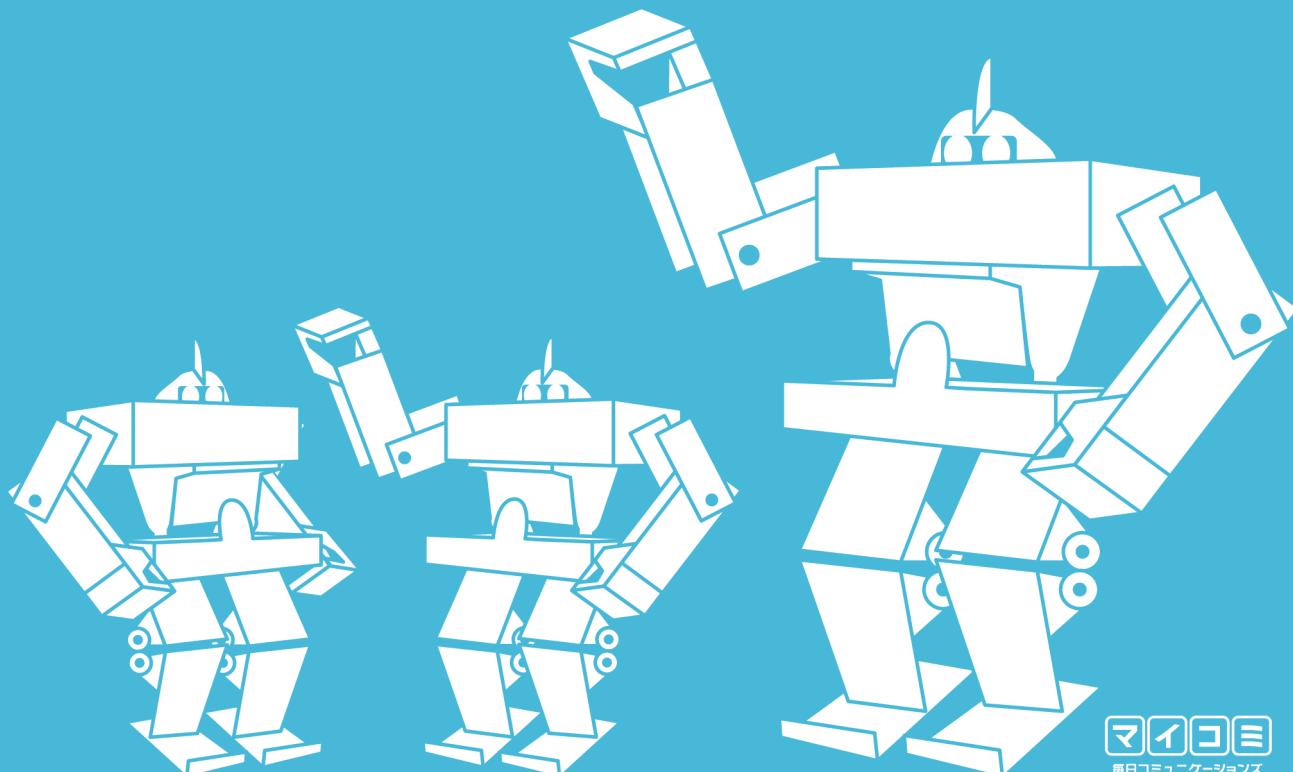
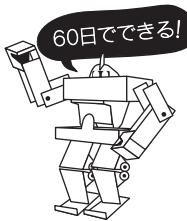


60日でできる!



吉野耕司 [著]





吉野耕司 [著]

読者サポートサイトのご案内

・読者サポートサイト

http://www2.plala.or.jp/k_y_yoshino/pen4/pen4_support/

本書で解説しているサンプルプログラム、設計図面、回路図は、すべて上記の「読者サポートサイト」からダウンロードできます。サンプルプログラムについては、1日の作業が終わる段階ごとにファイルを分けてご提供いたします。ご利用方法は、上記サイトにアクセスしてご確認ください。

免責事項

- ・本書に記載された内容は、情報の提供のみを目的としており、本書を用いての運用は、すべてお客様自身の責任と判断において行ってください。
- ・本書の内容に関してなんらかの保証をするものではなく、本書の記述の実行、プログラムコードの使用により発生した損失や損害、その他いかなる事態についても、弊社および著作権者、各ソフトウェアの製作・提供者は一切の責任を負いません。あらかじめご了承ください。
- ・本書に記載の記事、製品名、URL等は 2007 年 8 月現在のものです。これらは変更される可能性があります。
- ・本文中に登場する会社名・製品名等は、一般に各社の登録商標または商標です。本文中では ® ™ 等は明記しておりません。

はじめに

ロボットといえば、ずっと前からSFやアニメではお馴染みでした。1980年前後の産業用ロボットブームの後、しばらく下火になった時期もありましたが、ここ数年は、またペットロボットやパートナーロボットといった、日常的なシーンで見かける（あるいはそういう方向を目指した）ロボットが相次いで発表され、ある種のブームといってもいい状況もあり、ロボットというものに対する一般的な関心も高まってきたように思います。

「物を理解するのは、実際にそれを作つてみるのが一番」という考え方がありますが、この本では60日間で小さいけれども本格的な二足歩行ロボットを、模型のフルスクラッチの感覚で手作りします。そしてその製作過程を通して「現在ロボットといわれているもの一般の仕組みについて、なんとなく理解してしまおう!」というのがこの本の趣旨です。ロボットに興味があって、ラジコン、ガンプラ、電子工作、プログラミングなど、何か作るのが好きな人もしくは好きだった人には、特におすすめです。

最近は、PCや電子部品の高性能・低価格化などを背景として、新しいホビーの1分野として小型の二足歩行ロボットキットなども発売され、比較的簡単に組み立てて動かせる環境も整ってきましたが、上記の趣旨のため、電子部品などをバラで買いそろえ、ロボットのフレームは素材から加工し、制御ソフトウェアも自作するといった流れで進めて行きます。

こう書くと、大変なことのように思われる読者の方もいらっしゃるかもしれません、本書で取り上げる二足歩行ロボットは、筆者が趣味として、ラジコンの模型でも自作するような感覚で製作したロボットです。材料には、秋葉原や近所のホームセンターなどで買った部品や素材を使っています。手先の器用さは多少要求されますが、基本的には丁寧にプラモデルを作るような感覚で作れます。

気になる材料費については、いちおう形になる時点での**本体価格5万円程度**を目安に考えました（PCなどの道具類の費用は別です……）。作りながら楽しめる期間と、飽きたら部品をまた別のロボットに転用して

楽しめることを考慮すると、下町育ちの筆者の金銭感覚からいってもそんなに高くないと思います（ものの性質上、お金をかければ、いくらでもかけられます。また、筆者はしばしば、ロボットに関しては「財布のひもが切れている」といった類のこともいわれていますが、1台完成させたら、読者の方々にも、きっと筆者の気持ちはわかってもらえる！と信じています）。

「ただ作る」のではなく、仕組みを理解するための本なので、理屈は抜かずにつきの説明がつきます。専門的な用語を含むような書き方をすると、正確にはなっても、作って理解するという趣旨に沿う説明にならないと思うので、多少不正確でも、本質的な内容からはずれないことを目標として、できるだけ平易な言葉で説明するようにしました。といったわけで、学校で習うような知識については「高校卒業程度」で理解できる内容になることを目指しました。技能的なものでは、いわゆる普通の動く模型の工作の技術と、ワープロや表計算などのソフトを普通に使いこなせる程度のPCの知識程度を前提としています（ちなみに例として使うのはWindows XPのPCです）。

ロボットに関連する分野としては、骨格などを構成する「機械・構造」、モーターやセンサなどを動かす「電子回路」、そして全体を制御する「ソフトウェア」の3つが主なところです。筆者を含め、理科系の学校を卒業して、技術的な仕事で生計を立てている方でも、全部の分野に関連する仕事をしているというケースは少ないことだと思います。そうした方も、関係のない分野についてはやはり素人なので、どの部分も平易に書くのはいいことだと思っています。みなさんの得意な部分については、本の内容よりも高度なテクニックを盛り込んで発展させ、ロボット製作を楽しんでください。

製作期間を、作業の種類や所要時間の区切りのいいところで分割し、60日としています。ロボットのハードはそのままの製作時間ですが、ソフトのほうはコピーして使えるため、手入力する作業時間ではなく、説明の区切りで1日をまとめています。

実際の工作は、筆者の作業スピードで4時間前後を目安としていますが、かなり個人差があると思いますので、あまり本の日数表示にはこだわらずにマイペースで製作されればいいと思います。

読むだけでも得るところがあるように考えていますが、やはり、百聞は一見にしかずともいいますし、作って動かしてみると自然にわかってしまう事柄というのも多いと思いますので、この本のとおりにする必要はありませんが、作ってみることをおすすめします。では、はじまりです。

2007年7月

目次

60日でできる！二足歩行ロボット自作入門

1	日目	どんなロボットにしようか？	9
2	日目	設計はこんなふうにやっています	25
3	日目	I/Oボードの組み立て（その1）	45
4	日目	I/Oボードのチェック	60
5	日目	I/Oボードの組み立て（その2）	71
6	日目	ケーブル改造とマイコンボードの準備	86
7	日目	マイコンの動作確認をしよう	95
8	日目	LEDを点滅させてみよう	102
9	日目	LEDの明るさを調節してみよう	124
10	日目	ちょっとRCサーボで練習しよう	133

11 日目	フレーム加工をしてみよう(足)	141
12 日目	フレーム加工をしてみよう(足～脚)	164
13 日目	フレーム加工をしてみよう(脚)	175
14 日目	フレーム加工をしてみよう(脚)	183
15 日目	フレーム加工をしてみよう(脚)	189
16 日目	I/Oボードの組み立て(その3)	198
17 日目	PCからRCサーボを操作する(その1)	206
18 日目	PCからRCサーボを操作する(その2)	222
19 日目	フレーム加工をしてみよう(腰)	241
20 日目	軸保持版とRCサーボの組み立て	252
21 日目	脚の仮組みと各部の調整	263
22 日目	フレーム加工をしてみよう(腰、胴体)	276
23 日目	フレーム加工をしてみよう(胴体)	283
24 日目	距離センサで遊んでみよう	288
25 日目	フレーム加工をしてみよう(腕)	303
26 日目	フレーム加工をしてみよう(腕)	307
27 日目	配線しよう(脚・腕)	319
28 日目	配線しよう(胴体)	331
29 日目	ロボットハンガーを作ろう	342
30 日目	すべてのRCサーボを動かそう(その1)	346

31 日目	すべてのRCサーボを動かそう(その2)	358
32 日目	スムーズな動きにしよう	364
33 日目	関節パラメータ測定	380
34 日目	歩行のアルゴリズム	386
35 日目	逆運動学計算	393
36 日目	色を塗ってドレスアップ	406
37 日目	足先軌道生成	411
38 日目	オフライン計算モーションの再生	430
39 日目	通信機能を拡張	435
40 日目	Pen4号、ついに歩く	441
41 日目	歩行試験と調整	459
42 日目	後進モーションを追加	464
43 日目	旋回モーションを追加	469
44 日目	ジャイロを使えるようにする	479
45 日目	ジャイロを使って歩行を安定化	489
46 日目	補間モーションエディタを作ろう	494
47 日目	補間モーション再生	502
48 日目	補間モーション作成	507
49 日目	カウルの製作(その1)	514
50 日目	カウルの製作(その2)	519

51	日目 距離センサ用プログラムを追加	529
52	日目 最寄のものに近づいてパンチ	536
53	日目 自律行動プログラムにする	540
54	日目 加速度センサを追加する	552
55	日目 転倒を検知できるようにする	556
56	日目 「目」を付ける(ハード)	564
57	日目 「目」を付ける(ソフトとテスト)	575
58	日目 「目」を付ける(画像処理・認識)	584
59	日目 操縦システムの紹介	592
60	日目 今後の展開	604
<hr/>		
付録	調達用お買い物リスト	606
	二足歩行ロボット「Pen4号」完成形ギャラリー	631
	通信コマンド一覧	635
索引		636

どんなロボットにしようか？

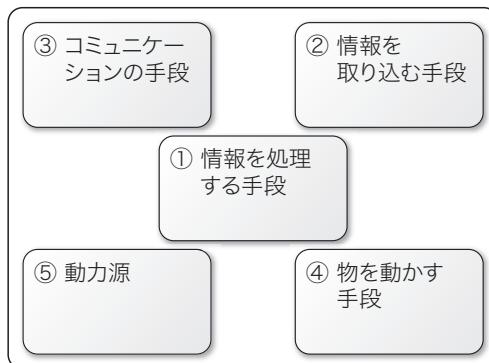
- ① ロボットシステムの構成
- ② ロボットの形
- ③ 設計コンセプト
- ④ 具体的には……
- ⑤ 主な材料
- ⑥ 主な道具

① ロボットシステムの構成

はじめにみなさんに質問します。ズバリ、ロボットとは具体的にはどんなものでしょう？

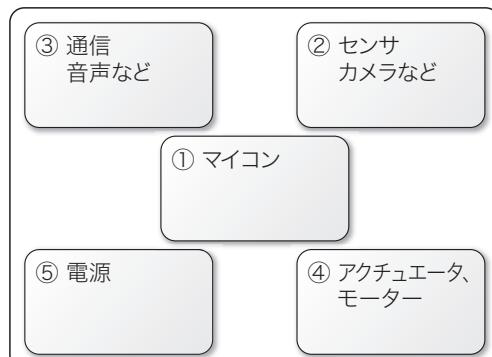
皆さんのイメージの大部分は「生き物の全体的な形か、もしくは部分的な形をまねて作った機械」で、「決められた動きをするもの」といったところにまとまるのではないでしょうか。そして「決められた動き」は、からくり人形や機械式の時計のように作った後からは変えられないものではなく、ソフトウェアで変えられるものをイメージしている方が多いようです。

そういった機械の仕組みを、それぞれの機能を示す言葉で簡単に図示すると、次のような構成になると思います。



「ロボット」の構成要素

これでは漠然としすぎているので、それぞれの構成要素に、もっと具体的なものの名前を入れてみましょう。



要素に具体的な名前を入れてました

こうすると少し形が見える気がしませんか（それぞれは通常隠れている部品なので、見える気がしない人のほうが多いことは思いますか……）？ それぞれの役割を人間の場合のたとえ話も交えて、順番に考えていきます。

① マイコン

いろいろなことを記憶したり、記憶の内容で計算し、いろいろな判断をしたりと、動きが固定されていない機械を作るのに必要不可欠です。ここでは「マイコン」という、身の回りの機械（家電とか、自動販売機とか、自動車とか）に組み込まれているコンピュータを挙げましたが、ロボットが大きい場合（内部に設置スペースを確保できる）や、無線LANなどを使う場合（ロボットの外に置いてコントロールに使う）は、PCでもかまいません。要は一般的にいう、電子式の計算機です（機械式の計算機でもいいですが、サイズ、処理速度、信頼性などの点で現実的ではありません。とはいえ「ディファレンスエンジン」にはロマンを感じます）。人間にたとえると、脳や神経に相当します。

② センサ

ロボットが「状況・状態」を知るために必要です。これで、「状況・状態に応じた動きをする」ことが可能になります。ロボット内部のことを知るためのものと、外部のことを知るためのものとに分けられます。現在のロボットのセンサでは、何かに触っているかどうかを知るためのセンサ、周囲のものまでの距離を測るセンサ、姿勢を知るためのセンサなど、いろいろなものがあります。人間の場合にたとえると、前者が「おなかが空いた」とか「今横になっている」とかの、体内的感覚。後者が「寒い・暑い」とか「明るい・暗い」など、環境に関する感覚に相当します。

③ 通信

ロボットが命令を受け取ったり、状態や状況判断結果を報告したり、製作中に動きを定めていないロボットの場合、後から何をさせたいのか指示しなければならないので、必要となります。無線・

ディファレンスエンジン：蒸気機関などで駆動する、19世紀頃に考案された純機械式（歯車でガチャガチャ動く）の計算する機械です。考え方的には現在のコンピュータの先祖にあたるようなもの（?）で、同名のSF小説（ハヤカワ文庫）がとても面白いです。

有線いろいろなタイプがあります。人間の場合は、声・話し言葉と聴覚、身振り・表情・文字（書き言葉）と視覚などのペアが相当します。

④ アクチュエータ

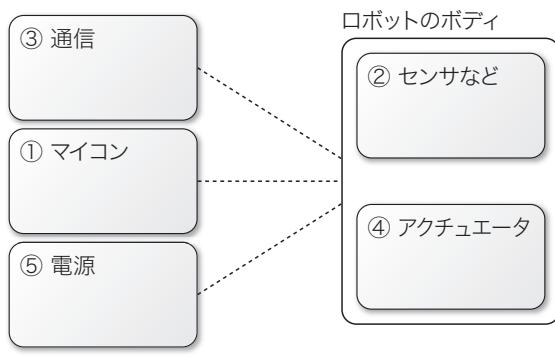
物理的にロボットのボディを動かすための手段として、何かしらの動力が必要となります。ロボットは「電子制御」が一般的なので、やはり、電気で動くモーターとの相性がいいです。そのほか、電圧をかけると含む水の量が変わり、その材質の体積が変わる性質でのものを動かそうという趣旨の、「人工筋肉」といわれるアクチュエータや、油圧、空気圧のシリンダーなどもあります。人間の場合は、骨と筋肉のペアが相当します。

⑤ 電源

マイコンも、モーターも、センサもみな電気で動くので、電池や発電機など、電気を出すものが必要になります。人間の場合はエネルギーを貯めておく専用の器官はありませんが、強いていえば食べものを消化して、各器官が利用できる形にする消化器官が相当するといえます。

ロボットのシステムとしてはこれら①～⑤を含みますが、ロボット単体でこれらをすべて備えているとは限りません。現在のロボットは、作られた目的によっていろいろあります。たとえば、すべてをロボットのボディに内蔵したタイプ（惑星探査用のロボットとか）、コンピュータや電源がボディの外に置いてあるタイプ（工場の生産ラインで使うロボットアームとか）、電源は外部で、コンピュータは大きいものが外に、小さいものがロボットのボディ内部にあるタイプ（有線式の水中ロボットとか）などです。

「仕組みの理解」という観点からロボットを製作する場合、ひととおりの要素を含みつつ、勉強の途中で力尽きたりしないように、なるべく簡単に作れるものがいいでしょう。そういう観点からは、下の図のように、ロボット本体になければいけないもの以外はすべてロボットのボディの外に置いてしまい、ケーブルで接続する形態が適しています。研究テーマのデータが採ることだけを目的とした、ちょっと前の研究用の二足歩行ロボットなどによくあるタイプです。

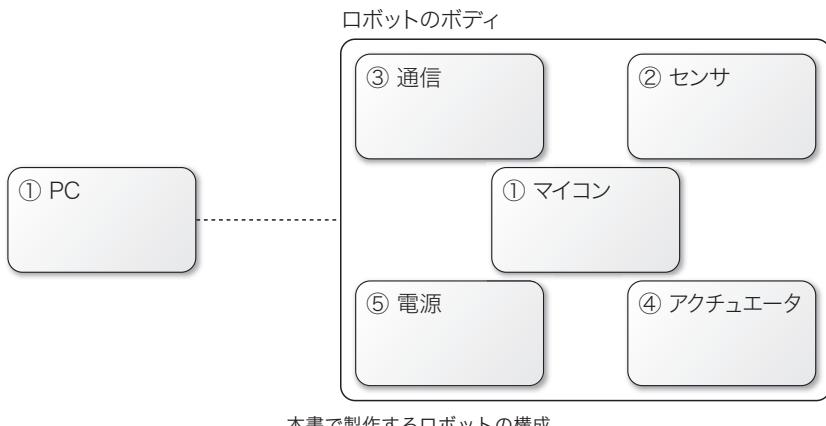


最低限のものをボディに積んだ構成

ただ、見栄えがよくないと、できあがった後で動かして遊ぼうと思ったときに、動ける範囲が制約され

るのが欠点です。

すると「やっぱりロボットといえば、オールインワンのタイプでしょう」ということになるのが人情です。そこで、本書で製作するロボットは、作るのが難しいわりに、性能的にはつましいものになってしまいますが、できあがったときに見栄えがして、遊んで面白いので、「すべてを内蔵している」タイプとします。ただし、電源を内蔵の電池のみとすると、動かせる時間に制約が大きくなり、プログラム開発中にバッテリ切れになるなど不便なので、外部電源も使えるようにします。また、内蔵できるマイコンの性能も制約が大きいので、PCからも有線で制御できるようにします。



余談ですが、最近の自動車や家電の中には、上記の要素を備えているものも見られます。そういったところが自動車や家電の「ロボット化」などといわれる所以でしょう。また、これらの技術をセットで、システムとして何かに適用する場合「ロボット技術を応用した」などといわれるようです。

2 ロボットの形

タイトルからして『二足歩行ロボット自作入門』という、二足歩行ロボットを作る本なので、これは決定事項です。といてしまえばそれまでですが、いちおう二足歩行タイプを選んだ理由を説明します。

ロボットの仕組みの理解を目的とするならば、別に二足歩行タイプである必要はないのですが、やはり多くの人は、ロボットといえば、二足歩行型というか、人間っぽい格好をしたヒューマノイドタイプを思い浮かべることでしょう。それにまあ、そのほかの地面の上を移動するタイプのロボット（車輪とかクローラーとかで走るタイプなど）と比べると、手間も材料費もかかりますが、やはり動きが多彩で面白いし、かっこいいほうが「作ろう」とか、「より深く知ろう」などといったモチベーションも高くなり、継続もしやすいうだろうということで二足歩行型としました。

ちなみに、本書で紹介する技術や部品を少し応用すれば、多脚歩行型も作れますので、昆虫や動物のよう

クローラ : 一般にいう「キャタピラ」のことです。ただし「キャタピラ」は登録商標なので、専門的には「クローラ」や「無限軌道」などといいます。

な形がお好みの方は、そういう形で製作されてもいいでしょう。そのほかのタイプに対しても、著者の（独断と偏見に満ちた）所見を述べておきます。

無重力状態用（人工衛星タイプ）

作ったら面白そうですし、手作りできないこともない気がしますが、無重力の場所まで持っていくのがかなり困難です。それに勝手にロケットとかを打ち上げると、今の時勢では、かなり怒られます。以上より実際に製作する教材ロボットには向いていないと思います。

空中用（飛行機／ヘリコプタータイプ）

テストのときに、たとえばちょっと制御ソフトにバグがあって暴走しただけで、全損状態になり、機体が失われてしまいます。それに、落ちる場所によってはいろいろ周囲に対しても危険ですし、自由に飛ばせるだけの広い場所が近所にある人も少ないので、これも面白いけれども不向きでしょう。

水中用／水上用（船／潜水艇タイプ）

これもテストのときに、潜ったまま浮かんでこなかったり、水上用でも沈没したりなど、機体を回収できなくなってしまうことが高確率で起こりえる点や、簡単にテストできる場所がない点、空中用と同様、不向きでしょう。

3 設計コンセプト

さて、ロボットの形が二足歩行型に決まったところで、多くの読者が作れて、なおかつ本書の目的（くどいようですが、「製作過程を通してロボット一般の仕組みをなんとなく理解してしまおう！」です）を達するためにはどうしたらいいか考えました。そしてこれから述べるような事項を設計コンセプトとして盛り込むこととします。

①簡単に手に入るものを使って作れる

ホームセンターや、秋葉原、ラジコンショップなどで、趣味として常識の範囲内（人によっていろいろ違いはありますが……）の費用で手に入るものを使って製作します。

何か新しいものを作ろうとした場合、作るものと要求性能を明らかにして、それに合わせて材料や、加工方法を決めます。要求性能を実現するために、たとえば材料の研究からはじめる……なんて場合もあります。

しかし本書の場合は、ロボットの製作に使えそうなものにあらかじめ見当をつけて、それを使ってできるロボットを設計するという、逆の流れでいきます。

②「おうち」で作れる

具体的には、日本の都市部の住宅事情を考慮し、製作中も、できあがったものも場所を取らない

水上ロボット：生物進化の足跡を辿る……という意図ではありませんが、筆者は最初に魚型のロボットで開発しました。水に浮くためパワーと自重のバランスがシビアではない点や、少ない関節で一応動くものが作れるあたりでハードルが低いため、全長1m程の魚型ロボットから作りはじめました。ただ、テスト場所の確保に苦労したり、沈没しそうになったり……苦労しました。

(机の上でできる) ことや、大きな音が出るような工作の方法を避けるなどの点を考慮しました。

③現実的な日数で作れる

本書のタイトルにも含まれるコンセプトです。なんなくですが、60日（2ヶ月）くらいで形になるようならば、作れる読者も多いだろうということで考えました（ちなみに、著者が「二足歩行ロボットを作りたい！」と思って勉強をはじめてからできあがるまで、すごく時間がかかりましたが、本に沿っていけば簡単です。たぶん……）。

4 具体的には……

ここまで話をまとめ、手に入るモーター、マイコン、電池、材料などの機能・性能を考慮して、検討や実験を重ねた経験を元に、下記のようなロボットの概略仕様を引き出しました（ここに落ち着いた理由は後ほど説明しますね）。

仕様の概略

・基本性能

1秒間に2歩程度のペースで二足歩行できる

倒れている状態から自分で起き上がる

周囲の障害物の状況を調べられる

自分の姿勢がわかる

・寸法と重量

寸法（身長）：35cm

重量：1.5kg

・装備の種類と数量

関節の数（自由度といいます）

脚：2（片脚あたり）×2（両脚）

腕：3（片腕あたり）×2（両腕）

腰：2

首：1

合計：13個のRCサーボモーターで駆動する

距離センサ：1個

ジャイロ：1軸

・その他

後からセンサなどを追加できること

本書の範囲を最後まで作った時点で、電力、スペース、マイコンのI/Oなどに余裕を残し、拡張可能な状態とすること

本書では「手に入る材料で作れるロボットを作る」というスタンスでロボットを作りますので、ロボットの設計よりも先に、部品や材料、そしてやはり同じ制約がある道具の話からはじめましょう。

5 主な材料

ここでは「筆者がロボットの構成要素として考えるもの」が、実際にどういう材料や部品なのかを紹介します。「主な」と断るからには、主ではないけれど必要なものは色々ありますが、だいたいイメージがつかめるのではないかと思います。「今日すぐに買ってください」というわけではないので、ここでは気楽に眺めてみてください（まあ、3日目から使うものもあるので早いに越したことはありません）。

マイコン、センサ、アクチュエータ、電源、構造材料（ロボットのボディの材料）の順に話を進めていきます。前節の構成要素には「通信」がありましたが、これはマイコンとPCに含まれているので独立したセンテンスは設けませんでした。

PCについては「ロボットを制御する構成要素」という観点からするとロボットの一部なのですが、マイコンのプログラムを開発したり、CADでロボットを設計するのに使ったりするという面では、ロボット製作のための道具であるともいえます。どちらかというと道具としての比重が高いので、本書では道具の節に登場させようと思います。

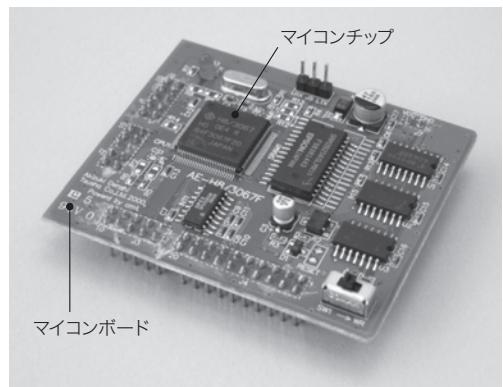
▶ マイコン

マイコンのチップ単体で購入し、マイコンの周辺回路を載せた基板（マイコンボード）に取り付けて自作する方法もありますが、「ロボットの仕組みを学ぶ」という本筋から逸れる上、製作のハードルが高いので、本書では用途に沿ったマイコンボードキット（マイコンのLSIとマイコンを動かすのに必ず付ける回路の類が1枚の基板に載っている形のものです）を買って使うこととします。

本書執筆時点で、ホビー用途として妥当な値段で手に入るマイコンボードキットはいろいろあります。値段、性能、入手しやすさ、筆者の好みなどを総合的に考えつつ、特にC言語でプログラムを作りやすい点を評価して、ルネサステクノロジー製の「H8/3067F」というマイコンチップが載っている、秋月電子通商製のマイコンボードキットを使うこととします。マイコンチップが載った、こんな形の基板です（ちなみに「基板」のことを「ボード」ともいいます）。

「マイコンボード」と「マイコンチップ」という言葉を区別して使っていますが、「マイコンチップ」という場合は、写真中央の部品の類を指します。マイコンの周りの黒い部品のような半導体部品一般を指して単に「チップ」ともいいます。

ちょっと話が逸ましたが、写真中で「マイコンチップ」と説明が付いている部品がH8/3067Fというマイコンチップです。CPU、ROM、RAM、A/Dコンバータ、カウンタタイマなどの機能を持った部品です。普通のPCに機械の制御用のボード類を足して、CRTやキーボードなどに関連する回路を

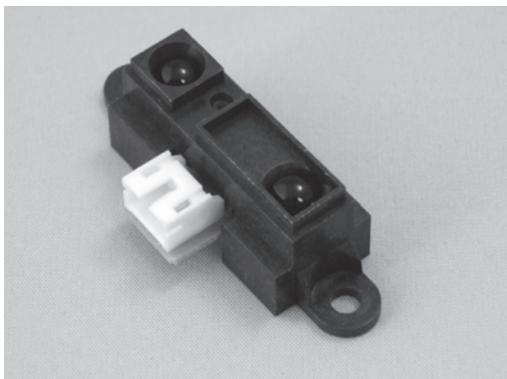


マイコンボードキット

削った、という感じの機能です。

▶ センサ

「センサ」と一口にいっても非常にたくさんの種類があります。マイコンと接続したり、ロボットに搭載したりすることが非常に難しいものもありますので、「ロボットの外のことを知るセンサ」(外界センサ)、「ロボット自身のことを知るセンサ」(内界センサ)ともに、やはり自作ロボットに載せる観点から扱いやすいものを選びました。外界センサとしては距離センサを、内界センサとしてはジャイロを載せます。



距離センサ



ジャイロ

距離センサを搭載することによって、「操縦オンリー」のリモコン式ロボットではなく、たとえごくわずかでも、ロボットに状況を判断させて、行動に反映させることができ、「自動的に行動する」という意味でロボットらしいロボットになります。

また、ジャイロを搭載することにより、歩行の動作が「あらかじめ決めた動作の完全な再生」ではなく、歩いている場所の微妙な凹凸を感じて、歩き方に反映できるようになり、これまたロボットらしくなります。

ロボットに搭載するセンサを簡単に紹介します。

距離センサは、赤外線を使って距離を測るセンサです。スポットライトのような形で、赤外線を出す部分と、その赤外線の「スポット」の方向を読み取る部分がセットになっています。3角測量の原理で距離を求める仕組みです。シャープ製の「測距センサユニット GP2D12」という機種を使います。

ジャイロは、回転運動を測るセンサで、ラジコン飛行機の揺れを抑えるための部品として売られているものを使います(GWSというメーカーのPG-03という機種です)。ジャイロと呼ばれるセンサは、もともとはコマ(地球ゴマとかベーゴマとかのあれです)の回転軸が元の方向を向き続けようとする性質を利用していました。ちなみに、このジャイロは「圧電振動式」と呼ばれる方式のジャイロでコマは入っていません。

そのほか、本書では機能拡張の一環としてロボットの傾きを測るセンサとカメラも使っています。

▶ アクチュエータ

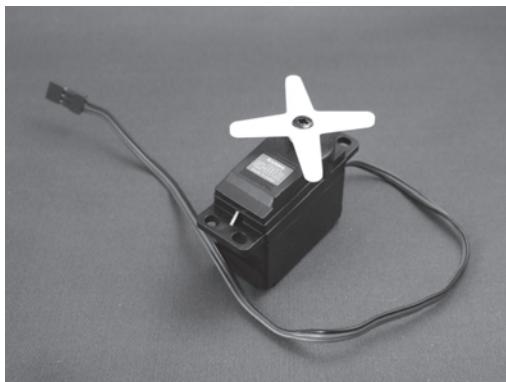
ロボットを動かす部品を、専門的な呼び方をすると「アクチュエータ」といいます。「アクチュエータ」とは「動かすもの」という意味で、実際のところ現在のロボットでは直流電動モーター、交流電動モーター、リニアモーターなどの電気で動くモーターと思ってほぼ間違いないですが、油圧シリンダー、空圧シリンダーや人工筋肉など、形式が異なる「動かすもの」を使う場合もあるので、広く「アクチュエータ」という言葉を使うようです。

さて本書のロボットでは、先に述べた大きさと重さの中に、システムをまとめることができ、個人で入手できるアクチュエータが必要になります。となると、直流の電動モーターが唯一の選択肢にならうかと思います。

直流のモーターを使う線で考えてゆくと、効率よく力を取り出すためには、1分間に1万回転くらいのオーダーでモーターを回し、ギアなどで減速・増力してロボットの関節を動かす必要があります。

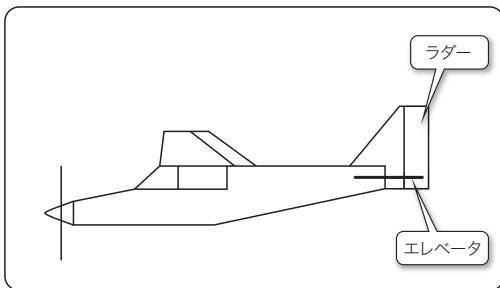
また、ミニ四駆などでご存知のとおり、直流モーターは電池をつなぐと一方向に回り続けるだけなので、ロボットの手足を動かすのに使おうと思った場合、制御回路を付ける必要があります。

直流モーター、ギア、制御回路を別々に作るなり、買うなりしてロボット用のアクチュエータを構成する方法もありますが、これら3つの要素が1つにまとまっていて、しかも手ごろな値段で買える商品がラジコン用のサーボモーターとして売られていますので、本書ではこれを利用します（ホビーロボットを自作される方々の間ではラジコン（RC）用のサーボモーターを縮めて「RCサーボ」と呼ばれています。ラジコンをされる方々の間ではただ「サーボ」と呼ばれています）。



RC サーボ

ラジコン模型での使われ方としては、車では、ハンドルやアクセルのコントロールをするために、飛行機では、エンジンのスロットルやラダーやエレベータなどの舵面を動かす用途に使われます。



ちなみにラジコン飛行機でラダーやエレベータとはここのこと

実際に購入するとなると、ピンキリで高いのも安いものもあります。本書では安くて手に入りやすいものという原則にのっとり、筆者が知っている中で一番安いサンワというメーカーのSX-101Zという機種を主に使います。腰に使うRCサーボは、SX-101Zでは力不足なので、やはりサンワ製のERG-VRとERG-VBという機種を使います。首は、力はなくても軽い機種が適しているのでGWS製のPICOという機種を使います。

▶ 電源

広く電源というと、身近なところでは乾電池や、自動車のバッテリなどの電池、屋台などでよく使われているエンジン式の発電機、そして光から直接電気を起こす太陽電池など、いろいろな種類がありますが、以下の選択基準から、Ni-MH（ニッケル水素）電池を使います。

① 性能

本書のロボットに積める大きさと重量で、必要なパワーが出せるか？

② 入手性

簡単に買えるか？

③ ランニングコスト

なるべく安いこと

入手可能なNi-MH電池にも、さまざまな形や性能のものがあります。本書ではロボットのサイズと消費電力から判断して1.2V 1100mAhの電池が5本パックにされた形で、ラジコン用として売っていた6Vの電池を使うこととします。

技術の進歩とともに、同じサイズでも新型のほうが容量は増える傾向にありますので、製作時点できれども大容量で同サイズ以下の電池が手に入れば、それを使ってもかまいません。

また、これよりも容量が少ないものでもかまいません。ロボットが動く時間が変わったり、起き上がったりの特にパワーとバランスに関する動作ができなくなっ



6V 1100mAh の Ni-MH 電池

たりと、影響はありますが、歩けなくなることはないと思います。

▶ 構造材料

実際にロボットを作ろうと考えると、ロボットの骨格というかボディというか、ロボットらしい形を作るための材料が必要になります。ロボット用の材料に要求される基本的な性質は、軽くて丈夫で、安くて、手に入りやすく、切ったり貼ったりの加工がしやすいもの……といったところです。

ちなみに筆者の「手に入りやすい」は「ホームセンター、模型屋、秋葉原のパーツショップ」のどれかで買えるというのとほぼ同義語です。この4カ所で買えるものを数年にわたりいろいろと試した結果、プラモデルと同じプラスチックの板（以下、プラ板：タミヤから「プラバン」として売られている）に落ち着いたので、本書のロボットでも使います。

プラ板は、材料の強度（同じ形の部品を作って引っ張ったとき、どれだけの力に耐えられるか……といった材料試験で求められる数値）が低いので、気を付けて形を決めないと、すぐに壊れるロボットになってしまいます。しかし、柔らかいぶん加工が簡単で、形を作りやすいので筆者は好んで使っています。

お店では下のような袋に入って売られています。本書では、組み立てるときに寸法に1mm以下の端数がないことと、単体でいちおう形が保てる程度の強度を持っている点を重視して、1mm厚のプラ板を使います。



1mm厚のプラ板

ここまでで、主な部品と材料の紹介は終わりです。ご覧のように「電子工作」「ラジコン」「プラモ」の材料の組み合わせです。この全部を趣味としている方は比較的少ないかと思います。筆者の複数の知人からの話を総合すると、電子工作の方々はプラ板で立体構造を作るような工作について、ラジコンの方々はマイコンやソフトの部分について敷居が高いと感じられているケースが多いような気がします。何をするにも「はじめて」はありますし、まあ、やっているうちに慣れますので、これを機会にぜひトライしてみてください。

製作ではこのほかにもいろいろな部品や材料を使いますが、ここで一気に紹介すると、読者の方も退屈されることだと思いますので、それらについては、実際に使う場面で紹介することとします。また、実際に買い出しに行くときに便利なように巻末に写真と説明付きの買い物リストを用意しましたのでご利用ください。買い物リストはPDFでダウンロード提供もしています。

6 主な道具

さて、主要な部品・材料の紹介が終わりましたので、ここからは本書で使用する主な道具を紹介します。ロボットは電気・機械・ソフトの要素を含むので、それぞれの分野の道具を使います。道具自体は家庭で使えるものなので、簡単なものばかりです。使い方はそれぞれ製作で使用するときに詳しく説明しますので、ここでは写真と簡単な説明で、だいたいどの程度のものかイメージをつかんでいただければと思います。

▶ 機械関係

機械関係の工作的内容は、フレームの加工というか、プラ板の工作です。そういうわけで、道具はプラ板を切ったり穴をあけたりするものになります。1mm厚のプラ板は、カッターで傷を付けて折るという方法で切れます。穴はピンバイスという道具であけます。



カッターナイフ



金尺



シャープペン



ピンバイス

プラ板同士の接着にはラッカーうすめ液（シンナー）を使います。プラスチックを溶かして付けるので、うまくやれば、はじめから1つのものであったかのような強度で接着できます。



接着剤

RC サーボをフレームに固定するためには両面テープを使います。



両面テープ

プラ板の工作は、まったくのところ「ガンプラの改造」のノリです。本書の中ではあまり「飾り」は付けていませんが、お好きな方はプラモのつもりでいろいろ付けて遊んでもOKです。

ネジ止めしてあるところも何所かがあるので、ドライバーとボックスレンチも使います。



精密ドライバーセット



ボックスレンチ

▶ 電気関係

電気関係の工作（電子工作）の内容は、基板の配線と、基板とRCサーボ間の配線など、配線作業です。よって、使う道具は半田ごて関係、電線の被覆をむく道具（ワイヤストリッパーといいます）、配線のチェックに使うテスターなどです。使う電線がちょっと細いので、ピンセットなども使います。



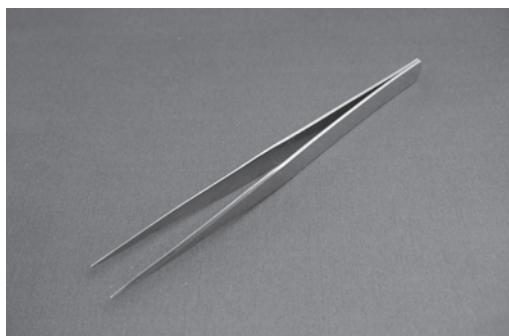
半田ごて



ワイヤストリッパー



テスター



ピンセット

▶ ソフト関係

その昔「パソコンもソフトなければただの箱」などといわれていたこともありました。マイコンも動かすためにはソフトがいります。マイコン用のソフトはPCで作りますので、道具としてPCが必要です。また、ロボットのテストや、H8マイコンには重すぎる計算もPCのソフトで行います。

そういうわけで、PCと、マイコンやPCのソフトを作るための開発用ソフトが道具として必要となります。これまた代表的なものを紹介しておきます。

PC

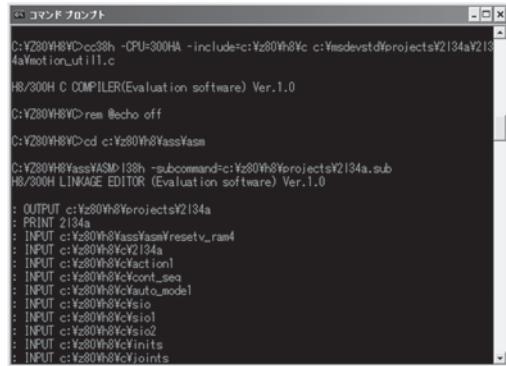
自分で使えるものならば何でもいいともいえるのですが、たとえばMacとか、PC9801とかを

使おうと思った場合、開発用のソフトから作らなければならず大変なので、本書では、いろいろと環境が整っている**Windows XP**が入ったPCを使います。Windows XPが問題なくインストールできているPCならばOKですが、参考までに筆者が使っているノートPCのスペックを示します。

CPU	Mobile Duron-1GHz
メモリ	512MB
ハードディスク	40GB
OS	Windows XP SP2 Home Edition

マイコン用のコンパイラ

コンパイラとは、たとえばC言語のような人が読める形のテキストファイル(元になるコードという意味で「ソースコード」といいます)をコンピュータが読める形のコード(2進数なのでバイナリコードもしくは、ただバイナリと呼ばれます)に変換するためのソフトです。この変換処理にはいくつかの段階があるので、その段階ごとにコンパイラの一部を構成するプログラムがあります。



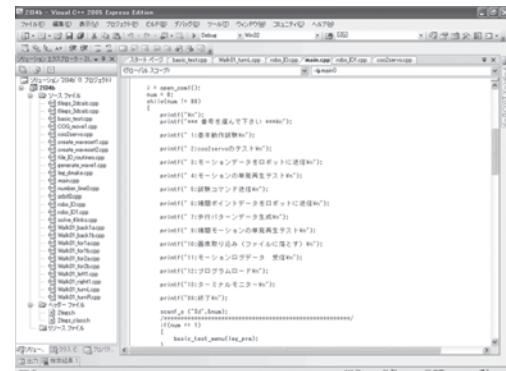
H8用コンパイラの実行画面

マイコンの機種ごとにコンパイラにも種類がありますので、本書ではマイコンボードキットに対応している、H8用のものを使用します。

PC用のコンパイラ

Windowsと相性が非常にいいこと、
無料で使えることから、Microsoftの
Visual Basic 2005 Express Edition
(長いので以降VBと呼びます)と**Visual
C++ 2005 Express Edition**（こちら
も以降VCと呼びます）を使います。

VCは、マイコン用のプログラムを作る道具としても使います。



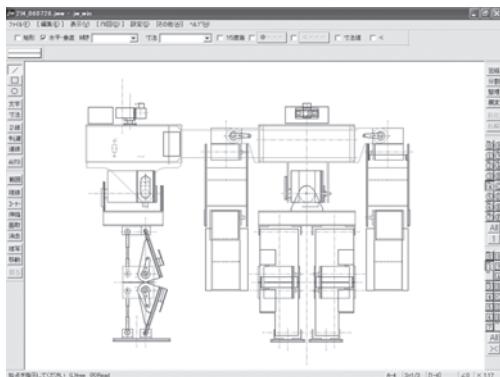
VC の画面

その他

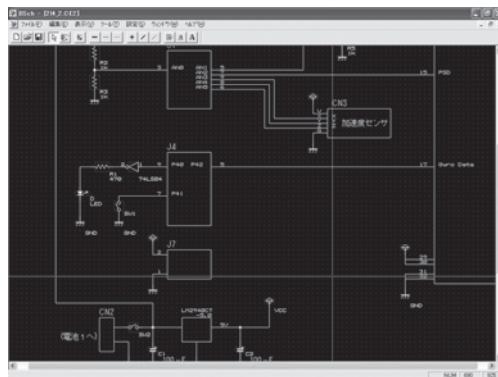
マイコンのプログラムをPCからマイコンに書き込むときに使うソフトや、PCとマイコンの間で通信するときに使うソフトなどがあります。

CAD

CADとはコンピュータで何かを設計するときに（いい換えると図面を描くとき）に使うソフトの総称です。読者の皆さんは本書の範囲では「書いてあるとおりに作っていく」ことが前提となるため必要ありませんが、本書の範囲を終えた後でロボットを発展させていくとき、図面を書く必要も出てくると思いますので、いちおうここで触れておきます。



建築用図面を描くためのフリーソフトの CAD。
機械関係の図面を描くときに使用



簡単な回路図を描くためのフリーソフトの CAD

明後日（3日目）には実作業に入りますので、それまでに必要なものはそろえておいてくださいね。巻末付録は一括購入にも分割購入にも利用できるように整理したので、ご活用ください。

全体の構想／イメージ、使用する材料／部品そして道具が決まりましたので、次は製作するロボットのイメージを具体化させます。ただし、製作の前に詳細まで説明しきろうとすると、あまり関係のない会議に呼ばれたときや、難しい講義を聞いたときのように眠くなってしまうことが予想されます（笑）。

「作ること（手を使うこと）を通して理解する」のが本書の趣旨なので、次の章も概要の説明と製作の目標を示すにとどめて製作につなげていきます。

2 日目

構想 設計 調達 電子工作
プログラミング 工作 テスト

設計はこんなふうにやっています

- ① 悩んでみるのが面白い
- ② 機械の設計
- ③ 電気系の設計
- ④ ソフトの設計

① 悩んでみるのが面白い

ここでは、前章で決めた仕様に向けて具体的にロボットの制御回路を設計します。「設計」といっても読者のみなさんに設計していただくわけではなく、この本のロボットがどういう考え方でこの形になったかのプロセスを知っていただければと思います。

実際に筆者が新しいロボットを作るとときは、機械、電気、ソフトの各パートを同時に、あーでもない、こーでもないとアイデアを頭の中で考えつつ、たくさん紙にメモやスケッチを書いて、システムが成立するか検討します。そして、成立しそうだと思ったら、具体的に電子部品のデータシートや機械部品の寸法などを確認しながら「図面」を描き、計算できるものはして、本当に大丈夫か確認します。

途中で「これは、やってみないとわからないなー」という項目が出た場合は、その要素だけを取り出したものを作り、動かして確認します。そして確認の結果を反映してまた設計を進めます。「こんな感じになる！」と思うところまで設計が固まったら、もしくは「後は作りながら考えよう！」というところまで来たら作りはじめます。筆者はロボットを趣味で作っているので、普段、ここから先は自分の興味の向くままに作っています。

ただ、いつもの調子で作っては「たどっていくとロボットができあがる」本にならないので、機械→電気系→ソフトの順番で、「やってみないとわからない」ことはなしで（つまり筆者の試行錯誤の苦悩はあまりお見せせず）、きっちと設計を説明していきます。

② 機械の設計

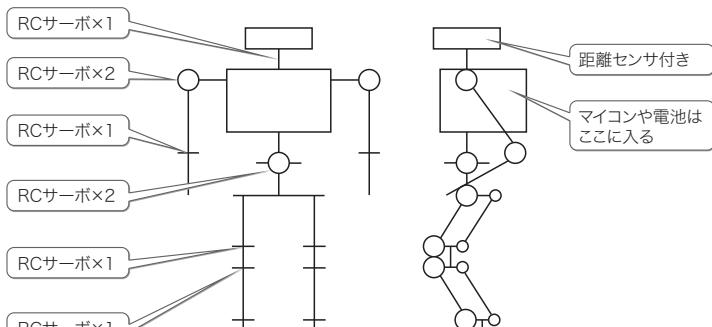
ここでは、ロボット全体のコンセプトに向けて、ロボットのフレームを設計します。ロボットのフレーム

とは、外から見えるロボットの形をなしている部分です。機能・性能を考慮しつつ、自分の気に入ったデザインにまとめます。

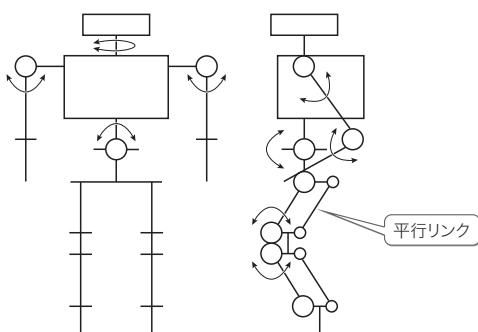
▶ フレーム設計の留意点

- ①すべての部分は必要十分な範囲で「軽くて丈夫」であること
- ②二足歩行や、そのほかの実現したい動きができる関節の配置になっており、かつそれぞれの関節は、十分な動ける範囲（関節可動範囲といいます）を持っていること
- ③マイコンや電池などを積み込むのに十分なスペースがあること

以上のようなことに留意しながら、ロボットのだいたいの形を決めたのがこれです。



フレームのイメージ図（その1）



フレームのイメージ図（その2）

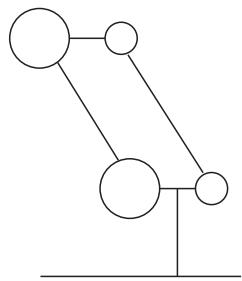
その1は、マイコンや電池、センサやRCサーボの配置と個数を示しています。その2では、各関節がどの方向へ動けるようになっているかを示しています。このように合計13個のRCサーボを配置しています。「ヒューマノイド型ロボット」といっても、ご覧のように関節の数（だいたいモーターの数と同じ）は非常に少ないです。

ちなみに、ロボットの形や大きさ、そして重さは、RCサーボの寸法と性能によって決まります。たとえば腕に付いているRCサーボは、脚のRCサーボにとってはただの重りになってしまふので、脚が支えられる重さまでしか数を増やせません。

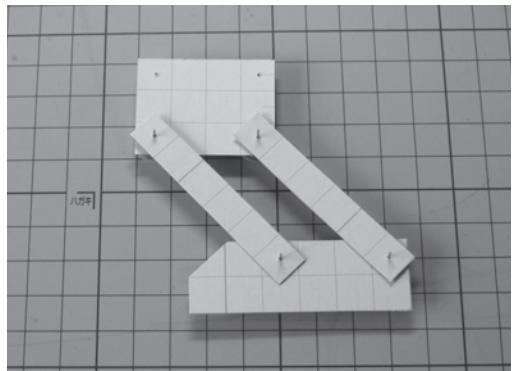
そんなわけで、ある程度の速さで歩けて、安いRCサーボで構成できるバランスポイントとして「13個」という数字に落ち着きました。

▶ 平行リンクと脚のしくみ

脚には「平行リンク」と呼ばれる機構を採用しています。モノを見てしまえば特にそれ以上の説明はいらない簡単な形です。この図はすねの部分を抜き出したものです。次の写真は、どのように動くかをわかりやすく示すため、同じように動く模型をボール紙で作ってみたものです。この全体を平行リンクといいます。そして、それぞれの要素を「リンク」（英語で「連結するもの」の意味です）といいます。リンクが平行につながっているので、見たままに平行リンクと呼ばれています。

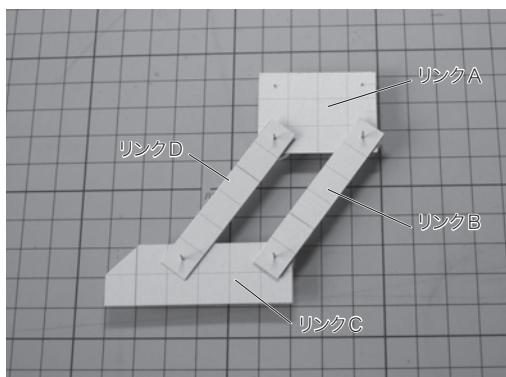


すねの部分



すねの平行リンク

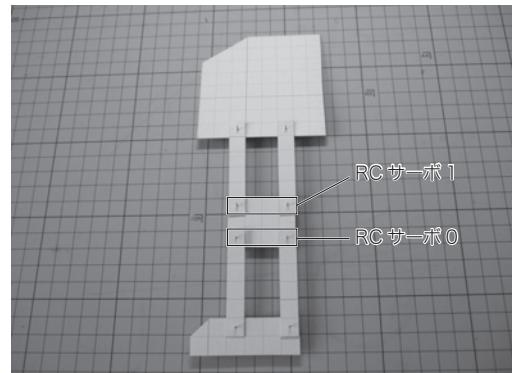
簡単でよくわかるので、みなさんも同じ模型を作って手で動かしてみることをお勧めします。ちなみにこの模型は、関節の部分が画鋲でとめてあり、自由に動くようになっています。



平行リンクを動かしてみる

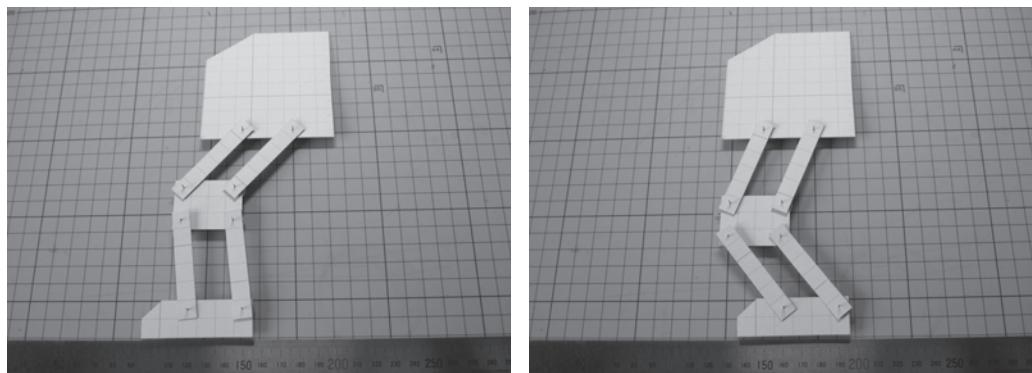
リンクAとCのように向かい合ったリンクが、どのように変形させてもいつも平行になるという特徴があります。

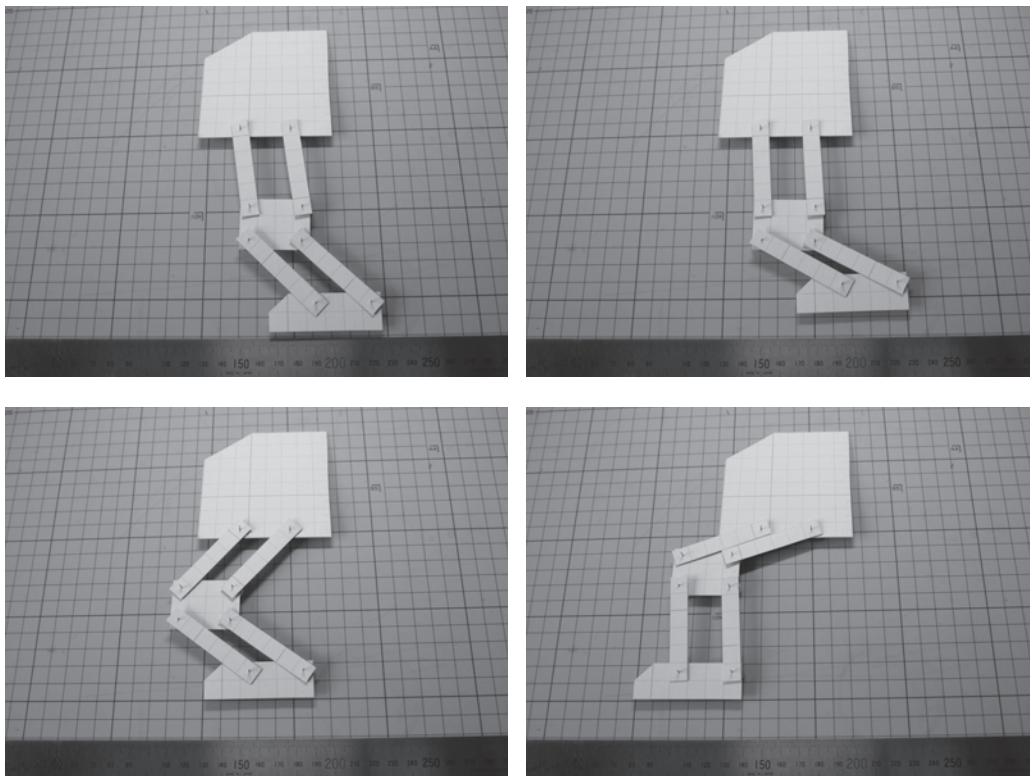
本書のロボットは、ひざの上と下に平行リンクがあり、ひざのところに、上の平行リンクを動かすRCサーボ1、下の平行リンクを動かすRCサーボ0が付いています。



胴体まで含めた模型

次の写真は、本書のロボットが歩くときに、どのように脚を動かすかを示しています。このように、脚をどの位置に持ってきてても、いつも足裏の面が地面と平行になります。

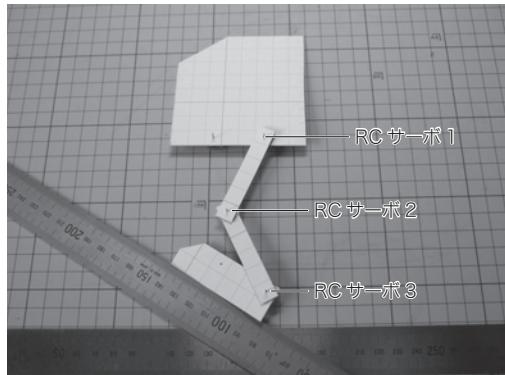




今まで何機か二足歩行ロボットを作っていましたが、水平な地面の上を歩かせるだけでもけっこう面白かったです。一方、関節を多くして試したこともあるのですが、水平ではない場所を歩かせるのは、制御ソフトウェアの開発などが容易ではないということがよくわかりました。

ですので、本書では割り切って、このように片脚あたり2つのRCサーボで、そこそこの速さで二足歩行できる簡単な機構を採用しました。

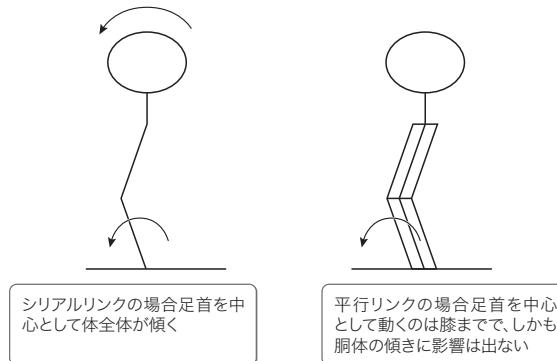
参考までに、平行リンクを使わない場合の例も示します。このように足裏の面を自由な向きにできる代わりに、RCサーボが1つ増えます。そのわりに、水平な場所を歩く場合は、平行リンクを使う場合とほとんど同じ動きです。両脚で2個のRCサーボが増え、100gほど重くなり、電池の減りも早くなります。



平行リンクを使わない場合（こういうのを「シリアルリンク機構」といいます）

後は、これが一番大きな理由ですが、それぞれの関節はマイコンから目標の角度を与えられて動くようになっているのですが、ロボットが立ったり、歩いたりしている状態で、各関節に体重がかかると、目標の角度からずれた位置になります。安いRCサーボは、軸受けの機械的なガタ（遊び）が大きく、力がかったときの目標からのズレも高価なRCサーボと比較して大きい傾向があります。

たとえば足首のRCサーボで3度の誤差が出たときの重心のズレ（誤差）を、平行リンクとシリアルリンクで比較すると、シリアルリンクのほうが大きく影響がでます。二足歩行ロボットは、重心の制御がキモなので、安いRCサーボを使っても安定して歩けるように平行リンクを採用しました。



感覚的にはこんな感じ

実際に計算してみましょう。だいたい、本書のロボットに近い値として、仮に以下の数字を使って計算します。

重心までの高さ H : 150mm

各リンクの長さ L : 50mm

角度誤差 θ : 3°

重心移動量 Δx

シリアルリンクの場合は、足首を中心として体全体の傾きが変わるので、重心移動量は以下となります。

$$\Delta x = H \times \sin \theta = 150 \times 0.0523 = 7.85\text{mm}$$

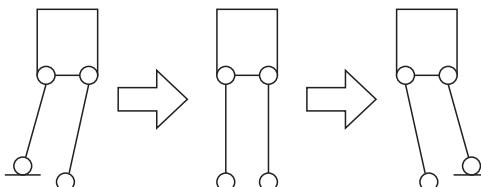
平行リンクは、足首からひざまでの傾きが変わるだけなのでこうなります。

$$\Delta x = L \times \sin \theta = 50 \times 0.0523 = 2.62\text{mm}$$

このように3倍近く重心位置の制御がよくなる効果が出ます。こうして割り切った設計にすることで大幅な制作費のダウ^{*}ンもできたというわけです。式の中にsinという記号がでていますね、高校の数学で習う三角関数というものです。忘れてしまった……とか、知らない……とかいう方は、本のとおりに作るぶんには差し支えないのですが、自由にロボットを発展させよう!!というときには必要ですので、学校の教科書ではなく、社会人向けの教養を深めるための本などで、高校までの数学は身につけておくといいと思います。

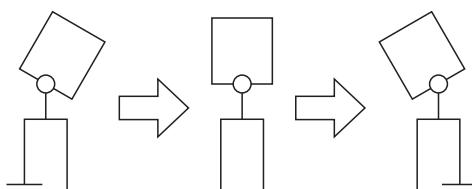
▶ 重心移動の方法

本書の二足歩行ロボットも、人間が歩くときと同じように左右の足に交互に重心を移しながら歩きます。有名な自動車メーカーや電機メーカーのロボットは、腕や首などを省いた簡単な、前から見た図で説明すると、ざっくり下のような感じで重心移動を行います。



一般的な二足歩行ロボットの左右方向の重心移動

このように動くために、図中で○印の部分に合計4個の関節（とモーター）が付いています。一方、本書のロボットは次のように重心移動を行います。



本書で作るロボットの重心移動

このように、重心移動のために1つの関節しか使わないため、モーターが少ないぶんロボットが軽くなり、また重心移動のための制御（コンピュータがすべきこと、すなわち、製作する人が書かなければならぬプログラム）が簡単になるので、このような方式を採用します。

制作費のダウ^{*}ン：本書執筆時点では、SX-101Zは1個1,400円くらいで買えましたが、ホビーロボット用の高性能なRCサーボは、1個6,000円～15,000円くらいします。これを3つぶん節約できたと考えると大きいです。

ちなみに、人間の場合は両方の組み合わせで重心移動しているようです。

▶ 重量の見積もり

使う部品を決めたので、ロボットがだいたいどのくらいの重さになるか見積もります。この情報は、使う予定の電池とモーターで動かせる範囲に入っているかチェックするために使います。

移動能力がある機械（いろいろな乗りものなど）について広くいえることですが、たとえば車だと、エンジンの馬力のわりに重くてスピードが出ないとか、飛行機だと馬力のわりには重すぎて飛べないとか、そういう形で、馬力と重さのアンバランスの影響が出てきます。

	品名	質量	備考
1	RCサーボ SX-101Z	460	(×10個)
2	プラ板、プラ棒	322	プラ板×3枚、プラ棒×2袋
3	6V Ni-MH電池	106	
4	マイコン&I/Oボード	95	
5	RCサーボ ERG-VR	60	
6	RCサーボ ERG-VB	60	
7	ネジ	45	
8	USBケーブル	30	
9	電線	24	15m
10	半田	20	
11	センサ	20	
12	シャフト	20	
13	ボールリンク	10	
14	コネクタ	10	
15	RCサーボ PICO BB	5	

合計 1287g

写真のように2kgまで量れるキッチン量りで部品や材料の重さを量って表にまとめてみました。だいたい約1.3kgになりそうです。表は測定したものを、重いもの順に並べてみましたが、ご覧のようにRCサーボとプラ板の重量が大部分を占めています。



質量計測の模様

▶ 脚の寸法

脚を動かすサイクルが同じならば、脚が長いほど、一步で進む距離が大きくなり、速く歩けます。でも、RCサーボが出せる力には上限があるので、これから説明するように脚の長さにも制限があります。これからどうやって脚の長さを決めたかを説明します。

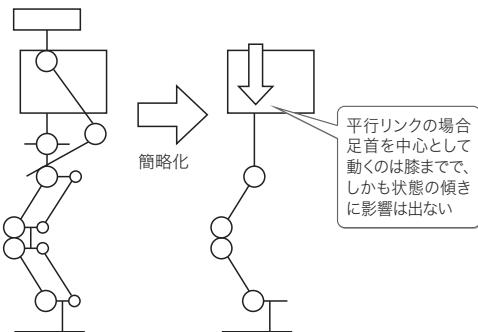
計算はただの掛け算です。それは小学校で習う範囲なのですが、そこにたどり着くまでの考え方の説明がちょっと長いので、ここで結論を要約しますとこういうことです。

ロボットの重さの半分と、脚の長さの半分を掛けた結果の数字が、ロボットのひざやすねに使うRCサーボのカタログに「トルク」と書いてある数字を超えてはいけない

この数字に近かったり、超えていたりするとロボットは重すぎて歩けないかもしれません。この先の説明のような話が「あまりお好きでない」読者の方も多そうな気がしますので、そういう場合は、これだけ覚えておいていただければ、後はとばしてもOKです（笑）。

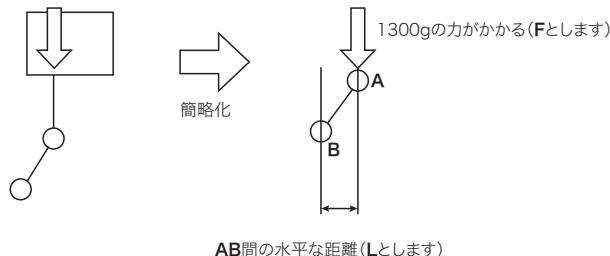
さて説明をはじめます。脚の話をするので腕や頭などの関係ない部分を除いた簡略化した図を使いましょう。また、実際には体のそれぞれの部分に重さがあるわけですが、それをそのまま計算しようとすると大変なので、胴体の部分にすべての質量が集中していることにします。簡略化こそしていますが、ロボットを完成して、実際にある場所にかかる力を測った場合よりも、**厳しい計算結果が出る方向に簡略化している**わけです。

考え方は、厳しへにラフな計算でRCサーボに必要なパワーを計算しておいて、使う予定のRCサーボのパワーがその計算結果を上回るようならば、実際にロボットが完成したときはパワーに余裕があるだろう……というもので、機械を設計するときに一般的な考え方です。



頭、腕、腰と平行リンクを省いて簡略化

1つの関節に着目して計算をするため、もう一段図を簡略化して、ひざから上を取り出します。すると、このような形になります。



AB間の水平な距離 (Lとします)

ひざから上だけに簡略化

このときにAの軸を回そうとする力は、距離 (L) と力 (F) の掛け算で出てくる数字（トルクといいます）で大小が比較できます。

なぜこの数字が欲しいかというと、モーターやエンジンなどの、機械の端に回転する軸が突き出でていて、ここに何かをつないで回すための機械（RCサーボも含めて）が出せる力はこの数字でスペックに出てきます。だから、それらを使う場所にかかるトルクを計算して比べてみれば、性能が足りているかどうかわかることがあります。

具体的にロボットのほうに話を戻しますと、ひざをいっぱいに曲げてしゃがんだときにAB間は水平になるので、Lが一番長くなります（この状態から立ち上がるときは、両足で体重を支えますので、かかる力を半分にして計算します）。このときにひざに発生するトルクを、条件を変えて2パターン計算してみます。Lが100mmの場合は次のようにになります。

$$1300[\text{g}] \div 2 \times 100[\text{mm}] = 65000[\text{g} \cdot \text{mm}]$$

RCサーボのトルクは、カタログに [kg · cm] という単位で示されていますので、ロボットの計算結果の単位もこれに合わせると以下になります。

$$1.3[\text{kg}] \div 2 \times 10[\text{cm}] = 6.5[\text{kg} \cdot \text{cm}]$$

ここで本書のロボットのひざに使うRCサーボ「SX-101Z」のスペックを調べてみると、トルクは3.3kg·cmとなっています。よって、2倍近く超えていることになります。

次にLを50mm (5cm)にしてみます。

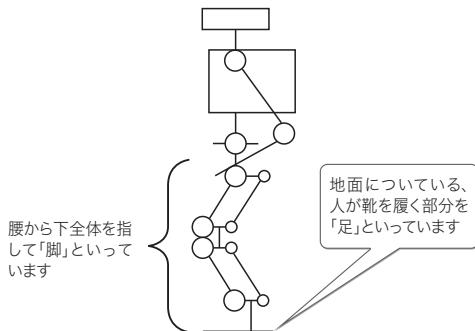
$$1.3[\text{kg}] \div 2 \times 5[\text{cm}] = 3.25[\text{kg} \cdot \text{cm}]$$

ぎりぎりです。ちなみに、RCサーボには、かかる力の大きさによって、マイコンからRCサーボに命令した目標の角度とのズレが大きくなる性質があります。以前計ってみたところでは、SX-101Zの場合、最大トルクに近いところでは10~20度くらいのズレが出ましたが、平行リンクを使ったことと、ひざから上と下のリンクを同じ長さにして、ひざの上と下でズレが相殺されることで、ロボット全体の傾きに現れてこないことを期待します。

このような過程で、起き上がりはできないかもしれないけれども、最悪でも歩くぶんには差し支えない範囲に入っているようなので、脚の長さはだいたい5cmに決めました。

Column 「脚」と「足」の使い分け

普通の日本語の範疇なので、特に問題にすることもないような気もしますが、いちおう図で明確にしておこうかと思います。



▶ 腕の長さ

このロボットの場合、腕は周囲のものに触ったり、ジェスチャーをしたり、起き上がるときに体を支えたり……という目的で付いています。はじめの2つだけなら、まあ見た目でそれらしいものが付いていればいいのですが、3つめの「起き上がり」については、脚同様、機能についての検討が製作前に必要です。

出せる力は、倒れているときは腕以外の場所も地面に着いているので、腕と、たとえばひざなど3点以上の場所に体重を分散して支えます。では具体的に体重をどれだけの比率で支えるかといえば、そのときの姿勢や、勢いの付き加減で大きく変わってきます。

「すごく変わるからわからない」といってしまっては、ロボットは形にならないので、ここは適当に、四つんばいになっているときは体重の1/4を1本の腕が支えると考えて、脚と同じような考え方で、脚の半分でいいわけだから、リンクの長さは倍の100mmくらいまでOKと考えます。

そのほかに考慮すべきなのは、関節が少ないせいで人間ほど体を柔軟に曲げることができない点です。それを補うため、しゃがんだとき、腕の先が地面に着く程度の長さに決めます。

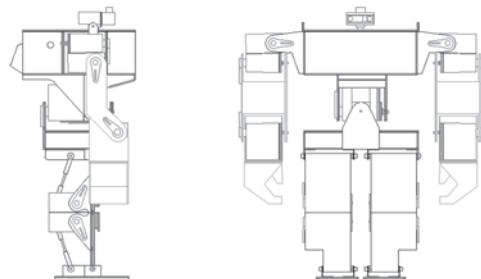
図面に描いてみると、肩から肘までと、肘から手先までの長さを90mmにすると上記の条件を満たすようなので、この長さに決めます。

▶ 胴体の形と寸法

胴体の中には、マイコンと電池を収めます。基板も電池も四角のシルエットをしています。だから、たとえば丸い胴体に入れようとすると、中に隙間が多くなり無駄です。また、曲線から構成される立体は作るのに手間がかかるので、マイコンと電池を収めて、マイコンの周囲の配線を収める余裕がある直方体に形と寸法を決めました。

▶ 全体の形

以上のような流れで主要な寸法を決めてきましたが、胴体以外の部分についても、定規とカッターナイフをあててプラ板を切り出すという工作方法をとるため、四角い板にするのが一番簡単です。そういう理由で、形については好き嫌いがわかれることとは思いますが、直方体をつなぎ合せたような形で全体を構成しました。

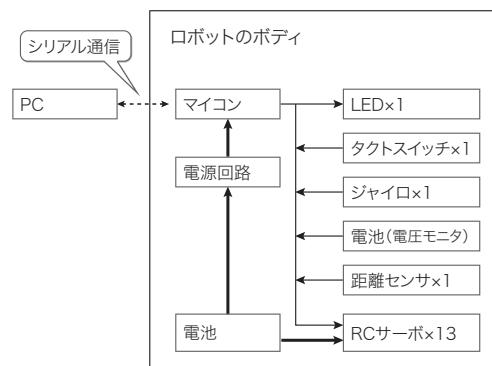


全体図

この図面には、「機能拡張編」で製作する飾りのカウルや、体の外に出てる電線などが描いてあります。が、だいたいこんな感じです。

3 電気系の設計

構想の部分で定義したロボットの概略仕様からマイコンを中心としてシステムの構成を書いてみます（電池部分以外の矢印は、信号の方向を表しています）。



システム構成図

上の図では、同じ種類のものは複数あっても1つの箱に書いてあります。実際にはたくさん電線をつなぐので、できあがったものを見ると複雑な印象を受けるかもしれません、本質的にはこの程度のものです（とっても簡単でしょ？）。

マイコンボードはそれだけでは動かないで、マイコンボードと周辺回路を載せるための基板（I/Oボードと呼びます）の製作が必要です。電気関係での工作は、それに加え、その基板とLEDなどの各部品との間の配線ということになります。

よく聞くおなじみの言葉だと思いますが、ここからの説明にたくさん使いますので、ごく簡単に、本書のロボットに関係付けて、言葉の意味を説明しておこうと思います。

デジタル（回路）

このロボットもデジタル回路でできたデジタルコンピュータで動かすので、とっても関係の深い言葉です。数字や状態の変化を整数でとびとびに扱う方式のことです。よくコンピュータは0と1で動いているといわれますが、実際の回路の状態としては「0」の状態が「0V」、「1」の状態が「5V」などと対応しています。

細かい話をしてると、信号（回路の途中に出てくる電圧）が0Vと5Vの間の値を通過する瞬間もありますし、実際のところ0Vと5Vの中間のある値の上下で0と1の解釈を分けているのですが、基本的には信号は0Vか5Vか（0か1か）の状態しか知らないと考えて回路を作っていく方がデジタル回路です（実際に回路を作ると問題となるケースも出てきますが、それは脇においときます）。

実際にマイコンでソフトを走らせたときは、たとえばマイコンの外に出ているピンと対応した変数を1にすると、そのピンは5Vになる、といった反応をします。

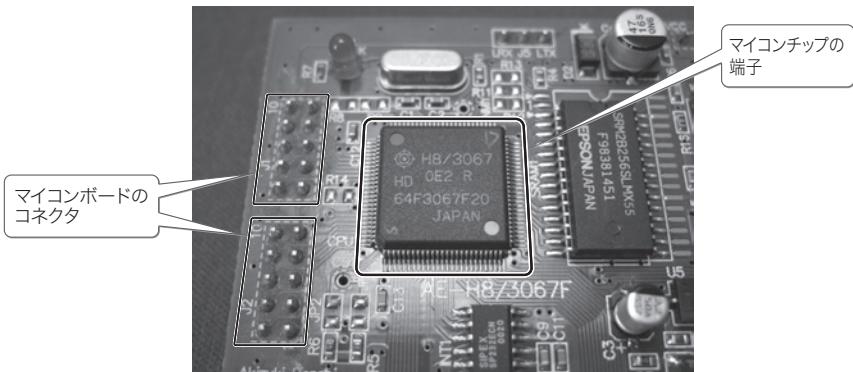
アナログ（回路）

数や状態の変化を、とびとびではなく、連続したものとして扱う方式です。回路を作る場合も、たとえば回路の内部の電圧が-12Vから+12Vまで連続的に変化し、任意の中間的な電圧にも意味があるものとして扱います。

ロボットとの関係でいえば、ロボットの関節の角度、距離センサの測定結果、電池の電圧など、すべてともとはアナログ量です。でもそれではデジタルコンピュータでは扱えないので、どこかにアナログ←→デジタル変換をする回路を挟んでいます。

Column

「マイコンに接続する」とは



本書で使用するH8/3067には写真に「マイコンチップの端子」と書いて示しているムカデの足のようなものが全周に100個生えていて、それぞれに決まった役割があります。

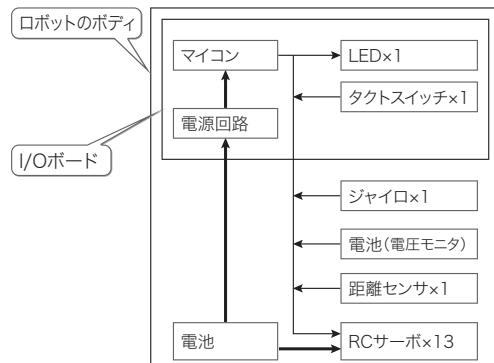
この中に「I/Oポート」と呼ばれる、デジタル信号を入出力するための端子があります。そして、「マイコンボードのコネクタ」と示している部分にマイコンボード上でつながっています。

たとえばRCサーボはこれらマイコンボードのコネクタの端子のうちの1つ、「J2」という名前が付いているものに接続します。実際の配線では、組み立てや調整時に便利なようにこのコネクタとRCサーボの間にはいくつかコネクタを経由しますが、最後にはここをとおってマイコンチップの端子につながる（すなわち「マイコンに接続する」）わけです。

デジタル信号を入出力：本書のロボットの具体例で説明すると、入力するとは、マイコンチップの外から来ている電線の今の状態が0Vか5Vかをチップの中に取り込んで、最終的にはソフトの変数に0か1かが入ってくることをいいます。反対に、出力するとは、ソフトの変数にセットした0か1が、マイコンチップの端子に0V、5Vの形で出てくることをいいます。

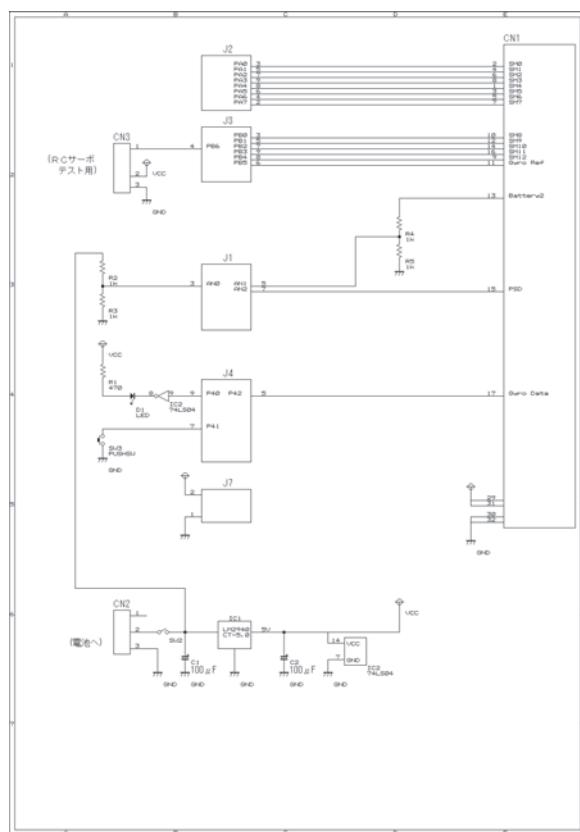
▶ I/Oボードの図面を描く

下の図は、37ページのシステム構成図からロボット内部に含まれる部分を抜き出したものです。この中でも「I/Oボード」と吹出しを付けて囲ってある部分が、本書で「I/Oボード」と呼称する1枚の基板になります。



I/O ボードの範囲

こんなふうに基本的な構成を決めて考えをまとめたら、実際に作れるようにより詳細に設計を進めます。そして、後で製作するときに詳しく説明しますが、システム構成図で1本の線で引いてある部分を、それぞれの部品への電線のつなぎ方（インターフェースといいます）を調べて、I/Oボードを図面に描くとこんな感じになります。



I/O ボードの回路図

こういった図面をはじめて見るかたは「なにこれ？」みたいな感じをお持ちになるかもしれません、この基板の場合は、根気よく、正確に線を半田付けしていくだけでOKなのでご心配なく。

簡単なものでも自分で回路図を描いて作る、マイコンを使った電子工作的経験をお持ちの方ならば、おそらくまったく問題なく完成できるのではないかと思いますが、「これがはじめて」という方も、半田付けだけうまくできれば大丈夫だと思います。

完成した形の回路図を見ると、意味が読み取りにくいといったこともあります、電子回路も、いわゆるメカが、ギアやネジのような部品からできているように、意味のある小さな部分が集まってできているので、それぞれの部分に注目していくと、意外と単純です。製作は、こういった「意味のある小さな単位」で行い、説明もその単位で進めていきます。

無線やオーディオ関係の電子工作などと異なり、マイコンを使った電子工作は、回路に関しては微妙な調整がなく、極端な話「つながっていれば大丈夫」といった意味で簡単なので筆者は好きです。

4 ソフトの設計

ここでは、前節と前々節で決めたハード（機械・電気）部分の構成を踏まえて、それを動かすためのソフトについて設計します。ソフトは、H8マイコン用ソフト、PC用ソフト（GUIなし）、PC用ソフト（GUIあり）の3種類から構成します。では、順番に概要を見ていきます。

▶ H8マイコン用ソフト

まず、ロボットはH8マイコンで制御するので、H8マイコン用のプログラムが必要になります。たとえばノートPCをそのまま積んでいるロボットでは、必要なことをすべてそのPCにやらせる場合もありますが、本書のロボットが搭載するのは、PCと比べて非常に計算能力やメモリの容量が限られているマイコンです。ですので、ロボットに（いい換えると、マイコンに）載っていなければならない部分のみ、マイコン用のソフトの機能として持たせ、後は外部のPCで処理する形態を採用します。

マイコンになければならない機能は、RCサーボを動かす信号を出したり、センサの信号を読んだりといった、ロボットのハードに直接関わる部分です。「1秒間に何回信号を出さなければならぬ」とか、「ある信号を出した後、1/1000秒以内に信号を読み取らなければならぬ」などプログラムが動くのに時間制限があるところが特徴です（こういったシステムを一般に「リアルタイムシステム」とか「リアルタイムプログラム」と呼び、Windowsのように好きなだけ時間を使うプログラムと区別しています）。

本書で使うH8マイコン用には、キットに使えるプログラミング言語としてアセンブラー^{*}、C言語^{*}、BASIC^{*}などが選択可能です。アセンブラーでの開発は骨が折れる作業で効率がよくないため避けます。そうなると、

アセンブラー：マシン語／機械語などと呼ばれることがある言語です。CPUの機種ごとにそのハードをじかに制御する記号でプログラムを作ります。CPUの機能をすべて個別に使えるという利点があります。

C言語：アセンブラよりは人に優しいプログラミング言語です。Linux等のOSのような、ハードウェアに近いソフトウェアを作るのによく使われます。なんにでも使えます。

BASIC：簡単な英語のようなプログラミング言語です。その昔、PCで初心者用のプログラミング言語として一般的でした。

C言語かBASICのどちらかになりますが、C言語のほうがマイコンと相性がいいので、C言語で開発します。

▶ PC用ソフト（GUIなし、コンソールアプリケーション）

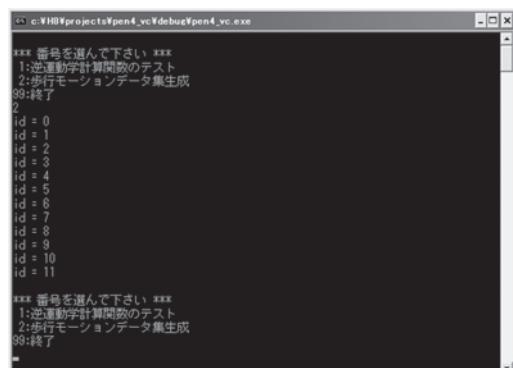
マイコンの処理能力では対処できない計算をするプログラムもあります。本書のロボットは、粘土アニメーションのように、ある一瞬のポーズにおける各関節の角度を数値化し、このポーズを1秒間に50回のペースで変更することによって動きます。このとき、各瞬間のポーズの集合が全体として歩くような動作することによって歩きます。

このような仕組みなので、ポーズ集のデータ（以降モーションデータと呼びます）を作るプログラムが必要です。モーションデータを作るのには、粘土アニメのようにその瞬間のポーズを実際にロボットにとさせて、それを記憶していく手法（教示/ティーチングといいます）と、歩行のように周期的な運動を、一気に計算で生成してしまう方法があります。

後者の方法の計算は、PC用のCPU（ペンティアムなど）ならば、1秒間に50回のペースで計算可能ですが、H8にはそれほどの計算能力がないため、PCであらかじめ計算し、歩行中は、マイコンボードのメモリに蓄えてある計算結果を使うことになります。このように、ロボットから離れたコンピュータで計算を行う形態を「オフライン計算」などといいます。本書では、以降これにちなんで、あらかじめ外で計算した歩行等のモーションのデータを「オフラインモーション」データと呼ぶことにします。

このPC用のプログラムはVBで開発してもいいのですが、将来もっと処理能力の高いマイコンに積み替えたときは、ロボットのマイコンのほうに持たせたいので移植性を考えて、マイコンと同じC言語で開発することにします（プログラムのフォーマットはC言語で、使うコンパイラはVCです）。

プログラムを実行したときの見た目は、いわゆる「Dos窓」で動くプログラムです。このような形態のプログラムを「コンソールアプリケーション」といいます。



```
***番号を選んで下さい ***
1:逆運動学計算関数のテスト
2:歩行モーションデータ集生成
99:終了
2
id = 0
id = 1
id = 2
id = 3
id = 4
id = 5
id = 6
id = 7
id = 8
id = 9
id = 10
id = 11
***番号を選んで下さい ***
1:逆運動学計算関数のテスト
2:歩行モーションデータ集生成
99:終了
```

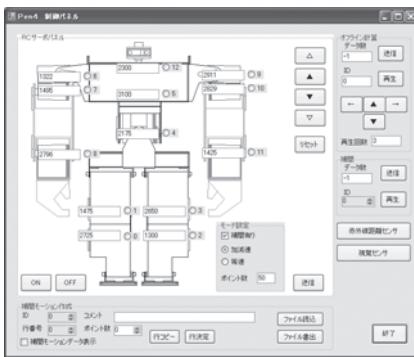
PC用のコンソールアプリケーション

▶ PC用ソフト (GUIあり)

そのほか、ロボットを動かしていると、センサの値などを見たくなるときがあります。でもロボットには、液晶ディスプレイなどが付いていないので見られません。そこで、情報をPCに送り、PCの画面で見るためのGUIプログラムも作ります。

実際にポーズをとらせながらモーションデータを作るプログラムもGUIが必要なので、こちらのプログラムに含めます。このようにして作ったデータは、ロボットを動かすときには、ポーズとポーズの間を自動的に補間しながら再生するので、以降「補間モーション」データと呼びます。

また、こちらのプログラムはロボットとデータをやり取りする必要があるので、通信機能も備えます。開発には、WindowsのGUIを簡単に扱えるのでVBを使います。



「H8用ソフト」の中に「自律行動制御」というのがありますが、これは、たとえば何かセンサに反応があったら歩く向きを変えるとか、腕を振るとかのロボットが自分で動くための機能です。なんだか大仰な名前が付いていますが、マイコン側に持たせるとささやかなものになります。PC側に持たせると、ロボットとPCをケーブルや無線で接続する必要がありますが、PCの処理能力やLAN、そしてインターネットなどを使えるので、かなりの可能性を秘めたものになります。本書では、まあ「ちょっと紹介」という感じなので、ロボットがケーブルなしで独立して動けるように、マイコンのソフトに簡単な機能を付与します。

さて、今日は、これから製作するロボットの概要について電気、機械、ソフトの面から紹介してきました。本のオビに写真があるので、だいたいどんなものかはご理解いただけていたものとは思いますが、ここまで読んで、実際に作るものとしてのイメージを持っていただけたでしょうか？

今までに「プログラミング」や「電子工作」など、まったく未経験の分野の場合は「なんだかよくわからなかった」という感想をお持ちの方もいらっしゃるかもしれませんね。

でも、本書のように特に何かを作るような具体的な話は、「やれば」わかりますのであんまり気にしないで先に進みましょう！明日はいよいよ製作開始です。

3 日目

構想 設計 調達 電子工作
プログラミング 工作 テスト

I/O ボードの組み立て (その 1)

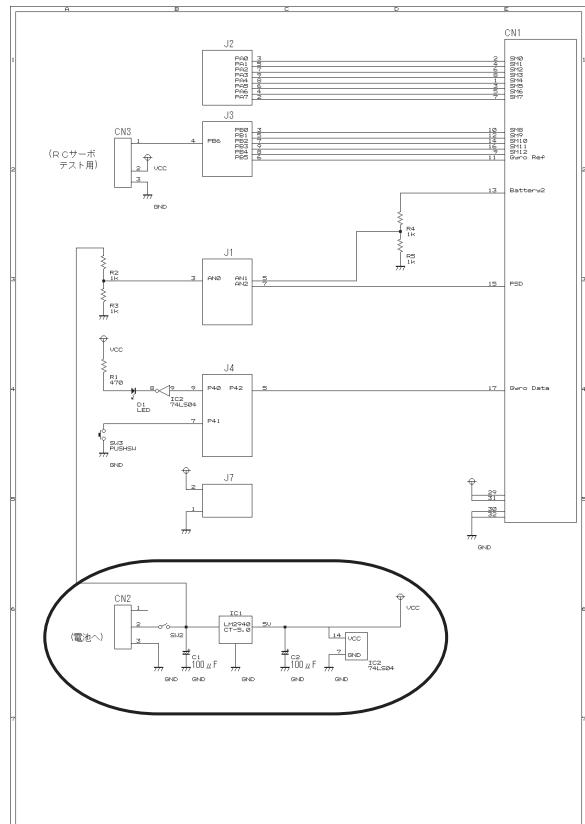
- ① いよいよ実作業!
- ② 電源回路とはどんなもの?
- ③ 基板の準備
- ④ 3端子レギュレータとコンデンサの取り付け
- ⑤ 3端子レギュレータ付近の配線
- ⑥ 電源スイッチの配線

1 いよいよ実作業!

さて、今日からいよいよ実作業に入ります。みなさん、材料や道具はそろえましたか? まだの方は巻末のリストを見て急いで買ってきましょう。

ではマイコンボードのI/Oボードを組み立てます。まず電源回路からはじめます。電源回路ができあがっていると、その後で作る部分を実際に動かしてテストしながら進められるからです。それに電源回路はI/Oボードの上にまとまっていてわかりやすいので手はじめとしてはうってつけです。①回路図の読み方の基礎、②基板の配線方法、③回路のチェックとテスト方法について、はじめですので少し丁寧に説明します。

次の図に示すように、I/Oボード全体の中では、マルで囲った部分が電源回路になります。

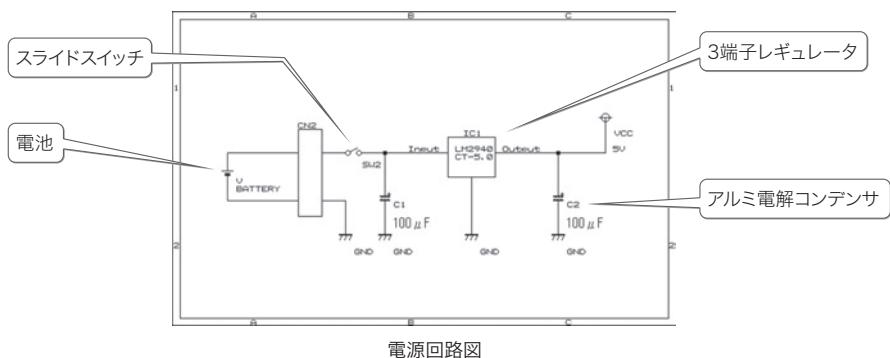


I/O ボード全体回路図

2 電源回路とはどんなもの？

本書で使うマイコンボードは、 $5V \pm 5\%$ の、電圧が安定した電源で動きます（デジタル回路は0と1に相当する電圧を具体的に決めて動いているので、たいていの場合電圧の安定した電源を必要とします）。一方、電池は残量によって電圧がだんだん下がる性質があります。そこで、電池から安定的に5Vを作り出す回路が必要になります。この回路を「電源回路」と呼んでいます。

わかりやすいように電源回路を部分的に抜き出した回路図を描いてみました。



電源回路図

回路図に登場している記号を実際の部品と対比させつつ、回路図の読み方から説明していこうと思います。部品の説明の途中で「GND」（グランド）という言葉が出てきますが、組み立ての途中で詳しく説明しますので、御存じない方もとりあえず読み進めてください。

3 基板の準備

さて、ここからは実際に手を動かしていただきます。I/Oボードを作る基板の準備として、基板にスペーサーを付けます。

▶ 部品の紹介

ユニバーサル基板

マイコンキットに入っていた基板を使います。これは両面スルホール、ガラスエポキシというタイプの基板ですが、本書の場合、部品を基板の片面にしか付けませんし、電気的な要求も特に厳しいものはないので、この大きさ（72×95mm）のユニバーサル基板ならば、なんでもOKです（片面、紙エポキシというタイプが安いです）。

両面スルホール：金属の輪が基板の両面にあり、穴の中も金属でつながっているもののことです（Through Holeです）。



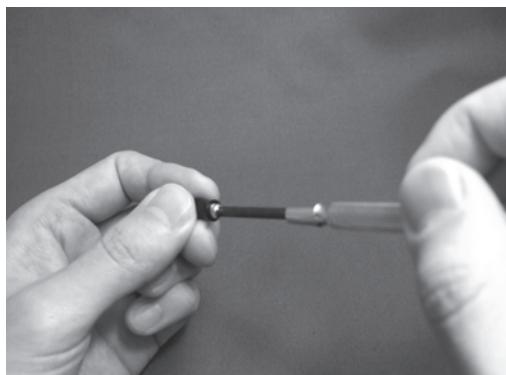
スペーサー

基板の足です。ロボットのボディに基板を取り付けるために使いますが、製作中は基板を置いたときに配線がある基板裏面が何かに触らないようにという目的で付けています。



▶ 組み立て

基板にスペーサーを付けます。スペーサーは両端にネジが付いていますので、片方だけドライバーではすします。

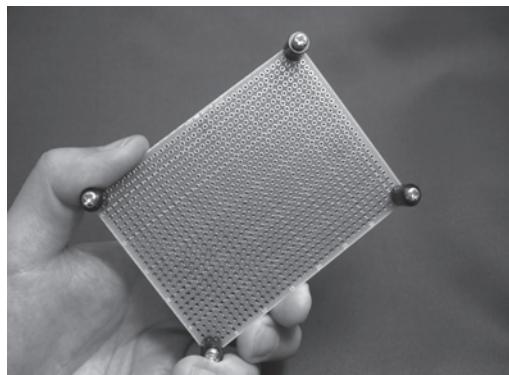


ネジをはずす



はずし終わったところ

スペーサーからネジをはずし終わりました。簡単ですね。では、今度はこれを基板に付けます。



基板に付ける

基板の4隅にスペーサーのネジとちょうど合う穴があるので、こんな具合に付けます。ちなみに基板を持つときは、このように半田付けする面を触らないように持ちます。半田付けする面に汗や油が付くと、半田が付きにくくなってしまうためです。

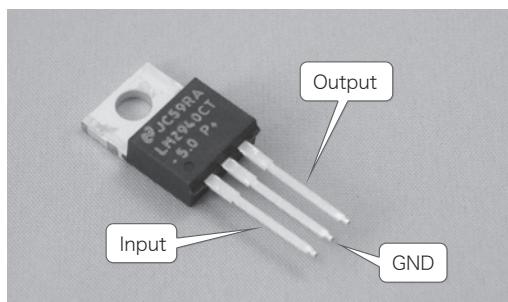
④ 3端子レギュレータとコンデンサの取り付け

次に、電源回路を構成する部品を基板に半田付けします。まずざっと部品を説明しておきます。

▶ 部品の説明

3端子レギュレータ

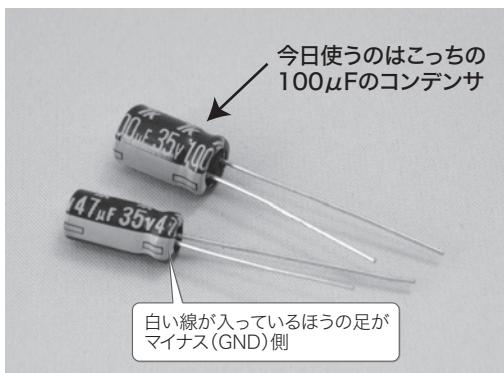
回路図の中にIC1と描いてある部品です。型番は「LM2940CT-5.0」というものです。電池からの電圧（残量や負荷によって一定しない）やACアダプタからの電圧（負荷によって一定しない）を決まった電圧に安定化します。回路図中の「Input」「GND」「Output」と、実際の部品の足との対応は、それぞれ次のようになっています。



アルミ電解コンデンサ

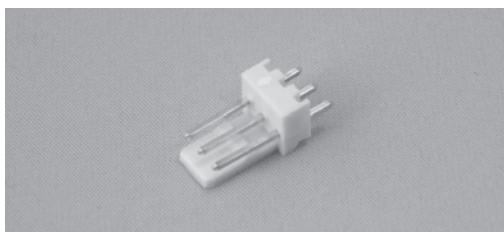
電気を一時的に貯めるための部品です。回路図の中ではC1、C2と描かれています。3端子レギュ

レータからの電圧を安定させるために、先の機種の3端子レギュレータに限らず必ず付けます。ここで使うのは、容量が $100\mu F$ （マイクロ ファラドと読みます）のアルミ電解コンデンサです。



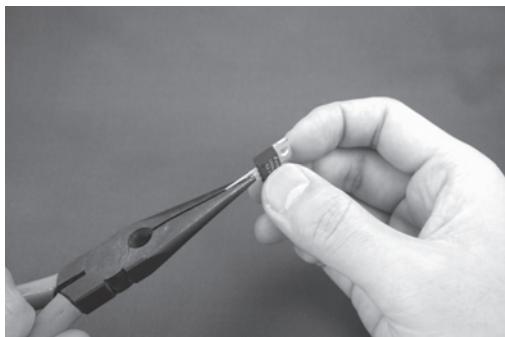
コネクタ (オス)

回路図中ではCN2と描いてあるコネクタです。基本的には基板に半田付けして使うものです。Ni-MH電池パック（ラジコンのプロポセット用）に付いているコネクタ（メス）を挿せるものを選びました。

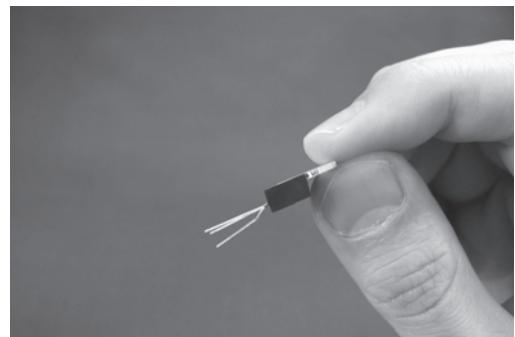


▶ 組み立て

3端子レギュレータの真ん中の足（GND用）を、写真のようにラジオペンチを使って基板の穴1列分だけ曲げるように曲げます。

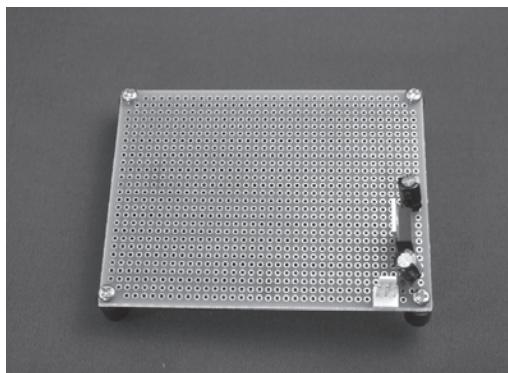


ラジオペンチを使う

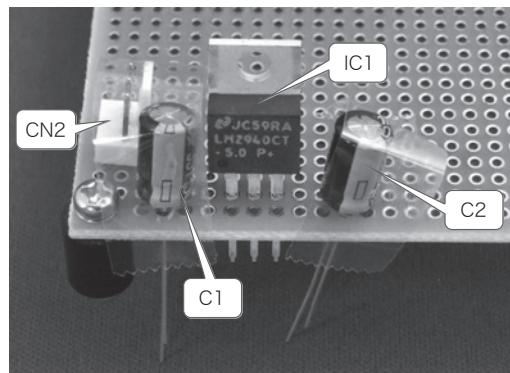


できあがり

基板の上に3端子オスコネクタ (CN2)^{*}、アルミ電解コンデンサ (C1)、3端子レギュレータ (IC1)、アルミ電解コンデンサ (C2) を写真のように並べ、裏返して半田付けするときに落ちないようにセロハンテープで仮止めします。写真をよく見て取り付け向きに十分注意して下さい。

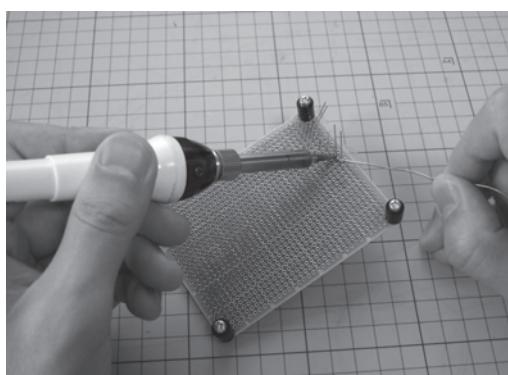


基板の上に並べます

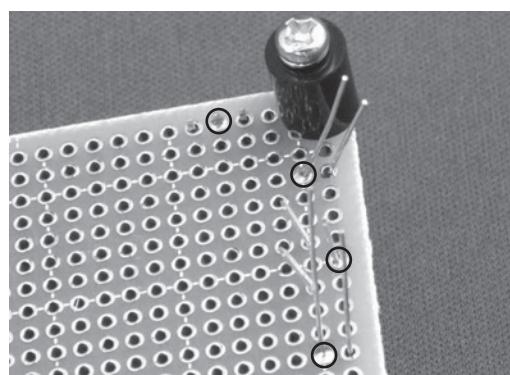


C1、IC1、C2 はマイナスの足が手前の同じラインに乗るようにします

部品の足を、それぞれ1力所だけ半田付けします。セロハンテープで仮止めした状態で半田付けすると、きれいに部品を付けられないので、このようにはじめに1力所だけ半田付けし、後で位置を調整します。この仮付けの段階では、部品が斜めにささっているのがわかります。当然ですが、半田を盛りすぎてほかの穴にくっつかないように注意してください。失敗した場合の復元方法は65ページのコラムを参考にしてみてください。それからこれも念のため、失敗した状態でいきなり電源を入れるとショートしてしまうので、ここでぐっとこらえて、手順どおりに進めてくださいね。

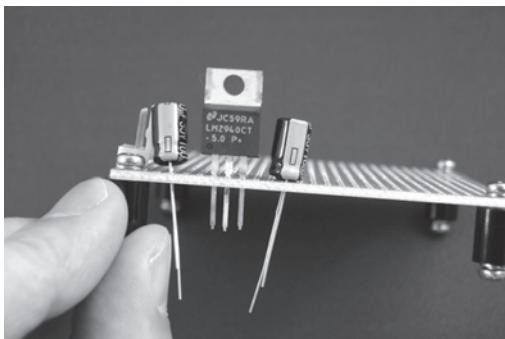


半田付け



仮付け完了（裏面）

*以下、本書では部品名は"3端子コネクタ (CN2)"のように、"一般名(回路図の記号)"で表記します。型番の場合は「」で囲って表記します。

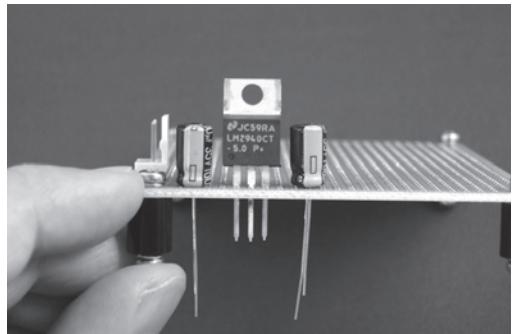


横から見ると少し曲がっていますね

基板の表側から部品を軽く指で押した状態で先ほど付けた半田を溶かすと、基板にピッタリの位置に部品位置を調節できます。まっすぐになったら、ほかの足もすべて半田付けしましょう。これできれいに並びました。この部品は、別に曲がって付いていてもかまわないのですが……まあ、気持ちの問題です。

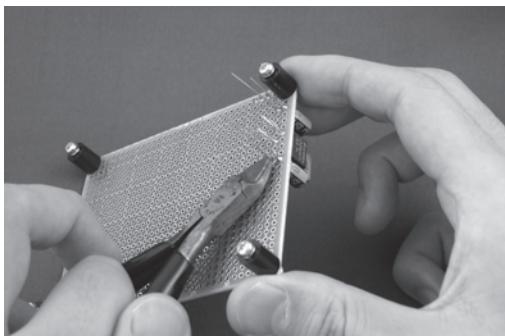


部品を押さえながら半田を溶かすと……



まっすぐ付けられる

邪魔になるので、余分な足をニッパーで切り取ります。切り取った足は配線材料（ジャンパ線と呼びます）として使いますので捨てないでとっておきます。



余分な足を切る

部品は、基板の上でその回路が占有する面積がなるべく少なくなるように、そして、配線がなるべく短くなるように考えて基板の上に配置しますが、ここでは、回路図との対応が理解しやすいように、回路図の形

に似せて並べました。そんなわけで、電子工作の腕に覚えがある方は、この回路については、結果的に回路が同じになれば、配置は同じにしなくともかまいません。

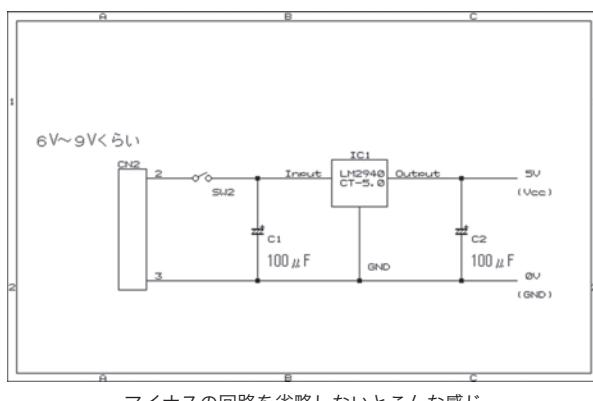
5 3端子レギュレータ付近の配線

さて、部品を付け終わったら、次は基板上で部品同士の配線を行います。

▶ 回路図の見方（実際に配線するためには）

回路図を見慣れていない方にまず説明が必要と思われる「Vcc」と「GND」についての説明からはじめます。まあ、あたりまえといってしまえばあたりまえなのですが、電子部品は電気で動くので、電源を接続しなければなりません。というわけで、ほとんどの部品に必ず電源の接続を書き込むことになるので、実際に電源回路までの配線を描くことは省略して、その電子部品のどこに電源の接続があるのかだけを描く習慣になっています。

下の回路図は電源回路のマイナス側を省略せずに描いてみました。46ページの回路図と比べてみてください。結局のところ、このロボットの回路図では、「GND」とあつたら電池のマイナスへ、「Vcc」とあつたら3端子レギュレータの出力端子へ接続すればいいだけの話となります。



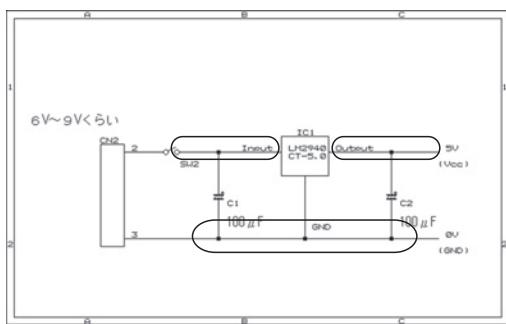
マイナスの回路を省略しないとこんな感じ

ここだけの回路図ですと「GND」記号を使っても、使わなくてもほとんど変わりませんが、部品がたくさんある回路図では大違いになってくるので、ここはこういうものとして理解してください。

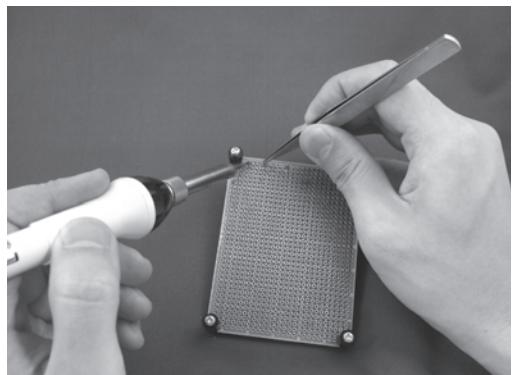
さて、これで実際の部品のつながりがわかりましたので、次は回路を組み立ててみます。

▶ 電源回路の組み立て

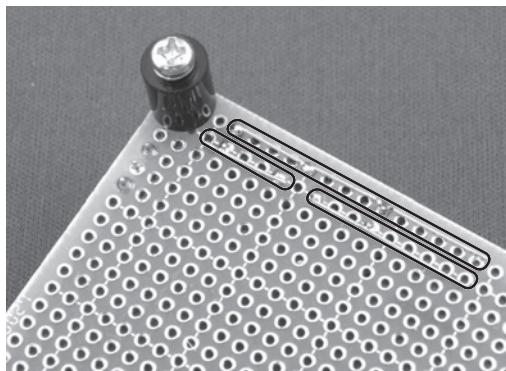
回路図中で黒い点で示した部分の配線を行います。配線にはさっきコンデンサの足を切り取ったもの(ジャンパ線)を使います。必要なジャンパ線は4本、半田付けする点は、7力所です。



ここをジャンパ線でつなげる

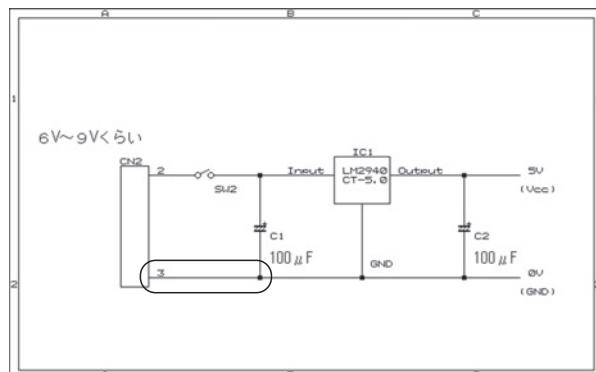


ピンセットでジャンパ線を持って慎重に半田付けする



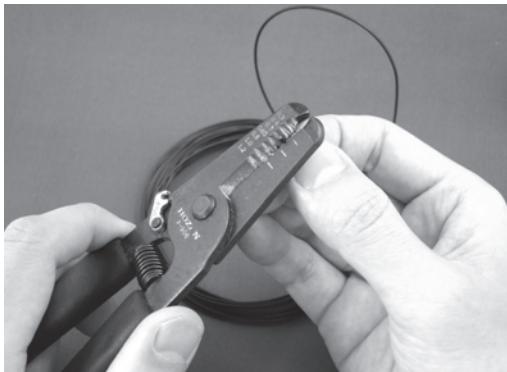
こんなふうに付ける

次は、GNDとコネクタ（CN2）の3番ピン間の配線を行います。ここもジャンパ線で行ってもかまいませんが、基板を裏側から見たときにGNDがわかりやすいように、黒い被覆の電線（より線）を使って配線します。

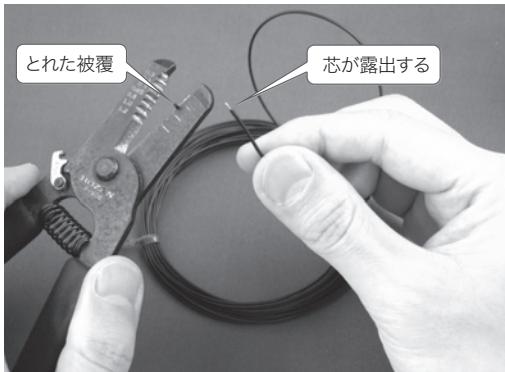


ここを黒い電線でつなげる

適当なサイズの場所でワイヤーストリッパーを使い被覆を剥きます。電線の中身の径よりも小さなところで挟むと、中の銅線がパラパラっと切れて落ちてきますので、そういうときは1つ上のサイズの場所を使います。大きすぎる場所を使ってしまったときは、引っ張っても被覆が取れないのでわかります。そのときは1サイズ下げます。

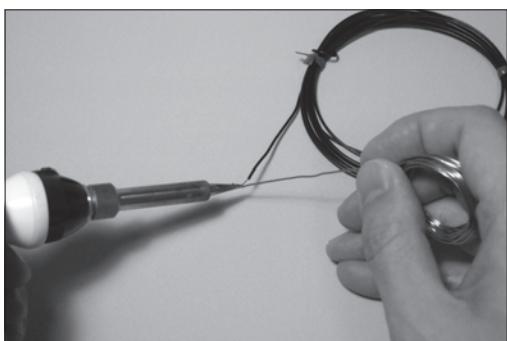


ワイヤーストリッパーで電線を挟む

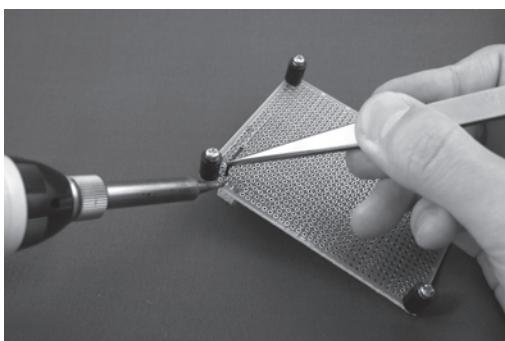


そのままスッとひっぱると被覆だけをとれる

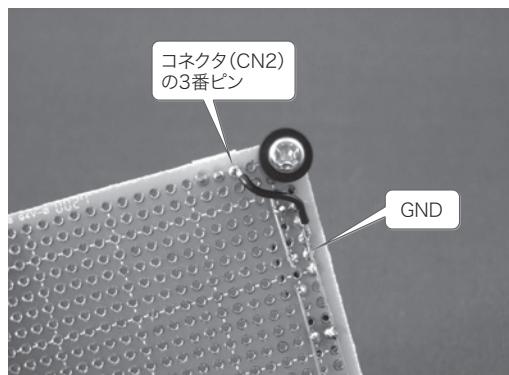
電線を半田付けする場合は、下のようにあらかじめ電線に半田を乗せておくと、付けやすくなります。



電線に半田を乗せる（これを半田めっきといいます）



電線に乗せた半田を溶かして半田付けする



配線完了

GNDとコネクタ（CN2）の3番ピン間の配線が完了しました。ちなみに、コネクタ（CN2）のどのピンをGNDにして、どのピンを3端子レギュレータのInputピンに接続するかは、一般的な決まりがあるわけではないので、電池のコネクタに合わせて筆者が決めてあります。

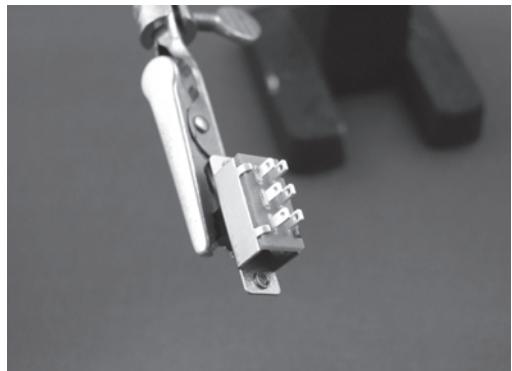
6 電源スイッチの配線

うまく半田付けはできましたか？ここまでで半田付けに関する主なテクニックはすべて実践していただきました。後は繰り返していくうちに慣れてきます。それでは続けて、電源スイッチの配線に入りましょう。

▶ 部品の説明

スライドスイッチ

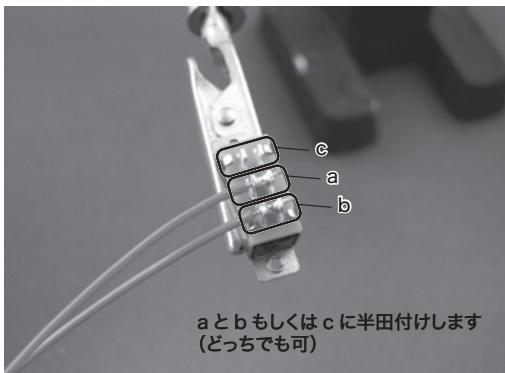
横にスライドさせてON/OFFするスイッチです。回路図中ではSW2として描かれています。このスイッチは接点が2組付いていますが、1組のタイプでもかまいません。



裏側には端子が付いている

▶ スイッチの配線

では配線です。基板をロボットに内蔵した後でもロボットのボディ表面にスイッチを埋め込んで使えるように、このスイッチは基板から引き出した形としてあります。ですので、10cm程の長さに切った電線を、端子に半田付けします。ここでは、電源のプラス側なので赤いより線を使いました。写真ではヘルピングハンドという、台座の先にクリップが付いた道具を使っています。部品を手で持って作業してもいいのですが、両手が自由になるのでこれを使ったほうが楽に正確に半田付けできます。



スイッチに電線を半田付けする



こんな具合になります

アルミ電解コンデンサ（C1）のプラス側の足付近と、コネクタ（CN2）の2番ピンへ、スイッチ（SW2）から伸びている電線を半田付けします。隣り合った穴ならジャンパ線なしで半田で直接つなげます。

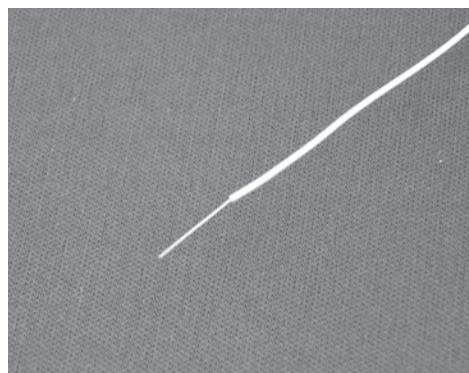
Column

電線の種類

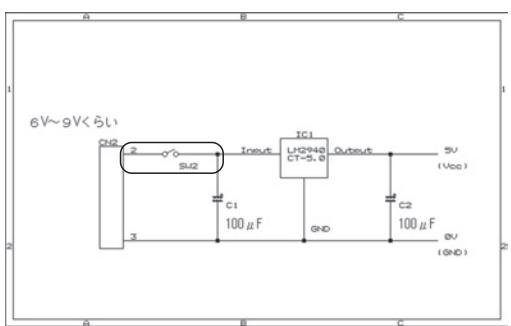
ここで使うのは「より線」という種類の電線で、中には細かい銅線が複数本入っています。柔らかいのが特徴です。本書では、基板から伸びる配線と、電源部分に使っています。そのほかに「単芯線」という電線も使います。こちらは字のとおり中に1本だけ銅線が入っています。形付けやすいので、基板の上にルートを固定して配線するときに使っています。一般的にプラス側に赤色、マイナス（GND）側に黒色のものを使用します。



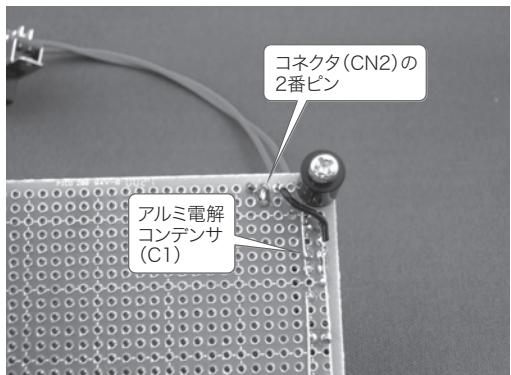
より線



単芯線



ここがスイッチと赤い電線



電源回路 配線完了！

今日の製作はここまでです。正しくできていれば、この部分は電池をつなぐだけで働きますが、回路が誤配線でショートしていたりすると、電源投入と同時に煙が出てきたり……ということもありますので、このまま動かしたい気持ちはやまやまですが、いったん時間を空けて、注意力が十分な状態で回路のチェックをしてから電源投入を行いましょう。

そういうわけで、明日テストができるように、電池を充電しておいてください。

注意

例の充電器は、電池にあわせた充電電流に調整されてはいますが、ただのACアダプターです。「充電完了時に自動的に切れる」機能は付いていません。14時間程度の時間をかけて充電する場合は時間を超過しても特に問題ありませんが、忘れないようにしましょう。

Column

Ni-MH電池の充電

充電してお休みになる前にもう1つ。簡単に電池の充電について紹介します。

1. 通常充電



電池（6V 1100mAhのNi-MH電池）を充電しているところです。一晩ほどをかけて行う「通常充電」用の充電器は、ご覧のようにACアダプタです。この電池は、ハードの完成後はRCサーボを動かすのにも使いますが、製作中はマイコンを動かすのに使います。

充電器も電池もラジコンショップでRCサーボ用として合わせて貰えるものです。このようにペアで貰って説明書のとおりに使えば、何も考えずに使えるので楽で安心です。

使うのにはそれで十分なのですが、せっかくですから充電の仕組みについて少し説明します。本書で使用したNi-MH電池の容量は1100mAhです。いい換えると110mAの電流を1時間流せるだけの電気エネルギーを蓄えているということになります。

Ni-MHの充電は、一般的にだいたい、容量の1/10の電流で、容量の1.5倍ぶんの時間、電流を流し込むと満タンになります。たとえば1100mAhの場合は、110mAで15時間充電すると満タンということです。

ただし、何ボルトかけたときに所定の電流（ここでは110mA）になるかは、何ボルトの電池かによって変わってきます。だいたい電池電圧の1.2倍くらいの電圧で、電池に流

Ni-MH電池は1本あたりの電圧は公称1.2Vです。写真の電池は、5本の電池を直列つなぎにしてパックしたものなので $1.2 \times 5 = 6V$ となります。ただし、満タンに近くなると公称電圧よりも高い電圧が出る性質があるため、1.2倍くらいの電圧をかけないと満タンになりません。

す電流の倍くらいまでの電流が流せるACアダプタを探して、後は抵抗で調節するという方法も可能ですし、充電時間で調節してもかまいません。

これは筆者が別の自作ロボットで使っている充電器です。いらなくなった携帯電話用のACアダプターを流用しました。こうしたことがわかると充電器の自作だってできちゃうわけです。



通常充電と急速充電

電池の充電は、電流を水の流れにたとえると、容器に水を貯めるのに似たところがあります。少しづつ入れると貯まるのに時間がかかりますが、勢いよく注げば早く貯まります。

Ni-MH電池も充電の電流を増やせば早く充電が終わります。ただ、電流が多いだけに、充電しすぎると電池は壊れてしましますし、電池がなくなった後で、また充電して使える回数も、通常充電で使う場合よりも減る傾向になります。それじゃあ、ちょっともったいないですよね。

製作中、特にマイコンのプログラム開発時には電池でマイコンを動かしますが、本書の電池とマイコンの組み合わせで1回の充電で半日以上は電池がもちますので、本書執筆時においては通常充電を採用しています。

4 日目

構想 設計 調達 電子工作
プログラミング 工作 テスト

I/O ボードのチェック

- ① 目視確認
- ② ショートのチェック
- ③ 性能確認

1 目視確認

今日は、昨日組み立てた電源回路のチェックをします。ここでは、簡単なLED点灯回路を例に取って、回路図の読み方に対する理解を深めつつ、テスト/チェックのやり方を紹介します。

まず、回路図のとおりに、部品が付いて配線がされているか、今一度目で見て確認します。チェックのポイントは以下のとおりです。

①コンデンサのプラス、マイナス（極性といいます）を正しく付けているか

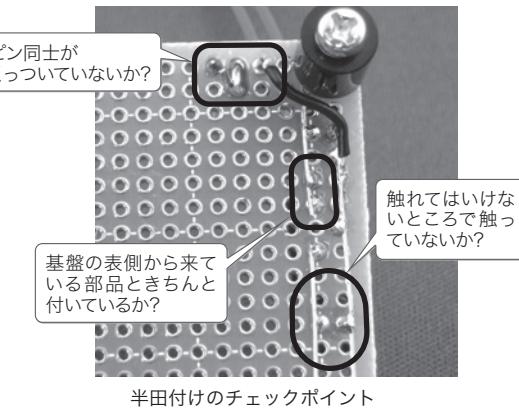
49ページの写真を見るとアルミ電解コンデンサにはマイナス（GND）側に白い線が入っているのがわかります。この線を頼りに確認しましょう。

②3端子レギュレータのInputとOutputを間違えていないか

これも48ページの写真とよく見比べて確認してください。

③半田付けは正しく付いているか

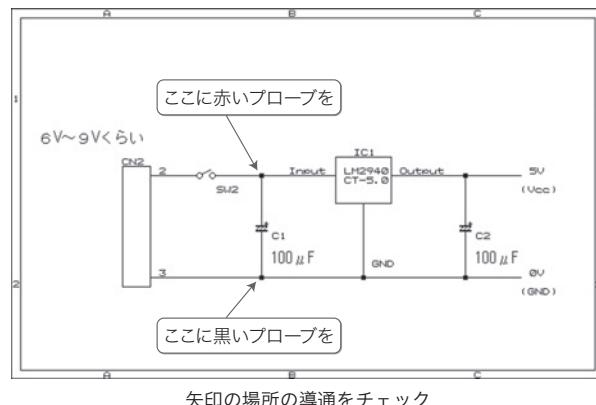
回路図とよく見比べながら、そのとおりにつながっているか確認します。筆者の場合は時々ジャンパ線のつなぎ目や、ジャンパ線と3端子レギュレータの足などが付いていないときがあります。このあたりに注意してみるといいでしょう。



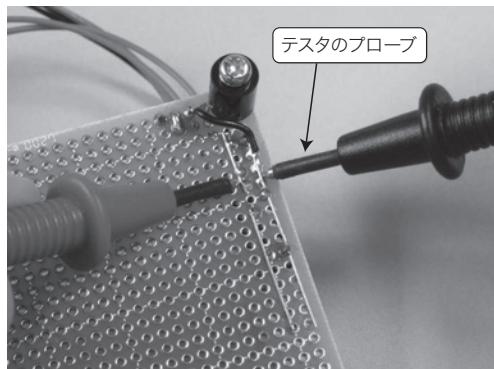
配線を行うときは、済んだ場所の回路を蛍光ペンで塗っていくなど、どこまで済んだのかがわかるようになると、配線忘れが防げます。そして、目視チェックのときにもう一度印をつけます。基板に対して電源ラインのVccに沿って赤いライン、GNDに沿って黒いラインを油性ペンで書いたり、部品の記号を基板に書いたりなどすると、配線のミスが減りますし、思うように動作しなかったときの原因追及にとても便利です。「わかりやすくする!」ことが確実に動作する回路を作るためにとても大事なので、ぜひ実行しましょう。

2 ショートのチェック

電源回路のプラスとマイナスが直接つながってしまっている（つまりショートしている）場所がないかテスターで確認します。まずは、下図の矢印の場所でショートしていないかどうかをチェックしましょう。



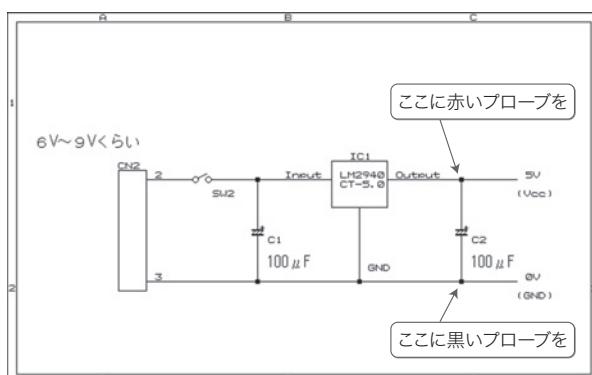
写真のこの場所が回路図で矢印の場所に対応しています。



アルミ電解コンデンサ（C1）の足がショートしていないか

「導通チェックモード」^{*}があるテスターの場合は導通チェックモードにセットします。ここはショートしていい場所なので、ブザーが鳴らないのが正しい状態です。導通チェックモードがないテスターの場合は「抵抗測定モード」で測定します。レンジいっぱいの大きい数字が出れば（アナログテスターの場合はレンジいっぱいの数字を針が指せば）正しい状態です。

今度は、下図の矢印の場所でショートしていいかどうか、テスターでチェックします。



矢印の場所の導通をチェック

基板上での場所は下の写真的とおりです。

導通チェックモード: つながっている場所をテスターのプローブで触るとブザーが鳴って知らせるモードです。こういったチェックをするときは、プローブをあてる手元を見ながら作業するので、テスターの表示を見るために視線を離さなくていいところが便利です。

プローブ: テスターなど測定器から出ているケーブルの先端の回路を触る部分のことです。和訳すると「探し針」という意味です。