

二足歩行ロボットの設計と実装  
—二足歩行のハードウェアとアルゴリズムの学習体系の開発—

野 津 伸 治

Shinji NOTSU : A Design and Implementation of a Humanoid Robot

鳥取看護大学・鳥取短期大学研究紀要 第81号 抜刷

2020年7月

# 二足歩行ロボットの設計と実装

## —二足歩行のハードウェアとアルゴリズムの学習体系の開発—

野 津 伸 治<sup>1</sup>

Shinji NOTSU : A Design and Implementation of a Humanoid Robot

本研究の目的は、二足歩行のアルゴリズムの学習ツールの設計と実装である。二足歩行を行うためのフレームの設計と歩行アルゴリズムの検証に焦点をあてたものである。成果としては、第1にはサーボブラケットの加工のしやすさと精度と強度を考慮してアクリル板のL字アングルの利用を前提に加工工具ルータを導入したこと。第2には、片軸サーボモーターの回転精度を安定させるための両軸化を行ったこと。第3には、プログラミング初心者でもアルゴリズムを実装しやすいブロック組み合わせ型言語を利用したことである。

キーワード：二足歩行ロボット micro:bit ブロック・プログラミング

### はじめに

ロボットは、先進国において、産業界では工場における製造ラインの組み立て用途や、警備用途など様々な分野で用いられて久しい。また一般家庭向けでも、愛玩用や癒し用途、清掃用途など様々な製品が浸透している。また近年の介護現場で、人型ロボットと介護者が協調しながら介護にあたる可能性も出てきた。人型ロボットの移動の要である二足歩行は車輪型歩行と比較して自由度が大きく、重心の不安定さを考慮して実装していくことが求められる。本研究では人型ロボットの二足歩行の一つである、静的歩行のロボットへ実装を行う。

### 1. ロボットの基本設計

人型ロボットのプロトタイプを実装するにあたって、歩行の安定度とメカニズムの簡素化から二足歩

行で4自由度のロボットの設計を目標とする。1本の脚を2自由度とすると、で3次元空間（側面から見て前後に回転するピッチ軸、正面から見て左右に回転するロール軸、上から見て地面と平行に回転するヨー軸）では2平面の自由度が確保されることになる。ロボットは前後左右の移動を可能にするが、移動するときに重心の移動を注意しないと転倒につながる。

ヒトはあえてバランスを崩すことで二足歩行を行っているが、その自然な動き（動歩行）の制御のためにセンサーやアルゴリズムの精緻化が必須である。一方歩行動作のどの時点を捉えても重心のバランスが取れる歩行（静歩行）はぎこちないが、アルゴリズムの実装が簡便にできる。

### 2. ロボットの電子回路

基本的な構成は複数のサーボモーターをマイクロコンピュータから制御するものとする。様々な歩行のパターンをプログラムで変更できるものとする。将来の脚の自由度の増加にも対応できるものとす

1 鳥取短期大学生活学科情報・経営専攻

る。従って、マイクロコンピュータとサーボモーターは直接接続するのではなく、中間に制御ボードを介在して行う。

(1) マイクロコンピュータ

この度設計した二足歩行ロボットでは最大4個のサーボモーターを制御するために32ビットARM Cortex-M0 ベースのプロセッサを搭載したマイコンボード micro:bit を用いた。マイクロコンピュータと制御ボードの間でI<sup>2</sup>C通信を行える機能を有している。初心者がプログラミングするとき利用できるブロック型プログラミング言語のライブラリの充実もマイコン選択の決め手である。

(2) サーボモーター

二足歩行プログラムの開発において本体重量の軽量化を最大限に図るためとコスト削減、最小限のトルクの観点からアナログサーボモーターSG90 互換機を利用した。

表1はSG90の物理的及び電気特性の仕様である。

表1. SG90 サーボモーターの仕様

PWMサイクル	20mS
制御パルス	0.5ms~2.4ms
制御角	±約90° (180°)
配線	茶=GND, 赤=電源 [+], 橙=制御信号 [JRタイプ]
トルク	1.8kgf・cm
動作速度	0.1秒/60度
動作電圧	4.8V (~5V)
温度範囲	0℃~55℃
外形寸法	22.2 × 11.8 × 31mm
重量	9g

SG90 サーボモーターのブラケットを自作するためにこの形状の寸法の詳細をノギスで計測してまとめたものが図1のとおりである。

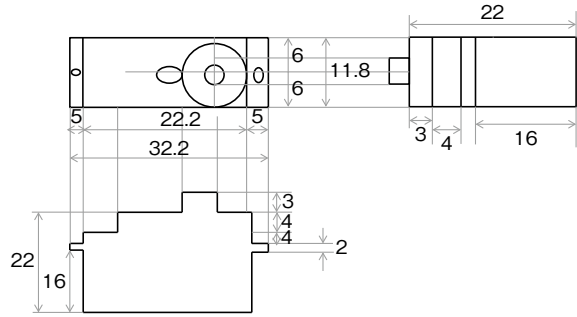


図1. SG90 サーボモーターの詳細な寸法 (単位: mm)

試作の段階で片軸サーボモーターの動作の精度の問題が生じたので、SG90を分解して、軸の反対側のフレームにドリルで穴を開けて、半径Φ=2mmで長さL=10mmのネジを通して図2のように両軸化することで回転精度の著しい向上が確認できたのでこれを生かすブラケットを製作することにした。

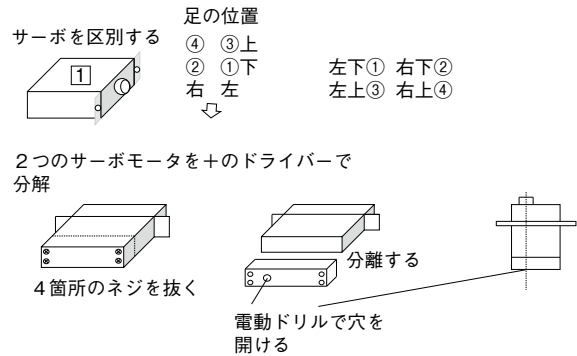


図2. SG90 サーボモーターの両軸化

(3) サーボ制御コントローラー

1個の制御コントローラーで最大16個のサーボモーターを12bitの分解能で扱うことができれば十分であるのでPCA9685を用いることにする。

表2. マイコンボードとPCA9685の結線

AE-PCA9685 側	micro:bit 側
+5V	別電源よりサーボへ給電
OE	NC
SCL	P19
SDA	P20
GND	GND
VCC	+3V3

Micro:bit とPCA9685 の結線は表 2 のとおりである。

(4) 回路の実態配線図

マイコンボード micro:bit にマイコン制御ボード PCA9685 を経由して配線した。サーボモーター SG90 を 4 個制御するマイコンボード用とサーボモーター用のそれぞれの電源も含めた全体の回路図を図 3 に示す。

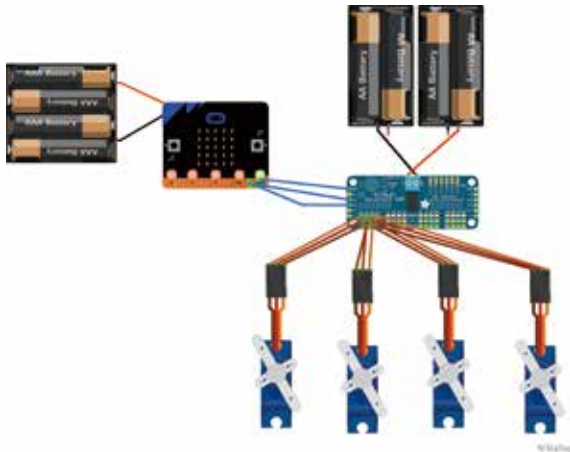


図 3. 二足歩行ロボットの電子回路実態配線図

3. ロボットのフレーム設計

二足歩行を実現するためには、例えば前進なら左右の脚のうち、まず右下脚を右側に傾け、次に右上脚（腰に相当）を時計回りに回転させる。右下脚を元の位置に戻す。続いて左下脚を左側に傾け、続いて左上脚（腰に相当）を反時計回りに回転する。左下脚を元へ戻す。フレーム加工と想定した歩行アルゴリズムの試行の推移を以下で説明する。

(1) L 字アクリル板による 3 次元フレーム

個々の脚の構造と形成用の素材が課題である。素材アクリル板にすることで、一定の重量で強度が確保できる。また平面を垂直に貼り合わせるもしくは融着する場合の精度が問題になると考え、L 字のアンクル状態のアクリル板を切り出して加工することで問題の解決を図った。加工上の問題は、サーボブラケッ

トを組み合わせる場合に、ブラケットの形状が複雑になると組み立ての難易度が上がることである。

(2) 順運動

2 自由度の脚の先端を座標  $(x, y)$  へ移動するために 2 つのサーボモーターに与える角度  $\theta_1$  と  $\theta_2$  の関係を考察する。前提として上側の脚の長さを  $L_1$ 、下側の足の長さを  $L_2$  とする。  $L_1$  を座標  $(u, v)$  へ移動するために角度  $\theta_1$  動かし、さらに  $L_2$  を座標  $(x, y)$  へ移動するために角度  $\theta_2$  動かす。これを図示したものが図 4 である。(式 1) は角度  $\theta_1$  と  $\theta_2$ 、および長さ  $L_1$  と  $L_2$  から座標  $(x, y)$  を表す。

$$\begin{cases} x = L_1 \cos \theta_1 + L_2 \cos(\theta_1 + \theta_2) \\ y = L_1 \sin \theta_1 + L_2 \sin(\theta_1 + \theta_2) \end{cases} \sim (\text{式 1})$$

～角度  $\theta_1$  と  $\theta_2$  を入力 → アーム先端の座標  $(x, y)$

$L_1, L_2$  を二つのアームの長さ

$\theta_1, \theta_2$  を二つの関節の角度

$(0, 0)$  を根元の関節の位置

$(x, y)$  をアームの先端部分の座標

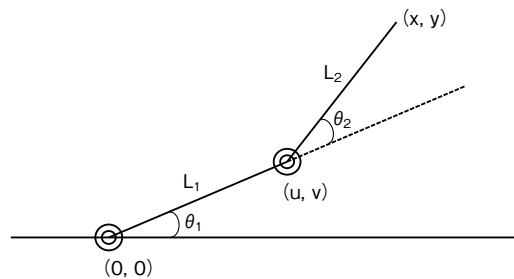


図 4. 順運動：2 自由度の脚の長さ、角度と位置の関係

マイコンから各サーボモーターへ与える命令は角度  $\theta_1$  と  $\theta_2$  になるので、図 4 の逆関数を求めなければならない。上記の順運動の逆運動に相当する (式 2) と (式 3) である。

$$\theta_2 = \pm \text{atan2} \sqrt{((L_1 + L_2)^2 - (x^2 + y^2)) / ((x^2 + y^2) - (L_1 - L_2)^2)} \sim (\text{式 2})$$

$$\begin{cases} \theta_1 = \text{atan2}(y/x) + \text{atan2}(L_2 \sin \theta_2) / (L_1 + L_2 \sin \theta_2) \\ \theta_1 = \text{atan2}(y/x) - \text{atan2}(L_2 \sin \theta_2) / (L_1 + L_2 \sin \theta_2) \end{cases} \sim (\text{式 3})$$

～アーム先端の座標  $(x, y)$  を入力 → 関節  $\theta_1$  と  $\theta_2$  の角度が出力

これに基づき2自由度の1本の脚の動かし方を図示したものが図5である。

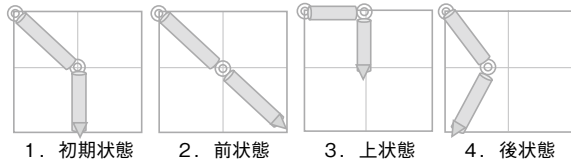


図5. 2自由度の1本脚の運び

この原理を考慮してサーボブラケットの加工とフレームとしての組立を考えた。

(3) ロボットのフレーム加工0

3次元空間で1平面(ピッチ軸)の移動精度にサーボモーター2個を利用するのではなく、2平面(ピッチ軸とヨー軸)に1サーボモーターずつを用いることで前後左右の移動の精度と速度を確保する形に歩行アルゴリズムと対応するサーボブラケットの接合の仕方を変更した。表3がこの設計に基づく利用パーツの一覧である。

アクリル製のL字アングルからサーボブラケットを切り出して加工する際の寸法と穴の位置は図6のように行った。

図7は図6で加工したブラケットにサーボモーターSG90を組み込み回転させるイメージである。

図8はサーボに二本の脚を取り付けるフレーム加工を示している。

表3. 二足歩行ロボットのパーツ一覧

種類	品名	数	備考
マイコン	micro:bit	1	firmware=0254
ブレークアウト兼サーボコントローラー	kitronik-5612	1	PCA-9685
サーボモーター	SG90	4	
アルカリ乾電池	単3アルカリ乾電池 DC1.5V	5	
電池ボックス #1	単3乾電池×2	1	リード線・スイッチ付き
電池ボックス #1	単3乾電池×3	1	リード線・スイッチ付き
L字アングル		1	厚さ=2mm 40×40×1,000mm
アクリル板		1	厚さ=2mm 200×200mm
ビス		9	φ=2mm L=6mm
ビス		3	φ=2mm L=8mm
ナット		12	φ=2mm

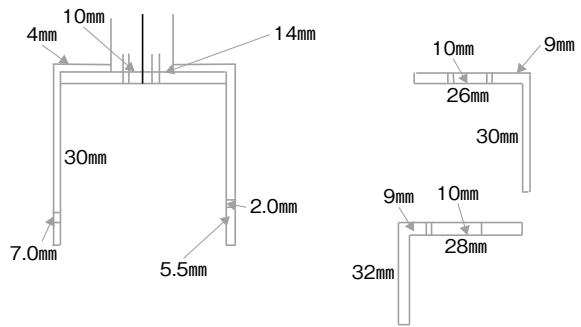


図6. L字アングルからサーボブラケットの加工

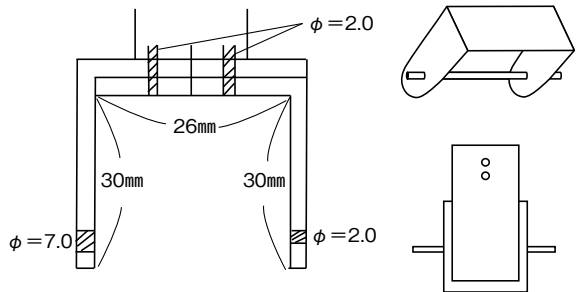
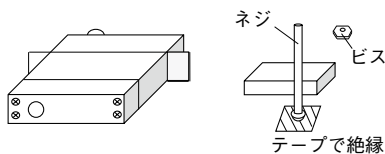


図7. サーボモーターの回転

穴を開けが終わったサーボは+ドライバーで元のように緩める



L字アングルカット

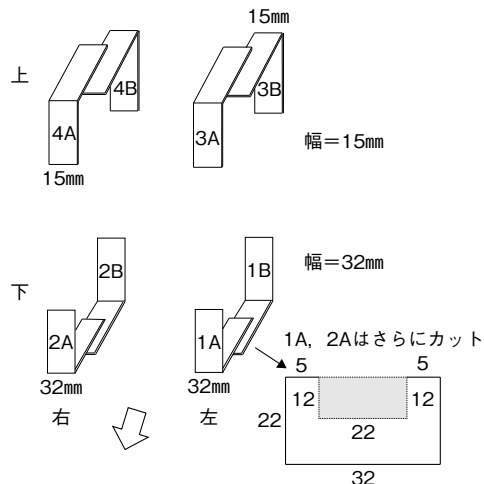
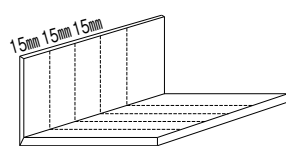


図8. 両脚のパーツの加工

二足歩行ロボットの設計と実装

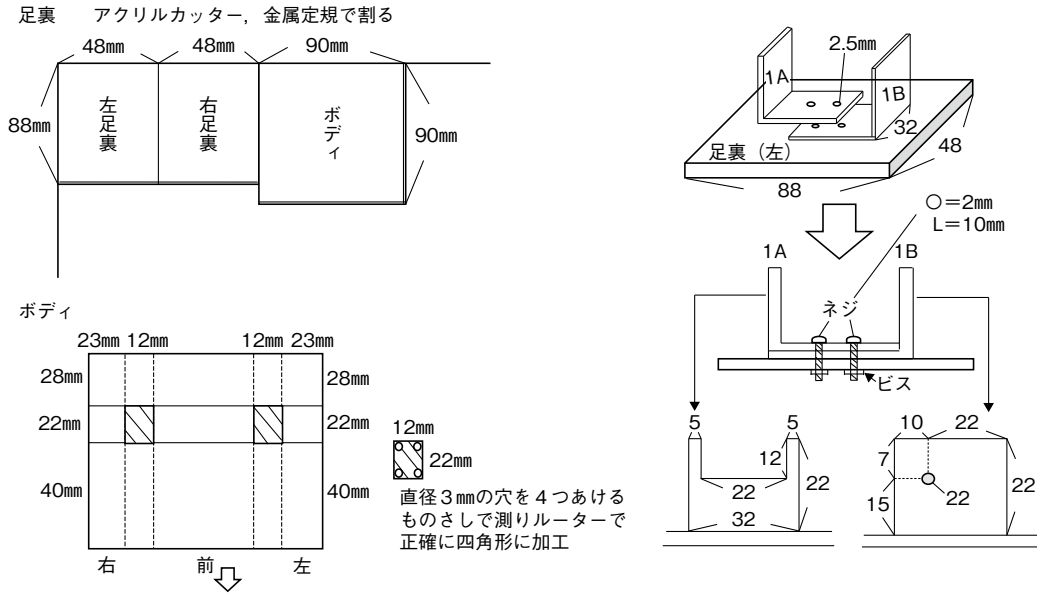


図9. ボディと足裏のパーツの加工

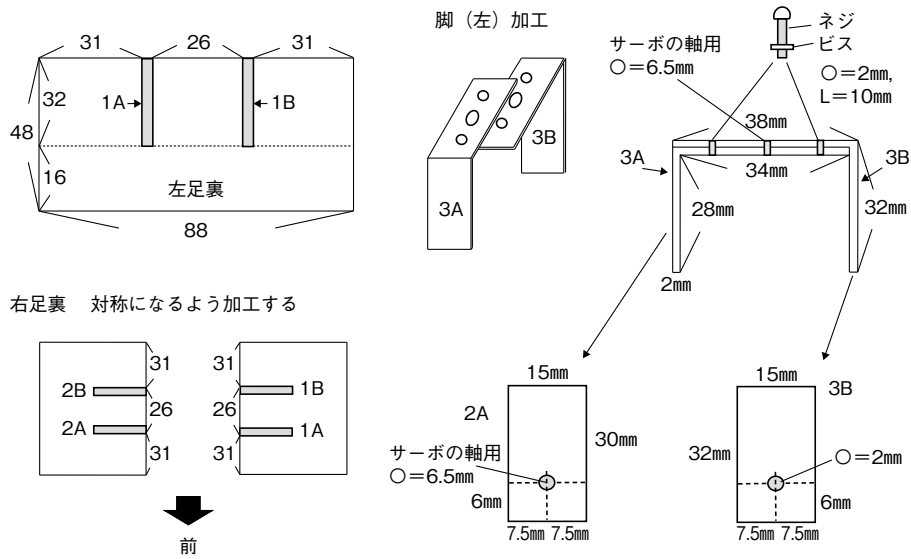


図10. 脚と足裏の接合部の加工 (1)

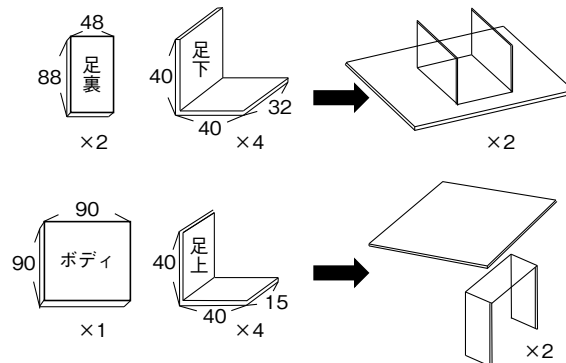


図11. 脚と足裏の接合部の加工 (2)

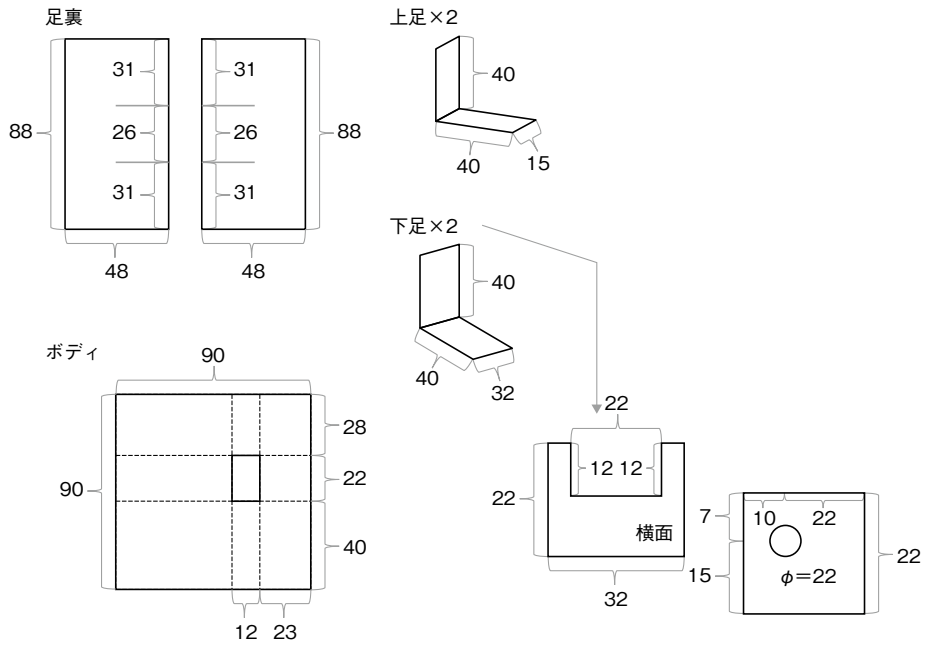


図 12. ボディへの脚の取付け加工

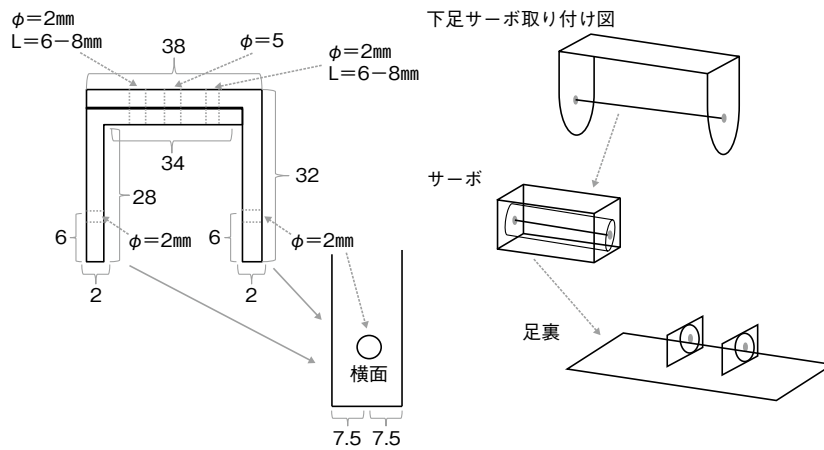


図 13. 足裏とサーボの回転の位置

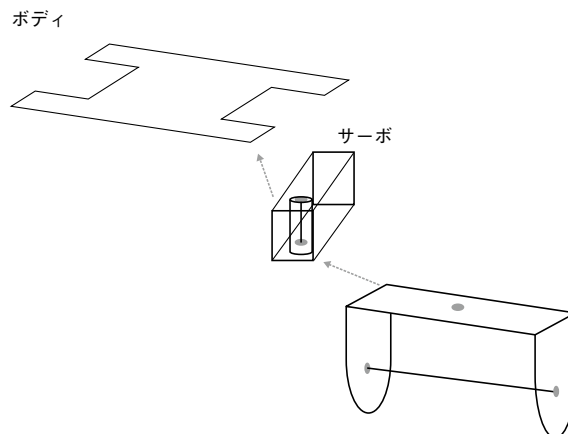


図 14. ボディ側サーボと脚の回転

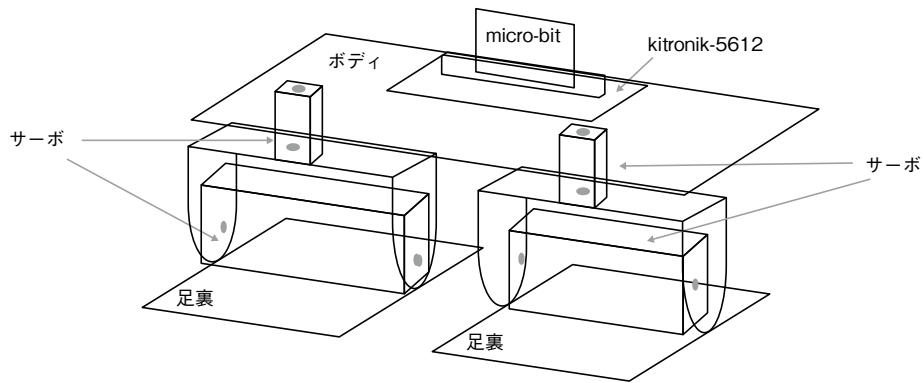


図 15. ボディ, 脚と足裏の組み立て

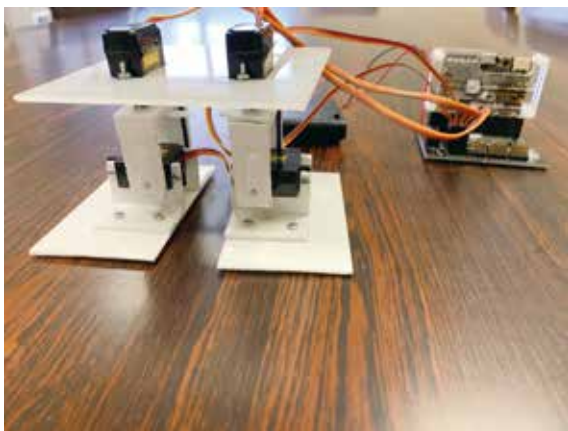


図 16. 二足歩行ロボットの全景 (正面から)

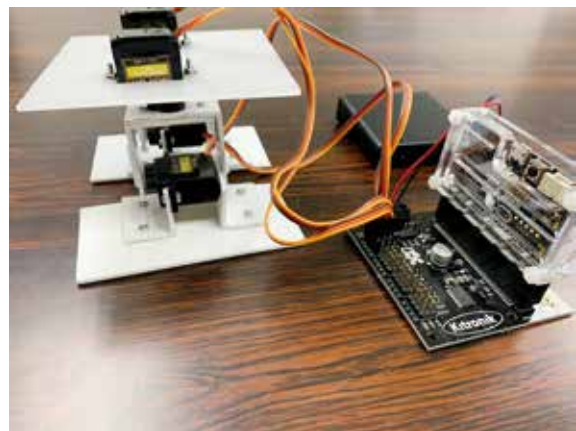


図 17. 二足歩行ロボットの全景 (側面から)

図 9 は二本の脚を取り付けるボディ側と足裏側のフレーム加工を示している。

図 10 と図 11 は脚と足裏の接合部の加工を示している。

図 12 は脚とボディの接合部の加工を示している。

図 13 は足裏とサーボの回転を示している。

図 14 はボディ側サーボで脚の回転の関係を示している。

図 15 は加工してきたボディ, 脚と足裏を組み立てを示した図である。

以上の過程を経て組み上げた二足歩行ロボットの全景が図 16 と図 17 である。

当初の設計通りに前後左右の動作の精度と速度が確保できた。

#### 4. ロボットの歩行アルゴリズム

二本脚で歩行する動物が前進する場合の脚の運びを観察すると、例えば、まず右脚を前に動かして重心を移動し、次に左脚を前に動かし重心を移動する。動物は敢えて安定状態を乱すことで自然な動きをしており、このことは動的歩行と呼ばれている。一方同じ二本脚で動くロボットの動作はぎこちない。これは、二本脚のうち地面に接している脚（接地脚）裏で定義される四角形の内側に重心が常にあるように非接地脚を動かしているからである。この歩行は静的歩行（図 18）と呼ばれている。

今回は二足歩行の学習のため後者の静的歩行で実装していくことにする。



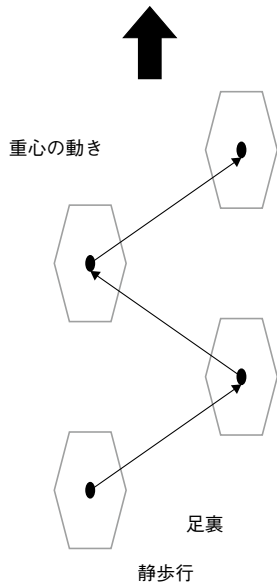


図 18. 静的歩行

(1) アルゴリズム

二足歩行ロボットの2つの脚の上部と下部にそれぞれ割り当ててあるサーボモーターに、1番から4番までの番号を割り当てて制御の時の識別子にする(図19).

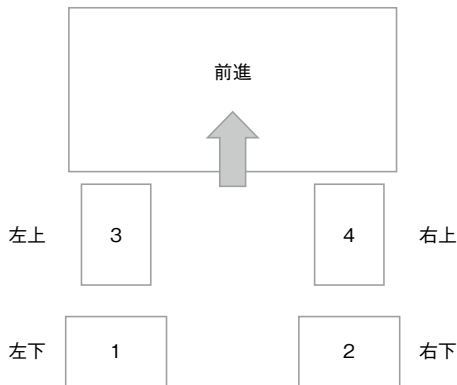


図 19. 二本4自由度の脚とサーボモーターの対応番号

静的歩行の脚の運びをロボットの真上から見た図(ヨー軸)と、その時にサーボモーターが動作させるべき角度を決定していくことにする。

ロボットが通电されたときのホームポジション(すべてのサーボモーターが90度(稼働角度の範囲は0度~180度)を示している)を表している。

前進するときの開始状態を考える。ステップIと

して右下脚を $\theta_1$ だけ右側に傾ける。次のステップIIとして右上足(腰に相当)を時計回りに $\theta_2$ だけ動す。ステップIIIで右下脚を元の位置に戻す( $-\theta_1$ )。ステップIVで、左下脚を左側へ $\theta_3$ だけ傾ける。ステップVで左上脚(腰に相当)を反時計回りに $\theta_4$ だけ回転させる。ステップVIで左下脚を元へ戻す( $-\theta_3$ )。これを繰り返すことでボディを前進させ続けていくことが可能である。

ロボットの後退は、前進運動の逆で実現できる。左右への歩行は向きの方向の脚の腰に相当するそれぞれの脚の回転する角度を前進以上にすることで

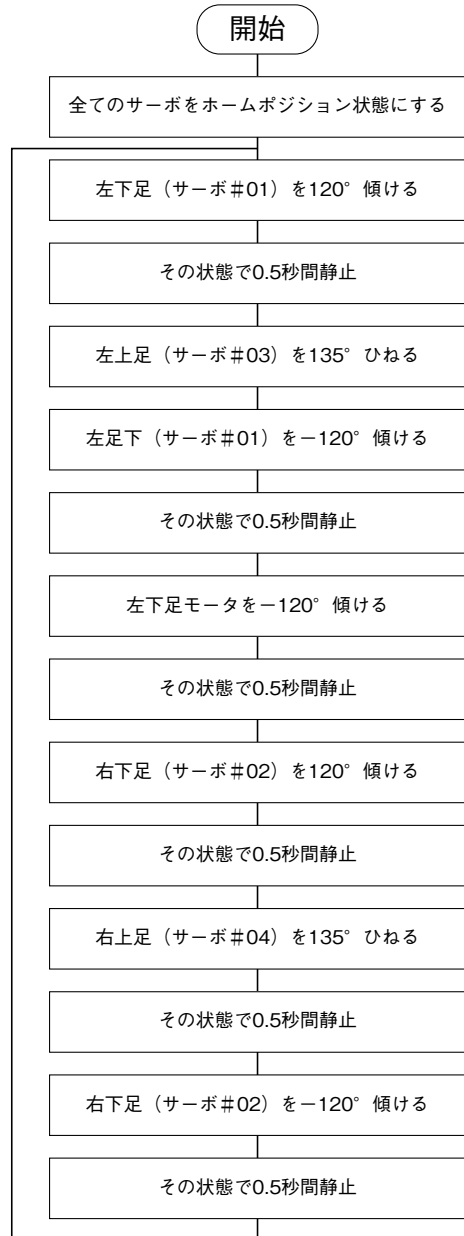


図 20. 前進の流れ図

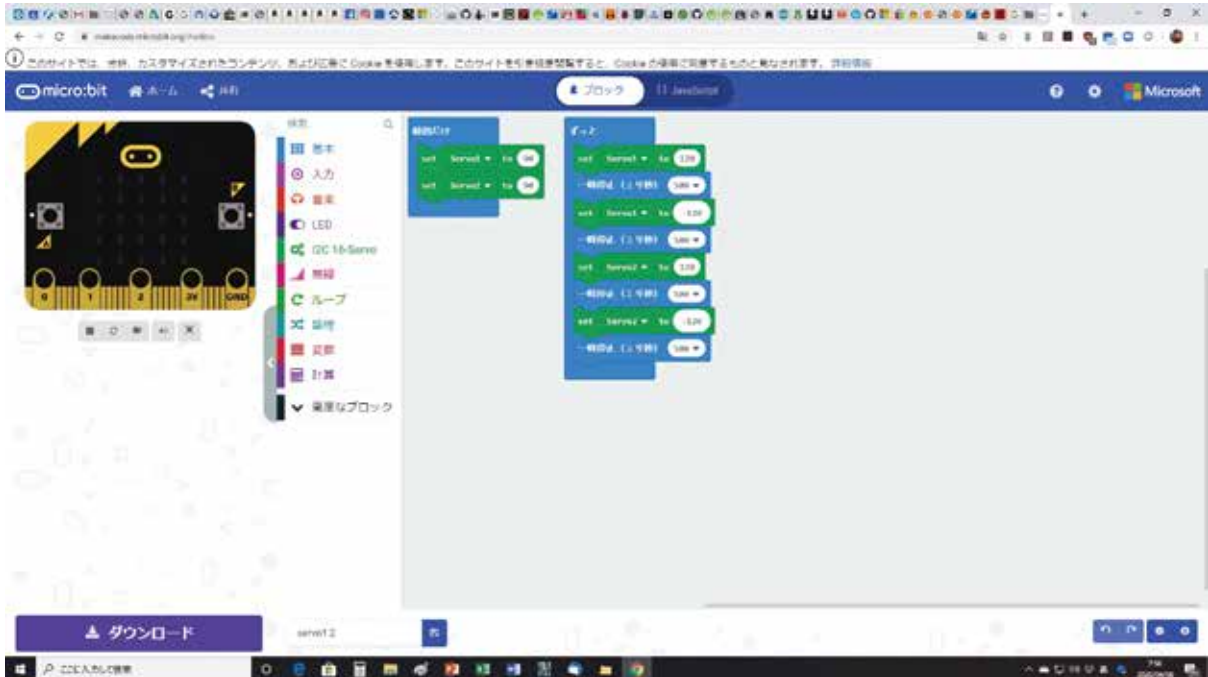


図 21. 二本脚 4 自由度の動作確認のための MakeCode エディタでのコード

実現できる。

この一連の処理の流れの流れ図で表すと図 20 のようになる。

図 20 は歩行の基本アルゴリズムであるので、対象とするマイコンボードやサーボモーターをより高性能なものに交換して量産タイプで実装する場合も、このアルゴリズムを適用することができる。

## (2) JavaScript のブロック・プログラミング言語による実装

Micro:bit のプログラム開発環境は Web ブラウザで <https://makecode.microbit.org/> にアクセスする形で利用することが可能である。開発言語は JavaScript なのでそのままコーディングも可能であるが、プログラミング初心者のためブロックを組み合わせるスタイルで開発することも可能である。

Micro:bit の JavaScript 開発向けに PCA9685 用サーボモーター・ライブラリ<sup>3)</sup>が提供されているので利用した。

また、各サーボモーターの位置出しを行うことで、サーボモーターの初期状態の角度を定義しておくことがその後の歩行の出発点となる。

更に注意が必要な点は、サーボブラケットの組み合わせによって位置出しされた角度からプラスマイナスの移動量が逆転する場合があるので、フレームとして組立後、四本の脚の上下で記録しておくことである。

4 個のサーボモーターが動作することを確認した JavaScript のブロック型プログラミング開発環境でのコードが図 21 である。

以上を踏まえて当初の設計通りの前後左右の移動が行える二足歩行ロボットの稼働部に特化した実装を確認することができた。

## おわりに

前後左右に移動する 4 自由度二足歩行ロボットの躯体と制御アルゴリズムの設計と実装ができた。本研究の成果としては、第 1 にはサーボブラケットの加工のしやすさと精度と強度を考慮してアクリル板の L 字アングルの利用を前提に加工工具ルータを導入したこと、第 2 には、片軸サーボモーターの回転精度を安定させるための両軸化を行ったこと、第 3 には、プログラミング初心者に学習しやすいブ

ロック型プログラミング環境を利用して前後左右の二足歩行アルゴリズムの実装ができたことである。一連の設計と実装は鳥取短期大学生生活学科情報・経営専攻1年生のグループワーク（1グループ5名で10グループ）によるPBL(課題解決型学習)の演習(9回分)としても確認できた。

動歩行の実装に向けては、逆運動で脚の運びを計算してプログラミングすることが考えられる。ロボット自体に歩行を学習して習得させるためにROS2 (Robotics Operating System, Version 2) の

利用や、学習も実機ではなくシミュレーションを高速PCで行わせることが考えられる。

#### 引用・参考文献

- 1) micro:bit 統合開発環境 <https://makecode.microbit.org/>
- 2) 野津伸治「四足歩行ロボットの設計と実装」, 『鳥取看護大学・鳥取短期大学研究紀要』, 第79号(2019), pp. 39-47.