

みかんでも動くロボットの開発

3年A組 樋口幸太郎

3年B組 西田惇

3年C組 前澤俊哉

4年A組 中嶋研人

4年A組 岡田慎太郎

指導教諭 末谷健志

1 要約

私たちは、中学生と高校生の共同研究として、モーションキャプチャシステムをコントローラとして採用した遠隔操作可能なロボットを開発した。通信にLANネットワークを利用しているため、原理的には地球の裏側からでも操作が可能であり、ロボットのカメラがとらえた映像を、離れたPCでも閲覧できることなどが、特徴として挙げられる。また、任意の物体をコントローラとすることができるため、このロボットはみかんでも操作が可能である。

コントローラに用いた自作したモーションキャプチャシステムは、USB接続のカメラを利用して、人間や物体の動きを記録できる。このシステムの特徴は、一台のカメラで物体の3次元座標を取得することが可能な点である。これを利用すると、3次元空間を操作できる新しいマウスを実現することができるなど、様々な応用が可能である。

このモーションキャプチャシステムに加えて、無線LANやPC、PIC、サーボモータなどを組み合わせ、自前で“0から”ロボットを作り上げた。

キーワード ロボット、LAN、モーションキャプチャ、PIC、Webカメラ

2 研究の背景

キットを組み立てて動かすロボットが多数市販されるようになったが、その内部の仕組みを知ることは難しい。そこで私たちは、有線リモコンで操作するロボットや二足歩行ロボットを0から製作してきたが、遠隔操作をするにはコードを延長するしかなかった。そこで、無線LANやシリアル通信を使うことによって離れた位置からロボットを操作できるのではないかと思うに至った。

3 研究の目的

無線LANやシリアル通信を使うことによって離れた位置から直感的に操作できるというロボットを実現するためには、いくつかの課題を解決する必要がある。そのうちの3つをここで示す。(図1)

研究 I

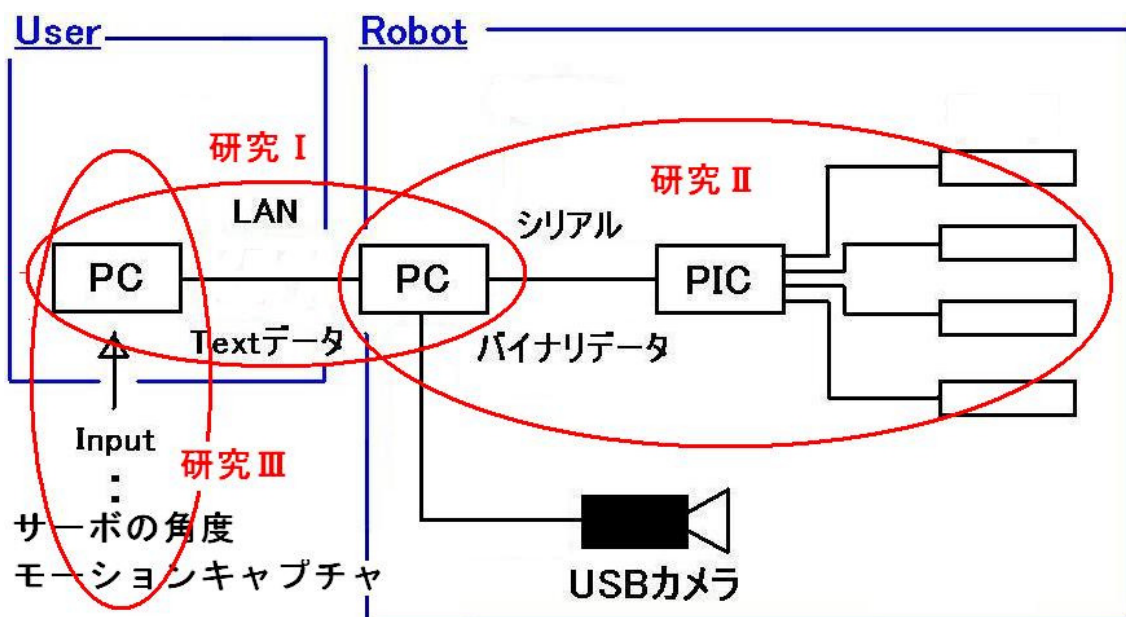
ユーザーから入力されたサーボモータの角度を、LAN ネットワークを経由して、ロボットと通信する。

研究 II

研究 I で送られてきた命令を読み取って、ロボットのサーボモータを制御するシステムを構築する。

研究 III

ロボットを操作するためのモーションキャプチャシステムの開発を行う。



<図 1 : 研究>

4 研究内容

(1) 概要と仮説

<研究 I 通信方法の研究>

① 概要

ロボットのサーボモータを動かすための Text ファイルの通信を Windows の共有フォルダ機能を用いて行った。これは、ネットワークプログラムより簡単な方法である。

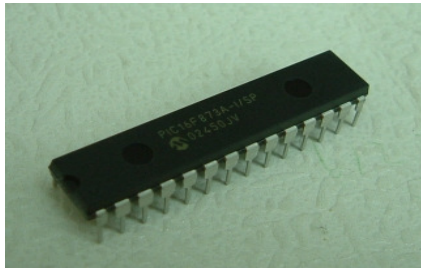
② 仮説

Windows の共有フォルダ機能を用いて、ロボットのサーボモータを動かすための Text ファイルを通信できる。

<研究 II その 1 二足歩行ロボットの開発>

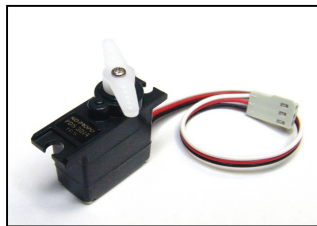
① 概要

サーボモータを制御する信号を生成するために、PIC を用いた。PIC とは、Microchip Technology 社が製造開発しているマイクロコンピュータである。PC からプログラムが可能で、非常に安価であるため、現在日本で普及しているマイコンの一つである。今回は 16F873A という PIC を用いた。(図 2)

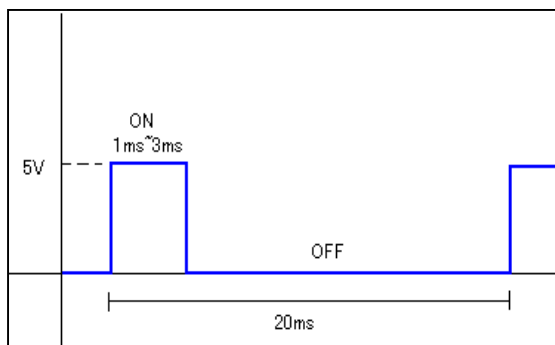


<図2：PIC マイコン>

また、サーボモータ(図3)とは、DCモータと違い、電圧と制御信号を与えることによって正確に回転位置を決めることができるモータである。信号は、PWMと呼ばれる信号を用いる。これは図4のようなパルス信号になっており、ONの時間を変化させることで、モータの回転位置を定められる。



<図3：サーボモータ>



<図4：パルス信号>

サーボモータを制御する信号を生成するユニットをサーボモータ用コントローラと呼ぶ。製作したコントローラはサーボ用の電源とPIC用の電源と、分けてある。なぜなら、電源を一つにまとめた場合、モータは大電流を必要とするためPICへの電源供

給が不安定になりプログラムがストップしてしまうと考えたからである。供給を安定して行うため9V電池からの電圧を下げ、PICが必要とする5Vを供給している。また、外部ハードウェア、ソフトウェアと通信するためにシリアル通信のレベル変換ICを搭載している。これにより、PCとの通信や他の機器との通信が可能になった。

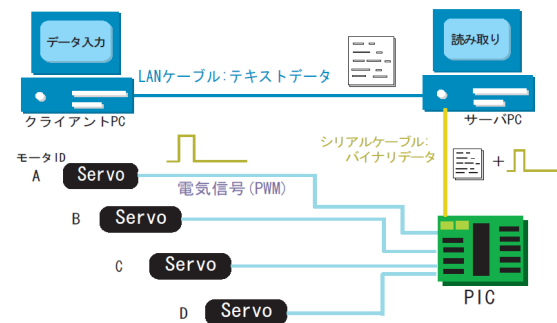
② 仮説

PICマイコンを用いて、サーボモータを制御し、二足歩行型ロボットを製作できる。

<研究Ⅱ その2 PCとPICの通信>

① 概要

PICでサーボモータを制御する技術を活用し、PCとPICをリンクさせ、クライアントPCからLAN経由で送られてきたテキストデータを、サーバPCを経由してPICに転送し、サーボモータの制御を行った。(図5)



<図5：通信>

② 仮説

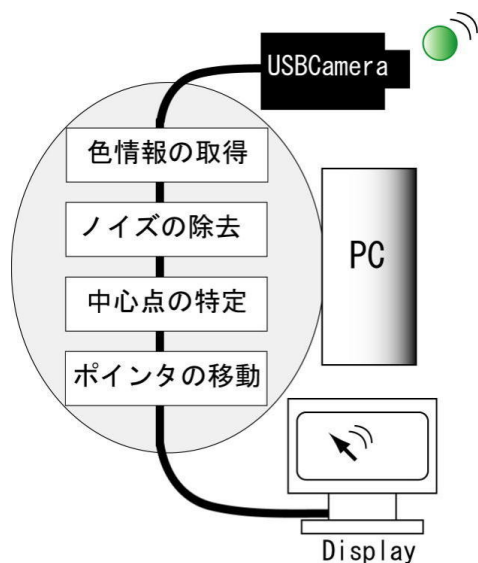
ネットワークを利用してサーボモータの制御信号を送受信し、ロボットアームを制御できる。

<研究Ⅲ モーションキャプチャの開発>

① 概要

私たちが開発したモーションキャプチャシステムは、図6で示すような処理を行うことで、物体の三次元座標を取得することができる。カメラに映し出された物体の色情報をフィルタにかけ、私たちが”重心法”や”弦法”などと称する方法を用いて物体の中心を計算する。このとき、物体が球に近似できる場合、球の半径を割り出すことができる。得られた半径は、カメラからの距離に逆比例すると考えられるので、奥行きを求めることができるのである。

ロボットのサーボモータを動かすためには、軸を回転させる角度を教えなければならない。その角度をモーションキャプチャで得た物体の座標情報で与えることによって、物体の動きとロボットの動きをリンクさせることができるのではないかと考えた。



<図6：モーションキャプチャ>

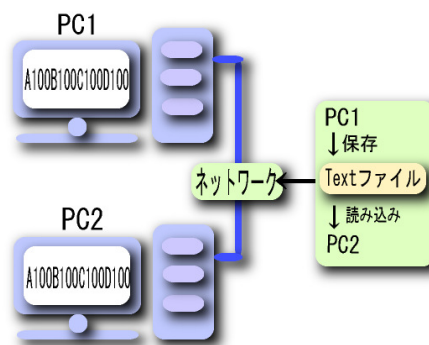
② 仮説

モーションキャプチャシステムを用いて人間や物体の動きでロボットの動きをコントロールすることができる。

(2) 研究方法

<研究Ⅰ 通信方法の研究>

実際にテキストファイルを通信するプログラムを **Borland C++Builder5** という開発環境を用いて作成した。プログラムの手順としては、以下の通りである。(図7)

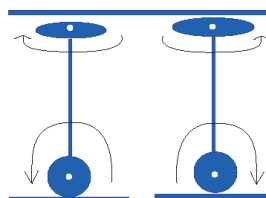


<図7：通信>

- i ネットワーク上のテキストファイルに PC 1 で入力されたデータを次々に保存していく。
- ii ネットワーク上に保存されたテキストファイルを PC 2 で連続的に読み込む。

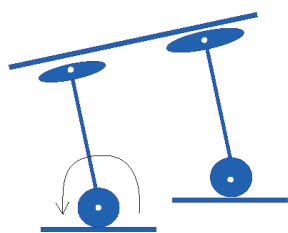
<研究Ⅱその1 二足歩行ロボットの開発>

二足歩行ロボットを実際に製作し、歩行させた。歩行方法は、次に示す手順である。ロボットの自由度が低いため、人間の歩きとはかなり違った歩き方をする。

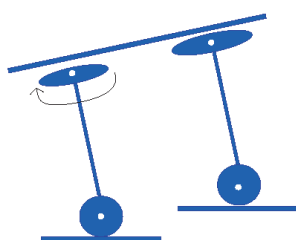


1つの足に、2つのサーボモータを配置した。

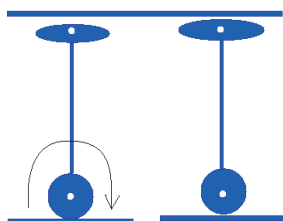
i 右足首を傾ける



ii 右足の太ももを回す



iii 右足の足首を元の位置に戻す



※右足も同様に行う。

※全体として、がに股歩行となる。

<研究Ⅱその2 PC と PIC の通信>

クライアント PC から、サーバ PC のファイルにサーボモータの ID(A~I)と数値をテキストに書き込み、保存する(これには、フォルダ共有機能を利用)。保存したテキストファイルをサーバ PC が、VB2005 で制作したソフトで読み込み、ID ごとのテキスト

の数値から、数字の数値に変換する。次に、PIC に対して、①ID②数値の順にシリアルケーブルを経由して送信する。PIC は ID ごとに送られてきた数値を PWM に変換し、サーボモータを制御する。データ例を下記に示す。(図 8)

A127B220C010D100
└──────────┬──────────┘ 回転角
└──────────┘ サervo ID

<図 8 : データ例>

<研究Ⅲ モーションキャプチャの開発>

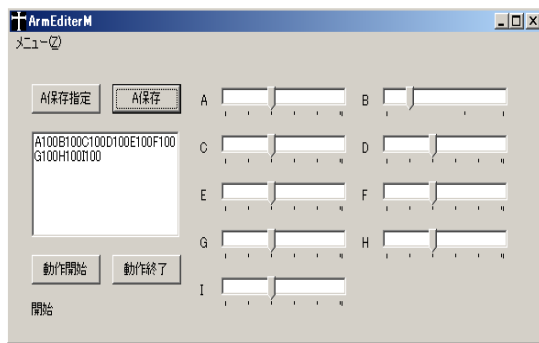
モーションキャプチャシステムによって取得した物体の三次元座標を、x 座標をロボットカメラの水平軸を制御するサーボモータ、y 座標をロボットカメラの鉛直軸を制御するサーボモータ、z 座標(奥行き)をロボットの前進または後進を制御する DC モータにそれぞれ対応させた。

(3) 研究の結果

<研究Ⅰ 通信方法の研究>

図 9 のようなアプリケーションを開発することができた。

しかし、実験中にエラーが頻発した。これは、一つのファイルに対して保存と読み込みを同時に行うため、命令が衝突するからである。そこで、図 10 のようなエラー処理を行った。エラーが発生した場合、保存・読み込みをする時間を与え、もう一度プログラムの最初にもどって保存・読み込みをやり直す、というようにした。



<図 9 : アプリケーション>

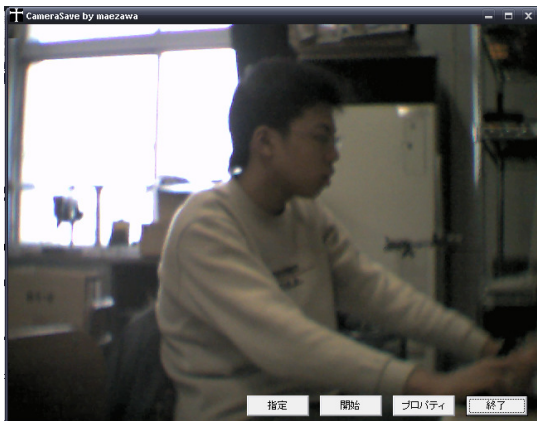
```

label1:
// エラー処理
try {
// エラー処理の対象
}
catch(...) {
Sleep(10); // 1/100 秒待つ
goto label1; // label1 へ
}

```

<図 1 0 : エラー処理>

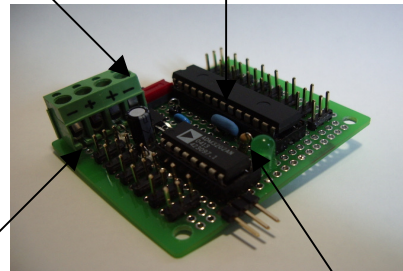
また、研究 I を応用させて USB カメラの画像の通信も行うことができた。カメラの映像の保存は 0.1 秒に一回の割合で行い、その結果、読み込み側の PC でリアルタイムの映像を見ることができるようになった。(図 1 1)



<図 1 1 : アプリケーション>

<研究 II その 1 二足歩行ロボットの開発>
PIC でサーボモータを制御し、二足歩行ロボットを制御することができた (図 1 2)。

PIC用電源入力端子 PIC16F873A

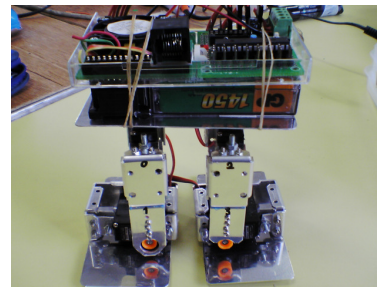


サーボ用電源入力

発振子

<図 1 2 : 制作したコントロール部>

二足歩行ロボット製作を成功させて気づいたことであるが、歩行を実現するためには、サーボモータに送る信号の微調整が頻繁に必要であることが分かった。制作したロボット (図 1 3) の仕様を以下に示す (図 1 4)。



<図 1 3 : 二足歩行ロボット>

CPU : PIC16F873A 10MHz
プログラム : 自作 (アセンブリ言語)
動作電源 : サーボ 6V , PIC9V
外部 I/F : シリアル通信 (RS232C) × 1
自由度構成 : 右足 2、左足 2
製作費用 : 10000 円

<図 1 4 : 仕様>

しかし、制御信号を微調整するためには、プログラム上で変更し、PIC をコントローラから抜き差ししなければならないことがわかった。

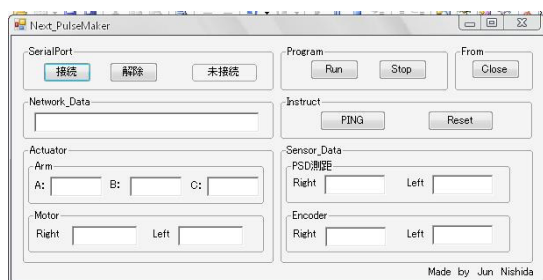
そこで、PC と PIC をシリアルケーブルで接続し、PC で数値を変えながら、サーボモータの位置を決めると効率が上がるのではないかと考えた。

そこで、PIC-PC 通信を成功させ、PC で数値を変更、それと同時にサーボモータが動くという環境を整え、歩行モーション制作の効率を向上させた。

<研究Ⅱ その2 PC と PIC の通信>

ネットワークを利用してサーボモータの制御信号を送受信し、ロボットアームを制御できた。

図 15 のソフトは VisualBasic2005 で制作した。100 ミリ秒ごとに共有フォルダにあるリッチテキストファイルを読み込み、各 ID の数値を割り出し、PIC にサーボモータごとの ID と数値をシリアルデータとして送信する。



<図 15 : ソフト>

しかし、PIC に対して連続して同じ数値を送信した時、サーボモータが激しく振動することが分かった。

サーボモータが激しく振動した原因を考

えてみると、PC の CPU はクロック数が 1GHz~2GHz と非常に高く、プログラム 1 命令を実行する時間が極めて短い。しかし、PIC のクロック数は 10MHz であり、PC のクロック数と比べると非常に低速である。このため PIC が、PC から毎回送られてくるデータの間隔の速さについていけず、異常な PWM(電気信号)をサーボモータに対して送信していると考えられる。

そこで、PC が PIC に毎回送信するデータの間隔をさらにあけた。このことでサーボモータの激しい振動は解消された。

<研究Ⅲ モーションキャプチャの開発>

モーションキャプチャシステムのカメラの前でみかんなどの物体を動かすことによって、ロボットの動きをコントロールすることができた。

5 考察

制作したロボットは、モーションキャプチャを利用することによって、従来のキーボードやマウスでロボットを制御するよりも、ずっと直感的で分かりやすい操作性がある。

しかもロボットとの通信は、共有フォルダ機能を利用したシンプルなファイル交換方式であるため、メンテナンスや機能の拡張も非常に楽であることが分かった。

つまり、“0 から”制作したロボットは、操作性と機能の拡張にすぐれているといえる。

6 まとめと今後の課題

現在は、今までの研究成果をもとにして、図16のようなロボットを実現することができた。このロボットは、物体や人間の動きでコントロールでき、無線で遠隔操作ができる。また、ロボットに搭載されたカメラの映像を離れたPCにリアルタイムで送ることができるなどの特徴を備える。

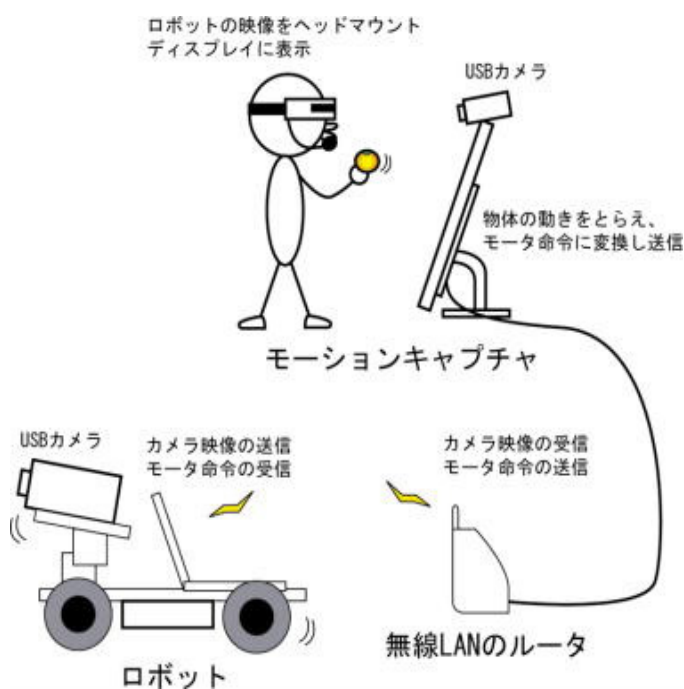


<図16>

今後は、図17に示すように、ロボットのカメラの映像をヘッドマウントディスプレイに映し出し、ロボットのカメラの動きを、ヘッドマウントディスプレイを被った人の頭の動きに合わせて合わせるなどして、より操作性の良いロボットを開発したいと思っている。

また、モーションキャプチャシステムをロボットに搭載したPCに搭載することにより、完全に自立運動することのできるロボットを構築したいとも考えている。

これらのシステムは、すでに開発に取りかかっているが、手持ちのノートPCが重すぎてロボットに搭載できないなどの問題が生じている。駆動系に強力なモータを用いたり、小型で軽量なノートPCなどを利用したりして、近い将来、実現させたいと考えている。



<図17>

7 謝辞

末谷先生には、理論面や技術面で大きなバックアップをいただきました。児玉先生には、技術指導や電気回路に関して様々な相談にのってくださいました。さらに、予算面で、植野校長先生や吉田信也先生に多大なご協力をいただきました。

お世話になった方々、大変ありがとうございました。