

## 1 4. LEGO を用いた知的ロボットの製作と プログラム作成学習教材の開発

担当学生

黒田 純（工学部 電気電子システム工学科 4年）

指導教員

滑川 徹（工学部 旧・自動制御研究室）

## 1 はじめに

LEGO Mindstorms Education NXT はプログラム可能なブロック NXT (Mindstorms ロボットの頭脳に相当)を含むキットである。NXT は、センサによる計測、モータなどの出力装置を制御するためのコントローラなどを装備した高性能マイクロコンピュータを搭載している。NXT には、図 1 のように、4 つの入力ポート(1~4:センサ用)、3 つの出力ポート(A~C:モータ用)、各種ボタン、ディスプレイ、USB ポート(PC との接続用)がある。

LEGO Mindstorms Education NXT には、3 つのサーボモータ (エンコーダ内蔵) とタッチセンサ(ロボットの触角を与える)、光センサ(明暗を感じる)、サウンドセンサ(音圧を計測)、超音波センサ(物体との距離を感じる)が付属している。



図 1: LEGO Mindstorms Education NXT (拡張セット)



図 2: NXT



図 3: 左から、サーボモータ、タッチセンサ、光センサ、サウンドセンサ、超音波センサ

この NXT, センサ, モータ, LEGO ブロックを使用して、例えば以下のような機能を備えた LEGO を創ることができる。

- ・前進, 後進, 左右にまがるなど, 移動ができる
- ・コースを認識する
- ・障害物を認識する

本稿では、ロボットのリアルタイム制御の習得を目的として、LEGO を動かすための NXC(MindStorms 用言語:Not eXactly C の略)を用いたプログラミングの基礎について述べる。

## 2 制御プログラムの作成, 制御プログラムの実行

図4に示すようなLEGOによるロボットが完成したら, つぎはNXTでの制御を考える必要がある。NXTで制御するためのプログラミングにはMindStorms用言語であるNXCを用いる。

NXCでのプログラミングには, BricxCC ( Bricx Command Center の略) を利用する。

また, NXTとPCの通信を行うにはPCにドライバをインストールする必要がある。本節では, BricxCC, ドライバのインストール方法について述べ, さらに制御プログラムの作成方法, 制御プログラムの実行方法について述べる。



図4: サンプル

### 2. 1 BricxCC のインストール

まず, NXCによるプログラミングを可能にするには, BricxCCをインストールする必要がある。インストールの手順を以下に示す。

1. まずインストールのための実行ファイルをダウンロードする。

[http://sourceforge.net/project/downloading.php?groupid=68600&use\\_mirror=nchc&filename=bricxcc\\_setup\\_33718.exe&65520166](http://sourceforge.net/project/downloading.php?groupid=68600&use_mirror=nchc&filename=bricxcc_setup_33718.exe&65520166)

にアクセスし, 実行ファイル「bricxcc\_setup\_33718.exe」をダウンロードする。

2. ダウンロードした実行ファイルをダブルクリックして, BricxCCをインストールする。

また, PCとNXTを通信するには, PCにドライバをインストールする必要がある。インストールの手順を以下に示す。

### 3 NXCでの基本プログラミング

ここではNXC(MindStorms 用言語: Not eXactly C の略) での基本的なプログラミングについて述べる。まず、NXC でのプログラムでは、必ず“task main()” を含み以下のような書き方をする。

```
task main()
{
  コマンド1;
  コマンド2;
}
```

コマンドは後ほど紹介する。BricxCC では大文字小文字の区別があるので注意が必要である。また、BricxCC にはすでに登録されているコマンドがいくつかある。登録されているコマンドは特別な意味をもち、BricxCC 上では色が変わるのでプログラムの際に参考にするとうい。

### 4 モータ制御用コマンド

LEGO を動かすためには、モータを駆動させなければならない。そのため、ここではモータを制御するためのコマンドについて説明する。

**Of**(引数 1);

引数 1 に書かれたモータを止める

(引数 1 : OUT\_A, OUT\_B, OUT\_C)

**Fbat**(引数 1);

引数 1 に書かれたモータへの On 状態を解除する

(引数 1 : OUT\_A, OUT\_B, OUT\_C)

**OnFwd**(引数 1, 数字);

引数 1 に書かれたモータを前進に設定して動かす。

回転のパワーは数字(0~100) で指定する。

ただし、-100~0 に指定した場合は回転の方向が反転する。

(引数 1 : OUT\_A, OUT\_B, OUT\_C)

(数字 : 0[min]~100[max])

OnRev(引数 1, 数字);

引数 1 に書かれたモータを後進に設定して動かす。

回転のパワーは数字(0~100) で指定する。

ただし, -100~0 に指定した場合は回転の方向が反転する。

(引数 1 : OUT\_A, OUT\_B, OUT\_C)

(数字 : 0[min]~100[max])

☆ワンポイント: 引数は OnFwd(OUT\_AB) という書き方ができる。

以上が, モータ制御用のコマンドである。つぎに, 例題を通して実際にここで紹介したコマンドを組み合わせて使用してみる。なお, この例題はモータを A と C に取り付けた LEGO を想定している。モータを 1 つだけ使用した場合や他の場所に取り付けた場合は, 作成した LEGO に合わせてプログラムを修正していただきたい。

【例題 1】 4 秒間前進して, 4 秒間後退するプログラムを作成せよ

【解答】 Ex1-1, -2, -3 は, いずれも同じ動作をする

|   |               |   |               |   |               |
|---|---------------|---|---------------|---|---------------|
| -----   | <i>Ex.1-1</i> | -----   | <i>Ex.1-2</i> | -----   | <i>Ex.1-3</i> |
| <pre>task main() {   OnFwd(OUT_A, 40);   OnFwd(OUT_C, 40);   Wait(4000);   OnRev(OUT_A, 40);   OnRev(OUT_C, 40);   Wait(4000);   Off(OUT_AC); } }</pre> |               | <pre>task main() {   OnFwdSync(OUT_AC, 40, 40);   Wait(4000);   OnRevSync(OUT_AC, 40, 40);   Wait(4000);   Off(OUT_AC); } }</pre> |               | <pre>task main() {   OnFwd(OUT_AC, 40);   Wait(4000);   OnRev(OUT_AC, 40);   Wait(4000);   Off(OUT_AC); } }</pre> |               |
| -----   | <i>Ex.1-1</i> | -----   | <i>Ex.1-2</i> | -----   | <i>Ex.1-3</i> |

## 5 おわりに

本稿では, BricxCC の使用方法と NXC を用いた LEGO Mindstorms のプログラミングについて簡単に説明した。より詳しい情報は, 参考文献や Web にもあるので必要に応じて各自で調べていただきたい。

## 6 謝辞

無事研究を終えることが出来ました。研究をするにあたり技術指導等で大変お世話になった武田先輩と卒業研究の合間に指導して下さった吉田先輩, また滑川先生や研究室の方々, 研究を通して私を支えて下さった皆様に最大の感謝を表します。