
『ロボットの製作（モータドライバボード）』資料
1.1 版

2014/9/25(木) 10:00～12:00 徳島大学 工学部知能情報工学科 C 棟 1F 実験室

辻 明典

目次

1. ロボットの構成	2
1.1. 基本構成	2
2. モータドライバボード	3
2.1. 基本構成	3
2.2. モータ用電源スイッチ	3
2.3. ARDUINO ピン割り当て	3
2.4. モータドライバ基板（表）	4
2.5. モータドライバ基板（裏）	5
3. 部品一覧	6
3.1. モータドライバボードの部品	6
4. 部品実装	11
4.1. 配付基板	11
4.2. 実装イメージ	12
4.3. サーボモータの取り付け	13
5. サーボモータの改造（改造済み）	15
5.1. ストッパーの除去	15
5.2. ポテンショメータ配線の除去	16
6. サーボモータのキャリブレーション	18

6.1.	サーボモータの動作原理.....	18
6.2.	キャリブレーション.....	19
6.3.	ARDUINO によるサーボモータのキャリブレーション.....	20
7.	付録.....	21
8.	改編履歴.....	23

1. ロボットの構成

1.1. 基本構成

ロボットは，Arduino 互換（Arduino Leonardo）ボードとモータドライバボードの 2 枚より構成される．

(1) Arduino 互換ボード

- ① ATmega 32U4 マイコン(AVR 社)
- ② フルカラーLED（1 線シリアル接続）
- ③ ブザー（圧電素子）
- ④ 赤外線受信センサ（リモコン）
- ⑤ Arduino 互換拡張ソケット
- ⑥ ZigBee 無線モジュール XBee（オプション）

(2) モータドライバボード

- ① サーボモータ（GWS 社 PICO/STD/F）
※ 連続回転サーボに改造する必要がある．（5 章参照）
- ② 電池ソケット（単 3×2 本：ニッケル水素充電電池またはアルカリ乾電池）
- ③ フォトインタラプタ
- ④ 車輪
- ⑤ キャスター
- ⑥ 高効率 DCDC コンバータ（5V 出力）

2. モータドライバボード

2.1. 基本構成

モータドライバボードには、サーボモータ 2 個、電池ソケット、フォトインタラプタが実装される。高効率昇圧型 DCDC コンバータにより、Arduino 互換ボード、サーボモータへの電源供給を行う。Arduino 互換ボードとは 10 ピンコネクタによりフラットケーブルで接続される。

2.2. モータ用電源スイッチ

モータへの電源供給は、電源スイッチにより行う。

電源スイッチの状態	サーボモータの電源	注釈
ON	モータ電源供給	USB から電源供給の場合、モータ駆動の電流が足りないことがある。
OFF	モータ電源切断	プログラム書き込み中は、モータ用電源スイッチを「OFF」にする。

※ バッテリィ動作時、電源電圧がサーボモータの回転開始に低下するがマイコンの動作に影響はない。

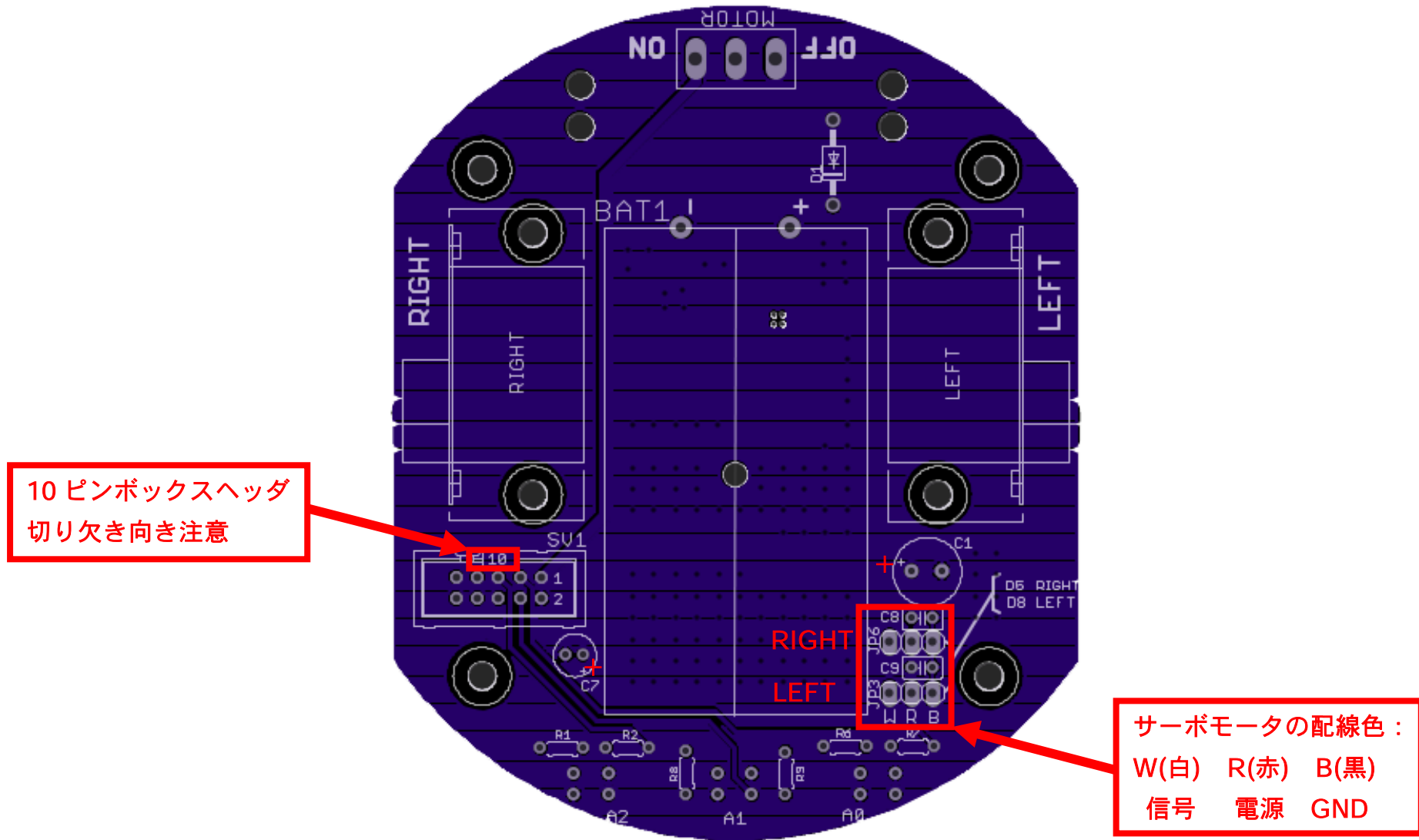
※ サーボモータの消費電流は約 300mA (150mA/個)

2.3. Arduino ピン割り当て

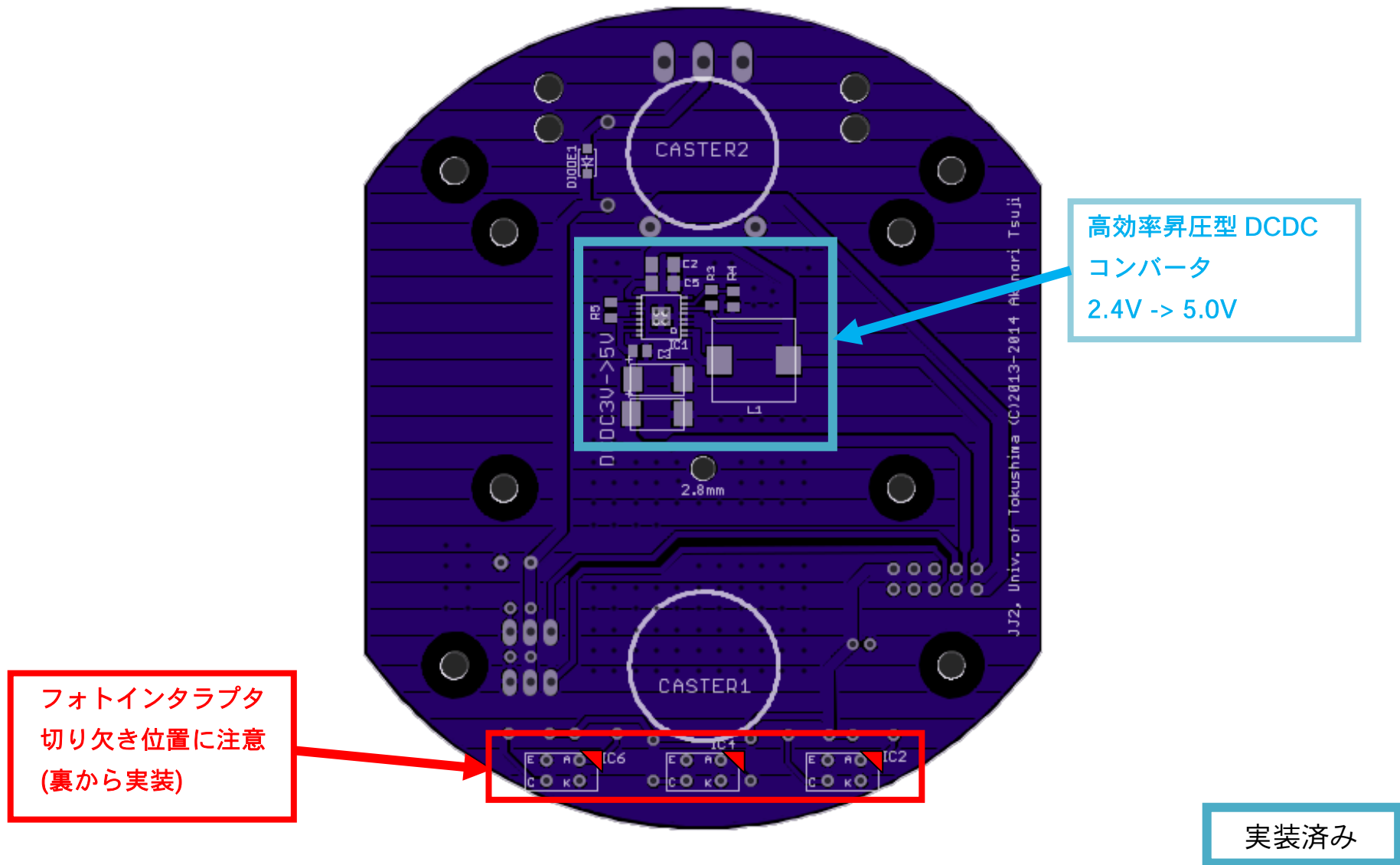
Arduino ピン	機能	注釈
D5	サーボモータ (右)	Servo ライブラリ使用
D9	サーボモータ (左)	Servo ライブラリ使用
A0 (左), A1 (中央), A2 (右)	フォトインタラプタ	AnalogRead を使用

※ Arduino 用拡張シールドを使用する場合、上記の端子は使用できないので接続前に確認を行うこと。

2.4. モータドライバ基板（表）



2.5. モータドライバ基板 (裏)



3. 部品一覧

3.1. モータドライバボードの部品

(1) 未実装部品

✓	実装順	名前	値	パッケージ	内容
	1	R1	330(橙橙茶)	0204/5	抵抗(1/6W)
	2	R6	330(橙橙茶)	0204/5	抵抗(1/6W)
	3	R8	330(橙橙茶)	0204/5	抵抗(1/6W)
	4	R9	5.1k(緑茶赤)	0204/5	抵抗(1/6W)
	5	R2	5.1k(緑茶赤)	0204/5	抵抗(1/6W)
	6	R7	5.1k(緑茶赤)	0204/5	抵抗(1/6W)
	7	SW1	CS12AAP1	CS12AAP1	モータ用電源スイッチ
	8	JP3	1x03	1X03	3ピンシングルヘッダ (サーボモータ左)
	9	JP6	1x03	1X03	3ピンシングルヘッダ (サーボモータ右)
	10	C7	100uF / 35V	E2-5	アルミ電解コンデンサ ※極性有
	11	C1	470uF / 25V	E3,5-8	アルミ電解コンデンサ ※極性有
	12	SV1	ML10	ML10	10ピンボックスヘッダ ※実装向き有り
	13	BAT1	TAKACHI_AA3	TAKACHI_AA3	電池ソケット (単 3×2), M2.6mm ねじ, ナット, ワッシャ
	14	H1,H2,H3,H4	MOUNT-HOLE3.2	3,2	Lアングル(サーボモータ固定) M3.0mm ねじ, ナット
	15	MOTOR1	SERVO	PICO/STD/F	サーボモータ (左) ※ 両面テープで固定
	16	MOTOR2	SERVO	PICO/STD/F	サーボモータ (右) ※ 両面テープで固定
	17	WHEEL	SERVO	WHEEL	サーボ車輪 (サーボ付属部品使用)
	18	WHEEL	SERVO	WHEEL	サーボ車輪 (サーボ付属部品使用)
	19	CASTER1	CASTER	CASTER	キャスター ※ 両面テープで固定
	20	CASTER2	CASTER	CASTER	キャスター ※ 両面テープで固定
	21	IC2	PHOTOSENS	PHOTOSENS	フォトインタラプタ ※実装向き有り, 高さ調整最後に実装
	23	IC4	PHOTOSENS	PHOTOSENS	フォトインタラプタ ※実装向き有り, 高さ調整最後に実装
	24	IC6	PHOTOSENS	PHOTOSENS	フォトインタラプタ ※実装向き有り, 高さ調整最後に実装
	25	H5,H6,H7,H8	MOUNT-HOLE3.2	3,2	金属製スペーサー(40mm) 3mm ねじ, ナット

※ 完成後、10 ピンフラットケーブルにて Arduino 互換マイコンボードと接続

(2) サーボモータの改造（連続回転サーボ）

（5章 参照）通常のサーボモータをロボット用連続回転サーボに改造を行う。（改造済み）

✓	実装順	名前	値	パッケージ	内容
✓	1	R1	3.6k Ω （橙青黒茶）	0204/5	金属皮膜抵抗（1/6W）1%
✓	2	R2	1.5k Ω （茶緑黒茶）	0204/5	金属皮膜抵抗（1/6W）1%

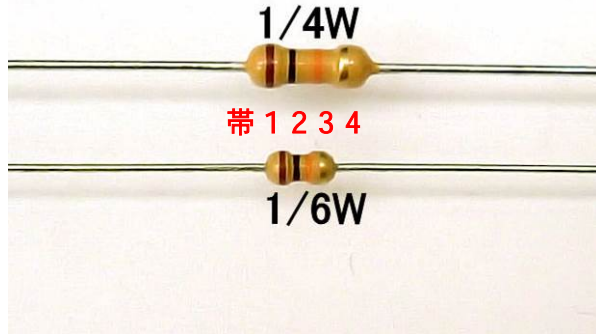
(3) 実装済み部品（DCDC コンバータ）

実装順	名前	値	パッケージ	内容
1	IC1	TPS61032	HTSSOPWP16	DCDC コンバータ IC
2	R3	560 k Ω	R0805	チップ抵抗
3	R4	180 k Ω	R0805	チップ抵抗
4	R5	1 M Ω	R0805	チップ抵抗
5	C3	2.2 μ F	C2012	チップ積層セラミックコンデンサ
6	C2	10 μ F	C3216	チップ積層セラミックコンデンサ
7	C5	10 μ F	C3216	チップ積層セラミックコンデンサ
8	C6	100 μ F	D2	チップタンタルコンデンサ
9	C4	100 μ F	D2	チップタンタルコンデンサ
10	L1	6.4 μ F	L_B82464G4	チップコイル
11	DIODE1	40V/2A	SS2040FL	ショットキーダイオード

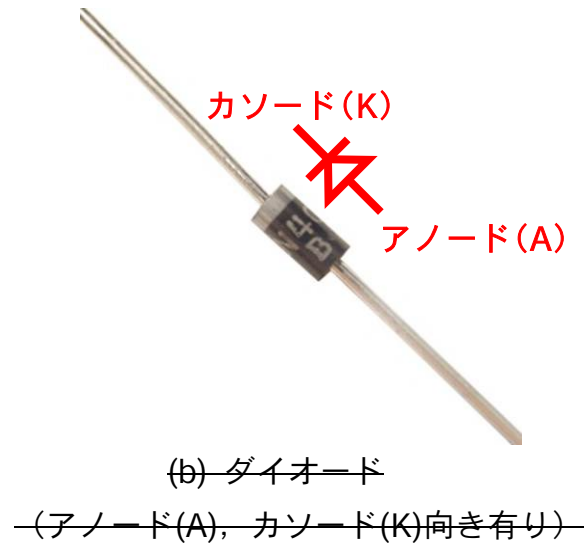
(4) オプション部品（ノイズ対策）

実装順	名前	値	パッケージ	内容
1	C8	0.033 μ F	C025-025X050	セラミックコンデンサ
2	C9	0.033 μ F	C025-025X050	セラミックコンデンサ

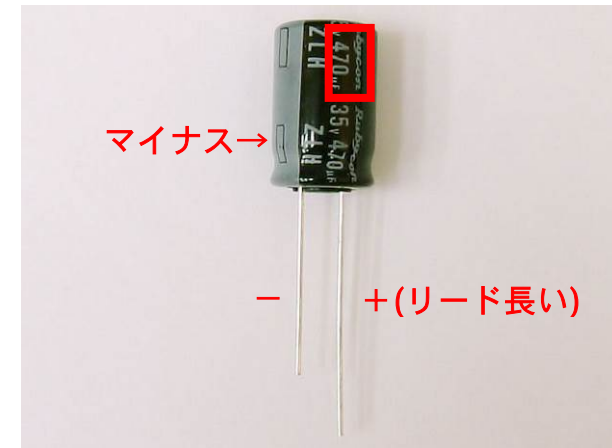
(5) 部品 (写真の一部は, 秋月電子通商(a)(b)(c), マルツパーツ館(f),(o)より転載)
サイズ目安



(a) 抵抗 (抵抗値の読み方は付録参照)



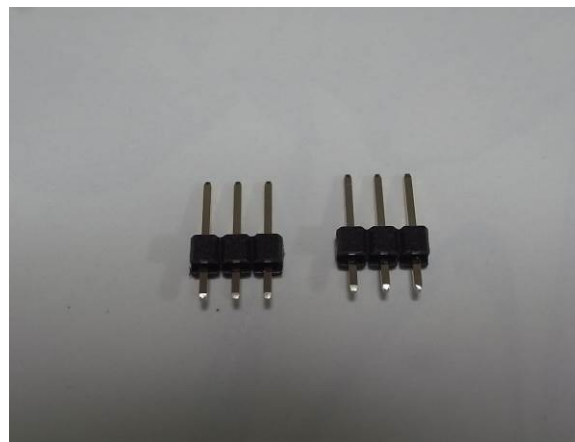
(b) ダイオード
(アノード(A), カソード(K)向き有り)



(c) アルミ電解コンデンサ
(極性有り, 足の長さまたはマイナス表記)



(d) モータ用電源スイッチ



(e) 3ピンヘッダ (W, R, B)



(f) 10ピンボックスヘッダ (実装向き有り)



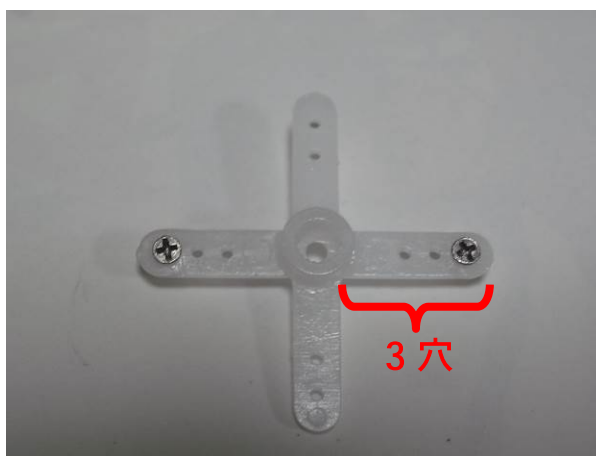
(g) 単三電池 × 2 基板用ソケット



(h) Lアングル, ねじ, ナット
(サーボ取り付け用)



(i) サーボモータ 2 個
配線: W(白) R(赤) B(黒)



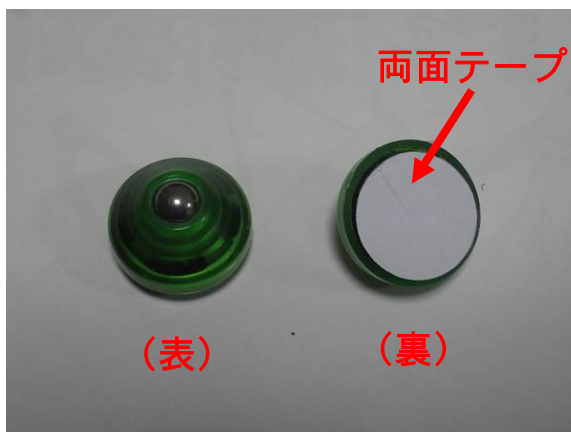
(j) サーボホーン十字, ねじ 2 個
(サーボモータ付属品)



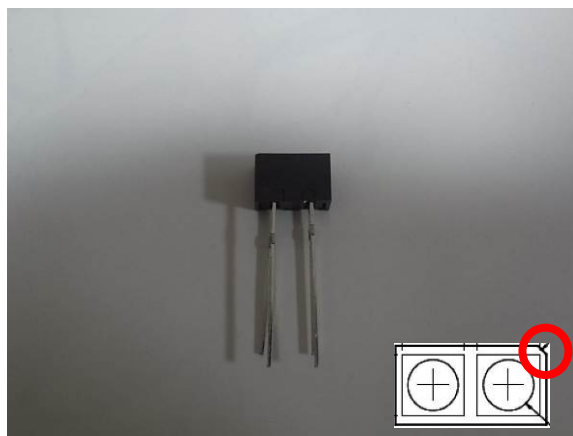
(k) サーボホーン (中央) 取り付けねじ
(サーボモータ付属品)



(l) モータ用車輪 40mm (2 個)



(m) キャスター 1個
(裏面, 両面テープ)



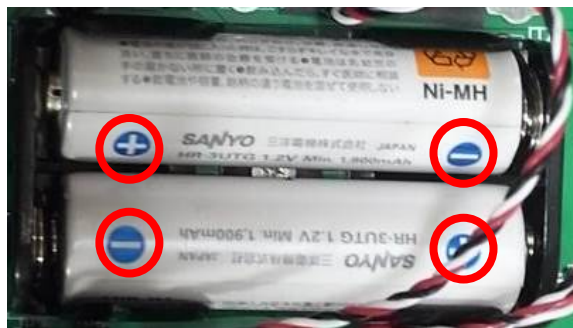
(n) フォトインタラプタ (実装向き有り)
※切り欠き位置確認



※写真はLFCBL14-01の物です。
(o) 10芯フラットケーブル
(10P ボックスの切り欠き向きに挿入)



(p) 金属製スペーサー40mm
(3mm ねじ, ナット)

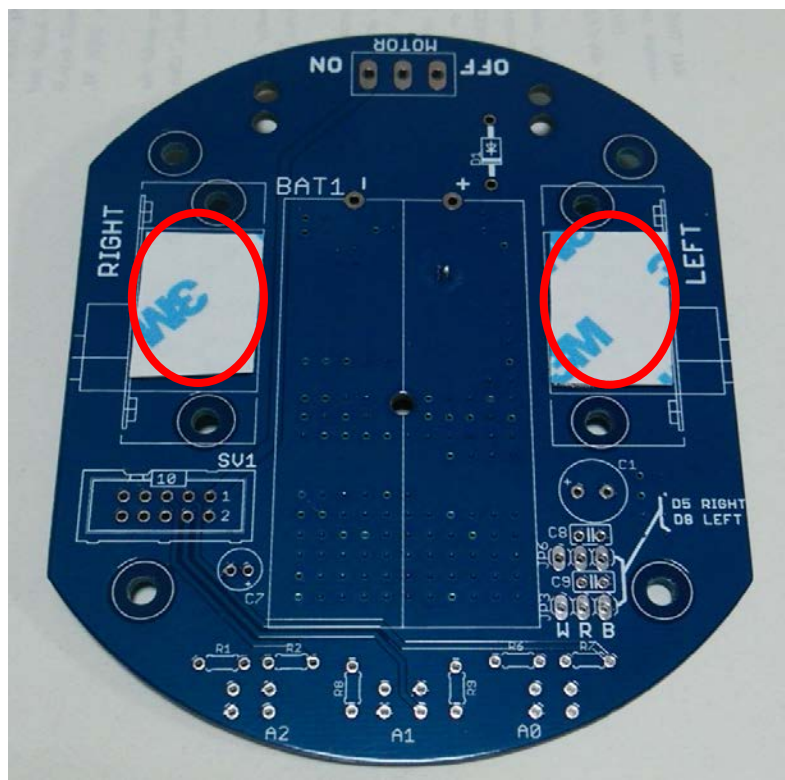


(q) 充電電池 (1.2V) 又はアルカリ乾電池 (1.5V) (オプション)
(極性有り (+, -))

4. 部品実装

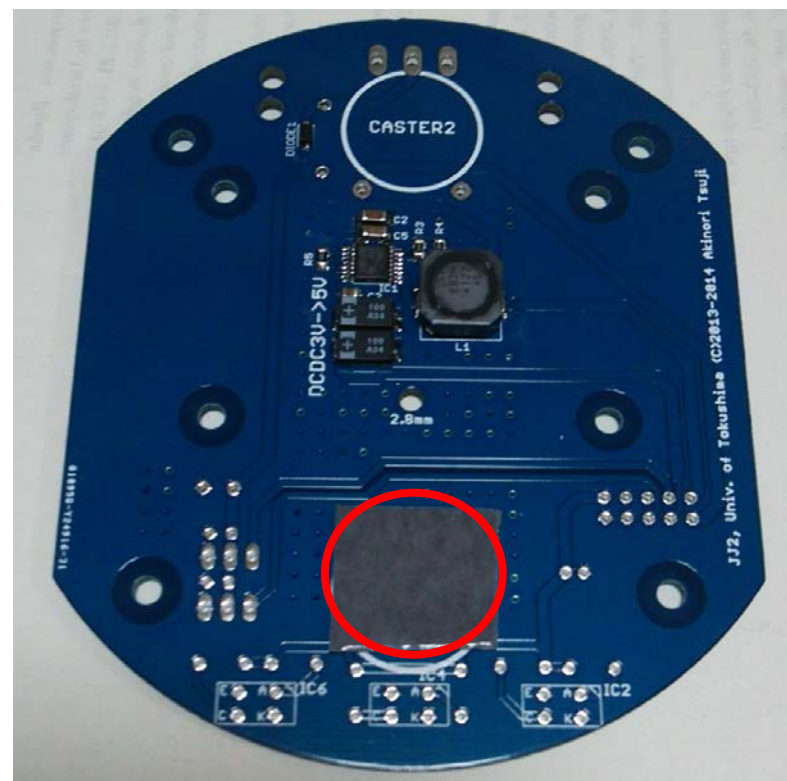
4.1. 配付基板

- ・ 配付基板の表面，裏面を確認する.



(a) 表

両面テープは「サーボモータ」取り付け時にはがす

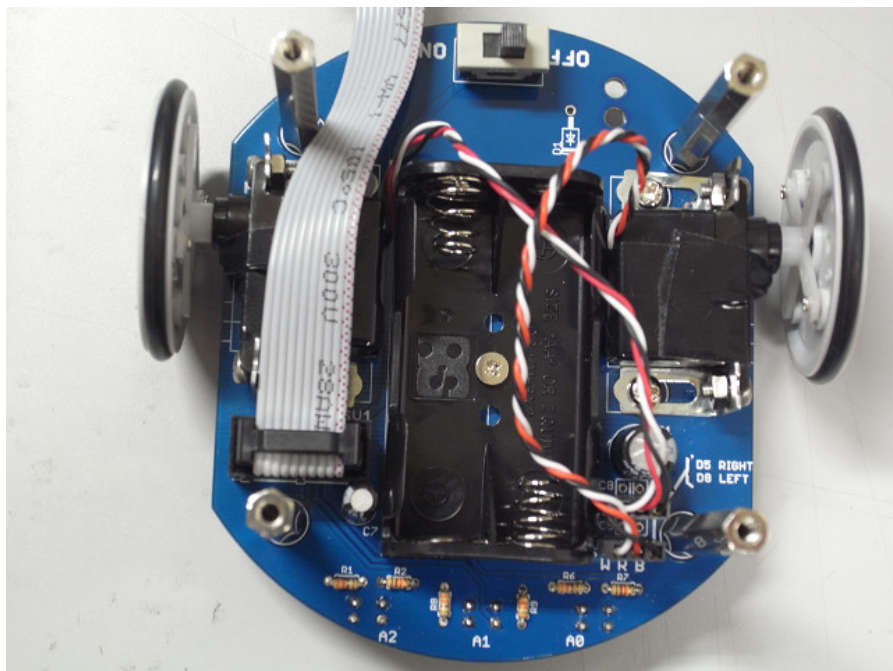


(b) 裏

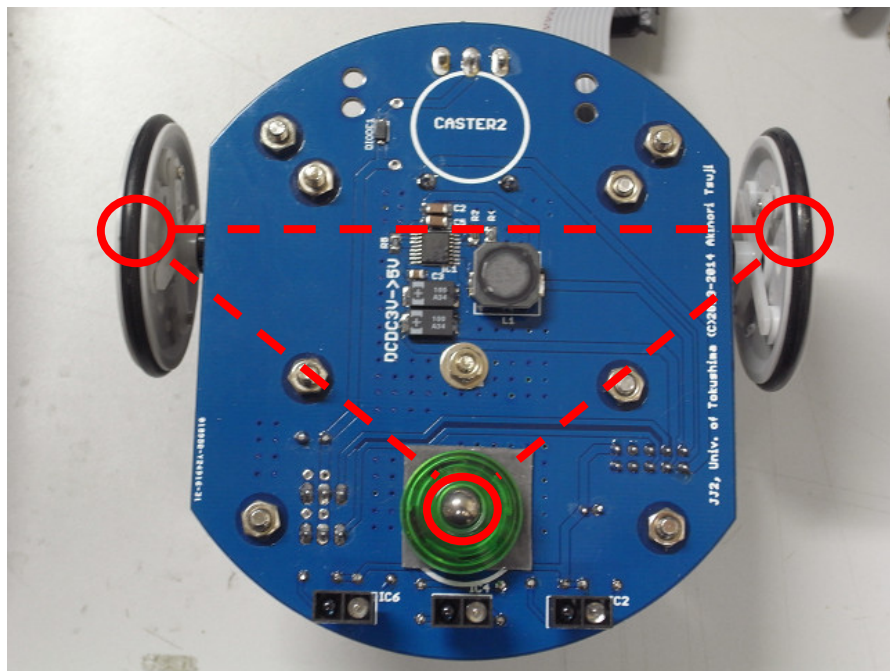
両面テープは「キャスター」取り付け時にはがす

4.2. 実装イメージ

サーボモータ、電池ボックス、キャスト、スペーサー、ボックスヘッド取り付け済みの写真



(a) 表



(b) 裏

(タイヤ2本, キャスターの三点で接地)

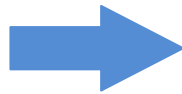
※ サーボモータの取り付け向き要確認！ (シルクパターンと逆向き)

4.3. サーボモータの取り付け

(1) サーボホーン（中央）のねじの組み合わせ（サーボ付属品）

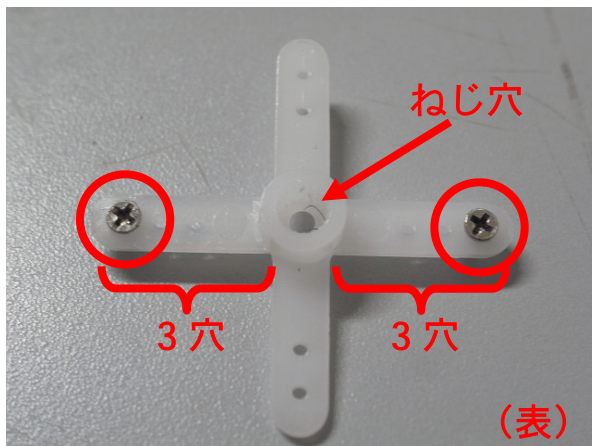


(a) サーボホーンねじ

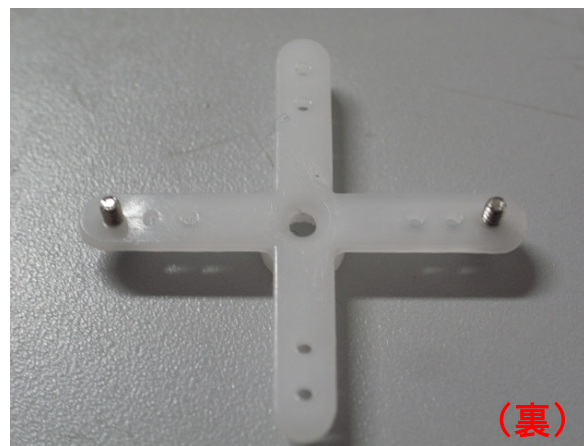


(b) サーボホーンねじ組み合わせ

(2) サーボホーンにねじを取り付け（2本）（サーボ付属品） **※ ねじ穴の挿入向き注意**

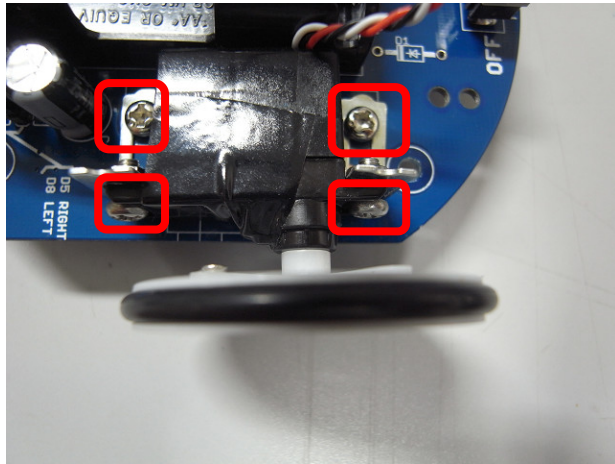


(a) サーボホーンにねじ 2 本取り付け（表）



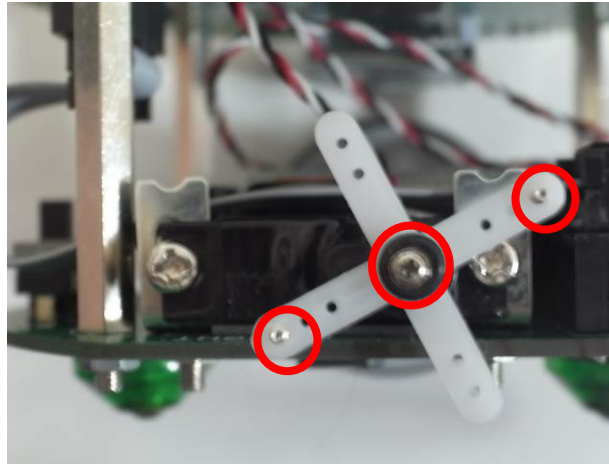
(b) サーボホーンねじ取り付け（裏）

(3) サーボモータ取り付け (Lアングルをねじで固定した後)



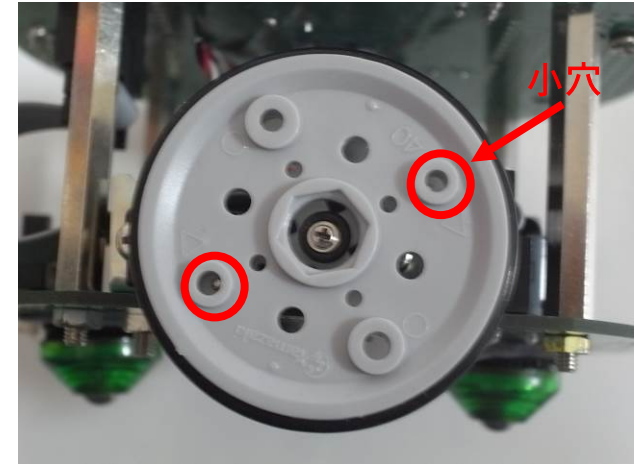
(a) サーボ取り付け

※ サーボ取り付け向き注意



(b) サーボホーン取り付け

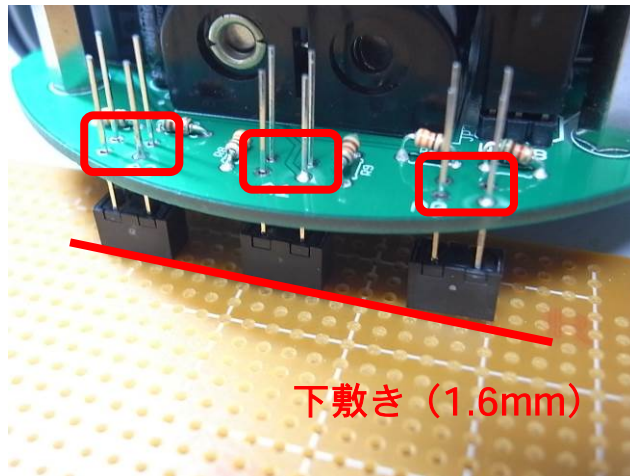
(1),(2)で準備



(c) 車輪取り付け

(サーボホーンをはめ込む又は接着)

(4) フォトインタラプタの実装方法



- すべての部品を実装した後、「最後に」フォトインタラプタの実装を行う。
タイヤ接地面から「1.6mm~2.0mm 程度」隙間をあける。

左図では、高さ調整のためフォトインタラプタとタイヤ接地面の間に、基板(1.6mm)を敷いた後に半田付けを行っている。

※ フォトインタラプタの実装向き (切り欠き位置) をシルク (裏面) に合わせる。

※ フォトインタラプタのケースは壊れやすいのでテープやボンドで固定する。

5. サーボモータの改造 (※改造済みのため作業不要)

5.1. ストッパーの除去

(1) モータのラベルをカッターではがす



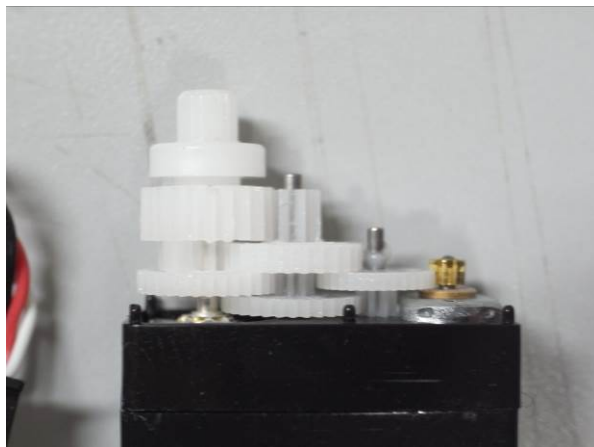
(2) ラベルをはがした後



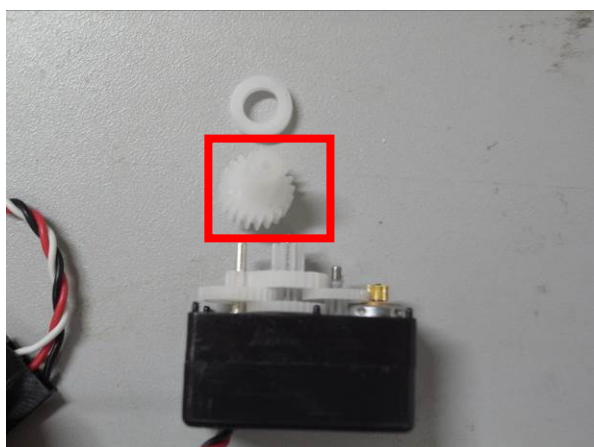
(3) 上ぶたを開く



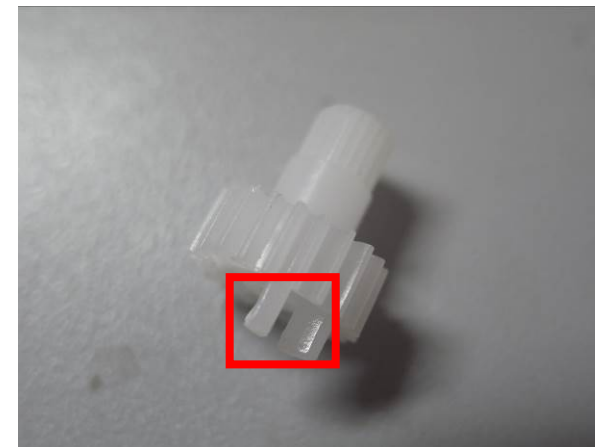
(4) ギヤ (5 枚) とモータ軸を確認



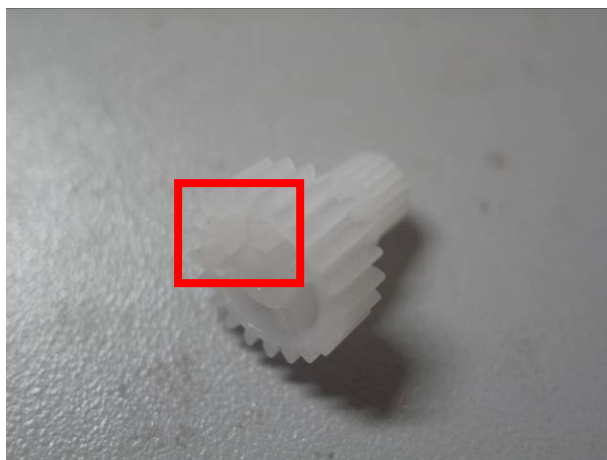
(5) 一番上のギアを取り外す



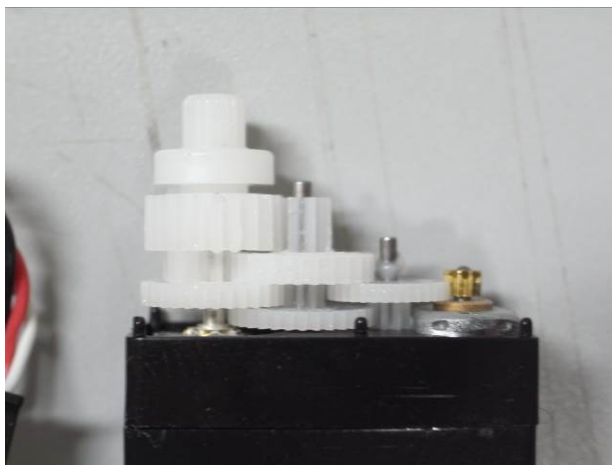
(6) ギアの「ストッパー」をニッパーで切る



(7) ニッパーで切った後 (平らに)



(8) ギアを元の位置に戻す



(9) カバーをする



5.2. ポテンショメータ配線の除去

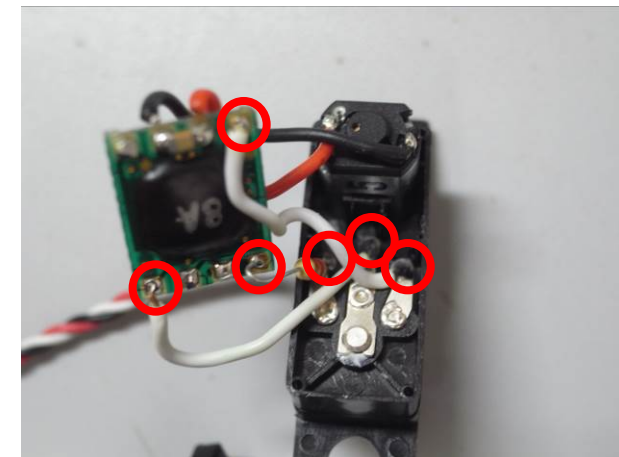
(1) 底ぶたを開く



(2) 制御基板とモータが見える位置に固定

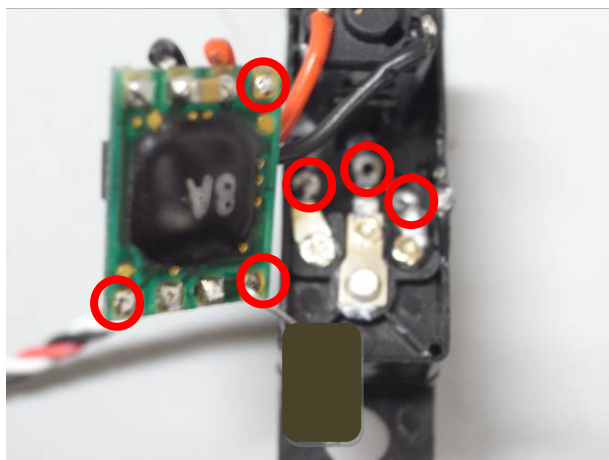


(3) ニッパーで白線と抵抗のリードを切る

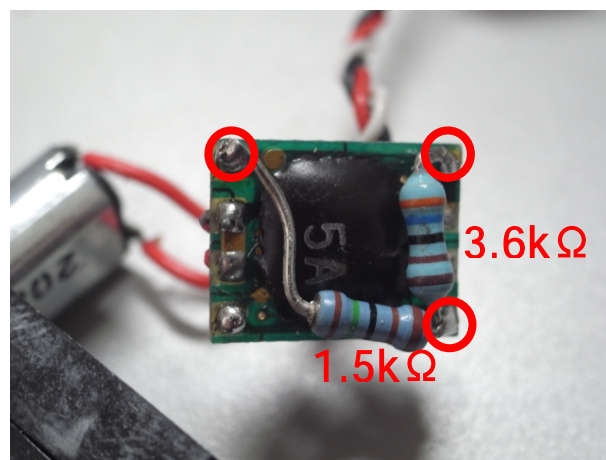


※ 抵抗は取り去る

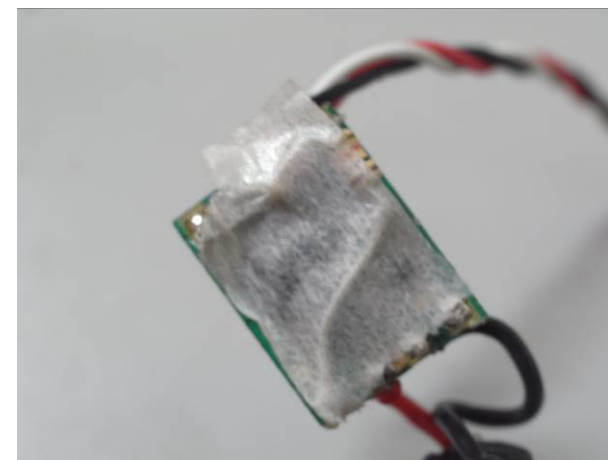
(4) 白線と抵抗のリード線を切断した後



(5) 3.6k,1.5kΩの抵抗を半田付けする

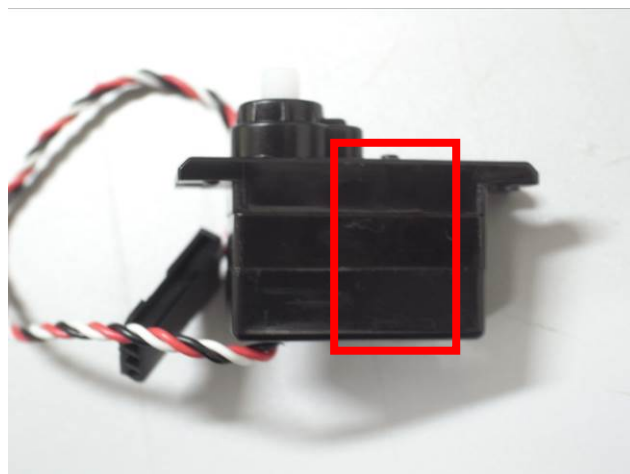


(6) テープを貼り制御基板を保護する



※ 抵抗のリード線が基板のパッドに接触しないように注意

(7) ふたをしっかりと閉じてビニールテープを巻く



(8) 次章のテストプログラムによりモータの動作確認（正転／逆転）を行う

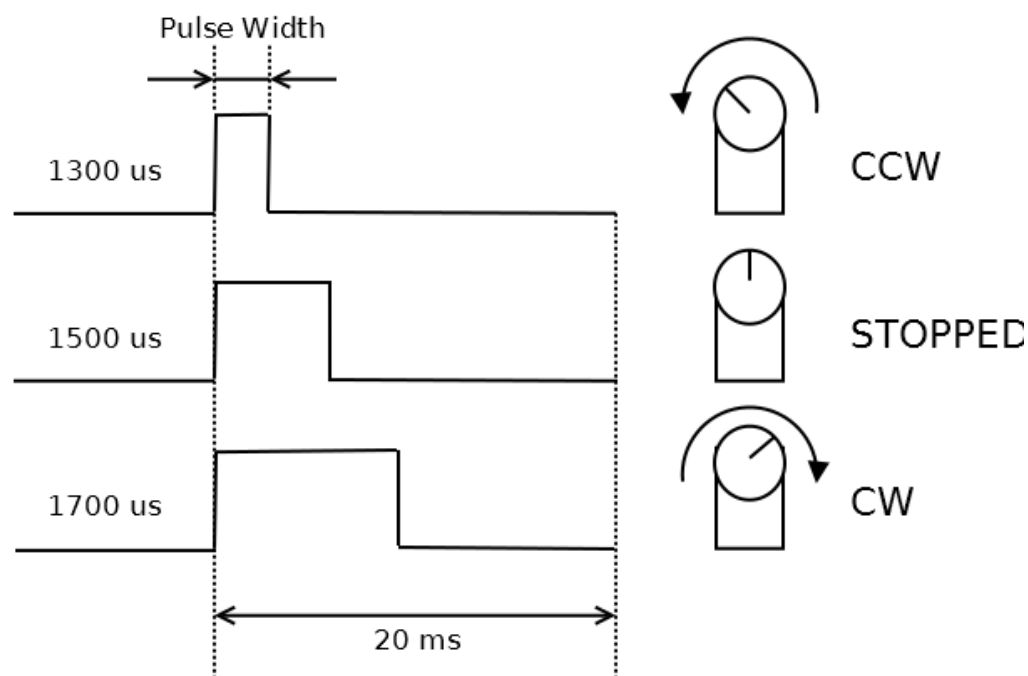
- ふたとふたの間に隙間がないよう制御基板をケース内におさめる.
- ふたを強く閉めすぎると、モータが押されギアが噛み合わず回転しなくなる.
- ビニールテープは、引っ張りながら強く巻く.

6. サーボモータのキャリブレーション

6.1. サーボモータの動作原理

サーボモータは、直流モータとサーボ制御回路を内蔵したモータである。サーボモータの制御端子に一定周期のパルス幅(Pulse Width)の信号を与えることでモータを一定の角度回転できる。一般にサーボモータは、パルス幅が約 $1500\mu\text{s}$ のとき停止して、それより短いと反時計回り (CCW: Counter Clock Wise), 長いと時計回り (CW: Clock Wise) に回転する。このパルス信号を 20 ミリ秒間隔 (周期) で繰り返し与える。ロボットの速度制御は、サーボモータに与えるパルス幅を変えることで行える。

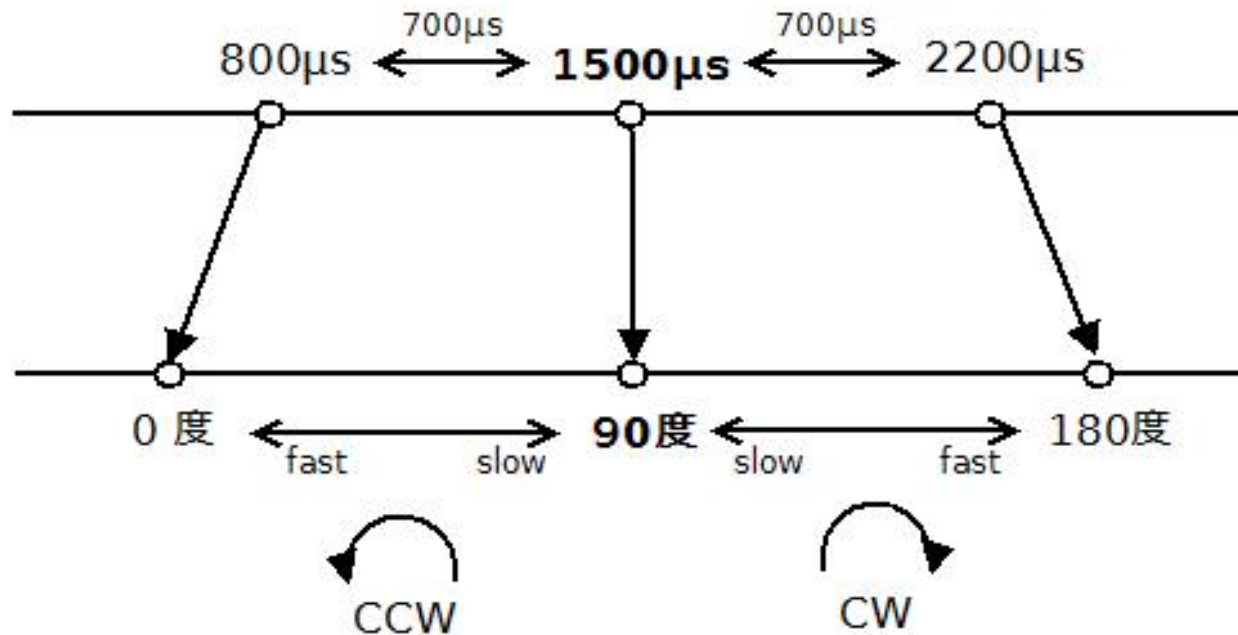
連続回転サーボでは、モータが停止状態 (約 $1500\mu\text{s}$) を基準として、モータの回転速度はパルス幅が $1500\mu\text{s}$ に近いと回転角が小さいため「遅く」、遠いと回転角が大きくなり「速く」回転する。



6.2. キャリブレーション

サーボモータの速度制御を行うため、サーボモータの可動範囲と停止位置を調べるキャリブレーションを行う。サーボモータに入力できるパルス幅は、**800 μ 秒~2200 μ 秒の範囲**である。この範囲をサーボモータの回転角に対応させて**0度~180度**とする。このとき、回転角90度が停止位置、0度が反時計回り（最大速度）、180度が時計回り（最大速度）となる。サーボモータのキャリブレーションでは、停止位置(90度)を基準に、0度回転方向、180度回転方向にパルス幅を均等に割り当てる。

たとえば、下図のようにサーボモータの停止位置（90度）が1500 μ 秒のとき、パルス幅の範囲 $\pm 700\mu$ 秒を割り当てると800 μ 秒が0度、2200 μ 秒が180度となる。



6.3. Arduino によるサーボモータのキャリブレーション

Arduino のサーボモータ制御ライブラリ (Servo) を用いてサーボモータのキャリブレーションを行う。Arduino のサーボモータ制御は、パルス幅または回転角度により行える。スケッチ 1 は、パルス幅を 1400μ 秒から 1600μ 秒まで徐々に変化させてサーボモータの基準位置 (90 度) を探している。スケッチを実行して、パルス幅をシリアルモニタに表示してモータが停止する値を確認する。ここでは、左モータ、右モータ、各々に関して Arduino シリアルモニタ上で基準となるモータの停止位置 (90 度) のパルス幅 (モータが完全に停止する位置) を調べる。(モータに軽く手を当てて、モータ筐体の振動がなくなるときが停止位置の目安になる。)

※ スケッチの書き込み中は、モータのスイッチを OFF にしておく。(プログラムロードと同時にロボットが走り出すため)

※ スケッチの書き込み時、電源のスイッチを BATTERY 側にしておく。(電池からの電源供給でロボットを走らせるため)

スケッチ 1. servo_calib.ino

```
#include <Servo.h>
const int servoL_Pin = 9; // uses timer1
const int servoR_Pin = 5; // uses timer1
Servo servoL; // left servo
Servo servoR; // right servo

void setup() {
  delay(1000);
  servoL.attach(servoL_Pin); // left servo is connected to d9
  servoR.attach(servoR_Pin); // right servo is connected to d5
  Serial.begin(9600); // open serial port by setting 9600 bps
}
```

```
void loop() {
  for (int i=1400; i<1600; i++) { // 左, 右, 交互に確認
    servoL.writeMicroseconds(i);
    // servoR.writeMicroseconds(i);
    Serial.println(i);
    delay(200);
  }
}
```

スケッチ 1 の停止位置のパルス幅の確認に続いて、パルス幅の最小値、最大値を決定する。関数

Servo.attach(Pin, min_pulsewidth, max_pulsewidth)

に、パルス幅の最小値、最大値を設定して**ロボットがまっすぐ走るよう**調整する。ここで、パルス幅の最小値、最大値は 6.2 節を参考に、停止位置を基準にしてパルス幅 ± 700 (μ 秒)(参考値)を設定する。左右モータのスピードに差がある場合には、パルス幅 $\pm \Delta t$ (μ 秒)を狭めたり、広げたりして調整する。スケッチ 2 では、サーボモータの設定 (ピン, 最小, 最大のパルス幅) を行い、前進, 後退, 停止を行う。

※ スケッチの書き込み中は、モータのスイッチを OFF にしておく。(プログラムロードと同時にロボットが走り出すため)

※ スケッチの書き込み時、電源のスイッチを BATTERY 側にしておく。(電池からの電源供給でロボットを走らせるため)

スケッチ 2. servo_angle.ino

```
#include <Servo.h>
const int servoL_Pin = 9; // uses timer1
const int servoR_Pin = 5; // uses timer1
Servo servoL; // left servo
Servo servoR; // right servo

void setup() {
  delay(1000);
  servoL.attach(servoL_Pin, 800, 2200); // 800,2000 を書き換える
  servoR.attach(servoR_Pin, 800, 2200); // 800,2000 を書き換える
  Serial.begin(9600); // open serial port by setting 9600 bps
}

void loop() {
  servoL.write(45);
  servoR.write(135); // 180 - servoL
  delay(1000);
  servoL.write(135);
  servoL.write(45); // 180 - servoL
  delay(1000);
  servoL.write(90); // 停止
  servoR.write(90); // 停止
  delay(1000);
}
```

7. 付録

(1) 抵抗値

色	銀	金	黒	茶	赤	橙	黄	緑	青	紫	灰	白
数字	—	—	0	1	2	3	4	5	6	7	8	9
指数	10^{-2}	10^{-1}	1	10	10^2	10^3	10^4	10^5	10^6	10^7	10^8	10^9
誤差	±10	±5	±20	±1	±2	—	—	±0.5	±0.24	±0.1	±0.05	—

例) (帯 1,2,3,4 の順) 茶, 黒, 橙, 金 (数字, 数字, 指数, 誤差)

$$10 \times 10^3 \Omega = 10 \text{ k}\Omega (\pm 5\%)$$

(2) 略数字の例

表示	抵抗	コンデンサ	コイル
126	12 M Ω	12 μ F	—
125	1.2 M Ω	1.2 μ F	120 mH
124	120 k Ω	0.12 μ F	12 mH
123	12 k Ω	0.012 μ F	1.2 mH
122	1.2 k Ω	1200 pF	1200 μ H
121	120 Ω	120 pF	120 μ H
120 (12)	12 Ω	12 pF	12 μ H

例) 104 (コンデンサ)

$$10 \times 10^4 \text{ pF} = 0.1 \mu\text{F}$$

単位 : メガ : M=10⁶, キロ : k=10³, ミリ : m=10⁻³, マイクロ : μ =10⁻⁶, ナノ : n=10⁻⁹, ピコ : p=10⁻¹²

(3) サーボモータの規格

パルス幅 (ms)	ホームポジション	ホームポジション (連続回転)
0.7 ~ 0.8 ms	安全領域 (CW)	安全領域 (CW)
0.8 ms (800 μ s)	+90 度	180 度
1.15 ms (1150 μ s)	+45 度	90 度
1.5 ms (1500 μ s)	0 度	90 度
1.85 ms (1850 μ s)	-45 度	45 度
2.2 ms (2200 μ s)	-90 度	0 度
2.2 ~ 2.3 ms	安全領域 (CCW)	安全領域 (CCW)

(4) プログラム書き込み中のエラー回避

Arduino のメニュー→ツールより、マイコン、シリアルポートが正しく選択されているか確認する。

1. Serial Port COMx Already Used …, と表示される.

シリアルポートが使用中のため、USB ケーブルを抜き挿しした後に、しばらく待ってから書き込む。

2. 書き込みが途中で止まる. (シリアルポートを用いたプログラム実行中に発生することが多い.)

再度書き込みを行い、下欄に PORTS {COMn} / {COMn} => {} が表示されている最中に、基板上の RESET ボタンを押す。

3. マイコンボードに書き込もうとしましたがエラーが発生しました. と表示される.

USB ケーブルの接続を確認、ツールのシリアルポートを確認した後に再度書き込む。

4. Couldn't find a Leonardo on the selected port., と表示される.

USB ケーブルが接続されていない、またはマイコンボードの選択を間違えているので、それらを確認した後、再度書き込む。

8. 改編履歴

日付	名前	内容
2014年1月30日(木)	辻 明典	新規作成
2014年2月1日(土)	辻 明典	0.9版 サーボモータ記述追加
2014年2月15日(土)	辻 明典	1.0版
2014年9月25日(木)	辻 明典	1.1版(ワークショップ)