような Arduino と同様のフォーマットに変換され，XBee によりゲートウェイに送信される．このとき，データの送信のタイミングが Arduino と重ならないようにデータの取得開始を 30 秒遅延させる．

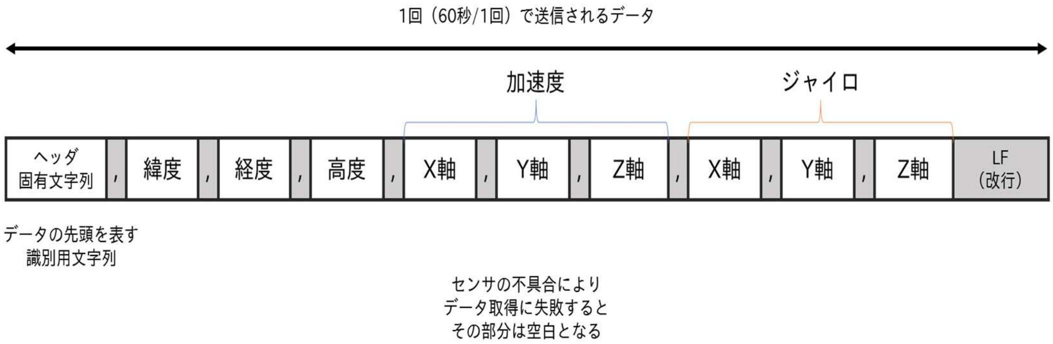


図 5-6 センサノード (mbed) のデータ送信フォーマット

センサノード (mbed) のヘッダ固有文字列は「aaaaa」であり，Arduino と共通である．

ヘッダ固有文字列が「aaaaa」，緯度 35° ・経度 135° ・高度 150m・加速度 (x・y・z 軸) が $20\mathrm{m/s^2}$，$20\mathrm{m/s^2}$，$9.8\ \mathrm{m/s^2}$・ジャイロ (x・y・z 軸) が 45° ，74° ，85° のとき，センサノード (mbed) からゲートウェイに送られるデータは,
「aaaaa,35,135,150,20.0,20.0,9.8,45,74,85 (LF)」となる．

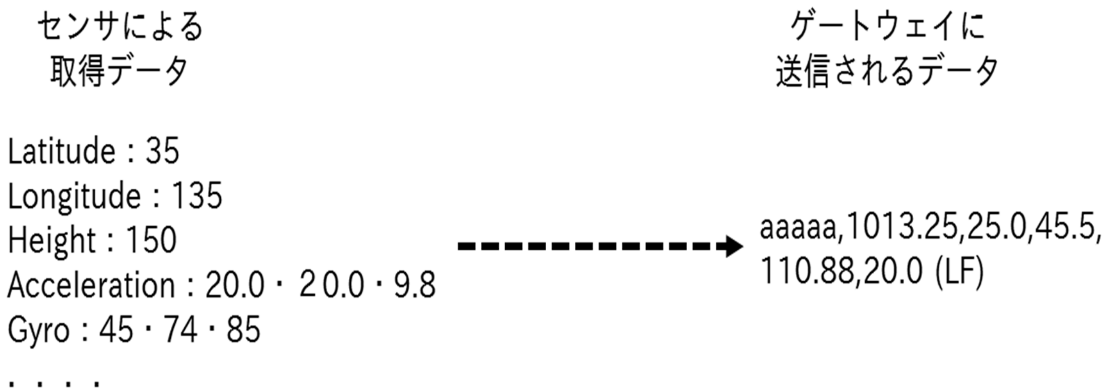


図 5-7 mbed による文字列変換

5.3 ゲートウェイ

ゲートウェイの役割は「センサノードから送られてきたデータを，CSV 形式に編集する」，「編集した CSV ファイルを，SMTP によりメールで気球外部に送信する」，「カメラモジュールを制御し，空撮を行い，画像ファイルを SMTP によりメールで気球外部に送信する」の 3 点である．これらの制御は Raspberry Pi と Python プログラムにより行っている．Python プログラムのフローチャートは，12 章に示してある．

Raspberry Pi は，シングルボードコンピュータの一種であるため，USB 通信や Linux をベースとした OS などに対応しており，非常に高度な制御が可能である．今回は，Raspbian と呼ばれる OS を用いて，Raspberry Pi およびその接続機器の制御を行った．

センサノードから送信されたデータを受信する XBee は，USB 経由で接続した．XBee からデータを受信しファイルに保存する Python プログラムを常駐させ，XBee にデータが送られると，そのデータのヘッダ固有文字列を参照し，データが破損していないかを判断する．その後， データ文字列に対してタイムスタンプ (日時と日付の情報，図 5-8 において赤色で示した部分) を付加し，ファイルに追加保存する．

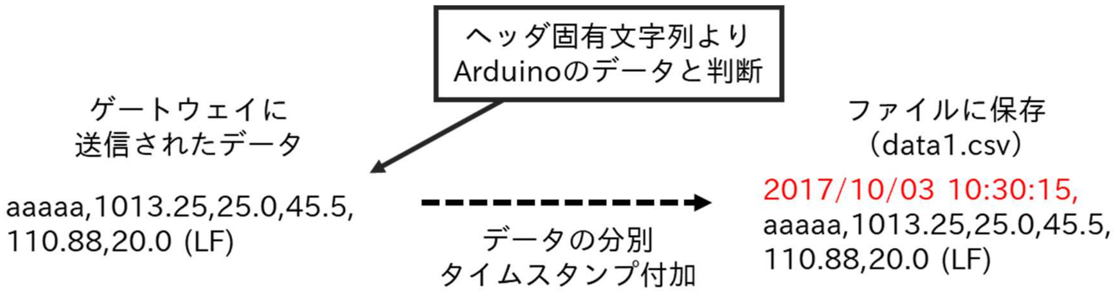


図 5-8 データの分別と保存

ゲートウェイから送られるデータが CSV に近いフォーマットであるため，Raspberry Pi で行う処理は，ヘッダ固有文字列によるデータの分別とタイムスタンプの付加のみである．

また，Arduino と mbed から同時にデータが送られてデータが破損した場合や，通信エラーによりデータが消失した場合等を判断して，データの取得を打ち切る設定になってある．

カメラモジュールについては Raspberry Pi に USB カメラを接続することで実現した．カメラの制御は motion と呼ばれる Raspberry Pi のアプリケーションを用いた．撮影は 1 分間隔で行われ，画像は/home/Pi/motion ディレクトリに jpeg 形式で保存される．新しい画像が撮影されると古い画像が削除されるようになっている．撮影した画像は，後述の SMTP によりメールで送信される．

SMTP によるメール送信は，python プログラムにより実現される．メールには，センサノードで取得したデータを CSV ファイルに編集したものと，カメラにより撮影された画像を添付して送信する．プログラム中では MIME (Multipurpose Internet Mail Extension) を利用してメールを作成している． (図 5-9)

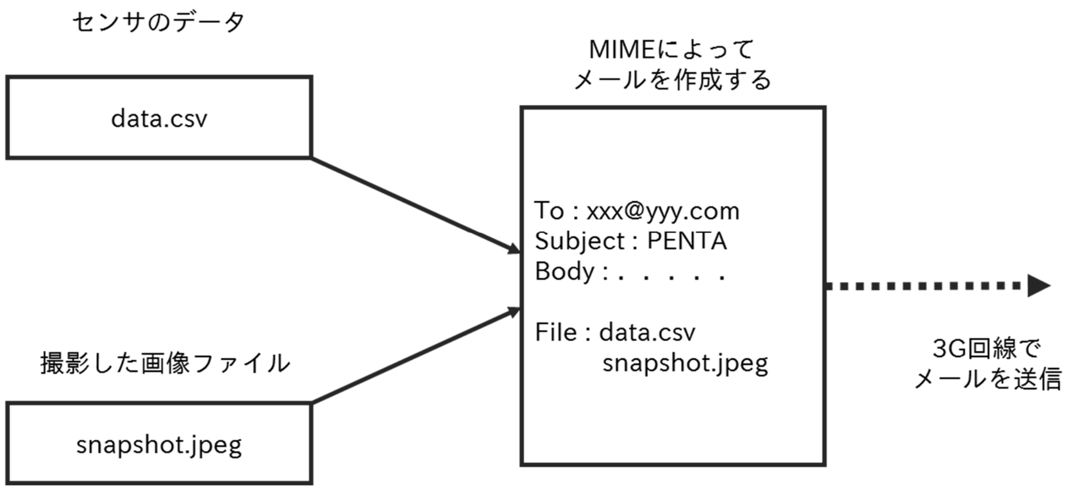


図 5-9 メール作成の原理

メールの送信間隔はセンサノードからのデータ受信回数が 20 回になった時に設定している．去年の気象観測機器コンテストでは，データの受信回数にかかわらず，一定時間間隔でデータを送信していたが，この方法ではデータを受信していない場合でも送信を実行してしまい，大きな負荷が Raspberry Pi にかかってしまうため，変更した．

5.4 XBee による通信

XBee による通信は, センサノードとゲートウェイを接続するために行う. この通信には, ゲートウェイを Coordinator とする，スター型通信を用いた.

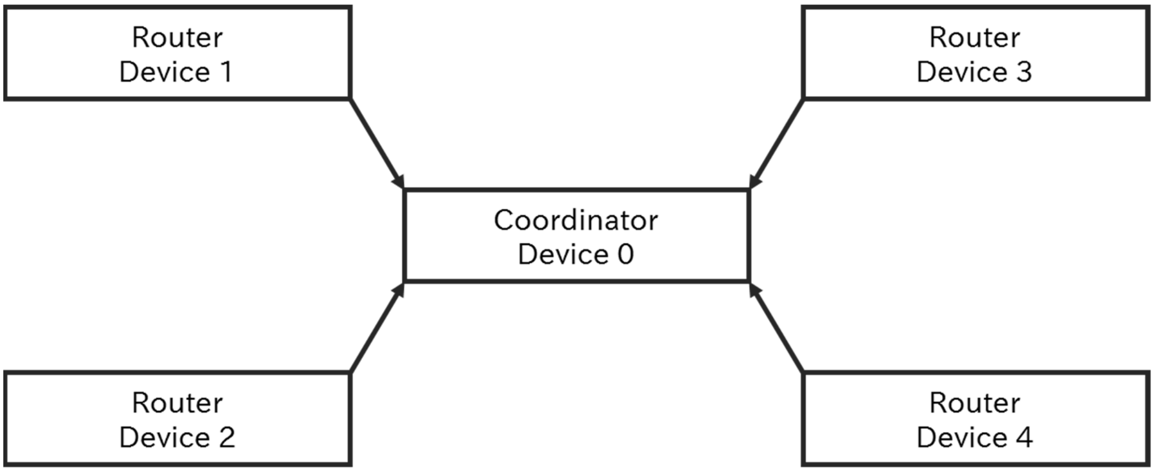


図 5-10 スター型通信のイメージ

スター型通信では，1 台の Coordinator となるデバイスと，複数の Router となるデバイスが通信をする．今回は，センサノードが Router デバイス，ゲートウェイが Coordinator デバイスである．

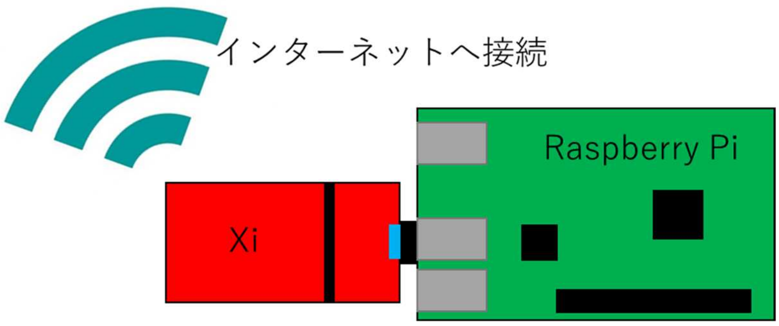
XBee は，送信先デバイスの指定を Mac アドレスによって行う．また，それぞれの XBee は PAN ID と呼ばれる固有の ID を設定することができ，この ID が同じネットワーク内でのみ通信を行う.

その他，XBee には AT モードと API モードと呼ばれる 2 つの通信モードが存在する，今回の機器には，AT モードを利用している．AT モードでは，2 つの XBee 間で，無線化されたシリアル通信を行うことが出来る．XBee には 20 本のピンが存在するが，そのうちの電源ピン，GND ピン，DOUT ピン (UART 入力) ，DIN ピン (UART 出力) のみを用いてシリアル通信を実現する. 他の通信機器と違い, 通信用のデータフォーマットやハンドシェイクを考慮する必要がなく, 有線のシリアル通信がそのまま無線化するだけなので, センサノードのプログラムを変更する必要がないなどの利点が存在する.

5.5 Wvdial について

今回 Raspberry Pi の 3G 通信機材として docomo Xi (L-02C)を用いた．Xi を利用するために Wvdial というプログラムを導入した．以下に詳細を記す．

Wvdial はインターネットに接続するためにモデムに発信して pppd(Point-to-Point Protocol dial)を起動するプログラムである．私たちのように無線でインターネットに接続するために使用する(図 5-11)ほか，Raspberry Pi 自体をアクセスポイントにし，他のデバイスをインターネットに接続するために使用する場合もある．Wvdial の設定方法は，12 章に示す．



5-11　接続図

6. 機器の概要 (衛星外装)

この章では，今回使用したバルーン，パラシュートとキャリアの詳細を記す．

6.1 キャリア

今回はキャリアに発泡スチロールの箱を使用した．発泡スチロールは非常に軽く，適度な強度があるため衛星のキャリアに最適であると判断した．

次にキャリア内部を図 6-1 に示す．HEPTA-Sat，Arduino，Raspberry Pi を重さが偏らないよう均等に配置した．センサ部は正確に外部の気象を測定するためにキャリア外部に出し，カメラは景色等を綺麗に撮影するために側面に固定した．また，水が入らないようグルーガンとテープで防水加工も施した．(図 6-2)

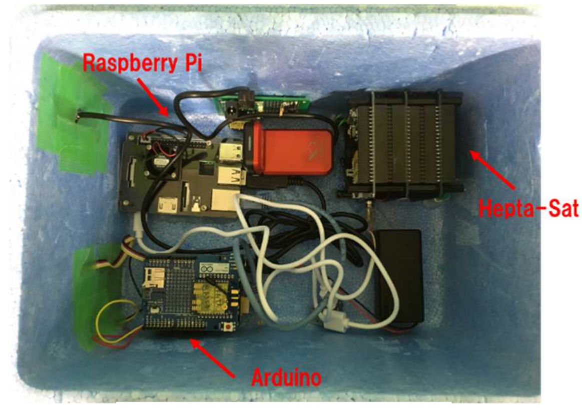


図 6-1　キャリア内部

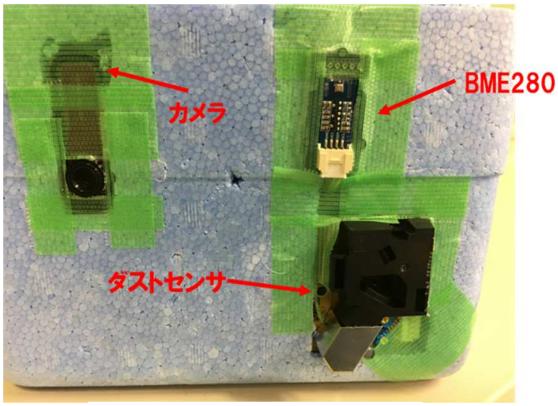


図 6-2　キャリア外部

6.2 パラシュート

今回のパラシュートに使用したのは去年同様，ヨットの帆を再利用した．実際に風を受けるものなので，パラシュートとして十分に使用できると判断した．このパラシュートは 8 本の糸でそれぞれ独立にキャリアと固定している．去年の結果より時速約 2.5 m まで落下速度を抑えることができる．パラシュートは各対角線が 1 m の正八角形 (図 6-3) で，中心に直径 10 cm の穴を空けている．この穴を空けることで落下速度を調整することができる．また，バルーンとキャリアを結ぶロープが通る場所も兼ねている．以下に実際使用したパラシュートの画像を示す (図 6-4) ．

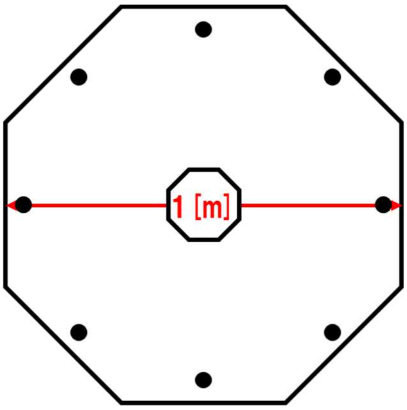


図 6-3　パラシュートの寸法

図 6-4　実際のパラシュート

6.3 バルーン

今回はラジオゾンデ専用の 3000L バルーン (株式会社気球製作所社製) を使用した．重さは 1 kg で直径は約 1.5 m である (図 6-5) ．参考として，ヘリウムガスは 1000 L で 1 kg のペイロードを浮かせることができる．今回はバルーン 1 kg + キャリア 1.52 kg で計 2.52 kg となった．始めは余裕をもって 3000 L 注入して係留実験を行っていたが，係留ロープの重さもあったため十分な浮力が得られなかった．したがって，4000 L 注入して係留実験を行った．

図 6-5　バルーン

7. 放球の概要

本章では CUSF Landing Predictort と昨年度の打ち上げ記録を用いて打ち上げ場所および落下予測地点の選定を行った.

7.1 CUSF Landing Predictort について

このソフトは，気象観測バルーン用に作られた飛行シミュレータである．地図には Google map が採用されており，任意の地点にカーソルを合わせることでサイト右上に緯度と経度が表示される．(図 7-1) 以下の表と図より具体的な使用方法を説明する．

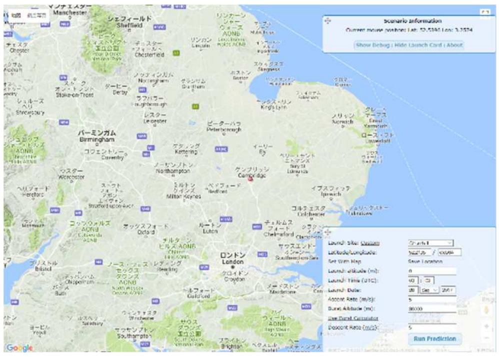


図 7-1　CUSF Landing Predictort

表 7-1 操作パネル

Launch Site	放球地点
Latitude/Longitude	緯度/経度
Launch altitude	放球地点の高度
Launch Time	放球時刻
Launch Date	放球日
Ascent Rate	上昇速度
Burst Altitude	破裂高度
Descent Rate	下降速度

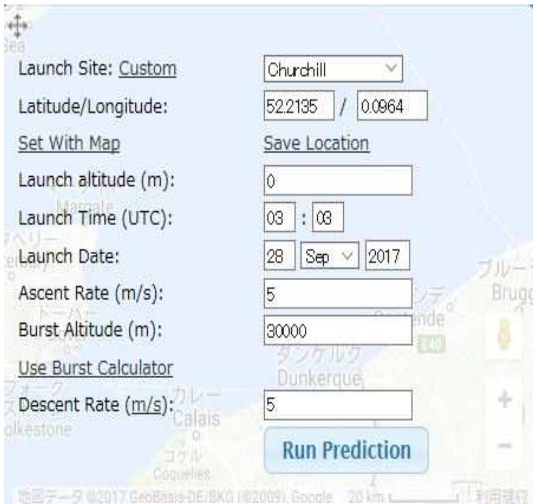


図 7-2　操作パネル

表 7-1 より，放球地点は初めにサイトに登録されている地点から打ち上げる場合は Launch Site のリストから地点を選択すると緯度/経度が自動的に入力される．その他の地点の場合は Launch Site を other に選択し，緯度/経度を入力する．次に放球地点の高度が海抜何 m の地点なのかを入力し，放球日時を入力する．このサイトは 3 週間後までの飛行予測を知ることができる．上昇速度，破裂高度，下降速度を入力することで正確な落下予測地点を算出することができる．

図 7-3　緯度/経度表示パネル

7.2 放球場所の選定

上記のサイトより以下の放球場所を選定した.

・放球場所：土佐清水総合公園 (図 7-4)
(32.7879, 132.9578)
・放球日時：9 月 23 日 (土) 10:30
・現地天気予報：曇り

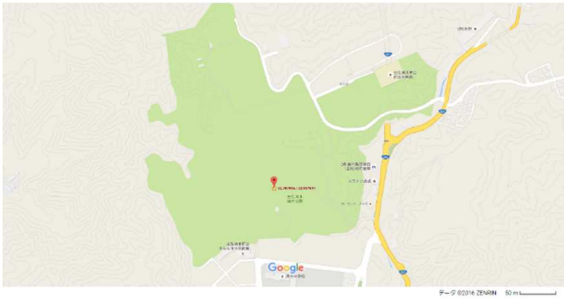


図 7-4　放球場所

7.3 落下予想地点

上記サイトより以下の落下地点を算出した.

・落下予想地点 (図 7-5)
(33.1468, 133.498)
・落下予想地点北 10km
(33.2369, 133.498)
・落下予想地点南 10km
(33.2369, 133.498)
・落下予想地点東 10km
(33.1468, 133.607)
・落下予想地点西 10km
(33.1468, 133.388)

図 7-5　落下予想地点

7.4 係留実験

7.4.1 係留実験の概要

係留実験は，本番用の機器を用いて行う．地上約 150m の気象状況を知ると同時に機器が正常に動作するか (熱暴走やバグ等がないか) を検証する．(図 7-6)

得られたデータをもとに更に高高度での気象情報を予測・考察する．

実際に放球する前に係留実験を行い，万が一に備えバックアップとして測定データを保管する．

また，係留実験はロープが切れるのを防ぐため 2 本取り付けた．

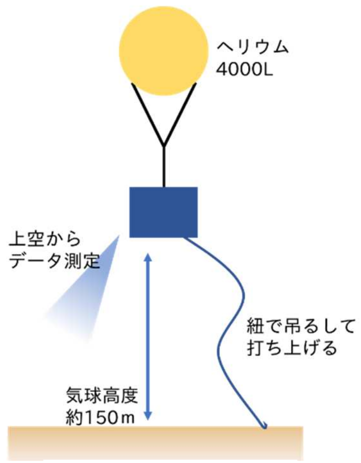


図 7-6　係留実験

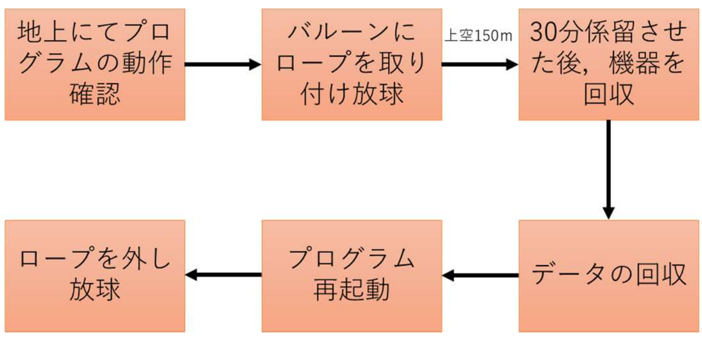


図 7-7　放球までの流れ

7.4.2 当日の実証実験

当初予定の打ち上げ時期 (9 月下旬) は季節風が強く吹いている．その為打ち上げを行うのは危険であると判断し，放球を断念した．よって，以下の日時および場所にて係留実験のみを行った．

・放球地点 : 香川高等専門学校高松キャンパス
　　　　グラウンド (図 7-8)
　　　　(34.3107，134.0103)
・放球日時 : 9 月 25 日 (月)　18:00～19:10
・現地天気 : 晴れ
・最大高度 : 132.7m

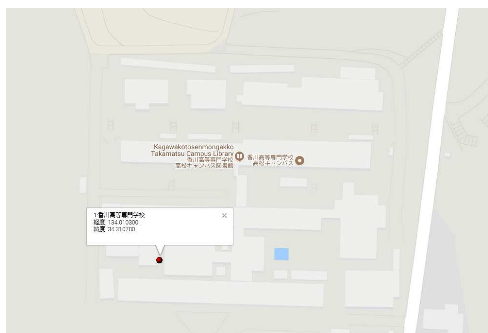


図 7-8　係留実験場所

8. 実験結果

7 章で示した通り，放球は不可能であったため，係留実験のみを行った．以下に係留実験の結果を示す.

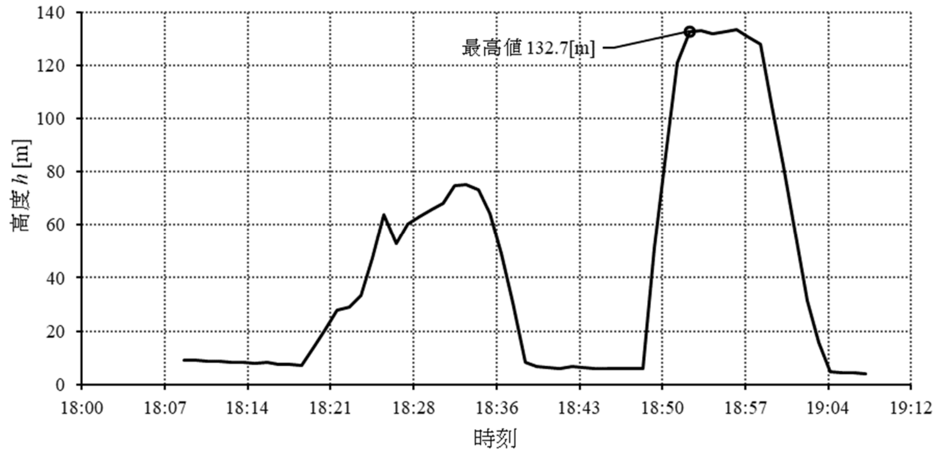


図 8-1 時刻と高度の関係

図 8-1 に示したのは，時刻を横軸，高度を縦軸にプロットしたグラフである．高度は Arduino のセンサノードに取り付けた BME280 の気圧センサによる気圧から，算出された高度である. 係留実験は 18:00～19:10 までに行い, 2 回上昇させている. 図より, 1 回目は 18:30 ごろ，2 度目は 18:53 ごろに最高到達点に達していることが分かる.

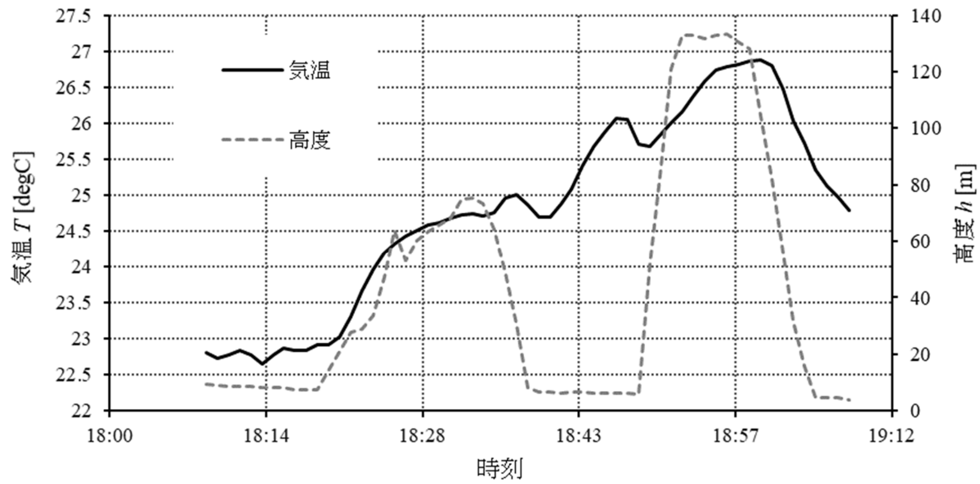


図 8-2 時刻と気温・高度の関係

図 8-2 に示したのは，時刻を横軸，気温を第一縦軸，高度を第二縦軸にプロットしたグラフである．この図より，気温は高度が高くなると上昇していることが分かる．加えて，時間の経過に従って気温が上昇していることが分かる．

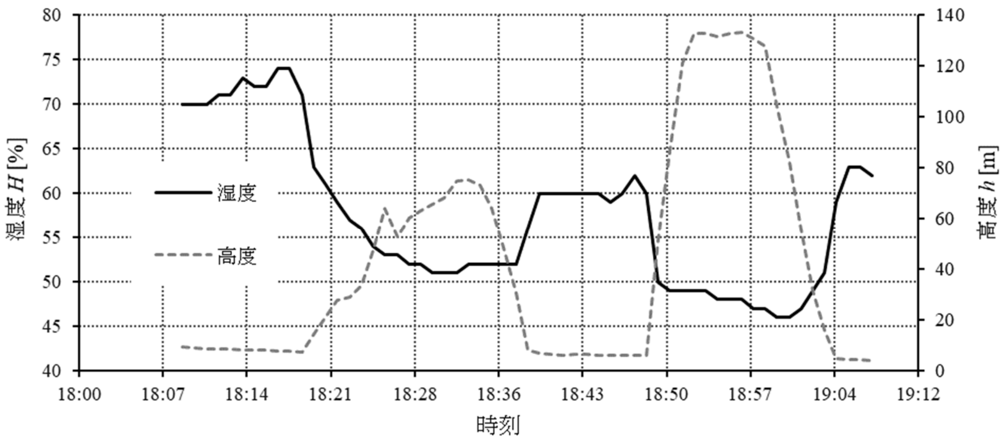


図 8-3 時刻と湿度・高度の関係

図 8-3 に示したのは，時刻を横軸，湿度を第一縦軸，高度を第二縦軸にプロットしたグラフである．この図より，湿度は高度が高くなると上昇し，ほとんど時刻によらないことが分かる．

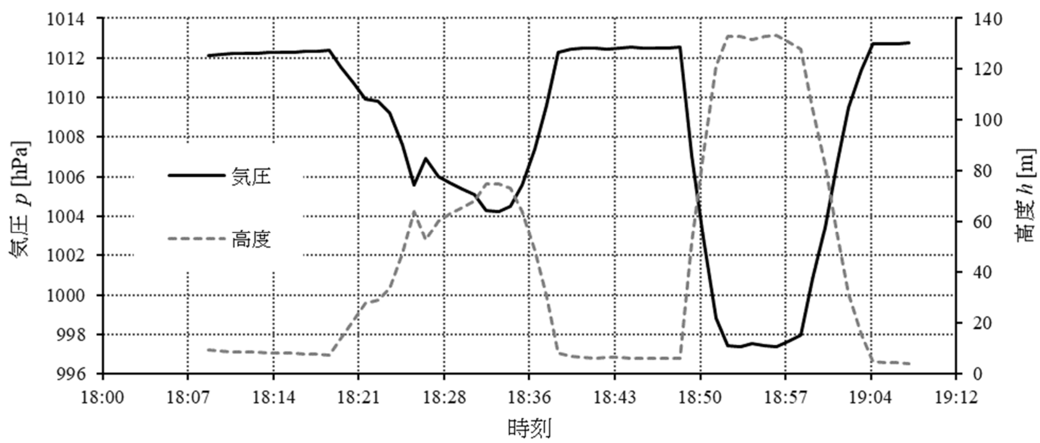


図 8-4 時刻と気圧・高度の関係

図 8-4 に示したのは，時刻を横軸，気圧を第一縦軸，高度を第二縦軸にプロットしたグラフである．図より，高度が高いほど気圧が低くなっていることが分かる．気圧と高度は 5.1 節の計算式に基づく関係性がある．その為，気圧と高度は対象の関係になっている．

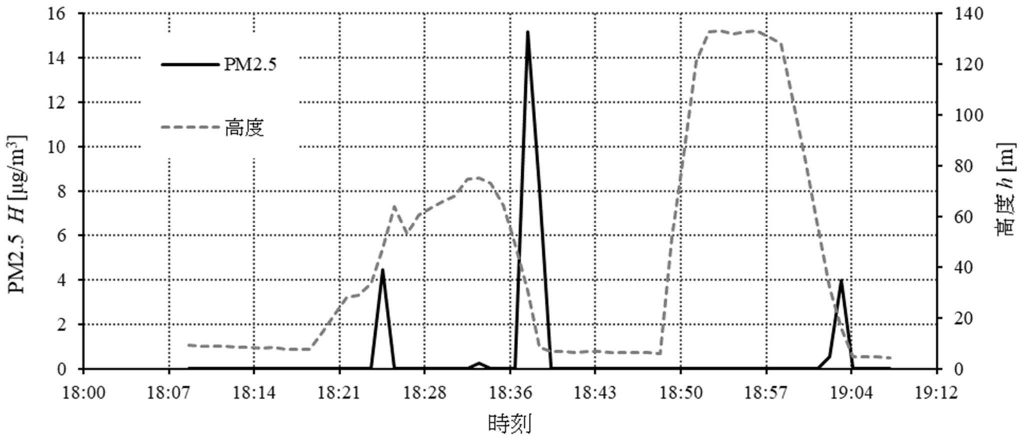


図 8-5 時刻と PM2.5 濃度・高度の関係

図 8-5 に示したのは，時刻を横軸，PM2.5 濃度を第一縦軸，高度を第二縦軸にプロットしたグラフである．図より，PM2.5 は低高度で特定の時刻に高く検出されるが，それ以外の点ではほとんど検出されないことが分かる．

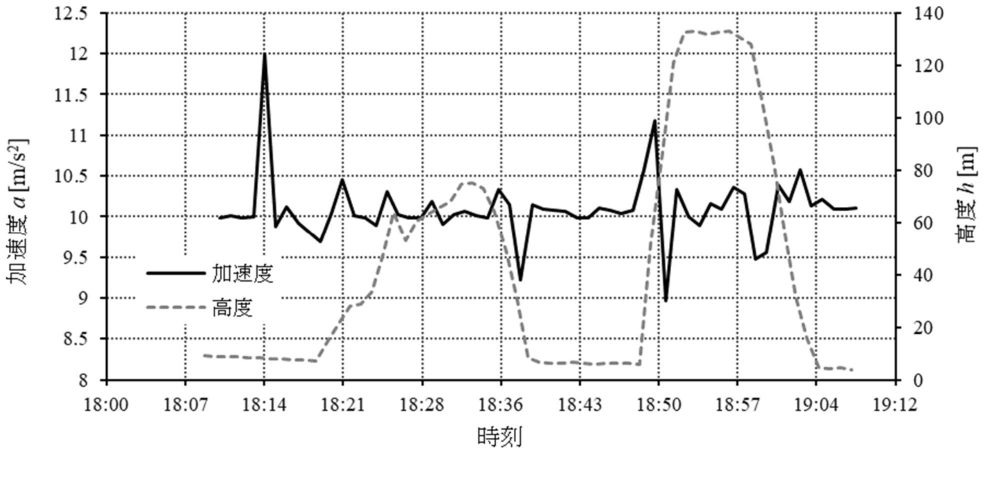


図 8-6 時刻と加速度・高度の関係

図 8-6 に示したのは，時刻を横軸，加速度を第一縦軸，高度を第二縦軸にプロットしたグラフである．加速度は，各方向の成分でなく大きさで計算している．

9. 考察

この章では 8 章の結果を元に地上の気象や比較しながら考察していく．また，考察をよりわかりやすくするため 8 章の実験結果のグラフを用いる．

9.1 気温と湿度の関係

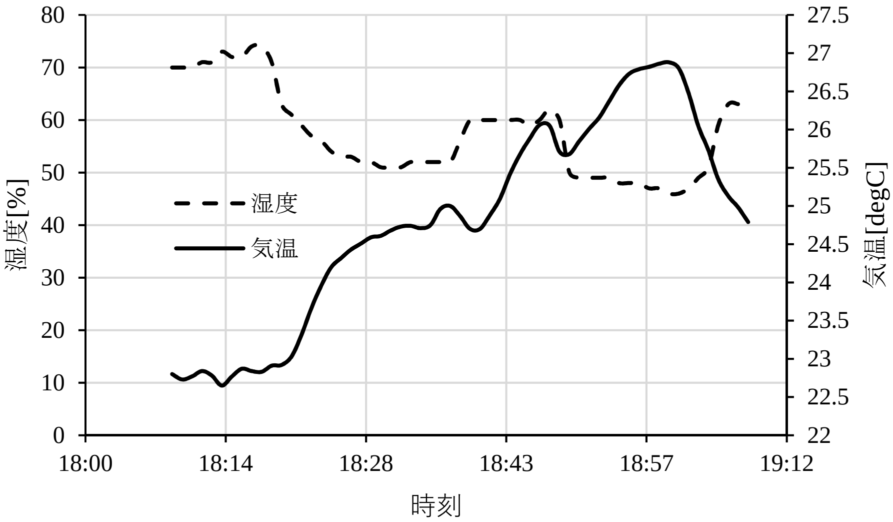


図 9-1　気温・湿度のグラフ

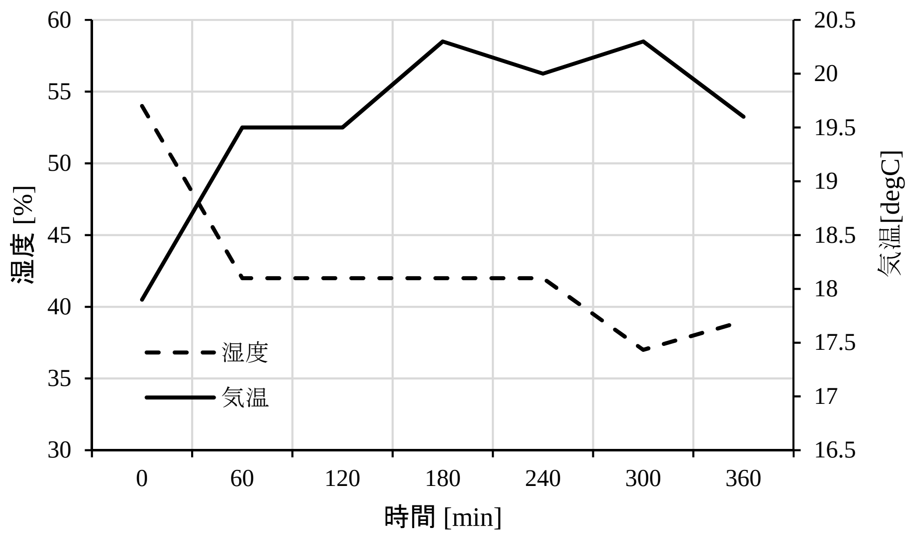


図 9-2　2015 年度地上での気温・湿度のグラフ

図 9-1 より，気温と湿度の関係は概ね対称の関係であることが分かる．このような現象が起こるのは空気中にある水蒸気が関係していると考えられる．空気中にはほぼ均等に水蒸気が存在し，温度によって水蒸気量が変化する．1m^3 の空間に存在できる水蒸気の質量を g で表したものを飽和水蒸気量と呼ぶ．この飽和水蒸気量は気温が高くなると増加し，相対的に湿度が下がる．また，気温が低くなると相対的に湿度が高くなる．そのため，気温と湿度の関係は対称の関係になっていると考えられる．

図 9-2 は 2015 年度の気象観測機器コンテストに出展した「ヘイズカスーミー ～視界距離自動判定装置～」が測定した気温・湿度のグラフである．このグラフも同様に，気温と湿度の関係は対称の関係になっていることが分かる．この 2 つのグラフより，飽和水蒸気量の関係は地上でも上空でも起こり，全体的な水蒸気量が異なっても同様の推移をすることが分かる．

9.2.1　気温と高度の関係

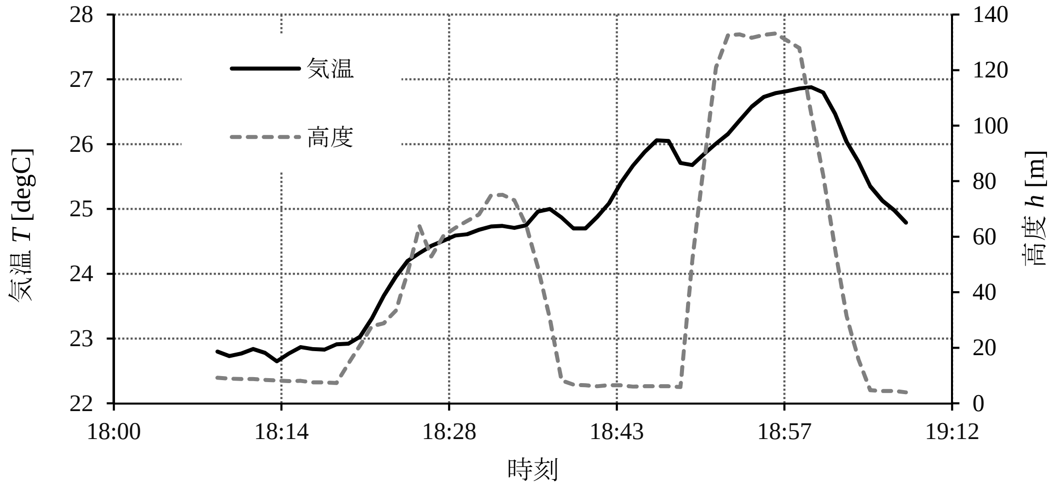
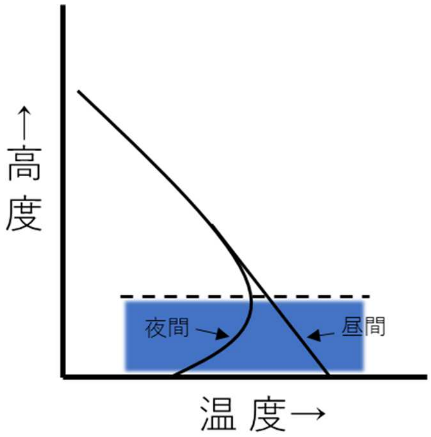


図 9-3 (図 8-2)　気温と高度の関係

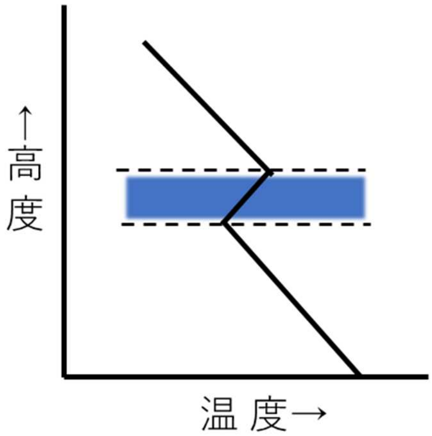
図 9.3 より，時間が経つにつれて気温がどんどん上昇し最後は急激に低下していることが分かる．通常は高度が高くなると気温が下がるはずだが，このグラフを見ると高度が大きく変化しても，気温は大きく変化することなく上がり続けている．これは係留地点に逆転層というものが存在していたからだと考える．逆転層は高度が上昇すると気温もそれに伴い上昇する層のことを言う．詳しくは次の 9.2.2 にて述べる．

9.2.2　逆転層について

一般的に対流圏内では，高度が高くなると気温は低くなる．しかし，ときには逆に高度が高くなると気温も高くなる層がある．この層を逆転層という．型や成因はさまざまなものがあるが，代表的な 2 つの逆転層について図 9-4 にまとめる．



(a)　接地逆転層



(b)　沈降逆転層

図 9-4　代表的な逆転層

この 2 種類の中でも一般的なものは図 9-4 (a) である．夜間，放射冷却で地面が冷え，それに接した空気が冷やされてできる．逆転層内では大気の成層は安定しており，大気の乱れは弱くなる．地面付近で発生した大気汚染物質は上方に拡散されず，霧も発生しやすい．これを放射霧という．夜が明けて太陽が昇ると大気は次第に地面に暖められるので，逆転層も放射霧も消える．

同じ型の逆転層は，暖かい空気が海面上を流れてもできる．この時も海面上に霧が発生しやすい．これを海霧という．

さらに，この型の逆転層は冷たく重い空気層の上にほかの地域から暖かくて軽い空気が水平に流れ込んできてもできる．これを温暖前線という．

図 9-4 (b) の型は，暖かい海面上の高気圧域内で発生しやすい．大気の下層は下から暖められ，不規則な小さい対流が発生し，よくかき混ぜられるので，気温は乾燥断熱減率に近い割合で高度と共に低くなる．一方，高気圧域には下降気流がある．沈降してくる空気はまわりの気圧が高くなるので，圧縮されて温度があがる．この沈降してきた空気と，下降でかき混ぜられた空気との境が逆転層となる．貿易風帯ではいつもこの型の逆転層があり，冬季，寒波が吹きだす日本海上でも発生しやすい．

気象学では(a)の型を接地逆転層，(b)の型を沈降逆転層と呼ぶ．

今回，当日に霧が発生していたため係留実験時に影響を及ぼしたのは(a)の設置逆転層であったと考えられる．

9.3　PM2.5 濃度・高度の関係

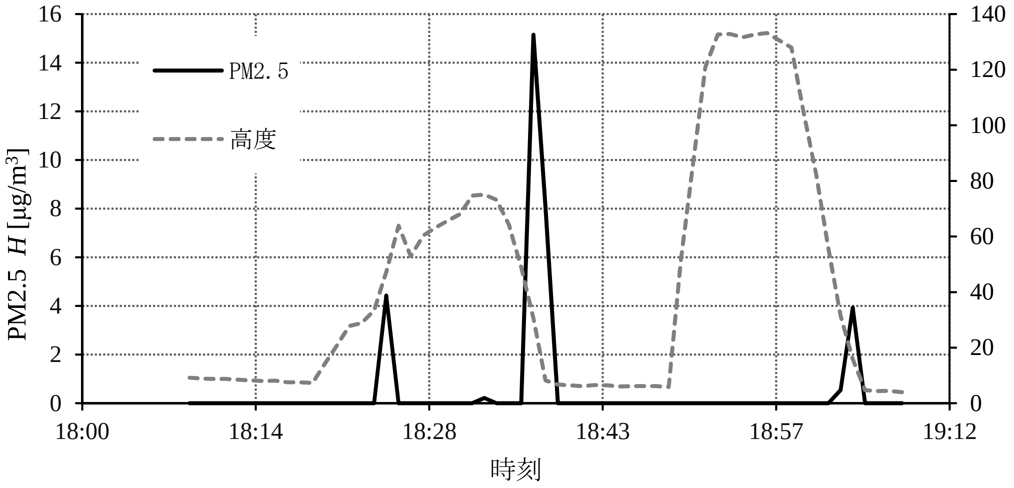


図 9-5 (図 8-5)　PM2.5 のグラフ

図 9-5 より，大きく 3 ヶ所の値が上昇していることが分かる．一番大きく上昇している部分は，接地逆転層によって発生した放射霧が関係しているのではないかと考える．他の 2 ヶ所に関しても同じく放射霧が関係していると思われるが，値が小さいため，センサのノイズが原因という可能性も考えられる．今回は高度による大気汚染物質の観測はできなかったが，接地逆転層による拡散されなかった大気汚染物質の観測はできた．

9.4　加速度・高度の関係

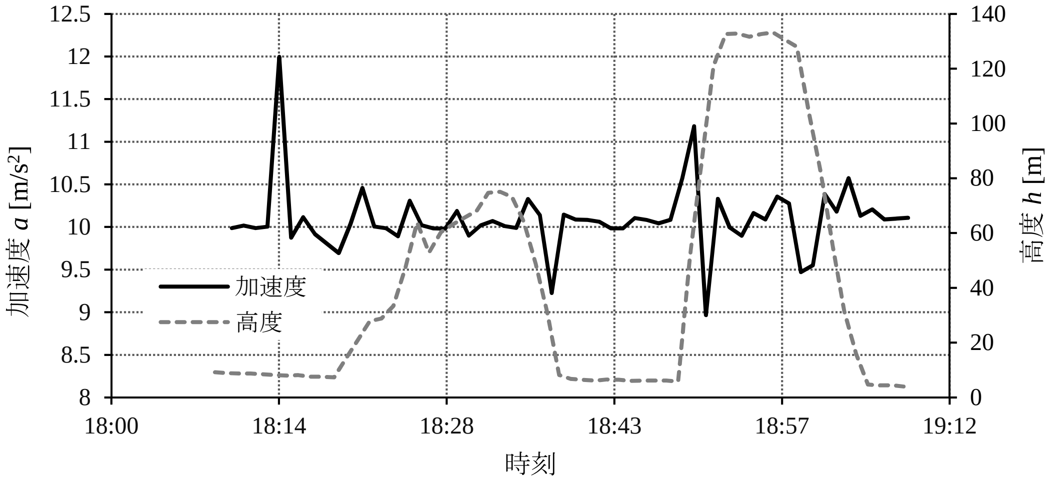


図 9-6 (図 8-6)　加速度・高度のグラフ

図 9-6 より，係留実験であったため，高度が上昇しても加速度が大きく変化することはなかった．参考として 2015 年度の気象観測機器コンテストに出展した「ガッスン From Examine project」のデータより，上空 8000m あたりでは季節風により時速 600km もの速度が出ていた．グラフより加速度が小刻みに変化しているのは係留ロープによる可能性もあるが，上空の風によって変化している可能性もあると考える．

9.5　カメラ

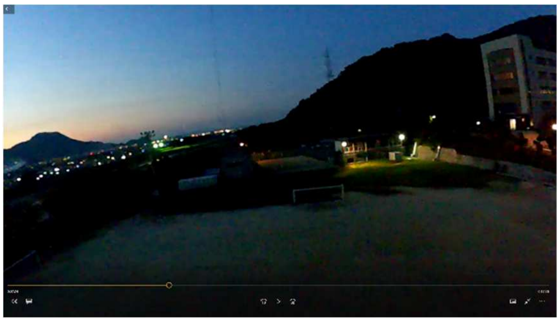


図 9-7　7 分後

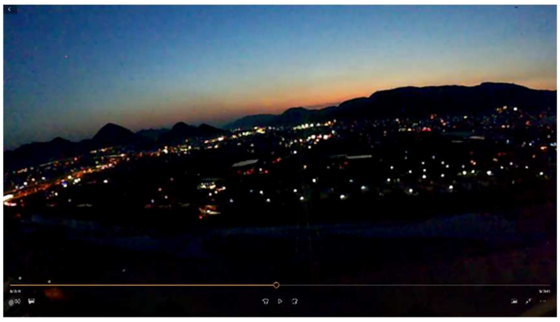


図 9-8　12 分後

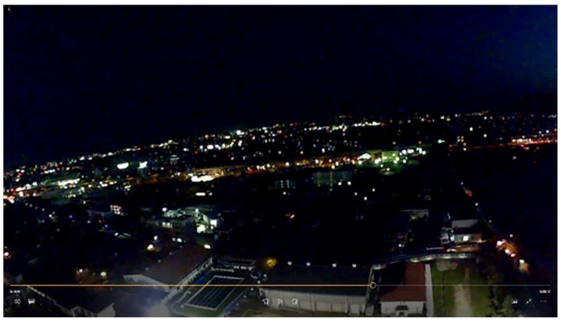


図 9-9　16 分後

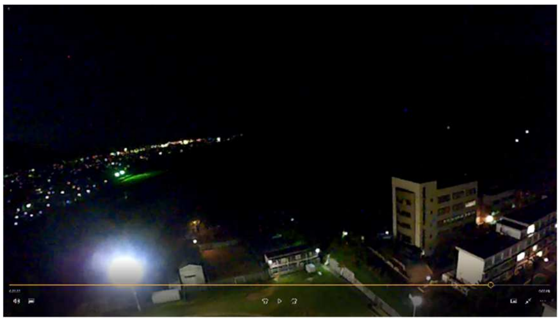


図 9-10　22 分後

上図は気球に搭載したカメラの画像である．上図を見ると，時間が経過していくにつれて，日が沈んで暗くなっていることが分かる．上図と国立天文台暦計算室の日の入り時刻(17:57)より，日の入り直前から日の入り直後までの上空の画像を撮影することができたと考えられる．

10 評価

この章では, 3 章で示した各ミッションの達成度を判断・評価するとともに機器の完成度・性能・有用性を評価する.

10.1 ミッション別の評価

① 「上空の気象の測定・考察」について

今回は，係留実験しか行っていないため，上空 3 万メートルの気象を測定するには至らなかった．しかし，11 章の考察でも述べているとおり，係留実験の結果から多くのデータを測定することができた．また，実際の放球ルートについても 7.1 節で示した CUSF Landing Predictort を用いることで昨年度よりも精密に予測が可能となった.

②「大気汚染物質の測定」について

8 章で示したようにデータの測定自体には成功した．しかし，測定場所 (香川高専グラウンド) において大気汚染物質濃度が低かったために, 有用な測定結果は得ることができなかった．例えば，工場地域の付近での大気汚染物質の鉛直濃度を測定する等，測定場所の場所の選定も必要である.

③「気体の状態をリアルタイムで測定」について

今回は，加速度センサ・ジャイロセンサ・GPS の 3 種類のセンサによって気体の状態を測定し，またそれらの測定データをメールで定期的に送ることに成功した．これによって得られたデータは，次回以降の放球において飛行経路を予想する重要なものとなる.

特に GPS は，昨年度まで既製品のものを使用していたため，データのフォーマットが他のセンサと一致せず，またコストも非常に高かったので，大きな進歩と言える.

④「安定したデータの取得」について

前年度と同様，センサノードに既製品を一切使わなかったため，プログラムを変更することでデータの取得間隔やデータのフォーマットを用意に変更できる．また，前年度は取得したデータをサーバにアップロードしていたが，この方法ではサーバの管理に多くの手間とコストが必要であった．今年はメールでデータを送信する方法に変更したため，サーバの管理を行う必要がなくなり，よりデータの取扱いが容易になった.

⑤「衛星の低コスト化」について

本章④でも述べたように，今年度もセンサノードに既製品を用いなかったため，制作費を大きく削減することができた．また，GPS についても今年度からは既製品を用いず，センサノードに組み込むことに成功した．また，カメラモジュールについても，昨年度はスマートフォンを用いて実現したが，今年度は USB カメラを用いたことで，コストを抑えることに成功した．

⑥「上空から地上の撮影」について

機器に搭載した USB カメラを用いて，定間隔で上空の様子を撮影することに成功した．これらの画像データは，気球の姿勢を知る上で重要なデータとなる．

10.2 前回との比較

前回の作品からの大きな変更点は以下の 3 つである．

10.2.1 mbed の使用

4 章②でも述べたように，mbed を用いることにより，基本的なセンサを安定して利用できる．この特性を活用し，取得するデータに機体の状態を追加できた．また，GPS は前回，既製品を用いたが，今回は mbed による取得をすることができた．そのため，GPS のデータもその他のデータと同様に確認できるようになった．

10.2.2 空撮機構

前回はスマートフォンを用いたが，今回は，Raspberry Pi に Web カメラを接続しての撮影としている．これにより，測定データとともに映像を送信することができる．また，今回用いた Web カメラはスマートフォンよりも低価格である．さらに，スマートフォンの場合はキャリアの外に取り付けると落下などの危険があるため，キャリアにのぞき窓を開けての撮影だったが，カメラモジュールではカメラ部をキャリアの外に出しての撮影が可能になった．その結果，鮮明な映像を取ることができる．

10.2.3 データ送信

前回は取得したデータを Web サーバにアップロードをしていたが，今回はメールにデータを添付して SMTP を用いて送信している．多数の人がみるには Web サーバは便利だが限られた人数ならばメールの方が閲覧しやすく，手続きの手間もかからない．

10.3 総合的な評価

今回の作品の総合的評価を行う．9 章より，気球を打ち上げではなく係留での実験のため高高度のデータとしては物足りないが，得られたデータは妥当なものであり，安定して取得できていることから機器の可用性は充分であるといえる．また，10.2.1 より，前回よりも測定項目を増やすことに成功している．さらに，10.2.3 より，データ送信システムを開発・導入し，データ確認の利便性が上がった．加えて今回は，すべてをマイコンで制御しているため，データを一括で制御でき，メールの添付ファイルで全てのデータを確認できる．

以上から，機器・システムにおいては私たちの過去の作品と比較すると，格段に優れたものを製作できた．このプロジェクトは成功したといえる．

10.4 次回へ向けて

今回の作品の問題点は，10.3 で述べたように，打ち上げを行っていないため，高高度のデータが収集できていない点である．これは，機器の完成時期が秋であったためである．前回までの経験から，その時期に打ち上げると季節風に流されて高度があまり上がらないので，打ち上げは断念した．しかし，9 章や，10 章などにあるように，優秀な機器を作ることはできたため，次回の気象観測機器コンテストではほとんどシステムを変える必要はないだろう．よって，次回は今回より早く完成させ，比較的風の影響の小さい初夏での打ち上げが可能ではないかと考える．

11. 感想

私はセンサノードとゲートウェイの製作を担当しました．気球の製作は昨年度に引き続いて 2 度目でしたが，それにふさわしいものが製作できたのではないかと思います．昨年度は初めて気球製作に携わったということもあり，ろくに気球の知識もないまま，正常に動作するだけの機器をなんとか製作しました．

今年度は，少しでも利用しやすく，気球に興味を持ってもらえるような機器の製作を目標にしました．メールで測定データを送信するシステムや，カメラを用いて上空を撮影するシステムはその一例です．高高度の気象観測機器というと最初は少し身構えてしまいましたが，機器の原理は地上のものとほとんど変わらず，一昨年以前に参加していた気象観測機器コンテストでの知識を存分に活かすことができました．

私が気象観測機器コンテストに参加するのは今年が最後になるかと思います．このコンテストは私の中で 4 度目ですが，3 万メートルまで浮上する気球を目にすることができないまま，後輩にこのプロジェクトを引き継ぐことになるのは残念でなりません．しかし，今後も卒業研究等で成層圏気球や，更に上空から気象データを測定する Cube-Sat の製作に携わり続けたいと思います． (電気情報工学科　4 年　大西 哲)

私は今回で気象観測機器コンテスト 4 回目の出場となります．また，成層圏気球プロジェクト(Examine Project)も 4 度目になります. Cloud Examine,「ガッスン」From Examine Project, Stratosphere Examine Satellite (成層圏観測衛星) 「SES」，そして「PENTA」と 4 つを見てきて思ったことがあります．

1 つ目は回を増すごとにキャリアの重量が軽くなっているということです. Cloud Examine のキャリアの重量は 3 kg ととても重いものでしたが，今回の PENTA はキャリアの重量が 1 kg と始めに比べて 1/3 も軽量化されていたのです．特に軽量化を意識しようと思ったことはなかったのですが，既製品の使用を極力控える，低コストで制作するというのを目標にしていると自然と無駄なものは減り，重量も軽くなっていくのかなと思います．

2 つ目は，データの取得率，安定率，正確性が格段にアップしたということです.「ガッスン」の時にデータが取れず悔しい思いをしたということを教訓に安定してデータを取得することができる，さまざまな気象を測定する，などデータの取得方法も年々研究してきました．今回は新たな試みでデータを SMTP で送信するという方法を採用しましたが，結果は良好で，メール送信に遅延やバグ等は発生せず安定的にデータを取得することができました．

4 年間 Examine Project と題して成層圏の気象研究を行ってきましたが，やはり，一人では成し遂げることができない大きなプロジェクトであったと思いました．また，4 年間プロジェクトのリーダー的なことをやってきて，バルーンを飛ばした時の安全に対する責任の重

さや，失敗に対する不安などを多く感じました.

最後に，成層圏の気象について，自らが制作した機器で測定，考察できたことは非常に面白く，なかなかできないことだと思いました．私は今年で気象観測機器コンテストへの出場はおそらく最後となります．しかし，Examine Project はここで終わりではありません．今まで培った技術，知識を優秀な後輩に委ね，さらにプロジェクトが大きく発展すると私は信じています．私は卒業研究等でこの 4 年間の経験を生かし，さらに気球や，成層圏の気象について知識を深めていきたいと思います.

このような貴重な体験を 4 年間にもわたりさせていただきましたこと，チームを代表し深くお礼申し上げます．ありがとうございました． (電気情報工学科　4 年　新谷 悠真)

私は今回で本コンテスト 4 回目の出場となります. なお, 成層圏気球 (Examine Project) での参加は 3 回目になります. 1 年生のときから参加していますが, そのころと比べると知識, 技術，経験が向上し，いいものがつくれたのではないかと思います.

当部活動では，本コンテストをメインイベントの一つとしています．豊富な予算をいただけるお陰で，充分な機材を購入することができます．そのため毎年新しいことに挑戦して，自分たちの技術と作品の質の向上を図れ，とても感謝しています．また，本コンテストに参加することで，プログラミング，ネットワークなど，私たちの専門である工学系の知識・技術だけでなく，気象に関する知識も得ることができました．特に今回は逆転層という，この活動をしていなければ知ることはなかっただろう現象を知ることができました.

私たち 4 年生は今年で本コンテスト出場はおそらく最後となります．気球の打ち上げの断念など，悔いがないといえば嘘になります．しかし，当部活動には信頼出来る優秀な後輩たちがいるので，次回以降，本コンテストでのお目見えを期待していただきたいとおもいます． (機械電子工学科　4 年　前川 雄壱)

12. 付録

本章では，機器の動作フローチャートと wvdial の設定方法を示す．

12.1 機器の動作フローチャート

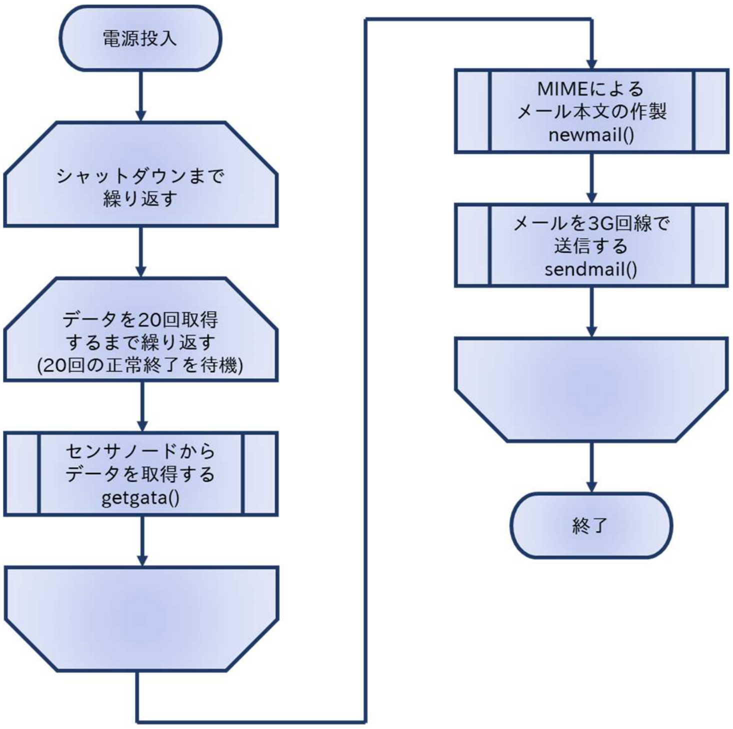


図 12.1　ゲートウェイの動作フローチャート (メインプログラム)

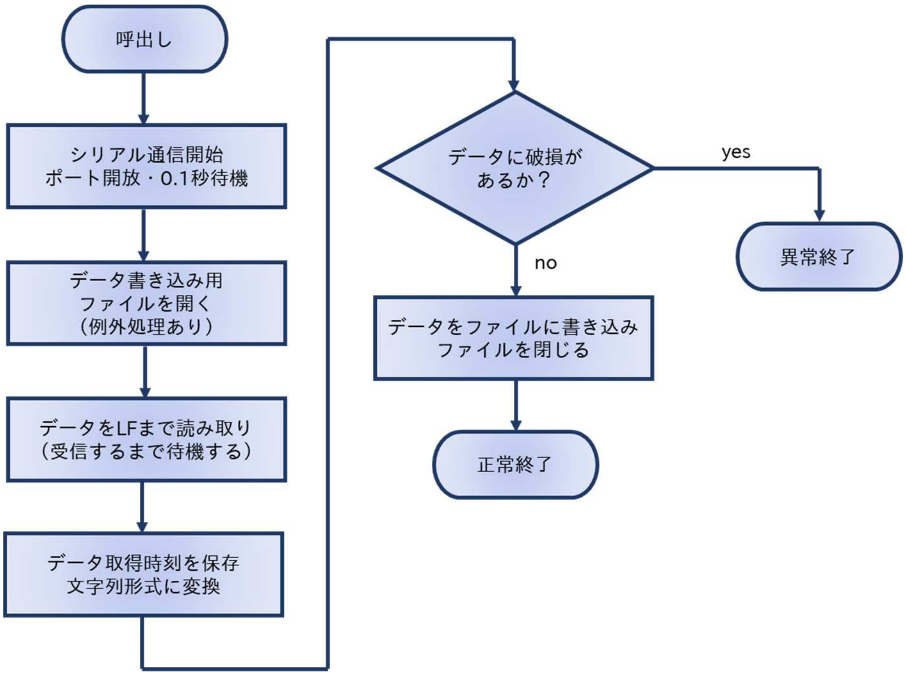


図 12.2 ゲートウェイの動作フローチャート (データ受信部)

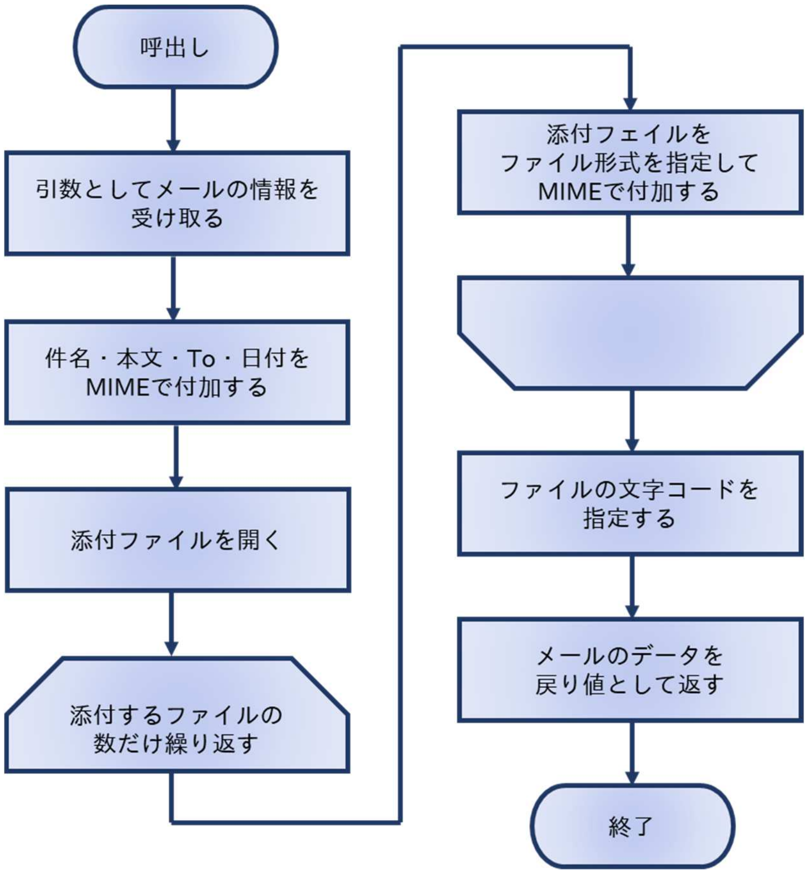


図 12.3 ゲートウェイの動作フローチャート (メール作製部)

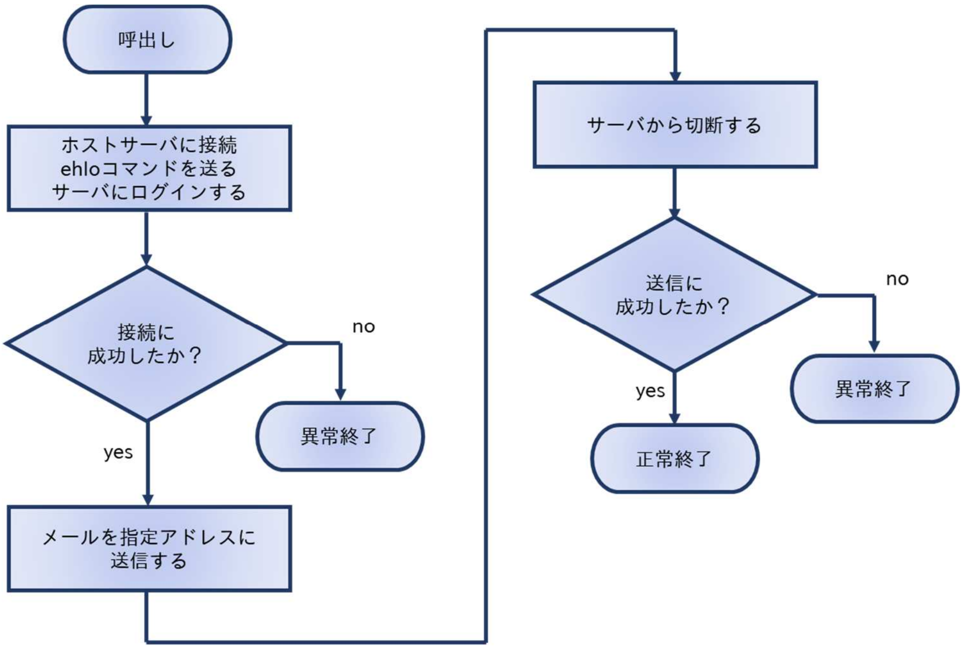


図 12.4　ゲートウェイの動作フローチャート (メール送信部)

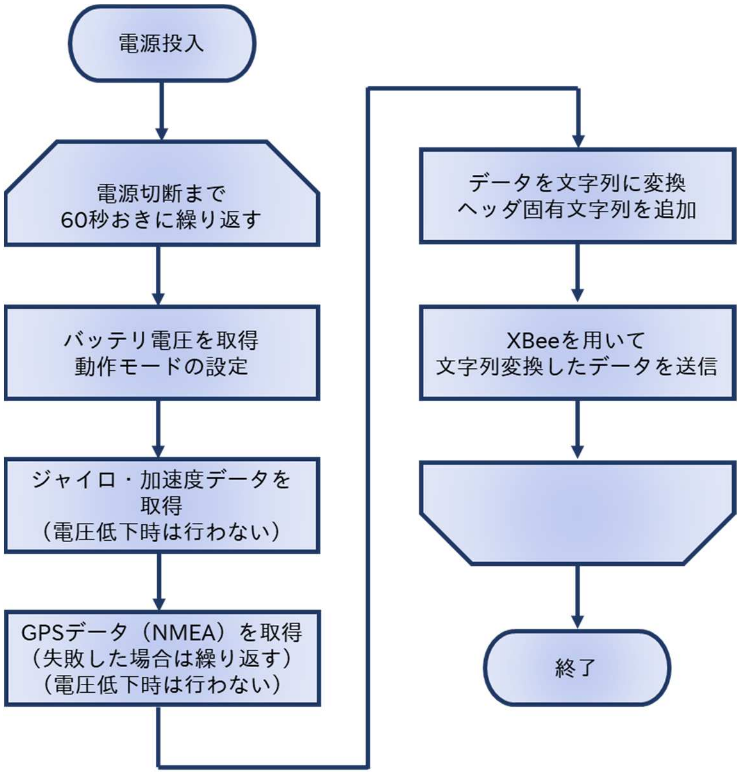


図 12.5 センサノード (mbed) の動作フローチャート

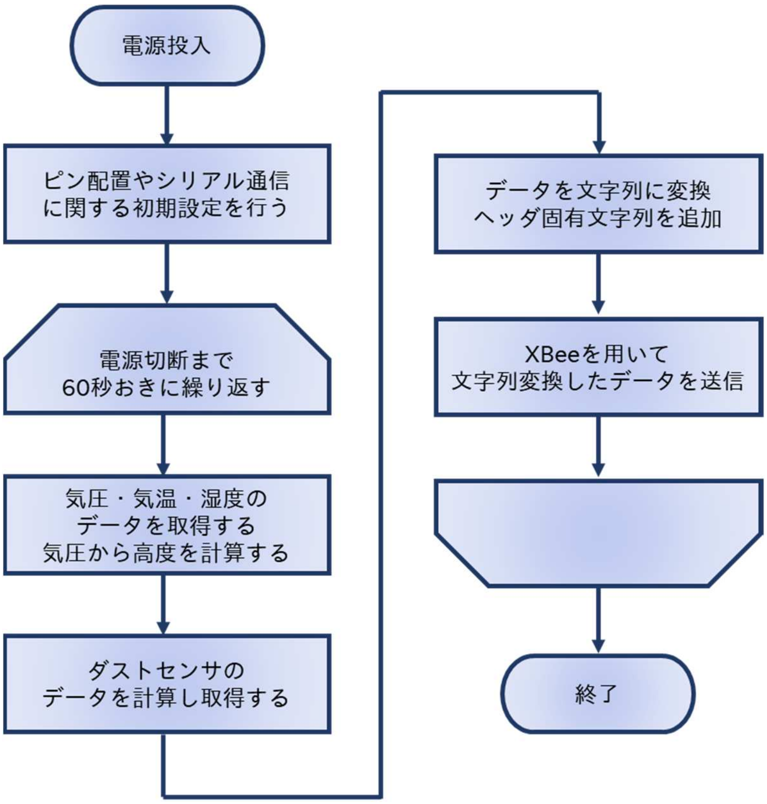


図 12.6 センサノード (Arduino) の動作フローチャート

12.2 Wvdial の設定方法

①Wvdial のインストール

$sudo apt-get install wvdial で Wvdial をインストール

②ポートの固定

次の③でポートの指定を行うため, Xi が使用するポートを固定した．ポートの固定には usb_modeswitch というプログラムを使用した．

$sudo apt-get install usb_modeswitch で usb_modeswitch をインストール．

$sudo leafpad /etc/ usb_modeswitch.conf で設定を変更, 下記の赤丸で示した部分を追記する．

```
# Configuration for the usb_modeswitch package, a mode switching tool for
# USB devices providing multiple states or modes
#
# Evaluated by the wrapper script /usr/sbin/usb_modeswitch_dispatcher
#
# To enable an option, set it to "1", "yes" or "true" (case doesn't matter)
# Everything else counts as "disable"


# Disable automatic mode switching globally (e.g. to access the original
# install storage)

DisableSwitching=0

# Disable check for MBIM module presence and configuration globally (to aid
# special embedded environments)

DisableMBIMGlobal=0

# Enable logging (results in a extensive report file in /var/log, named
# "usb_modeswitch_<interface-name>" and probably others

EnableLogging=0


# Optional increase of "delay_use" for the usb-storage driver; there are hints
# that a recent kernel default change to 1 sec. may lead to problems, particu-
# larly with USB 3.0 ports. Set this to at least 3 (seconds) in that case.
# Does nothing if the current system value is same or higher

#SetStorageDelay=4

# LG L-02C LTE
DefaultVender=0x1004
DefaultProduct=0x61dd
TargetVendor=0x1004
TargetProduct=0x618f
```

③SSID 等の設定

$sudo leafpad /etc/wvdial.conf で設定を変更

```

[Dialer Defaults]
Init1 = ATZ
Init2 = ATH
Init3 = ATQ0 V1 E1 S0=0 &C1 &D2 +FCLASS=0
Init4 = AT+CGDCONT=1,"IP","lte-d.ocn.ne.jp" #SSIDを設定

Dial Attempts = 3
Stupid Mode = yes
Modem Type = Analog Modem #ルーターの種類を設定
Dial Command = ATD
Baud = 460800
New PPPD = yes #pppdの許可
APN = ocn.com #APNの設定
Modem = /dev/ttyUSB2 #使用するポートを設定
ISDN = 0
Phone = *99***1# #電話暗号を設定
Password = mobile #キャリアのパスワード
Username = mobileid@ocn #キャリアのユーザーネーム
Carrier Check = no #エラーチェック
Auto DNS = 1 #DNS自動取得
Check Def Route = 1

```

④プログラムの起動

$sudo wvdial で Wvdial を実行する．以下のように connect と表示され，DNS address が取得できれば成功である．

```
$ sudo wvdial
--> WvDial: Internet dialer version 1.61
--> Initializing modem.
--> Sending: ATZ
ATZ
OK
--> Sending: ATH
ATH
OK
--> Sending: ATQ0 V1 E1 S0=0 &C1 &D2 +FCLASS=0
ATQ0 V1 E1 S0=0 &C1 &D2 +FCLASS=0
OK
--> Modem initialized.
--> Sending: ATD*99***1#
--> Waiting for carrier.
ATD*99***1#
CONNECT
--> Carrier detected.  Starting PPP immediately.
--> Starting pppd at Sun Nov 23 12:04:47 2014
--> Pid of pppd: 3403
--> Using interface ppp0
--> local  IP address 153.157.38.xxx
--> remote IP address 10.64.64.64
--> primary   DNS address 222.146.35.1
--> secondary DNS address 221.184.25.1
```

⑤起動実行の設定

$sudo leafpad /etc.rc.local で起動実行の設定を行う．最後の行に $sudo wvdial -n & を追加して保存，再起動する．

13. 参考資料

[1] Arduino

https://store.arduino.cc/usa/arduino-uno-rev3

[2] Arduino Wireless Shield

https://store.arduino.cc/usa/arduino-wireless-proto-shield

[3] Raspberry Pi

https://www.raspberryPi.org/products/raspberry-Pi-2-model-b/

[4] BME280

https://www.seeedstudio.com/Grove-TempHumiBarometer-Sensor-BME280-p-2653.html

[5] Base Shield

https://www.switch-science.com/catalog/1293/

[6] XBee PRO ZB S2B

https://www.switch-science.com/catalog/2611/

[7] mbed

https://os.mbed.com/

[8] HEPTA

http://unisec.jp/service/hepta.html

[9] Grove - Dust Sensor

http://wiki.seeed.cc/Grove-Dust_Sensor/

[10] CUSF Landing Predictor 2.5

http://predict.habhub.org/

[11] Xi(L-02C)

https://www.nttdocomo.co.jp/support/trouble/data/l02c/

[12] Raspberry PI+docomo L-05A+SORACOM Air で USB モデム自動起動
https://qiita.com/hiroyasu55/items/e0ad9d04a9d8feb5c654

[13] USB 3G モデム L-02C を Raspberry Pi 3 で使えるようにして MVNO の SIM カードで通信する
https://signal-flag-z.blogspot.jp/2016/11/usb-3g-l-02c-raspberry-Pi-3.html? m=1

[14] Raspberry Pi で USB モバイルモデム
http://hashimototakao.blogspot.jp/2015/05/raspberry-Pi.html?m=1

[15] インターネット接続設定(OCN)
http://support.ntt.com/ocn/support/Pid2900000584

[16] ラズパイで 3G データ通信
http://www.drone-engineer.com/articles/220/

[17] Raspberry Pi で L-02C
http://uepon.hatenadiary.com/entry/2017/04/08/115936

[18] 株式会社気球製作所
http://www.weatherballoon.co.jp/pages/japanese/1cosmo.html

[19] Google map
https://www.google.co.jp/maps

[20] OCN モバイル ONE プリペイド公式サイト
http://www.ntt.com/personal /services/ mobile/one/prepaid.html

[21] Simon Monk (2014)『Raspberry Pi クックブック』株式会社オライリー・ジャパン

[22] 『日経 Linux 2017 年 9 月号 (通巻 216 号) 』日経 BP 社

[23] 小倉義光 (1994)『お天気の科学』森北出版

[24] 2013ExamineProject「高層気象観測バルーン」(第 2 回気象観測機器コンテスト)

[25] 2014ExamineProject「Cloud Examine」(第 3 回気象観測機器コンテスト)

[26] 2015ExamineProject「ガッスン」(第 4 回気象観測機器コンテスト)

[27] 2014 気象観測機器コンテスト「KOINOBORI PROJECT」

[28] 2015 気象観測機器コンテスト「ヘイズカスーミー　～視界距離自動判定装置～」

[29] 2016ExamineProject「Stratosphere Examine Satellite(成層圏観測衛星)「SES」」

[30] HEPTA ハンズオン・トレーニング in 香川高専資料