Mark3製作マニュアル

～2017年度版～

Hirokazu.Yamagata

©Hikasa Ayano

KEIO.UP UNDERWATER PIONEER 非公認

本マニュアルに関する権利と責任

キット・マニュアル・設計データについて下記の条件を満たした場合に自由に活用することが出来ます.

1. 使用の条件
   本 HP 上に有る, キット・マニュアル・設計データは教育及び研究に活用する場合において, 無償でこれを利用することを許可します. これに当たらないものについては, Contact より作者に連絡して許可を得て下さい.
2. 著作権表示
   本 HP にて確認ができるキット・マニュアル・設計データは山縣広和(YAMAGATA Hirokazu)に帰属します.
   教育や研究においてこれを活用するに当たって引用が必要な場合は, そのコミュニティにおいて定められた引用ルールに基づいて HP や関連したデータ群の引用を明記して下さい. 学術研究などによる論文文献の引用が必要な場合は, Research 内の論文で該当するものを引用して下さい.
3. 変更箇所の明示
   使用条件を満たして二次配布を行うにあたって, 内容に加筆修正を加えた場合はその変更箇所を明示して下さい.
4. 商用利用
   許可なき商用利用に関してはこれを禁止します.
5. 配布
   使用条件を満たした, 対価の発生しない二次配布についてはこれを許可します. 判断が難しいものについては, Contact より作者に連絡して確認して下さい.
6. 免責
   本 HP 上に有る, キット・マニュアル・設計データを用いた行為について発生したあらゆる不利益について一切の責任を負いません. 上記内容の閲覧利用については全て自己責任とします.
   ・リンクされている外部サイトについて, 何らの保証はしません.
   ・上記内容の読者, 利用者に対して適応される法令に対し, 本 HP を通して提供される内容が合法であることを一切保証しません.
   ・予告なくコンテンツの内容を改変, 提供や公開を中止する場合があります.

以上.

Mark3 製作マニュアル

H.Yamagata

2017 年 7 月 19 日

目次

概要 i

第 1 章 はじめに/本書の読み方 1
1.1 本書の目的 1
1.2 PDCA サイクルを回そう 2
1.3 つまずいた時は 3
1.4 本書の PDCA 5

第 2 章 水中ロボットの動きを理解ためのお約束 6
2.1 水中ロボットに関連する力学 6
2.1.1 座標系 6
2.1.2 力のベクトル表現 9
2.1.3 作用反作用 10
2.1.4 力のつりあい (並進) 11
2.1.5 力のモーメント 12
2.2 水中ロボットにはたらく力 13
2.2.1 推力 14
2.2.2 重力 14
2.2.3 浮力 14
2.2.4 流体抵抗 15
2.3 本章のまとめ 16

第 3 章 水中ロボットの分類について 18
3.1 なぜモノを作るのか？ 18
3.2 水中ロボットの分類 20
3.2.1 目標達成型ロボットの分類 21
3.2.2 グライダ型水中ロボット（広範囲・みる） 22
3.2.3 航行型水中ロボット（中範囲・みる） 23
3.2.4 ホバリング型水中ロボット（狭範囲・みる/とる） 24
3.2.5 建設機械型水中ロボット（狭範囲・とる） 25
3.2.6 研究開発型ロボットの分類 26
3.3 本章のまとめ 26

第 4 章 ハードの作成 27
4.1 製作の前に . . . 27
4.2 機体の構造 . . . 27
4.3 耐圧殻を製作する . . . 28
4.3.1 耐圧殻の構造 . . . 28
4.3.2 耐圧殻の加工 . . . 29
4.3.3 耐圧殻の防水 . . . 29
4.3.4 耐圧殻の防水テスト . . . 30
4.4 メインフレームを製作する . . . 31
4.4.1 フレームの切り出し . . . 31
4.4.2 面ファスナの貼り付け . . . 32
4.4.3 組み上げ . . . 33
4.5 スラスタを製作する . . . 33
4.5.1 灯油ポンプの分解 . . . 33
4.5.2 モータの交換 . . . 34
4.5.3 水密作業 . . . 34
4.5.4 プリンタパーツの準備 . . . 35
4.5.5 各部品の組み立て . . . 36
4.5.6 スラスタの動作テスト . . . 37
4.6 まとめ . . . 37

第 5 章 電子回路の作成 38
5.1 モータドライバの作成 . . . 38
5.1.1 モータドライバの構造 . . . 38
5.1.2 部品の準備 . . . 40
5.1.3 ハンダ作業 . . . 40
5.1.4 各部の確認 . . . 41
5.2 マイコンボードの作成 . . . 41
5.2.1 マイコンボードの構造 . . . 41
5.2.2 部品の準備 . . . 42
5.2.3 ハンダ作業 . . . 42
5.2.4 各部の確認 . . . 42
5.3 USB シリアル変換ユニットの作成 . . . 43
5.4 各ケーブルの圧着 . . . 44
5.5 まとめ . . . 45

第 6 章 マイコンとパソコン接続 46
6.1 Mark3 のソフト構成について . . . 46
6.2 ソフトのインストール . . . 47
6.3 回路の接続 . . . 49
6.4 動作の確認 . . . 50

6.5 まとめ 51

第 7 章 浮力調整とメンテナンス 52
7.1 水に入れる前に 52
7.2 浮力を調整する 52
7.3 水に入れたあとに 53
7.4 まとめ 53

第 8 章 改造と最適化をすすめる 54
8.1 改造とは何か 54
8.2 目標を決めよう 54
8.2.1 目標を明確にしよう 54
8.2.2 問題を洗い出そう 56
8.2.3 目標を明文化しよう 58
8.2.4 作業を具体化しよう 58
8.3 実際に改造をしよう（ハードウェア編） 60
8.3.1 スラスタの配置を変更する 60
8.3.2 フレームを作りなおす 61
8.3.3 メカをつくる 61
8.3.4 カメラを作る 64
8.4 実際に改造をしよう（ソフトウェア編） 66
8.4.1 ソフトウェアの詳細 66
8.4.2 プログラムを書き換えてみる 68
8.4.3 アクチュエータを増設する 69
8.5 センサを搭載してみよう 72
8.5.1 内界センサと外界センサ 72
8.6 完成品を評価しよう 73
8.6.1 追加した機能が要求を満たしていない 74
8.6.2 着目する対象が間違っている 74
8.6.3 気づかない相互作用がある 74

第 1 章

はじめに/本書の読み方

1.1　本書の目的

このマニュアルの目的は，水中ロボコンのジュニア部門に向けて，参加者の皆さんが水中ロボットを作るにあたって必要となる情報をまとめたものです．でも，マニュアルと言っても，作り方がただ書かれているだけのものではありません．ひどく大雑把に言ってしまうならば，例えば目の前にキットがあって組み立て説明書があれば，相当にひどい説明書であったり難易度が高過ぎることがなければ誰でも動くものが作れるでしょう．実際に皆さんは，そういった経験を部活動や趣味であれば日常的に，そうでない人も中学校の技術・家庭科でキットを作ったりする途上でそのような経験していると思います．~~本書としても~~筆者としても，上記のものと同様のクオリティのマニュアルを提供したいと考えておりますし，実際に提供しようとしています．[*1]

しかし，「ロボコンで勝つ」とか「自分にとり好ましいモノを作ろう」とか考え始めれば，その事情は大きく異なります．

例えば図 8.26 のような車体のロボットを考えてみましょう．

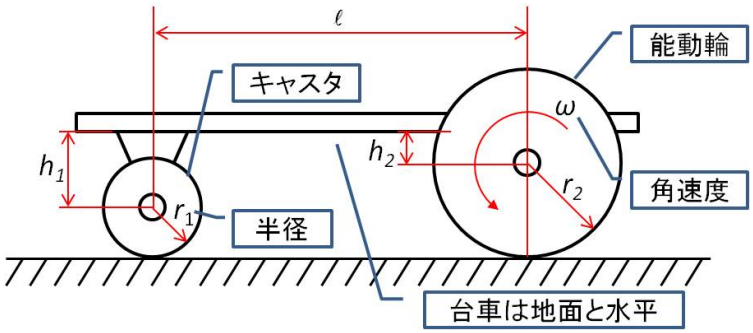


図 1.1　タイヤロボット

この機体を高速化（最高速度を引き上げる）しようと考えた時に皆さんはどうしますか？この質問に適切な方法と結果を解答するためには，少なくとも物理や数学についての知識があることと，それを適切に使うために問題のキモになる部分に着目する能力を身につけている必要があるでしょう．さて，思いつくであろう方法では以下の様な物があるでしょうか．

[*1] 筆者も実は執筆が始めてなので，どうなるかはやってみないとわかりません．

- タイヤを大径化する
- タイヤの角速度（回転速度）を向上する

これ，どちらをすることが有効でしょうか？もちろん，「どちらも」とることができればそれがベストな回答でしょうが，実際にはモータなど機能や，大きさ制限のルール，予算，時間などの制約によって採れない選択肢が大量に生まれるという現実があります．例えば，今回の場合は以下の内容が制約になります．

- 車体が水平である
- タイヤとキャスタ間の距離が固定されている

では，どうすればそういった範囲の中でもっともよい方法を選べばよいのか？本書では，こういった問題の解決のための考え方と，継続的にそれに取り組む方法を皆さんに習得してもらえればと考えています．具体的には以下のような内容です．

原理・原則の習得　物理の基礎についてまとめます．議論・検討のための言語として用いるためです．
目標の設定・検討　目標を設定し，達成するためにその内容を細分化/明確化する方法について考えます．
改造・改善の方法　設定した目標に対して，なるべく良い手段を選択する考え方を解説します．

*

1.2　PDCA サイクルを回そう

さて，これを身につけるために，皆さんには以下の様な行動を常に考え，繰り返して欲しいと思っています．

- 問題を設定する
- 問題の解法を考えて実際に問題を解く
- 解いた結果が問題を解決しているのか，意味があったのかを考える
- 結果を元に更なる改善する

「問題を設定できる」ということはごく当たり前に見えるでしょう．しかし，実際にはこれが実は最も難しいのです．実際に，大学などで行われる研究活動や新しい法律を作るなどの行為において，最も重要視される点は「改善するべき現象において，本当に問題になっているのはどこであるか」を見極めて問題を設定することなのです．「解法を考えて解く」にあたっては，過去の事例などから探せば問題なく解けるものが多いです．そして，「解いた問題の結果がどういった意味を持っているか」考えます．結果を客観的に評価して，うまくいった部分を伸ばしたり，ダメだった部分を補強することを考えます．ヘタをすれば問題設定時点が誤っている場合もありますから，これを見直すことも必要です．そのためには，「問題設定」の時点で，どのようにして評価を行うべきか考えることが重要でしょう．そして，「結果を元にした改善」は，その結果を元にして次に何をするべきか検討して行動することを指します．これらの行為の繰り返しを一般には PDCA サイクル (Plan Do Check Action Cycle) と呼びます．（図 1.2）このサイクルを繰り返していくことにより，物事の質を高めることが出来ます．

例えば，学校の授業に当てはめるならこのようになります．

Plan　科目で学ぶ内容を設定する

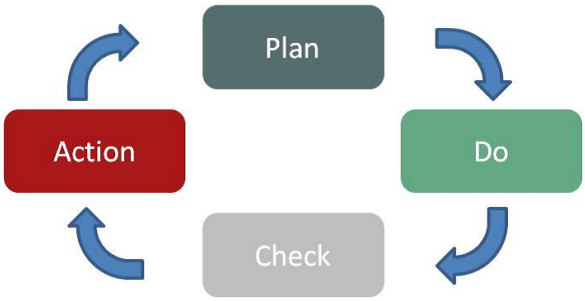


図 1.2　PDCA サイクル

Do　　授業をうけて，復習する
Check　テストを受けて結果を知る，答え合わせをして理解していない部分を確認する
Action　理解していない部分を復習する

今回の水中ロボット作成ならこうでしょうか？

Plan　機体開発の目標を定める（大きさ，重さ，速さ，動作の条件，など）
Do　　機能試験などを行いつつ実際に作成する
Check　機体の性能を確認する，大会に出場する
Action　結果を元に一度改善してみる．その後，次に何を行うかを検討して Plan に戻る．

このサイクルさえ身につけてしまえば，どんな内容にも当てはめて改善を継続する事ができるでしょう．また，他者の行動に当てはめて見ていけば，その人の成功の秘訣や，失敗の理由が見えるかもしれません．

1.3　つまずいた時は

PDCA サイクルがあれば大丈夫と言ってみたものの，このマニュアルのクオリティのせいで皆さんが Do の最中につまずくことも有ると思います．そんな時にはまず，わからない部分は何かを確認して，その部分に戻って下さい．一応，本書ではマニュアル全体や各章で PDCA が構成されるような執筆方法を採っています．ですから，Do の最中にわからないことがあれば，Plan の部分を再確認して下さい．その章の Plan の意図がわからないならば，更に前の章へ戻れば良い（はず）です．図 1.3，1.4 に章の構成と読み方のフローチャートを挙げます．

始めてのことへの挑戦は，予測できない位置から急に出てくるハードルをいかにして超えるかを問われる行為です．筆者としては皆さんが順繰りに超えていけるようなハードルを用意しているつもりでありますが，残念なことに私も執筆というものは始めてです．ハードルのふりをした鳥居を用意していることがあるかもしれません．[*2]ですから，戻ってすら解決が難しい問題に遭遇した時は，先人（＝学校の先生方や運営/筆者）へ質問をして下さい．質問を遠慮する必要はありません．皆さんのために自分たちよりもより質の高い学びの環境を提供することは先人の義務です．

[*2] ハードルは前を向いて乗り越えるもの，鳥居は頭を下げて拝んで潜るもの．

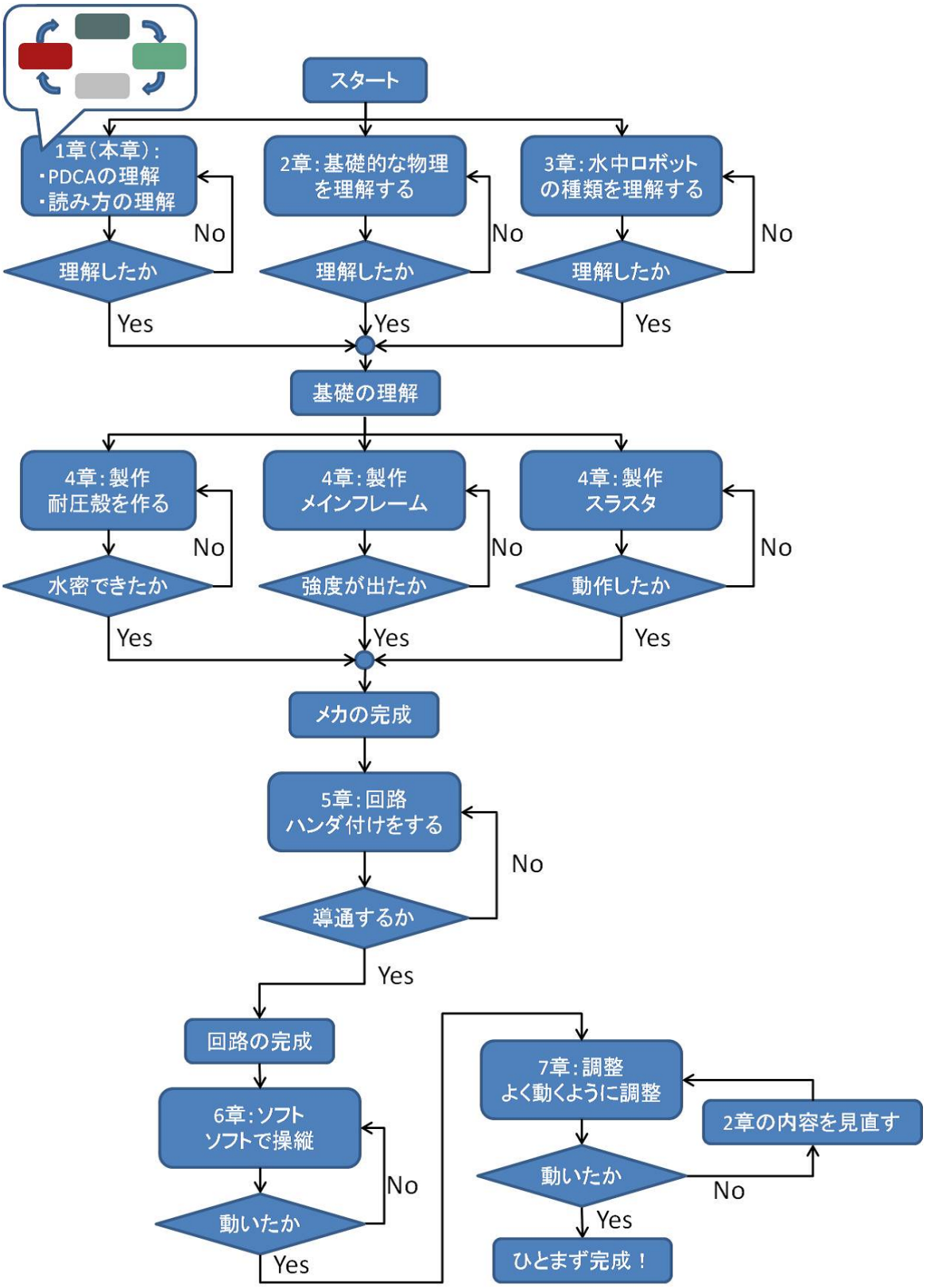


図 1.3　本書の読み方前半

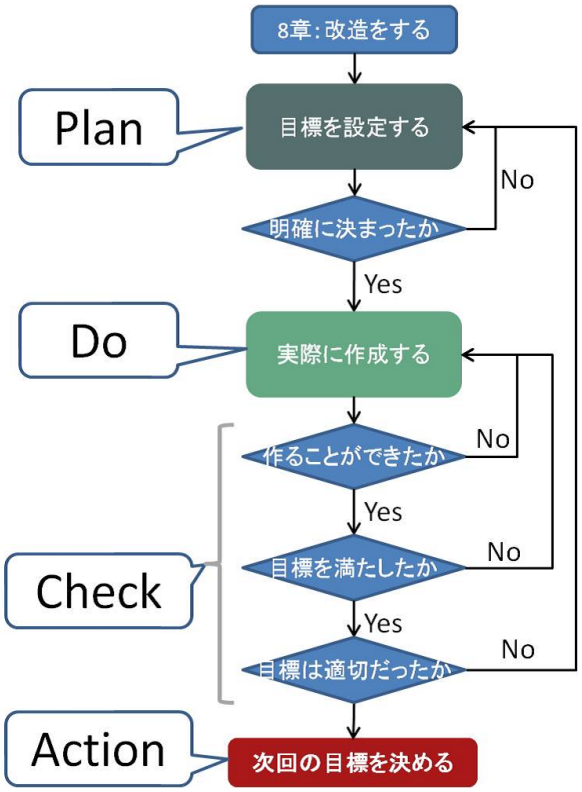


図 1.4　本書の読み方後半

1.4　本書の PDCA

ということで，本書が情報を提供することで，皆さんに実践してほしい PDCA を挙げて本章を終わります．

Plan　本書を読み，キットを製作することで水中ロボットをある程度自由に作れる能力を読者が身につけて継続的に大会に参加する．（して下さい）

Do　本書を元に，キットを製作・改造を行う．（して下さい！）

Check　読者自身で機体の動作確認を行う．大会に参加（して下さい！！）して，ほか参加者と自身の相対評価を行う．

Action　次に行うべき改造を設定し，継続的に大会に参加する．（して下さい！！！お願いします！！！！）

それでは次の章からまず，水中ロボットの種類と分類の説明に移っていきたいと思います．よろしくお願いいたします．

第 2 章

水中ロボットの動きを理解ためのお約束

本章の PDCA は以下のようになっています.

Plan　読者が水中ロボットの動きについて考えるに当たって必要な力学を理解する
Do　　読者が章を熟読する．必要に応じて物理の教科書を確認する．
Check　図に描かれた力の総合から，機体の動作がイメージできる
Action　次章の内容を，力学から理解することを目指す.

筆者的には，水中ロボットの分類の話や機体の製作の話にどんどん進みたいのですが，その前にまず水中ロボットがどうやって動いているのか，その仕組や力学的な原理について先に理解してもらおうと思います.[*1]まずは初歩の初歩，中高の物理学と，それにすこし進んだ物理の内容について，これでもかというくらいに簡略化して説明したいと思います．基本的内容は，中学や高校で学ぶ理科や物理の内容です．よって，読み飛ばしても全然理解できる読者はたくさんいると思いますが，熟読を奨めます．ここの部分を抑えておかないと，本当の意味で狙った通りに動く水中ロボットは作れません.

2.1　水中ロボットに関連する力学

2.1.1　座標系

座標系，皆さん見たこと有ると思います．例えば，おそらく中高の数学で描くグラフは縦軸（大体 y 軸）と横軸 (x 軸) の 2 軸で座標軸を定義して，その上に点を描画して検討を行うものと思います．これは，２つの変数を含むので，二次元直交座標系というやつです．皆さん，中高で二次曲線と直線の交点とか解いた記憶があるでしょう.[*2]これを使えば平面的な位置や運動を説明することが可能になりますよね．例えば，囲碁や将棋やチェスをやったことがある人であれば，自分の駒の位置を説明するために縦横の値を使って説明するのは一般的であることを知っているかと思います．ロボットにも同じように座標軸を定義して，動きの方向や動き方を説明します．但し，水中ロボットは三次元で動作しますから，軸は 3 つ必要になります.[*3]水中ロボットは重心を原点として進行方向に x 軸，重力の向きを z 軸として右手座標系を定義します．右手座標系と言うのは，図 2.1 のように右手の親指，人差し指，中指を直交するように

[*1] 長ったらしい前提の説明ですね．理系という生き物は前提とか定義といった事前の「お約束」を大事にします．なぜかといえば，ものの見え方は方向によって変わってしまうから，前提を決めておかないと無用な議論が生まれるためです．でも折角前提を用意しているのに，その前提にまでツッコミが入ることも有るんですよ．嗚呼.

[*2] ありますよね？

[*3] マイ○ラとかの３ D ゲームをやりこんでいるヒトには馴染み深いと思います.

開いた時，親指から順に，x/y/z と定義した状態を指します．また，更にその各々の軸方向に対して右ねじを正とした方向への回転を「ロール」,「ピッチ」,「ヨー」と定義します．図 2.2 のような形です．右手座標の図と比べて理解してもらえればと思います．また，物体には大きく分けて 2 つの動きがあります．x,y,z 軸で分解できる方向に移動することを並進と言います．そして，ヨー，ピッチ，ロールの動作を行うことを回転といいます．さて，今の考え方で，機体の重心に座標をとったのが図 2.2 ですが，当然機体が傾けば座標も傾きますね．では，どうすれば機体の傾きを客観的に説明できるでしょうか．これにはもうひとつ動かない座標をとって，その座標軸どうしのなす角を見ていけば角度を決めることができます．当然，固定する位置がわかっていれば，その固定された座標との相対距離で移動量なども説明できます．この，何があっても動かない座標を「ワールド座標系（世界座標系）」と呼びます．このマニュアルにおいて，ワールド座標系は，右手系で z 軸を重力方向にとり，x 軸方向は機体の初期方向と平行で定義します．今後の説明で，機体の座標が傾いた時はどこかに世界座標があると思っていてください．（必要があれば記述します．）

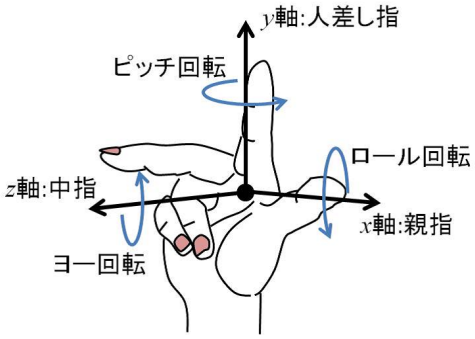


図 2.1　右手系

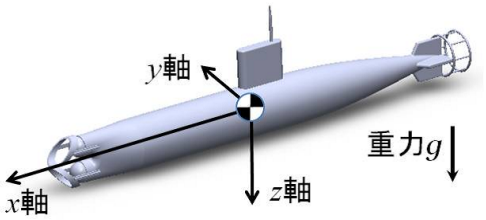
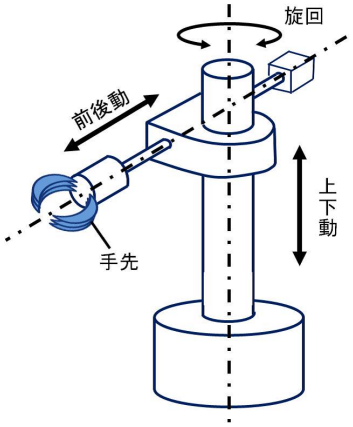


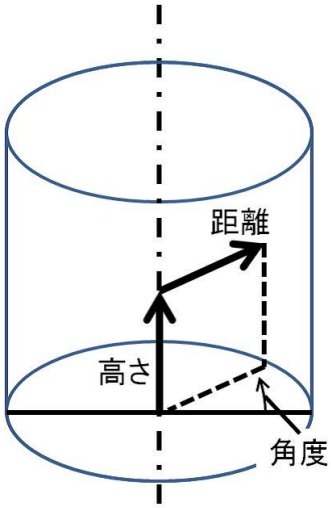
図 2.2　水中ロボットの座標系

色々な座標軸

今回は三軸直交座標系で水中ロボットの動作を説明しますが，世の中には他にも色々な座標系が有ります．ロボットであれば，円筒座標系や極座標系 (球体座標) などが使われる場合があります．これらは実際の動作に対応した形で軸を取ることでより効率的に位置や動かし方を考えることができます．例えば，図のようなロボットアームであれば，根本での旋回とアーム位置の上下，腕の長さの伸び縮みができます．これの位置を説明しようとすれば，先ほどの右手座標で説明をしようとすると，上下はさておき，腕の伸び縮みと根本の旋回に関しては sin,cos で計算をして求めなければならず，面倒です．しかし，ここで図のような円筒座標系を用いてやれば，高さ・中心からの距離・旋回角度でロボットの手先位置を定めることができ，話し合いをするお約束としては有効かつ理解しやすいわけです．



ロボットアーム



円筒座標系

2.1.2　力のベクトル表現

さらりと”ベクトル”という言葉を使いましたが知らない読者もいると思うので，ベクトルとスカラーについて説明から始めたいと思います．[*4]ベクトルとは「大きさと向きを兼ね備えた量」のことです．そして，スカラーは「大きさだけをもつ量」を指します．例えば，力や速度はベクトルです．作用している向きがあって意味をなします．天気予報の表現で「○○向きの風，風速△ m/s」などという表現を考えればイメージしやすいでしょう．一方，重さや面積，体積などはスカラーです．大きさだけが存在し，方向に意味がないのが想像できると思います．

さて，力のベクトル表現について説明したいと思います．力はベクトルなので，力の「作用している点」,「方向」と「大きさ」を持ちます．そこで，これを表現するために図 2.3 のように矢印を使って表現します．とても端的に言えば，これが力のベクトル表現です．

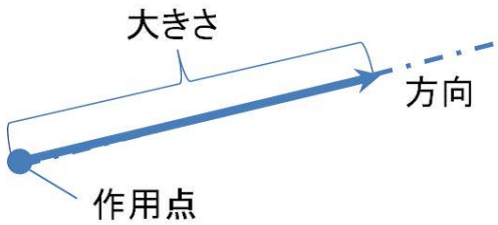


図 2.3　ベクトル表記

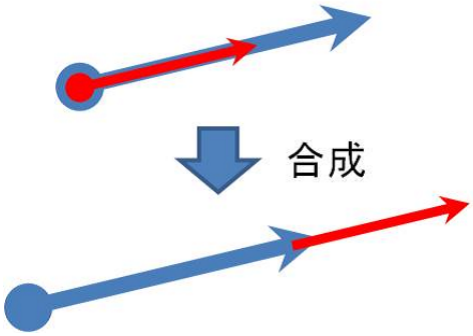


図 2.4　同軸上のベクトル合成

力のベクトルは合成や，分解が可能です．例えば，同じ方向を向いているベクトルは図 2.4 そのまま矢印の長さを足し算すれば OK です．しかし，当然のごとく，力は常に同じ方向を向いているとは限りません．そこで，方向の違うベクトルを合わせて合計して力の向いている方向を捉えるために合成と分解の手法を用います．普通はここで合成を先に説明しますが，あえて分解から説明します．さて，ベクトルには方向がありますが，どこを向いているかは基準を決めなければ説明できませんね．ということで，先ほど触れた座標系を使いましょうか．二次元直交座標系にベクトルを置いて考えます．座標軸のおかげで，どこから力のベクトルが発していて，どこを向いていているかが，ｘ軸やｙ軸となす角度（傾き）で説明ができますね．そして，座標軸を決めることで x 軸方向の力と y 軸方向の力に分解して考えることができます．分解が分かれば，合成はその逆をするだけです．図 2.5 であれば，x 軸方向のベクトルと y 軸方向のベクトルを合成したものが，元の力ベクトルです．同じ方法で，図 2.6 のように任意の向きを向いている 2 つ以上のベクトルを足し算して，その合わせた力の大きさと方向を考えることができます．では図 2.7 はどうですか．この場合は，最終的に作られた合ベクトルにはｙ軸方向の力がなくなっているのがわかります．これは，ベクトルを分解するとわかりやすいと思います．一度全てを座標軸の方向の力に分解してしまえば，一番はじめと同じ「同じ方向を向いているベクトルの足し算」と変わりません．最後に，図 2.8 のように合掌して両手を同じ力で押してみましょう．手が上に向かって進んでいくと思います．この時の右手が受ける力と左手が受ける力，そして合力のイメージが出来ますか？

[*4] 昔スカラー波とか話題になりましたね... え？知らない？

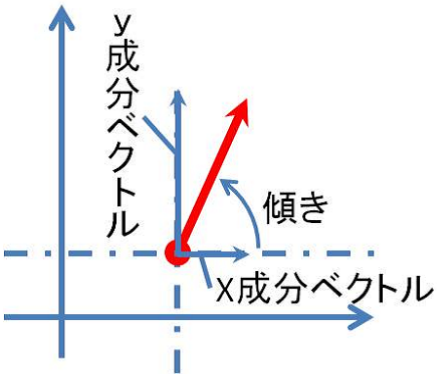


図 2.5　ベクトルの分解

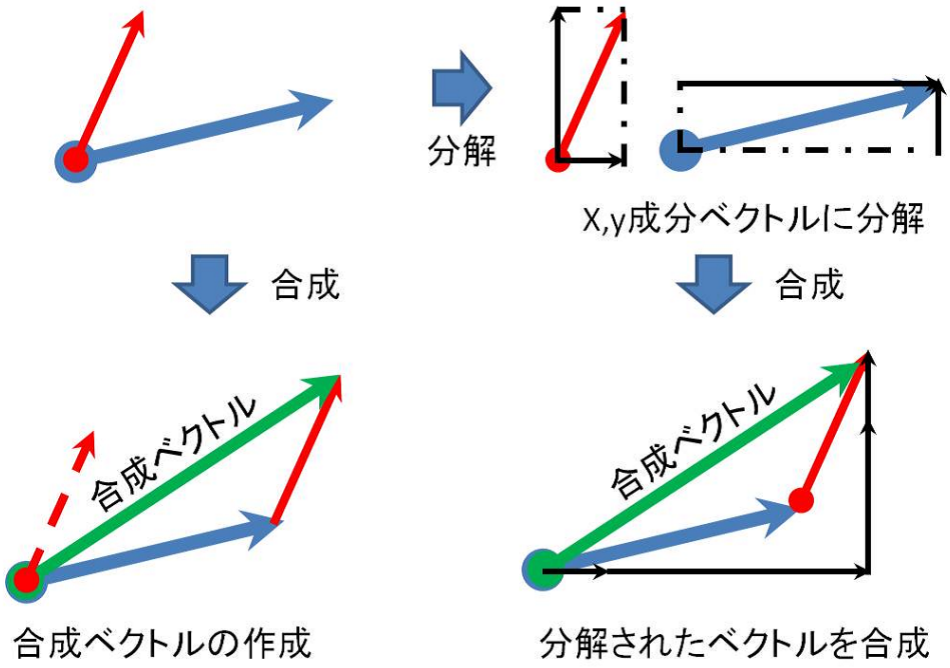


図 2.6　ベクトルの合成と分解

2.1.3　作用反作用

力は必ずある物体から別の物体に対して発生しています．振るう相手なしに力というものは存在しえません．そして，図 2.9 のように，力を与えた場合，与えられた側は同じ大きさの力を相手に返します．これが作用反作用です．同じ大きさの力が返ってくるので，作用点は同じで，ベクトルの向きは真逆で同一直線上になります．なお，重心に作用している重力と，物体が机を押す力と抗力は全て同一直線上にあるべきですが，そうすると見難いので，あえてずらして記述している点に留意して下さい．また，作用反作用に有るのは，物体が机を押す力と，机が物体を押す力であって，物体にかかる重力はこれらの関係では有りません．ただし，物体に関わる重力は物体が机を押す力であることから，力の大きさは同じです．

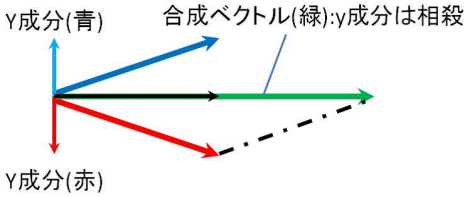


図 2.7　ベクトルの合成 2

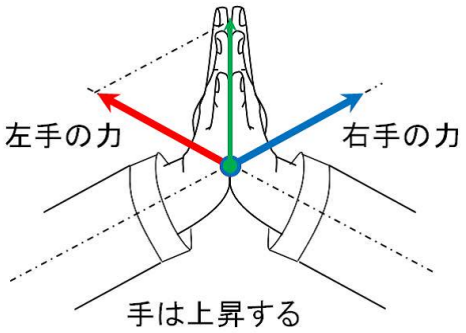


図 2.8　ベクトル表記

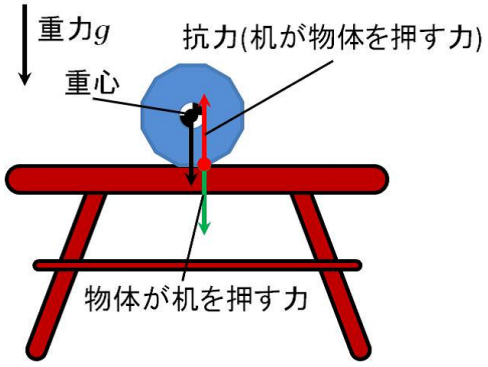


図 2.9　机と物体の作用反作用

2.1.4　力のつりあい (並進)

地球上であれば，質量のある物体には常に重力が働いていますね．ということは，少なくとも私達の目に見える物体は動いていなくても何かしらの力を常に受けているという事になります．例えばテーブルの上の物体が重力をうけている場合，その物体は反作用によって，テーブルから何らかの力を受けています．[*5]先ほどの図 2.9 を思い出して下さい．物体に関わる力は重力とテーブルから受ける抗力が有り，これらが釣り合うことで静止しています．

この物体に複数の力を受けている状態でありながら，物体が動いていない状態において「力がつり合っている」といいます．上の例では y 軸のみでの力のつり合いを論じていますが，横方向にもつりあいが成立します．例えば，図 2.10 のように 3 つの力が一つの物体に掛かっていて，つり合い状態にあるとするとします．このとき，すべての力を x 軸，y 軸に分解して考えればすべての力は各軸方向で 0 となります．

$$x \text{ 方向成分} = a + b - (a + b) = 0 \tag{2.1}$$

$$y \text{ 方向成分} = (c + d) - (c + d) = 0 \tag{2.2}$$

以上が並進方向における力のつり合いの基本的な考えとなります．そして，実はこの力のつりあいは動いている物体にも応用して考えることができます．この式を見たことは有りますよね？

$$Ma = F \tag{2.3}$$

[*5] これはテーブルだぞ，鳥居じゃないぞ．

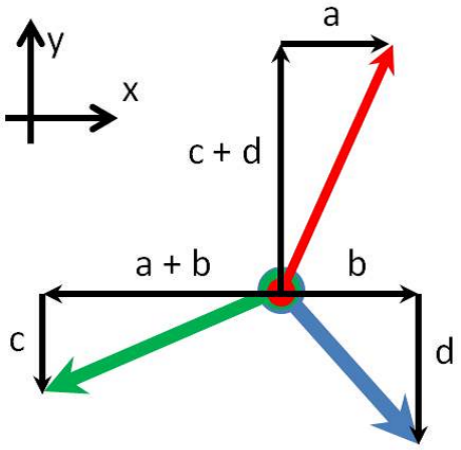


図 2.10　3 つのベクトルの合成

次の章で説明しますが，物体が加速していない状態を考えると，力 F は 0 になりますよね．ということは，物体が等速運動をしている状態（つまり加速していない状態）での物体にかかる力の合力 F は 0 といえるわけです.[*6]でも，力がかかっていなければ物体は動かないわけですから，合力が 0 でも力は存在しているということになります．つまり，等速運動を行う物体にも力の釣り合いは成立しているわけです．これは水中ロボットの動作を考えるときに重要なポイントになるので，次の章で水中ロボットに影響する力について説明した後にもう一度説明します．

2.1.5　力のモーメント

さて，ここまでは並進を基本に説明をしてきましたが，皆さんもよく知っているように世の中の多くの物体は並進だけでなく回転を行います．物体を回転させようとする力のはたらきをさして，力のモーメントと呼びます．例えば，図 2.11 のように回転軸が固定された円盤に，力 F が掛かっている状態を考えると．力のモーメント N は回転軸から力の作用する方向に垂直に交わる長さを腕 α として，腕と力の積で以下のように示すことができます．

$$N = F\alpha \tag{2.4}$$

てこの原理も力のモーメントで説明ができます．仮に支点と力点，作用点が図 2.12 のように捉えると，支点から力点までの距離を腕としたモーメントと，支点から作用点までの距離を腕として作用点で発する力によるモーメントを見ることができます．なお，力のモーメントは反時計回りを「正」ととります．この力のモーメントにも同じようにつり合いの概念が存在します．例えば，先ほどのテコの原理ですが，作用点側の腕が力点側の腕より短い場合に作用点側の発揮力が大きくなることを説明できるのは，「モーメントが釣り合っている」＝「支点を回転軸とした作用点側の力のモーメントと力点側の力のモーメントが同値である」ということを前提としているからです．今後，力がつりあいの状態にあると表現する場合は，並進と回転の力がすべてつりあっている状態であることとします．

[*6] 多分ここで混乱すると思うけど冷静に考えればわかるはず．合力が 0 というのがポイント．筆者は力がかかっていないとは言ってない．

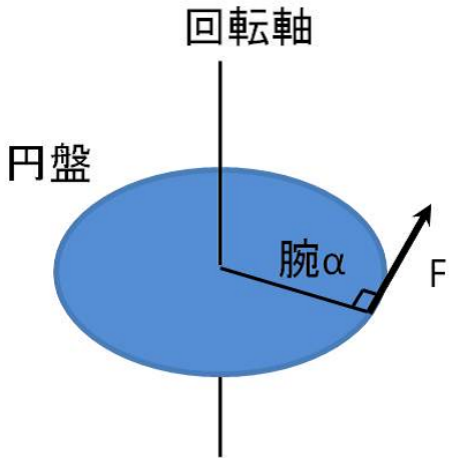


図 2.11　力のモーメント

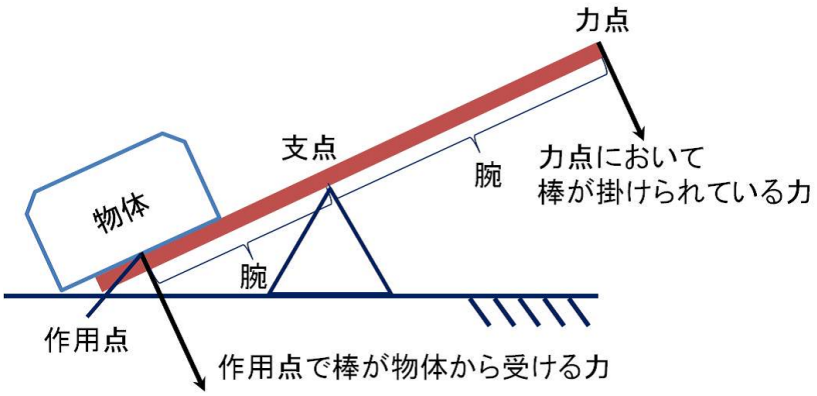


図 2.12　テコの原理

2.2　水中ロボットにはたらく力

水中ロボットが等速で移動している（力がつりあっている状態である）場合，大雑把には図 2.13 のような力がかかります．以下に詳細に説明していきます．

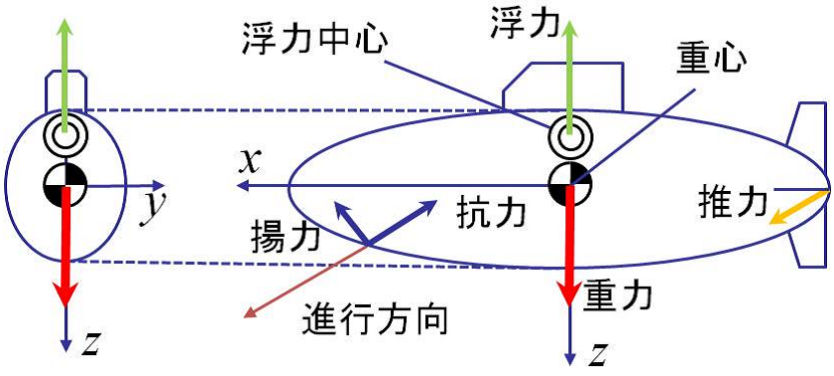


図 2.13　等速直線運動する水中ロボットに働く力

2.2.1　推力

推力とは，移動する物体を進行方向に進めるための力を指します．[*7]水中ロボットの場合，多くはスクリュープロペラを回して推力を得るものが一般的です．今回のキットでも，用いるスラスタはスクリュープロペラとなっています．

2.2.2　重力

地球上において重力という言葉を使う場合，一般には万有引力による地球中心への力と，地球の自転によって発生している遠心力の合力を指します．重力を発生させる加速度は，$9.81\mathrm{m/s^2}$ でほぼ一定です．物体に重力が掛かっていることを表現する場合，この力が作用する点をさして重心 (質量中心) と呼びます．物体を中心に座標系を取る場合は一般的に重心を座標の原点に置きます．(今回も，図 2.13 に示しているように重心を原点にとっています．) 重心は，物体を別々の複数箇所を個別に意図で吊るしたとき，これらの糸の向いている交点に有ります．

2.2.3　浮力

浮力は，水などの流体中にある物体に重力とは逆の方向に作用する力をさします．浮力の原因はアルキメデスの原理で考えると理解しやすいです．例えば，図 2.14 のように物体が水中にある場合，物体は周囲の水から圧力（静水圧）を受けます．このとき圧力は物体の上と下では異なります．(より深いほうが強い圧力を受ける.）この全ての力の合力を考えると，水平方向の力は相殺されるので，縦方向の力の差が発生します．これが浮力です．浮力は水中にある物体の体積に比例して大きくなり，以下のような式で説明できます． [*8]

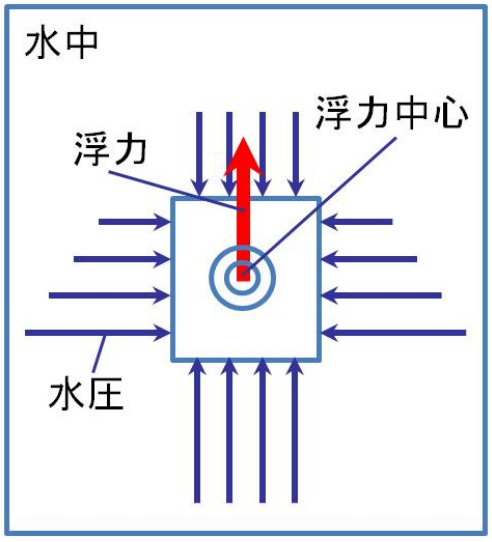


図 2.14　浮力

[*7] 宇宙で戦うタイプのロボットアニメなどでよくスラスター (Thruster) なるものが出ますが，これは推力を指す英単語 Thrust から来たものです．

[*8] 式嫌いな人多いと思います．実際書籍に式が出ているだけで売上が半減するという噂があるくらいで．でも式は読めるようになるとわかりやすいものなのです．

$$Fb = \rho V g \tag{2.5}$$

$[Fb$: 浮力 $[N]$
ρ : 周囲流体の密度（水であればほぼ 1）$[kg/m^3]$
V : 物体の体積 $[m^3]$
g : 重力加速度 $[m/s^2]$

また，浮力の作用点をさして浮力中心（浮心）と呼びます．浮力中心は物体の比重が一様である（つまり全て同じ物体で出来ていると仮定したとき）の重心とおなじになります．[*9][*10]

2.2.4　流体抵抗

流体抵抗とひとくくりに書いていますが，ここでは抗力と揚力について説明していきます．流体中にある物体が周囲の流体と相対速度を持つとき．噛み砕いて言えば，物体が動いていたり，風を受けている状態では物体には抗力と揚力が発生します．揚力という言葉は聞いたことある方はいると思います．飛行機の翼部分で発生して機体を持ち上げる力が揚力です．一方で抗力は，動作に対してその動作を邪魔する力のことを指します．流れの方向に対して平行に邪魔する方向に働く力が抗力で，垂直なものが揚力です．同じ流体であることから，水中でも揚力と抗力が発生します．翼や舵は揚力を利用して動作しており，抗

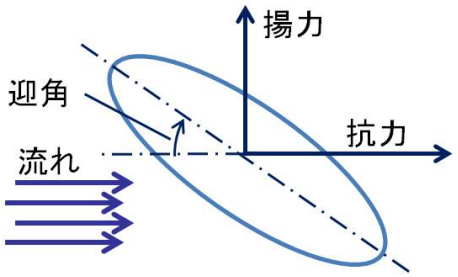


図 2.15　揚力と抗力

力はただただ邪魔な存在です．よって，揚力と抗力の比，揚抗比が大きい形状を活用することが重要といえます．揚力を使わないにしても抗力の影響は非常に大きいことから，素早く動きたければ抗力が小さくなるような形状を選ぶことが大事です．[*11]揚力と抗力は式のような形で求めることができます．

$$\text{揚力 } L = (1/2)\rho V^2 S C_L \tag{2.6}$$

$$\text{抗力 } D = (1/2)\rho V^2 S C_D \tag{2.7}$$

ρ : 周囲流体の密度（水であればほぼ 1）$[kg/m^3]$
V : 周囲の流れ $[m/s]$
S : 代表面積 $[m^2]$
C_L : 揚力係数 $[-]$
C_D : 抗力係数 $[-]$

[*9] 逆に言えば，狙って作らないと浮力中心と重心は一致しないということ．
[*10] つまりは，キットのフレームだけとか浮力材だけとかスラスタだけだったら浮力中心の位置はだいたいわかるということになりますね．
[*11] カバーつけたり，ケーブル隠してみたり...

式を見ると，代表面積と揚力（抗力）係数，速度の 2 乗に比例する事がわかります．この係数は迎え角と呼ばれる，物体と流れの成す角と，レイノルズ数と呼ばれる無次元数，物体の形状によって定まっています．参考に，形状による抗力係数の違いを並べた図 2.16 を示します．もちろん寸法などでかなり変わりますのであくまで参考に．[*12]さて，この式からは，より高速に移動するためには代表面積を最小化して，抗力係数が小さくなる形状を選ぶことが重要でしょう．[*13]これで，進行方向への等速並進運動についてのみ考えると，機体の速度は揚力と抗力，推力のつり合い式を立てやることができれば，その時の速度を求めることができるようになるはずですね．

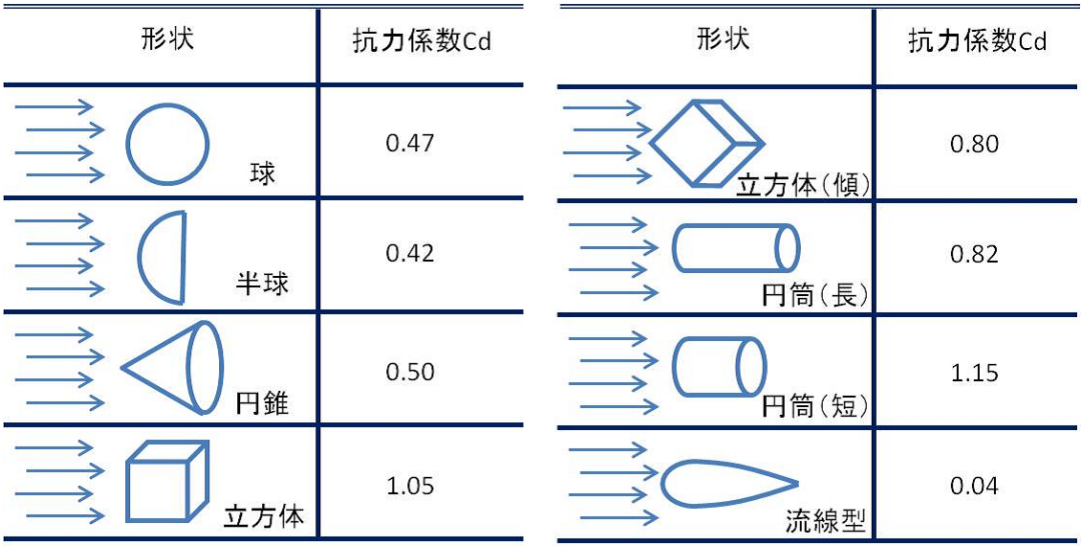


図 2.16　形状による抗力係数の違い

さて，ここで先ほどの等速運動中の力の釣り合いをもう一度考えてみましょう．$Ma = F$ の式において，M は機体の質量，a は加速度とした場合，移動中の機体に関わる力が F ですね？簡単のために，重心と浮力中心が一致していたと仮定して考えると，進行方向 x 軸に影響する力は推力と流体抵抗です．

$Ma = F$　　$F =$ 推力 $-$ 流体抵抗

力の進行方向からはこのように F の式をかけると思います．これを $Ma =$ の形に代入します．そして，等速直線運動ですので，$a = 0$ とします．すると，推力と流体抵抗が釣り合っているという式が成立しますね？よって，流体抵抗が速度の 2 乗に比例することから，流体抵抗の係数と推力がわかれば，機体の速度が計算で求められます．また，どの係数を変更すれば機体を高速化できるかが簡単に検討できるはずです．必ずしも，推力を大きくするだけが速度向上の方法であるとはいえないことがここからわかるはずです．[*14]

2.3　本章のまとめ

この章では，水中ロボットを動かすために必要な力について，説明しました．水中ロボットに関わる力，理解できたでしょうか？次の章では，ここで知った力の考えかたを用いて水中ロボットの種類や動き

[*12] この辺は流体力学という学問分野になります．これだけで 1 つの学問分野なので正直この説明ではかなり端折っています．流体の先生が見たら怒るレベルで．興味がある人は本を読みましょう．

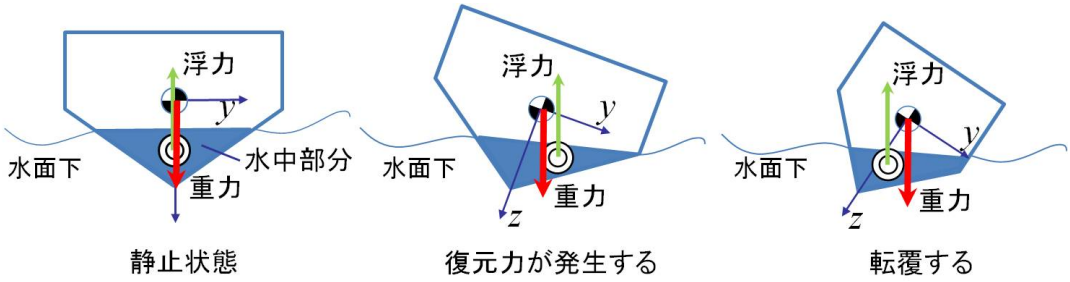
[*13] 陸上でも山手線と新幹線では先端の形状が違うのは，後者が抗力の影響をなるべく削減しようとしているからです．

[*14] 現象を単純図示して (モデル化といいます) 数式化して行くと，カンではなく理論で良い状態を選べるようになるわけです．1 章にあった車輪ロボットの図も，図から式を立てれば，できる条件，出来ない条件を検討できるようになりますね．

について説明していきますので，動きと力の関係についてイメージしながら読み進めて貰えればと思います．また，完成した機体を動作させるときも，動作の裏でどのような力が働いて動いているのかを考えながら調整や改造を進めてもらえればと思います．

船と水中ロボットの安定性

船も水中ロボットも，同じように水と接しているモノですから，浮力と重力の影響を受けます．では，船と水中ロボットの大きな違いは何かと考えると，これは全体が水中にあるかそれとも一部が水中にあるかの違いです．船の場合は，一部のみが水中に有り，その部分だけで浮力が発生します．よって，船は姿勢によって浮力中心の位置が変化するということがわかります．ここで，図のような右に傾いた船の状態を考えてみましょうか．よく似ている中央と右の図ですが，浮力中心の位置が違っていますね．中央は復元できます．なぜかといえば，重心よりも浮力中心が右に有りますから，反時計周りに回転しようとする力のモーメントが浮力によって発生します．一方，右はそうは行きません．浮力中心が重心より左にありますからそのまま船は右回りに回転して転覆してしまいます．よって，船を安定させるためには，傾いたときに船の浮力中心が常に重心よりも外側にいる必要があるわけです．[a]では，水中ロボットの場合はどうでしょうか．水中ロボットは潜っていれば，傾きに対して浮力中心の位置が変化することはありません．よって，基準姿勢の時に浮力中心が重心より上にあれば，傾いた場合も必ず姿勢が元に戻るというわけです．では，浮力中心と重心の位置関係が復帰力にどのような影響を与えるでしょうか．これは，先ほどの力のモーメントの概念で考えることが出来ます．機体が傾いた時のことを考えると，重心と浮力中心の x 軸方向の距離と，浮力がそれぞれ，腕と力になって，力のモーメントを発揮することかがわかります．ということは，仮に浮力の位置を調節できるとするならば，動作していない機体はどういった姿勢で静止するか．そして，重心と浮力中心の距離をどうすれば，機体がより強い復帰力をもつ（傾きにくくなる）かがわかると思います．[b]



船の浮力と重力の位置関係

[a] これを説明する手段としてメタセンタというものが有ります．きょうみあるひとは調べてくださいね！
[b] わからない場合は，力のモーメントの概念を理解しなおして下さい．

第 3 章

水中ロボットの分類について

本章の PDCA は以下のようになっています．

Plan　読者が水中ロボットの分類について理解する
Do　　読者が章を熟読する
Check　水中ロボットの分類について説明できるか確認する
Action　最終的に作りたい水中ロボットをイメージする

一般的にモノは必要であるから作られます．もちろん，水中ロボットもその例にもれず，何かしらの目的があって作られていますし，その目的に適した形に作られています．そこで，本章では水中ロボットの必要性を説明するために，まずモノを作るという行為そのものについて触れた後に，水中探査などの水中ロボットの開発が必要となる事項※とそれに適したロボットの対応がどのような構造であるかを説明して行きたいと思います．

3.1　なぜモノを作るのか？

さて，何故モノを作るのか．本書では，実存と本質という視点からこれについて考えていきたいと思います．これらの言葉は，実存主義哲学という学問分野で主に用いられる言葉です．実存主義哲学などと漢字が 6 文字も続けば読むのがしんどくなると思うので詳細は省きますが，要は「(実際は) どうなっているか」と「(理想/本質は) どうあるべきか」を論ずる考え方と思ってください．前者は「実存」です．いまそこに有る姿はどうなっているのか現実の存在，実情を指します．後者は「本質」です．どうあるべきか．その存在の本来求められる目的や機能を指した言葉です．この世の中に有るモノは大半が本質を達するために存在します．(図 3.1)

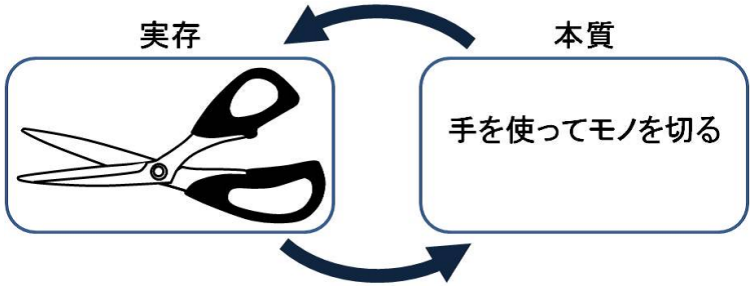


図 3.1　実存と本質

わかりやすいようにハサミを例に取り上げてみましょう．前者であれば，ハサミは「対象物を切断するため」に作られています．これがハサミの本質です．文房具のハサミなら紙を切るために適したハサミ，金属板を切るために金切りバサミ，裁縫用なら糸切りハサミなど，色々な活用目的のために適した形のハサミを作り，活用するわけです．これらの多様な形（専門用語では設計解といいます）がハサミの実存です．

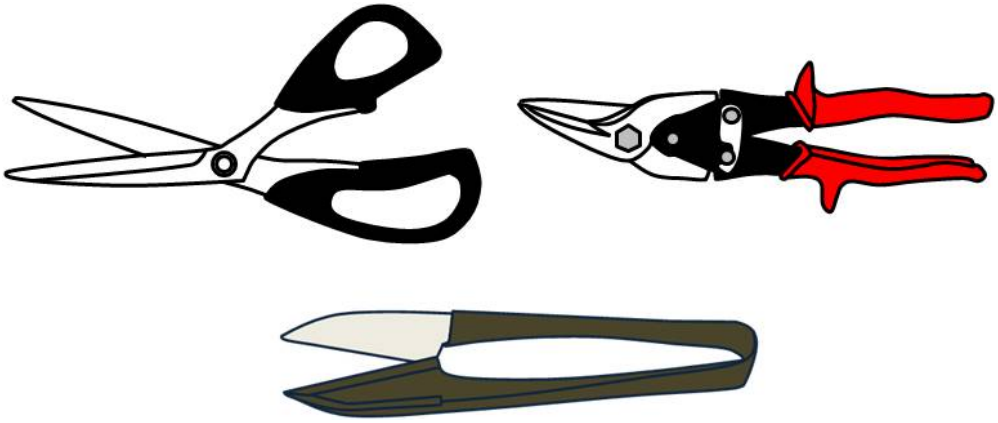


図 3.2　様々なハサミ

さて，よほどの事情がなければ，糸切りバサミで紙を切りたくないし，普通のハサミで金属板を切ろうとはしないでしょう．つまりハサミとは，本質として求められる機能が先立ってそれに応じた実存がそこにあります．「目的達成型のモノ作り」というわけです．いま海で活躍する多くの水中ロボットはこの「目的達成型」に該当します．探査や観察という目的を達成することを本質と捉えて，それを達成する機能を満たすロボットを開発して活用するわけです．

では逆の考え方はできるでしょうか．すなわち，「そこに存在があって，あり方が定義される様な状況があり得るか」ということです．同じようにハサミを例に取るならば，すでにそこにハサミがあって，そのハサミを見ながら「ハサミとは何か？」「ハサミとはどうあるべきか？」「良いハサミとは何か？」「どうすればより良いハサミになるのか？」といったことを考えるわけです．

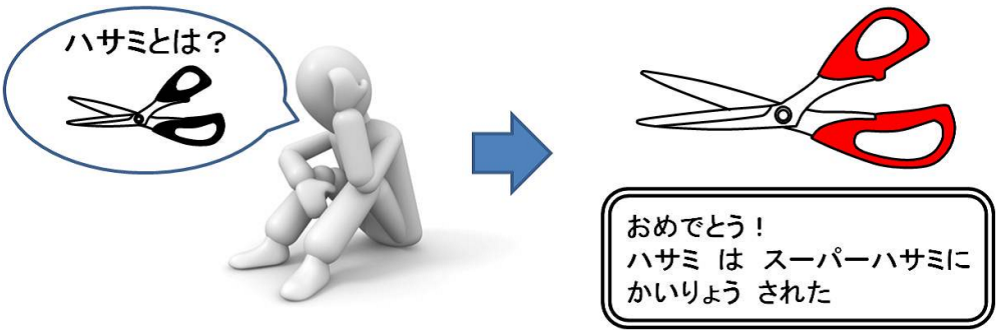


図 3.3　ハサミを研究する

「～とは何か？」「～とはどうあるべきか？」と考えることはより思想的であるとともに根源的な内容であり，「良い状態とはどんなものか？」「どうすればより良くなるか？」と考えることは学術的仮説の確認と実証であってより具体的かつ前者に比して実用に近い内容です．実例を挙げるならば，前者はヒューマノイドの開発があります．ヒトは生まれた時に何かをするために生まれるわけではないのですから，実

存が先にあってその後に本質が続くわけです．だから，ヒトがヒトを作ることによって，ヒトが「ヒトとは何か？」「ヒトのもつ機能は何か」を考える契機とするわけです．

図 3.4　ヒトの本質を探る

実際に開発の歴史を紐解けば，「二足歩行における上半身の存在意義」の様に自身がヒトだから知っていたはずなのに，実際にはわかっていなかった「新たに発見」された知見も存在します．[*1]後者は，より一般的な研究になります．「どんな状態がよいか」もしくは「どうすればより良いものになるのか」を元に仮説を立てて，これを確認します．改善，改良といった行為はこれに含まれます．[*2]「目標達成型」と対比するならば「研究開発型」とでも言いましょうか．分類の図を以下に示します．なお，目標達成型と研究開発型は言い換えれば，実用と研究でありこれらは相互に情報を交換することで，よりモノの質を高めることのできる関係にあります．

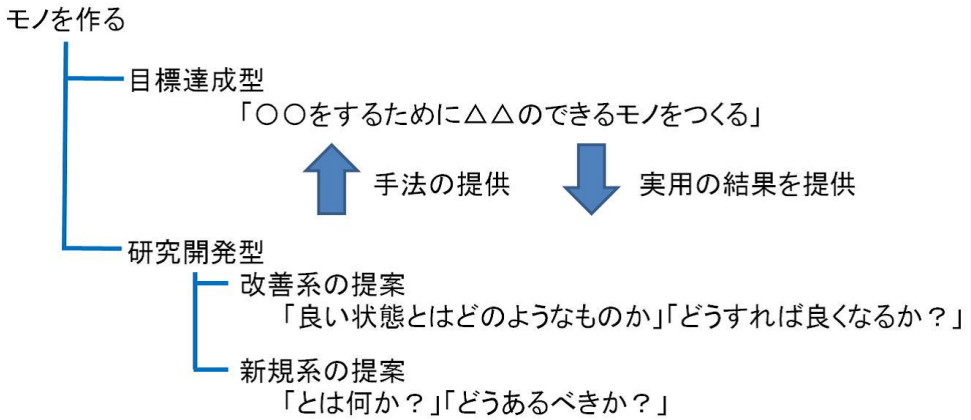


図 3.5　モノをつくる理由の分類

3.2　水中ロボットの分類

さて，前節ではモノを作る意味について考えてきました．「目標達成型」と「研究開発型」の違いが理解できましたか．先に述べたとおり，水中ロボットの大半はこちらの目標達成型に分類されます．目標達

*1 筆者はア○モとかガ○ダムとかにあこがれてこの世界に来たので，もともとはこっちの研究をしようとしていました．
*2 この差は 0 を 1 にする行為と 1 を更に大きな数にする行為の違いです．

成型であるということは，水中ロボットには作られるにあたって「目標」があるということになります．そこで，今回の機体の分類は「目標・移動の範囲」と「移動形態」の対応をとって分類しています．

3.2.1　目標達成型ロボットの分類

水中ロボットは「ヒトが行う事ができない，もしくは行った場合非常に危険かつ非効率な作業」を代わりにするために作られました．まず，「行う事ができない，もしくは行った場合危険な作業」について考えていきましょう．

ヒトが水中で作業をしようとした場合，皆さんがテレビでよく見るようなダイビングの装備や，図 3.6 に示すような鎧のような常圧潜水服と呼ばれる服を着る必要があります．当然服は重いですし，流される危険や取り残される危険だってないわけではありません．当然潜れる最大深度にも限界がありますから，ヒトが行えない作業は水中には膨大に存在します．

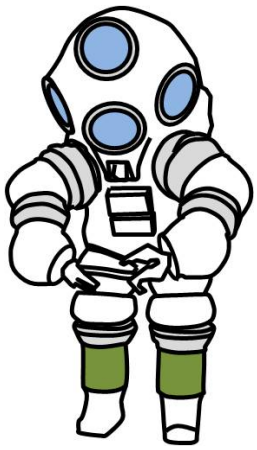


図 3.6　常圧潜水服

そこで，ロボットを用いることでより安全に作業を代行させるわけです．次に，「非効率」ですが，例えば海底資源を探査することを例に挙げて考えてみましょう．流石に生身のヒトに海底を歩かせるわけにいかないので，潜水艇を使うでしょう．しかし，潜水艇といえど，軍事用でもなければそれほどの大きさもなく，運用するためには母船と呼ばれる船が同時に必要になります．しかも，一度の潜行で調査できる範囲はとても広いというわけではないですから，費用対効果[*3]はそれほど良いわけではありません．そこで，ロボットを用いることを考えます．例えば，広範囲を調べるためにロボットを放ち，範囲を探索させて回収します．自動で探索をしてくれれば，その間船のほうは，潜水艇を動かすよりはよほど自由な状況が維持できるうえ，何よりロボットは人と違って疲れることがないわけですから，長時間水中にいてくれるわけで，一度の調査活動で得られる効率もあがることが期待できるわけです．

さて，以上のような理由から水中ロボットが活用されているわけですが，この場合「作業」はどういったものになるでしょうか．この作業というものは，いわば目標を達成するために行う行為であり，基本的には「とる」と「みる」の 2 つの作業に大別されます．

「とる」とは，水中にあるものを，つかむ，触れる，回収するといった，ヒトでいう手を用いる作業です．「とる」作業を行うためには，ヒトの制御を受けて，ある程度詳細かつ高速にロボットを動作させる必要があります．しかし，水中では電波が届かないことから，こういった詳細な動作をさせるためには有

[*3] 最近はやりのコスパってやつ

線で遠隔操作せざるを得ないため，必然的にこれらの機体は移動範囲が狭くなりがちです．

一方で「みる」とは，水中にあるものを確認する作業です．海底資源の分布から，海底ケーブルやダムの壁面状況など見る対象はかなり多彩です．海底資源の調査といった作業の場合は広範囲を調べる必要があります．また，この場合は詳細な制御よりも自由に移動できることが優先されることから，自動制御で動作しています．ただし，ダムの壁面調査などの場合は，任意の狭い範囲を詳細に見たい場合がありますからむしろ取る作業に似た特性が必要になる場合があります．そこで，さきの「目標・移動の範囲」の組み合わせにあわせて，「広範囲・みる」「中範囲・みる」「狭範囲・みる/とる」「狭範囲・とる」の大きく 4 つに分類して説明をしていきます．

3.2.2　グライダ型水中ロボット（広範囲・みる）

広範囲を見て回る調査作業，特に特にグライダ型の調査作業は数ヶ月単位に及びます．そのため，通信・給電用のケーブルが邪魔になり，自律で広範囲を移動することが求められます．ケーブルを繋いでいないわけですから，水中ロボットは自身に搭載されたエネルギー源，例えばバッテリで駆動を行います．言うまでもなくバッテリの容量は有限ですからなるべく少ないエネルギーで浮上・潜航と移動を行う必要があります．そこで，グライダ型水中ロボットは図 3.7 のようなバラストタンクと翼を利用する機体構造となっています．

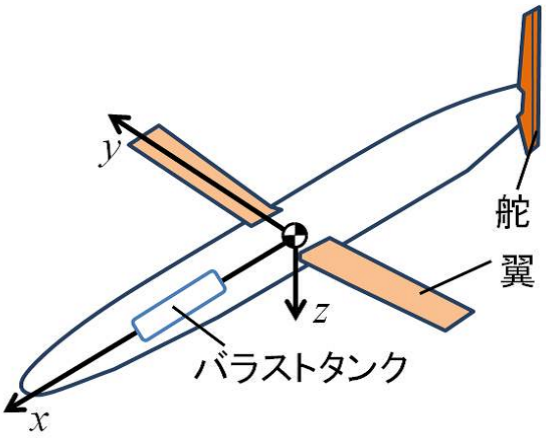


図 3.7　グライダ型水中ロボット

バラストタンクは，機体の重心位置と浮力の大きさを変更できます．*4これによって，図 3.8 のように重心位置を前後することで機体の姿勢を変更することができます．*5そして，浮力と重力の総量によって，潜る状態と浮く状態を作り出し，潜りながらな（ないしは浮きながら）翼の揚力を利用することによって，図 3.9 のように水中を滑空することで移動します．*6まさに水中のグライダというわけです．動作自由度としては，x 軸方向への並進と，ピッチ，ヨー回転が可能なものが一般的です．

*4 水中を滑空ってなんかおかしいけど表現が他に思いつかないんですよね．
*5 わからなかったら 2 章力の釣り合いを見るんだ！
*6 業界ルールとして，潜るは能動的に行う行為（つまり潜航），沈むは受動的に行う行為（つまり沈没）とされています．間違えるとかっこ悪いよ！

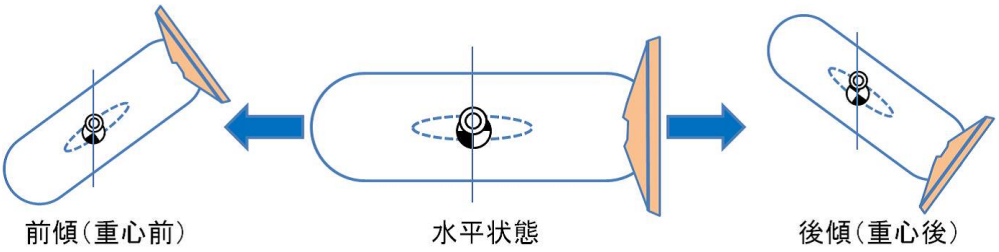


図 3.8　重心位置による姿勢変化

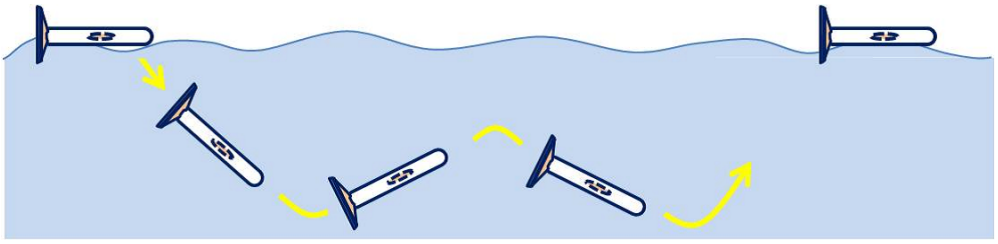


図 3.9　グライダ型水中ロボットの挙動

3.2.3　航行型水中ロボット（中範囲・みる）

グライダよりも狭い範囲，でも，下の狭い範囲より広い範囲をくまなく調査したいといった時には，航行型のロボットが用いられます．例えば，広範囲な環境を探索して海底の状況を分析するために用いられると行った用途で開発されました．そこで，広範囲をグライダ型よりも能動的に移動するために，例えば図 3.10 に示すような機体構造が見られます．また，機体の重力と浮力の大きさと作用点をほぼ一致させることによって，浮上潜航やピッチ回転をより少ない力で行える設計になっています．[*7]このような構造を取ることによって，図 3.11 のように海底面をはうように移動します．

以上から，航行型水中ロボットは航空機に喩えるならば，いわば飛行機の様な挙動をするものと思って下さい．動作自由度としては，例えば x 軸方向移動，ピッチ回転，ヨー回転が行えます．ちなみに，本キットは初期でこの，航行型のスラスタ配置になっています．

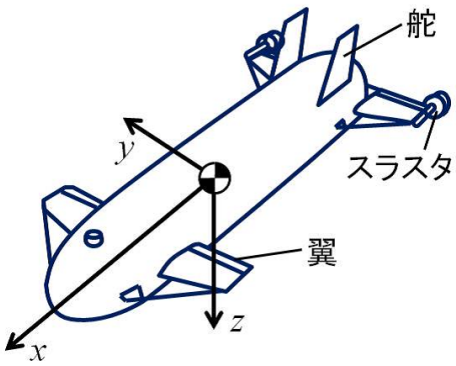


図 3.10　航行型水中ロボット

[*7] 一致させれば，潜るのに力はいらないし，旋回動作とかでも流体抵抗だけ考えれば済むものね．

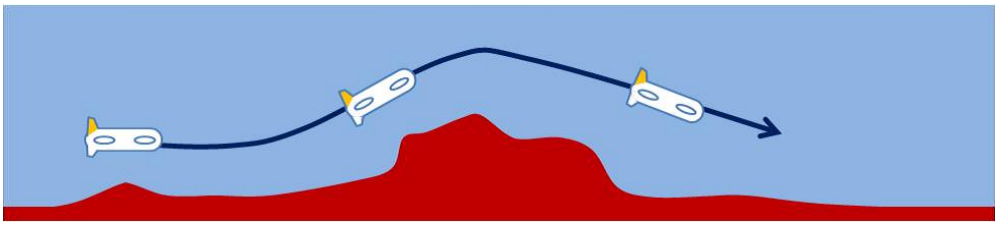


図 3.11　航行型水中ロボットの挙動

3.2.4　ホバリング型水中ロボット（狭範囲・みる/とる）

いままでの例と比べた時に，ホバリング型の水中ロボットはより狭い範囲で用いられることが多いです．そして，狭い範囲を移動するかわりに詳細な移動が行いやすい仕組みになっています．具体的には，図 3.12 のように垂直方向にダイブスラスタと呼ばれるスラスタを配置することで，x-y 平面（水平面）の移動と z 軸方向（深度方向）の移動を切り分けて行うことができます．

浮力と重力は，浮力が若干勝るかほぼ同じであり，作用点の距離は離れていることが一般的です．これによって，潜る動作にあまり力を使わないようにすることでエネルギー消費を抑え，作用点距離で機体姿勢の安定化を図ります．ちなみに，図 3.12 に示したものは最もシンプルなホバリング型のスラスタ配置になりますが，他にも変則的な配置が有ります．

この配置であれば，潜航方向は独立で駆動でき，更にヨー回転と x 軸方向の移動が可能な構造になりますよね．例えば，図 3.13 のようにダイブスラスタに角度を付けて配置することによって，y 軸方向の移動機能を無理矢理に持たせようとするもの．[*8]他にも，図 3.14 のように斜めにスラスタを取り付けたり，スラスタ配置を工夫することによって，望んだ移動自由度を確保する事が可能になります．このホバリング型水中ロボットは航空機でたとえるならば，若干変則的では有りますがヘリコプターであるといえるでしょう．

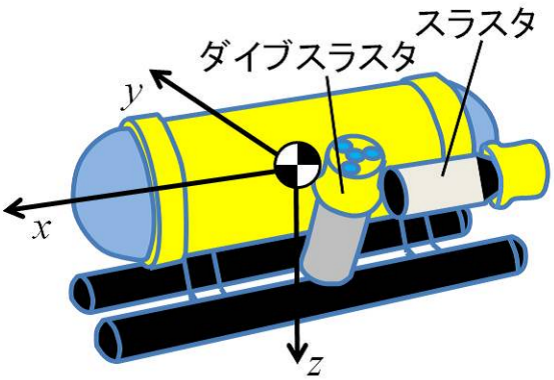


図 3.12　ホバリング型水中ロボット

[*8] 重心を軸にとって，浮力とダイブスラスタの推力についてモーメントの釣り合いを考えれば，この構造ではロール回転を行いつつ y 軸方向に移動してしまうのは自明ですよね？

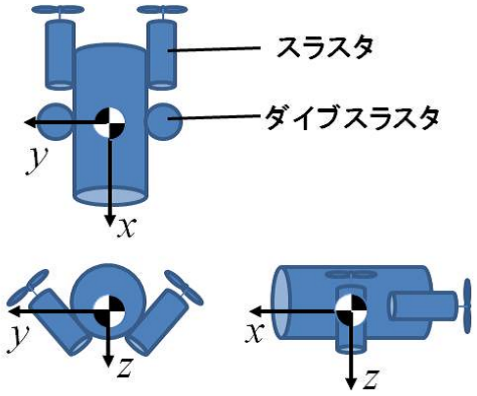


図 3.13　z 軸移動と y 軸移動の連成

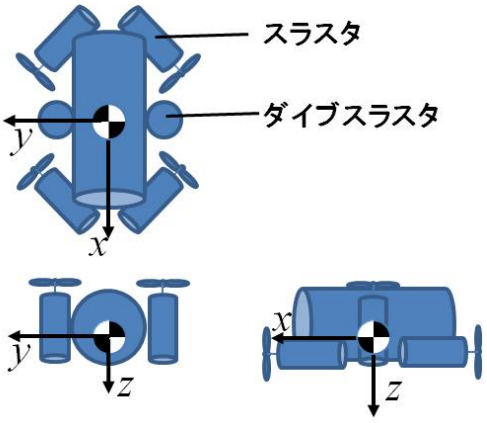


図 3.14　全方向に移動可能なスラスタ配置

3.2.5　建設機械型水中ロボット（狭範囲・とる）

今まで挙げられている水中ロボットは，水中を浮遊して移動するものでしたが，建設機械型の水中ロボットは地上の建設機械のように海底などの面に接して移動を行うものになります．多くの機体が図 3.15 のように下部にクローラを配した様な構造となっています．また，機体によってはホバリング型の水中ロボットと同じようにスラスタを配置しているものも有ります．これは，水中を移動する場合は，浮いて移動してしまったほうが楽だからです．その理由としては，陸上と違って水中では浮力がはたらくことで海底面との十分な摩擦力が取れなくなるためです．これを解決するために，壁面や船底に張り付いて移動するタイプの水中ロボットは，ダイブスラスタによって機体をその面に押し付けています．特殊な事例として，脚によって移動するタイプのものも存在します．

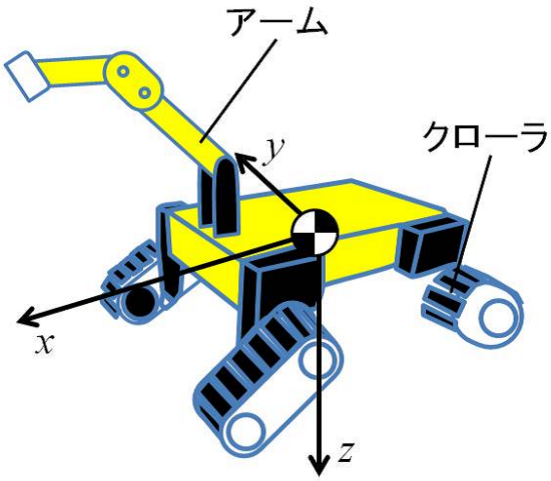


図 3.15　建設機械型水中ロボット

3.2.6　研究開発型ロボットの分類

今までの実用のためのロボットに対して，研究開発型のロボットは新しい仕組みの検証を行うために作られるものが多く，個性的な形状をしているものが多いです．スラスタの位置や角度が変わるものや，機体の形状が変わるもの，人のような形状をしたものなど本当に多くの形状があります．[*9]例えば，エイや魚を模すことで，生物の機能を模倣しようとするバイオミメティック系のロボット等が挙げられます．図 3.16 に示しているのは，イカの泳ぎを模して移動可能な水中ロボットになります．こういった羽ばたきによる推進はプロペラのように回転運動ではないために障害物の多い環境で動かしても絡んでしまうことが無く有利であるという主張があります．

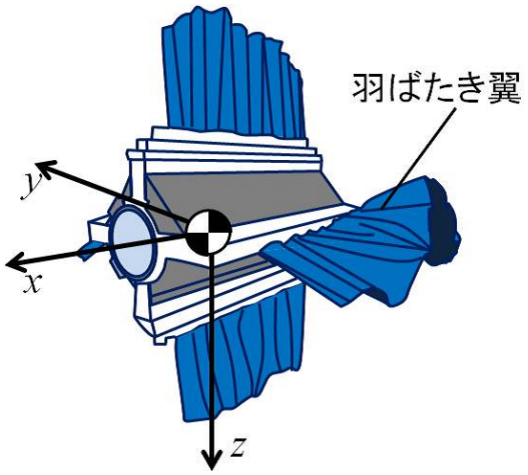


図 3.16　波動水新型生物模倣水中ロボット

3.3　本章のまとめ

大雑把にですが，研究と開発の違い，水中ロボットの成り立ちと種類について説明してきました．なんとなく内容をつかむことができましたか．このマニュアルを読んでいる皆さんの場合は，多くが開発ベースのロボット製作になると思いますので，「何のために作るのか」「そのためにはどのような機能を達成するべきなのか」に意識をおいて開発をすることが重要であると思います．そして，達成するべき機能が決まれば次に持たせるべき機体の構造が決められるはずです．この辺の話については，もっと先の章で説明していきます．さて，次の章からいよいよ製作に入ります！

[*9] あまりにも種類が多すぎるので今回は話をだいぶ端折っています．Google の画像検索で”水中ロボット”とか”ROV”とか”AUV”とか”Underwater Robot”とかで検索して貰えればたくさん見られるのでぜひ試してみてください．

第 4 章

ハードの作成

本章の PDCA は以下のようになっています．

Plan　読者が今回製作する水中ロボットの構造を理解する
Do　　読者が実際にハードウェアを製作する
Check　実際に水密，動作が可能か確認を行う
Action　次章電気回路の製作に以降する

4.1　製作の前に

おまたせしました．いよいよ機体の製作を開始します．しかし，何事も事前の準備なくして成功はなく，何も考えずに走りだせば必ず不幸を呼ぶことになります．計画と準備こそが，物事を完遂するためにもっとも重要なことであり，これを無視して得られる成功はまぐれに過ぎません．[*1]ということで，この章では製作に入る前に，これから作るキットの構造について説明し，その後で製作を開始します．

4.2　機体の構造

このマニュアルで製作方法を説明していくのは，図 4.1 に示す水中ロボット Mark3-TM です．水中ロボット本体は，耐圧殻と，スラスタ，メインフレームによって構成されています．耐圧殻は水圧に耐えて，電子部品やバッテリを水に触れされないために用いられます．必要とされる機能は，耐水・耐圧であることと，スラスタや作業用アームなどに動力や指令を送るためのケーブルが引き出せる必要があります．

次にスラスタですが，これは言うまでもなく推力を得るために用いられます．スクリュープロペラとそれを回転させるためのモータによって構成されます．また，モータはそのままでは使えないため，防水処置が可能な仕組みになっています．

そして，最後にこれらを固定するためのフレームです．Mark3 では，アクリル板によってこれを構成しています．また，繰り返しですが機体は浮力や重力の影響を受けて姿勢が決定します．そこで，錘や浮力材，スラスタ，耐圧殻の位置を自由に変更できるようにするために面ファスナを用いて固定することが可能です．

[*1] 彼を知りて己を知れば百戦して危うからず，彼を知らずして己を知らば勝敗分かれ，彼を知らず己を知らざれば戦うごとに必ず敗れる（孫子）

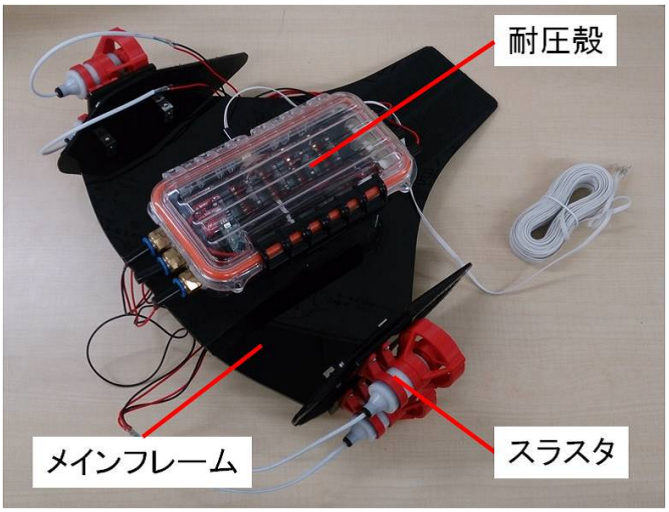


図 4.1　Mark3 概観図

4.3　耐圧殻を製作する

4.3.1　耐圧殻の構造

耐圧殻はロボットの脳みそや神経系といえるような部品を導入する場所であるため，その防水性には気を使う必要があります．一般には，圧力に耐えるために円筒形の耐圧殻を用い，その端面を O リングと呼ばれるゴムリングで抑える事によって防水しますが，円筒形の部品を加工するには旋盤などの工作機械を用いる必要があります．普通科の学校ですと，旋盤を持っていない学校も多いかと思いますので，このキットでは追加工などが簡単に行えるよう，専用の耐圧殻を同梱しています．

図 4.2　西川精機製作所製防水ボックス

さて，耐圧殻そのものが防水性を備えていても，スラスタやアームのように外部にとりつける部品には配線を繋げる必要があります．簡単につくるためには，穴を明けてケーブルを通した後に接着などで固定をしてしまうといった方法も有りますが，それでは後から位置やケーブルの数を変更できないので面白くはありません．そこで，Mark3 の耐圧殻では，ケーブルの出口にケーブルグラントという部品をもちいて防水しています．今回用いている製品は AVC 社の長足型ケーブルグラント (MGB12L-05B) です．同様の規格で他社でも作っておりますので代用が可能です．

図 4.3　ケーブルグラント

4.3.2　耐圧殻の加工

耐圧殻の加工について説明します．今後みなさんが全く新しく機体を作りたいといった場合に 1 からつくれるように全ての作り方を説明していきますが，今回は時間短縮のために一部の加工（穴あけ，タップ，ねじ込み，接着）を事前に加工しています．皆さんが行うのはケーブルグラントの固定との防水テストになります．ケーブルグラントを固定するための穴を耐圧殻に開けていきます．専門道具の定盤やハイトゲージをつかって位置を決めてから，ボール盤を使って精度よく開けていくことが望ましく，おすすめです．しかし上記の様な工具がない場合は，図のように穴位置がわかる紙を貼り付け，ドリルで穴を開ける方法をとる事もできます．

図 4.4　穴あけ

ドリルの穴は，ケーブルグラントに最適な径で開ける必要があります．今回のグラントの最適系は直径 12.5mm です．できあがったものに，ケーブルグラントを固定します．防水ボックス両面にグラントのネジ部とナットを当てて，両方にレンチを噛ませて締め上げて固定します．このとき，締め上げ時に負荷を感じてから 90 度程度回すのが適正とされています．ゴムワッシャはキットの外側に来るようにして下さい．

4.3.3　耐圧殻の防水

これで表面上はフタをすれば防水されそうですが，実際はこれではまだ防水は不十分で浸水します．防水ボックスは防水性が有ると言っても，深い水深まで持ったまま潜ることを想定していないので，穴を埋める追加工を施して防水性を高める必要が有ります．この防水ボックスの場合は，留め具の部分に気圧を

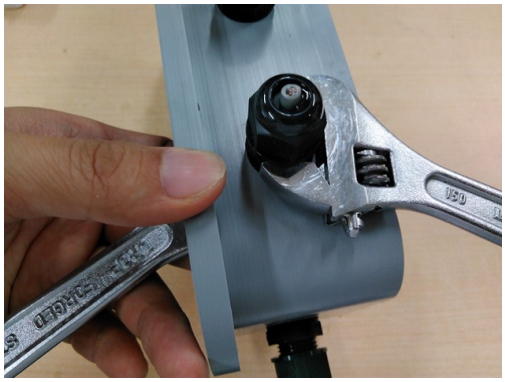


図 4.5　レンチによる締め上げ

図 4.6　防水ボックスの完成

一定にするための穴が空いているので，これを接着剤でうめてしまいましょう．私は，瞬間接着剤と硬化剤を用いて行いましたが，次以降の作業で用いるような二液混合のエポキシ接着剤で行っても問題ありません．

4.3.4　耐圧殻の防水テスト

防水準備ができたら，防水テストをしてみましょう．写真のようにケーブルグラントにモータ用ケーブルやテザーケーブルを奥まで差し込みます．テザーケーブルは細い部分をグラントに 10〜15mm 程度通して締め上げれば防水できます．ケーブルグラントにケーブルを通して締め上げ，防水します．この時，グラントは工具を用いて締め付け，負荷を感じてから大体 90 度程度の回転を目安にします．

全てが差し込み終わったら，ゴムパッキンと，フタのパッキンが当たる部分，ボックスのフランジ部分をきれいに拭いた後，シリコングリスをぬります．それが終わったら，フタをします．このとき，ネジにはワッシャを通してからネジを対角締めで締めていきます．[*2] 締め量は，フタとボックスに隙間がなくなるくらいです．締めすぎないように注意しましょう．水の中に数分間沈めてみましょう．水が入らなかったら OK です．水が入ってしまったらどこから入っているのかをよく観察し，入ってくる場所がグラントを締め直し，フタであればシリコングリスを更にぬって試しましょう．水が入らなくなれば，耐圧殻は完成です．

[*2] 対角締めの方法：http://www.jikuryoku.net/tejyun.html

図 4.7　ケーブルの接続

図 4.8　防水テスト

ここが難しいところです，どうしてもうまくいかなかったらフォーラムで質問してくださいね．それが終わったら，裏面に面ファスナを貼って完成です．面ファスナは柔らかい面（以降メスと表記）を貼って下さい．

この後，ケーブルにハンダなどの作業をするために取り外す必要が出てくると思います．グラントの固定部分をレンチで抑えて，上側を回せば取り外すことができます．このとき，ケーブルを締めている部品と一緒にグラントの根本を緩めないように注意して下さい．

4.4　メインフレームを製作する

4.4.1　フレームの切り出し

メインフレームはアクリル板とナットブロック，面テープで構成されます．アクリル板は，レーザーカットの後にダンボール板に固定して郵送しています．保護紙は剥がして利用します．また，加工した板には若干バリなどが有ります．注意しつつヤスリなどでバリを落としましょう．

図 4.9　フレームに用いる部品

4.4.2　面ファスナの貼り付け

メインフレームの中央部に縦に 2 本，写真のようにフレーム中央左右に面ファスナを貼ります．なお，フレーム部で用いる面ファスナは全て面が硬いもの（以降オス）を使います．次に左右フレーム部にも同様に面ファスナを貼っていきます．フレームを組み立てた時に外側に来る面に面ファスナを貼ります．フレームに空いた穴の位置に注意しつつ図に青で示した辺にそって貼っていけば OK です．全て合わせた状態がこの写真です．[*3]

図 4.10　面ファスナの貼り付け

[*3] 水の中で使っていると剥がれてくる場合がるので，その場合は接着剤などで面ファスナを固定してしまいましょう．

4.4.3　組み上げ

ナットブロックで左右フレームと中央部をネジ止めしていきます．ネジにはワッシャを通してから，ネジ止めすることで固定を安定させます．以上で，フレーム部は完成です．

図 4.11　組み上げ

4.5　スラスタを製作する

スラスタ部は，機体の速度を決める重要な部品です．作業としては大きく分けて，モータの水密とプロペラの取り付けに分かれます．

4.5.1　灯油ポンプの分解

灯油ポンプは工進社の EP-105 という機種を使っています．ちなみに，同社の灯油ポンプは中の防水部分はすべて同じです．写真で示す位置に，カッターやドライバなどを差し込み，回転すると簡単に分解することができます．

図 4.12　灯油ポンプの分解

内部のケーブルを切断し，先の方から引き出せば分解は終了です．なお，キットにある灯油ポンプはすでに分解済みです．

4.5.2　モータの交換

灯油ポンプに入っているモータは 280 という規格の大きさですが，対応する電圧が 3 V と低く，出力が稼ぎにくいのでモータを交換します．キット内にあるモータは RC-280RA という種類で，最大 9 V まで対応しています．さて，交換のために防水ケースを分解します．写真の様に，ケースを持ちながらペンチ等で抑えて首を折るように回すと上蓋が外れるので，モータを引き出します．

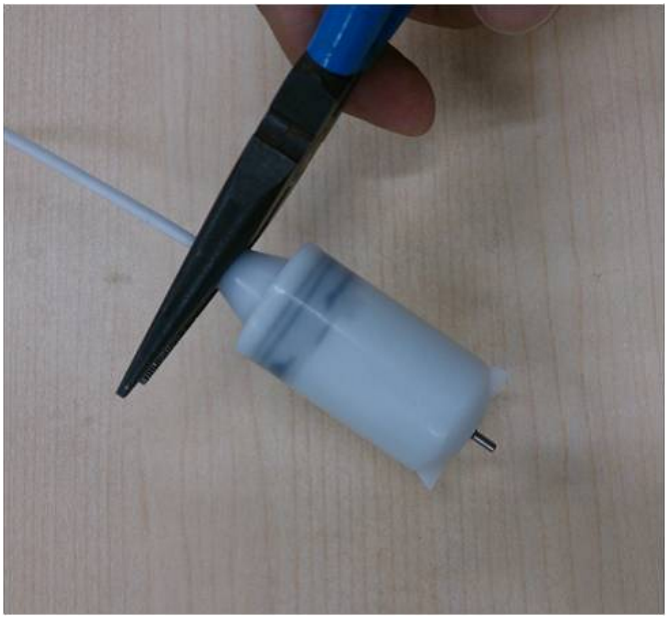


図 4.13　ケーブルの取り外し

端子部分のハンダを外し，新しいモータと付け替えます．モータの蓋部には＋の刻印があるので，それと青の配線を合わせて配線しましょう．次に，配線の反対側は皮膜を剥がします．そして，先ほど防水テストで使ったモータ用ケーブルのゴム皮膜を剥がして，中から白と黒のケーブルを露出させます．ポンプの青とケーブルの白，ポンプの白とケーブルの黒をそれぞれ結線してハンダし，熱収縮チューブをかぶせて絶縁します．（写真ではモータ接続用のケーブルが赤黒ですが，このキットでは白黒のケーブルになっています．）熱収縮チューブはかぶせた後に加熱すると縮み，しっかりとフィットしてくれます．モータケース側のケーブルは樹脂で覆われているので，一度剥がす必要があります．ケーブルを正面から見て，配線と配線の間にニッパーを入れていけば，比較的簡単にケーブルを引き出すことが出来ます．モータ側のケーブルには余裕を持っているので，練習がてらやってみてください．なお，配線のためであれば樹脂を剥がす長さは 30mm 程度で十分です．この後，テストに用いるので，モータ用ケーブルの短い側も 5mm ほど被覆を剥がしておきましょう．

4.5.3　水密作業

水が入ってくる箇所にシリコングリスを塗りつけて水密処置を行います．写真のように，ケースの軸受け部分と，モータのフタ部分にしっかりとシリコングリスを塗布して下さい．また，塗りつけるときにはモータの内部には入れないように気をつけましょう．ここでモータ内に大量のグリスが入ると抵抗が増えてスラスタの出力が落ちてしまいます．

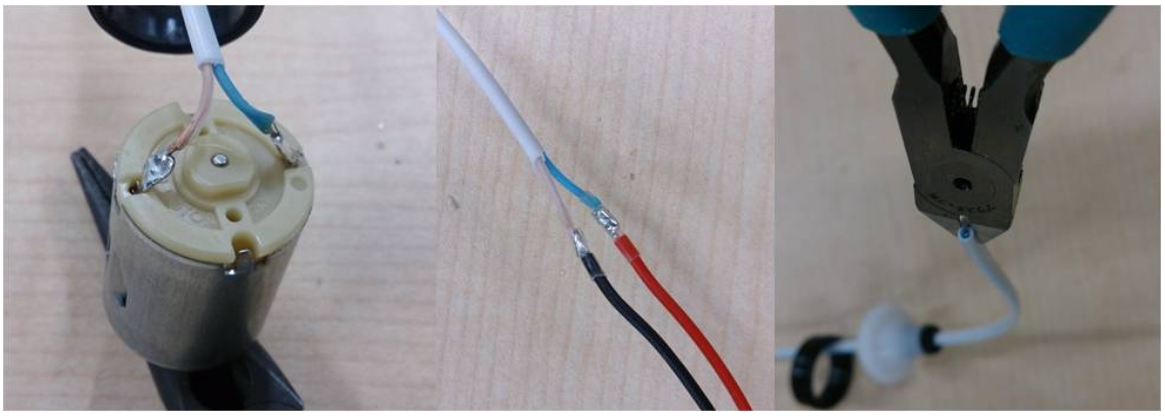


図 4.14　モータとケーブルの接続

図 4.15　グリスの塗布

4.5.4　プリンタパーツの準備

プリンタパーツのそれぞれの名称は写真 4.16 のようになっています．タミヤコネクタについてはここでは使いませんので，無くさないようにしまっておいて下さい．プリンタパーツにはサポートと呼ばれる固定用の不要な部分が存在しています．ニッパやペンチ等で部品をおらないように丁寧に剥がしていきます．くれぐれも部品を壊さないように注意して外しましょう．プロペラは図 4.17 の位置でサポートを外します．

図 4.16　3D プリンタパーツ

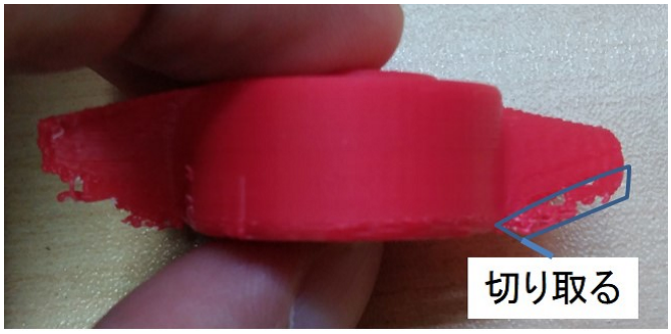


図 4.17　プロペラのサポート

4.5.5　各部品の組み立て

全てのパーツが揃ったら組み立てを行います．スタンドを固定板にネジで固定します．スタンドにはばりがあるので，ヤスリで取り除いて下さい．また，スタンドのネジ穴は小さい場合が有るので，その時はドリルなどで拡張して下さい．スタンドの固定板にはネジ穴が切られていないので，タップを使ってねじ山を切って下さい．図 4.18 のように万力などで固定板を固定して，付属のタップを回し入れてネジ山を切ります．このときにタップが板に対して垂直であるかどうか常に注意しましょう．固定板の裏には，メ

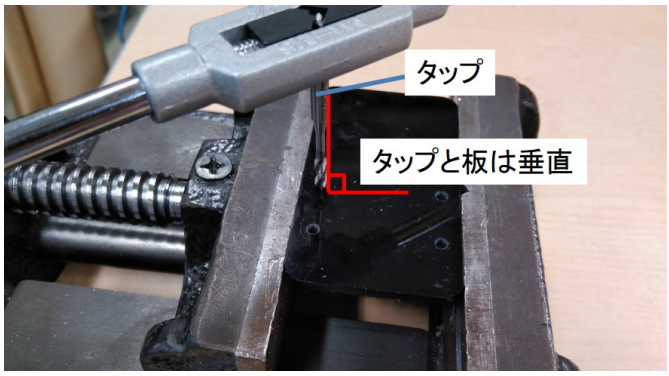


図 4.18　タップを切る

スの面ファスナを貼っておきましょう．次に，プロペラをモータの軸に差し込みます．プロペラは面に出っ張りが有る方が表で，モータの方に向きます．軸を差し込むときは垂直に入れるよう注意しましょう．軸のズレは推力を大きく低下させます．プロペラが抜けやすい場合は，瞬間接着剤を穴に流し込んで

から差し込んで固定します．（接着後は抜けなくなるので注意して下さい）あとは，プロペラカバーとスタンドを順にモータケースのケーブル側から入れていけば OK です．プロペラがペラカバーと接触しないように注意して位置を決めましょう．これらのパーツがどうしても入らない場合は，紙やすりなどで少し内面を削ってしまいます．ただし，スカスカになってしまわないように気をつけて行いましょう．

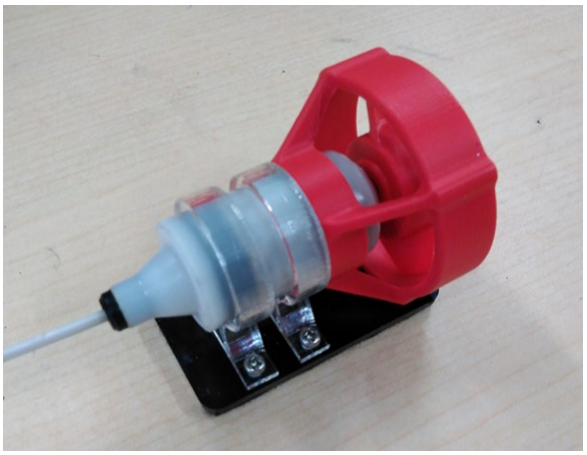


図 4.19　スラスタの組み上げ

4.5.6　スラスタの動作テスト

組み上がったら，水の中に入れて，バッテリとモータを接続してスラスタを動作させてみましょう．しっかりと動けば完成です！

4.6　まとめ

すべての部品を無事に製作できたでしょうか？無事にモータが回り，フレームが完成したら，仮組みをしてみましょう．スラスタと，耐圧殻をフレームに乗せたら，もうすでに機体完成写真とそっくりの機体が出来上がっているはずです．

第 5 章

電子回路の作成

本章の PDCA は以下のようになっています．

Plan　読者が水中ロボットの電装系の構造を理解しつつ，基板の作成を行う．
Do　　読者が章を読みつつ実際にハンダ作業を行う．
Check　それぞれの回路で接続に問題ががないか確認し，マイコンに書き込みを行う．
Action　次章で PC との接続を行う．

ここでは，水中ロボットの頭脳や神経系を担当する電子部品の作成を行います．作業は大きく分けて 4 つです．モータードライバ，マイコンボード，USB シリアル変換ユニットのハンダ付け，そしてケーブルユニットの端子圧着です．

5.1　モータドライバの作成

5.1.1　モータドライバの構造

このキットで用いられるモータは直流モータです．皆さんご存知だと思いますが，直流モータは電流を流す方向で回転方向が決まります．さて，このキットではスクリューを回すわけですから，モータの回転方向を変えられなければバックが出来ません．では，どうすればモータの回転方向を変えることができるでしょうか．例えば，図 5.1 の様にプラスとマイナスの電位を用意して，2 つのスイッチを用意すればモータは正逆転を行うことができますね．

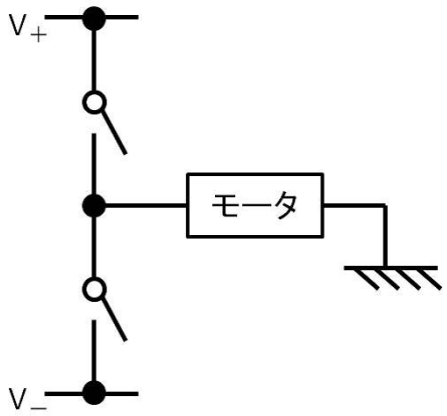


図 5.1　T ブリッジ

これを T ブリッジといいます．動かすモータが多ければ，スイッチの数が少なくて済む分製作が楽だという利点がありますが，マイナス電位を扱うという不利な点が存在します．そこで，H ブリッジという構造が用いられます．図 5.2 のようにモータを挟むように 4 つのスイッチが付けられた状態です．例えばこれで，A と D をオン，B と C をオフにすれば A からモータを経由して D からグラウンドと電流が流れ，モータが回転します．一方で，このスイッチの関係を逆にすれば，C からモータを経由，B からグラウンドに電流が流れます．この時，モータに流れる電流の方向は逆になりますね．以上が H ブリッジと呼ばれるモータ回転回路の原理になります．モータの制御回路はマイナスを扱うことに弱いので，このような回路構造を用いられることが多いです．

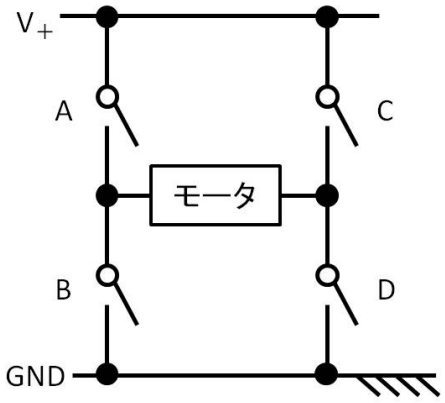


図 5.2　H ブリッジ

この H ブリッジですが，しっかりと設計するにはそれなりの電気回路の知識が必要になります．また，壊れてしまった時にどこが壊れてしまったかを調べるのも大変ですし，何より動かすのに 4 本の信号線を使うのは大変ですので，今回はこれが内蔵された IC の TA8428K を用いています．TA8428K はモータ電源と，入力信号を入れることによってモータの回転方向と速度を変化できるようになっています．例えば，下の表 5.3 の様に入力を変化させることによってモータの回転速度が変えられます．

入力A	入力B	出力
High	High	ブレーキ
Low	High	逆転
High	Low	正転
Low	Low	停止

図 5.3　モータドライバの入出力対応

今回は，これをさらに動かしやすくするために専用の基板を作っており，マイコン基板からは最大でモータ 6 機分の回路情報を入力できるようになっています．どの基板がどの信号を受けるようにするかは，基板裏に有るジャンパをつなげることによって選べるようになっています．以下の様な対応表となっているので，製作時には間違えないように注意しましょう．

対応モータ	対応ジャンパA	対応ジャンパB
M1	JP1	JP2
M2	JP3	JP4
M3	JP5	JP6
M4	JP7	JP8
M5	JP9	JP10
M6	JP11	JP12

図 5.4　モータのジャンパ対応表

5.1.2　部品の準備

部品を準備します．まず，基板を切り離しましょう．折れ目にそって曲げれば簡単に割れます．今回はこの内の 4 枚をドライバにします．次に 2 列 L ピンヘッダを 9 コマ毎に切断します．今回はモータ 4 つ分作成するので，4 つ作ります．その次は，2 列 10 ピンの L ピンヘッダの 5 つ取り出し，写真 5.6 のようにペンチで使わない左端の端子 2 本を引き出します．これも同様に 4 つ用意します．最後に，1 列ピンソケットを 7 ピンずつ，4 つ切り出して下さい．ニッパやカッターなどで取り出せば OK です．あとは 1 列 2 ピンのピンソケットを 4 つ袋から取り出せば準備完了です．

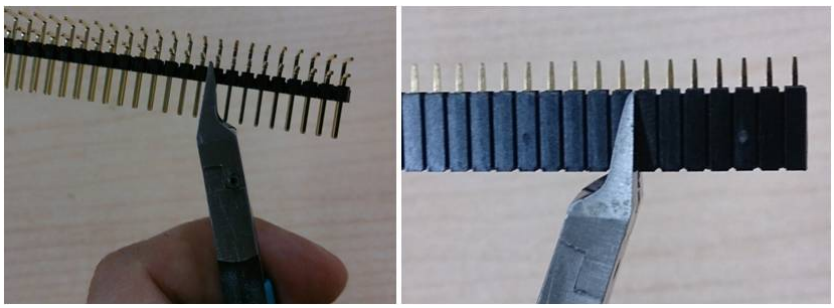


図 5.5　ピンヘッダの分割

5.1.3　ハンダ作業

ハンダを行います．ハンダは高さが低い部品から順に行うと安定します．今回は，先に L ピンソケットと L ピンヘッダを先に付けてしまって，その後 1 列ピンソケット 2 つをハンダすれば良いと思います．部品の向きに十分に注意しましょう．裏面のジャンパは表 8.21 にある，M2,M3,M4,M5 の組み合わせで回路を製作してください．（M1,M6 はモータを増やしたい場合に作成して下さい．）モータドライバを挿したら完成です．ドライバの方向に注意しましょう基盤のプリントに切り欠きがある方に面を向けてください．

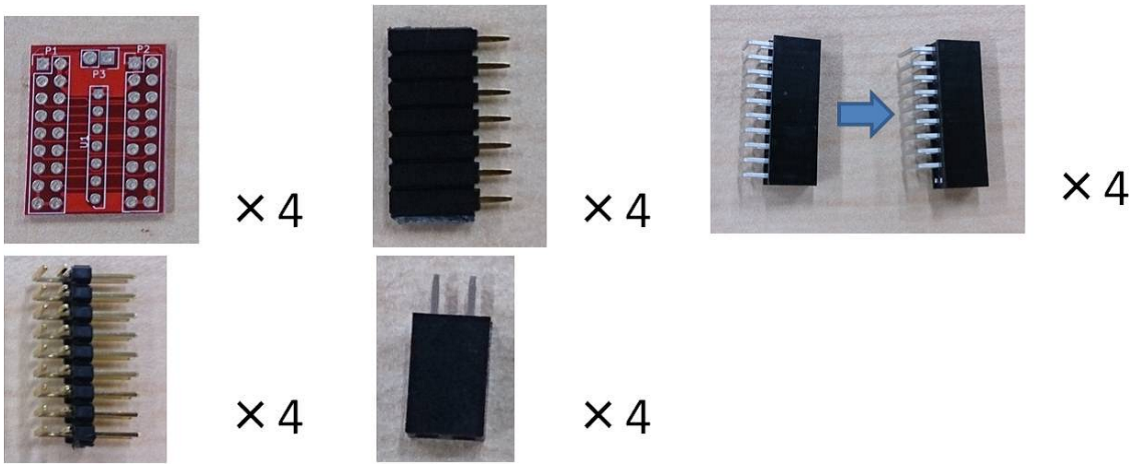


図 5.6　部品リスト

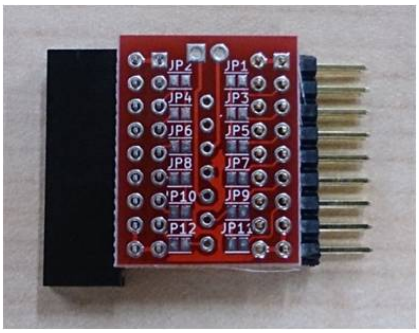


図 5.7　ハンダ作業

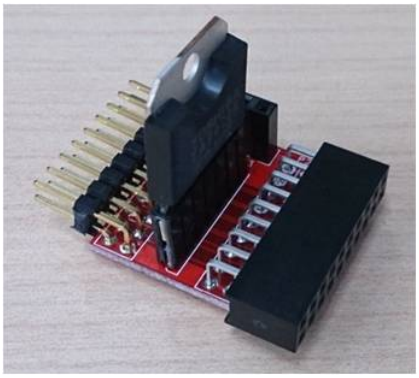


図 5.8　モータドライバの向き

5.1.4　各部の確認

各部の部品が隣のハンダと繋がっていないか，ちゃんとランドに乗っているか確認しましょう．また，ジャンパ部分が写真のように接続できているのかよく確認しましょう．

5.2　マイコンボードの作成

5.2.1　マイコンボードの構造

今回用いるマイコンは ArduinoMicro というものになります．このマイコン基板で電源から出力迄全て揃うようになっていますが，より使いやすくするために今回は基板を用意しました．基板としては I2C，シリアル通信，モータドライバ接続端子，その他の引出線を備えています．キットとしては，今回はモータドライバ接続端子とシリアル通信の端子を主に用いる予定です．完成品の写真を以下に示します．

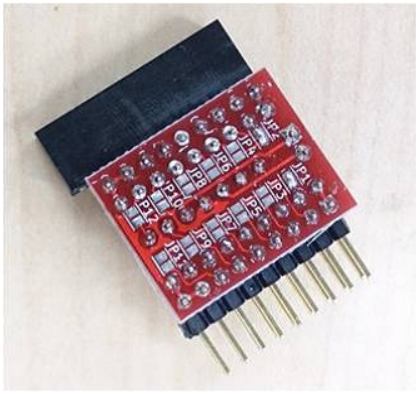


図 5.9　ジャンパを行う（JP1 と JP2 の場合）

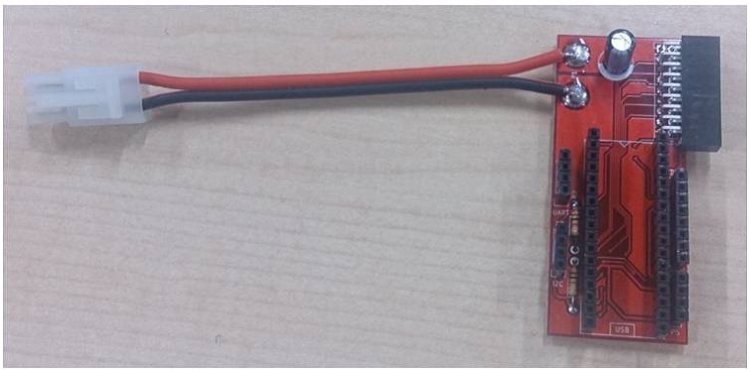


図 5.10　マイコンボード完成図

5.2.2　部品の準備

1k Ω抵抗 2 本，電解コンデンサ 1 本は袋にそのまま入っている状態で用います．その他，先のモータドライバの要領で以下のものを揃えてください．1 列 17 ピンのピンソケット 2 本，1 列 4 ピンのピンソケット 3 本，1 列 8 ピンのピンソケット 1 本，2 列 9 ピンの L ピンソケット 1 本です．また，バッテリ端子ケーブルも用意して下さい．全て揃った写真が以下になります．

5.2.3　ハンダ作業

モータドライバと同じようにハンダ付けをしていきます．抵抗，L ピンソケット，コンデンサ，ケーブルの順につけると良いかと思います．コンデンサには足の向きがあります．足の短い側が白の半円を向くように取り付けます．写真 5.11 を確認し，間違えないように気をつけましょう．すべての部品が半出し終わったら，写真のように上からジャンパ線を必ず配線してください．これを行わない場合は，PC との通信が不可能となります

5.2.4　各部の確認

すべての部品がしっかりと固定され，他の部品とのショートがないかしっかりと確認します．また，部品の足は不要部をニッパで切り落としましょう．全ての部品が完成したらつなげてみましょう．ソケット

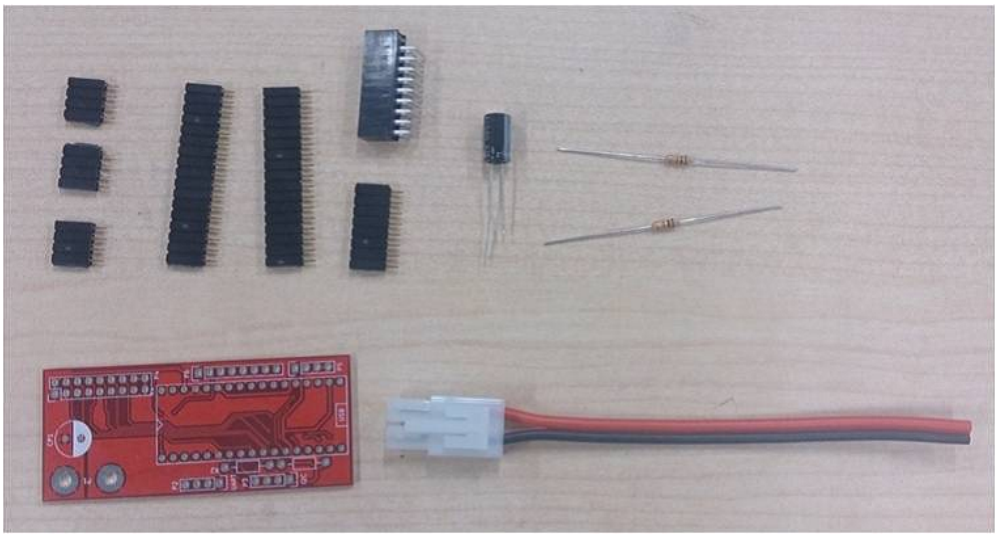


図 5.11　必要な部品

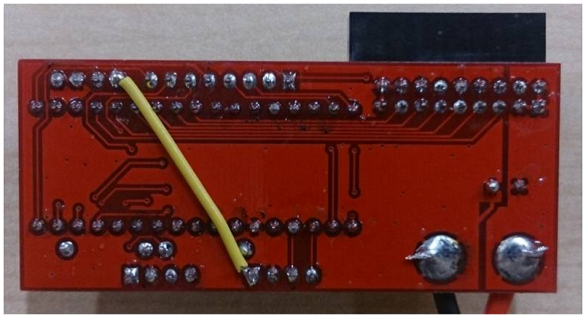


図 5.12　ジャンパ配線を行う

部分がずれて刺さらないようによく注意しましょう．また，全て確認する前には絶対にバッテリを接続してはいけません．

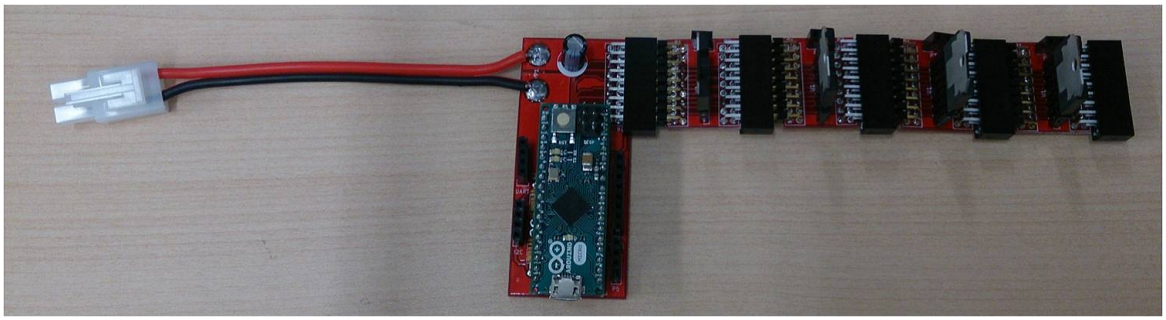


図 5.13　回路の接続

5.3　USB シリアル変換ユニットの作成

キットには 1 列 4 ピンのピンヘッダが付属していますが，これは用いません．1 列 4 ピンのピンソケットを用意して，ハンダしましょう．基板の向きに気をつけて下さい．

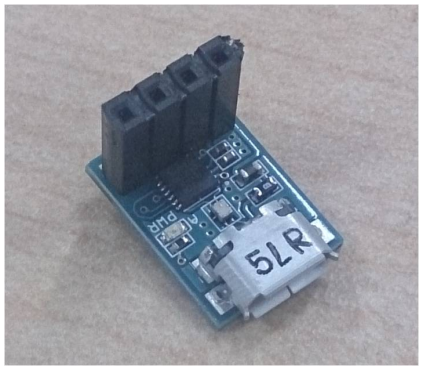


図 5.14　USB シリアル変換ユニット

5.4　各ケーブルの圧着

テザーケーブル，モータケーブルに差し込み用のピンを圧着します．今回用いるピンは TJC8 と呼ばれる種類のものです．全てのケーブルの皮膜を頭から 5 mm 剥がします．テザーケーブルは，まずケーブルを分離した後に行います．分離の仕方は，先のモータケーブルと同じ要領です．先程より難易度は高いので気をつけて行って下さい．全ての皮膜がはがせたら，写真のように根本から順に圧着します．写真ではラジオペンチを用いていますが，お持ちの方は圧着工具を用いることを強く推奨します．作業がとても楽に，なおかつ高精度になります．ここで固定が不安な場合は，先側にハンダをしても構いません．また，隣のピンとのショートをさけるために全ての端子に熱収縮チューブをかぶせましょう．

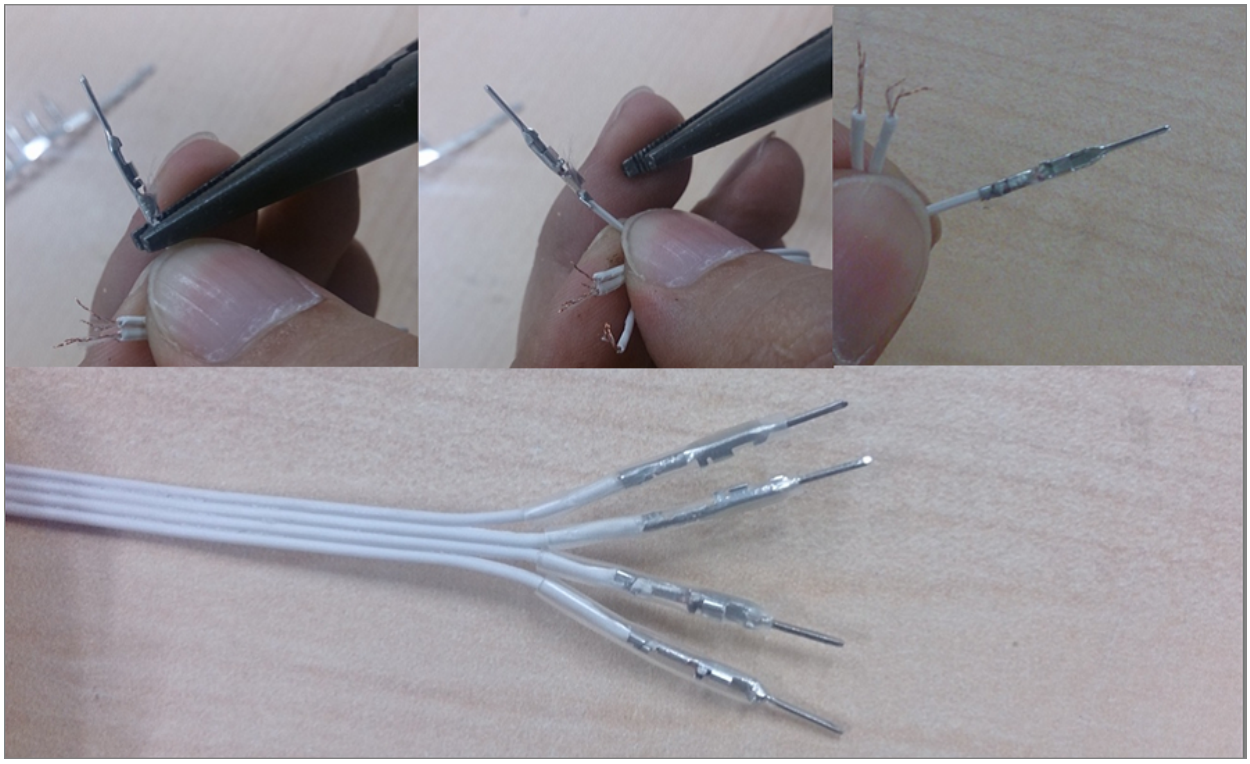


図 5.15　配線材に端子を圧着する

5.5　まとめ

全ての部品が揃ったらこの章は終了です．いよいよ PC と接続をしてロボットが動作をはじめます．

第 6 章

マイコンとパソコン接続

本章の PDCA は以下のようになっています．

Plan　読者が水中ロボットの電装を接続し，PC と通信を成功させる．
Do　　読者が章を読みつつマイコン，ドライバ，モータ，PC とコントローラを接続する．
Check　実際に接続が行われているか確認する．
Action　次章に向けてバッテリの充電を行う．

流れとしては，ここで皆さんにプログラムを学んでもらってから，実際に動作させてもらおうと思っていたのですが，今回はまず動かすことを優先したいと思います．[*1]

6.1　Mark3 のソフト構成について

Mark3 のソフト構成は図 6.1 のようになっています．PC 側では，接続されたコントローラの内容を読み込んでマシン内部のマイコンに送信します．ここでの通信方法は RS232C というものです．今の PC には直接出力する機能がないものが大半なので，前章で製作した USB シリアル変換回路を利用します．テザーケーブルを介して命令を受信したマイコンはそれを元にスラスタの動作を決定します．ここのプログラムについては付属 CD に導入済みですが，8 章以降でその詳細について説明していきます．センサなどはマイコンの方に接続して，その情報とコントローラの情報を合わせて考えることで，より臨んだ動作をできるようにすることもできます．これらの内容も 8 章に導入を入れます．ではまず，ソフトのインストールからはじめましょう．

[*1] なんでって？大人の都合です．

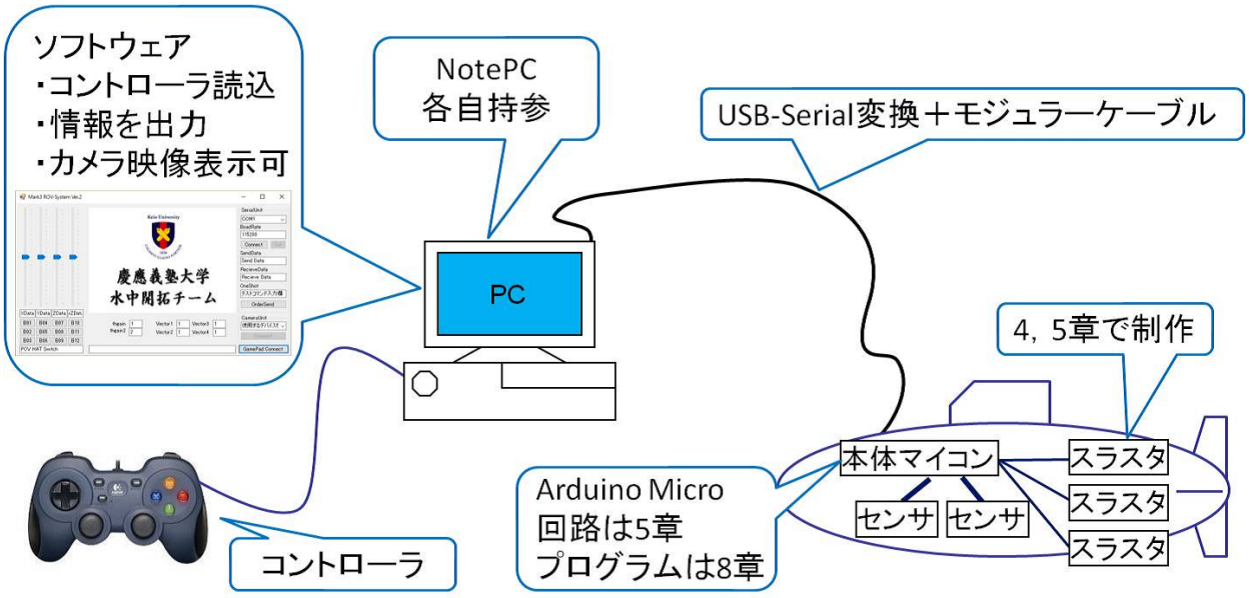


図 6.1　Mark3 と機体の接続図

6.2　ソフトのインストール

ソフトは，付属 CD にある，コントローラというフォルダの setup.exe を選択することでインストールが始まります．なお，対応している OS は Windows7，8.1，10 となっています．XP には対応していません．Apple も対応してません．*2発行元不明となっていますが，これは公式登録がなされていないだけなので気にせずにインストールしてください．インストールが完了するとソフトが開かれるはずです．

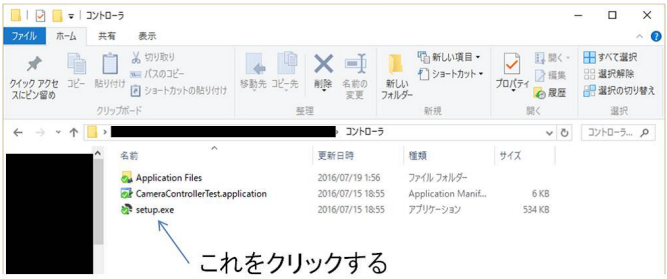


図 6.2　setup.exe のクリック

（一度閉じた場合は，スタート→すべてのプログラム内にある，CameraControllerTest の exe を選択してください．）起動すれば，ひとまずインストール作業は完了です．

各機能を確認してみましょう．画面中央部は映像表示用スペース，その左側がコントローラ情報表示部，右半分がシリアルコントロールなどの設定スペースになっています．まず，ゲームパッドを PC に繋いでみましょう．ドライバのインストールが終わったら，右下の GamePadConnect ボタンを押してみてください．押したボタンや，アナログスティックに合わせてスライドバーやボタンが動作することが確認でき

*2 というか Apple には対応する気はありません．宗教上の理由で．

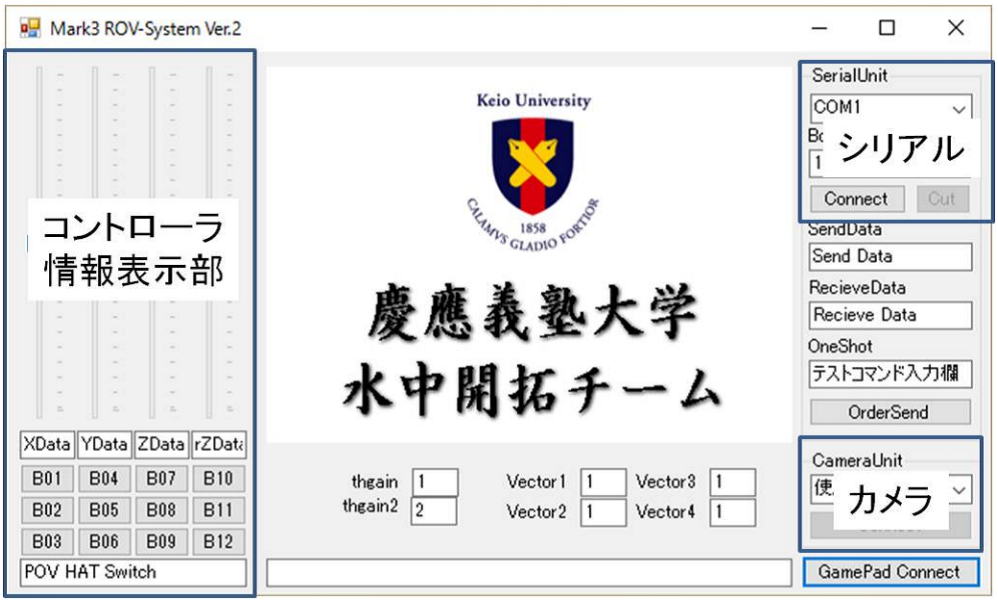


図 6.3　ソフトウェアの起動

るとおもいます．なお，このソフトはコントローラの操作モードが LED 消灯状態で利用する形になっています．LED が点灯しているときは，Mode ボタンを押して消してください．また，スライドバーが触れていない時に 0 点に戻らない場合は，コントローラ側でキャリブレーションをしてください．[*3]次に，先ほどハンダした，USB シリアル変換ケーブルを USB を介して PC に接続してみましょう．ドライバがインストールされたら，デバイスマネージャを開きます．写真のように，ポート（COM と LPT）のツリーを確認すると USB Serial Port の項があるはずですので，そこの COM ナンバーを確認してシリアルユニットのプルダウンメニューからその値を選択してください．

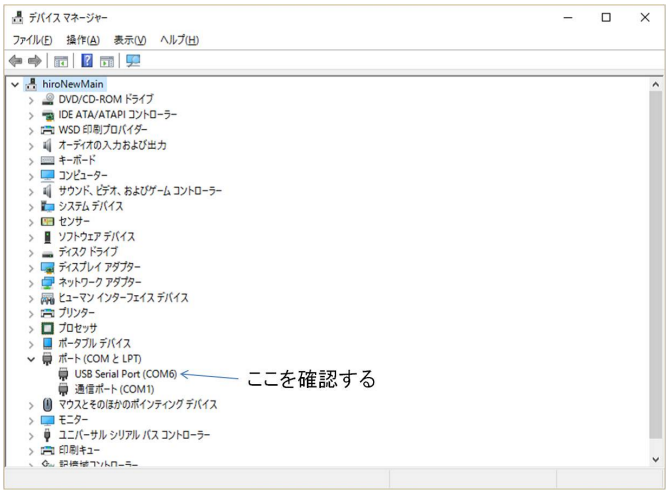


図 6.4　COM ポートの確認

あとは，Connect ボタンを押せばシリアル通信が成功するはずです．エラーが発生する場合は，接触不良などの可能性があるので確認してください．最後にカメラユニットです．カメラユニットは USB 接続の Web カメラやビデオキャプチャーアダプタなどの情報を入手して画面に表示する機能になっていま

*3 デバイスとプリンタのウインドウから，コントローラを右クリック，ゲームコントローラの設定を開いてプロパティへ．設定タブから調整ボタンをおしてあとはウィザードに従ってください．

す.[*4]ただし，複数の接続カメラが有る場合は，一番最初にインストールされたものが表示される仕様になっているので，写ってほしくないカメラが有る場合は事前にデバイスマネージャなどから，一度無効化するなどの処置をして読み込まれないようにしてください．使い方は簡単で，CameraUnit のプルダウンメニューから DSShow.Net（略）を選択して Connect ボタンを押すのみです．カメラの相性などの問題も有り，一部のカメラは読み込めない場合があります．ご了承ください．なお，中央部の Thgain などは，今後増える拡張機能です．触っても何も起こりません.[*5]

図 6.5　COM ポート接続

6.3　回路の接続

回路を写真のように接続します．

USB シリアルのケーブルの向きに気をつけてください．マイコンボード上の UART と書かれている端点と USB シリアル変換ユニットの 5 V が同じピンとなる向きで接続します.[*6]モータドライバは，マイコンに近い順に，5,2,3,4 の順で接続をします．モータドライバの 2 ピン部分がモータの接続部です．まずは適当にすべて同じ向きで繋いでおいて，後で回転方向を確認してケーブルをつなげる順番を変更しましょう．

[*4] 私はデジ蔵を使ってビデオカメラをキャプチャして使っています．
[*5] ほんとだよ，なんの機能もないからね．
[*6] 逆差しダメ絶対！

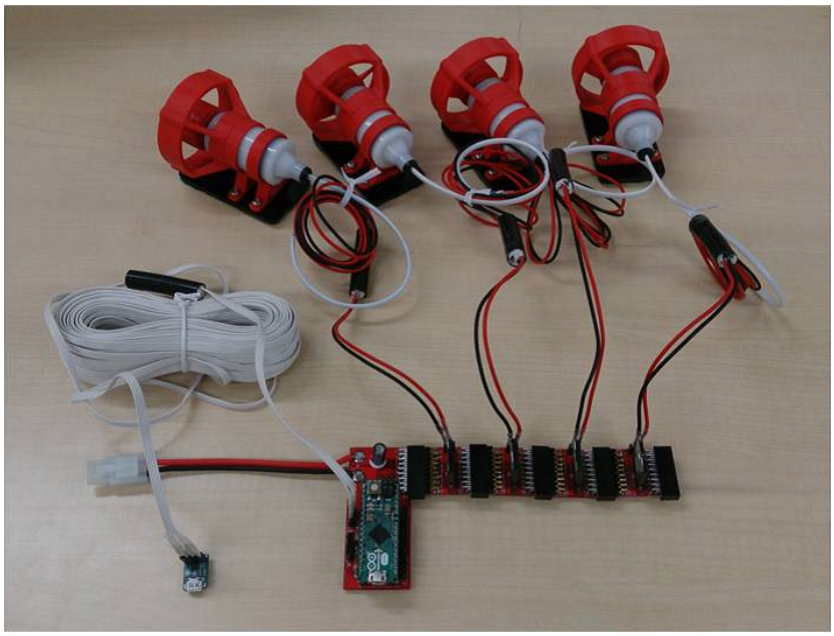


図 6.6　回路の接続

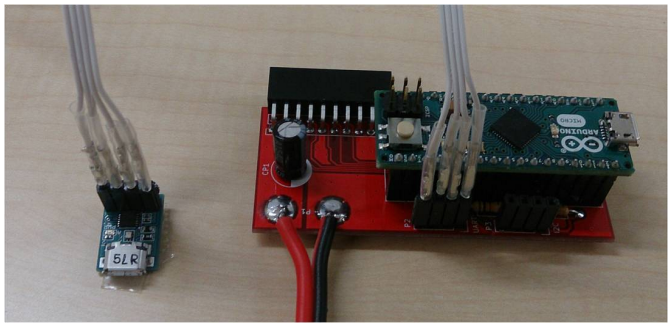


図 6.7　ケーブルの向き

6.4　動作の確認

すべての接続が終わったら，コントローラと USB シリアル変換ユニットを PC に接続します．先の要領でソフトで，コントローラの読み込みとシリアルの接続を行ったら，バッテリを繋いでみてください．コントローラのアナログスティックを動かすとスラスタが回転するはずです．現状のコントローラは下の図のようなコントロールになっています．これであとはすべての電子回路とバッテリを耐圧殻に入れたら動作可能な状態となります．バッテリにオスの面ファスナを，耐圧殻の底にメスの面ファスナを貼り付けて耐圧殻内にバッテリを固定します．その上に電子回路を入れて，蓋の裏側中央部に錘を一本貼り付けます．そうすれば機体は完成となります．お疲れ様でした．

まずは，スロットルのみを倒し，すべてのスラスタが正方向に回転することを確認します．ここで，正方向に推力が出ていないものがあったら，モータドライバのモータ接続部のピンを逆にします．すべてが正回転する状態になったら，機体を正面に見て図のように左下から時計回りにスラスタを固定していきます．

図 6.8　コントローラの設定

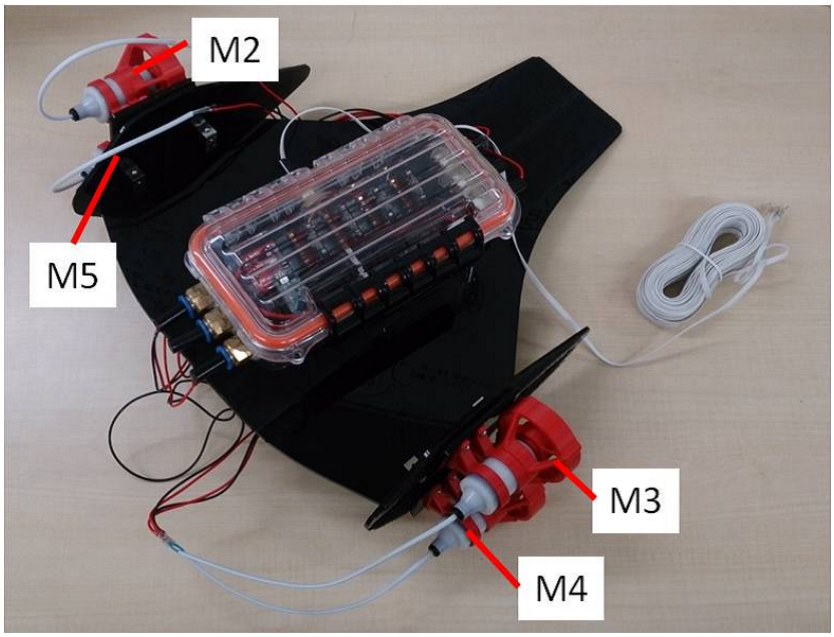


図 6.9　スラスタの配置

6.5　まとめ

この章ではあまり中身の説明がなくて申し訳ありません．興味がある人は，8 章でもう少し細かくこの内容を説明するので，お楽しみに．次章ではいよいよ実際に水中ロボットを泳がせます．バッテリを充電し，プールを用意して次に進みましょう．

第 7 章

浮力調整とメンテナンス

本章の PDCA は以下のようになっています．

Plan　読者が水中ロボットの動作を力学を用いて理解する
Do　　読者が 2 章の内容を考えながら浮力調整を行う．
Check　狙いどおりの動作をするか確認する．
Action　上記の作業を反復的に行い，自分のイメージする機体の動作と実際の機体の動作を，力学の概念ですりあわせることを可能にする．

7.1　水に入れる前に

いよいよ水中ロボットを水に入れて本当に水中ロボットにするわけですが，事前のチェックを怠れば必ず不幸な結果を招きます．[*1]確認事項は主に防水と接続，そして機体構造に分かれます．まず，防水です．チューブがしっかりと耐圧殻に接続されているか確認しましょう．次に，耐圧殻のシリコングリスが，しっかりと塗られているか見極めます．あとは，蓋を閉める時にケーブルを挟まないように注意しましょう．接続側は，先ほどの章を参考にケーブルがすべてしっかりと接続されているか確認しましょう．先に動いた構成を防水ボックス内で再構成できれば OK です．最後に機体構造ですが，ネジなどの緩みが無いか，スラスタの取り付けがしっかりしているか確認しましょう．これらのチェックが全て出来たら，水に入れてみましょう．

7.2　浮力を調整する

水に入れたら，機体は浮いているはずです．浮いているということは，浮力が重力より勝っているはずですよね．浮力中心の位置は想像できるでしょうか？重心の位置は？浮かべた段階で機体が傾いているならば，重心の位置や浮力中心の位置を想像して，バッテリや耐圧殻の位置を調整しましょう．とりあえずは機体が水平になれば，静的な力の釣り合いを体で理解したということで第 1 段階クリアです．次に第 2 段階として，浮力中心と重心位置を考えながら，スラスタの角度や位置をどう調整すれば機体がよりうまく泳いでくれるか考えて調整しつつ，泳がせてみましょう．狙いがあるならば，残った錘を切り分けて，フレームや耐圧殻内に貼ってみても良いと思います．うまく泳げば第 2 段階クリアです．ひとまずは動く水中ロボット完成です！

[*1] 何をするにもまず準備．とにかく準備．

7.3　水に入れたあとに

この世の中にあるおおよその動くものはメンテナンスを必要とします．油を塗る，ネジの緩みを締め直すなどです．最近は電子デバイスが多く，物理的に動かないためにこのメンテナンスというものを忘れがちになりますが，これを無視してものが動作し続けるということはありえません．水中ロボットは，特に電子機器や金属という水と相性の悪いものを水に入れていくわけですから，これらについて注意してメンテナンスを行っていく必要があります．具体的には，耐圧殻ならば，ケーブルの緩みやシリコングリスの塗り直し，中が湿気てしまえば乾燥作業などが挙げられます．スラスタはもっと深刻です．回転部からの浸水は避けられませんから，ひどくなれば中に水が溜まってしまいモータの出力が大きく低下します．振ってみて水の音がする，もしくは外見から疑問を覚えるような色にケースが変色しているなどの問題が見つかったら，ケースを開けて水を出しましょう．それでも回復しなければ，モータを交換してしまったほうが良いでしょう．ちなみに，ケースの開け閉めを頻繁にしすぎても劣化しますのでほどほどに.[*2]それ以外に見逃しがちなのはバッテリです．NiMH（ニッケル水素）バッテリは，長時間使わないと複数回の充放電を行ってのリフレッシュが必要になります．また，メモリー効果というものがあって，継ぎ足し充電を頻繁にすると容量の低下を招きますので，充電前にバッテリをある程度使っておくほうが良いでしょう．（ただし完全放電すると容量が急激に低下するので注意してください．保管は充電してからにしましょう.）

7.4　まとめ

浮力調整とメンテナンス，理解できましたか．せっかく作ったものです，長く使って愛着を持ってもらえると作者としては嬉しいところです．浮力調整はカンでもできますが，本当に狙ったとおりに動かしたいならば，しっかりと重力と浮力の関係を考えて調整するべきです．ぜひ機体の図を書いて，どこに浮力が掛かっている，どこに重力があって，推力はここで出ているといったように力のベクトルを書いて実物と照らしあわせてください．モデル化して実機とのすり合わせを行うことは，科学の第一歩です．次章以降は，今まで行った内容を活かして更に突っ込んだ内容に触れていきます．目的を設定し，自分好みの機体を作っていっていもらえればと思います．

[*2] 何事も過ぎは良くないのです

第 8 章

改造と最適化をすすめる

本章の PDCA は以下のようになっています.

Plan　読者が改造の方法について理解，実際に行えるようになる
Do　読者が本章を読みつつ改造目標の設定を行う
Check　改造した結果が目標とした内容と合致しているか確認する
Action　上記の作業を反復的に行い，自分のイメージする機体の動作と実際の機体の動作をすりあわせる

8.1　改造とは何か

今回の章タイトルは「改造と最適化を進める」というタイトルですが，みなさんは改造がどのような意味を持つことば知っていますか？手近な国語辞書で「改造」を引くと，「建物・機械・組織などをつくり直すこと．別の用途にかなうようにつくりかえること.」とあります．つまり「作りなおす」だけではなく，「用途にかなう」ことが改造の真髄であると言えるでしょう．よって，構造や仕組みを変えただけでは「改造」と言えません．この章では，この「改造」について，PDCA の流れにそって行っていきたいと思います.

8.2　目標を決めよう

さて，前節では「用途にかなう」ことが改造の真髄であることをお話しました．ではどうすれば「用途にかなう」ということを証明できるでしょうか．これには改造にあたっての目的となる「用途」を明確にすることがもっとも重要であるといえます．PDCA サイクルに当てはめるならば，この部分は Plan に対応します．実際に制作する行為は Do であり，その後 Check をしてその結果から Plan が成立していることを示すわけですから，ここでの目標は達成が確認しやすい内容であることが重要でしょう.[*1]

8.2.1　目標を明確にしよう

では，達成の確認しやすい目標とはどんなものでしょうか．例えば私であれば，以下の様なルールを守って目標を決めると良いと考えています.

[*1] 作業の難易度が低いからやりやすい！という意味ではないです.

- 目標は解釈が分かれない文章にする
- 達成評価は数値化，もしくは二値化（Yes ／ No）で判断できるものにする
- 達成確認を行う”締切”を明示する
- 目標を達成するために複数の作業が必要になる場合は，目標を細分化してツリー構造にする

一つ一つ説明していきましょう．一つ目は，もっとも重要な項目です．目標とする内容に複数の解釈が可能な文章とすると，開発を行う人間の都合によってその評価が変わってしまうからです．例えば期末テストに向けて「テストで良い点を取る努力をする」という目標を決定したとしましょう．果たして何点からが良い点でしょうか？おそらく人によって良い点というのは変わってしまうのではないでしょうか．努力とは？努力は行為であって結果ではありません．[*2]よって，この文章からは明確な目標が設定できません．誤解を招かないような明確な文章である必要があります．例えばロボットであれば「このロボットは（方法）によって（結果）を達成することを目的とする」といった具合に．

次に，目標の数値化，二値化です．これも同じようにテストで捉えるならば，明確に「できたかできていないか」を示すために最も簡単な方法は，この 2 つの方法を使うということになります．例えば，目標点数を定めることで，その点数にたどり着いたかどうか，満たせなかった場合にどれほど足りなかったかが明確にできます．

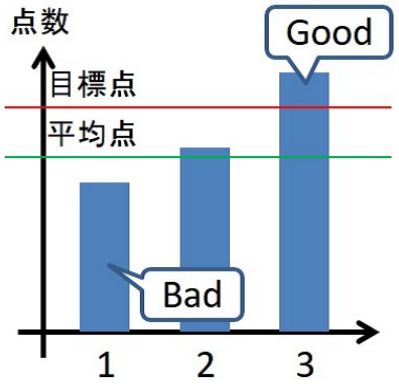


図 8.1　目標を明確にすればゴールしたかわかる

先のテストの例を取るならば「数学で 70 点をとる」というふうに定めれば，得点が足りているのか足りていないのか，あと何点必用だったのかを明確に見ることができるでしょう．ロボットだったら「目標速度〇 m ／ s の達成を目指す」とか「重量〇 kg の物体を持ち上げられるようにする」などでしょうか．何か評価をするために単位のあるものを選ぶことで，結果を明確にすることを心がけましょう．

締切を明示することは意外と大切なことです．学校のテストやロボコン参加といった目的であればそのタイミングが自動的に決まりますが，趣味や長期計画になる目標はゴールもぶれがちになり完成せぬままにのびのびとなっていきます．[*3]だからこそ，決定する目標には明確に期日が見えるといいでしょう．

最後は目標を更に細分化していくという内容です．第一章のクルマの図 8.26 を覚えていますか？この図で，クルマを高速化することを目的とすると（今回は単位が明確ではないから大雑把に書きます），タイヤ径や回転速度などいろいろな改造の手段を挙げることができるでしょう．こういった場合は，目標を

*2 大人の言う「善処します」はまず行わないものと思いましょう．

*3 期日を定めずに親に預けたお年玉の行方はどこに行ったのでしょうね．これだって，「お金を増やす・管理する」という目標の元に「親に代行を依頼する」という手段を採っただけなのですよ．「いつまでに」「どれくらい」という明確な条件がなければ一生預かることももちろんできますねぇ

ツリー構造状に分解していきます．例えば図 8.3 のように．

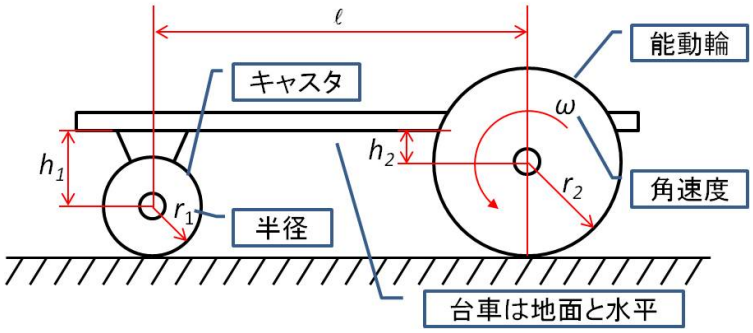


図 8.2　タイヤロボット

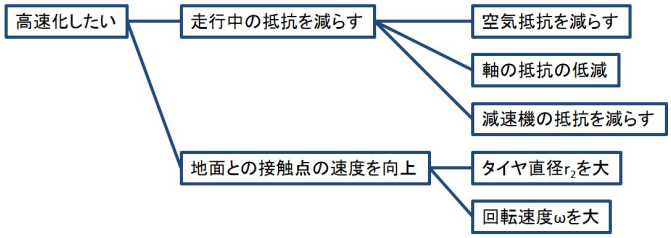


図 8.3　高速走行に選ぶことの出来る手段（例）

なお，今回は明確に無視していますが，一般的にはその小目標同士が矛盾する関係であることが往々にして有ります．「高速化するためにパワーの有るモータをつもうとしたら重くなってしまう」，とか．こういった，一つのシステムの中で複数の問題が絡み合っている場合に，バランスを取って全体の機能を向上しようという学問はシステム工学というものが存在しています．興味がある人は是非調べてみてください．

8.2.2　問題を洗い出そう

先程，「目標を明確にしよう」の章で，あっさりと小目標のツリー構造化の話を行っていますが，どうやれば良いのかわからないという人も多いと思います．この節ではそれについて簡単に説明したいと思います．さて，小目標のツリー構造化するためには以下の流れを用いるのが一般的です．大きく分けて 2 つの方法があります．

- ブレインストーミング＋ KJ 法
- マインドマップ表記法

順番に説明していきます．ブレインストーミング＋ KJ 法はブレインストーミングによって洗い出された項目をと KJ 法という方法でまとめることによって，目標と要素の関係性をまとめていく方法です．ブレインストーミングは以下の手順を繰り返して要素を作ります．

- テーマを決める
- 参加者で順繰りに関係する要素を一つ以上出していく
- 出された要素ごとののカードを作成する

- 最低参加者で 3 順以上は繰り返す

ブレインストーミングを行うときは，以下のルールを守ることが大切です．

- 批判しない：他者の出したアイディアについて批判してはならない
- 自由に発言する：とにかく思いついたことを挙げること
- 質より量を重視：価値は関係ないのでとにかく提案する
- 連想して結合する：他人の意見から連想したり，相手の意見と自分の意見を結合して提案しても良い

これによってカードが得られたら KJ 法によってまとめていきます．ブレインストーミングで記述したカードを全て広げて，関連性の有ると思わしきカードごとにまとめて中グループをまとめていきます．大体まとまったら，それに表札のカードをつけて束ねます．その中グループ束に対して，先程の様に関連性に合わせてまとめて大グループを作っていきます．大体大グループができたら，グループ同士を漫画の人物相関図の様に配置して関係性を付記して完成となります．例えば，雲を例にブレインストーミングと KJ 法をしてみたらこのようになりました．

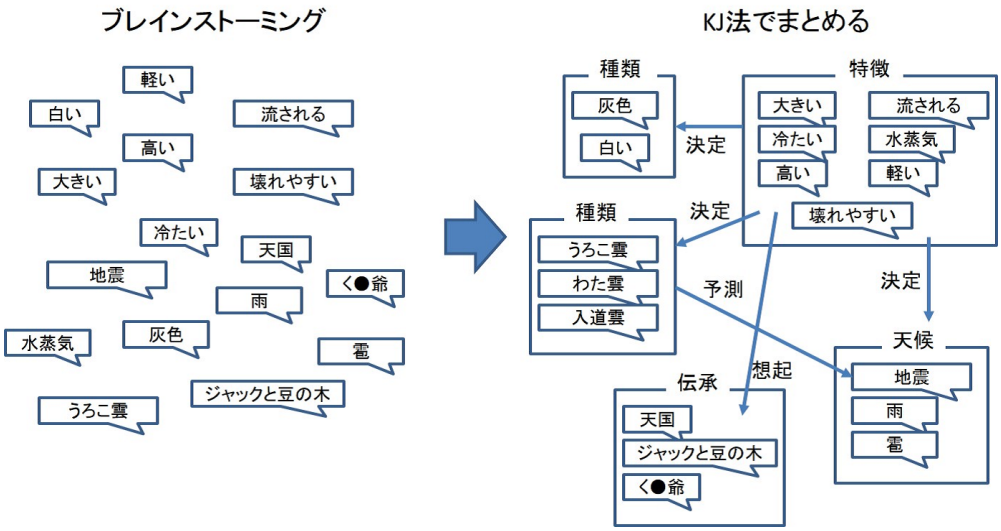


図 8.4　雲に関する考察

次に，マインドマップです．マインドマップは中央から外側に向かって思考を広げながらその関係性を整理する方法です．以下の順番でマインドマップを書くことができます．

- 中央にテーマを書き込む
- 思いつくことを枝にしていく
- 更にそれに連想される内容の枝を広げていく
- 俯瞰して関係性を考える

例えば，図 8.3 は左から右になっていますが，これは実質的には限定されたマインドマップです．コツとしては，連想ゲームになるので，一本目の枝をどれくらい太いものにわけられるかによって決まると思います．

ブレインストーミング＋ KJ 法とマインドマップは両方共思考整理のための方法ですが，必ずしも同じ内容や結果を生むものではありません．特徴を整理するとブレインストーミングはより思いつきを組み上げていく，下から上に向かってアイディアを育てる方法で，マインドマップは逆に前提ができていてそこから枝葉を探していく方法になります．アイディアが出てこないときはよりブレインストーミングが向いていて，状況を明確にしたいときはマインドマップが向いているかもしれませんね．ちなみにブレインストーミングは上に向かって行くだけですが，KJ 法をつかってまとめることでマインドマップのような機能も求めることができます．*4

8.2.3　目標を明文化しよう

目標を明確にできたら，それを文章化してチーム内もしくは個人で確認を行いましょう．ここで，ひとつ目の目標そのものはなるべく端的に一文で示します．先にも挙げた，「このロボットは（方法）によって（いつまでに）（結果）を達成することを目的とする」という具合ですね．次に，それに付属する目標を以下のような文章で並べていきましょう．「これを達成するために，（ポイント）について（評価）を満たす（手段）を構成する」ただし，この小目標文は大目標と矛盾しないこと，目標を達成するにあたってその少目標が構成の一部となっていることに気をつけましょう．この，大目標と小目標の組み合わせが成立していなければ最終的に良いものが出来上がることはありません．前節のツリー構造化ができていればこれはほとんど防げます．

8.2.4　作業を具体化しよう

目標が明文化され，小目標が決まってしまえば，あとはこれに合わせて詳細を決めていくだけです．少目標を達成するためにやるべき作業を列挙し，スケジュールを立てて作業を進めましょう．*5「目標を明確にしよう」の章において行ったツリーがあれば，それを元に明文化された目標に合わせて作業に分解すると良いと思います．作業に分解したら，それに合わせて必要になる時間を考えてスケジュールを建てます．ここで，決まった締切までに終わらなそうな作業量がある場合は思い切ってツリーの一部を切り捨ててしまうと良いかと思います．スケジュールを建てる場合は，例えばガントチャートという手法を用います．下に例を示します．ここで，表の行は日程です．そして，列には作業を項目別に分けて記載し，実際に作業をしている日程の部分に色を塗ります．特に細かいルールは無いですが，下の図 8.5 では人によって色を替える方法を採っています．つまり，表を日で区切って列で見た時，異なる色が存在するのは人が並行で作業をしている状態，同じ色が存在する場合は一人の人間が複数の作業を並行で行っている状態です．前者は積極的に行なって暇な人間を減らし，後者は避けることによって個人の負担を減らすことをオススメします．（図 8.5 ならば 5 日に青の人が機械加工と機械設計を同時にしていて，赤の人は基板の設計をしています．はっきり言って完全に破綻したガントチャートです．真似しないで下さい．）人が色で決まっていて，作業量が線ですから，色の面積は（サボってなければ）作業量に相当します．仲間同士で出来る限り一定にすると良いと思います．この図 8.5 では青さんが随分苦労している気がします．赤は（この中では）余裕ある作業をしていますね．また，作業に分解したときに「設計」と「製作」の様に同時に行えないものの線がかぶっていないように気をつけましょう．設計しながら開発していると，設計変更で行った開発が全部パーになりますよね？そういう意味では，同時に行えない作業同士には若干の隙間を

*4 上から下に至る場合をトップダウン，その逆をボトムアップといいます．
*5 なかなかスケジュール通りには行かないものだけれど，行かないからといっていらないものではないのですよ．

開けておくことも大切です．これは作業の遅れを取り戻したり，リフレッシュに用います．

図 8.5　破綻しているガントチャート

スケジュールの使い方

うまく行こうが行くまいがスケジュールには製作する意味があるものです．それは何故でしょう．今の自分の考えている能力と実際の能力が明確にわかるからです．例えば先のガントチャートで作業日程を決めたとしましょう．これで実行後にわかる実際に掛った時間と比べれば，その差分は実際の自分の想定していた能力と実際の能力の差ということになります．この差分を識ることによって，今後の自分の能力を伸ばす方法を考えることができます．また，次の計画をたてるときに遅れの発生しにくい計画をたてることができるようになります．遅れの発生しにくい計画を建てることができるようになれば，作業をしたり休むことが出るのでより心に余裕が出るわけです．これも PDCA 的改善活動の一つといえますね．

8.3　実際に改造をしよう（ハードウェア編）

本章では，具体的に改造を行っていくにあたってこちらが用意しているエッセンスを説明します．もちろん，自分たちで行う方法が思いつけばどんどんそれを行ってください．

8.3.1　スラスタの配置を変更する

最も簡単なのはこれでしょう．マジックテープで固定されていますから剥がして貼り直せば移動の方法を自由に変えることができます．例えば，今は航行型のスラスタ配置にしていますが，ホバリング型に変更することも可能です．ただし，もともとされているプログラムでは狙い通りの動きをすることは難しいと思うので，もしホバリング型の移動形態を取るならばプログラムの変更を考えましょう．プログラムの解説は次の節で説明します．他にも，今はスラスタの幅を広く，高さを小さくしていますが，これを変更することでも移動特性は大きく変わります．2 章で挙げた力学的内容について検討をしつつ，有利な配置を考えてみましょう．例えば，基本構造を同じにしてより潜りやすい構造にしようとすれば，重心を中心として，回転に影響する力である推力，浮力，流体力を考える必要がありますね．この場合，モーメントの釣り合いからスラスタの位置や角度をどうすればよいでしょうか．このように物理の基礎的な思考から，機体の動作を考えて合理的により良い構造を探していきましょう．また，配置を変更することで浮力中心や重心の位置が動いてしまうことが有ります．これは，マジックテープを浮力の有る発泡スチロール等に貼り付けて浮力材としたり，付録の錘を使って調整しましょう．

浮力と重心の位置を調べる方法

これらの位置を実際に測定するならば，例えば写真のように物体を異なる二箇所以上で紐で吊れば重心の位置を出すことができます．これは水中ロボットが浮力中心と重心の位置で姿勢が決まることと同じように，紐による引っ張る力（張力）と重力が完全に釣り合い，モーメントがゼロになる点で静止するためです．

では浮力中心はどうすれば求められるでしょう？事前に重心の位置がわかっていれば，案外簡単に求められます．水中でおもりを追加して機体を傾かせればその姿勢，おもりの位置，重さから浮力中心の大体の位置を識ることが出来ます．これは重心を求めるのと同じ要領で，重心を動かして浮力中心と釣り合う位置から求める方法です．おもりを付けたときの重心変化による姿勢の変動を陸上で測っても良いのではないかと思う人もいるかも知れませんが，これはいけません．重りの体積による浮力によって，水中と陸上で重りから伝わる重量が変化するためです．

8.3.2　フレームを作りなおす

スラスタ配置と同様に力の釣り合いから考えていけば，今のフレームが目的に対して必ずしも有利であるとは限りません．例えば，このままスラスタ配置のみをホバリング型の配置に変更してしまうと，潜行時にうける流体抵抗は膨大なものになってしまいます．そこで，場合によっては新しいフレームを作ってみるのも良いかと思います．もちろん今までと同じように，アクリル板にマジックテープで固定するのも良いですが，スラスタの固定具などは固定具の寸法を ϕ 3mm で 5mm 等間隔穴のパンチングアルミ板やタミヤのプレートに合わせてあります．例えばこの写真は，タミヤの固定プレートにスラスタを固定してみた写真です．パンチングアルミ板は簡単に曲げることができますから，板の加工によって箱組にすることも可能です．目標に合わせて加工して見て下さい．

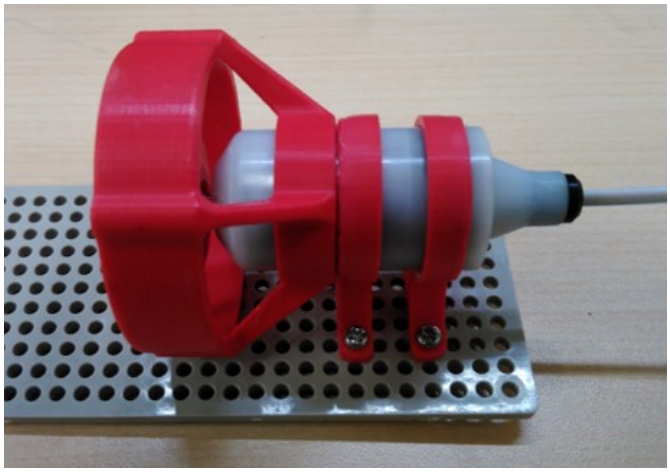


図 8.6　タミヤフレームへのスラスタ固定

8.3.3　メカをつくる

先に挙げたようにフレームはタミヤの構造材やキットと互換性を得られるように作っています．アクチュエータも同じようにそのような機能を持たせてあります．例えば，タミヤコネクタは，タミヤの遊星ギヤボックスに防水モータをつなぐことができるようになっています．図 c のように，遊星ギヤボックスに接続し，モータユニットにギヤを圧入してからセットします．これを使うことで，より大きなトルクを得られますから，より大きなスクリューやリンク機構を動かすことが出来ます．写真は，レインボープロダクツのプロペラを接続したものです．径が大きくなったので 2 倍以上の性能を出すことが出来ますが，ただ軸に刺しただけですから軸がぶれてしまいますし，周囲のものを巻き込みます．カバーをつけたり固定方法を検討したりする必要があるでしょう．リンク機構の例として羽ばたき機構を示します．写真はスライダクランクというクランクの回転をスライダ部で往復に変換する機構を用いて羽ばたかせています． この羽根の配置は横方向に接続してあるのでエイやペンギンと同じ配置ですが，これを縦にすれば魚と同様ですね．リンク機構は昔から用いられてきた運動方向を変換する仕組み（変換機構）です．例えば回転運動を直線運動に変えたり，回転運動を往復運動に変えたりできます．（なお，ここからの図において，黒丸は地面に固定された動かない点で，白丸は動作を許す回転軸です．）

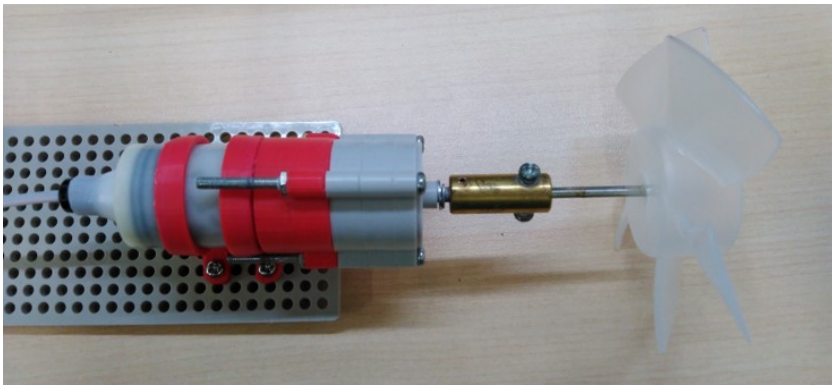


図 8.7　タミヤコネクタの使用

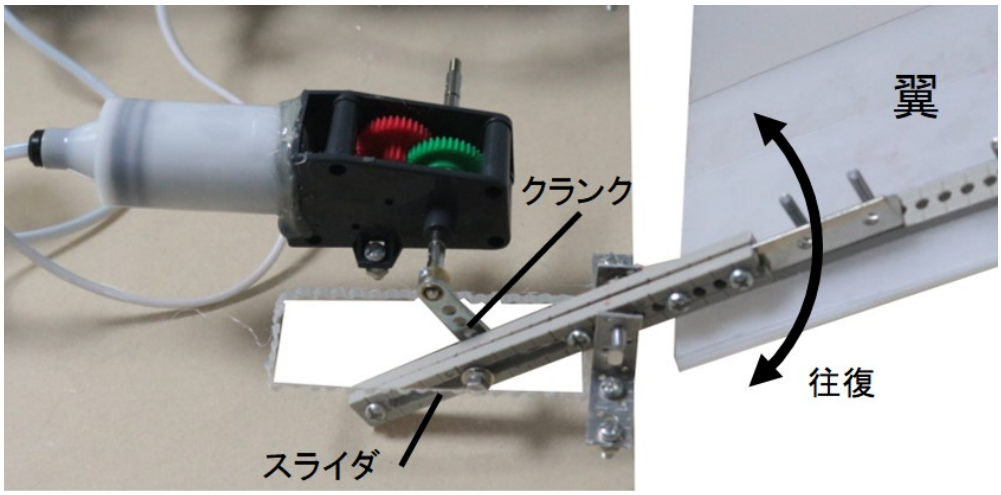


図 8.8　揺動スライダクランクによる羽ばたき機構

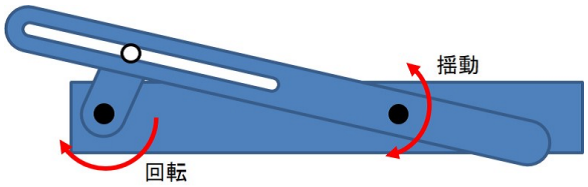


図 8.9　揺動スライダクランク

例として幾つかのリンク機構を以下に示します．

(a). スライダクランク：エンジンのピストン等

(b). 揺動クランク：クルマのワイパー等

(c) テオ・ヤンセンリンク：歩行ロボット

インターネットで検索すれば多くの例が出てきますから，ぜひ確認してみてください．動作は幾何的な作図を行えば簡単に求めることができますから，自分の好きなリンク機構を作ってみましょう．逆に言えば，リンクの長さの比を間違えてしまうとリンク機構は全く動きません．テオ・ヤンセンリンクなどは特に難しいので，作る場合はリンク比を確認しましょう．例えば，「リンク ハンド」などで検索すればかなりの数のリンク機構を見ることができます．

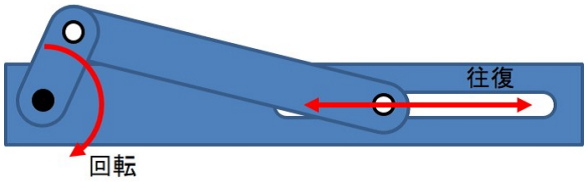


図 8.10　スライダクランク

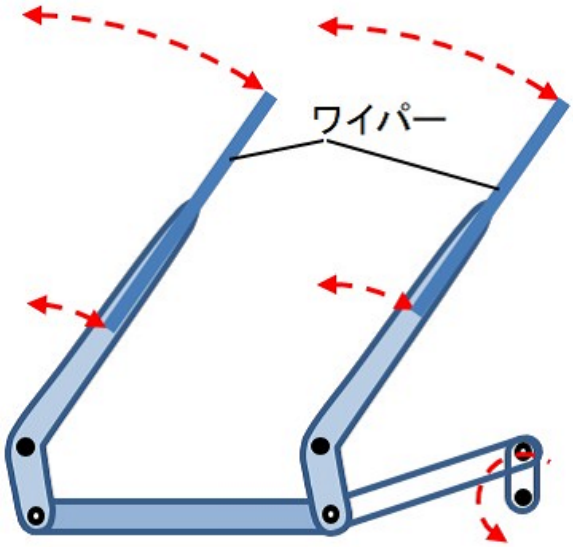


図 8.11　揺動クランク

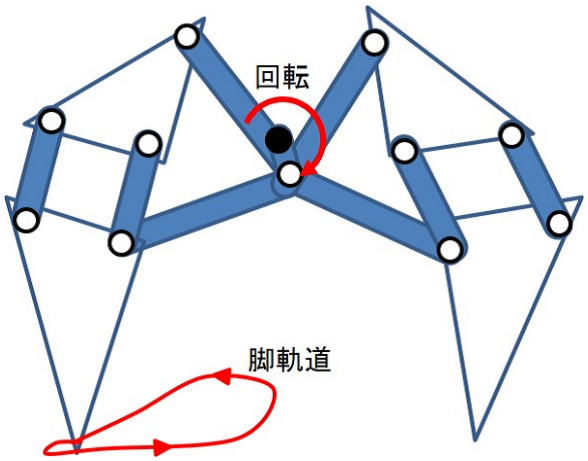


図 8.12　テオヤンセンリンク

8.3.4　カメラを作る

プールであれば水も綺麗ですし，近くで動かしていますから機体の状況を確認することが出来ますが，機体が遠くにいたり，水が汚れていると近くでモノを見るためにカメラが必要になることも有るでしょう．このキットでは，ソフト側でカメラ情報を表示する機能を持っていますから，後はカメラを搭載してしまえば簡単にカメラ映像を見ることが出来ます．では，カメラはどのように作ればよいでしょうか？今回は細いケーブルを用いるために，小型のカメラを用意しました．こういったカメラは，模型搭載用や車載用カメラとして販売されています．秋月電子でも類似した品が売られています．こういったカメラは入力に 9V〜12V が必要になることが多いです．また，モータなどと同じ電源を用いるとモーターによるノイズから安定しないので，別電源化しましょう．今回は，陸上から給電する形にしました．構造は至ってシンプルで，図 8.13 のように 3 つのパーツで防水箱を構成しています．蓋と筒は接着して内部に浸水し

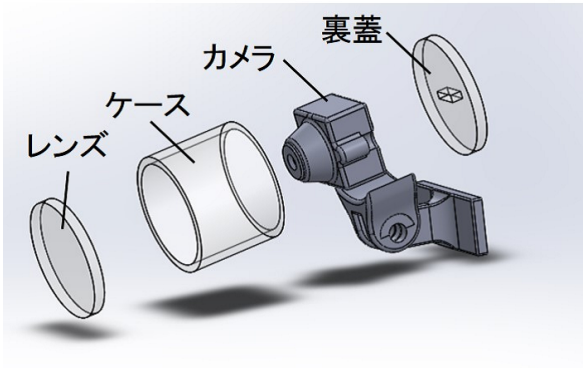


図 8.13　防水ボックスの構造

ないようにしています．これでは開け閉めできませんが，作りとしては最も簡単です．開け閉めを考えるならば，水道管や 100 円ショップの入れ物などを使って制作しましょう．蓋と筒の部分をそのまま利用して，覗き窓の部分はきれいなアクリル板などを接着すれば大丈夫です．水道管の接合部は塩ビ用接着剤があるので，それを用いて溶着します．ケーブルと管はエポキシ系の接着剤が良いでしょう．ケーブルは端

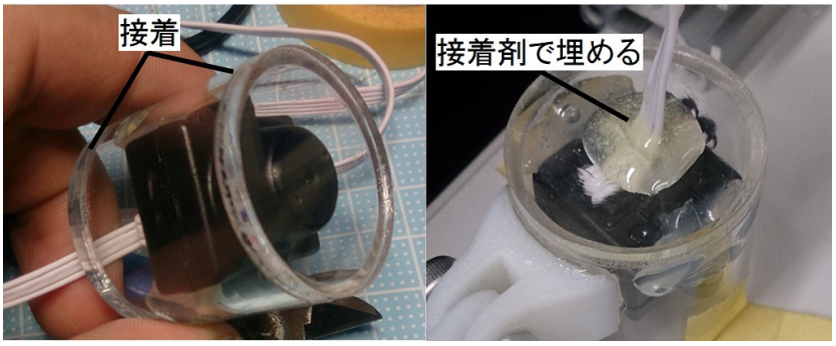


図 8.14　実際の接着

点部がアナログ出力となるようにしておけば，AV 入力機器を PC につなぐことでソフトウェアから映像を見ることができます．（相性によって映らないこともありますが，全ての製品のチェックは不可能ですのでご了承下さい．Web カメラなども使えます．）AV 入力機器を PC に接続したら，ソフトを起動して右下の Camera Unit のタブを押し，ユニットを選択します．選択すると connect ボタンが押せるように

なるので押して下さい．

図 8.15　ソフトの操作

湖の生物を探してみよう

カメラさえ出来てしまえば，色々なところに出掛けて水中を観察することが出来ます．例えば，下の画像は筆者が長野県女神湖でこのキットを用いて撮影テストを行った時のモノです．この時は日中で浅い位置で撮影したので写真が取れていますが，もっと深く迄行こうと考えるならば，ライトなどの搭載も必要かもしれません．溜池などで活動をするときは管理者に必ず許可を取り，安全を確保して行いましょう．

女神湖での撮影の様子

8.4　実際に改造をしよう（ソフトウェア編）

8.4.1　ソフトウェアの詳細

ここでは改造した機体を思い通りに動かすためのプログラムについて説明します．マイコンと PC の接続については 6 章にある図 6.1 を見てもらえればと思います．PC のソフトは，コントローラのスティックとボタンのすべての情報を読み込み，シリアル通信を利用してマイコンに送信しています．よって，マイコン側のプログラムを変更することで，コントローラの情報に対して機体の動作を自由に変更することができるようになります．マイコンのプログラムは CD の Arduino フォルダに有るので，ソフトをインストールして開いてみてください．（Arduino の使い方については今回は説明を省略します．各自調べてみてください）[*6]Arduino のプログラムは以下のフローチャートの様な構成になっています．

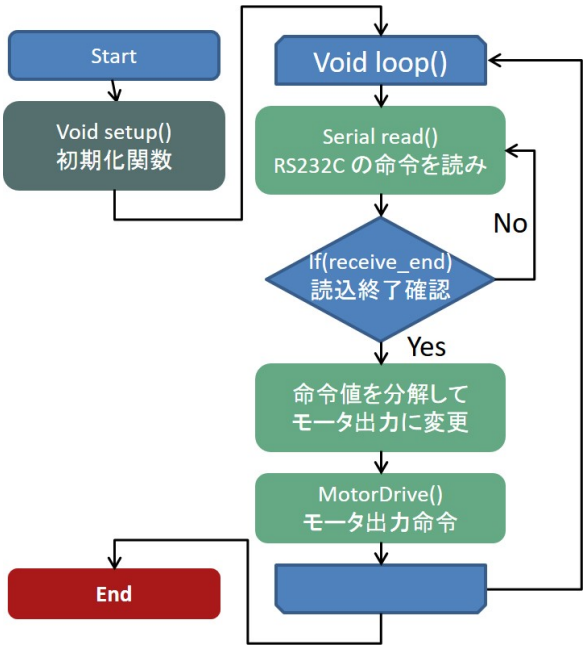


図 8.16　プログラムのフロー

以下にブロックを 1 つずつ説明をしていきます．

setup 関数

ここでは，入出力ピンの設定と，シリアル通信の速度を設定しています．A1〜A4 は正逆回転の入力，10〜14 ピンは PWM（速度入力）を担当しています．Serial1 は基板上のシリアルユニットを用いて通信するという設定です．Arduino 上の USB を用いてシリアル通信を行いる場合は，これ以降の Serial1 を Serial と書き換えるとシリアル変換ユニットを用いずに USB からシリアル通信ができるようになります．

[*6] 参考 ArduinoMicro の書き込み方　http://exiii-hackberry.com/forums/topic/arduino-micro にプログラムの書き込み方？/

loop 関数

この関数内を繰り返し処理することで，機体が動作しています．はじめに SerialRead 関数を呼び出して，ソフトからの通信情報を読み込んでいます．SerialRead 関数は chkstr の配列に，コントローラの情報を読み込みます．情報は下のコントローラの図 8.17 に示すような指定となっていて，a が先 p が後となり，数字が若い順に格納されています．chkstr［］=a1,a2,a3,a4,p1,p2,p3,p4.... といった具合です．p についてはコントローラーソフト左下に有るボタンがコントローラーの押したボタンに対応するものが反応するようになっているのでそれを参考にしてください．

図 8.17　アナログバーの配置

SerialRead 関数がすべての情報を引き出すと，recieve __ end の条件が正となり，コントローラの情報を元に以下の条件文の動作を行います．ここから MotorDrive の関数を呼び出すところまでは，左軸の a1,a2 を縦横の座標にとって象限を分け，それに合わせた動作を規定しています．Cont __ threshold はプログラムの頭で値を指定しているマージンです．このマージンはコントローラのアナログスティックの値が原点周りでぶれてしまったり，人の指の癖としてまっすぐに動かすことが難しいことから余裕を取るためのものです．

SerialRead 関数

SerialRead 関数は，PC からのシリアル通信を受け取る関数です．PC からは，1 文字単位でデータが送られてくるので，これが送られてくるたびに，RecieveData 配列に格納していきます．RecieveData 配列が，本来送られてこない文字数 50 字ないし，終了文字である”￥ n ”を受けた時，これを文字列から int 型の整数に変換して chkstr 配列に格納します．

MotorDrive 関数

MotorDrive 関数は motornom,motorspeed の 2 つの引数を伴って呼び出されます．motrornom とモータの対応は表のようになっています．回転数は motorspeed によって± 254 の範囲内で推力の指定が可能です．関数内ではこれを正負で場合分けして，デジタル出力ピンの組み合わせと PWM 出力でその速度の決定を行っています．

8.4.2　プログラムを書き換えてみる

前章に挙げたスラスタ配置の変更を行った時，今のコントローラとスラスタの対応では動かないでしょう．例えばスラスタ配置を図 8.18 のようにホバリング形態に変更した場合，今のコントロール方法では，前後進と上下運動が連動してしまいます．こういった場合，動作を割り振り直す必要がありますが，何も

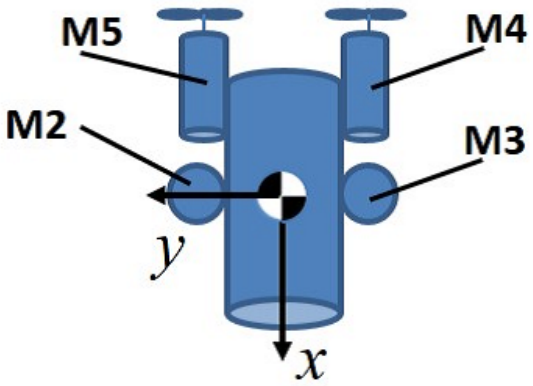


図 8.18　ホバリング型の推力配置

考えずに対応を決めることは出来ないでしょう．よって，スラスタの動作とコントローラの動きをどう対応させるか考える必要があります．具体的には独立と連動の関係を考えます．例えば，はじめに組んだこの機体のスラスタ配置 6.8 では，動作と 4 つのスラスタの出力する推力の関係は以下の表のようになります．なお，ここでの M2〜M5 は各モータを指し，各動作の部分にかかれているアルファベットは推力を

スラスタ＼動作	前進	ピッチ回転	ヨー回転
M2	A	A	A
M3	A	A	M2>M3 (M2=M5, M3=M4)
M4	A	M2>M5 (M2=M3, M4=M5)	M5>M4 (M2=M5, M3=M4)
M5	A	M3>M4 (M2=M3, M4=M5)	A

図 8.19　クルーズ型の推力関係

指します．例えば，この表では前進をする場合 M2〜M5 までのモータは全て前進方向の推力を同じ A という大きさで発揮していなければならないということです．また条件式は各モータの推力関係がこれを満たさないと所望の動作を行えないという意味です．取れる値はプログラムに合わせて 254〜-254 です．前後進であれば全ての推力が同じ値で有ることが求められますし，ピッチ動作を取る場合は M2 と M3 が M4 と M5 の推力を上回っていなければいけません．この表を見れば分かる通り，Mark3 の基本的なスラスタ配置では，全てのスラスタの推力が設定通りに動作しなければ全ての動作を達成することが出来ません．よって，これらの関係は連動下にあるといえます．一方で図 8.18 のホバリングであれば以下のような表になるはずです．

スラスタ＼動作	前進	潜行動作	ヨー回転
M2	-	B	-
M3	-	B	-
M4	A	-	A
M5	A	-	M4＞M5

図 8.20　ホバリング型の推力関係

ここを一見して分かる通り，潜航のためのスラスタと移動のためのスラスタは別々に動かすことができます．つまり，潜航動作のスラスタと，移動のためのスラスタは独立であるといます．但し，潜航 2 つ・移動 2 つは動作に条件がありますから，連動下であるといえますね．

このように条件が表にできれば次はコントローラとの関係を決めていきます．ここで，独立の関係に有る動作は，独立なコントローラスティックに割り振ると良いと思います．例えば，今回の例で言えば左スティックを移動，右スティックを潜航動作に割り当てて考えましょう．（当然ボタンに割り振るという方法もありますが，見妙な調整が必要ならばアナログスティックのほうが使いやすいと思います．）次に，アナログスティックの動作と機体の動作の対応を細かく決めていきます．例えば，アナログスティックでとり得る値を座標軸に落として動作を決めていきます．ここで，若干の余裕を採って決めてある前後進と左右旋回の部分以外をどのような推力配分にしていくかが腕の見せどころです．[*7]例えば，旋回方向軸に倒した量を旋回方向のスラスタ推力から引く形に設定する方法があります．アナログスティック a1,a2 の値が（x，y）=(-50，250) の時，スラスタの推力は左が 200，右が 250 となります．これならばどの推力も最大で 254 の値をとることが可能になります．推力の決定式ができれば後はプログラムの形に書き下すだけですね．

8.4.3　アクチュエータを増設する

本機は追加で 2 基の DC モータを追加することが可能です．また，Arduino の PWM ピンの条件をクリアすれば RC サーボモータの接続も可能な仕様となっています．独立に動作するモータが必要な場合は，予備のモータドライバ基盤を作成して接続し，動作プログラムを追記することで動作させることが出来ます．追加する場合は，以下の作業が必要です．

setup 関数に出力をプリセットする

setup 関数内にモータ出力を行うピンをデジタル出力で設定します．追加するモータ，M1 と M6 は以下のように追記してください．

```
pinMode(17,OUTPUT);//Motor1
pinMode(A0,OUTPUT);//Motor1
pinMode(15,OUTPUT);//Motor6
pinMode(A5,OUTPUT);//Motor6
```

[*7] なれてくると，こういった座標軸から条件わけをせずに，少ない式で条件を決められるようになります．そうなるためには，動かし方のルールをいかにしてわかりやすく捉えるかが重要になります．

MotorDrive 関数に動作条件を追記する

MotorDrive 関数に動作条件を追記します．setup 関数と同様に MotorDrive 関数内の switch 文の中に以下の様なプログラムを追記して下さい．

```
    case 活用にはバイパスが必要1://
      if(motorspeed <= 0){
        digitalWrite(A0,LOW);
        analogWrite(13,(-1) * motorspeed);
      }else{
        digitalWrite(A0,HIGH);
        analogWrite(13,(-1) * motorspeed);
      }
      break;

    case 6:
      if(motorspeed <= 0){
        digitalWrite(A5,LOW);
        analogWrite(11,(-1) * motorspeed);
      }else{
        digitalWrite(A5,HIGH);
        analogWrite(11,(-1) * motorspeed);
      }
      break;
```

loop 関数に動作をプログラムする

loop 関数内のプログラムに関しては自由に内容を追記して下さい．基本的には前節のコントローラとモータの関係の部分を参考にすれば簡単に作成できると思います．参考に，ボタン b01 を押している間，Motor2 を一方方向に回転するプログラムを下記に示します．

```
if(chkstr[4] < 0){       //がオンになっているとき，カッコ内のプログラムが動くb01
                MotorDrive(2,250);
}
```

回路を作成する

回路の作成は，5 章を参考に作成して下さい．ジャンパは下記に再掲する表のとおりです．また，マイ

対応モータ	対応ジャンパA	対応ジャンパB
M1	JP1	JP2
M2	JP3	JP4
M3	JP5	JP6
M4	JP7	JP8
M5	JP9	JP10
M6	JP11	JP12

図 8.21　ジャンパ接続表

コンボード上に図に示す白配線のジャンパを行って下さい．

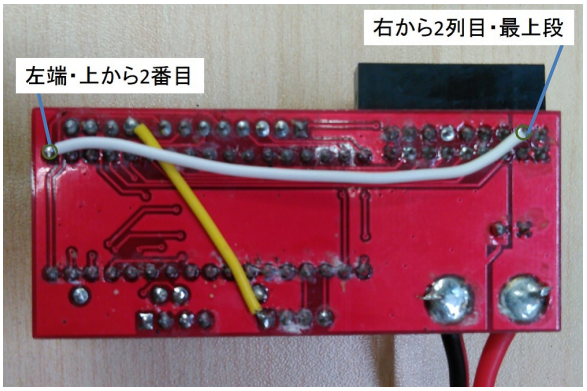


図 8.22　ジャンパ配線

DC モータ以外を接続する

マイコンボードで空いているピンは図 8.23 の部分に集約されています．Arduino のピンは下表の様にターミナルと接続されていますので，使用を希望するラインに接続しましょう．例えば RC サーボを使うならば，電源とグラウンドを接続し，サーボ信号は 8P コネクタの 5pin や 6pin などで Servo 関数を利用することで動作させることができます．（それによって一部の PWM 機能が止まる場合があるので注意して下さい．）

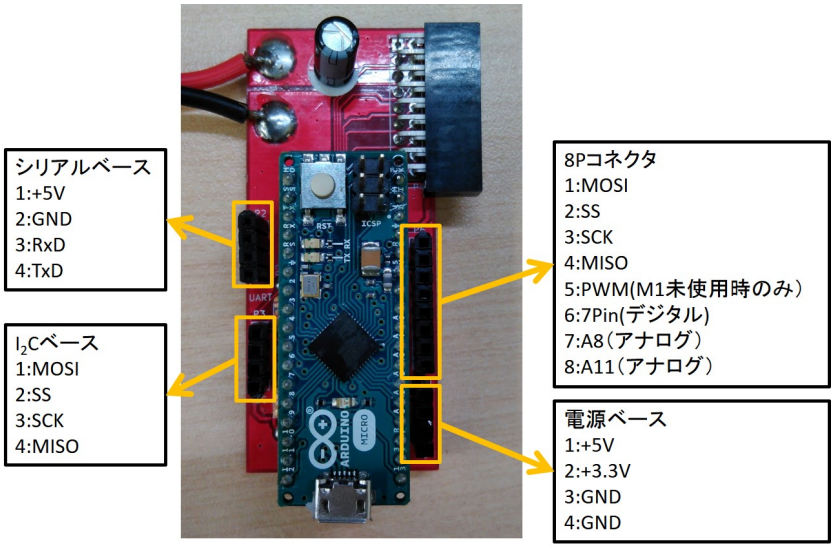


図 8.23　マザーボード配線表

8.5　センサを搭載してみよう

8.5.1　内界センサと外界センサ

最後にセンサについて簡単に触れていきたいと思います．センサは人間に言う感覚器のようなものです．本機は初期状態でセンサを一切搭載していませんからロボットは自分の状態を一切しることが出来ません．例えば，エンコーダやポテンショでリンクやモータの回転数・回転角を検知したり，加速度センサとジャイロセンサを併用することで現在の姿勢をしることができます．これらは，ロボット自身の状態をしる要素になるので「内界センサ」と呼ばれます．人間で言えば，三半規管や頭痛といった外からではない感覚を感じるような神経です．[*8]内界センサもあれば外界センサもあります．これは，外を見るためのもので，例えば距離センサで機体と壁の距離を測ったり，圧力センサで水圧を測定して今の深度を測ったりするわけです．人間に例えると，肌の触覚や，視覚等が代表例でしょう．センサの測定した値の多くは，アナログ電圧値によって返されますので，マイコンボードの 8p コネクタのアナログ入力に接続して取得することが出来ます．また，多機能なセンサ，例えば加速度とジャイロセンサが両方乗っているようなものだと，シリアル通信によって読み込む必要があります．このマイコンボードでは，I2C の接続端子を予め備えていますから，これを使ってセンサの値を取ることが可能です．[*9]センサの活用例は多く存在しますから，検索してみて下さい．

今回は例として，マグネットスイッチ（リードスイッチ）と加速度センサについて取り上げたいと思います．

マグネットスイッチ

マグネットスイッチは，磁力を検知して ON/OFF を切り替えるスイッチです．例えば，リレーと組み合わせることで，耐圧殻を開けずにバッテリの電源を落としたり，動作プログラムを変更することができます．他にも，蝶番などを利用してちょっとした仕組みを作れば，マグネットで何かキャッチしたときに反応する仕組みを造ることもできます．マグネットスイッチはデジタル入出力なので，8P コネクタに接続して利用できます．

加速度センサ

加速度センサやジャイロセンサのようなセンサには，上述のようにいろいろな通信方法がありますが，今回はアナログ出力のものを使います．例えば，これは秋月電子で販売されている”KXR94-2050”です．これは 5V の入力を受けて，最大で± 2G の重力加速度を測定することができます．重力加速度は常に地面に向かって 1 G かかっていますから，センサーに有る三方向の情報を利用すれば機体がどの方向に傾いているかを調べることができます．具体的には以下のような形です．図 8.24 のように定義した座標系について，センサが傾いた状態を考えます．センサが加速していない状態であれば，センサが取得している三方向のベクトルによって発生しているベクトルの大きさは 1 G のはずですね？ということは，三角関数を使ってやれば以下のような関係が取得できるはずです．[*10]ちなみに，前提として機体が加速していないことを定義していましたが，加速中でも姿勢を取るためにはジャイロセンサを併用する方法があ

[*8] 気圧変化で頭痛が起こるだろ！と指摘できる人，よくわかってる．人間の感覚器は内界センサと外界センサを両立する物が多いです．

[*9] I2C でマイコン同士をつないでさらにモータなどを増やすこともできますね．

[*10] 悩む人は，三軸が直行しているということに着目して考えればよいかと思います．

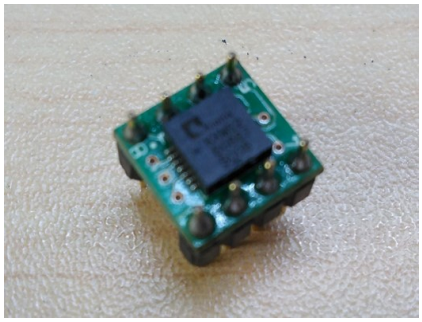


図 8.24　KXR94-2520

ります．ジャイロセンサの場合は，回転角速度を取得できますから回転角速度とその時間をかけ合わせれば元の角度からどれくらい姿勢が変化したか予測することができますね．

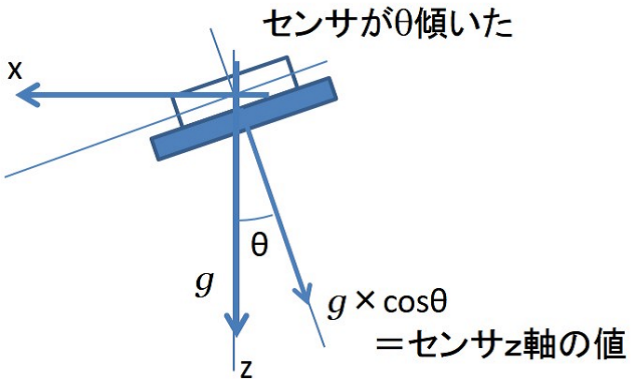


図 8.25　加速度センサで角度を取得する

8.6　完成品を評価しよう

さて，実際に改良が終わったら，これが成立しているか確認してみましょう．この章の頭で，改良には目標設定が必要であることは説明しましたね．つまり，目標が明確に設定されているわけですからこれが達成されているかを確認すれば良いわけです．測定の方法は実験内容によって異なりますから，ここでは細かい内容を控えますが中学や高校で行う理科の実験をイメージすれば確認したいことを確認する方法が想像できるのではないかと思います．むしろもっと難しいことは何か，それは「目標を達していなかったときの原因を探る方法」です．なぜならば目標が達成されなかったとき，それは改良に当たっての前提が間違っていることを示しているということです．つまり自分たちにとってそれが当然であると思っていることを疑う行為であるわけですから，それは簡単にはいきません．私はこの前提が間違っているという問題に対して 3 つの観点が有ると思っています．次の節からこれについて個別に検討しましょう．

8.6.1　追加した機能が要求を満たしていない

これは追加した個別の改良点を評価すればすぐにわかりますね．モーターのパワーが予測より足らないとか開発初期段階のものの歴史を紐解くと意外とよくあったりするんです．[*11]そもそも足らないということであれば，それを満たすものに交換することで問題を解決しましょう．ただし，実は 3 節の内容が原因である場合もありますから注意が必要です．

8.6.2　着目する対象が間違っている

この問題もそれほど難しくはありませんが，前節よりは検討するべきポイントが難しくなります．例えばさっきの車の図をまた引き合いに出します．ここで，車を高速化したいと考えたとき，本来ならば，タ

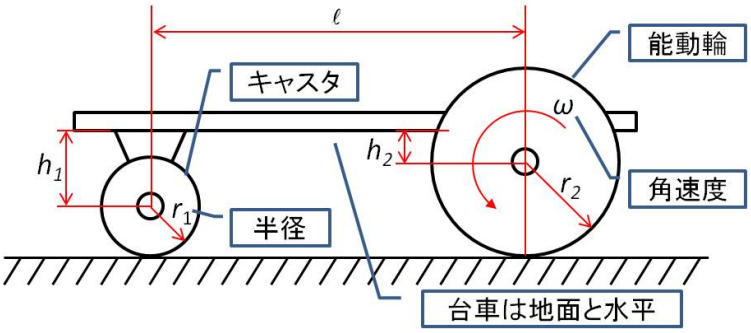


図 8.26　タイヤロボット

イヤの直径と回転数が要因になると思うでしょう．しかし，これがトルクが主要因で有ると思ってしまっていればどうなるでしょう．力強い走行をすることはできても，必ずしも高速にすることはできないでしょう．このように，原因と結果の関係が実質的に一対一であったとしても複数存在する原因になりうるものの中で違うポイントに着目してしまった場合はうまくいかないわけです．

8.6.3　気づかない相互作用がある

さて，ここまでの節は原因と結果が一対一の関係であることが前提になっていましたが，今回はそうではない場合です．例えば，しつこいようですが車の図で考えます．前節では車のトルクではなくタイヤの回転数と直径によると書きました．確かにどんな状況でも十分に発揮されることが前提であればこれは成立します．しかし，例えばモータの場合，一般的には回転数を挙げるとトルクが下がるという関係性があります．これに気づかなかった場合，回転数と直径を最大化しようとするとトルクが足らなくなって実際には走らなかったりむしろ速度が遅くなってしまったりすることが有るわけです．[*12]この関係性をいかに見出すかがある意味研究活動の最も面白いところであるわけですが，今回はここまでとしたいと思います．皆さんの技術を活かして良い機体が製作されることを祈ります．

[*11] 意外とよくあるからって OK って言ってるんじゃないよ

[*12] 例えばスクリューとかも静止状態と移動状態では出せる力が大きく変わるよ．周囲の流れとの関係性が有るからね

謝辞

本マニュアルの執筆公開には多くの方のご尽力，ご協力をいただきました．下記にご協力いただいた皆様の名前を挙げるとともにここに謝意を表すものであります．（この謝辞は 17 年度版のものであり今後継続的に増加します．）

・阿久津悠様
　水中ロボコン’13 にチームとして参加，その観察眼の鋭さと無言のサポートで Mark3 の受賞に貢献してくれました．彼の協力がなければ，このような結果を得ることはなかったと思っています．

・茂木雄一様
　Mark3 に関連したほぼすべての展示，大会運営において現在進行系で尽力してくれています．彼の細やかな配慮と協力があって始めて大会の運営を可能としています．

・日笠絢乃様
　本機のイメージキャライラストを描いて頂きました．思えば Mark3 の運用が上手くいき始めたのはこの頃で，イメージキャラの効果とそれを可能にした彼女のセンスには脱帽するしかありません．

・森田寿郎准教授
　工学的内容から乖離している本企画について，その理念にご理解と賛同を頂いたおかげでこの成功があります．感謝しております．

・松山工業株式会社の皆様
　シリコーングリスの供給とゴムパッキンの制作についてご尽力頂きました．また，今回の防水耐圧殻の量産については，社長である鵜久森様に多大な尽力を頂きました．鵜久森様からの協力の申し出がなければ今回の耐圧殻新造は不可能だったでしょう．

・西川精機製作所の皆様
　特に西川社長には耐圧殻の制作に関し非常に多くのご助言を頂いただきました．また，製作所の皆様には厳しい納期の中でクオリティの高い耐圧殻の制作を行っていただきました．感謝しております．

・杉浦機械設計事務所様
　杉浦様には水中ロボコン'16 で Mark3 の 3D プリンタパーツを供給していただきました. また, 経験に裏打ちされた豊富な知識でサポートしていただきました．ありがとうございました.

・アクアモデラーズミーティングの皆様
　多くの知識を教えていただきました．皆様の知識が本機のベースとして行きています. 特に, 平尾昌弘様と平田様には機体作成及び大会運用のノウハウを多くご教授頂きました．感謝申し上げます.

・慶應義塾大学理工学部機械工学科森田研究室の皆さん
　皆さんの協力と部屋を汚し続ける私への妥協なくしてこの企画の成功はありませんでした．ありがとうございました.

　なお, 水中ロボコン'17 における本マニュアルとキット提供は日本財団様からの助成金を用いて行われました．また，Mark3 のテストは国立研究開発法人海洋研究開発機構の多目的プールを使用させていただきました.
ここに謝意を示させていただきます.