

元気に生まれた OMNIHEAD



まえだ たけし
前田 武志

(ソフトウェア編)

2足歩行ロボットなんてそんなに簡単に作れるの?と誰しもが思っていたのではなかろうか。ROBO-ONE が開催されるまでは、私もそう思いながら第1回大会～第3回大会を傍観し、これは案外頑張れば自分でもできるんじゃないか?と来て来た。前号のハードウェア編に引続き、本記事ではOMNIHEADのソフトウェアについて解説する。ソフトウェアは、ロボット内部のマイコンで実行するファームウェアと、外部のPCで実行するモーションエディタ、そしてモーションのデータそのものから構成される。以下、それぞれを解説する。

モーションとモーションエディタ

ロボットの動かし方

動くロボットを作るには当然、ロボットの「動き」、すなわち各関節の角度をどう決定するかを考えなければならない¹。

その実装としては、

- a) あらかじめ決められた動きを再生する方法
 - b) リアルタイムに算出する方法
 - c) マスターズレイブにより操縦する方法
- 等が考えられる。

a) は最も簡単な方法で、多くのロボットがこの方法で動いているようだ。

b) は逆運動学やセンサ値を用い、何らかの理論に基づいて関節の角度を求める方法で、これも広く使われている。

c) はこれらの方法と違い、ロボットを動かす時に全ての関節角度を人間が決めるという方法で、普通はロボットを模した特殊な入力デバイスを用いる。人間がその場で対応するという意味で、最も融通が効く方式である。

OMNIHEADにおいては「手堅く作る」方針に基づき、最も簡単な a) を採用することに

した。ただし、第3回ROBO-ONE大会において剛王丸II²や、Adamant 3rd³がマスターズレイブ方式を非常に有効に用いていたのもあり、一部この考え方も取り入れることにした。

モーションとは?

以上、あらかじめ決められた動き(以下、モーションと呼ぶ)を再生する方法をベースがそのモーション自体をどう作るのかを説明する。

各関節の角度を決めると、ロボットはある姿勢(ポーズ)になる。すなわち、各関節の角度を自由に調整できるツールを作れば、それはポーズエディタとなる。

また、複数のポーズを連続的に再生することで、動き(モーション)となる。ポーズの順番や移行時間を編集する機能をシーケンスエディタと呼び、これらの機能を併せ持つツールをモーションエディタと呼ぶことにする。

これらの情報を数値で入力したり、専用のプログラミング言語で制御することも考えられるが、OMNIHEADでは自在に簡単に編集できることが重要と考え、GUIで直接的に操作できるエディタを作成した。以下に紹介す

るのがそのモーションエディタ"TopDancer"である。

システム構成

図1にモーションエディタのシステム構成を示す。

ロボットとPCは有線シリアル(RS-232C)で接続され、PC上で実行されているモーションエディタの操作内容が、リアルタイムでロボットに反映される仕組みになっている。ユーザはモーションエディタを操作し、ロボットの動きをその場で確認しながら、ポーズ編集やシーケンス編集を行う。

モーションエディタが送信する各関節の目標角度を受け取り、実際にサーボモータを駆動するのが、ロボット側のCPUボードの主な仕事となる。

また、ポーズ間の補間もロボット側で行い、補間された現在の値を返すことで、PC側で補間の状況を知ることできるようにした⁴。

こうして作成したモーションデータは、最終的にはロボット側に書き込まれる。今回は通信の確実性を重視し、無線はラジコン式を選択したため、本番ではPC無しで動かす必要があるためである⁵。

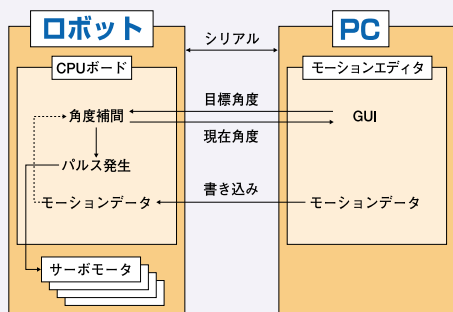


図1 モーションエディタのシステム構成

¹ 究極的には、何も教えずにでも自発的に動きを学習するロボットも考えられるが...
² ROBO-ONE 常連でもある津藤智さんのロボット。ゲーム機のパッドを改造した装着型のコントローラで操縦する。光ファイバーコミュニケーション回路はまだ採用していないようだ。
³ 同じく常連の滝沢一博さんの大型ロボット。「巨体はポリシーなのでやめません」の名ゼリフを残した。
⁴ 本当ならサーボモータ内のポテンシオメータから「真の現在の角度」を読み出したい所であるが、サーボモータを改造しないとイケないので今回は見送った。「俺サーボ」の利点はこれが簡単にできることであろう。
⁵ 無線シリアルや無線LANを使う方式には、こういったモーションエディタやプログラム開発環境がそのまま本番でも使えるというメリットがある。ただしPCは立ち上げるのに手間や時間がかかったりPC自体がトラブルの原因になり得る。

操縦モジュール

操縦モジュールの仕事は、受信機やセンサの情報から、次にどのモーションを再生するかを決めることである。

モーションエディタの説明でも触れた通り、基本的にプロポの2本のスティックで各種モーションの起動を行い、どの操作が何のモーションになるかという割り当てマップをその他のスイッチで切替えるという仕組みになっている。

スティックが倒れていない時はアイドルングのモーションを再生する。

アイドルング中に加速度センサが転倒を示す値になった場合、転倒した向きに応じた起き上がりモーションを再生するようになっていく。

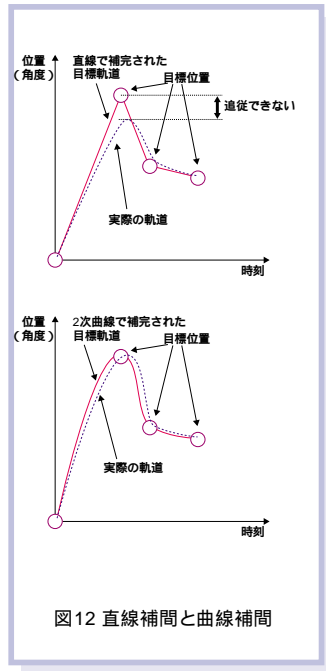


図12 直線補間と曲線補間

モーション管理モジュール

モーション管理モジュールは、上位の操縦モジュールやオートデモ管理モジュールの指示に従い、与えられたモーションを再生するモジュールである。

オートデモ管理モジュールからは、再生するモーションと繰り返し回数を指示される。

操縦モジュールからは、再生するモーションだけが指示され、後にモーション終了指示が与えられた時にモーション終了シーケンスに入る。これにより歩行や倒立といった動作中に終了指示が来ても、無理なくホームポジションに戻る仕組みになっている。

再生するモーションはポーズ単位に分解され、ポーズ補間モジュールに送られる。

ポーズ補間モジュール

OMNIHEADではポーズとポーズの間を滑らかに補間することでモーションを構成している。この補間を行うのがポーズ補間モジュールである。

補間アルゴリズムはいろいろな物が考えられる。最も単純なのは直線補間であるが、これは「行って戻る」モーションに対して追従性が悪いという問題がある(図12上参照)これは、頂点で速度が不連続に変化し、結果として無限大の加速度を要求しているのが原因である。

この対策として、補間は2次曲線で行い、かつ「行って戻る」指令値に対しては頂点でいったん速度がゼロになる加減速制御を行うことにした(図12下)

この補間処理は全サーボについて60Hzの周期で行っており、メインCPUにとっては最も重い処理となる。OMNIHEADのメインCPUは16MHzのH8/3664Fであるが、実測してみると各種割り込みも含めて平均的には18%程度、ピーク時で25%程度のCPUパワーを消費していた。

パルス発生モジュール

このモジュールは、上位のポーズ補間モジュールで決定した関節角度を入力とし、それに基づいた幅のパルス信号を発生してサーボを駆動するモジュールである。

ラジコン用サーボに入力するパルス信号は、周期はラフで良いが幅はシビアな精度が要求される。

OMNIHEADのメインCPUとなるH8/3664Fに搭載されているタイマーWには、コンペアマッチによりハードウェア的に出力ピンを操作できるアウトプットコンペア機能が4ch分実装されている。

これがロボットの軸数(24ch程度)だけあれば何も問題ないのだが、4chしかないので、一工夫必要になる。

ラジコン用サーボを制御するのに必要なパルス幅は最大2.5msec程度なのに対し、周期は50~60Hzすなわち20~16.7msec程度と長い。これを利用し、図13のように時分割することで、6~8倍の個数のサーボを動かすことができる。

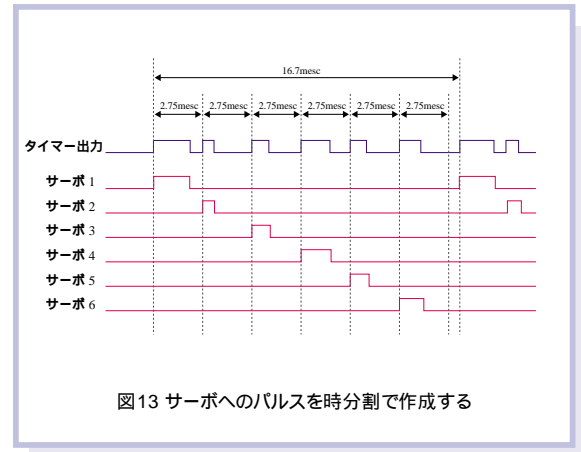


図13 サーボへのパルスを時分割で作成する

時分割するためには、一つのタイマー出力を順次切替えて別の出力ピンに分配しなければならぬ。この分配作業を当初、メインCPU内で割り込みによりソフトウェア的に実装していたが、他の割り込みによるジッタが気になったので、最終的にはPICマイコンを追加してハードウェア的に行うことにした。

謝辞

以上、ハードウェア編・ソフトウェア編と2回に渡りOMNIHEADを解説させて頂いた。

ロボットに関しては今後もハードウェア・ソフトウェアとも進化し続けるだろう。みなさんの作った、あっと驚くロボットに出会える日を楽しみにしている。

無事にOMNIHEADが完成し、しかも準優勝という結果まで出せたのは、さまざまな方々の御協力のおかげである。

そもそもラジコンサーボで2足歩行ロボットが作れると証明してくれた諸先輩の方々、子供^{*21}が{生まれる|生まれた}にも関わらずロボットに没頭することを許してくれた妻、ロボット製作を応援して頂き、工具や機械を勝手に使い、業務に混ぜて基板を作ったりしても怒らなかった会社の方々、楽しい遊び場所を作って頂いたROBO-ONE委員会の方々、会場やネット上で応援して頂いたの方々、この記事を書く機会を設けて頂いたロボコンマガジン編集部の方々に感謝の意を表明する。

*21 執筆時点ではまだアクチュエータの力不足と制御ロジックの未完成から歩行はできておらず、ROBO-ONE参加資格を満たせていない。