

目次

第 1 章	はじめに	2
1.1	研究の背景	2
1.2	研究の目的	3
第 2 章	メカナムホイールを採用した理由	4
2.1	二輪駆動ロボとメカナムロボの比較	4
2.2	三分間で必要とされる動き	5
第 3 章	メカナムホイールの説明	9
3.1	メカナムホイールについて	9
3.2	ホイールの向きと組み合わせ	10
3.3	各動作	10
3.4	操作プログラム	11
第 4 章	今回製作したロボットのメカナムホイールの問題点	14
4.1	問題点	14
4.2	原因	14
4.3	方向自動修正	16
第 5 章	おわりに	18
	参考文献	19
	謝辞	20

第 1 章

はじめに

1.1 研究の背景

1.1.1 2016 年度「ロボット・ニューフロンティア」ルール説明

2016 年度に行われた NHK 高専ロボコンは「ロボット・ニューフロンティア」という題目で港町から海を渡り、新大陸へと目指すというものである。使用ロボット台数は無制限、港町で灯台を組み上げ、地に接触しないよう海を渡り、丘の上に砦を築く競技で今年は高さを競い合うものであった。図 1.1 はこの競技のフィールドである。

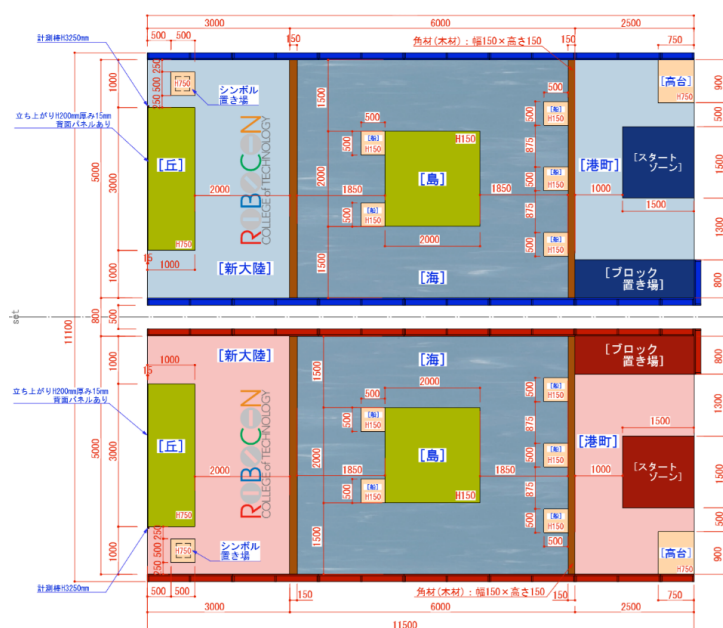


図 1.1 競技フィールド

1.2 研究の目的

今回ロボコン競技に使用した積み込みロボット移動手段となっている全方向移動可能なメカナムホイールを使用にジャイロセンサを加えることで継続的に直進させる。

第2章

メカナムホイールを採用した理由

何故、今回のロボコンでメカナムホイールを採用したかを以下に示す。

- 出題された多くの課題の中で制限時間3分は非常に短く、丘に箱を積み上げるだけでも困難になる。
- メカナムホイールを使用することで2輪駆動よりも圧倒的に速く課題をクリアすることができる。

2.1 二輪駆動ロボとメカナムロボの比較

メカナムホイールは2輪駆動よりも少ない工程で動作が可能である。例えば図○のように右側に移動し正面を向きなおす動作をする。2輪駆動ロボットの場合は図○の通り、右旋回→前進→左旋回と少なくとも3工程必要となる。これがメカナムホイールの場合は横に並行移動の1工程だけで終わることができる。また、メカナムホイールは図○のように移動しながらの旋回も可能で1工程の時間の中で2工程分の動きも可能なのである。このように、メカナムホイールを用いることで2輪駆動よりも少ない工程数で時間の短縮が可能なのである。

2.2 三分間で必要とされる動き

メカナムホイールを使っている積み込みロボットが競技中の三分間で必要とされる動きを以下に示す。

2.2.1 箱の持ち運び

特定の位置に並べられている箱を持ち上げ、距離 4 m ほど運搬する必要がある。この時の課題は箱を拾う際の位置調整である。箱は図 2.1 のように 400mm × 300mm × 200mm で 100mm の等間隔に設置されている。その 100mm の間を図 2.2 のように幅約 72mm のハンドを滑り込ませ、更に側面から見た図 2.3 のように縦中心で横が先端から約 110mm 前後の位置にハンドの中心がくるように設置しなければならない。(そのためメカナムホイールの全方向移動の微調整が 2 輪駆動より有効となる。)

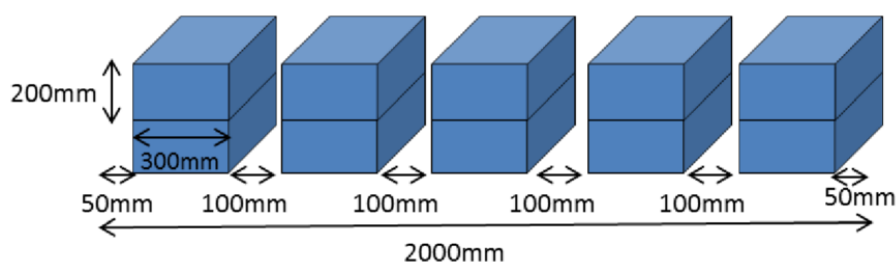


図 2.1 競技中の箱配置

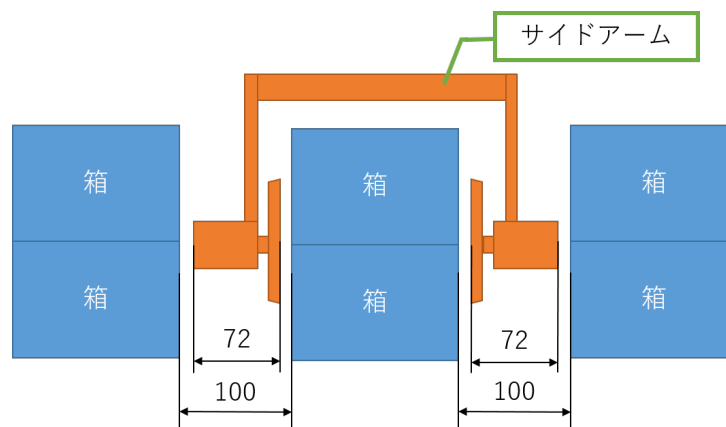


図 2.2 箱の幅とアームの幅

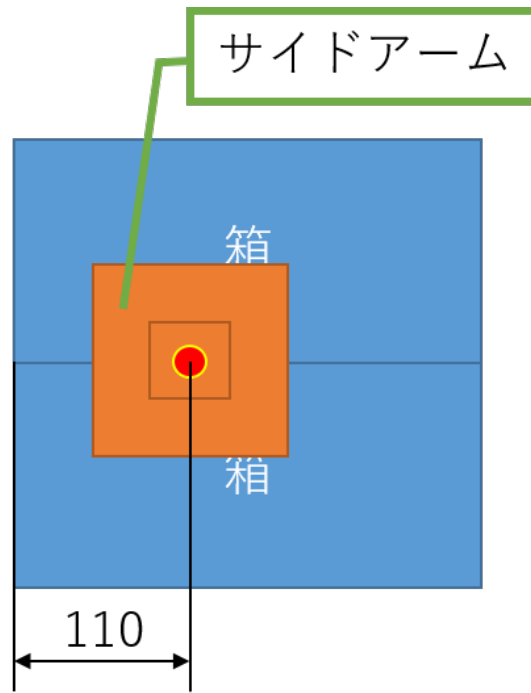


図 2.3 箱回収側面位置

2.2.2 塔建て

運んだ箱を高さ 750mm の台の上に持ち上げて、下段が大きく上段が小さくピラミッド型のように組み立てる。これは「高台」と「丘」によって異なり、「高台」では図 2.4 のように 2 列 1 列 1 列の順に組み立てる。「丘」では図 2.5 のように下段が上段よりも両側 25mm 程大きければいくらかでも積んで良いとなっている。製作した積み込みロボットは「丘」であれば動かずに積み上げることができる。だが「高台」の場合は 1 列に 2 段積み上げる必要があるため真横に並行移動できると時間短縮になる。

2.2.3 橋渡し

架橋ロボが設置した橋の上を渡るが、その橋に侵入する際に橋幅約 80mm でタイヤ 2 輪同時侵入の位置調整は難しく、2 輪駆動だと時間が掛かる。そのため、並行移動できるメカナムホイールは 2 輪駆動よりスムーズに位置調整が可能である。

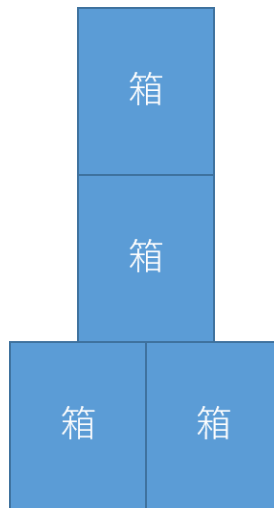


図 2.4 高台の箱積

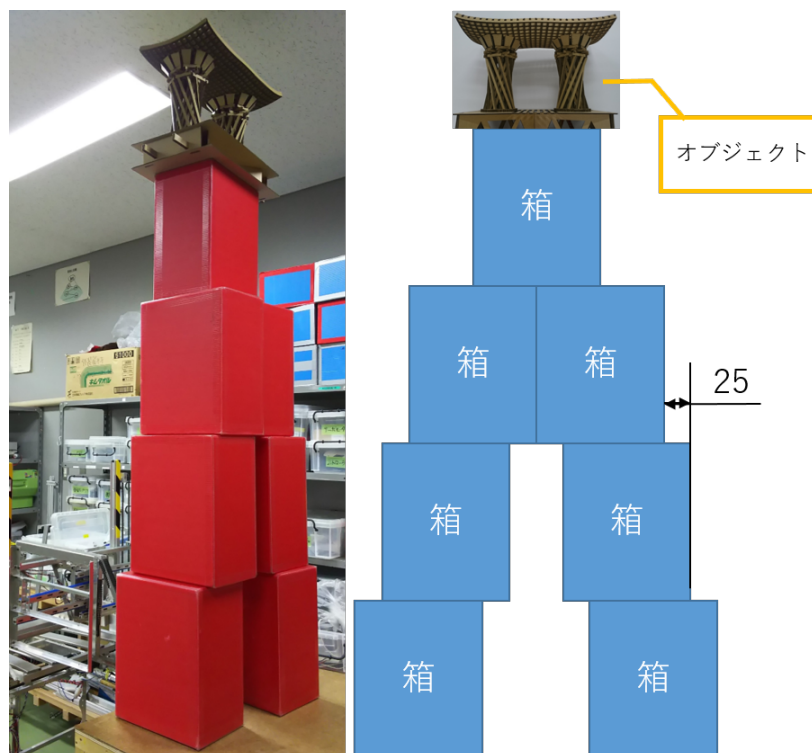


図 2.5 丘の箱積

2.2.4 オブジェクトの持ち運び

丘の左または右に設置されているオブジェクト置き場からオブジェクトの鼓門を持ち運び砦のてっぺんに設置する、この際2輪駆動では旋回を何度も必要となるため、並行移動ができるメカナムホイールが

有効的である。

第3章

メカナムホイールの説明

3.1 メカナムホイールについて

メカナムホイールとは車輪の円周面に回転軸から図 3.1 のように 45 度傾いた樽状のタイヤを一定の間隔を空けながら覆うようにして取り付け、各タイヤごとに任意の回転制御を与えることで任意の方向に移動することができる。

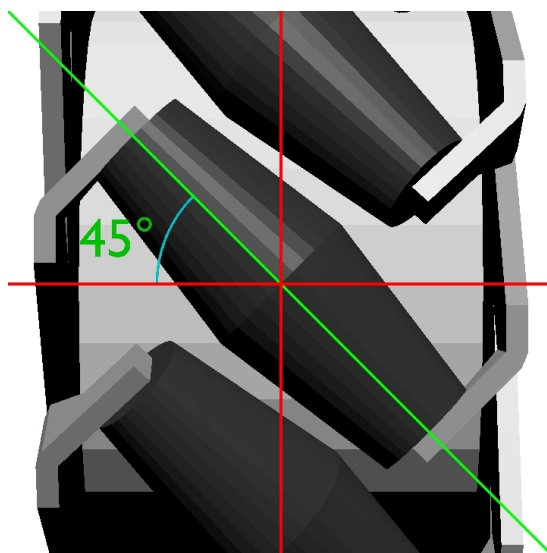


図 3.1 タイヤの角度

3.2 ホイールの向きと組み合わせ

メカナムホイールには順方向に傾いた車輪と逆方向に傾いた車輪の二種類があり、図 3.1 を順方向とする。これらを左右で 2 輪ずつの計 4 輪を図 3.2 のように取り付けることで正常に動作する。

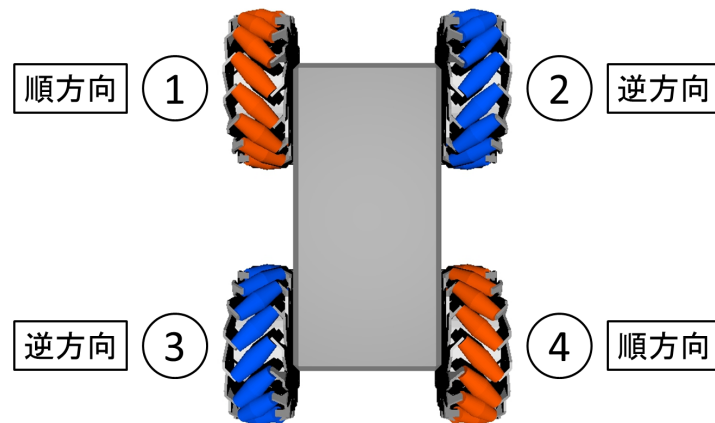


図 3.2 組み合わせ

3.3 各動作

ここではホイール単体の動作と四輪を組み合わせた全体の動作を説明する。

3.3.1 単体の動作

静止した状態では地面に接触しているタイヤの回転方向に進むが、回転時にはその逆方向に移動する。図 3.3 は地面から見たそれぞれの車輪の動きを示している。

3.3.2 全体の動作

四輪の状態での動作は車輪の回転によって発生する X 方向もしくは Y 方向のベクトルを互いに打ち消し合うことで、図??に示すように任意

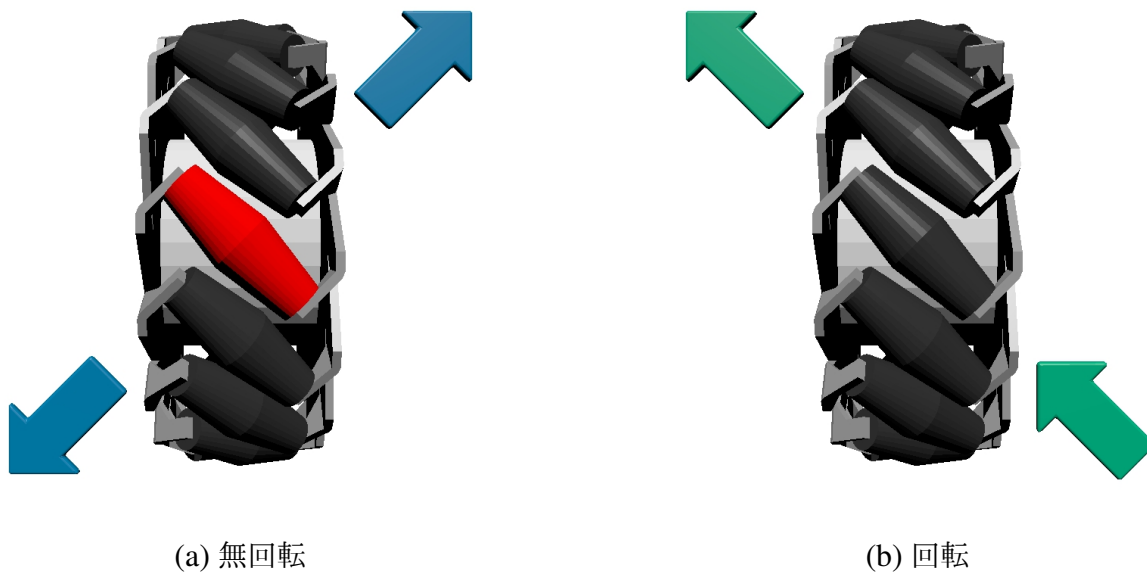


図 3.3 地面から見た車輪の動き

の方向に並行移動することができる。

3.4 操作プログラム

ここではメカナムホイールの回転数を制御する計算式や、そのために必要な定義などを説明する。

3.4.1 定義

メカナムホイールを操作するためには縦方向、横方向、回転方向の3種類の値が必要である。さらにこれらを計算してPWM値とするためアナログ値を出力できるコントローラでなければならない。そのため、これらの基準を満たしているDualShock3のアナログスティックを2つ用いて操作する。アナログスティックの初期位置を原点とし、そこからX方向とY方向に傾けることで、その量に応じた値が出力される。左スティックは移動用で倒した方向に並行移動することができる。右スティックは旋回用でこちらはX軸の値しか出力しない。

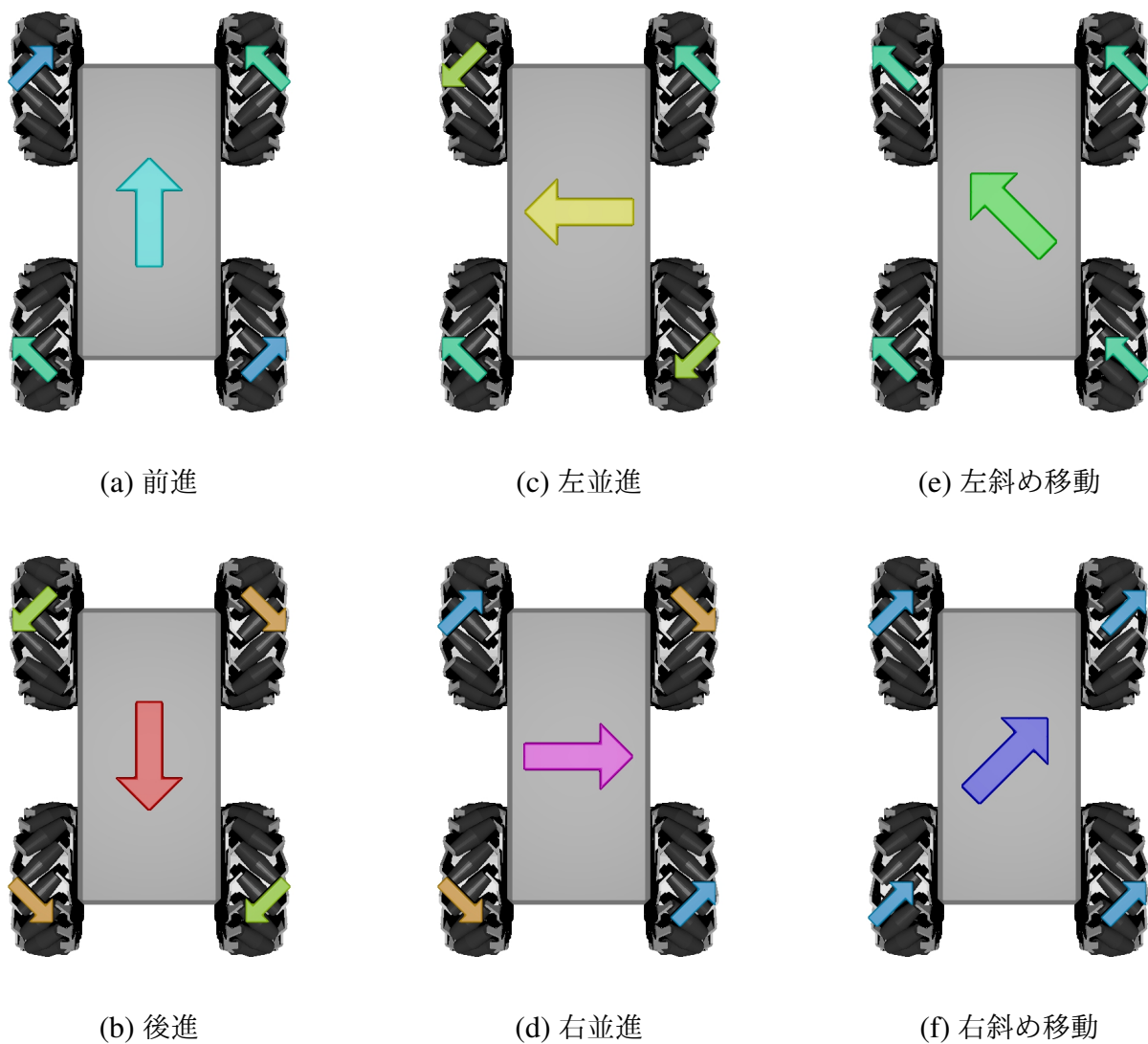


図 3.4 車輪のベクトルと本体の進行方向

3.4.2 計算式

左スティックはY方向に傾けることで L_y 、X方向に傾けることで L_x を出力する。右スティックはX方向に傾けることで R_x を出力する。モータのPWM値の計算式は以下のようになる。

$$M_1 = L_y + L_x + R_x \quad (3.1)$$

$$M_2 = L_y - L_x + R_x \quad (3.2)$$

$$M_3 = L_y - L_x + R_x \quad (3.3)$$

$$M_4 = L_y + L_x + R_x \quad (3.4)$$

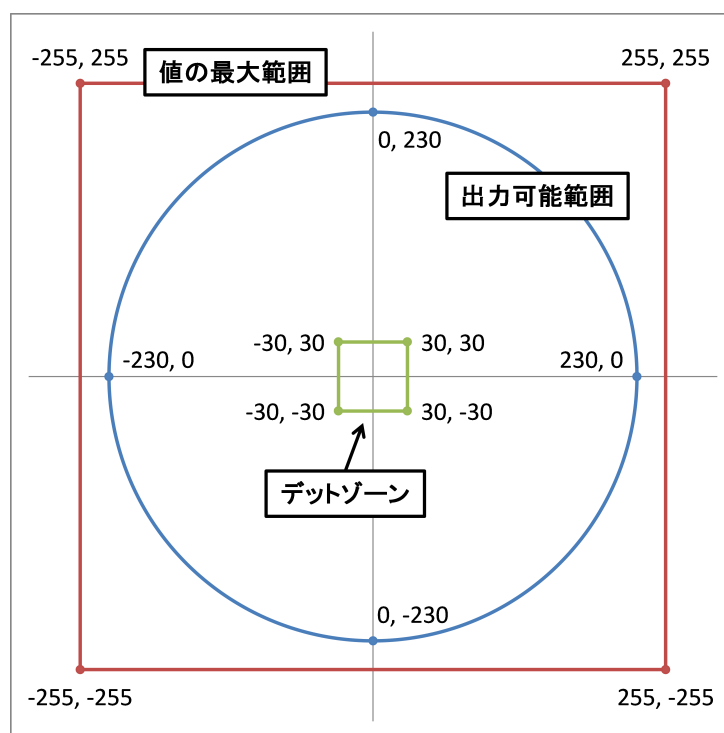


図 3.5 アナログスティックの座標

第 4 章

今回製作したロボットのメカナムホイールの問題点

4.1 問題点

- 直進中の環境によって進行方向がずれてしまい斜めに進んでしまう。
- 駆動開始時に右方向に大きくずれることがある。
- 停止時に慣性で右方向にずれる。

4.2 原因

4.2.1 項目

以下に 4.1 の問題点が発生すると思われる原因を示す。

- 1) ロボットの重心
- 2) 加工精度（タイヤの設置位置）
- 3) モーター個々の差
- 4) モータードライバの差

4.2.2 一番の原因と理由

この中で一番の原因となっているのは「ロボットの重心問題」である。理由は、図??に示すように積み込みロボットには、正面に箱を積み上げるためのラックハンド、片方の側面に箱を回収するためのサイドアームが取り付けられているため、図??に示すように重心の位置が中心よりも右上にずれている。これにより左後ろのタイヤが浮くことがあり、4輪全体が均等に床に力を伝えきれないことが問題なのである。

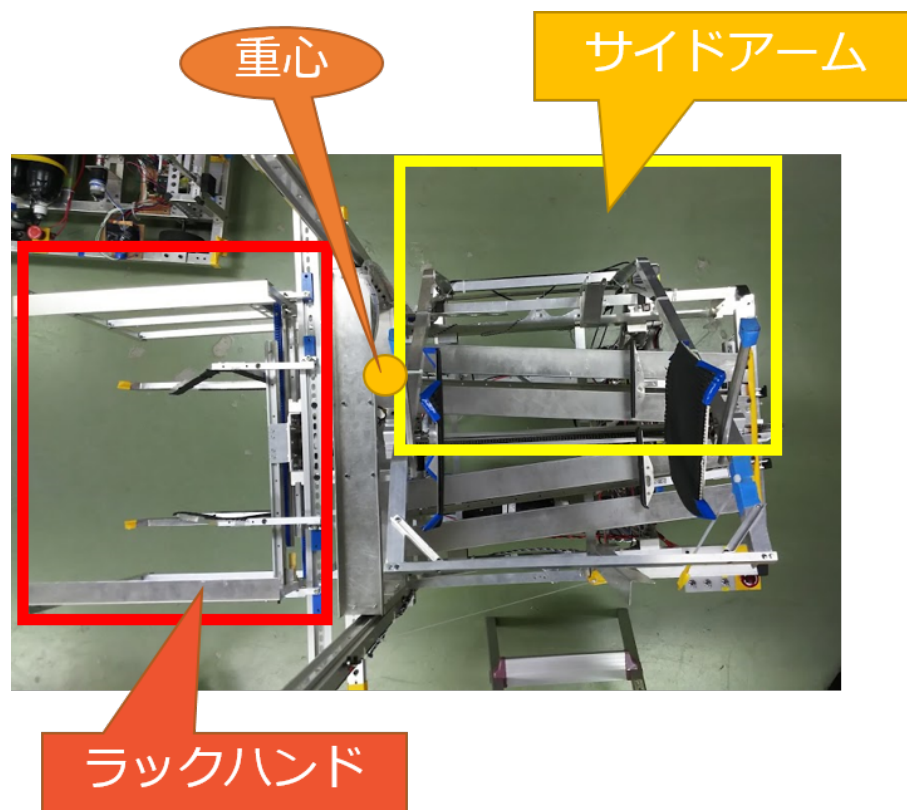


図 4.1 積み込みロボット重心

4.2.3 改善無しの直進実験

(失敗の例を表として示す予定)

4.3 方向自動修正

ロボットの重心が問題であるのならばカウンターウェイトを用いて重心を整えればいいのではないのかと考えられる。しかし、今回のロボコンの競技上、重量制限があるため制御面で改善することにした。そこで用いたのが図 4.2 の Arduino 9 軸 モーションシールドでその中に搭載されているジャイロセンサを使用し改善を行った。

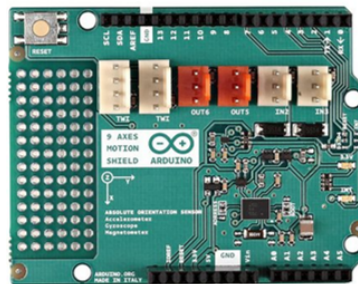


図 4.2 Arduino9 軸モーションシールド図

4.3.1 ジャイロセンサ

ジャイロセンサとは、角速度センサとも呼ばれ、角度が単位時間당たりにどれだけ変化しているのか計測できるセンサである。このセンサは身近にも用いられており、スマートフォンの加速度センサとの違い、加速度センサは重力方向によって左右上下の認識を行う。

4.3.2 どのような動きか（式に入れることも）

今回使用したプログラムでは $-180 \sim 180[\text{deg}]$ の角度を読み取ることが可能で $0[\text{deg}]$ からずれた分を補正值の PWM 値として式 3.1, 3.2, 3.3, 3.4 に err の形で代入し以下のような式となる。

$$M_1 = L_y + L_x + R_x + err \quad (4.1)$$

$$M_2 = L_y - L_x + R_x + err \quad (4.2)$$

$$M_3 = L_y - L_x + R_x + err \quad (4.3)$$

$$M_4 = L_y + L_x + R_x + err \quad (4.4)$$

このようにすることで Dualshock3 の左スティックを入力中も旋回の動作が自動で行われて自動方向修正が可能となる。

4.3.3 結果

(4.2.3 の直進実験からどのような変化があったか)

第 5 章

おわりに

メカナムホイールを用いることで容易にかつ素早く移動することができるが、ロボットの重心が中心から離れている場合、正しく移動することができない。それを改善するためには原因を修正するか、ジャイロセンサなどで補正する必要がある。

参考文献

[1] Arduino9AxesMotionLibrary,<http://www.arduino.org/learning/reference/9-axes-motion>,2017/01/16

閱 覽

謝辭